Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Multimodal Social Cue Integration for Attention Modeling and Robot Gaze Control

Dissertation

submitted to

Universität Hamburg,
Faculty of Mathematics, Informatics and Natural Sciences,
Department of Informatics,

in partial fulfillment of the requirements for the degree of

Doctor rerum naturalium
(Dr. rer. nat.)

by

Fares Abawi

Hamburg, 2024

Day of submission:
27.09.2024

Day of oral defense:
16.12.2024

Dissertation Committee:

Prof. Dr. Stefan Wermter (reviewer, advisor)
Department of Informatics
Universität Hamburg, Germany

Prof. Dr. Frank Steinicke (reviewer, deputy chair)
Department of Informatics
Universität Hamburg, Germany

Prof. Dr.-Ing. Timo Gerkmann (chair)
Department of Informatics
Universität Hamburg, Germany

Abstract

Cognitive modeling is the creation of models of human behavior that can also be used to inform the development of intelligent robots. A common cognitive modeling task is saliency prediction. Saliency models predict regions in an image or video where a group of observers are most likely to gaze. Existing work on saliency models formulates the task as an end-to-end problem, predicting attention as a function of the stimuli. In this thesis, we identify the importance of social cues in directing attention, and therefore, introduce priors representing social cues into our models. A model augmented with social cues is defined as a *social attention* model. We show that the explicit representation of social cues improves the performance of existing saliency models. In contrast to saliency models, scanpath models predict the gaze trajectories of individual observers. We extend saliency models with a fixation history module, transforming them into scanpath prediction models. This transformation is necessary for deploying attention models on robots, especially in Human-Robot Interaction (HRI) settings, as it allows robots to exhibit humanlike gaze patterns rather than infer gaze transitions based on the aggregated attention of a group of observers. Additionally, it allows for the personalization of scanpaths using a single unified model, which in turn reduces the training time significantly, as well as the number of models required to achieve the same objective.

Toward achieving our objective, we begin by evaluating the impact of non-verbal social cues on audiovisual saliency models. We design deep-learning models that integrate these social cues with existing saliency models, thereby improving saliency prediction in social settings. Saliency and social cues are represented as spatiotemporal maps and integrated through neural attention and gating mechanisms. A major advantage of our map representation approach is the ability to replace these maps at inference time without having to retrain or fine-tune the social attention model. We propose two architectures for integrating these maps. The first which we term *late integration*, combines features from multiple modality streams using convolutional Attentive Long Short-Term Memory (ALSTM) units. The resulting feature maps are then propagated to a Gated Multimodal Unit (GMU) model. The second integration architecture, which we term *early fusion*, lets one modality influence another via the GMU, which precedes the ALSTM, while maintaining separate streams for each modality. This allows us to weigh and quantify the impact of each social cue on task performance.

Given that the saliency representation maps closely resemble our social attention model output, there is a potential drawback for shortcut learning to occur. This means that the model might become overly dependent on the most reliable cue, ignoring all others. Thus, to mitigate shortcut learning, we develop a neural attention inversion module, which we term the Directed Attention Module (DAM), based on the Squeeze-and-Excitation network. The DAM predicts the inverse of the social cue and saliency representation maps, thereby uniformly distributing the attention weights among social cue modalities. Therefore, it allows our social attention model to rely on all modality representations rather than those of the most salient modality only.

Furthermore, to investigate the performance of our models under real-world conditions, we develop a software framework called Wrapyfi, which allows us to deploy and distribute the models on multiple machines and robots. Wrapyfi facilitates the distribution of models by introducing a common interface for different message-oriented and robotics middleware. This framework reduces the boilerplate code necessary for conducting robotic experiments by abstracting communication protocols and providing plugins that enable the exchange of many data types, including those defined by deep-learning frameworks. This allows us to focus on the design of our experimental pipelines, rather than the communication protocols between robots and software components. We utilize Wrapyfi to conduct HRI studies exploring the influence of robot social cues, namely gaze direction and facial expressions, on human behavior, collaboration, and perception. Moreover, Wrapyfi is used to manage the communication exchanges for our cognitive robotic simulations. These simulations rely on the embodiment of our social attention models into a physical robotic platform, demonstrating their resilience to sensor noise and their applicability in HRI. We introduce paradigms for quantitatively evaluating these cognitive simulations, allowing us to scale up the assessment of our models' performance on robots, without requiring human feedback.

Realizing the impact of sound on attention and gaze, we extend an existing audiovisual saliency prediction model with an additional auditory stream, effectively transforming it into a binaural model. This enables the model to localize sound in videos, thus expanding the capabilities of social attention models relying on its representation maps. Additionally, given that the attention patterns of individual observers are distinct from those of a group of observers, we extend our social attention saliency model into a scanpath predictor by integrating a fixation history module. Finally, the model is validated in a cognitive robotic simulation setup, allowing us to compare the robot's performance to that of humans.

Zusammenfassung

Cognitive Modellierung dient dazu, Modelle zum Verhalten von Menschen zu erstellen, sowie die Entwicklung intelligenter Roboter zu unterstützen. Eine häufige Aufgabe der kognitiven Modellierung ist die Salienzvorhersage. Salienzmodelle sagen voraus, wenn Beobachter ein Bild oder Video anschauen, auf welche Bereiche sie als Gruppe am ehesten schauen werden. Bestehende Forschungsarbeiten zu Salienzmodellen formulieren die Salienzvorhersage als ein Ende-zu-Ende Problem, bei dem die Aufmerksamkeit der Beobachter als Funktion der Eingabestimuli vorhergesagt wird. In dieser Dissertation untersuchen wir die Bedeutung sozialer Hinweise zur Lenkung der Aufmerksamkeit und führen daher A-Priori Faktoren, die durch soziale Hinweise repräsentiert werden, in unsere Modelle ein. Ein Modell, das mit sozialen Hinweisen erweitert wird, definieren wir *soziales Aufmerksamkeitsmodell*. Wir zeigen, dass die Einbeziehung sozialer Hinweise die Leistung bestehender Salienzmodelle verbessert. Im Gegensatz zu Salienzmodellen sagen Scanpfadmodelle die Blickverläufe einzelner Beobachter voraus. Wir erweitern Salienzmodelle mit einem Modul, das die Fixationshistorie einbezieht, und verwandeln sie in Scanpathmodelle. Diese Transformation ist notwendig, um Aufmerksamkeitsmodelle in Robotern einzusetzen, insbesondere im Kontext von Mensch-Roboter Interaktion (Human-Robot Interaction, HRI), da sie es Robotern ermöglicht, individuelle menschenähnliche Blickmuster zu generieren, anstatt Blickübergänge basierend auf der aggregierten Gruppenaufmerksamkeit abzuleiten. Darüber hinaus erleichtert es die Personalisierung von Scanpfaden mit einem einzigen vereinheitlichten Modell, was wiederum die Trainingszeit und die Anzahl der spezifischen Modelle erheblich reduziert, die erforderlich wären, um dasselbe Ziel zu erreichen.

Um unser Ziel zu erreichen, beginnen wir mit der Bewertung des Einflusses nonverbaler sozialer Hinweise auf audiovisuelle Salienzmodelle. Wir entwickeln Deep-Learning Ansätze, die diese sozialen Hinweise in bestehende Salienzmodelle integrieren und dadurch deren Performanz in sozialen Umgebungen verbessern. Die Salienz und die sozialen Hinweise werden als raumzeitliche Karten dargestellt und durch neuronale Attention- und Gating-Mechanismen integriert. Ein großer Vorteil unserer Kartenrepräsentation ist die Möglichkeit, diese Karten zur Inferenzzeit austauschen zu können, ohne das soziale Aufmerksamkeitsmodell neu trainieren oder feinabstimmen zu müssen. Wir schlagen zwei Architekturen zur Integration dieser Karten vor. Die erste, die wir *Late Integration* nennen, kombiniert Merkmale aus mehreren Modalitäten unter Verwendung des convolutional Attentive-LSTM (ALSTM) Modells. Die resultierenden Merkmalskarten werden dann auf ein Gated Multimodal Unit (GMU) Modell übertragen. In der zweiten Integrationsarchitektur, die wir *Early Fusion* nennen, moduliert eine Modalität eine andere, indem das GMU den ALSTM-Units vorausgeht, wobei die Modalitäten separiert bleiben. Dies ermöglicht es, jeden sozialen Hinweis zu gewichten und dessen Einfluss auf die Modellperformanz zu ermitteln.

Da die Salienzrepräsentationskarten den Ausgaben unseres sozialen Aufmerksamkeitsmodells ähneln, kann als Nachteil Shortcut Learning auftreten. Das heißt, das Modell verlässt sich nur auf den zuverlässigsten Hinweis und ignoriert alle

anderen Hinweise. Daher entwickeln wir zur Reduzierung von Shortcut Learning ein neuronales Aufmerksamkeitsinversionsmodul, das wir als Directed Attention Module (DAM) bezeichnen, basierend auf dem Squeeze-and-Excitation Netzwerk. Das DAM sagt das Inverse der sozialen Hinweis- und Salienzrepräsentationskarten vorher und verteilt somit die Aufmerksamkeitsgewichte gleichmäßig auf die sozialen Hinweise. Somit ermöglicht es unserem sozialen Aufmerksamkeitsmodell, sich auf alle Modalitätsdarstellungen zu stützen und nicht nur auf die einer Modalität.

Darüber hinaus entwickeln wir, um die Leistung unserer Modelle unter realen Bedingungen zu untersuchen, ein Software-Framework namens Wrapyfi, das es uns ermöglicht, die Modelle auf mehreren Computern und Robotern zu implementieren und zu verteilen. Wrapyfi erleichtert die Verteilung von Modellen, indem es eine gemeinsame Schnittstelle für verschiedene nachrichtenorientierte und robotische Middleware bereitstellt. Dieses Framework reduziert die Codebausteine, die für die Durchführung robotischer Experimente notwendig sind, indem es Kommunikationsprotokolle abstrahiert und Plugins bereitstellt, die den Austausch vieler Datentypen, einschließlich derjenigen, die von Deep-Learning Frameworks definiert werden, ermöglichen. Damit können wir uns auf das Design unserer experimentellen Pipelines konzentrieren, anstatt auf die Kommunikationsprotokolle zwischen Robotern und Softwarekomponenten. Wir nutzen Wrapyfi zur Durchführung von HRI-Studien, die den Einfluss robotischer sozialer Hinweise, nämlich Blickverhalten und Gesichtsausdrücke, auf menschliches Verhalten, Zusammenarbeit und Wahrnehmung untersuchen. Darüber hinaus wird Wrapyfi verwendet, um den Kommunikationsaustausch für unsere kognitiven robotischen Simulationen zu verwalten. Diese Simulationen basieren auf der Einbettung unserer sozialen Aufmerksamkeitsmodelle in eine physische robotische Plattform und demonstrieren deren Robustheit gegenüber Sensorrauschen und ihre Anwendbarkeit in HRI. Wir führen Paradigmen ein, um diese kognitiven Simulationen quantitativ zu bewerten und ermöglichen so die Skalierung der Leistungsbewertung unserer Modelle auf Robotern, ohne menschliches Feedback zu erfordern.

In Anbetracht des Einflusses von Geräuschen auf unsere Aufmerksamkeit und unsere Blickrichtung erweitern wir ein bestehendes audiovisuelles Salienzmodell um einen zusätzlichen auditiven Stream und verwandeln es effektiv in ein binaurales Modell. Dies ermöglicht es dem Modell, Geräusche in Videos zu lokalisieren, und erweitert so die Fähigkeiten sozialer Aufmerksamkeitsmodelle, die sich auf seine Repräsentationskarten stützen. Da die Aufmerksamkeitsmuster einzelner Beobachter sich von denen einer Gruppe von Beobachtern unterscheiden, erweitern wir unser soziales Aufmerksamkeits-Salienzmodell zu einem Scanpathmodell mittels eines Fixationshistorienmoduls. Das Modell wird schließlich in Experimenten mit Probanden und mit dem Roboter in einer Cognitiven Simulation validiert.

Contents

Chapter 1

Introduction

Upon entering a social environment, it is a common social behavior to observe the directional focus of individuals' gazes and to interpret their interpersonal reactions in general. These reactions provide *social cues* (defined in Section 2.1.3) that are essential tools for non-verbal communication. Social cues include gestures, like *gaze* (defined in Section 2.1.3) direction, and facial expressions. Understanding these cues is necessary for communication among humans. For robots to interact similarly with humans, they need to process and respond to these social signals as well. In this thesis, we aim to improve Human-Robot Interaction (HRI) by developing *social attention* (defined in Section 2.1.3) models, that can integrate social cues and direct robots to exhibit humanlike gaze. By endowing robots with the ability to detect and react to social cues, they can adapt to social settings, making them less intervenient and more relatable.

1.1 Motivation

We are exposed to social cues since birth. Our behaviors, decisions, and emotional states are defined by the interactions that are made possible by the exchange of such cues. The human species has evolved eyes to have a pupil and a sclera—a white fibrous layer surrounding the significantly darker pupil. The contrast between the pupil and the sclera makes the direction of a person's gaze visible from a distance. This enables us to perceive their gaze, which serves many purposes. One of these is non-verbal communication, allowing us to express intentions such as alerting others to danger, or expressing emotions such as joy, without speaking. Thus, gaze as a cue is essential for both communication and social cohesion.

The majority of existing computational models of *cognitive attention* (defined in Section 2.1.3) disregard the influence of social cues. Implicit features relating to such cues could be represented in cognitive attention models, however, interpreting those features could be challenging. For instance, at which neural layer would such features be observable? Should we consider auxiliary social cue representations as low-level features—edges, rotations, intensities—or high-level features—faces, proto-objects, gaze direction? The ambiguity in defining how the auxiliary social

cue features can be represented leads us to adopt a different approach. Rather than observing whether cognitive attention models are capable of representing social cue features, we explicitly represent those cues as spatiotemporal maps instead. We split our models into two separate stages. In one stage, we detect social cues from images and represent them as visual features. In the second stage, we augment a cognitive attention model with those auxiliary features. Our two-stage approach serves the purpose of allowing for feature interpretability (which social cue contributes to the cognitive attention model's output?), improving the model performance (does the explicit representation of social cues improve the model performance?), and modularity (can the social cue representations be replaced without retraining the model?)

Moreover, allowing robots to exhibit humanlike gaze patterns, motivated by the fact that gaze and social cues are fundamental elements of human communication and interaction, is what we aim to tackle. We envision a future where robots could be viewed as assistants, collaborators, and relatable social entities. This vision for robotics is not a reflection of the current state, but a compelling argument for why we should integrate social traits into robots' behaviors. In addition to the interactive elements brought to robots through social cognition, integrating humanlike gaze and other social behaviors into robots brings potential therapeutic value to individuals with neurodevelopmental disorders [29].

We address the challenge of enabling social robots to mimic human gaze by first predicting and modeling *social attention* [223]. Social attention is an umbrella term that describes non-verbal social communication—joint attention, gaze direction, facial expressions—and its effect on attentional preferences, expressed in the form of gaze and emotional expressions. By designing computational models capable of integrating information arriving from different non-verbal social cues, we can understand the effect of those cues on the attention of a group of observers. The task of statistically representing the attention patterns of a group of observers is known as *saliency prediction* (defined in Section 2.1.3). However, predicting the attention of the group does not lend itself to being capable of assimilating human gaze patterns. For one, individual gaze patterns differ from the group [126]. Moreover, our eye fixations (defined in Section 2.1.3) are sequential, meaning that the point in the space where we gaze affects subsequent fixations. The task of sequentially inferring fixations is known as *scanpath prediction* (defined in Section 2.1.3). We therefore adapt our social attention models to predict individual scanpaths. This adaptation allows us to infer scanpaths that are aligned to those of humans, which consequently elevates the naturalness of a robot's gaze. Social attention, however, is not only driven by non-verbal cues. Sounds capture our attention and guide our gaze. This is evident from the fact that humans tend to direct their attention more toward active speakers in social settings [272]. Thus, all our attention and gaze prediction models are audiovisual, meaning that they rely on auditory features in addition to visual features to accommodate such social interactions e.g., the active speaker vocalizes the speech (auditory) while simultaneously moving their lips (visual).

1.2 Research Overview

Understanding the behavior of others as well as objects in our surroundings is critical for interacting with and reacting to changes in our environment. However, the large influx of stimuli deters us from acting responsively without a mechanism to filter out irrelevant information. This mechanism is termed *selective attention* in the field of cognitive psychology. Computational neural models have taken inspiration from attentive selectivity, applying it to focus on relevant attributes across the activations of an entire neural layer, limiting the attention to a local scope of non-differentiable segments, or a combination of both [161]. Such advances contribute a linkage between the two fields—cognitive psychology and computer science—enabling the transferability of discoveries in human studies to algorithmic implementations. Attention in artificial neural networks is especially important for crossmodal systems, where multiple modalities influence the outcomes of one another. For instance, when listening to a person speaking in a noisy environment, we rely on auditory and visual stimuli for selectively attending to that person and filtering out the noise. This phenomenon is known as the cocktail party effect and is computationally modeled for the purpose of speech separation [248].

In previous work [3, 135], we constructed a deep-learning model that combines visual and textual data for end-to-end visuomotor robot grasping. Although the model relies on the Transformer [260] network, which employs self-attention for modeling language, the integration of the two modalities is simply performed through concatenation, i.e., attention is not utilized in selecting the most relevant features across modalities. When addressing social situations, the sporadicity and immensity of perceived stimuli are high, requiring an attention mechanism for fusing multiple modalities. Arevalo *et al.* [17] propose a Gated Multimodal Unit (GMU) for learning joint representations across different modalities through an intermediate fusion approach. The model has several advantages, including independence from the training task and the weighted combination of multiple modalities. However, it was intentionally designed exclusive of attention mechanisms to maintain agnosticism to the task. Nonetheless, employing attention is shown to improve neural model performance [51]. Hence, we explore combining attention mechanisms with the GMU, hereafter termed the Attentive GMU (AGMU).

One form of action exhibited in response to social stimuli is gaze. Our direction of gaze is controlled by the ability to focus our attention, to follow the attention of others, and the salience of objects in our receptive field. Typical and healthy humans are capable of following the gaze of others effortlessly. Estimating the direction of others' gaze informs us on their intention, acting as a direct and deictic form of communication [231]. Studies have shown that innate interests, goals, prior knowledge, and social cues [208, p. 212] including the direction of the observed individual's gaze, can affect our ability to perceive their target of attention [231], which in turn influences our own social attention.

In this thesis, we address the gap between robotic applications and psychological findings that emphasize the importance of integrating social cues in predicting and controlling gaze. Gaze prediction refers to the estimation of a location where an

agent is to fixate, whereas gaze control refers to the actuation of an agent's eyes and head, along with the strategies involved in directing such an action. We develop deep-learning models that combine features from multiple sensory and social cue modalities by:

1. Representing saliency and social cues as spatiotemporal maps.
2. Attending to the most relevant map features using AGMUs, resulting in a *joint representation* [22].
3. Decoding the joint representation using a model trained on a *downstream task* (defined in Section 2.1.2).

Input to the proposed models is presented in the form of auditory and visual data. These model and their variants, which we term *GASP*, rely on two stages, namely a social cue detection stage followed by a social cue integration stage. In the first stage, the social cue modules extract social cue representations by transforming the detected cues into spatiotemporal maps. In the second stage, the representations act as auxiliary features and are integrated using AGMUs and other hybrid fusion approaches, resulting in a joint representation. The joint representation is then propagated to a downstream model, which is then used to guide a gaze prediction model.

To control the gaze and social cues of a robotic agent, we first develop a framework called *Wrapyfi*, which acts as a bridge between different variants of GASP models and robotic platforms. Wrapyfi simplifies the bridging between deep-learning models, sensors, and actuators by providing a single interface for multiple middleware. Based on Wrapyfi, we conduct two HRI studies to understand the effect of robot gaze and social cues on humans interacting with social robots.

In the first HRI study, we present a triadic gameplay scenario, where two human participants play a cooperative tabletop game, while the robot displays facial expressions and establishes mutual or averted gaze with one of the players. The player conducting gaze interaction with the robot decides whether the robot is performing random or meaningful gaze shifts and facial expressions based on their perception of the robot's gaze patterns. The purpose of this study is to establish whether a robot engaging with humans through social cueing would attract their attention and affect their performance while performing collaborative tasks.

In the second HRI study, we focus on assessing the effect of using different interfaces for mirroring the gaze and facial expressions of humans on a robot. Moreover, we evaluate the responsiveness of models, sensors, and display interfaces communicating through Wrapyfi. Such evaluation helps us understand the limits of robots when conducting HRI studies, including a robot's physical (mechanical) capabilities and the communication latency between inferring social cues and controlling the robot. This is especially important for enabling realistic gaze shifts since both the communication latency and the mechanical limitations of the robot affect the accuracy of gaze prediction.

Next, to assess the performance of our gaze prediction models in physical settings, we use Wrapyfi to run these models on a robotic platform. We conduct

two cognitive simulation studies on a physical iCub [171] robot. The gaze prediction downstream model and evaluation methodology differ among the two. The goal of these studies is to evaluate the robustness of our models to sensor noise and their effectiveness in near-real-time settings. One study highlights the importance of auditory cues in directing attention. However, the model used in this study does not predict humanlike gaze patterns, since it is trained on the attention maps representing a group of observers. This makes it unsuitable for HRI studies, since gazing toward the peak of the attention map results in eye movement shifts that are unnatural. We, therefore, conduct a second cognitive simulation study with a similar setting but a different model that predicts personalized scanpaths. This model accounts for differences in gaze patterns and as a result, reduces the likelihood of eye movements that diverge from typical gaze patterns.

In the first cognitive simulation study, we compare human and robot attention responses to conflicting visual and auditory cues, with the visual cue being a gaze toward a direction, and the auditory cue being a speech utterance arriving from the same or opposing location to the gaze target. This study requires sound localization capabilities. However, two audio sources are needed to localize sound. This is similar in humans, who are able to localize sound by perceiving auditory stimuli through both ears. Therefore, we extend an audiovisual saliency prediction model with an additional auditory stream, transforming it into a binaural audiovisual sound localizer. The task for this study—localize sound regardless of the visual cue—is goal-directed, meaning that the similarities across humans are higher than they are when the goal is loosely defined, such as under the free-viewing [257, p. 26] condition. Therefore, we use a downstream social attention model, trained for predicting saliency. Attending to the most salient region as predicted by the model would correspond to the gaze prediction used to control the robot. We replace the saliency prediction model that feeds into the social attention model with a binaural audiovisual sound localizer.

In the second cognitive simulation study, we evaluate the similarity between individual human scanpaths and a robot's gaze prediction. For this purpose, we set the downstream task of our social attention model to scanpath prediction instead. We introduce a fixation history (defined in Section 2.1.3) module that encodes the preceding fixations for each stimuli observation per individual. This allows us to distinguish the target gaze pattern during training and inference. By changing the task to scanpath prediction, we can personalize gaze patterns and predict sequences, making the comparison with different individuals under the free-viewing condition possible. We devise a mechanism for projecting the ground-truth *priority map* (defined in Section 2.1.3) to a monitor (screen) within a simulated environment. We then match the positions of the predicted and ground-truth priority maps and compare them using common saliency metrics [43]. The predicted map is the output of the scanpath model, receiving input from the physical robot's sensors, whereas the ground-truth map is reprojected in simulation.

1.3 Objectives and Research Questions

Our objectives in this thesis focus on exploring methods for multimodal integration and fusion, employing attentive mechanisms in selecting the most relevant information, utilizing social cues for directing the gaze of a robotic agent, and personalizing robot gaze models—models for controlling the gaze direction of the robot. We aim to address the following research questions:

RQ1.1 *Does integrating social cues, like gaze direction and facial expressions, with saliency models improve the models' performances?*

Evidence shows that head-related non-verbal social cues, such as the gaze direction and facial expressions, as well as bottom-up and top-down saliency, guide overt attention—gaze in the form of head and eye movements—in social interactions. We investigate this question first by addressing the learning of saliency. We integrate social cues into a deep-learning-based saliency model and evaluate whether their integration contributes to an improvement in performance. By ablating the representation modules and quantifying the weight gain for each, we are able to assess the benefit of each social cue independently.

RQ1.2 *How can non-verbal social cues be integrated into social attention models?*

We detect and represent social cues as spatiotemporal maps that are then integrated into our social attention model through early fusion and late integration. Our model relies on audiovisual saliency representations alongside social cues. We explore different integration mechanisms to combine those representations when modeling static and dynamic stimuli.

RQ1.3 *How can social attention models be personalized?*

Group saliency models indicate the most conspicuous regions in a scene. However, looking toward such regions does not resemble natural human gaze, nor does it account for differences in gaze patterns. To address this limitation, we extend our social attention model with a fixation history module. The fixation history accounts for the sequence of preceding fixation points for each observer separately when viewing a scene. In other words, the fixation history represents the scanpath of an observer prior to them viewing the current scene, allowing us to specify which observer's gaze we would like to assimilate.

RQ1.4 *Which methods are needed to embody social attention models in robots?*

To embody our social attention models in robots, we develop the Wrapyfi framework for communicating information to the actuators and from the sensors of a robot. Along with pipelines to concurrently process and acquire

stimuli, we transport audio and video to our social attention models, eventually directing the robot's gaze toward the target of attention. The framework is designed to support multiple message-oriented middleware and distribute mirrored copies of scripts across machines, thereby allowing us to run computationally demanding social cue and attention models in near-real-time settings.

RQ1.5 *How can we assess the performance of a physical robotic gaze implementation?*

One method for evaluating robotic gaze involves conducting HRI studies, followed by an assessment of the participants' reactions and responses to tailored questionnaires. In this work, we conduct HRI studies to evaluate human perception of robots displaying social cues, in the form of gaze and affect. Moreover, we devise methods to quantitatively evaluate a robot's performance under real-world conditions, by simulating studies conducted on humans.

1.4 Novelties and Contributions

1. We introduce a novel approach for integrating social cues into social attention models. Our approach relies on visual representations of social cues. Unlike common approaches where latent neural representations or engineered features are integrated into downstream models, our social cue features are interpretable spatiotemporal representations that are consistent in shape. This makes our features easily interchangeable with those of other models, without needing to retrain the downstream model.

2. We devise two variants of neural attention mechanisms to integrate dynamic (sequential audiovisual) input. Late integration (defined in Section 2.1.2) refers to the combination of features from multiple streams using convolutional attentive LSTM [63] units followed by the Gated Multimodal Unit [17]. Alternatively, the early fusion (defined in Section 2.1.2) variant reverses the gating and attention operation order. This variant retains separable representations for each stream, allowing for the examination of each stream's contribution to the downstream task's performance.

3. We develop a neural attention inversion module based on the Squeeze-and-Excitation [118] network making it possible to enhance neural models by augmenting them with other representations while avoiding shortcut learning [95].

4. We discover that the reliance on fixation history—a sequence of previous scanpaths for an observer—as an input feature enables the learning of personalized scanpaths using a single unified model. This approach obviates the need to train separate models for each observer. Moreover, the simplicity

of the approach in comparison to relatively more complex personalization approaches, such as user embedding learning to represent individuals [160, 239], makes fixation history integration a suitable solution for training a single model that can be personalized.

5. We design a software framework to simplify message-oriented and robotics middleware communication. The framework adopts a non-opinionated design, introducing three communication schemes for implementing experimental pipelines. Moreover, the framework provides plugins to support deep-learning data-type exchanges across multiple middleware.

1.5 Dissertation Outline

This thesis is split into two parts. In Part I, we present our social attention models. Part II addresses robotic implementations and the experiments we conduct to evaluate social attention models, embodied in physical robots. Figure 1.1 illustrates the relation between the chapters and the components used to facilitate each study.

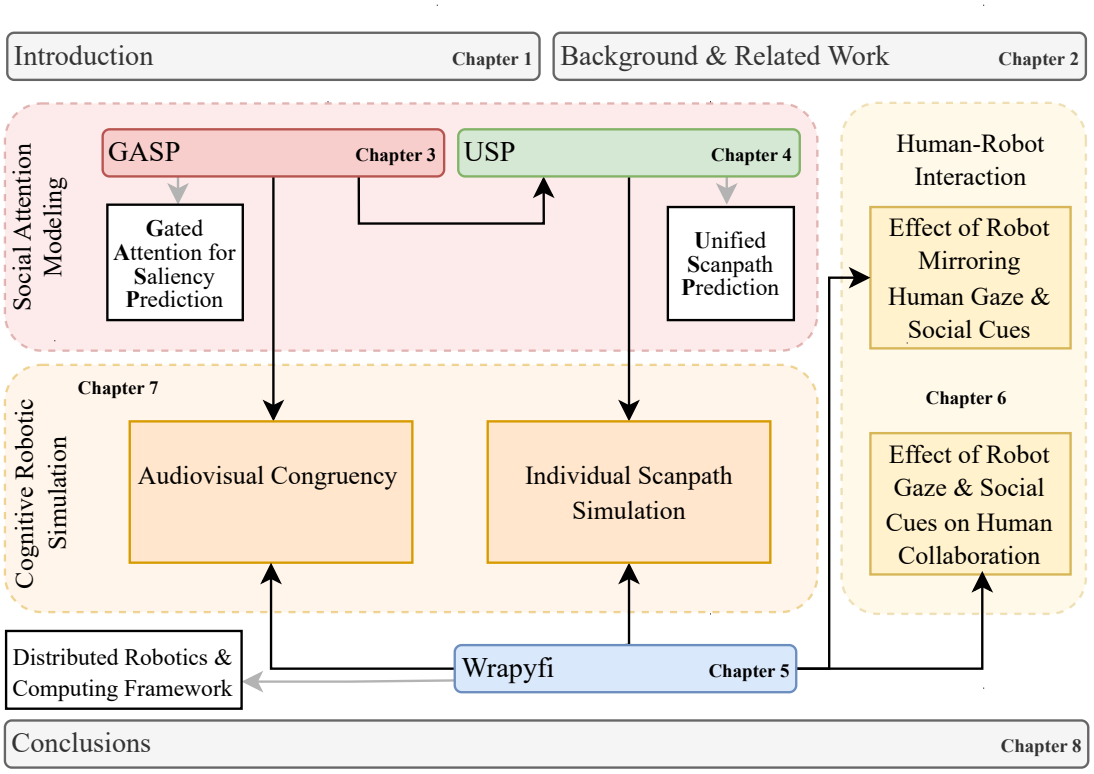


Figure 1.1: Thesis outline.

- In Chapter 1 we motivate our work, provide an overview of the methodologies, highlight novelties, and frame the outline of this thesis.

- In Chapter 2, we summarize the background and related work.
- Part I: Modeling Social Attention
  - In Chapter 3, we explore different means of integrating social cues, with the goal of enhancing dynamic saliency prediction in social settings [6].
  - In Chapter 4 we extend the social attention model developed in Chapter 3 by including a fixation history module, to predict scanpaths [5]. The fixation history allows our model to personalize scanpaths.
- Part II: Controlling Social Robots
  - In Chapter 5 we develop Wrapyfi, a message-oriented and robotics middleware framework [4]. The framework provides plugins to exchange data types across robot and deep-learning frameworks. Moreover, it simplifies parallelized computation and abstracts the communication boilerplate.
  - In Chapter 6 we present two studies to demonstrate the practicality and means of using the framework developed in Chapter 5. Additionally, we study human perception of robots displaying social cues during HRI:
    1. We conduct a study to assess whether the social cue expressions of a robot affect human interaction and collaboration in Section 6.1 [89].
    2. We conduct a study to evaluate the preferred means for robots to convey social cue expressions and capture human social cues in Section 6.2 [91].
  - In Chapter 7, we employ the framework developed in Chapter 5 to deploy models constructed in Chapter 3 and Chapter 4 on the physical iCub robot for HRI-free evaluation:
    1. We construct a binaural audiovisual sound source localization model and propagate its predictions to the model developed in Chapter 3. The model is deployed on the robot to evaluate its performance on a social cueing task in Section 7.1 [88].
    2. We evaluate the robustness of the model developed in Chapter 4 to physical environment noise, by deploying it on the robot with different fixation histories in Section 7.2 [7]. This allows for the personalization of the robot's scanpath trajectories.
- We summarize and discuss the work developed during the course of this thesis in Chapter 8. We additionally answer the research questions posed in relation to the methodologies devised in Part I and Part II of our thesis.

Chapter 2

Background

In this chapter, we describe the concepts that motivate our approaches. This thesis introduces several artificial neural network (deep learning) models for integrating different social cue modalities. As we address computational neural attention and cognitive attention, we clarify the terminology surrounding those two concepts in their respective domain. We also provide an overview of the existing literature, highlighting our novelties and distinguishing our technical contributions from the established body of work.

2.1 Definitions and Taxonomy

In this thesis, some terms may have different meanings depending on the context and field of study. To avoid confusion, we provide an overview of these related concepts, explaining each term and specifying how we use it throughout our discussions.

2.1.1 Gating and Attention in Neural Networks

Extracting information from sequences of input, whether they are text, video, audio, or other sensory data, requires models that encode order, i.e., representing knowledge while considering the order in which the information appears. Seminal approaches in modeling sequential information using artificial neural networks include the Recurrent Neural Network (RNN) [221], the Recursive Neural Network [100], the Transformer [260], and more recently, structured state-space model [102]. For over two decades, RNNs were the predominant architectures for modeling sequences, following the invention of the Long Short-Term Memory (LSTM) [115]. The LSTM mitigated the *vanishing gradient problem* [27], which was considered a major hurdle for the adoption of RNNs in practice. The introduction of *gating* with LSTMs allows for the selective propagation of activation, effectively bounding the outputs of the recurrent units, and in turn, reducing the potential for gradients vanishing.

Gating Mechanism

Gating not only mitigates the potential for vanishing gradients and enables the association of dependencies across long sequences [115, 57], but also facilitates selective propagation of neural activity [17]. A gate, as the name implies, allows or disallows things to pass through. In its most simple form, a gate could be likened to a function that requires certain conditions to be fulfilled, and results in a boolean operation. When the condition is completely met, the input element is multiplied by 1 and returned as itself, otherwise, it is multiplied by 0 resulting in no returns. In terms of artificial neural networks, gating relies on sigmoid-like functions, which are continuous and smooth step functions that are differentiable—differentiability is a prerequisite for backpropagation, which is the standard learning rule for the majority of artificial neural network models [227].

Soft and Hard Attention Mechanisms

Similar to gating, *attention* in artificial neural networks refers to mechanisms embedded into deep learning models for weighting[1] the interactions between units based on learning the strength of relations between correlated events. In deep learning, attention could be applied to a neural architecture, allowing for the account of dependencies that span a longer range than most recurrent models are capable of handling. Two overarching types of attention in deep learning are: Hard attention [178], where a discrete non-differentiable mask is sampled stochastically and applied to a region of interest; Soft attention [19], which models a continuous differentiable distribution representing regions of high relevance.

Hard attention, being non-differentiable, relies on alternative training approaches that allow its integration into neural architectures, such as reinforcement learning [178]. Soft attention is more prevalent in deep learning since it is learned as part of a model's parameter optimization process. Soft attention requires a matrix of learnable parameters which are multiplied with the outputs of a latent layer in the model, followed by a softmax operation applied to their product. The softmax is a differentiable function, resulting in a probability distribution, thus enabling backpropagation to the attention matrix parameters. Moreover, soft attention results in a continuous decision on relevance, meaning that multiple regions in the latent representation could be focused upon, unlike hard attention. In this thesis, we generally refer to soft attention when describing neural attention or attention in the context of deep learning architecture design.

[1]Throughout this thesis, we refrain from using the term "weights" to describe the learnable parameters of a model. This is to avoid confusion with weighting, a term we use to describe the scale of contribution different neural units have on each other. In the context of this thesis, examining the scale of the weights allows us to quantify the contribution of different neural layers or units, when attention or gating is applied to their representations post-activation.

2.1.2 Multimodal Integration in Neural Networks

Sensory modalities in biological systems indicate receptors perceiving stimuli across different mediums, such as vision, audition, touch, etc. However, multimodality in neural networks is loosely defined with varying degrees of interpretation [196]. In terms of computational modeling, a *human-centered* definition of multimodality refers to a model's ability to acquire input from multiple sources across media perceived by humans. Models that fall under this category are those that can integrate audiovisual [245, 256], visuotactile [79, 55], or a combination of multi-sensory [244] information. A *machine-centered* perspective views multimodality in reference to different encodings of representations, that carry distinctive information. In this view, models that integrate images and text are considered multimodal, even though both modalities are perceived visually by humans.

Both human-centered and machine-centered views do not cover the full spectrum of multimodality in neural networks. Such is the case with models integrating sensory information that could be encoded visually but perceived across different media [99], or could provide additional context that is represented through the same or multiple media [173]. The latter augments the visual medium with detections from the visual and auditory media. This approach provides additional information to a model in the form of visual representations, which might otherwise be difficult to learn in an end-to-end manner.

To avoid any ambiguities, we instead adopt the *task-relative* [196] definition of multimodality. Task-relative multimodality describes systems that integrate modalities that could arrive from the same sensor or be represented in the same medium, but provide different information for a given task. The term 'modality' in this thesis refers to a visual spatiotemporal representation providing unique information for our downstream models, regardless of its representation medium or acquisition source.

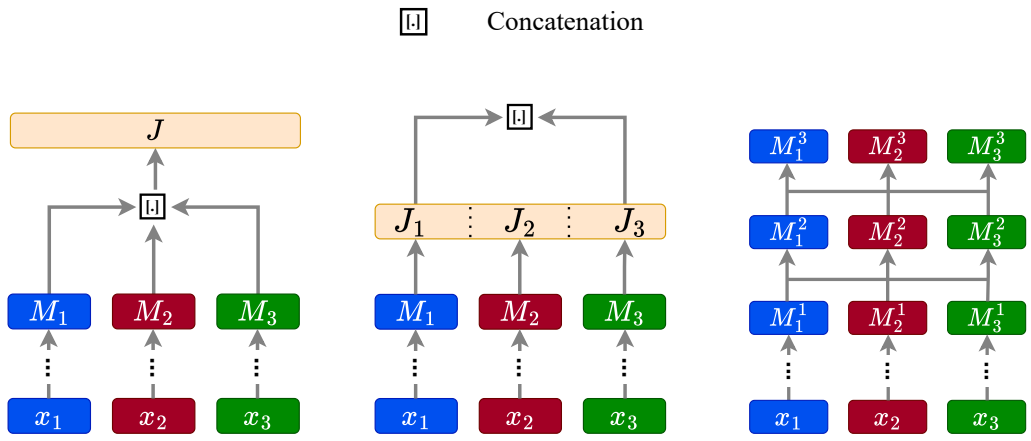
Downstream Model and Task

A *backbone* in deep learning is a foundation model trained on large datasets, covering a wide range of classes or tasks. Backbones represent features that can be generalized to subtasks within a domain and are commonly used for pretraining task-specific models. The backbone model commonly feeds into task-specific models or neural layers for further fine-tuning on narrow sub-tasks within a domain. These task-specific layers are described as *downstream models* optimized for *downstream tasks*. This reduces the computational cost and training time for downstream tasks, while also integrating domain-general knowledge into downstream models.

Backbone models could be trained in supervised, unsupervised, semi-supervised, or self-supervised fashion [290]. Very often, the parameters of a backbone are frozen—parameters remain unchanged during the training phase. The downstream model's parameters are updated during training and fine-tuning. In this thesis, we present downstream models, some of which rely on backbones with unfrozen parameters. This is usually the case when the backbone is treated as a modality encoder [59,

99], with no other layers representing the modality prior to the integration layers.

Late Integration and Early Fusion



(a) Late Integration JR (b) Early Fusion JR (c) Early Fusion CR

JR: Joint Representation; CR: Coordinated Representation;
x_N: Input for modality N; M_N: Modality representation N.

Figure 2.1: Multimodal integration categorized into early fusion and late integration. The **(a)** late integration models concatenate the modality representations before propagating them to a joint representation layer, **(b)** early fusion models gate or attend to each modality representation before concatenating their joint representations and propagating them to a downstream model, and **(c)** early fusion models with coordinated representations, where each modality exerts an influence on other modalities, but their representations remain separate.

The most common form of combining multiple modalities is through the concatenation of their representations, which are then propagated to a fully connected neural layer [117] or convolutional decoder [245]. Including concatenation, other forms of integration, such as additive models [10], belong to the family of *late integration* models, illustrated in Figure 2.1a. We describe such forms of integration as 'late' since they take place after the unimodal (single modality) representations are encoded, and are inseparable post-integration, i.e., late integration models prevent us from measuring the influence of each modality on the downstream task.

Early fusion models in neural networks are not concretely defined, since the term 'early' refers to the layer depth at which integration is commonly implemented, rather than a specific state of representation. Moreover, accounts of early fusion include its description as feature-based integration [22], whereas others [256] use the term to describe *coordinated representation* [22] models as illustrated in Figure 2.1c. We additionally describe early fusion models as those illustrated in Figure 2.1b, that maintain separability between the different unimodal representations, unlike late integration models. Early fusion is possible through gating [17], attention [274], or a combination of both [286]. This is due to the fact that attention models

and most gating approaches introduce additional parameters to the full model, that are optimized conjointly with the unimodal layers, yet do not directly affect the unimodal parameters. This results in the unimodal representations remaining independent, while exerting an influence on other modalities [256, 139].

2.1.3 Cognitive Social Attention and Saliency

Cognitive attention, and more specifically, *visual selective attention* [183], refers to our ability to selectively discard or emphasize visual stimuli. It describes the attentional patterns of humans, conditioned on observable attractors—objects or stimuli that attract human attention. Computational modeling of visual attention has been beneficial for a wide variety of applications, including the enhancement of object recognition systems [80], interaction between robots and humans [203], and the guiding of robotic vision systems [86]. We note that although attention in neural networks (described in Section 2.1.1) is inspired by cognitive attention, neural attention is not modeled to assimilate intrinsic mechanisms of cognitive attention [192].

Head Movement and Eye Gaze

The cognitive attention of a human is visible to others through *overt orienting*, which refers to the head and eye movements of the individual. *Gaze* is the term we use in reference to the overt orientation that leads to the *fixation* [13] of an individual or the majority of group fixations upon a target. Fixation refers to stable eye movements, more specifically, it describes the state of eyes being relatively stationary for an extended period of time. For instance, when observers are looking at a still or slowly moving object, their eyes would still oscillate. These oscillations are called *microsaccades* [107], where the eye would remain fixated upon a target with minimal but rapid movement. *Saccades* on the other hand, are sudden and voluntary movements of the eye that precede fixation.

Most gaze prediction models are trained on eye fixations, rather than head orientations. This is due to datasets used in this domain being collected using eye trackers [138] that generally account for eye movements only. Moreover, the reason fixations are used in such models rather than saccades, is that the latter's signals are rather noisy and require high sampling rates. We do not train any of the social attention models on saccade data in the context of this thesis. Our models are designed to be integrated into robotic platforms, thus, modeling saccades becomes challenging or not possible, when accounting for the physical constraints of the utilized robot.

Saliency: Bottom-Up and Top-Down Attention

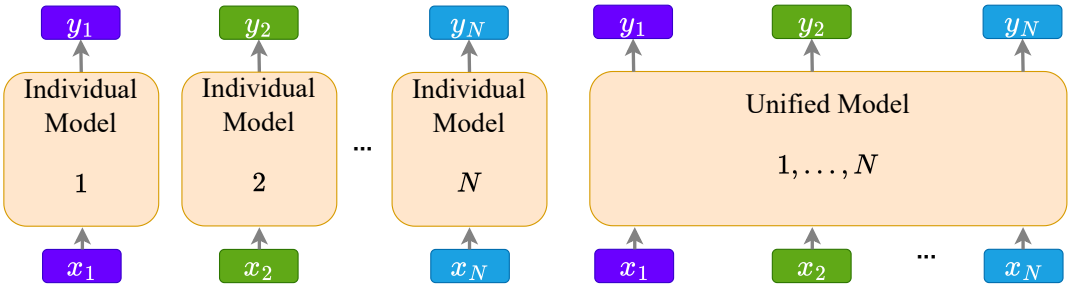
Cognitive attention driven by external stimuli is known as *reflexive attention (bottom-up attention)* [61]. These include bright and luminous attractors, movements, faces, and attractors with distinctive shapes or colors in comparison to

other perceived stimuli in view. We tend to respond immediately to such attractors by looking toward them or reacting with some action, hence the name reflexive attention. Seminal work on the computational modeling of reflexive attention [120] aligns with *feature integration theory* [255]. This work relied on predefined transformations that emphasized *salient* regions, hypothesized to attract attention [120]. These transformations are applied to images following findings from studies in psychophysics, such as extracting color, light intensity, orientations, indicators of motion, shapes of shadows in the image, etc. Models that follow a similar bottom-up structure, where the transformations are engineered based on known attractor features, produce a 2D heatmap. The heatmap called a *saliency map*, indicates salient regions by intensifying the magnitude around those regions.

To simulate human gaze, further transformations are applied to the saliency maps inferred by bottom-up *saliency detection models*. One such is the *Inhibition of Return* (IoR) [120]. IoR can be roughly simulated by suppressing previously attended salient regions, creating an approximation for natural gaze patterns. However, not all saliency effects can be approximated, as some do not result from temporal changes or stimuli conspicuity. They are also conditioned on the target task, which could be performed in different ways depending on the observer. Such differences are the result of innate factors, like past experiences or intrinsic motivation that are specific to each observer. These task-related factors are described as *innate attention (top-down attention)* [61]. Computationally, top-down attention and bottom-up attention can be modeled by what are known as *saliency prediction* models [35]. Saliency prediction models receive visual stimuli in the form of *static* images or *dynamic* videos and predict a Fixation Density Map (FDM), that represents group attention under the *free-viewing condition* [257, p. 26]. The FDM, also known as an *attention map*, is a 2D map with the accumulated *fixation points* of every observer viewing an image or a video frame, collected using an eye tracker. The fixation points are blurred by convolving them with 2D Gaussian functions, one centered at each fixation point. On blurring, the fixation points are aggregated in a single map and normalized, resulting in the FDM. The Gaussian function width is equivalent to 1° of viewing angle, that is the area foveated by the eye at a distance from the surface—monitor, screen, projection plane—upon which the stimulus is displayed.

Scanpaths: Personalized and Universal Attention

Gaze patterns are found to be similar in some aspects among healthy humans. Depending on the task and stimulus, these patterns appear to be consistent. For instance, when viewing natural images, humans tend to initially fixate on the center of mass in an image [30]. This phenomenon is known as *central bias* [219]. Age plays a role in altering the prominence of certain biases, such as *pseudoneglect (leftward bias)*. Typically, human attention is biased toward the left side of the visual field, however, attention shifts rightwards with age [237]. The common attributes that apply to a majority of the human population are known as *universal attention* [277]. These attributes alter the gaze behavior of each individual, resulting in sequences of fixation points. These sequences are described as *scanpaths* [190]. Another



(a) Individual Models (b) Unified Model

Figure 2.2: The **(a)** individual models are multiple models with identical structures, trained on data belonging to a single set, distribution, or task, whereas a **(b)** unified model is a single model trained on all tasks, with the task distinguishing token embedded in the input to the model.

universal attribute applies to the distribution of scanpaths. For instance, human scanpaths under the *free-viewing condition*—individual observers are not given a task, but rather asked to view images or videos without a specific objective—follow Lèvy flights at random. These are random walks based on steps with heavy-tailed distributions, similar to the movement distribution of animals foraging for food, with the difference being that human gaze is aimed at finding and maximizing visually acquired information rather than nutrition [24, 69]. Universal attention also describes the effects resulting from the physical properties of the stimuli, such as salient regions (bottom-up attention), and physiological limitations [111], such as the inability to rotate eyes at a velocity higher than 500° per second. *Personalized attention* [277] is specific to an individual. Although scanpaths of an individual are different on exposure to the same stimuli multiple times, they are *idiosyncratic*, meaning they are more similar than to the scanpaths of others viewing the same stimuli [84]. Moreover, depending on the individuals' experiences, the context provided to them on exposure, and their intrinsic goals, their attention could differ from that of the general population [126].

Scanpath prediction is the computational task of modeling scanpaths, where a sequence of fixations is inferred based on the stimuli. The output of such models is in the form of Cartesian coordinates, indicating the positions of fixation points. Another form of representation is a *priority map* [285], which is a 2D spatial map that resembles a Fixation Density Map (FDM). Unlike the FDM, the priority map indicates a high-intensity point around the attention target of an individual, rather than the attention of a group of observers. Scanpath prediction is a sequential task, meaning that preceding fixations alter the ones to follow. The sequence of preceding fixations is known as the *fixation history* [146]. Most scanpath prediction models encode the fixation history within the model [185, 276] since the fixation history defines the past trajectory of the scanpath. Additionally, both personalized and universal attention affect the trajectory of a scanpath prediction. Early models of scanpaths were commonly designed to account only for universal attention [120]. Data-driven approaches [67, 34], on the other hand, are inherently

aware of personalized and universal attributes, since they infer scanpath patterns according to human viewing experiences, which are conditioned on both. However, the vast majority of such models are trained on the data of a single observer, or that of multiple observers viewing unique scenes. Therefore, these models do not encode the distinction between the two types of attention as they are exposed to samples that contain both. We describe a scanpath model that is capable of encoding personalized and universal attention separately as a *unified model*. As a result of the separation in encoding, single unified models can be utilized for predicting the scanpaths of multiple individuals independently. Alternatively, a model capable of only encoding the scanpaths of a single individual is described as an *individual model*. The main distinction between a unified and an individual model lies in the input and ground-truth data each model is trained on, as illustrated in Figure 2.2.

Social Attention and Cues

Social cues are signals that form the basis of communication between humans. Social cues are categorized into verbal and non-verbal, with the non-verbal cues being body movements, gestures, gaze, and facial expressions. *Social attention* refers to visual attention driven by *orienting cues* [32] such as body gestures and gaze—head and eye movements. We additionally consider facial expressions as orienting cues, given that they play a significant role in overt orienting. For instance, people are more likely to attend to faces displaying fear or joy over boredom or calmness, as they appear more salient, due to the fact that these expressions of emotion are high on the *affect arousal scale* [264]. This phenomenon is known is *affect-biased attention* [251, 204]. In this thesis, we focus on modeling non-verbal social cues that are independent of cultural interpretations and fall under social attention. These include gaze direction and orientation [191], gaze following [231], and facial expressions [251]. When referring to gaze direction or orientation, we are describing the coordinates of an observed individual's head and eyes in tandem. However, the eyes are considered the primary indicator of gaze, whereas head orientation is used as a secondary cue when the eyes are not clearly visible or obscured. Gaze following refers to the inference of a target upon which an individual gazes, i.e., gaze following is based on the estimation of gaze direction with the additional step of inferring the specific region where a person looks. Facial expressions are the categories of emotional displays visible on a person's face. These include facial gestures, such as a smile to indicate happiness or a frown to indicate sadness, that reflect the affective state of the person displaying them. Computational models that infer gaze direction or model attention, such as saliency and scanpath prediction models integrating social cues are termed social attention models.

2.1.4 Robot Gaze Control

Human-Robot Interaction (HRI) addresses the study of robot behaviors and their influence on human perception. Robot gaze has been a popular subject of study in HRI and more so in the field of social robotics [9]. A majority of these studies rely on

predefined templates of gaze behaviors that elicit the attribution of humanlikeness to a robot [174]. Common gaze behaviors in robotics include *joint attention* [200], *mutual gaze* [25], and *gaze aversion* [141]. Joint attention refers to the shared attention between the robot and the human, where both look toward the same region within their visual field. Mutual gaze is the establishment of eye contact between the human and the robot. Gaze aversion is the opposite of mutual gaze, whereby the robot deliberately looks elsewhere to avoid eye contact with humans. Such behaviors are mimicked when evaluating a robot's perceived agency [269], humanlikeness [97], likeability [78, 145], and intelligence [243]. Other behaviors relate to eye and head movements rather than social attention. One such is *smooth pursuit*, whereby the robot follows a moving object or person with its eyes and head. The *Vestibulo–Ocular Reflex* (VOR) [215] is an oculomotor function, specific to robots that have eyes and a head. It stabilizes the eyes of the robot as its head moves. VOR comes into play as well during *gaze shifts*, which describes the reorientation of the head to follow the line of sight [104]. Other behaviors such as saccadic movements mimicked by a robot are also implemented for specific robotic platforms [171, 195]. However, saccades could lead to the distortion of visual input when the cameras are attached to the eyes by design, as is the case with the iCub [186] robot.

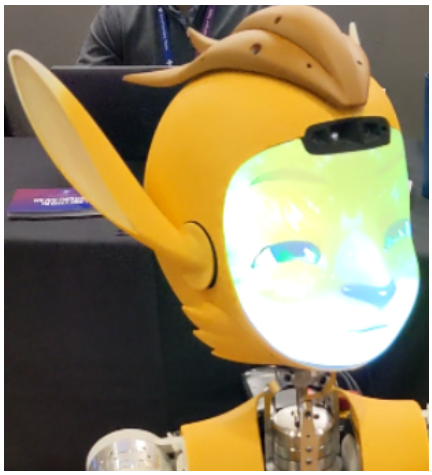
Robots with Eyes

The eyes of a robot could serve an *aesthetic* purpose, a *functional* purpose, or both. The aesthetic aspect refers to the eyes enhancing the robot's appearance to make it more approachable or lifelike. Moreover, the robot's eyes could signal a human to direct their attention to the robot itself or a certain area in their surrounding. On the other hand, the functional purpose of robotic eyes involves enabling the robot to perceive and interact with its environment. Functional eyes are equipped with cameras or sensors that provide visual information to the robot's processing system. In some robots, the eyes are designed to serve both aesthetic and functional purposes. For instance, a robot might have eyes that are aesthetically designed to look humanlike. The eyes could either be fixed in place [133] or employ mechanisms for actuation that mimic natural eye movement [171], as well as sensors for visual processing. This dual-purpose—aesthetic and functional—design can enhance the robot's social presence [220] and its operational capabilities by shifting its cameras toward a region of interest.

Social humanoid robots exhibit eye movements in several forms [56]. Mechanical eyes are driven by motors and tendons that rotate two spheres resembling eyes, along the azimuth and pitch axes [137, 171], as shown in Figure 2.3a. Both eyes rotate in the same direction, except in the case of *vergence*. Vergence is the movement of eyes in opposing directions along the azimuth. When an object is nearby, a *binocular vision*—relying on two eyes or two camera views—system perceives different views of it. Vergence avoids double vision in such instances by pulling the eyes closer to each other. For robots with functional eyes—a camera in each eye—such as the iCub [171] robot, vergence impacts applications of stereo depth estimation requiring

(a) Mechanical Eyes

(b) Integrated Display



(c) Face Projection

(d) Optical Illusion

Figure 2.3: The four common forms of robot eyes, with the **(a)** iCub [171] robot featuring mechanical eyes, **(b)** Navel [254] robot fitted with two round displays in place of the eyes, **(c)** Mirokaï [76] robot with a face projected on its display, and **(d)** Pepper [235] robot with fixed eyes caving into sockets to elicit mutual gaze with the observer.

both camera inputs [259]. Humanoid robot eyes were also engineered to enable *torsional eye movement*, which is the rotation of the eye along the roll axis [48]. Torsional eye movement is not only triggered by the observer's head movements but also by the movement of the stimuli [74]. However, due to the complexity of integrating torsional movement, the majority of humanoid robots with mechanical eyes do not support movement along this axis. Moreover, stabilizing or reorienting an image captured by a robot's cameras can be performed computationally rather than mechanically, which is less costly, more efficient, and can be readily adjusted.

Another form of robot eye movement relies on displays. All display-reliant forms of robot gaze signaling are purely aesthetic. Some models feature integrated displays that are localized to the region surrounding the eyes [254] as shown

in Figure 2.3b, while others rely on projectors fitted behind the robot's display as shown in Figure 2.3c. These projectors cast an animated face with two eyes onto the display [11, 76]. The quality of projection-based displays is more susceptible to light conditions, yet offer greater flexibility than localized displays. The last common form of eye movement in robots, as shown in Figure 2.3d, relies on optical illusions. A common illusion that invokes a sense of eye movement is the *Hollow-Mask illusion*. The Hollow-Mask illusion describes how the brain perceives images of faces that are displaced on a concave surface to be protruded instead. Robots with eyes that cave into the eye socket, elicit a sense of mutual gaze with the robot, regardless of the angle from where the eyes are viewed [235, 137].

Cognitive Robotic Simulation

Cognitive simulation refers to methods that computationally model human behavior. In this thesis, we adopt the term to describe social attention model simulation on a robot, a process which we describe as *cognitive robotic simulation*. This process differs from *robotic simulation*, which refers to the modeling of a robot's kinematic and physical properties using software [250, 252, 64]. Cognitive robotic simulation, however, involves the design of a computational model that learns a sequence of processes or actions that mimic human behavior, followed by the embodiment of the model in a robot. These models are not necessarily cognitively inspired, however, the resulting actions resemble prototypical human behavior. We design cognitive robotic simulation models that actuate a physical and simulated (robotic simulation) robot to evaluate our social attention models without involving humans in the evaluation process. This allowed us to measure the performance of our social attention models in physical environments while scaling the experiments to sizes beyond what would be practical with human participants.

Message-Oriented and Robotics Middleware

Software interfaces and drivers are developed to control motors and acquire sensory data, providing helper methods in multiple programming languages that enable control and acquisition. However, robots, and more so, social robots, are a composition of devices that must act as a single entity. Exchanging signals to and from these devices demands a near-real-time communication infrastructure, parallelization capability, and robust transmission. These specifications are packaged into single interfaces, known as *middleware*. Most existing middleware can be categorized into *message-oriented middleware* and *robotics middleware*.

Message-oriented middleware are software libraries and frameworks that define conventions for data exchanges, transmission of specific data types, and the support of one or several communication patterns. Common communication patterns include *publish-subscribe*, which is a non-blocking pattern for the transmission of data from a single publisher to many subscribing nodes, and *request-reply*, a one-to-one pattern where a single server awaits a request from one client and responds to the client when a request is made. Message-oriented middleware is designed primarily as

software communication interfaces, where some are built to fulfill one or more goals, such as reliably exchanging big data [144], running on edge devices by having lightweight APIs or optimizing for low latency [175], supporting a wide range of communication patterns [113], and flexibly supporting infrastructure [143].

Robotics middleware generally employ their own communication protocols [206], or rely on existing middleware packages, such as ROS 2 [163] which is built on DDS [197]. Beyond the middleware itself, robotics middleware provide additional tools and frameworks for simulation [140], motion planning [101], and control [249], that communicate over specific middleware. The purpose of robotics middleware is to enable full control and access to a supported robot's actuators and sensors. Moreover, robotics middleware are either specifically developed to support a certain robot [170] or act as a framework that is robotic platform independent [206, 163].

Controlling social cue expressions on a robot, and more specifically gaze direction, requires a number of signal exchanges between different devices. First, the robot must perceive the stimuli through multiple sensors. For instance, in this thesis, we only receive visual and auditory signals from the robot's sensors. This requires a camera to acquire an image and microphones to capture audio. The signals from these sensors are then digitized and filtered through the robot's onboard hardware and software processors. Second, the signals are transmitted for further processing or recording. The processing is handled on other devices and very often, the transmission is handled by middleware incorporated into the robot's framework. Third, models that process and infer gaze actions from the transmitted signals could also be distributed on multiple machines that communicate using message-oriented middleware or remote procedure calls. Such approaches are followed to reduce the load on a single machine or processor when the processes can be parallelized. Fourth, the inferred gaze behaviors are transmitted over the middleware to a specific address, topic, or port, depending on the middleware used. These addresses facilitate direct communication with a client node or multiple subscriber nodes that listen to the specific messages delivered to that address. Finally, a node that actuates a robot's eyes or head listens for the coordinates signaling a certain gaze movement and processes or transforms them. On transformation from one coordinate system to another, performing inverse kinematics when the coordinates are in Cartesian space [217], or simulating the VOR, the node generates the necessary commands to actuate the motors responsible for moving the robot's eyes or head. These commands are then sent to the actuators, which execute the movements, enabling the robot to express the desired gaze behavior.

2.2 Related Work to Social Attention and Gaze Control

We briefly describe some of the advances in both saliency and scanpath prediction, with an emphasis on dynamic variants, i.e., models designed for predicting either task on video content. Static variants are designed to operate on images. Such models elicit different eye movement patterns from those trained on dynamic (video)

content [72]. Interacting with and perceiving stimuli within real-world settings necessitates the encoding of sequential knowledge due to the influence that such information has on our behavior. Since the approaches presented in this thesis are intended for robotic interaction in physical environments, dynamic models become a requirement. Therefore, for the purpose of this thesis, we do not review static models. For a comprehensive overview on static scanpath models, we refer the reader to the work by Kümmerer and Bethge [146]. Furthermore, we exclusively address audiovisual dynamic saliency models, since our goal is to develop audio-aware models, given that auditory information plays a significant role in guiding our visual attention, especially in social settings [83]. For a detailed overview on static and dynamic saliency prediction models, we refer to the reviews by Borji [35] and Yan *et al.* [279].

2.2.1 Audiovisual Dynamic Saliency Prediction

Auditory and multimodal features have become a target of interest in the context of visual saliency modeling [245, 33, 173, 256]. Tavakoli *et al.* [245] propose a simple deep learning model based on 3D-ResNet [105] for encoding the visual (video) and auditory streams separately. The two streams are joined, reduced in dimensionality, and decoded as two-dimensional priority maps with high intensity in regions where participants (observers) tend to fixate. Their approach follows the late non-fusion paradigm for integrating the auditory and visual feature representations. However, Tsiami *et al.* [256] propose early fusion of audiovisual representations. At multiple levels of the visual stream, they introduce supervised attention modules. At a deep stage of the visual stream, the visual features guide the auditory stream to the most salient regions, resulting in a more accurate localization of sound sources. Eventually, the supervised attention modules and the auditory features are concatenated, after which their representations are reduced to a single two-dimensional feature map. The authors show that this approach of early fusion [256, 173] for integrating auditory and visual features outperforms late integration [245].

This model [256], however, is trained in an end-to-end fashion, leading to long training times when saliency features are more difficult to determine solely based on the stimuli. Such is the case in social settings, e.g., the gaze direction of an active speaker influences the distribution of attention. A more explicit form of embedding representations of known phenomena is required to address complex settings. One such approach is proposed by Min *et al.* [173], who present a multistage audiovisual saliency model that minimizes the discrepancy between the proposed locations arriving from auditory and visual modalities. The modalities generate spatiotemporal saliency maps, which are adaptively fused in the final stage. We adopt a similar approach for our social attention models, however, instead of introducing features representing visual and auditory spatial features [173], we augment our models with social cue features to address saliency in social settings.

Jain *et al.* [121] present a 3D convolutional encoder-decoder model for predicting saliency. They explore different audiovisual fusion techniques and show that introducing auditory input does not result in significant improvement to the

performance of their model. Dynamic saliency models relying only upon visual stimuli tend to perform on par with audiovisual models [263, 26, 72] supporting the finding of Jain *et al.* [121]. The consensus on whether auditory stimuli play a significant role in predicting saliency is not clear. Yang *et al.* [280] introduce an audiovisual graph convolution-based model, emphasizing the importance of multimodal input for predicting saliency in 360° videos. However, Xiong *et al.* [271] argue that audiovisual saliency models may underperform due to temporal inconsistencies between auditory and visual streams. Despite some studies indicating visual-only models might suffice, our work primarily focuses on social attention, which inherently depends on both visual and auditory stimuli. Therefore, we include auditory features in all our proposed models.

2.2.2 Dynamic Scanpath Prediction

Scanpath prediction on dynamic scenes requires accommodating changes to stimuli along with the modeling of fixation trajectories. Most existing research addresses the prediction of egocentric gaze in videos recorded using head-mounted cameras. For instance, Li *et al.* [154] present a model for learning the temporal dynamics in first-person activity videos, utilizing motion and pose features relating the head and hand of the actor—first-person observer. However, this approach relies on a set of predefined features and assumes the hands of an actor are visible. This limitation is addressed by Huang *et al.* [119], who propose a multitask model for predicting saliency and task-guided attention transitions using independent 3D-CNN streams. However, in their approach [119], the saliency is predicted for the current visual frames. Other approaches anticipate saliency by generating future frames and predicting saliency on those frames. Such is the model proposed by Zhang *et al.* [287], which predicts gaze on the future frames generated using a Generative Adversarial Network (GAN). The GAN is composed of a discriminator model that receives future observations and anticipates future frames produced by the 3D-CNN generator. Concurrently, an independent 3D-CNN predicts fixations on the generated frames.

Most of these models are goal-directed, whereby the objective is known and the gaze fixations are supervisory signals. However, Aakur and Bagavathi [1] address egocentric gaze prediction as an unsupervised task. Their model is separated into three stages, initially extracting appearance and motion features, followed by a symbolically represented stage indicating the direction of information flow between spatial regions in the video. Finally, the model generates an attention map indicating the predicted fixation corresponding to locations with maximum energy. However, in this thesis, we do not consider goal-directed models. More specifically, our models are conditioned on free-viewing, where the goal is unspecified and the observer is tasked with simply viewing the video.

Another line of research addresses the prediction of the egocentric scanpath under the free-viewing condition [276, 273, 185, 278, 218, 153]. Xu *et al.* [276] train their model on individual observer fixation trajectories while freely viewing 360° videos on a VR headset. The model receives a video frame at a given timestep,

concatenated with local and global spatiotemporal saliency features. A recurrent model encodes the fixation trajectories and its latent representation is concatenated with the visual encoding of all saliency features to predict the displacement in fixation for the next frames. Naas *et al.* [185] follow a similar approach, replacing local features with an optical flow representation. These approaches predict the scanpaths of a single observer for each video given their past fixation histories as priors. However, all aforementioned architectures are trained to model the scanpaths within a limited viewport [153], unique to each observer. Consequently, the learned universal attention patterns are applicable to any individual, rather than representing their personalized attention. Such approaches represent universal behaviors but do not address individual differences among multiple observers. To alleviate this gap, we design models that predict the personalized gaze patterns of individuals. By doing so, we can evaluate the magnitude of these differences and determine whether the uniqueness of gaze patterns necessitates the development of personalized models, tailored for each individual observer.

Scanpath prediction on social videos is a less explored domain. One such model is proposed by Coutrot *et al.* [67] who develop a generalizable framework for predicting and classifying scanpaths based on a Hidden Markov Model (HMM) and discriminant analysis. Their approach is examined on static natural scenes and dynamic social scenes, identifying three location states for the HMM. These states are then used for classifying information relating to the observers or the stimuli. However, this approach is visual only. Audiovisual approaches such as the method developed by Boccignone *et al.* [34] relies on multimodal social cues as priors to a stochastic model, simulating the fixation patch transitions as a Poisson process. Rather than simulating the transitions in eye movement, Lan *et al.* [148] design a psychologically-inspired model for synthesizing gaze. Their method addresses the detection of actions, including "verbal communication", based on simulated eye movements. The key difference between these approaches and our scanpath prediction models is that we do not attempt to identify patterns that result in scanpath trajectories under certain conditions, rather, we train our models to predict fixations based on the fixation history of an observation. This means that we do not embed knowledge pertaining to eye movement into the model, allowing the model to represent the patterns through training.

2.2.3 Gaze Control in Social Robotics

Applying strategies of robot gaze control is very often required in human-robot interaction, influencing the perception of a robot by humans [184, 9]. Shiomi *et al.* [233] present a robot control approach for attending to faces or objects, according to their existence within either the foveal or peripheral regions in the camera views of the robot. Their approach relies on the integration of visual features from the periphery and fovea through particle filtering. Although the proposed approach addresses a myriad of social cues, including lip movement detection, facial expressions, and motion recognition, it relies solely on visual information. However, Csapo *et al.* [42] employ an approach that relies not only on visual cues but also

design a state machine-based conversational system that acquires auditory and tactile stimuli for controlling the gestures, speech, and gaze movements of a Nao robot. Motor-controlling modules often register conflicting actions, resulting in erratic motion by the robot. This occurs since the model [42] lacks an integration mechanism for combining the module representations. To remedy this erratic motion, the model is designed to run conflicting modules in separate process threads. Instead, Zaraki *et al.* [284] address conflicting signals arriving from different sensory modalities including visual, auditory, and 3D sensors by introducing an attention layer for computing the most prominent features. Moreover, their model handles multi-person interactions, enlarging the effect of social stimuli on the behavior of the robot. Their approach, however, does not mitigate abrupt transitions in attention. To mitigate abrupt transitions, Duque-Domingo *et al.* [73] present a competitive neural network model for selecting between social and physical cues without relying on both simultaneously, which could lead to destructive interactions between their features. Compared to the work of Zaraki *et al.* [284], Duque-Domingo *et al.* [73] forego 3D sensors and enable smooth transitions between the regions of priority for directing gaze. In this thesis, we follow a similar concept for integrating social cues, however, using a gated multimodal attention mechanism [17] to propagate features from all social cue modalities instead of propagating the winning nodes only as proposed by Zaraki *et al.* [284], while simultaneously attenuating less meaningful representations.

Other approaches for robot gaze control rely on assumptions relating to gaze behavior and statistical data. Such approaches simulate cognitive processes to convey socially plausible characteristics rather than reflect scanpaths of human gaze. For instance, Lathuilière *et al.* [149] develop a recurrent gaze control model, based on a Deep Q-Network [179]. In their approach, visual landmarks resulting from the pose estimation of each observed person as well as speech locations are extracted at each time step. Along with the current state of the robot (the observer), these landmarks are then fed into a recurrent neural network as sequential observations. The reward employed is shaped to maximize the number of observed individuals in the robot's field of view. Alternatively, Pan *et al.* [195] design a model that controls the transition of gaze on a robot, according to a predefined library of motion behaviors. Mishra and Skantze [176] adopt a similar approach, adding a planner to their HRI pipeline, allowing for the anticipation of actions before their execution. This allows for better head-eye coordination and humanlike gaze behavior. Although the aforementioned gaze control approaches prove especially useful in understanding human-robot interactions, they do not address measuring the resemblance of generated eye movements to human gaze. The closest attempt at this is by Saran *et al.* [224] who employ knowledge acquired from modeled gaze patterns to enhance robot learning. However, their approach does not explicitly measure how closely the robot's camera movement resembles human gaze but rather uses gaze information to improve task performance and policy learning.

2.2.4 Differentiation from Related Work

There exists limited work on robots reproducing human scanpath trajectories based on learned behavior. To the best of our knowledge, none of those models the scanpaths of multiple individuals. Moreover, the performances of robot gaze control approaches relying on social interactions are measured through HRI studies or on the robot's successful achievement of a goal-directed task. On the contrary, our objective is to evaluate the efficacy of social attention models under the free-viewing condition and how closely physical robots can mimic gaze behaviors with the addition of social cue information. We propose methods for conducting such evaluations, without necessarily carrying out HRI studies. Therefore, our approaches allow for more efficient testing that can be performed on larger scales and guarantee repeatability when environmental conditions are controlled.

Part I

Modeling Social Attention

Chapter 3

Gated Attention for Saliency Prediction

Saliency models were initially designed for bottom-up attention detection on images, and more recently, built to model top-down and bottom-up attention on dynamic videos using both visual and auditory input. However, most recent approaches [121, 280, 271] do not consider the effect social cues have on such models. In social settings, such cues are known to attract cognitive visual attention more prominently than other features [191, 204, 272]. In this chapter, we design a dynamic audiovisual saliency prediction model that can be used to enhance end-to-end saliency models by augmenting their features with auxiliary social cue representations. We explore various methods of multimodal integration and measure the improvement brought by the integration of these cues.

3.1 Introduction

Attending to regions or objects in our perceptual field implies an interest in acting toward them. Humans communicate their attention by fixating their eyes on those regions. By modeling fixation, we gain an understanding of the events that attract attention. These attractors are represented in the form of a *Fixation Density Map* (FDM), displaying blurred peaks on a two-dimensional map, centered on the eye *Fixation Point* (FP) of each individual viewing a frame. The FDM is a visual representation of saliency, a useful indicator of what attracts human attention.

Early computational research focused on bottom-up saliency, by which the conspicuity of regions in the visual field was purely dependent on the stimuli [120, 39]. On the other hand, task-driven approaches are top-down models utilizing supervised learning for performing tasks and allocating attention to regions or objects of interest. Combining face detections with low-level features has been shown to outperform bottom-up saliency models agnostic to social entities in a scene. Birmingham *et al.* [31] corroborate the advantage of facial features in modeling saliency. They establish that when social stimuli are present, humans tend to fixate on facial features, a phenomenon weakly portrayed by bottom-up saliency detectors.

Moreover, studies on human eye movements indicate that bottom-up guidance is not strongly correlated with fixation, which is rather influenced by the task [82]. The existence of social stimuli in a scene alters fixation patterns, supporting the notion that even with the lack of an explicit task, we form intrinsic goals for guiding our gaze.

Although facial features attract attention, studies show that humans tend to follow the gaze of observed individuals [44]. Additionally, psychological studies [204] indicate a preference in attending toward emotionally salient stimuli over neutral expressions, a phenomenon described as *affect-biased attention*. By augmenting saliency maps with emotion intensities, affect-biased saliency models show considerable improvement over affect-agnostic models [81, 62]. These approaches, although exclusive to static saliency models, are not limited to facial expressions, allowing for a greater domain coverage irrespective of the presence of social entities in a scene.

In light of the social stimuli relevance to modeling attention, we design a model to predict the FDM of multiple human observers watching social videos. Such models employ top-down and bottom-up strategies operating on a sequence of images, a task referred to as *dynamic saliency prediction* [21, 35]. Our model utilizes multiple social cue detectors, namely gaze following and direction estimation, as well as facial expression recognition. We integrate the eye gaze and affective social cues, each with its spatiotemporal representation as input to our saliency prediction model. We describe the resulting output from each social cue detector as a *feature map* (FM). We also introduce a novel FM weighting module, assigning different intensities to each FM in a competitive manner representing its priority. Each representation is best described as a *target map* (TM), combining top-down and bottom-up features to prioritize regions that are most likely to be attended. We refer to the final model output as the Predicted FDM (PFDM).

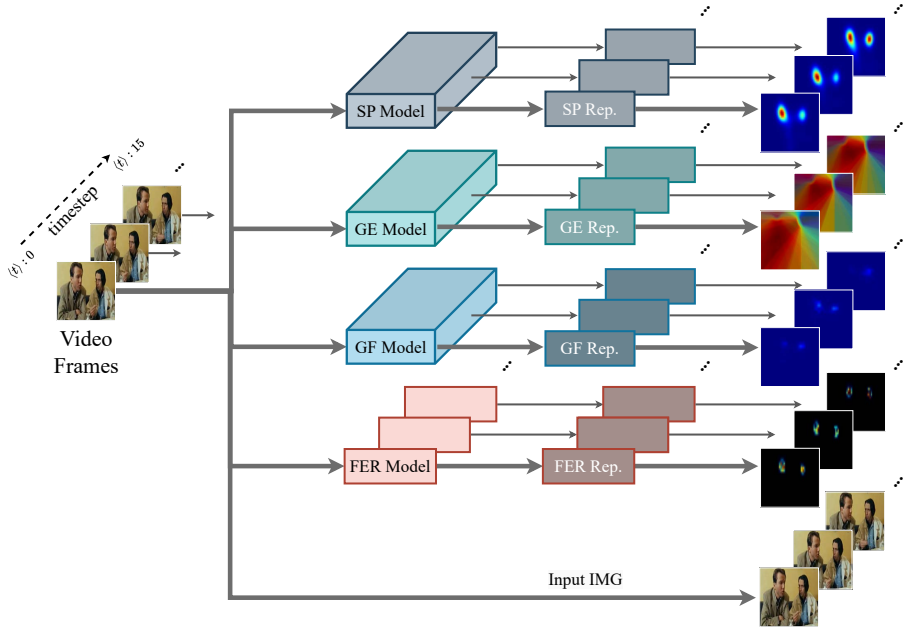
Our model architecture and task formulation—saliency prediction in social settings—decisions are guided by findings from recent research. Three important findings in the literature that are fundamental to the functionality of our approach:

F3.1 Task-driven strategies are pertinent to predicting saliency [82].

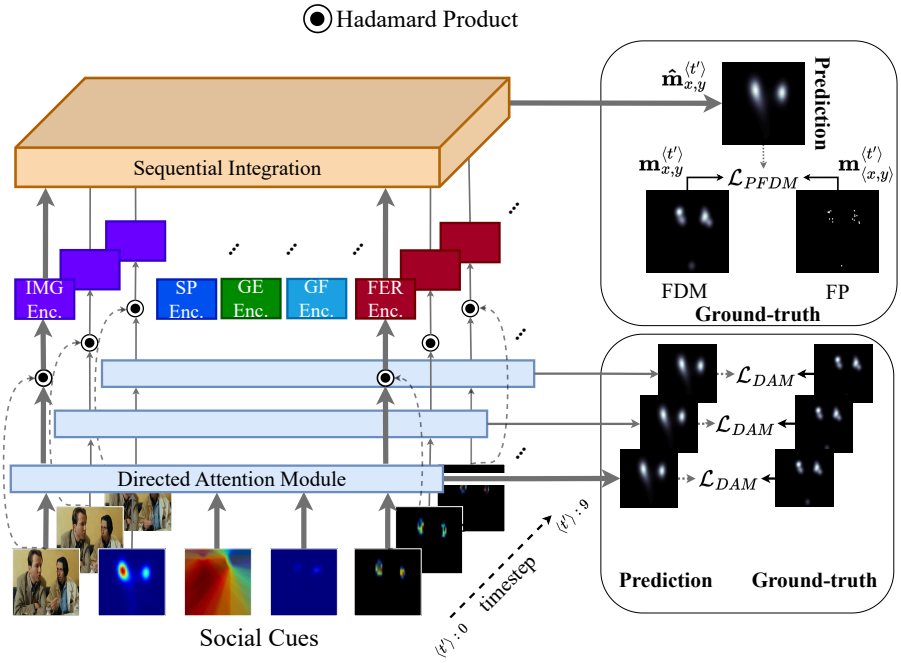
F3.2 Changes in motion contribute to the relevance of an object, underlining the importance of spatiotemporal features for predicting saliency [173].

F3.3 Psychological studies indicate that attention is driven by social stimuli [223].

To address the first finding, we state that our approach is task-driven by virtue of supervision since the objective is predicated on modeling multiple observer fixations. Although the datasets employed in this study were collected under a free-viewing condition, the top-down property is arguably maintained due to the intrinsic goals of the observer. These goals are driven by socially relevant stimuli addressed in our model through its reliance on multiple social cue modalities and facial information. We detect the social and facial features in a separate stage, hereafter described as the *Social Cue Detection* (SCD) stage.



(a) The social cue detection stage pipeline.



(b) The GASP model for sequential integration.

IMG: Input Image; SP: Saliency Prediction; GE: Gaze Estimation; GF: Gaze Following; FER: Facial Expression Recognition; DAM: Directed Attention Module. FDM: Fixation Density Map; PFDM: Predicted FDM; FP: Fixation Point. Rep.: Representation Transformation; Enc.: Modality Encoder.

Figure 3.1: Overview of our sequential two-stage model. SCD **(a)** extracts and transforms social cue features to spatiotemporal representations. GASP **(b)** acquires the representations and integrates features from the different modalities. $\hat{\mathbf{m}}_{x,y}^{\langle t' \rangle}$ represents the fixation density map predicted by the model at timestep t'.

To address the second finding, our approach relies on temporal features extracted or learned in two separate stages, respectively. Sequential learning in SCD is not a requirement but a result of the models employed for social cue detection, e.g., recurrent models or models pre-trained on optical flow tasks. In the second stage (GASP), we integrate social cues as illustrated in Figure 3.1, where sequences of frames composed of social cue representations are propagated to the model and attended independently. These representations act as auxiliary features that are not required but bring about improvements to the saliency model, as they guide the model to pay higher attention to social cues. GASP also employs sequential learning, not only registering environmental changes such as color and intensity but also features pertaining to motion. This is a direct result of the convolutional and recurrent attention modules employed in our GASP model, where saliency features (including motion) are emphasized should they contribute to the prediction of saliency.

Finally, we consider social attention by employing an audiovisual saliency prediction modality, as well as social cue detectors (also described as modalities) that specialize in performing distinct tasks. Each of these tasks is highly relevant to visual attention, from both behavioral and computational perspectives. We aim to explore feature integration approaches for combining social cues. We present gated attention variants and introduce a novel approach for directing attention to all modalities. To the best of our knowledge, our model is the first to consider affect-biased attention by using facial expression representations for dynamic saliency prediction based on deep neural maps.

3.2 Social Cue Detection

In the first stage (SCD), we extract high-level features from three social cue detectors and an audiovisual saliency predictor. We utilize the S^3FD face detector [288] for acquiring the face locations of actors in an image. The cropped face images are passed to the social cue detectors as input. The window size W, i.e., the number of frames fed simultaneously as input to each model, varies according to the requirements of each model. We sample and transform modality representations at output timestep T' for each social video in AVE [245] as shown in Algorithm 3.1.

Our motivation behind selecting social cue detectors lies in their ability to generalize to various in-the-wild settings, regardless of the surrounding environment or lighting conditions. All chosen models were trained on datasets consisting of social entities, captured from different angles and distances. Following the detection, we represent each social cue as a 2D spatiotemporal visual representation. We employ such representations to facilitate modularity and interpretability in our model. Modularity is made possible by enabling the replacement of social cue detectors without having to retrain our social attention model. Since the predictions of each social cue detection model are transformed into a predefined representation, their architectures and pipelines do not directly affect our model's functionality. Moreover, given the representations can be visualized as images, we are able to

Algorithm 3.1 SCD sampling

Definitions:
  Propagate: propagate transformed representations to GASP
  DetectFaces: detect faces and return face crops + bounding boxes
  Shift: shift left and discard first element

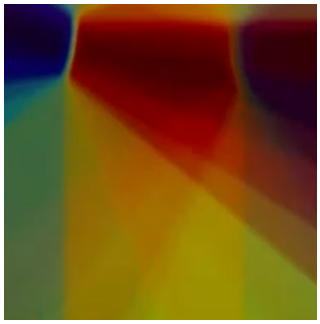
Input:
  Video and audio frames sampled from $\texttt{ds}$ = AVE dataset
Parameters:
  Window sizes $W_{\texttt{SP}} = 15$, $W_{\texttt{GE}} = 7$, $W_{\texttt{GF}} = 5$, $W_{\texttt{FER}} = 0$
  O/P steps $T'_{\texttt{SP}} = 15$, $T'_{\texttt{GE}} = 4$, $T'_{\texttt{GF}} = 0$, $T'_{\texttt{FER}} = 0$
Output:
  Modality windows $\texttt{mdl}_{\texttt{win}}$
  O/P buffers $\texttt{buf}_{\texttt{mdl}}$
for $\texttt{vid} \in \texttt{ds}$ **do**
    $t \leftarrow 0$
    for $\texttt{frm} \in \texttt{vid}$ **do**
        $\texttt{fcs} \leftarrow \texttt{DetectFaces(frm)}$
        for $\texttt{mdl} \in \{\text{SP, GE, GF, FER}\}$ **do**
            if $W_{\texttt{mdl}} > t$ **then**
                $\Delta \leftarrow W_{\texttt{mdl}} - t$
                for $\delta \in \{\Delta, \ldots, W_{\texttt{mdl}}\}$ **do**
                    $\texttt{mdl}_{\texttt{win}}[\delta] \leftarrow < \texttt{frm}, \texttt{fcs} >$
                end for
            else
                $\texttt{Shift}(\texttt{mdl}_{\texttt{win}})$
                $\texttt{mdl}_{\texttt{win}}[W_{\texttt{mdl}}] \leftarrow < \texttt{frm}, \texttt{fcs} >$
            end if
            $\texttt{buf}_{\texttt{mdl}}[t] \leftarrow \texttt{mdl}(\texttt{mdl}_{\texttt{win}})[T'_{\texttt{mdl}}]$
        end for
        $\texttt{Propagate}(\texttt{buf}[t])$
        $t \leftarrow t + 1$
    end for
end for

identify failures in the social cue detectors, which may consequently result in degrading our model's performance. Given these representations are spatial, not only can we visually identify the directions and positions of detected cues, but so are the features spatially correlated across all social and saliency cue representations, reducing our model's training duration.

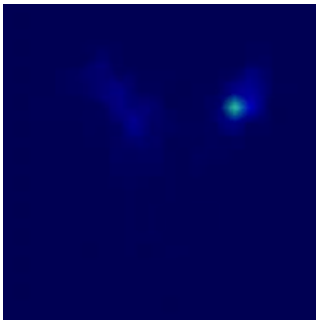
We additionally include a saliency prediction model in the stack of social cue detectors. The representation of the saliency prediction model guides our social attention GASP model by emphasizing regions predicted to be salient. This allows us to augment the saliency prediction model, rather than predicting saliency from raw images only. If we were to train our social attention model to predict saliency, this would require it to rely on large pretrained vision backbones. Consequently,

setting the raw image encoder to a pretrained backbone breaks symmetry across other social cue modality encoders. This is due to pretrained backbone models not correctly encoding the representations of social cue modalities and would require retraining. Therefore, our model would have to accommodate differences in modality encoders, which in turn could negatively affect modality weighting in our gating model.

(a) IMG + SP Representation



(b) GE Representation



(c) GF Representation

(d) FER Representation

IMG: Input Image; SP: Saliency Prediction; GE: Gaze Estimation; GF: Gaze Following; FER: Facial Expression Recognition.

Figure 3.2: SCD stage representations displaying the **(a)** fixation density map predicted by the DAVE [245] model superimposed on the image, **(b)** gaze estimation coordinates [130] rotating cones positioned over the actors' heads, **(c)** gaze following target maps [210] for both actors combined, and **(d)** Grad-CAM [229] representations generated for each actor as produced by the ESR9 [234] model.

3.2.1 Audiovisual Saliency Prediction Model

In the SCD stage, we utilize DAVE Saliency Prediction (SP) model by Tavakoli *et al.* [245], for predicting saliency based on visual and auditory stimuli. Separate streams for encoding the two modalities are built using a 3D-ResNet with 18 layers. The visual stream acquires 16 images (W_{SP}), each resized to 256 × 320 pixels. The auditory stream acquires log Mel-spectrograms of the video-corresponding audio frames, re-sampled to 16kHz. The model produces an FDM at the final output timestep T'_{SP} considering all previous frames within the window W_{SP}.

Transformation and Representation. We transform the resulting FDM from DAVE using a jet colormap as shown in Figure 3.2a. The color red indicates regions of high conspicuity, whereas blue indicates non-salient regions.

3.2.2 Gaze Estimation Model

We employ the Gaze360 Gaze Estimation (GE) model by Kellnhofer *et al.* [130]. The model infers the 3D gaze direction of an actor. The model receives face crops of the same actor over a predefined period, covering seven frames (W_{GE}) centered around timestep T'_{GE}. Each crop is resized to 224 × 224 pixels. The model predicts the azimuth and pitch of the eyes and head along with a confidence score.

Transformation and Representation. We generate cones and position their tips on detected face centroids. The cones are placed upon a zero-valued map with identical dimensions to the input image. The cone base is rotated toward the direction of gaze. The apex angle of the cone is set to 60°, corresponding to the angle of vision for a typical human. The face furthest from the lens is projected first with an opacity of 0.5, followed by the remaining faces ordered by their distances to the lens. A jet colormap is then applied to the cone map as shown in Figure 3.2b. Regions within the cone covering the angle of vision are displayed in red, whereas angles beyond the peripheral angle of vision (200°) are displayed in blue. The intensity of color reduces gradually between 60° and 200°. Moreover, the overlap in cones changes the representation color.

3.2.3 Gaze Following Model

We employ the VideoGaze Gaze Following (GF) model by Recasens *et al.* [210]. The model receives the source image frame that contains the gazer, the target frame into which the gazer looks, and a face crop of the gazer in the source frame along with the head and eye positions as input to its pathways. All frames are resized to 227 × 227 pixels. The model acquires five consecutive frames (W_{GF}) at timestep T'_{GF} and returns a fixation heatmap of the most probable target frame for every detected face in a source frame.

Transformation and Representation. The mean fixation heatmaps resulting from each face in the source frame are overlaid on a single feature map in the corresponding target frame timestep. We transform the fixation heatmaps using a

jet colormap as shown in Figure 3.2c. The representation indicates high-intensity regions that are estimated to be gazed upon by the actors in the video. A high certainty of the gazed position is displayed in red, whereas a low certainty is closer in color to blue (minimum intensity).

3.2.4 Facial Expression Recognition Model

We employ the Facial Expression Recognition (FER) model developed by Siqueira *et al.* [234]. The model is composed of convolutional layers shared across 9 ensembles. The model receives all face crops in a frame as input, each resized to 96×96 pixels and recognizes facial expressions from 8 categories. Since the model operates on static images, we set the window size W_{FER} and output timestep T'_{FER} to 0.

Transformation and Representation. Grad-CAM [229] features are extracted from all 9 ensembles. We take the mean of the features for all faces in the image and apply a jet colormap transformation on them. A 2D Hanning filter is applied to the features to mitigate artifacts resulting from the edges of the cropped Grad-CAM representations. We center the filtered representations on the face positions upon a zero-valued map with dimensions identical to the input image as shown in Figure 3.2d. High Grad-CAM gradients are displayed in red to indicate high intensity, and blue to indicate low intensity. We set regions beyond the detected actors' faces to black (no color), as expressions are localized to faces rather than the entire visual input.

3.3 Sequential Integration Model

We standardize all SCD features to a mean of 0 and a standard deviation of 1. The input image (IMG) and FMs are resized to 120×120 pixels before propagation to GASP. Based on the saliency prediction model by Tsiami *et al.* [256], we choose image dimensions larger than those set by the authors (112×112 pixels), to ensure that any performance degradation is not due to a reduction in image resolution.

3.3.1 Directed Attention Module

The Squeeze-and-Excitation (SE) [118] layer extracts channel-wise interactions, applying a gating mechanism to weight convolutional channels according to their informative features. The SE layer, however, emphasizes modality representations having the most significant gain, mitigating channels with lower information content. For our purpose, it is reasonable to postulate that the most influential FM channels are those belonging to the SP since it would result in the least erroneous representation in comparison to the ground-truth FDM. However, this causes the social cue modalities to have a minimal effect, mainly due to their low correlation with the FDM as opposed to the SP.

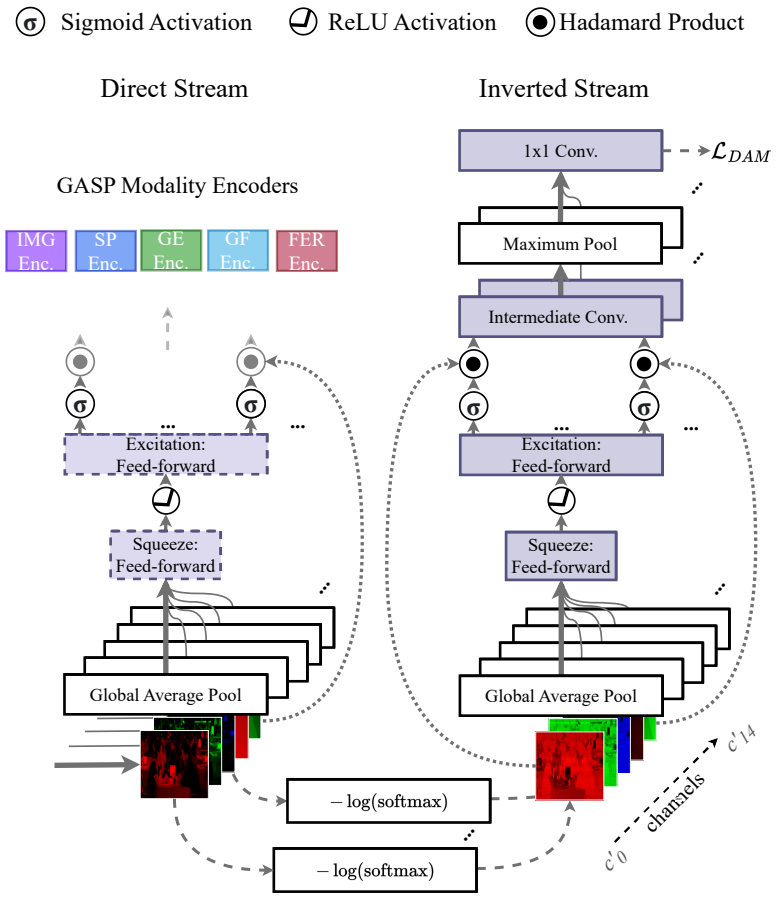


Figure 3.3: The direct *(left)* and inverted *(right)* streams of our Directed Attention Module (DAM). The parameters of the direct stream are frozen and tied to the inverted stream as indicated by the dashed borders.

To counter bias toward the SP, we intensify non-salient regions such that the model learns to assign greater weights to modalities contributing least to the prediction. Alpay *et al.* [14] propose a language model to skip and preserve activations according to how surprising a word is, given its context in a sequence. Similarly to how surprising words are propagated to the language model [14], we propagate visual channel regions with an unexpected contribution to the saliency model.

We construct a model for emphasizing unexpected features using two streams as shown in Figure 3.3: **1)** The inverted stream with output heads; **2)** The direct stream attached to the modality encoders of our GASP model. The inverted stream is composed of an SE layer followed by a 2D convolutional layer with a kernel size of 3×3, a padding of 1, and 32 channels. A *max pooling* layer with a window size of 2×2 reduces the feature map dimensions by half. Finally, a 1×1 convolution is applied to the pooled features, reducing the feature maps to a single channel. To

emphasize weak features, we invert the input channels:

$$\mathbf{u}_{c'}^{-1} = \log\left(\frac{1}{\text{softmax}(\mathbf{u}_{c'})}\right) = -\log(\text{softmax}(\mathbf{u}_{c'})), \tag{3.1}$$

where $\mathbf{u}_{c'}$ represents the individual channels of all modalities. The spatially inverted channels $\mathbf{u}_{c'}^{-1}$ are standardized and propagated as input features to the inverted stream. The direct stream is an SE layer with its parameters tied to the inverted stream and receives the standardized FM channels $\mathbf{u}_{c'}$ as input. Finally, the direct stream propagates the channel parameters multiplied with each FM to the modality encoders of GASP. The resulting weighted map is the target map (TM).

3.3.2 Modality Encoders

The modality encoder (Enc.) is a convolutional model used for extracting visual features from the target maps. The first two layers of the encoder have 32 and 64 channels respectively. A maximum pooling layer reduces the input feature map to half its size. The pooled layer is followed by two layers with 128 channels each. Finally, the representations are decoded by applying transposed convolutions with 128, 64, and 32 channels. The last layer has a number of channels equivalent to the input channels. All convolutional kernels have a size of 3×3, with a padding of 1. For GASP model variants operating on single frames (static integration variants), all modalities share the same encoder. For sequential integration variants, each modality has a separate encoder shared across timesteps.

3.3.3 Recurrent Gated Multimodal Unit

Concatenating the modality representations could lead to successful integration. Such a form of integration is commonly used in multimodal neural models, including audiovisual saliency predictors. We describe such approaches as non-fusion models, whereby the contribution of each modality is unknown. To account for all modalities, we employ the Gated Multimodal Unit (GMU) [17]. The GMU learns to weigh the input features based on a gating mechanism. For preserving the spatial features of the input, the authors introduce a convolutional variant of the GMU. This model, however, disregards the previous context since it does not integrate features sequentially. Therefore, we extend the convolutional GMU with recurrent units and express it as follows:

$$\begin{aligned}
\mathbf{h}^{(k)\langle t\rangle} &= \tanh\left(\mathbf{W}_x^{(k)} * \mathbf{x}^{(k)\langle t\rangle} + \mathbf{U}_h^{(k)} * \mathbf{h}^{(k)\langle t\rangle} + b_h^{(k)}\right), \\
\mathbf{z}^{(k)\langle t\rangle} &= \sigma(\mathbf{W}_z^{(k)} * [\mathbf{x}^{(1)\langle t\rangle}, ..., \mathbf{x}^{(K)\langle t\rangle}] + \mathbf{U}_z^{(k)} * \mathbf{z}^{(k)\langle t-1\rangle} + b_z^{(k)}), \\
\mathbf{h}^{\langle t\rangle} &= \sum_{k=1}^{K} \mathbf{z}^{(k)\langle t\rangle} \odot \mathbf{h}^{(k)\langle t\rangle},
\end{aligned} \tag{3.2}$$

where $\mathbf{h}^{(k)\langle t\rangle}$ is the hidden representation of modality k at timestep t. Similarly, $\mathbf{z}^{(k)\langle t\rangle}$ indicates the gated representation. The total number of modalities is represented

by K. The parameters of the Recurrent Gated Multimodal Unit (RGMU) are denoted by $\mathbf{W}_x^{(k)}$, $\mathbf{W}_z^{(k)}$, $\mathbf{U}_h^{(k)}$, and $\mathbf{U}_z^{(k)}$. The modality inputs $\mathbf{x}^{(k)}$ at timestep t are concatenated channel-wise as indicated by the $[\cdot,\cdot]$ operator and convolved with $\mathbf{W}_z^{(k)}$. The $\mathbf{z}^{(k)\langle t\rangle}$ representation is acquired by summing the current and previous timestep representations, along with the bias term $b_z^{(k)}$. A sigmoid activation function denoted by σ is applied to the recurrent representations $\mathbf{z}^{\langle t\rangle}$. The final feature map $\mathbf{h}^{\langle t\rangle}$ is the Hadamard-product between $\mathbf{z}^{(k)\langle t\rangle}$ and $\mathbf{h}^{(k)\langle t\rangle}$ summed over all modalities.

The aforementioned recurrent approach suffers from vanishing gradients as the context becomes longer. To remedy this effect, we propose the integration of GMU with the convolutional Attentive Long Short-Term Memory (ALSTM) [63]. ALSTM applies soft attention to single timestep input features over multiple iterations. We utilize ALSTM for our static GASP integration variants. For sequential variants, we modify ALSTM to acquire frames at all timesteps instead of attending to a single frame multiple times:

$$\mathbf{x}^{\langle t\rangle} = \text{softmax}(\mathbf{z}^{\langle t-1\rangle}) \odot \mathbf{x}'^{\langle t\rangle}, \tag{3.3}$$

where $\mathbf{z}^{\langle t-1\rangle}$ represents the pre-attentive output of the previous timestep, and $\mathbf{x}'^{\langle t\rangle}$ represents the input of the current timestep before applying attention. We adapt the sequential ALSTM to operate in conjunction with the GMU by performing the gated fusion per timestep. We refer to this model as the Attentive Recurrent Gated Multimodal Unit (ARGMU). Alternatively, we perform the gated integration after concatenating the input channels and propagating them to the sequential ALSTM. Since the modality representations are no longer separable, we describe this variant as the Late ARGMU (LARGMU). We refer to the total number of timesteps as the context size. Analogous to the sequential variants, we create similar gating mechanisms for static integration approaches. Replacing the sequential ALSTM with the ALSTM by Cornia *et al.* [63], we present the non-sequential Attentive Gated Multimodal Unit (AGMU), as well as the Late AGMU (LAGMU).

3.4 Experimental Setup

We trained our GASP model on the social event subset of AVE [245]. AVE is a composition of three datasets: DIEM [177], Coutrot Databases 1 [65] and 2 [66]. To train the model, we employed the loss functions introduced by Tsiami *et al.* [256], assigning the loss weights $\lambda_1 = .1$, $\lambda_2 = 2$, and $\lambda_3 = 1$ to *cross-entropy*, *CC*, and *NSS* losses respectively. The loss functions $\mathcal{L}_{PFDM}$ were weighted, summed, and applied to the final layer for optimizing the modality encoder and integration model parameters. The model was trained using the Adam optimizer, having a learning rate of .001, with $\beta_1 = .9$ and $\beta_2 = .999$. All models were trained for ∼10k iterations with a batch size of 4. We conducted five trials, reporting the mean in our results.

The models were evaluated on the test subset of social event videos in AVE. We employed five commonly used metrics in dynamic saliency prediction [245, 256]:

Normalized scanpath saliency (NSS); *Linear correlation coefficient* (CC); *Similarity metric* (SIM); *Area under the ROC curve* (AUCJ); *Shuffled AUC* (sAUC). The negative fixations for the sAUC metric are sampled from all the mean eye positions in the social event subset of AVE.

The inverted stream of our DAM layer has a separate output head for each timestep. We computed the cross-entropy between the DAM prediction and the FDM. For sequential integration models, the loss was summed over all timesteps. The loss $\mathcal{L}_{DAM}$ with a weight $\lambda_{DAM} = .5$ was computed for optimizing the inverted stream parameters. The parameters were transferred to the direct stream with frozen parameters.

An NVIDIA RTX 2080 Ti GPU with 11 GB VRAM and 128 GB RAM was used for training all static and sequential models. To extract spatiotemporal maps in the first stage (SCD), we employed an NVIDIA TITAN RTX GPU with 24 GB VRAM and 64 GB RAM to accommodate all social cue detectors simultaneously. We performed the SCD feature extraction in a preprocessing step for all videos in the AVE dataset.

3.5 Results

3.5.1 Static Integration

Table 3.1: Static integration results. *Top* rows represent non-fusion methods and bottom rows are fusion-based integration approaches. **Bold** denotes the best scores.

Model Architecture	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑
Additive	0.5842	0.5912	0.0882	1.19	0.1878
Concatenative	0.8782	0.6303	0.6614	2.71	0.4743
ALSTM	0.6881	0.5727	0.4503	2.05	0.3316
SE	0.5367	0.5597	0.0359	1.03	0.0972
LAGMU *(Ours)*	0.8347	0.6376	0.5576	2.48	0.4361
DAM + LAGMU *(Ours)*	0.8791	0.6379	0.6606	2.76	**0.5278**
GMU	0.8792	0.6374	0.6545	2.75	0.5172
AGMU *(Ours)*	0.6829	0.6359	0.2046	1.47	0.2212
DAM + GMU *(Ours)*	**0.8845**	**0.6397**	**0.6620**	**2.77**	0.5233
DAM + AGMU *(Ours)*	0.8587	0.6372	0.6372	2.71	0.5066

We examined integration approaches operating on a single frame in GASP. The *Additive* model refers to the integration variant in which the feature maps of all encoders are summed, followed by a 3×3 convolution with 32 channels and a padding of 1. The *Concatenative* variant applies a channel-wise concatenation to the feature maps, followed by the aforementioned convolutional layer. ALSTM, LAGMU, and AGMU employ the non-sequential ALSTM variant by Cornia *et al.* [63]. The

Squeeze-and-Excitation [118] (SE) model precedes the modality encoder. We note that all models excluding SE and DAM replaced the integration model with their own mechanisms, such as concatenation or element-wise addition. Finally, all model variants were followed by a 1×1 convolution resulting in the final output feature map.

In Table 3.1, the DAM + GMU (Ours) model achieved the highest AUCJ score, outperforming other fusion-based methods such as GMU and DAM + AGMU (Ours). This result indicated that the addition of DAM enhanced the integration process in combination with GMU, leading to improved detection performance. We also observed that DAM + GMU (Ours) produced the best CC score, demonstrating a strong correlation with ground-truth. This was closely followed by Concatenative and DAM + LAGMU (Ours), both of which performed well, showing that GMU, when combined with DAM, provided more accurate visual saliency maps.

In terms of SIM scores, DAM + LAGMU (Ours) achieved the highest value, indicating its greater similarity to ground-truth in pixel-wise predictions. It slightly outperformed both DAM + GMU (Ours) and GMU, suggesting that the incorporation of LAGMU with DAM offered a small improvement in this aspect. Finally, the AGMUs model, while showing moderate performance across most metrics, underperformed compared to other fusion-based methods, especially in terms of the CC and NSS scores.

3.5.2 Sequential Integration

We modified our GASP integration model to have a context greater than one. All models employ batch normalization applied to the temporal axis. The integration models are followed by a 1×1 convolution resulting in the final output feature map. In Table 3.2, we experimented with context sizes $\in \{2, 4, 6, 8, 10, 12\}$ and observed an overall improvement in performance with a context size of 4. The directed attention variant with late non-fusion gating DAM + LARGMU (Ours) achieved the best scores on all metrics. This implies that gated integration is beneficial, even though the representations preceding the GMU are not separable.

Comparing the results of static integration in Table 3.1 to dynamic integration approaches in Table 3.2, we observed that several static approaches perform on par with recurrent models. Nonetheless, the sequential DAM + LARGMU (Ours) with context sizes of 8 and 10 outperformed all integration methods. In Table A.1, we observed an insignificant difference in metric scores among the best sequential models for all context sizes. Compared to the best static model, the variances of sequential model scores were lower, indicating the stabilizing influence of attentive LSTMs with the addition of context.

Table 3.2: Sequential integration results. *Top* rows represent non-fusion methods and *bottom* rows are fusion-based integration approaches. **Bold metric** denotes the best scores across context sizes, with fusion and non-fusion approaches grouped separately. **Bold context** denotes context with the best overall scores across all metrics. **Bold architecture** denotes the architecture achieving the best score across all models for a specific context size.

Model Architecture	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑
Sequential ALSTM	**0.8849**	0.6428	0.6590	2.79	0.4522	0.8794	0.6439	0.6621	2.78	0.5244	0.8789	0.6425	**0.6612**	**2.76**	0.5291
LARGMU *(Ours)*	0.8791	**0.6444**	0.6572	2.76	0.5251	**0.8860**	**0.6460**	**0.6698**	**2.80**	0.5191	**0.8818**	**0.6433**	0.6556	**2.76**	0.5205
DAM + LARGMU *(Ours)*	0.8789	0.6437	**0.6703**	**2.78**	0.5354	0.8799	0.6443	0.6568	2.76	**0.5287**	0.8801	0.6423	0.6589	**2.76**	**0.5293**
RGMU *(Ours)*	0.8343	0.6239	0.6195	2.62	0.4717	0.8797	0.6350	0.6184	2.69	0.4614	0.7331	0.5851	0.5117	2.33	0.3823
ARGMU *(Ours)*	0.8793	0.6410	0.6607	2.74	**0.5359**	0.8819	0.6456	0.6556	2.75	0.5279	0.8656	0.6388	0.6534	2.72	0.5017
DAM + RGMU *(Ours)*	0.8726	0.6329	0.6547	2.73	0.5135	0.8714	0.6345	0.6539	2.73	0.5198	0.8718	0.6346	0.6510	2.73	0.5172
DAM + ARGMU *(Ours)*	0.8747	0.6421	0.6536	**2.78**	0.5305	0.8790	0.6363	0.6391	2.72	0.4934	0.8560	0.6271	0.6418	2.70	0.5133
	Context Size = 2					Context Size = 4					Context Size = 6				
Model Architecture	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑
Sequential ALSTM	0.8766	0.6389	0.6628	2.75	0.5307	0.8773	0.6412	0.6704	2.76	0.5306	0.8759	0.6352	**0.6665**	2.74	0.5275
LARGMU *(Ours)*	0.8702	0.6362	0.6529	2.72	0.5307	0.8791	0.6356	0.6511	2.72	0.5168	**0.8788**	0.6416	0.6624	**2.75**	0.5152
DAM + LARGMU *(Ours)*	**0.8872**	**0.6529**	**0.6903**	**2.84**	**0.5520**	**0.8830**	**0.6527**	**0.6980**	**2.87**	**0.5566**	0.8775	**0.6418**	0.6612	2.74	**0.5328**
RGMU *(Ours)*	0.7982	0.6031	0.5763	2.48	0.4368	0.8229	0.5981	0.5197	2.53	0.4502	0.8147	0.5717	0.4119	2.30	0.3276
ARGMU *(Ours)*	0.6892	0.5680	0.3253	1.84	0.2549	0.8467	0.6179	0.6116	2.62	0.4707	0.8457	0.6130	0.6007	2.56	0.4593
DAM + RGMU *(Ours)*	0.8678	0.6344	0.6622	2.75	0.5212	0.8579	0.6284	0.6540	2.72	0.5207	0.8759	0.6327	0.6549	2.74	0.5123
DAM + ARGMU *(Ours)*	0.8580	0.6259	0.6274	2.66	0.4874	0.8663	0.6303	0.6446	2.69	0.5161	0.8561	0.6249	0.6441	2.70	0.5117
	Context Size = 8					**Context Size = 10**					Context Size = 12				

3.5.3 Modality Contribution

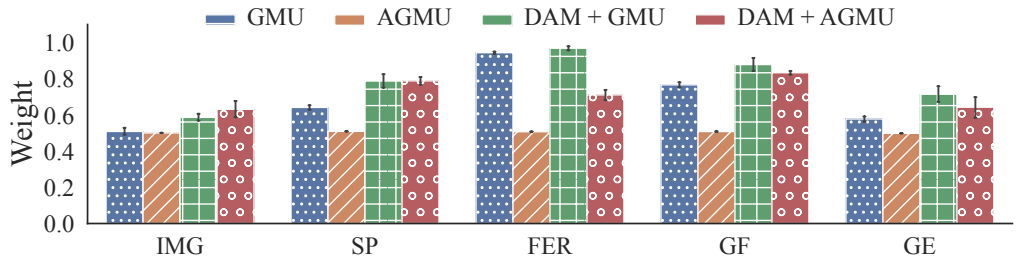
Table 3.3: Social cue modality ablation applied to our best GASP model (DAM + LARGMU; Context Size = 10). **Bold** denotes the best scores.

IMG	SP	GE	GF	FER	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑
✓	✓	-	-	-	0.8767	0.6338	0.6542	2.72	0.5228
✓	✓	-	-	✓	0.7535	0.5951	0.4466	2.17	0.3578
✓	✓	-	✓	-	0.6893	0.5679	0.3222	1.84	0.2539
✓	✓	-	✓	✓	0.8778	0.6442	0.6652	2.76	0.5350
✓	✓	✓	-	-	0.8769	0.6272	0.6493	2.70	0.4798
✓	✓	✓	-	✓	**0.8859**	0.6505	0.6840	2.86	0.5381
✓	✓	✓	✓	-	0.8776	0.6367	0.6543	2.74	0.5216
✓	✓	✓	✓	✓	0.8830	**0.6527**	**0.6980**	**2.87**	**0.5566**

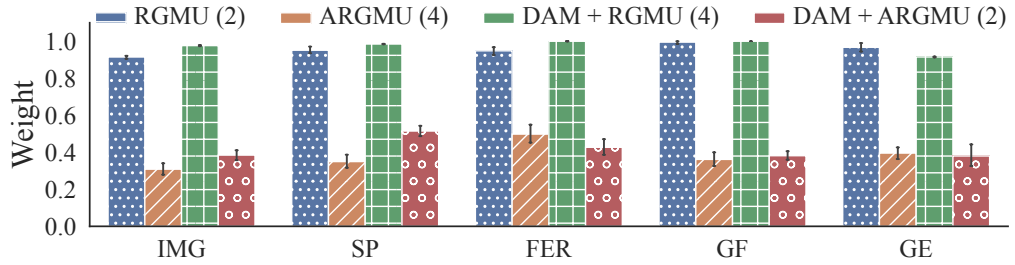
IMG: Input Image; SP: Saliency Prediction; GE: Gaze Estimation; GF: Gaze Following; FER: Facial Expression Recognition.

We measured the contribution of each modality to the final prediction by computing the mean activation of the gates across channels and timesteps as shown in Figure 3.4. This evaluation method is only applicable to fusion models in either static or sequential forms of integration, since their representations are separable, unlike non-fusion models. We observed that the DAM does not alter the modality contribution of the static GMU. For sequential variants, introducing the DAM allows modalities to have a uniform contribution to the final output.

We examined the modalities contributing an improvement to the best non-fusion sequential model. We included the raw input image IMG modality for all combinations since the model cannot operate without encoding the input image as there is insufficient information represented solely in the social cues. The SP was included as well since the model would require significantly more training iterations to reach scores on par with the baselines having all cues included. This would make the comparison unrepresentative of the cue contribution as models with the SP included required fewer training iterations, and therefore, vastly different performance from what we would observe in this experiment, should we have excluded SP. As shown in Table 3.3, FER in combination with GE achieved results on par with the best model. The exclusion of GF had a minimal effect on the model due to the sparsity of its representation. Significant degradation in the model variant with social modalities ablated implies the necessity of social cues in concert.



(a) Static fusion variants.



(b) Sequential fusion variants.

IMG: Input Image; SP: Saliency Prediction; GE: Gaze Estimation; GF: Gaze Following; FER: Facial Expression Recognition.

Figure 3.4: Aggregated modality weights of **(a)** static and **(b)** sequential fusion methods. Context sizes are shown within parentheses.

3.5.4 Comparison with State-of-the-Art

Table 3.4: Comparison with state-of-the-art by varying the SCD SP of our best GASP model (DAM + LARGMU; Context Size = 10). **Bold** denotes the best scores.

Model Architecture	**Test**	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑
UNISAL (Visual Only)	AVE	0.8640	0.6545	0.4243	2.04	0.3818
TASED (Visual Only)	AVE	0.8601	0.6515	0.4631	2.19	0.4084
DAVE (Visual Only)	AVE	0.8824	0.6138	0.5136	2.45	0.4080
DAVE (Audiovisual Baseline)	AVE	0.8853	0.6121	0.5453	2.65	0.4420
STAViS (Visual Only)	STA	0.8577	0.6517	0.4690	2.08	0.4004
STAViS (Audiovisual)	STA	0.8752	0.6154	0.4912	2.79	0.4774
UNISAL + GASP *(Ours)*	AVE	0.8771	0.6334	0.6494	2.70	0.5244
TASED + GASP *(Ours)*	AVE	0.8602	0.6195	0.5736	2.50	0.4725
DAVE + GASP *(Ours)*	AVE	0.8830	0.6527	**0.6980**	2.87	**0.5566**
STAViS + GASP *(Ours)*	STA	**0.8910**	**0.6825**	0.6052	**3.08**	0.4324

We compared the performance of our model with four dynamic saliency predictors. We replaced DAVE [245] with STAViS [256], TASED [172] and UNISAL [72] in the SCD stage during the evaluation phase. Due to the overlap in datasets between DAVE and STAViS, we retrained our GASP model with STAViS as the SCD audiovisual saliency predictor. We evaluated and trained our STAViS-based model on social event videos according to the data splits concocted by Tsiami *et al.* [256] to avoid data leakage, as the video samples overlap with those found in the AVE dataset train and test splits.

Combining our best GASP model with different saliency predictors improved their performances, as shown in Table 3.4. Although the GASP model was not retrained, it extracted information from the social cue modalities and the saliency predictor (SP) pertinent to the prediction. The sequential GASP also exhibits greater resistance to central bias as shown in Figure A.1 (middle row) compared to other models, where the actor closest to the center is incorrectly predicted as a fixation target. The integration of social cue features and sequential inference in both stages of GASP contributed to such resistance.

3.6 Discussion

In Section 3.5.1, we evaluated GASP on static images following common integration and fusion techniques, as well as novel approaches proposed in this article. In most cases, the DAM contributed to an improvement over all other variants excluding the DAM. The DAM partially inverted the learning process—learns the opposite social attention of the target—and deterred the model from *shortcut learning* [95], whereby the model would rely heavily on the saliency representation arriving from the social cue detection stage. Shortcut learning in this context refers to the model propagating activations only from the saliency prediction model since its representation most closely resembles the ground-truth, consequently ignoring all other auxiliary social cue representations.

In Section 3.5.2, we extended the context of our model beyond one frame, to account for dynamic changes in the audiovisual stimuli. The size of the context had a significant influence on each model's performance, with a context size of 8 resulting in the best performance for most model architectures in terms of all saliency metrics. Moreover, we observed that most integration architectures outperformed fusion architectures. We hypothesize that integration architectures—attention precedes gating—attend to relevant features across all modalities, which are then filtered by the gating mechanism to emphasize the most salient modalities. Reversing the operation with fusion architectures would filter out, or down-weigh, the modalities that contribute the least to the task. Therefore, modality features that could potentially be relevant are mitigated before the attention operation, resulting in a lower overall performance.

In Figure 3.5, we observe that the attention is directed toward the face of either speaker. The static and dynamic models predicted salient regions that were centered on the faces of the individuals. However, in terms of attention distribution,

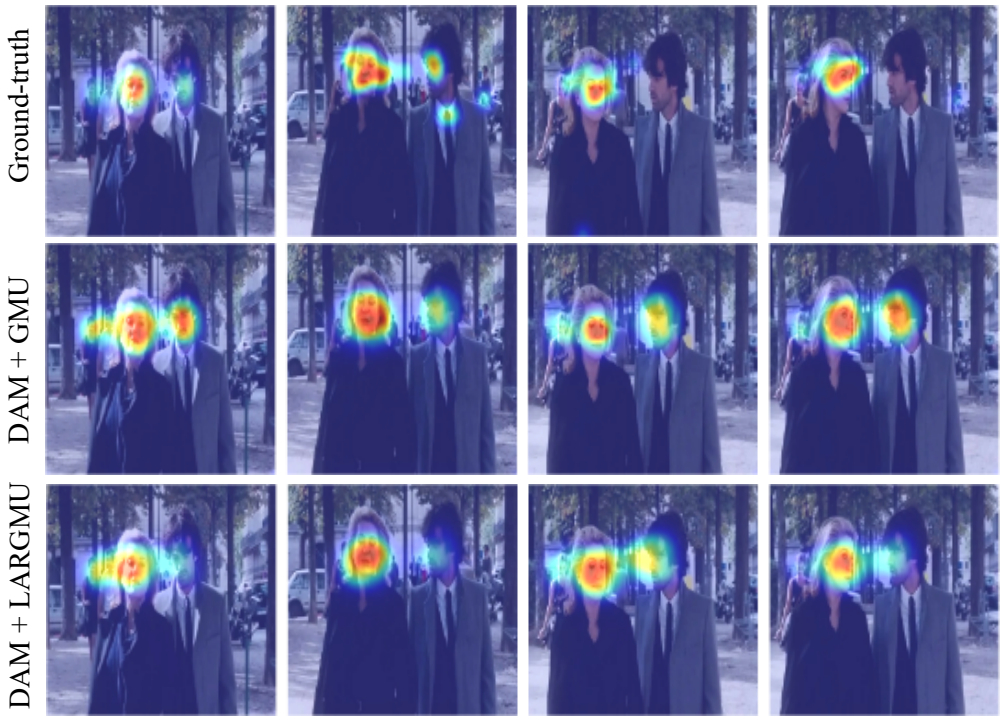


Figure 3.5: Frame predictions on the Coutrot Database 1 [65]. Our DAM + GMU: Directed Attention Module followed by the Gated Multimodal Unit for static integration *(middle)*; Our DAM + LARGMU: Directed Attention Module followed by the Late Attentive Recurrent GMU for sequential integration *(bottom)*.

the dynamic model corresponded more closely to the ground-truth than the static model. Since the dynamic model employs sequential integration for encoding the context of the video, it can better represent attention when it is driven by temporal changes. For example, since the person actively speaking attracts an observer's attention the most [158], a model aware of the dynamic changes indicating speech, such as lip movement, is better suited for predicting attention in social videos. The person speaking in Figure 3.5 is challenging to detect without relying on the context. The context indicated changes in the lip movements over time, giving the dynamic sequential integration model an advantage over the static model, which lacks context.

We observed minimal influence of the gaze following modality on the model's performance in Section 3.5.3. Gaze following is a task that is much more challenging than gaze estimation. The task is inherently a step beyond gaze estimation, in that it not only estimates the direction of gaze but also detects the region upon which the gazer looks. This results in higher noise, where the uncertainties produce more false-positive detections. A higher noise would also affect our downstream GASP model, as it would consider all modality representations. This contributed to a reduction in the performance of our social attention model, especially when the gaze following modality was the only social cue employed. When all three cue modalities were included, the model still performed moderately better.

Finally, in Section 3.5.4, we evaluated our GASP model by replacing the saliency predictor employed in the SCD stage, with other state-of-the-art models. We found that our social attention approach—inclusion of auxiliary social cue representations—improved all model performances. More importantly, replacing the saliency predictor did not require retraining or fine-tuning our GASP model. This trait makes our model adaptable to the other social cue detection and saliency prediction models, meaning that any of the trained modality encoders can receive input from another model performing the same task at inference time, and the GASP model can still operate in the same fashion.

Chapter 4

Unified and Individual Scanpath Prediction

In Chapter 3, we show that augmenting saliency prediction models with auxiliary social cue representations improves their predictions. However, saliency models represent the attention of a group of observers. To develop models that can simulate gaze, they should also be conditioned on sequences of fixations, also known as scanpaths [190]. In this chapter, we extend our social attention model with a fixation history module, and train it to instead predict the scanpaths of individual observers. Unlike the vast majority of existing scanpath prediction approaches [287, 1, 153], our model is capable of personalizing scanpaths, such that they are conditioned on the fixation history, allowing us to train a single unified model rather than individual models for each observer.

4.1 Introduction

Gaze shapes and guides social interactions, both for signaling and perceiving intent of others [47, 223]. Similarities across human eye movement patterns are described as *universal attention* [277] and are attributed to memory effects, bottom-up saliency, oculomotor biases, and physical constraints [150]. However, gaze patterns are influenced by socio-ecological factors and behavioral traits, that could differ depending on the observer. These factors contrive *personalized attention* [277].

In this study, we focus mainly on modeling human gaze patterns, known as *scanpaths*, to better simulate the cognitive behaviors exhibited during social interactions. The ability to simulate scanpaths is especially necessary for conducting human-robot interaction studies, where the gaze of the robot could greatly impact humans' perception and social acceptability of it [25, 149]. We model scanpaths under the free-viewing condition, whereby the observer is instructed to freely watch a video, without any predetermined objectives or tasks. The uniqueness of the free-viewing condition lies in the fact that it does not require any explicit gaze target. The viewing patterns under this condition are comparable to those exhibited by animals when foraging for food [24, 69]. This implies a universal goal that can

be associated with all observers. However, deviations in gaze patterns from the norm can be attributed to the intrinsic motivation of observers, shaped by their personalized attention [45].

Our task addresses the learning of scanpaths on observing dynamic (video) visual scenes. A closely related task is saliency prediction [245, 271], in which attention maps are learned based on the gaze of multiple observers. With minor modifications, saliency models can be used as predictors of scanpaths [146]. Saliency prediction models represent statistical measures of fixation distributions, visualized as spatial attention maps. However, equating the peaks of these maps with the fixation targets throughout a sequence might not accurately reflect an individual's scanpath. This could lead to abrupt transitions between fixations.

One potential solution is to train or fine-tune saliency models to predict individual scanpaths, maintaining traits emerging as a result of universal and personalized attention. However, this approach could be hindered by the sparsity of the predicted fixation maps (hereafter denoted as *priority maps* [285]), particularly if there's no input signal maintaining a record of fixation sequences for a single observer. Nevertheless, saliency predictors have shown great effectiveness in modeling human attention and could provide features pertaining to universal attention, therefore, useful for predicting scanpaths. While both saliency and scanpath prediction are closely related, there are critical differences that necessitate modifying the former task to better address the latter. We motivate the need for modifications based on the following observations:

O4.1 Saliency models predict group attention, whereas scanpath models predict the sequential attention of an individual. Namely, saliency models predict the distribution of fixation probabilities for multiple observers, whereas scanpath models predict the fixation trajectory of an individual observer viewing a scene. To repurpose a saliency model for predicting scanpaths, we need to fine-tune the model for each observer independently in order to represent their unique viewing patterns. To circumvent this limitation, we could alternatively train a single unified model by additionally providing a prior that can separate different scanpath trajectories. By focusing on individual scanpaths, we can represent the prototypical personalized attention patterns.

O4.2 Scanpaths of observers are non-deterministic and unique to the individual, meaning that they can vary from one viewing of a scene to the next. Previous research has shown that repetition of trials can increase the similarity between subsequent scanpaths for each observer, suggesting that humans have a natural tendency to follow different scanpaths without prior exposure to a given stimulus [84]. This phenomenon highlights the importance of using sequential models that can maintain the previous context, such as a memory component for storing the fixation history. This is in contrast to saliency prediction models, which infer fixation distributions that remain unchanged given the same stimuli.

O4.3 Salient stimuli and low-level features influence scanpaths. Unlike saliency prediction, scanpath prediction models the unique fixations of a single observer. Jiang *et al.* [124] have identified several factors that can influence the uniqueness of individual scanpaths, including low-level visual features, semantics, central bias, and fixation shift distribution. Based on these findings, we introduce a sequential audiovisual scanpath prediction model that can implicitly represent personalized patterns. More concretely, our model is not designed to explicitly represent these factors, but it can infer them from the input features, such as auxiliary social cue representations and attention maps. This allows our model to infer the patterns of attention that are characteristic of human observers.

Training a separate *individual* model for each observer could lead to the optimal prediction of their scanpaths. However, this approach is prohibitively expensive, given a large number of observers. We investigate whether individual models are necessary as opposed to a single *unified* model, distinctively predicting each scanpath based on the individual fixation histories. We pose the following questions:

RQ4.1 *Are the fixation histories adequate priors for differentiating scanpath trajectories?*

RQ4.2 *Does a model trained on individual viewing patterns independently, yield better predictions of scanpaths compared to a unified model trained on all fixations?*

RQ4.3 *How many multi-step-ahead fixations in a sequence can be reliably predicted from a given scanpath and stimuli before model predictions diverge?*

To tackle these questions, we propose a scanpath prediction model and framework that allows for the exploration of each. Given that social cues play a crucial role in modulating visual attention [112], we utilize GASP (detailed in Chapter 3), a dynamic saliency prediction model employing sequential gating mechanisms to augment raw audiovisual samples with representations of social cues. The best-performing GASP model concatenates the social cue representations, weighing their contributions by emphasizing weaker ones, and finally, combining them using an attentive convolutional LSTM [63] followed by a gating module [17]. We discard the gaze following social cue representation since it is shown to have an insignificant or even detrimental effect on the performance of GASP. Moreover, we extend the model with a fixation history channel that maintains a fixed number of previous fixation masks—a sequence of fixation points blurred by a 2D Gaussian filter with width equating to a 1° viewing angle.

In summary, we introduce a framework for modeling and evaluating dynamic scanpaths, inspired by existing methodologies in scanpath prediction for images [146]. Additionally, we present a modular multimodal architecture, designed for flexibility in accommodating various modules for detecting social cues. A key aspect of our approach is the utilization of an observer's fixation history, enabling the model to learn the scanpaths of multiple observers using a single unified model.

4.2 Gated Attention for Scanpath Prediction

We modify the GASP model for predicting scanpaths and adopt the two-stage approach for extracting and representing social cues, followed by feature integration. Separating the feature representation and scanpath modeling into two stages is both biologically plausible and computationally efficient.

From a psychological standpoint, our approach follows the feature integration theory [255], which states that low- and high-level features are processed in an initial stage. During this stage, only features of all objects are extracted, since prior knowledge about the relevance of an object is not yet processed. In the second stage, the features are clustered into objects, and each object is assigned a relevance, allowing for selectivity in attention toward the most conspicuous one. From a computational perspective, each social cue and saliency representation can be computed in parallel given that features relating to the interactions between different modalities are not required. We note that the lower bound in terms of time complexity, is determined by the slowest detection and representation modality.

4.2.1 Sampling and Social Cue Detection

We represent social cues following the paradigm introduced in Chapter 3. We retain DAVE [245] as the saliency predictor, Gaze360 [130] as the gaze estimator, and the facial expression recognizer developed by Siqueira *et al.* [234]. We discard the gaze following modality [210] due to its high time complexity and insignificant improvement to GASP.

During the fine-tuning phase, the image captures are downsampled to 10 frames per second. This aligns with the finding that eye fixations change within an interval of 100 to 500 ms [209]. The frames are pushed to a queue with a maximum size matching that of the modality with the longest context: DAVE with a context size of 16 visual frames.

Auditory signals are resampled to 16 kHz for accommodating videos irrespective of their original sampling rate. Resampling requires audio recordings of at least one second to avoid introducing artifacts. During training, we split one-second recordings beginning with the first visual frame in the context window into 16 chunks. We then extract 64 bands of the log mel-spectrogram with overlapping windows of .025s having a hop length of .01s following the same preprocessing technique adopted by Tavakoli *et al.* [245]. The resulting coefficients are propagated to the auditory stream of DAVE.

4.2.2 Fixation History Module

For predicting the scanpaths of individual observers, the model requires a mechanism for recalling previous fixations. This becomes relevant considering the scanpath differs for each observer exposed to the same stimuli and their scanpaths are dependent on their previous fixation points. During training, the fixation history is set to a sequence of 2D priority maps, created by applying Gaussian blur on

the fixation points. The Gaussian blur filter's width corresponds to a 1° viewing angle as a function of distance from the display monitor. This filter is applied to the previous ground-truth fixation points preceding the last timestep for a given sample. This translates to the teacher forcing strategy [266] during training i.e., the ground-truth maps of previous timesteps are fed as model inputs to predict the map of the current timestep.

During evaluation, our model predicts a priority map indicating the target of attention for an individual observer. The maps are then queued in the fixation history. The previous fixations define the context of attention, enforcing a foveated region upon the different input modalities and assisting in the prediction of the next fixation for an observer. The overall process equates to scanpath prediction with the added benefit of operating on dynamic contexts given an arbitrary number of timesteps. For predicting scanpaths, the fixation history cannot be discarded, especially in a unified model, as it serves as the primary mechanism for distinguishing between scanpath trajectories.

In Algorithm 4.1, we present the scanpath evaluation pipeline for each observer. The context size T' defines the number of recurrent timestep representations arriving from the different cue detectors. The predicted fixation $\hat{\mathbf{m}}^{\langle t' \rangle}$ at timestep t' is fed back into the fixation history for an arbitrary number of multiple steps ahead. We note the model's performance is primarily evaluated based on the output from its first prediction step. This output reflects the model's initial predictions, without extending into multi-step-ahead evaluations. To evaluate a model's capability in handling extended sequences without relying on the ground-truth after initializing the fixation history, we can iteratively input the model's predictions into the fixation history queue. This approach resembles the detection pipeline, with the key distinction of not acquiring the fixation history from ground-truth for subsequent stimuli detection. Evaluating multiple steps ahead allows us to assess the model's accuracy in forecasting future steps, closely reflecting real-world scenarios where we don't have access to the ground-truth data.

4.2.3 Sequential Integration Model

We describe the components of the sequential integration method of the GASP model presented in Chapter 3. The attentive recurrent gating mechanism as well as its late integration variant are detailed in this chapter. We also provide an overview of the directed attention module's role in improving a model's performance.

Directed Attention Module

The Directed Attention Module (DAM) is based on the Squeeze-and-Excitation [118] model for extracting the channel-wise interactions between the input modalities. The number of channels C' is defined by $K \times C$ where K is the total number of modalities and C denotes the number of image channels per feature map, assuming that all modalities have an equal number of image channels and dimensions. The initial aggregation in the form of average pooling across the channel pixels is

Algorithm 4.1 The dynamic scanpath evaluation pipeline incorporating the fixation history, similar to Kümmerer and Bethge [146]

Definitions:
  vf: video frames, ac: audio chunks
  fh: fixation history
  DetectCues: detect social cues
  Shift: shift left and discard first element
  UpdateFrame: update last video frame
  UpdateChunks: update audio chunks if new 1 s sample reached
  Integrate: sequential integration
  Sample: get video or audio at specified rate
  Eval: evaluate saliency metrics

$t' \leftarrow$ `current sub-sampled video frame index`
`vf` $\leftarrow$ `Sample(16 frames, 10 FPS)`
`ac` $\leftarrow$ `Sample(1 s audio, 16 chunks)`
$t' \leftarrow t' + 16$
for $t'' \in$ `{1, . . . , context size` T'`}` **do**
    `cues`$[t'']$ $\leftarrow$ `DetectCues(vf, ac)`
    `fh`$[t'']$ $\leftarrow \mathbf{m}^{\langle t' \rangle}$
    `vf` $\leftarrow$ `Shift(vf) & UpdateFrame(vf)`
    `ac` $\leftarrow$ `UpdateChunks(ac)`
    $t' \leftarrow t' + 1$
end for
for $n \in$ `{0, . . . , multi-step-ahead predictions}` **do**
    $\hat{\mathbf{m}}^{\langle t' \rangle} \leftarrow$ `Integrate(cues, fh, vf[16 -` T'`:16])`
    `Eval`$(\hat{\mathbf{m}}^{\langle t' \rangle}, \mathbf{m}^{\langle t' \rangle})$
    `vf` $\leftarrow$ `Shift(vf) & UpdateFrame(vf)`
    `ac` $\leftarrow$ `UpdateChunks(ac)`
    `cues` $\leftarrow$ `Shift(cues)`
    `cues`$[T']$ $\leftarrow$ `DetectCues(vf, ac)`
    `fh` $\leftarrow$ `Shift(fh)`
    `fh`$[T']$ $\leftarrow \hat{\mathbf{m}}^{\langle t' \rangle}$
    $t' \leftarrow t' + 1$
end for

expressed as follows:

$$\boldsymbol{\ell}^{[1]} = \frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} \mathbf{r}_{c'}(h, w), \tag{4.1}$$

where $\boldsymbol{\ell}^{[1]}$ represents the squeeze operation, W and H represent the width and height of the feature maps, respectively, whereas $\mathbf{r}_{c'}$ signifies the standardized feature map channel representation. The aggregated representations of all channels are then compressed and expanded using two linear fully-connected layers, with a non-linear activation following the first:

$$\boldsymbol{\ell}^{[2]} = \sigma(\mathbf{W}_{\ell}^{[2]} \cdot \mathrm{relu}(\mathbf{W}_{\ell}^{[1]} \cdot \boldsymbol{\ell}^{[1]})), \tag{4.2}$$

where $\boldsymbol{\ell}^{[2]}$ is the non-linear channel weight vector for scaling the channel contribution. The Sigmoid activation function σ is used as a gating mechanism rather than an attention mechanism, simply to avoid having a single active channel as would be the case if it were Softmax (attention) instead. The compression and expansion layer parameters are $\mathbf{W}_{\ell}^{[1]}$ and $\mathbf{W}_{\ell}^{[2]}$, respectively, where $\mathbf{W}_{\ell}^{[1]} \in \mathbb{R}^{\frac{C'}{\gamma} \times C'}$ and $\mathbf{W}_{\ell}^{[2]} \in \mathbb{R}^{C' \times \frac{C'}{\gamma}}$. The reduction ratio γ is a hyperparameter controlling the factor by which the channels are compressed. Finally, each feature map $\mathbf{r}_{c'}$ is scaled by its corresponding gain arriving from the $\boldsymbol{\ell}^{[2]}$ layer.

Following the approach presented in Chapter 3, we duplicate the Squeeze-and-Excitation model, denoting one by the direct stream and the other by the inverted stream. The former has a direct path to the scanpath model, whereas the inverted stream has a separate output head. The output head is composed of a 2D convolutional layer with 32 channels, a kernel shape of 3×3, and a padding of 1, followed by a max-pooling layer with a 2×2 window size, effectively reducing the feature map by half its size. The final layer aggregates the pooled representation to a single channel by applying 1×1 convolution.

The direct stream receives the concatenated channels of the social cue, saliency prediction, and fixation history representations. The final weighted feature maps (target maps) of the direct stream are propagated to our scanpath model. The number of target maps corresponds to the number of channels received by the direct stream. As the name implies, the inverted stream acquires the chromatically inverted modality representations by applying a non-linear transformation $\mathbf{r}_{c'}^{-1} = -\text{softmax}(\mathbf{r}_{c'})$ to the modality channels. The output of the inverted stream predicts a sequence of fixation density maps (attention maps), corresponding to universal attention learning. These fixation density maps represent the top-down and bottom-up attention of multiple observers, which is prior knowledge that the individual does not possess. However, the plausibility of a bottom-up saliency detector is evaluated based on its resemblance to fixations of multiple individuals, when the task is designed to minimize top-down effects [230].

We assume that these attention maps represent the ideal saliency maps since a clear separation cannot be formed between bottom-up and top-down attention. Moreover, we hypothesize that individual differences in attention should be large enough to distinguish between the scanpath trajectories. This implies that extrinsic factors that attract attention would have the highest impact on the fixation density per frame on average.

The motivation behind introducing the directed attention module lies in avoiding bias toward the saliency prediction representation, being both an input and target of the model. As a result, autoencoding the saliency input would be the optimal outcome, reducing all other modality connection parameters to zero. A more performant saliency model would amplify the biased reliance on its representation, leading to better performance overall, however, the generalization suffers. This is evident from the observation that training models on biased datasets leads to incorrect feature learning, albeit successful on the provided samples, a phenomenon in deep neural networks known as shortcut-learning [95]. To address this bias, the

model emphasizes weaker modality representations, i.e., modality representations that have a low spatial match to the ground-truth maps. Stronger representations are inhibited, causing the model to assign a larger gain weight to their representations during the learning phase. However, this is the case only when weaker representations lack information content that is sufficient to guide loss minimization.

Modality Encoders

The modality encoders are 2D convolutional neural networks with a structure similar to encoder-decoder models, i.e., feature compression of visual modalities through a bottleneck followed by decompression. We follow the same encoder structure described in Section 3.3.2 to initialize or model parameters with those of the pretrained models.

Attentive Convolutional LSTM and Gated Multimodal Unit

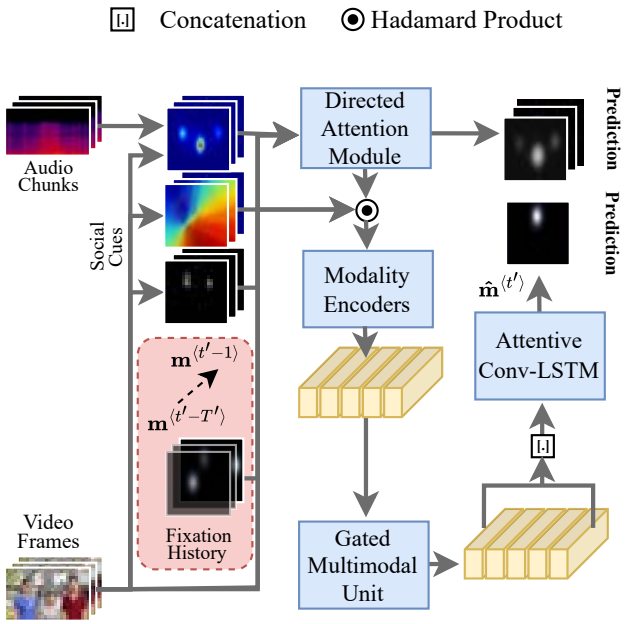
The Attentive Convolutional LSTM (ALSTM) [63] is an adaption to the convolutional LSTM model for recursively attending to feature maps. This structure has demonstrable advantages over conventional recurrent convolutional models and proves effective in modeling saliency as illustrated by Cornia *et al.* [63].

A convolutional LSTM is expressed as follows:

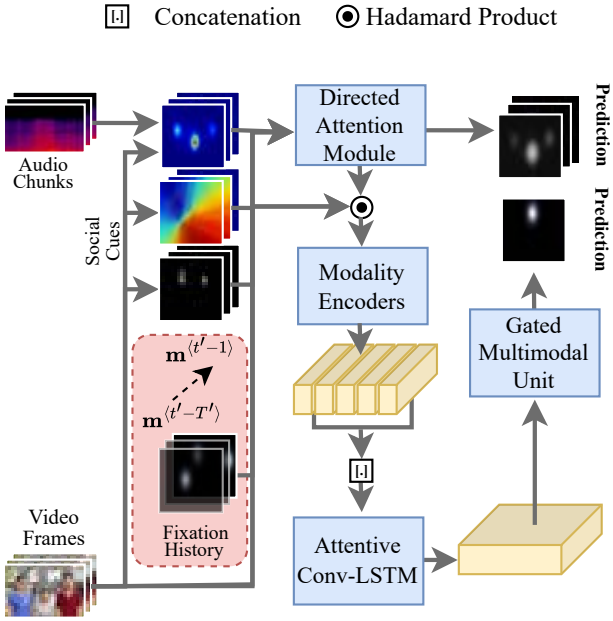
$$
\begin{aligned}
\mathbf{i}^{\langle t \rangle} &= \sigma(\mathbf{W}_i * \mathbf{s}^{\langle t \rangle} + \mathbf{U}_i * \mathbf{h}^{\langle t-1 \rangle} + b_i), \\
\mathbf{f}^{\langle t \rangle} &= \sigma(\mathbf{W}_f * \mathbf{s}^{\langle t \rangle} + \mathbf{U}_f * \mathbf{h}^{\langle t-1 \rangle} + b_f), \\
\mathbf{o}^{\langle t \rangle} &= \sigma(\mathbf{W}_o * \mathbf{s}^{\langle t \rangle} + \mathbf{U}_o * \mathbf{h}^{\langle t-1 \rangle} + b_o), \\
\mathbf{g}^{\langle t \rangle} &= \tanh\left(\mathbf{W}_c * \mathbf{s}^{\langle t \rangle} + \mathbf{U}_c * \mathbf{h}^{\langle t-1 \rangle} + b_c\right), \\
\mathbf{c}^{\langle t \rangle} &= \mathbf{f}^{\langle t \rangle} \odot \mathbf{c}^{\langle t-1 \rangle} + \mathbf{i}^{\langle t \rangle} \odot \mathbf{g}^{\langle t \rangle}, \\
\mathbf{h}^{\langle t \rangle} &= \mathbf{o}^{\langle t \rangle} \odot \tanh(\mathbf{c}^{\langle t \rangle}), \\
\mathbf{q}^{\langle t \rangle} &= \mathbf{W}_q * \tanh\left(\mathbf{W}_a * \mathbf{s}^{\langle t \rangle} + \mathbf{U}_a * \mathbf{h}^{\langle t-1 \rangle} + b_a\right),
\end{aligned}
\tag{4.3}
$$

where $\mathbf{W}_i$, $\mathbf{W}_f$, $\mathbf{W}_o$, and $\mathbf{W}_c$, represent the kernel parameters of the input $\mathbf{i}^{\langle t \rangle}$, forget $\mathbf{f}^{\langle t \rangle}$, output $\mathbf{o}^{\langle t \rangle}$ gates, and cell state, respectively. The bias unit for each projection layer is denoted by b_i, b_f, b_c, and b_o. The input map $\mathbf{s}^{\langle t \rangle}$ is convolved with all gate parameters at each timestep. The cell state is denoted by $\mathbf{c}^{\langle t \rangle}$ and the hidden state by $\mathbf{h}^{\langle t \rangle}$. The hidden state is convolved with the recurrent parameters $\mathbf{U}_i$, $\mathbf{U}_f$, $\mathbf{U}_o$, depending on the projection layer to which they apply. The convolution kernels are of size 3×3, with a padding of 1, having 32 channels each. We note that $\mathbf{q}^{\langle t \rangle}$ represents the pre-attentive output of the model at each timestep. The pre-attentive output has separate $\mathbf{W}_q$ kernel parameters that are convolved with the activated input map and previous hidden state. The input map and previous hidden state are convolved with the attention kernel parameters $\mathbf{W}_a$ and $\mathbf{U}_a$, respectively. The corresponding bias unit is denoted by b_a.

The ALSTM is a simple extension of the convolutional LSTM by which the input image $\mathbf{s}^{\langle t \rangle}$ is repeatedly propagated to the recurrent model and multiplied



(a) Fusion (DAM + ARGMU)



(b) Late Integration (DAM + LARGMU)

Figure 4.1: Our two GASP variants extended with fixation history modules for predicting scanpaths, where **(a)** is the modality *fusion* variant ARGMU, and **(b)** is the non-fusion late *integration* model LARGMU. The directed attention module (DAM) is applied to each variant with the fixation density maps for the entire sequence as ground-truth during training. T' represents the context size for each model, whereas t' indicates the current timestep (frame index) in the video. $\hat{\mathbf{m}}^{\langle t' \rangle}$ represents the priority map predicted by the model at timestep t'.

with $\text{softmax}(\mathbf{q}^{\langle t-1\rangle})$. In Section 3.3.3 we modify the ALSTM by restoring its sequential input property and attending to each element in the input sequence:

$$\mathbf{s}^{\langle t\rangle} = \text{softmax}(\mathbf{q}^{\langle t-1\rangle}) \odot \mathbf{s}'^{\langle t\rangle}, \tag{4.4}$$

where $\mathbf{s}'^{\langle t\rangle}$ represents the input map of the current timestep before applying attention. To integrate the modalities, we use the convolutional gating mechanism introduced by Arevalo *et al.* (GMU) [17]:

$$\begin{aligned} \mathbf{j}^{(k)\langle t\rangle} &= \tanh\left(\mathbf{W}_j^{(k)} * \mathbf{s}^{(k)\langle t\rangle} + b_j^{(k)}\right), \\ \mathbf{d}^{(k)\langle t\rangle} &= \sigma(\mathbf{W}_d^{(k)} * [\mathbf{s}^{(1)\langle t\rangle}, ..., \mathbf{s}^{(K)\langle t\rangle}] + b_d^{(k)}), \\ \mathbf{j}^{\langle t\rangle} &= \sum_{k=1}^{K} \mathbf{d}^{(k)\langle t\rangle} \odot \mathbf{j}^{(k)\langle t\rangle}. \end{aligned} \tag{4.5}$$

Here, $\mathbf{j}^{(k)\langle t\rangle}$ and $\mathbf{d}^{(k)\langle t\rangle}$ represent the gated projections for modality k of all modalities K at timestep t, along with their respective kernel parameters $\mathbf{W}_j^{(k)}$ and $\mathbf{W}_d^{(k)}$. The corresponding bias units are denoted by $b_j^{(k)}$ and $b_d^{(k)}$. The output $\mathbf{j}^{\langle t\rangle}$ represents the final feature map of the gating module resulting from the Hadamard product of all modality-specific projections $\mathbf{j}^{(k)\langle t\rangle}$ and $\mathbf{d}^{(k)\langle t\rangle}$. The modality inputs $\{\mathbf{s}^{(1)\langle t\rangle}, \ldots, \mathbf{s}^{(K)\langle t\rangle}\}$ at timestep t are concatenated across the channels as signified by the $[\cdot,\cdot]$ operator, and convolved with $\mathbf{W}_d^{(k)}$. We follow the integration paradigms introduced in Section 3.3.3 and adapt the sequential ALSTM to operate in conjunction with the GMU. One such integration paradigm entails performing sequential gating followed by modality gating. This model is referred to as the Attentive Recurrent Gated Multimodal Unit (ARGMU), illustrated in Figure 4.1a. Alternatively, performing the gated integration after concatenating the input channels and propagating them to the sequential ALSTM is illustrated in Figure 4.1b and denoted by the Late ARGMU (LARGMU) variant. We describe ARGMU as a *fusion* model, since feature integration occurs on modality-specific representations. LARGMU is a *late integration* model, since the modality representations are concatenated before integrating them into a single representation.

4.3 Evaluation Metrics

Common scanpath prediction metrics measure the proximity of human fixation trajectories to those generated by the model [68, 71]. One shortfall of such approaches is the requirement to temporally align the scanpaths under comparison. This, however, adds a layer of complexity to streamed dynamic stimuli which could potentially cause scanpaths to diverge over time. We instead follow the approach detailed by Kümmerer and Bethge [146]. Each sequence of input features along with fixation histories is used to generate a priority map for a single timestep. These maps are fed recursively to the model by appending them to the fixation history. One advantage to approaching scanpath evaluation as such is that it

enables the usage of common metrics used to validate dynamic saliency models. Saliency metrics are generally more robust to incorrect predictions and do not require as many parametric assumptions as is the case with scanpath metrics [146], for instance, ScanMatch [68] and MultiMatch [71]. Moreover, our models do not generate sequences. Instead, they predict the last fixation map—blurred fixation point—which is conditioned on previous ground-truth maps in the fixation history. Comparing the entire sequence using scanpath metrics would not accurately represent the performance of our models, since they predict single fixation maps at each timestep instead of the entire scanpaths. We, therefore, use the following nonprobabilistic saliency metrics [147] to evaluate our model predictions:

Normalized Scanpath Saliency (NSS) is a location-based metric [43] to measure the correspondence between ground-truth and predicted attention maps according to fixated locations. False positives have an effect on the NSS score, making it suitable for quantifying the quality of noisy predictions. A high positive NSS score indicates that the model accurately predicts the locations of fixations as expressed by:

$$NSS = \sum_{x,y} \frac{\hat{\mathbf{m}}_{x,y} - \mu(\hat{\mathbf{m}}_{x,y})}{\rho(\hat{\mathbf{m}}_{x,y})} \odot \mathbf{m}_{<x,y>}, \tag{4.6}$$

where $\mathbf{m}_{<x,y>}$ refers to the ground-truth fixation point rather than the continuous priority map expressed as $\mathbf{m}_{x,y}$. The mean of the priority map is denoted by μ, while the standard deviation is denoted by ρ. Our model predicts the probability of fixations, which is not restricted to a single point in space. Having multiple predicted fixations is the desired outcome in our scanpath modeling approach since a single prediction is an unrealistic assumption and would imply that our model is not robust, i.e., humans do not always look toward the same point when shown a sequence of images multiple times, therefore, having a definite fixation prediction indicates overfitting. Multiple fixations result in a lower NSS score for the priority map of an individual as compared to the group attention map. We, therefore, rely on the NSS rather as an indicator of the relative difference between the individual scanpath prediction models.

Area Under the ROC-Curve (AUC) is another location-based metric, which classifies whether a pixel in space is fixated or not. We rely on an AUC variant developed by Judd *et al.* [127], denoted hereafter by **AUCJ**. The advantage of using AUC as a measure of quality for our task is that the true and false positive rates are functions of the number of fixated and unfixated pixels, respectively. Since we have a single fixated ground-truth pixel for an individual, the weighing of true positives avoids skewing our evaluation toward false examples, providing a clear interpretation of our model's performance.

4.4 Experimental Setup

In this section, we describe the components of our experimental pipeline for conducting model training and evaluation. We also present the datasets used for training our models and the hyperparameter values chosen for those models.

4.4.1 Datasets

For our study, we used two existing datasets comprising eye-gaze data from observers viewing conversational videos. These datasets, sourced from YouTube and Youku, feature social videos with gaze data from a nearly identical observer count, ensuring consistency in our comparisons between the datasets. We note that each observer watched all the videos within a given dataset, enabling our model to distinguish prototypical gaze patterns for predicting scanpaths.

A dataset by Xu *et al.* [272] consists of 65 conversational videos. The dataset contains 39 participants (female = 13, male = 26) who took part in an eye-tracking experiment. Participants were between 20 and 49 years of age. All participants had normal, corrected-to-normal, or uncorrected vision. Two participants were experts in the field of saliency prediction, while the remaining had no experience in the field nor were they made aware of the purpose of the experiment. Hereafter, we refer to this dataset as *FindWho* [272].

The *MVVA* [158, 205] dataset, on the other hand, is more extensive with 300 conversational videos. The dataset contains 34 participants (female = 13, male = 21) who took part in an eye-tracking experiment. Participants were between 20 and 54 years of age, with a mean age of 24. All participants had normal or corrected-to-normal vision. 34 subjects (out of 39) were included in the dataset since they passed the eye tracking calibration procedure. In all analyses involving the MVVA dataset, one observer was excluded due to noisy data, reducing the total number of observers to 33.

4.4.2 Model Training and Evaluation

The individual and unified scanpath prediction models were trained on an NVIDIA GeForce GTX 3080 Ti GPU with 12 GB VRAM and 32 GB RAM. The individual model training process requires separate models for each observer. This is a highly demanding procedure, necessitating the distribution of models across multiple machines and GPUs. We orchestrated these processes through a custom workflow manager, developed using the Wrapyfi (detailed in Chapter 5) framework, allowing us to exchange completion logs across training instances over message-oriented middleware. All social cue detectors and models are described in Section 3.2 and are implemented in PyTorch [199].

The two model architecture variants, DAM + ARGMU (context size $T' = 8$) and DAM + LARGMU (context size $T' = 10$) were initialized with their GASP parameters, trained on the social subset of the AVE [245] dataset. We fine-tuned the individual and unified models on the MVVA and FindWho datasets separately, for 10 and 50 epochs, respectively. We used early stopping with $\delta_{min} = .0001$ and a patience of 3. This resulted in all models and architectures converging on average at epoch 6 for MVVA and epoch 11 for FindWho.

The individual models had a predefined observer set for all samples, whereas the unified model randomly selected an observer for each training sample. During training, frame samples overlapped by 90%. For the late integration architecture,

this resulted in a training time of 90 minutes per epoch on the MVVA dataset and 20 minutes on the FindWho dataset. For the early fusion architecture, the training time per epoch was 76 minutes on the MVVA dataset and 17 minutes on the FindWho dataset. The time required to train a single individual and unified model was identical given the same architecture and dataset. However, since individual models were trained separately for each observer, this resulted in 33 $\times$ epochs for the MVVA dataset and 39 $\times$ epochs for the FindWho dataset, in comparison to the unified models.

The batch size was set to 48 with gradient accumulation over 4 mini-batches, where each batch element contained the entire sequence of modality representations to match the context size of any given trained model. Each model was trained using the Adam optimizer, setting $\beta_1 = .9$, $\beta_2 = .999$, and the learning rate $\alpha = .001$.

Models were evaluated on subsampled video frames at 10 FPS, with no overlap between consecutive frames. The evaluation was performed on the basis of one-step-ahead prediction unless specified otherwise. All observer predictions were evaluated independently for both individual and unified models. All unified and individual models were trained and evaluated over 5 trials.

4.4.3 Saliency Losses

To train our model, we employ the loss functions introduced by Cornia *et al.* [63]. The loss functions are weighted, summed, and applied to the final layer, implying that the learnable parameters of our model, specifically the modality encoder and fusion model parameters are optimized. We denote the overall loss function by $\mathcal{L}_{PFDM}$ and define it as:

$$\mathcal{L}_{PFDM} = \mathcal{L}_{NLL} + \mathcal{L}_{KLD}, \tag{4.7}$$

where $\mathcal{L}_{NLL}$ computes the *negative log-likelihood* loss as expressed by Sun *et al.* [240] between the ground-truth and predicted priority maps, followed by minimization of the *Kullback-Leibler divergence*.

$$\begin{aligned}
\mathcal{L}_{NLL} &= -\lambda_{NLL} \cdot \sum_{x,y} \mathbf{m}_{<x,y>} \odot \log(\hat{\mathbf{m}}_{<x,y>}) \\
&\quad + (1 - \mathbf{m}_{<x,y>}) \odot (1 - \log(\hat{\mathbf{m}}_{<x,y>})), \\
\mathcal{L}^{+}_{KLD} &= \sum_{x,y} \mathbf{m}_{x,y} \odot (\log(\mathbf{m}_{x,y}) - \log(\hat{\mathbf{m}}_{x,y})), \\
\mathcal{L}^{-}_{KLD} &= \sum_{x,y} (1 - \mathbf{m}_{x,y}) \odot ((1 - \log(\mathbf{m}_{x,y})) \\
&\quad - (1 - \log(\hat{\mathbf{m}}_{x,y}))), \\
\mathcal{L}_{KLD} &= -\lambda_{KLD} \cdot (\mathcal{L}^{+}_{KLD} + \mathcal{L}^{-}_{KLD}).
\end{aligned} \tag{4.8}$$

Algorithmically, the cross-entropy loss utilized in GASP and $\mathcal{L}_{NLL}$ are identical, however, $\mathcal{L}_{NLL}$ operates on the fixation point, replacing the priority map $\mathbf{m}_{x,y}$ with

$\mathbf{m}_{<x,y>}$. Without the negative log-likelihood loss, the models require more epochs (3 to 7 additional epochs) to converge to similar states relying purely on $\mathcal{L}_{KLD}$.

The inverted stream of our DAM layer has a separate output head for each timestep. We compute the cross-entropy between the DAM prediction and ground-truth fixation density maps for all pixels summed over all timesteps. The $\mathcal{L}_{DAM}$ is computed to optimize the inverted stream parameters. These parameters are transferred to the tied direct stream. The direct stream parameters are frozen throughout the training phase. In this manner, we are able to emphasize weaker modalities, intensifying the propagation of noisy signals to the sequential integration model, effectively acting as a regularizer.

We employed the Tree-structured Parzen Estimator (TPE) method using Hyperopt [28] for hyperparameter optimization, to identify the optimal loss weights.[1] The considered weight range was $\in [.01, 1]$ sampled from a log-normal distribution. Based on the TPE's results after 90 trials, the loss weight for $\mathcal{L}_{KLD}$ is set to $\lambda_{KLD} = .94$, whereas the loss weight for $\mathcal{L}_{NLL}$ is determined to be $\lambda_{NLL} = .03$. For $\mathcal{L}_{DAM}$, the loss weight is established as $\lambda_{DAM} = .61$.

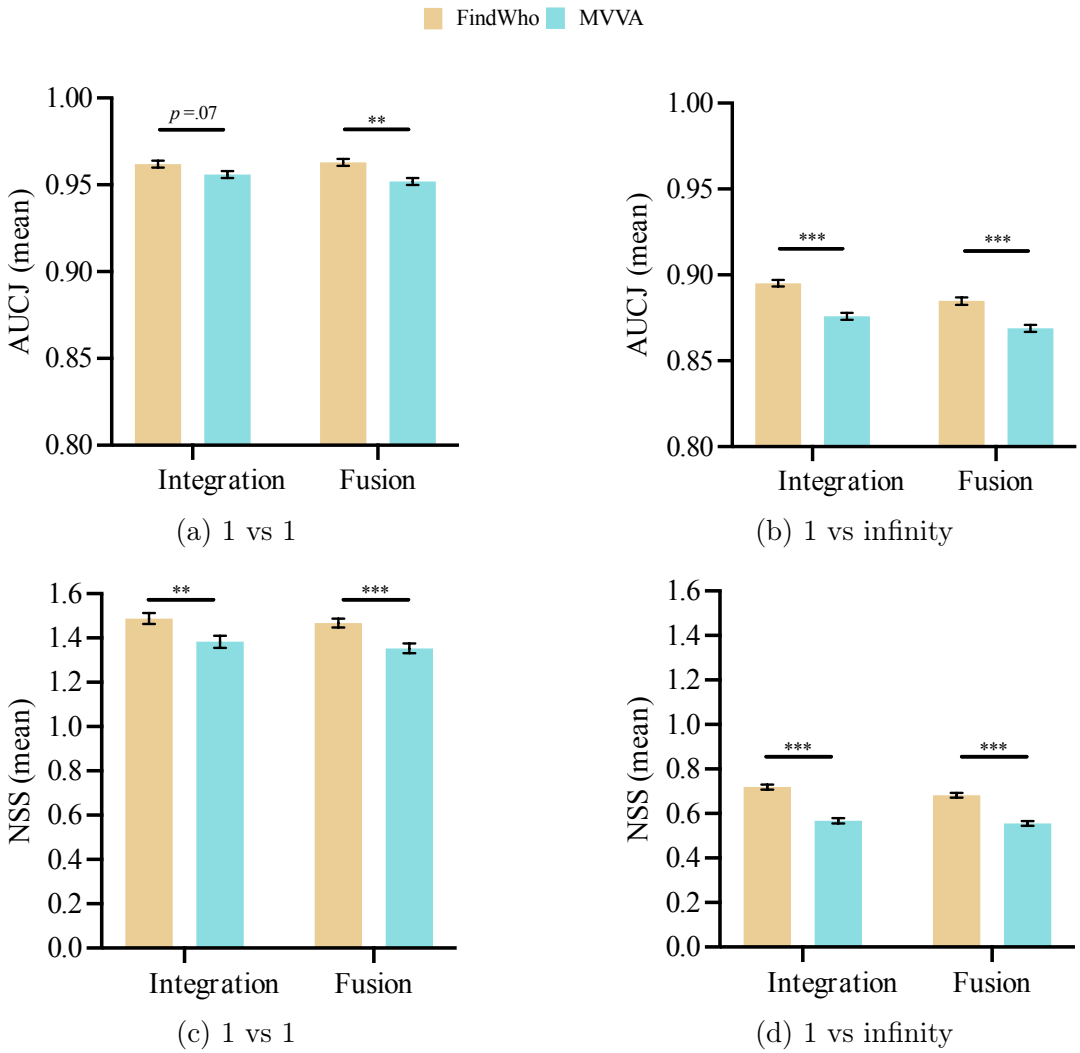
4.5 Results

We evaluated our late integration and early fusion architectures on the FindWho and MVVA datasets. This assessment was conducted by comparing each individual model's prediction against the last fixation in the individual observer's scanpath (*1 vs 1*) and against the group—all observers—fixation density map, excluding the individual's data (*1 vs infinity*). Moreover, we conducted statistical analyses to compare the unified and individual models. We then performed a social cue ablation study on the two unified model variants: late integration and early fusion. Finally, we tested the unified models to quantify the degradation of predictions over longer horizons beyond the next fixation point as detailed in Section 4.2.2, under multi-step-ahead evaluation. The mean values of the metric scores represent the performance of our models independently across all evaluation videos for every observer. Trial mean values are reported in the results unless stated otherwise.

4.5.1 Individual Models

To examine the impact of the model architecture, dataset size, and their interaction effects on the models' performances, a 2 (integration vs fusion)×2 (FindWho vs MVVA) mixed analysis of variance (ANOVA) was conducted. Specifically, the model architecture was a within-subject factor, and the dataset size was a between-subject factor. The performances of the models were measured in terms of the AUCJ score (mean and std) and NSS score (mean and std). All metrics were

[1] All search trials were applied to the late integration variant (DAM + LARGMU, $T' = 10$) with encoders pretrained on the AVE [245] dataset—excluding the fixation history module—and fine-tuned for 6 epochs on the MVVA [158] dataset. The TPE minimized the validation loss on the MVVA dataset.



(a) 1 vs 1

(b) 1 vs infinity

(c) 1 vs 1

(d) 1 vs infinity

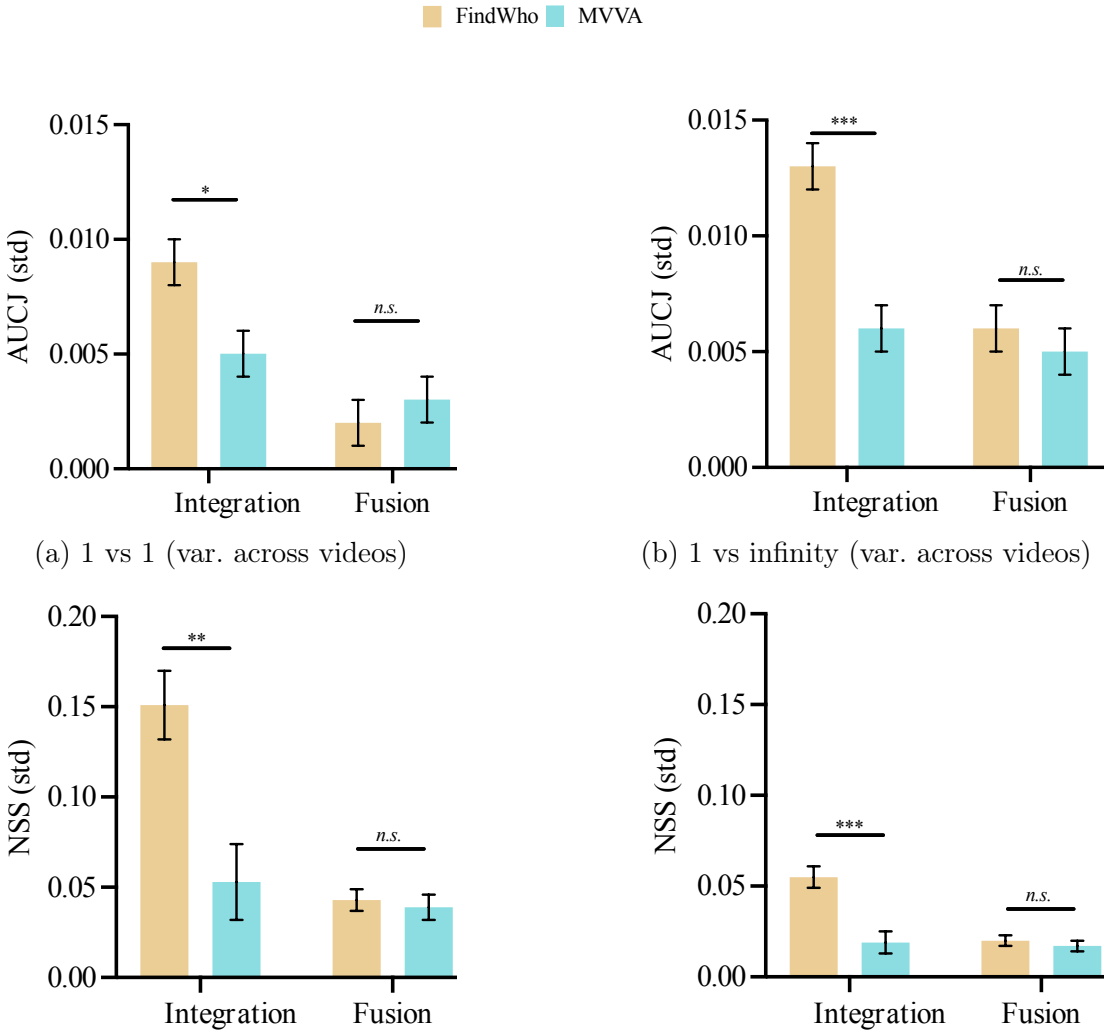
∗∗ denotes $.001 < p < .01$ and ∗∗∗ $p < .001$.

Figure 4.2: The individual model *1 vs 1* and *1 vs infinity* evaluations on the FindWho [272] and MVVA [158] datasets, across the two GASP variants extended with fixation history modules. **(a,d)** visualize the mean values of the scores across all samples.

measured between observers, except for 'variance across videos' experiments, where the metrics measure the variance in video results per observer—the score variances across the videos averaged for all observers.

1 vs 1 Evaluations

Significant main effects and interaction effects were observed in terms of the AUCJ and NSS metric scores. Assessing model performance on the datasets, we observe that the models trained on the smaller FindWho dataset outperformed the larger MVVA dataset, with significant differences (AUCJ: $F(1, 68) = 7.04$, $p < .05$,



(a) 1 vs 1 (var. across videos)

(b) 1 vs infinity (var. across videos)

(c) 1 vs 1 (var. across videos)

(d) 1 vs infinity (var. across videos)

$*$ denotes $.01 < p < .05$, $**$ $.001 < p < .01$, $***$ $p < .001$, and *n.s.* denotes no significance.

Figure 4.3: The individual model *1 vs 1* and *1 vs infinity* evaluations on the FindWho [272] and MVVA [158] datasets, across the two GASP variants extended with fixation history modules. **(a,d)** visualize the standard deviation of the scores across all testing videos per individual observer.

$\eta_p^2 = .09$ as shown in Figure 4.2a; NSS: ($F(1, 68) = 11.79$, $p < .01$, $\eta_p^2 = .14$) as shown in Figure 4.2c). The interaction effect between model architecture and dataset size was significant in terms of the AUCJ score ($F(1, 68) = 11.08$, $p < .01$, $\eta_p^2 = .14$), where the integration architecture significantly outperformed the fusion architecture on evaluating the MVVA dataset ($p < .01$). However, no significant differences were found on evaluating the models trained on the FindWho dataset ($p = .19$). The NSS metric, however, did not show a significant interaction between architecture and dataset size ($F(1, 68) = .19$, $p = .67$, $\eta_p^2 = .003$).

Variance across videos indicates instability in the model predictions. In terms

of the AUCJ score, the integration architecture resulted in a significantly larger variance compared to the fusion architecture ($F(1, 68) = 20.94$, $p < .01$, $\eta_p^2 = .23$). The models trained on the MVVA dataset performed significantly better—lower variance—within the integration architecture ($p < .05$) and no significant performance difference within the fusion architecture ($p = .62$) in terms of the AUCJ score as shown in Figure 4.3a. The interaction effect between architecture and dataset size was also significant in terms of the AUCJ score ($F(1, 68) = 6.44$, $p < .05$, $\eta_p^2 = .08$). Similarly, for the NSS score, the fusion architecture outperformed the integration architecture ($F(1, 68) = 17.96$, $p < .001$, $\eta_p^2 = .20$), and a significant interaction between architecture and dataset size was observed ($F(1, 68) = 10.78$, $p < .01$, $\eta_p^2 = .13$). The models trained on the MVVA dataset again performed significantly better within the integration architecture ($p < .05$) but no significant differences were observed within the fusion architecture in terms of the NSS score as shown in Figure 4.3c.

1 vs infinity Evaluations

In terms of the AUCJ and NSS metric scores, significant main effects were observed for both architecture and dataset size, with no significant interaction effects in some cases. In terms of the AUCJ mean values, the integration architecture demonstrated better performance compared to the fusion architecture ($F(1, 68) = 36.91$, $p < .001$, $\eta_p^2 = .35$). The models trained on the FindWho dataset outperformed those trained on the MVVA dataset in terms of the AUCJ score ($F(1, 68) = 40.98$, $p < .001$, $\eta_p^2 = .37$) as shown in Figure 4.2b. However, the interaction between architecture and dataset size was not significant in terms of the AUCJ score ($F(1, 68) = 1.32$, $p = .26$, $\eta_p^2 = .02$). Similar trends were also observed in terms of the NSS score, where the integration architecture also resulted in better performance compared to the fusion architecture ($F(1, 68) = 24.35$, $p < .001$, $\eta_p^2 = .26$). The models trained on the FindWho dataset significantly outperformed those trained on the MVVA dataset in terms of the NSS score ($F(1, 68) = 82.88$, $p < .001$, $\eta_p^2 = .54$) as shown in Figure 4.2d, due to the smaller size of the former dataset's test set. A significant interaction effect between architecture and dataset size was observed in terms of the NSS score ($F(1, 68) = 6.79$, $p < .05$, $\eta_p^2 = .088$). More specifically, the integration architecture models trained on the FindWho dataset outperformed the fusion architecture ($p < .001$), with no significant difference observed when trained on the MVVA dataset ($p = .12$).

Unlike *1 vs 1* variance across videos, a higher variance in *1 vs infinity* is interpreted as a positive outcome, since it indicates that the model predicts scanpaths that are personalized to the individual observer. Significant effects were observed for the architecture and dataset size in terms of the AUCJ and NSS metric scores. In terms of the AUCJ score, the integration architecture outperformed the fusion architecture ($F(1, 68) = 27.10$, $p < .001$, $\eta_p^2 = .28$). The FindWho dataset performed better than the MVVA dataset in terms of the AUCJ score ($F(1, 68) = 9.94$, $p < .01$, $\eta_p^2 = .12$) as shown in Figure 4.3b. A significant interaction effect was also observed, particularly within the integration architecture, where the models trained

on the FindWho dataset significantly outperformed those trained on the MVVA dataset in terms of the AUCJ score ($p < .001$). Similar significant main effects were observed for both architecture ($F(1, 68) = 15.36$, $p < .001$, $\eta_p^2 = .18$) and dataset size ($F(1, 68) = 16.59$, $p < .001$, $\eta_p^2 = .19$) in terms of the NSS score. The integration architecture models trained on the FindWho dataset also performed significantly worse than the models trained on the MVVA dataset in terms of the NSS score ($F(1, 68) = 12.36$, $p < .01$, $\eta_p^2 = .15$) as shown in Figure 4.3d. However, no significant differences were observed within the fusion architecture.

Comparison Between *1 vs 1* and *1 vs infinity* Evaluations

A group of paired-samples t-tests showed that for both architectures, performance in the *1 vs 1* evaluation was significantly better than in *1 vs infinity*, in terms of AUCJ and NSS metric scores, $ps < .001$. Details of the analyses on the FindWho and MVVA datasets can be found in Table 4.1 and Table 4.2, respectively. The results are shown in Figure 4.4 and Figure 4.5.

Table 4.1: Individual models trained and evaluated on the FindWho [272] dataset with *1 vs 1* and *1 vs infinity* comparisons in terms of AUCJ and NSS scores. The t-test degrees of freedom are shown within parentheses.

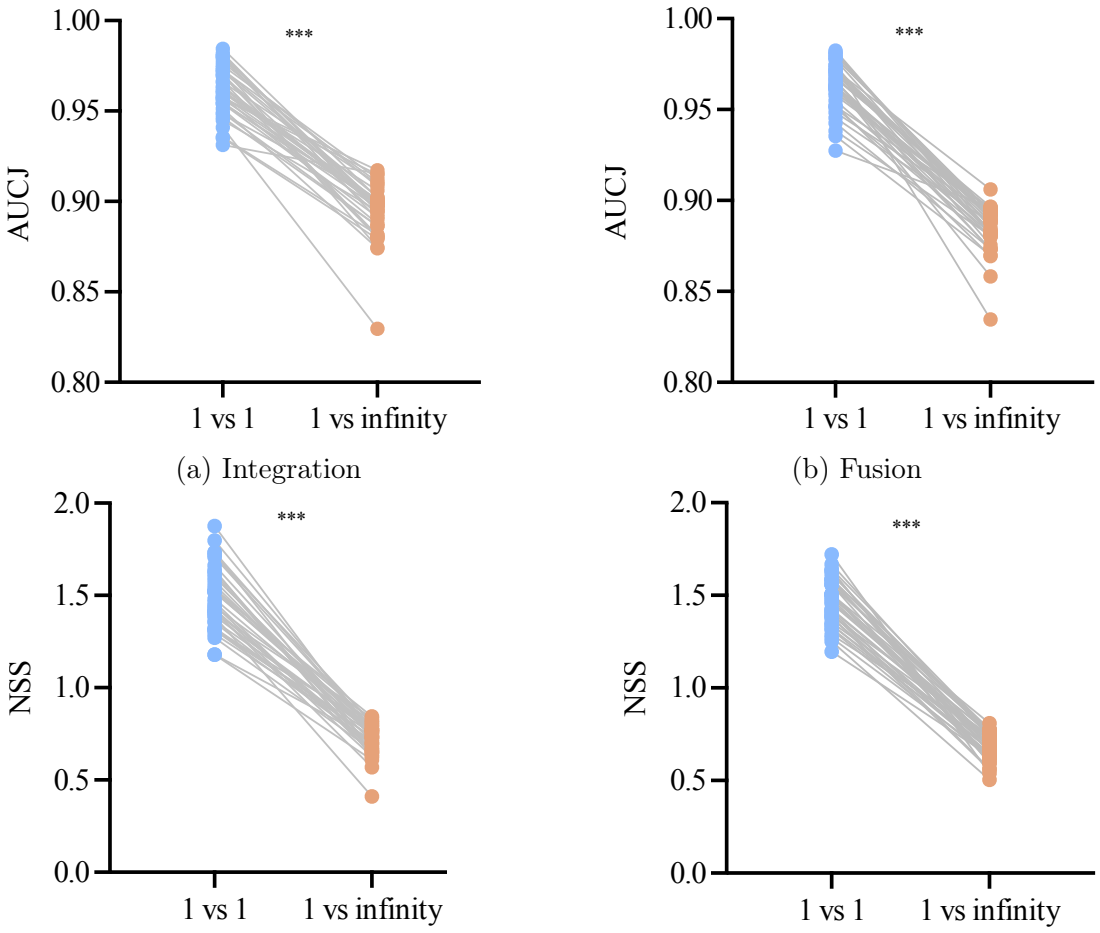
	Integration		**Fusion**	
	AUCJ↑	NSS↑	AUCJ↑	NSS↑
1 vs 1	0.962	1.488	0.963	1.467
1 vs infinity	0.895	0.719	0.885	0.682
t-value (df = 38)	23.23	27.89	27.45	36.90

Table 4.2: Individual models trained and evaluated on the MVVA [158] dataset with *1 vs 1* and *1 vs infinity* comparisons in terms of AUCJ and NSS scores. The t-test degrees of freedom are shown within parentheses.

	Integration		**Fusion**	
	AUCJ↑	NSS↑	AUCJ↑	NSS↑
1 vs 1	0.956	1.383	0.952	1.353
1 vs infinity	0.876	0.567	0.869	0.556
t-value (df = 32)	25.54	32.57	27.41	36.42

4.5.2 Unified vs Individual Models

To examine the impact of the model, model architecture, dataset size, and their interaction effects on the models' performances, a 2 (unified model vs individual model)×2 (integration vs fusion)×2 (FindWho vs MVVA) mixed analysis of variance



(a) Integration (b) Fusion

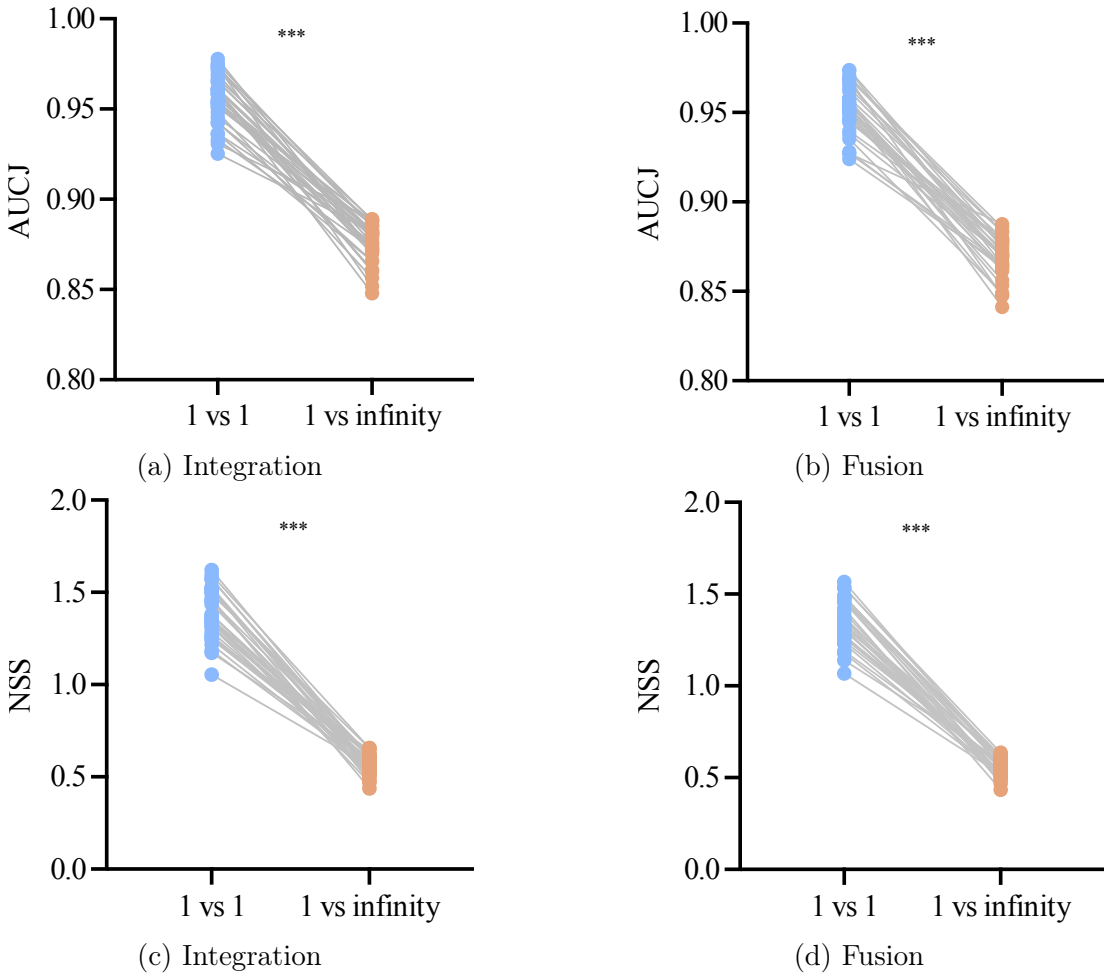
(c) Integration (d) Fusion

∗∗∗ denotes $p < .001$.

Figure 4.4: The individual model *1 vs 1* and *1 vs infinity* evaluations on the FindWho [272] dataset, across the two GASP variants extended with fixation history modules. The AUCJ scores are measured for the **(a)** integration and **(b)** fusion architectures, as well as NSS scores for the **(c)** integration and **(d)** fusion architectures.

(ANOVA) was conducted. Specifically, the model and model architecture were within-subject factors, and the dataset size was a between-subject factor.

In terms of both the AUCJ and NSS metric scores, significant main effects were observed. The unified model demonstrated better performance over the individual models (AUCJ: unified $.964 \pm .001$ vs individual $.958 \pm .002$, $p < .001$; NSS: unified $1.480 \pm .022$ vs individual $1.424 \pm .016$, $p < .01$). The integration architecture significantly outperformed the fusion architecture (AUCJ: integration $.964 \pm .001$ vs fusion $.959 \pm .001$, $p < .001$ as shown in Figure 4.6a and Figure 4.6b; NSS: integration $1.548 \pm .020$ vs fusion $1.356 \pm .016$, $p < .001$ as shown in Figure 4.6c and Figure 4.6d). Additionally, the models trained on the smaller FindWho dataset achieved better results compared to the those trained on the larger MVVA dataset



(a) Integration (b) Fusion

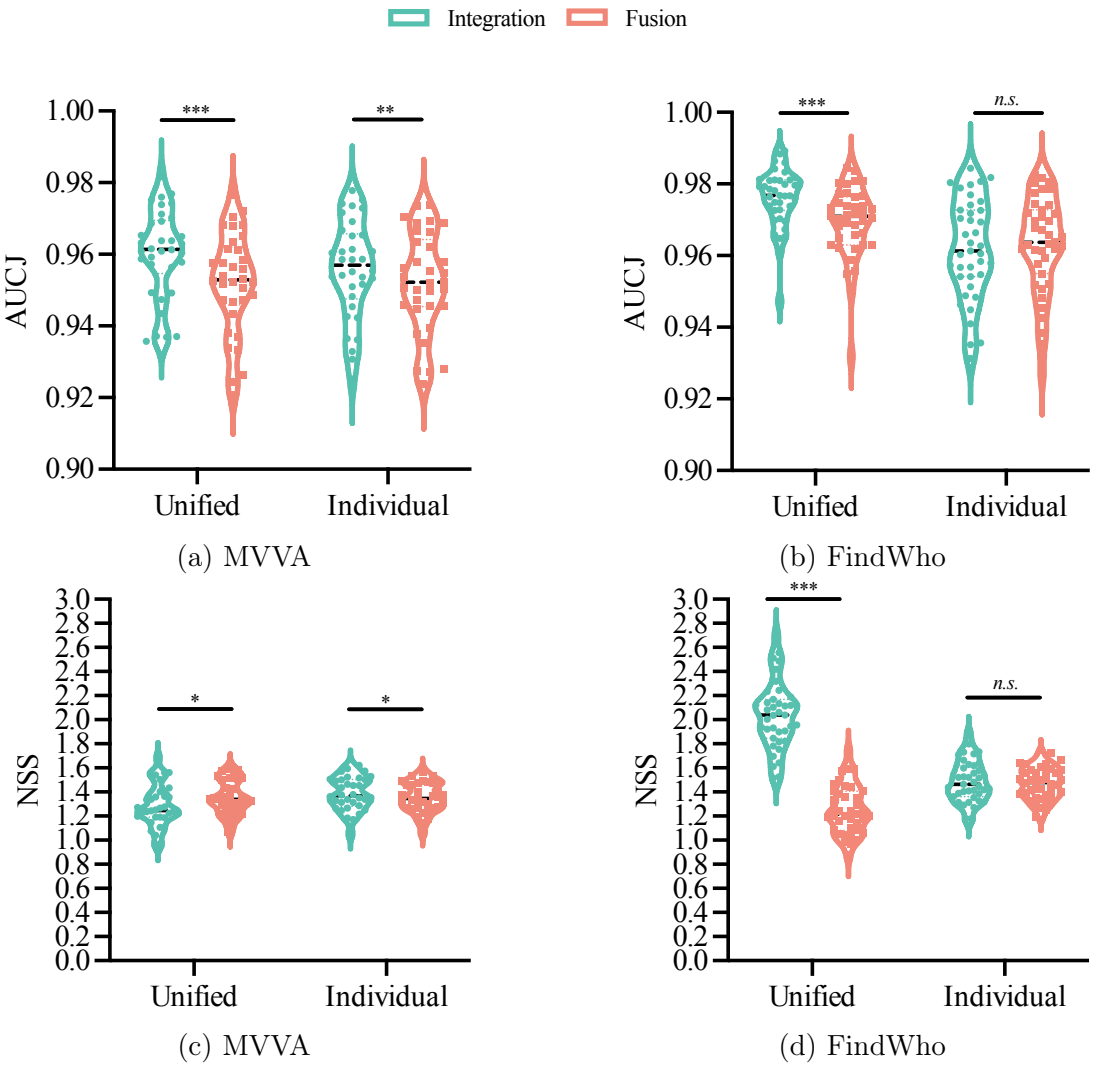
(c) Integration (d) Fusion

*** denotes $p < .001$.

Figure 4.5: The individual model *1 vs 1* and *1 vs infinity* evaluations on the MVVA [158] dataset, across the two GASP variants extended with fixation history modules. The AUCJ scores are measured for the **(a)** integration and **(b)** fusion architectures, as well as NSS scores for the **(c)** integration and **(d)** fusion architectures.

(AUCJ: FindWho $.968 \pm .002$ vs MVVA $.955 \pm .002$, $p < .001$; NSS: FindWho $1.561 \pm .023$ vs MVVA $1.344 \pm .024$, $p < .001$).

For two-factor interactions, we observed a significant interaction in the unified model which performed better with the integration architecture compared to the fusion architecture (AUCJ: unified with integration $.968 \pm .001$ vs fusion $.961 \pm .001$, $p < .001$; NSS: unified with integration $1.660 \pm .023$ vs fusion $1.302 \pm .020$, $p < .001$). The unified model significantly outperformed individual models when trained on the FindWho dataset (AUCJ: unified $.973 \pm .002$ vs individual $.963 \pm .002$, $p < .001$; NSS: unified $1.645 \pm .030$ vs individual $1.478 \pm .023$, $p < .001$). However, this difference diminished with models trained on the MVVA dataset. The integration architecture consistently yielded better results across both datasets, with particularly better



(a) MVVA (b) FindWho

(c) MVVA (d) FindWho

$*$ denotes $.01 < p < .05$, $**$ $.001 < p < .01$, $***$ $p < .001$, and *n.s.* denotes no significance.

Figure 4.6: The unified and individual model comparisons across the two GASP variants extended with fixation history modules. The AUCJ scores are measured when evaluating the models on the **(a)** MVVA [158] and **(b)** FindWho [272] datasets. The NSS scores are also measured for the **(c)** MVVA and **(d)** FindWho datasets.

performance on the FindWho dataset (AUCJ: integration $.969 \pm .002$ vs fusion $.966 \pm .002$, $p < .01$; NSS: integration $1.763 \pm .028$ vs fusion $1.360 \pm .023$, $p < .001$).

For three-factor interactions, we observed a significant interaction across the dataset, model architecture, and model ($p < .05$). The unified model performed better when trained on the FindWho dataset regardless of the architecture used in terms of the AUCJ score (integration $.98 \pm .002$ vs fusion $.97 \pm .002$). The unified model performed significantly better with the integration architecture model trained on the FindWho dataset in terms of the NSS score ($2.035 \pm .038$ vs individual $1.488 \pm .025$, $p < .001$), but individual models performed better with the fusion architecture model when trained on the MVVA dataset.

4.5.3 Social Cue Ablation

Table 4.3: The unified models of the two GASP variants extended with fixation history modules having social cue modalities ablated. The fusion and integration model architectures are trained and evaluated on the MVVA [158] and FindWho [272] datasets. The first combination group from the *top* signifies models with a single cue modality, the second combination group signifies models with two cue modalities, and the final combination group signifies models with all cue modalities included. **Bold** denotes the best scores for each combination group and dataset.

Model	**Cue**				MVVA [158]		FindWho [272]	
Architecture	IMG	SP	FER	GE	AUCJ↑	NSS↑	AUCJ↑	NSS↑
Fusion	✓	-	-	✓	0.842	0.424	0.966	1.022
Integration	✓	-	-	✓	0.925	0.820	0.967	1.294
Fusion	✓	-	✓	-	0.938	0.846	**0.968**	1.072
Integration	✓	-	✓	-	**0.954**	1.140	0.965	**1.691**
Fusion	✓	✓	-	-	0.949	0.999	0.962	0.900
Integration	✓	✓	-	-	0.928	**1.171**	**0.968**	1.519
Fusion	✓	-	✓	✓	0.945	0.860	0.949	0.805
Integration	✓	-	✓	✓	**0.951**	1.051	0.963	**1.591**
Fusion	✓	✓	-	✓	0.947	1.040	**0.968**	1.183
Integration	✓	✓	-	✓	0.932	0.580	0.951	1.489
Fusion	✓	✓	✓	-	0.947	0.926	0.801	0.121
Integration	✓	✓	✓	-	0.943	**1.104**	0.950	1.050
Fusion	✓	✓	✓	✓	0.952‡	**1.352**‡	0.969‡	1.252‡
Integration	✓	✓	✓	✓	**0.960**‡	1.283‡	**0.976**‡	**2.035**‡

IMG: Input Image; SP: Saliency Prediction; FER: Facial Expression Recognition; GE: Gaze Estimation.
‡ denotes the mean of 5 trials.

To measure the contribution of each social cue, we ablated each modality of the unified integration and fusion models independently and in combination with other modalities. We then trained the models on the MVVA and FindWho datasets separately. For this set of experiments, we report the best of 5 trial scores in Table 4.3, since the variance across trials was large and the mean value was not representative of any of the trained model scores.

When we compared the model's performance with single cue modalities, we observed that the model had the best performance with the Facial Expression Recognition (FER) modality only and the worst with the Gaze Estimation (GE) modality only. When we trained the model with two social cue modalities, we found that on ablating the GE modality, the model performance was negatively impacted with the smaller FindWho dataset but not the larger MVVA dataset. The

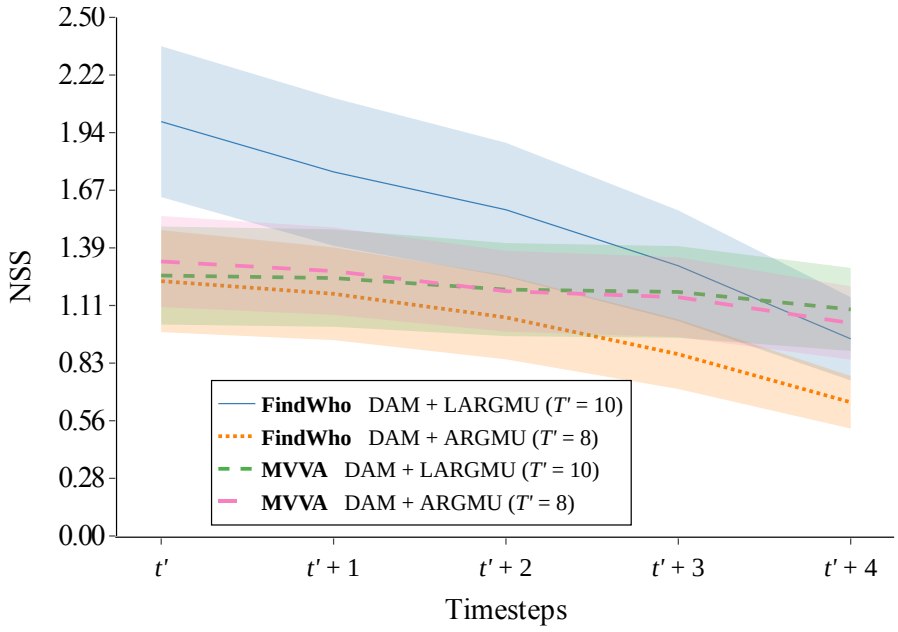
degradation in the MVVA dataset performance was due to the discrepancy between the number of faces visible in a video and those detected by the GE modality's face detector [288]. The FindWho dataset contains 30/65 (46%) videos and the MVVA dataset 142/300 (47%) videos with two or fewer faces. However, when measuring the majority (80%) of face counts per video frames detected by the GE model, the FindWho dataset had 23/65 (35%) videos, whereas the MVVA dataset had 211/300 (70%) videos with two or fewer faces. Moreover, on ablating the saliency prediction or FER modality, there was no impact on the model's performance. We also observed that the inclusion of all social cues consistently yielded the highest performance for both datasets.

4.5.4 Multi-Step-Ahead Fixation Prediction

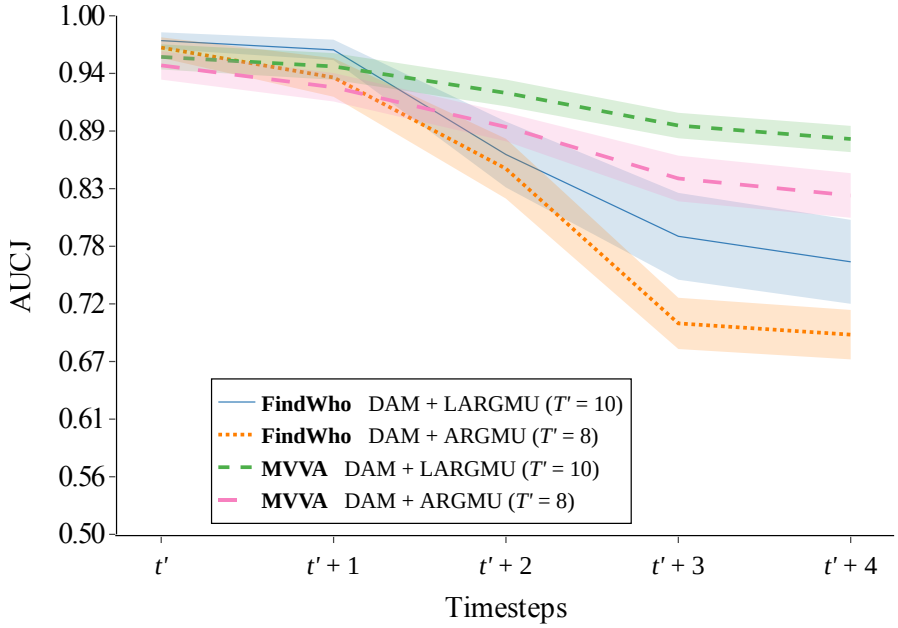
We measured the performance degradation of the two unified model architecture variants (DAM + LARGMU: integration vs DAM + ARGMU: fusion) in terms of the NSS and AUCJ scores, over the step-ahead increments $t' + n$, where $n \in \{1, \ldots, 4\}$. The evaluation was carried out on the MVVA and FindWho datasets independently. For each dataset and model architecture, we selected the top-performing model in one out of five training trials. On the MVVA dataset, the fusion model architecture showed a degradation of 22.42% in the NSS score and 13.17% in the AUCJ score by $t' + 4$. The integration model architecture exhibited smaller declines of 13.02% in the NSS score and 8.24% in the AUCJ score. More significant drops in performance were observed on the FindWho dataset. The fusion model architecture had a 47.53% decrease in the NSS score and a 28.55% decrease in the AUCJ score. The integration model architecture exhibited the highest declines with 52.42% in terms of the NSS score and 21.85% in the AUCJ score by $t' + 4$. The results indicate a trend of performance reduction over extended prediction horizons across all model variants due to the accumulation of errors. Models trained and evaluated on the FindWho dataset, particularly, showed more significant degradation. Figure 4.7 illustrates that models trained on the small FindWho dataset are less robust in predicting multiple steps ahead than when trained on the larger MVVA dataset, as indicated by the steeper negative trends in terms of both the NSS and AUCJ score. In Table A.4, we also observe that the late integration architecture tends to outperform early fusion over longer horizons.

4.6 Discussion

Studies on goal-directed human attention indicated insignificant differences in scanpaths among individuals [281, 282]. Contrary to such findings, free-viewing entails complex top-down influences, resulting in significant variance across viewing patterns among different observers [43, 53, 67]. We introduced a framework for predicting and evaluating individual scanpaths in social videos. We focused mainly on social videos due to their complexity, offering varied audiovisual cues and interactions for our scanpath prediction framework to analyze. Our main finding



(a) NSS



(b) AUCJ

Figure 4.7: The two GASP variants of the unified model, extended with fixation history modules showing a negative trend in terms of the **(a)** NSS and **(b)** AUCJ metrics, as we increase the number of steps ahead. The context size T' for each model is shown in parentheses.

was that the introduction of fixation history into the model was a sufficient prior for allowing a single unified model to predict scanpaths. However, this does not imply that the unified models can identify individual observers through their scanpath trajectories. In Section 4.5.1, we analyzed the individual models with two architectures (late integration vs early fusion) on the FindWho and MVVA datasets. We observed that when training and evaluating on the FindWho dataset, the fusion architecture performed on par with the integration architecture in terms of the NSS and AUCJ scores. Additionally, the fusion architecture exhibited lower variance across videos per participant when measuring scanpath correspondence between each individual model's prediction and its own (*1 vs 1*) as well as other observers' (*1 vs infinity*) ground-truth data. This suggests that the fusion architecture is more stable than the integration variant. Another important finding was that all model architectures performed significantly better on *1 vs 1* than on *1 vs infinity*. This indicates that each model learned the scanpath of one individual, and did not simply represent the group's attention. On the contrary, saliency prediction models would score significantly better on *1 vs infinity* (in this case, *infinity vs infinity*) than on *1 vs 1* (in this case, *infinity vs 1*) evaluations [43].

Moreover, the integration architecture's stability was contingent on the dataset. When a model architecture performs well on the FindWho dataset—which has fewer videos than the MVVA dataset but a similar distribution—it suggests that the model effectively infers patterns of attention that are shared or common across the majority of observers. This is related to understanding universal attention, where most observers converge in their attention patterns due to the limited video variability. On the other hand, with the MVVA dataset having a larger set of videos, there's a higher likelihood for individual variations in attention patterns to emerge. Thus, a model's success on MVVA can indicate its ability to discern and adapt to these individualized attention behaviors. This aligns with the notion of personalized attention, where the attention patterns might be more specific to individual inclinations, experiences, or abilities. Training and evaluating on the MVVA dataset resulted in significantly lower variance across videos compared to FindWho, in terms of NSS and AUCJ scores on both *1 vs 1* and *1 vs infinity* evaluations. As the number of observation videos increases, scanpath patterns begin to align among individual viewers. This suggests that although the models are exposed to more samples of personalized attentional behavior, on longer exposure to stimuli, universal attention exerts greater influence on viewing patterns. Consequently, integration models trained on the MVVA dataset underperform counterparts trained on the FindWho dataset, however, the variance across videos is also significantly lower. The lower variance in the case of the MVVA dataset is both due to "regression toward the mean" and bottom-up saliency affecting participants in relatively equal proportions, given its larger number of videos.

Training a separate model for each individual is costly in terms of computational resources, making it an impractical approach for training models on larger datasets with many observers. To overcome such a limitation, we devised a unified training approach, whereby the model is exposed to scanpaths of all individuals during training. Each scanpath is fed into the unified model in the same manner as it was

for the individual models, with the main difference being the sampling strategy. The individual models are only fed samples from a single observer during training. However, the unified model samples a random observer's scanpath, and is trained with the corresponding stimuli and fixation histories.

In Section 4.5.2, we compared unified and individual models. The unified model performed significantly better than the individual models. This improvement can be due to the fact that the unified model is trained on data from all observers, subjecting it to greater variability in the samples. As a result, the unified model is exposed to a larger spectrum of traits relating to universal attention. More importantly, the unified model, regardless of the architecture, can predict different scanpaths given the same stimuli, conditioned only on the fixation history. This is evident from the scores being comparable to the individual models, which represent the baseline for acceptable performance. According to these results, we infer that the fixation history is a sufficient prior since:

1. Scores of the unified model are on par or better than those of the individual models.

2. The fixation history is the only prior available to the model for it to differentiate scanpath trajectories. Without it, the model would generate arbitrary scanpaths, consequently performing significantly worse than the individual models.

In Section 4.5.3, our results showed that including all social cue modalities improved the performance of our models. Moreover, ablating the gaze estimation modality degraded the performance of both model architectures when trained and evaluated on the smaller FindWho dataset, yet had negligible effect on the larger MVVA dataset. The gaze estimations were represented as gaze cones, superimposed on the face positions of actors visible in a video frame. These cones were oriented according to the estimated gaze direction of each actor, after which they were normalized and aggregated. Having more than two gaze cones—more than two faces detected—in any frame distorts the gaze estimation representation. We therefore assume that occurrences of two or fewer face detections are optimal for gaze estimation. The MVVA dataset was found to result in more detections (70%) of videos with two or fewer faces than the FindWho dataset (35%), even though both datasets contained $\sim$46% videos with two or fewer faces. This implies that gaze estimation representations of the MVVA samples were inaccurate, resulting in our models relying less on that social cue.

Predicting one step ahead for each scanpath is useful only when the model has access to the ground-truth fixation history of an individual. In practice, however, this requirement renders a model unusable for most applications. Kümmerer and Bethge [146] present an evaluation framework that addresses this limitation. By feeding the output of a model recursively into its fixation history module, we can evaluate the model for multiple steps ahead without relying on the ground-truth fixations (as input) beyond the initial steps. This form of evaluation

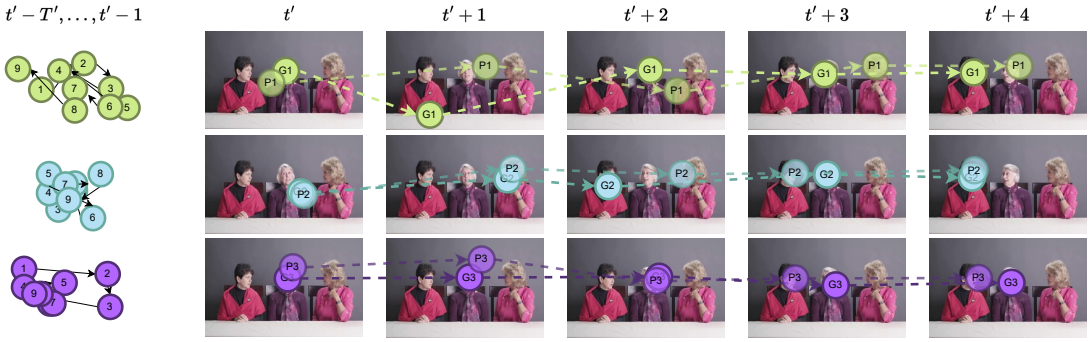


Figure 4.8: Our unified late integration model (DAM + LARGMU, context size $T' = 10$) predicting scanpaths of three observers (G1 - G3). At each timestep of multi-step-ahead (five steps) fixation, predictions (P1 - P3) indicate that the slightest divergence from the ground-truth has an impact on future predictions. The fixation history $t' - T', \ldots, t' - 1$ *(left)* is fed into the model during inference and training.

informs us on the performance of a model during inference, and how likely the output predictions are to diverge from the ground-truth over multiple steps.

In Section 4.5.4, we followed a similar approach in evaluating our unified models. Results indicated that although the unified integration model was not the most performant under one-step-ahead evaluation, it exhibited higher robustness compared to the fusion architecture as the number of steps ahead increased. This result implies that the late integration of cue representations, namely, applying recurrence before gating, is more beneficial for learning sequences, and vice versa. Moreover, training on the MVVA dataset stabilized the predictions of a model over longer horizons, as shown by the least reduction in scores over multiple steps ahead. We hypothesize that MVVA-trained models are exposed to more samples and therefore are less affected by the accumulation of errors as the number of steps ahead increases due to higher variability in the samples.

In Figure 4.8, we show the scanpaths predicted for three individual observers over multiple steps ahead. At any timestep, inaccurate predictions resulted in increasingly different fixations from the observer's ground-truth fixations at the corresponding steps ahead. The fixation of the first observer from the top in Figure 4.8 was predicted with an offset from the ground-truth at t'. Since the fixation at t' was appended to the fixation history, it led to a major divergence in subsequent predictions. Nonetheless, the predicted scanpaths of observers were closer to their own than to those of others.

Our findings, however, do not imply that our unified models could identify individuals from their scanpath trajectories. Individuals' scanpaths can be separated into clusters [67], indicating that although they are distinct from the group, they do not differ significantly with respect to all other individuals independently. Since the fixation history—covering no longer than a few seconds of low-dimensional data—allows the unified model to perform on par with the individual models,

we deduce that the viewing patterns of individuals are not significantly different. Differences in past scanpaths do influence subsequent fixations, but the unified model still ensures privacy because it does not tie predictions to specific observers, unlike the individual models where the observer is known a priori.

Overall, we introduced a mechanism for integrating fixation history into dynamic video models, designed to personalize scanpaths to a specific observer. A comparative analysis with existing models could be considered for measuring performance, yet many factors limit our ability to do so. For instance, an essential component for enabling the personalization of scanpaths is the fixation history. However, most existing dynamic models do not include a fixation history module. Moreover, dynamic models that encode fixation history [185, 276] are commonly designed for scanpath prediction in 360° videos, and adopt approaches such as image patching—splitting the visual frames into smaller segments to simulate foveation—that make them unsuitable for the dataset videos used in this work, due to their limited resolution. Additionally, adapting these models would require major modifications to their architectures and tasks, ranging from introducing a fixation history module, performing hyperparameter optimization, to retraining or fine-tuning the models on the FindWho and MVVA datasets. Consequently, any adapted models would deviate from their implementations, resulting in the creation of new models. Given these constraints, we evaluated our approach using our own models with different integration architectures and ablation studies, providing baselines for future comparisons.

In future work, we will extend our dynamic scanpath prediction model to handle the non-deterministic nature of eye movements. We intend to integrate techniques from static scanpath prediction models, particularly Generative Adversarial Imitation Learning [281] and Reinforcement Learning [53], adapting these methods for dynamic video inputs. We will also study the effect of introducing further auxiliary social cue representations into the model, such as full-body gestures, biological motion, intonation, and prosodic features.

Part II

Controlling Social Robots

Chapter 5

A Framework for Message-Oriented and Robotics Middleware

Our goal is to provide the means for communicating between neural network models and robots in order to facilitate the control of gaze on robotic platforms. However, robotic platforms support different middleware [170, 206, 163], each with its own API, communication properties, communication patterns, and limitations. Moreover, the integration of deep-learning models and external components into robotic pipelines requires modifications to every module involved. Since our goal is to provide solutions for controlling robot gaze, we do not want to restrict our approaches to a certain platform, robot, or model. Therefore, in this chapter, we introduce Wrapyfi, a Python framework designed specifically to be modular and non-opinionated, requiring minimal modifications to existing code bases. Wrapyfi supports multiple message-oriented and robotic middleware, as well as other communication patterns and schemes. Wrapyfi also provides plugins for deep learning, computer vision, and mathematical frameworks, enabling direct exchange of their data types without having to encode and decode them on transmission.

5.1 Introduction

Real-time robotic applications require exchanging multimodal data arriving from a variety of sensors. A framework that distributes sensory information across processes is necessary, especially for robot-robot and human-robot interaction [180]. Multiprocess and multithread instances are used to parallelize independent methods. However, such parallelization approaches are limited to single machines and may not be sufficient for applications with a large number of sensors or computationally expensive processing methods. Eventually, this leads to performance bottlenecks on consumer-grade computers. Message-oriented and robotics middleware, such as ZeroMQ [113], YARP [170], ROS [206], and ROS 2 [163], were developed to tackle such challenges. Middleware frameworks use communication protocols to exchange data and distribute operations across several machines and nodes [75].

ROS [206] is a middleware commonly used in the robotics community. ROS

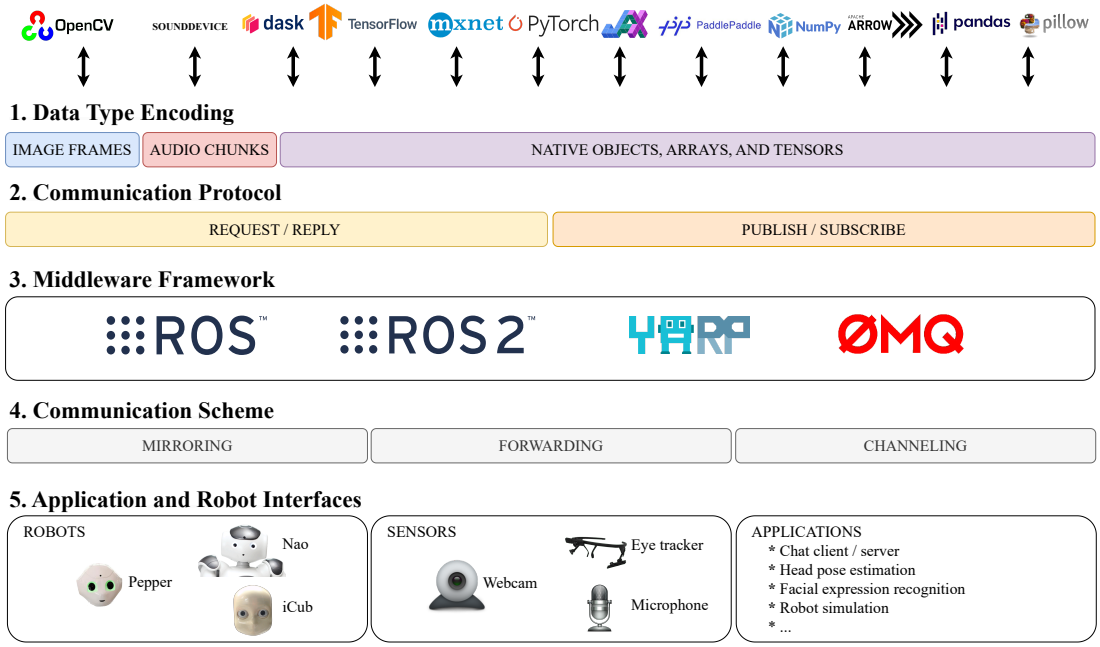


Figure 5.1: Overview of our Wrapyfi framework. From *top* to *bottom*: 1) Data types are encoded or decoded depending on the transmission mode; 2) Encoded objects are prepared for transmission using the Request/Reply or Publish/Subscribe communication pattern; 3) Messages are transmitted through the selected middleware protocol; 4) Messages sequenced according to the communication scheme; 5) Messages exchanged between robots, applications, and sensors.[1]

provides control hardware interfaces, visualization tools, and communication models for many robotic platforms [8]. Its widespread use is a direct result of its early adoption of open source and the vast amount of robotic tools provided by its developers and contributors. However, ROS is scheduled for deprecation in favor of ROS 2 [163]. Many robotic platforms and packages, nonetheless, have not been updated to support this transition yet. Cross-platform bridges were developed to enable communication between ROS, ROS 2, and WebSocket. However, integrating such bridges into existing pipelines entails major modifications to the underlying code and its structure. Middleware designed for certain robotic platforms, for example, YARP [170] which was developed specifically for the iCub [171] robot, provides a interface ROS [186] as well. However, the usage of such bridges dictates modifying the existing scripts to accommodate specific message types. This poses a hurdle for developers aiming to integrate robots and middleware, as a result, restricting the cross-compatibility of their applications with existing systems.

[1]The "nine dots" ROS and ROS 2 logos are trademarks of Open Source Robotics Foundation. TensorFlow, the TensorFlow logo, and any related marks are trademarks of Google Inc. The OpenCV logo is a trademark of https://opencv.org. The NumPy logo is used in accordance with the NumPy logo guidelines. The pandas logo is used in accordance with the brand and logo guidelines. PyTorch, the PyTorch logo and any related marks are trademarks of The Linux Foundation. The name ZeroMQ and the "ØMQ" logo are used in compliance with creative

To improve interoperability between different robotic platforms and reduce reliance on a particular middleware, we have developed the open source Wrapyfi[2] (overview illustrated in Figure 5.1) framework, a Python wrapper supporting multiple middleware bindings. Wrapyfi is a simpler alternative to GenoM3 [165]. GenoM3 adopts a model-driven approach and uses templates to define the components and data exchanges across middleware. Since it is specifically developed for Python scripting, Wrapyfi eliminates the need for having to learn another language or to define templates, unlike GenoM3. REMS [241] is a middleware built in Python with simplistic interfaces for educational purposes. Although REMS supports a large set of robots and simulation environments, it does not address interoperability between different middleware operating on them.

Wrapyfi's decorator-based design integrates easily with existing workflows, prioritizing minimal modifications for improved multi-robot communication. Beyond robotic applications, its adaptability is observed in supporting message-oriented middleware, facilitating communication even with interfaces that do not necessarily include the additional packages and tools provided by robotics middleware. deep-learning frameworks like JAX [36] and PyTorch [199], support multi-machine parallelization mainly through remote procedure calls. The approaches adopted in distributing models and data differ greatly, including the communication patterns used and the orchestration of communication, having either a single or several controllers. By offering a standard approach for multiple frameworks, and supporting two of the most common communication patterns, namely publish-subscribe and request-reply—also known as the request-response or client-server pattern—Wrapyfi offers greater control over communication dynamics in comparison to each framework's parallelization protocol.

Open Neural Network Exchange (ONNX) [20] is a deep-learning framework designed to standardize model structure and configuration, allowing for cross-compatibility with a wide range of deep-learning frameworks. However, using ONNX with any existing framework requires adapting the model formats. In contrast, Wrapyfi does not enforce such a constraint. Moreover, it not only allow for native Python object exchanges but also transports data structures such as arrays and deep-learning framework-specific tensors. This also makes Wrapyfi useful for developers wanting to create prototype applications, where they could take advantage of both robotics and deep-learning ecosystems.

[2] https://github.com/fabawi/wrapyfi

5.2 Communication Patterns and Middleware

The *publish-subscribe* pattern is a messaging communication pattern commonly used in distributed systems, where the message publisher is not pre-programmed to send its messages to specific receiving subscribers. Instead, the published messages are categorized into classes. Subscribers receive messages published to the classes without interacting directly with the publisher . The decoupling of publishers and subscribers allows for more flexibility as the number of subscribers grows. In Wrapyfi, th publish-subscribe pattern enables different nodes to communicate asynchronously, allowing nodes running on machines with vastly different capabilities to communicate at their own rates, thus enabling the the distributed of compute across heterogeneous systems.

The *request-reply* pattern, is a communication pattern for remote procedure calls, by which a requester sends a request message to a replier, and the replier responds back with another message. This pattern is usually synchronous, meaning that the requester blocks all its operations up until it receives a reply. However, asynchronous implementations also exist where the requester can perform other tasks while awaiting a reply. Wrapyfi currently exclusively supports the synchronous variant of this pattern. However, synchrony is usually a requirement when the communication between two endpoints is expected is contingent on the exchange of information between those two end-points. Moreover, Wrapyfi provides a common interface for selecting any of the supported middleware, including:

YARP [170] is a middleware for robotics and data exchanges. It provides support for hardware abstraction, device control, and communication. YARP is designed to be flexible and interoperable with other middleware, allowing developers to integrate it into their systems easily. YARP supports a wide range of communication protocols, including TCP, UDP, and shared memory, making it suitable for various applications. To use YARP with Wrapyfi, the `yarpserver`, which is the name server enabling port discovery, must be running.

ROS [206] is a common open-source middleware in robotics. ROS is structured to support modularity, allowing developers to reuse and combine different components using standardized message types. To use ROS with Wrapyfi, the `roscore` must be running. The `roscore` starts the name server and provides the central point of communication for ROS nodes, allowing them to exchange messages and data. Wrapyfi additionally supports custom ROS message types, which allows for communication with robots without adjusting their underlying library code.

ROS 2 [163] is a remodel of ROS, introducing enhanced security and support for industry-level communication patterns. Many of the core concepts from ROS are carried over to ROS 2, making it easier for developers familiar with ROS to transition to ROS 2. Unlike ROS, a server is not required to run ROS 2. Based on the Data Distribution Service (DDS) [197], ROS 2 provides low-latency communication, making it a suitable middleware for applications where fast and reliable data transfer is necessary. ROS 2 supports multiple distributions of DDS. Wrapyfi is

not designed with support for specific DDS distributions, nor is it explicitly tied to specific ROS 2 versions. However, we introduce custom ROS 2 message types for allowing Wrapyfi to communicate over ROS 2. These messages are compiled independently and would require further modification to their configurations in order to support message exchanges for different ROS versions.

ZeroMQ [113] is a messaging library that supports many communication patterns and protocols, making it a conveniant option for distributing applications across different systems. Wrapyfi provides a proxy broker implementation for ZeroMQ, allowing reliable message exchanges between mirrors. This implementation uses a polling mechanism to transfer messages between script mirrors. Currently, Wrapyfi exclusively supports TCP-based communication with ZeroMQ, as only the experimental radio-dish pattern supports UDP communication. Other pattern-specific communication protocols, such as PGM and NORM, have also not been integrated with Wrapyfi.

5.3 Wrapyfi Features

Configuring methods and middleware. Wrapyfi enables the adjustment of method behavior through either script or configuration files. The Wrapyfi `activate_communication` method allows users to establish the communication pattern between nodes by selecting appropriate modes for different patterns, such as `'publish'` or `'listen'` for the publish-subscribe pattern, and `'request'` or `'reply'` for the request-reply pattern. These configurations determine how messages are processed or returned over the middleware. Moreover, configuration files can be utilized to define properties for each system, which results in triggering the corresponding configurations for a specific middleware (e.g., ROS or YARP) on the first call to a method using that middleware. This eliminates the need to pass verbose configurations to all the `register` decorated methods, which would lead to unexpected behavior if the order of method calls changes or if there are differences in the arguments passed to the method decorators.

Blocking and non-blocking methods. In a publish-subscribe pattern, the `should_wait` argument determines whether the publisher waits for subscribers. By default, a method is non-blocking. When `should_wait` is `True`, the publisher waits for a subscriber to listen before messaging. Conversely, a subscriber blocks until a message is sent if `should_wait` is `True`. In request-reply patterns, both client and server are blocking, awaiting requests and replies.

Serializing device-specific tensors. Wrapyfi supports selecting a device or automatically mapping it from the received tensor. This allows users the option to allocate tensor listener decoders to specific devices, such as CPUs or GPUs, facilitating direct GPU|TPU mapping and re-mapping on mirrored nodes. This mechanism is supported for MXNet, PyTorch, and PaddlePaddle tensors, facilitating the adjustment of device allocation according to user preferences. For example,

in MXNet, specifying `map_mxnet_devices={ 'cuda:0': 'cuda:1' }` enables users to manually relocate device tensors mapped originally to GPU 0 onto GPU 1 on reception.

5.4 Data Types

Wrapyfi employs a type-aware serialization method that automatically transforms the objects exchanged between script mirrors into a format compatible with the selected middleware and library.

5.4.1 Native Object, Array, and Tensor Messages

Wrapyfi transmits many data types defined by popular Python frameworks and libraries. Before transmission, these data types are converted into JSON strings. Wrapyfi supports NumPy [106] arrays and enables their sharing across mirrored scripts. Moreover, Wrapyfi offers a plugin interface that can be used to customize the transmission of other types of objects other than the ones originally supported by Wrapyfi. The plugins feature allows encoding objects as strings, which can eventually be decoded back into their structure before encoding. Wrapyfi comes with built-in plugins for exchanging Arrow [213] vectors, pandas [168, 246] data frames, and Pillow[3] images. It also supports tensors from major deep-learning frameworks such as TensorFlow [2], PyTorch [199], MXNet [52], JAX [36], PaddlePaddle [162], and Dask [214]. These plugins make it possible to exchange data between different frameworks and to integrate deep-learning models into robotic systems. When specified, the tensors transmitted using Wrapyfi can be mapped to GPUs or CPUs different from the ones specified on a publishing script's end, allowing for the distribution of computationally demanding deep-learning models.

5.4.2 Image Messages

ROS, ROS 2, and YARP provide specialized message types for transmitting images. We use image messages to stream raw monochrome, RGB, and JPEG-encoded images. ZeroMQ does not provide such specialized message structures. Therefore, we make use of the multipart message structure to create an image interface, allowing us to standardize middleware behavior and transmit the image properties to a specified topic.

5.4.3 Audio Chunk Messages

ROS and ROS 2 do not provide messages structured for audio transmission, so we create custom messages and services to transmit audio along with its properties. The number of audio channels transmitted can vary in size, as long as the audio chunk structure follows the python-sounddevice format[4]. For YARP, we use the

[3] https://github.com/python-pillow/Pillow
[4] https://github.com/spatialaudio/python-sounddevice

existing sound port and transmit the audio as a sequence. Whereas, for ZeroMQ, we transmit a string, encoding the auditory signal along with its properties as a single multipart message.

5.5 Communication Schemes

Wrapyfi manages script interactions by introducing three communication schemes—Mirroring, Forwarding, and Channeling. Unlike communication patterns, schemes are agnostic to the communication protocol or blocking mechanism, and are rather interfaces along with guidelines for implementing concurrent functionality without introducing additional handlers. Mirroring enables concurrent execution of multiple scripts with synchronized actions. Forwarding creates chains of methods to tunnel arguments and return values across different middleware configurations. Channeling allows for the broadcasting of multiple return values via one method, each using potentially different middleware. Each scheme addresses a different set of challenges.

The `MiddlewareCommunicator` is a Wrapyfi class for establishing communication methods. It implements the `register` decorator for setting the middleware types, topics, and various communication parameters. Each method set to publish, subscribe, request, or reply should be encapsulated within this decorator. Listing 5.1 illustrates the use of the `register` decorator to register a method for YARP middleware communication, specifying object type, middleware, name of the class, YARP port (topic), communication protocol, and whether the method should await a response, which results in blocking the subscribing method until the publisher transmits a message. The `read_msg` method obtains user input from one process, allowing all other subscribing processes to acquire user input from a single process.

Listing 5.1: Decorated method registering the data type, middleware, topic, connection protocol, and blocking behavior. `'$0'` passes the first argument (`mware`) from the method to the decorator. Similarly, `'$blocking'` passes the keyword argument.

```
class MirrorCls(MiddlewareCommunicator):
  @MiddlewareCommunicator.register('NativeObject',
   '$0', 'MirrorCls', '/example/read_msg',
   carrier='tcp', should_wait='$blocking')
  def read_msg(self, mware, msg='', blocking=True):
    msg_ip = input('type message:')
    obj = {'msg': msg, 'msg_ip': msg_ip}
    return obj,
```

In Listing 5.2, setting the mode to `'publish'` triggers `read_msg` upon method call, whereas `'listen'` returns the message received over the middleware. These modes enable the establishment of communication following the publish/subscribe pattern. Alternatively, setting the `activate_communication` mode to `'request'` or `'reply'` triggers the request/reply pattern.

Listing 5.2: Activating a method in `'publish'` mode. When the method is called, its results are returned to the caller and transmitted to the listener.

```
mirror = MirrorCls()
mirror.activate_communication(
    'read_msg', mode='publish')
```

5.5.1 Mirroring Scheme

Mirrors are identical scripts running simultaneously within different processes. These scripts share their arguments and return values, after which they execute the same pipeline. However, their methods could either run a specific functionality in place or could acquire their return values from another publisher. By calling `read_msg` in Listing 5.1 using a single publishing script, all subscribing mirrors receive the same return object when invoked as well. Regardless of the communication pattern or blocking behavior, all scripts follow the same pipeline with similar method returns.

5.5.2 Forwarding Scheme

Listing 5.3: Demonstration of forwarding with two methods each using a different middleware.

```
class ForwardCls(MiddlewareCommunicator):
  @MiddlewareCommunicator.register('NativeObject',
   'yarp', 'ForwardCls', '/example/native_yarp_msg',
   carrier='mcast', should_wait=True)
  def send_yarp(self, msg):
    return msg,

  @MiddlewareCommunicator.register('NativeObject',
   'zeromq', 'ForwardCls', '/example/native_zmq_msg',
   carrier='tcp')
  def send_zmq(self, msg):
    return msg,
```

The forwarding scheme in Wrapyfi is designed for passing arguments across multiple methods, each with a employing a different middleware. With forwarding, we form chains of methods, each passing the arguments and return values within the script, and transferring the acquired and transmitted values to other methods with different middleware and topics. Forwarding works by assigning unique functionionality to each script. The scripts contain bridging methods that are connected by `register` decorators, which share middleware and topic. These bridging methods exchange data across multiple middleware, and acquire whichever data they receive depending on the available or enabled middleware. In Listing 5.3, we demonstrate data transmission between a system without ZeroMQ support and another without YARP support, using an intermediary system that supports both. The first system dispatches the message using YARP by invoking `send_yarp`. The

intermediary system then forwards it using ZeroMQ to `send_zmq`. The final system, with YARP disabled, receives the message via ZeroMQ by listening to `send_zmq`. This scheme is needed when strict specifications are required regarding the compatibility of software and middleware between systems, as in the case of robots.

5.5.3 Channeling Scheme

In the channeling scheme, Wrapyfi enables broadcasting to multiple middleware by encapsulating a method with numerous decorators, each corresponding to a return value with its own data type and middleware. The number of a method's returns defines the number of decorators specified for that method, following the same order of definition, i.e., the first decorator defines the communication properties of the first return, the second decorator for the second return, and so on. This is illustrated in Listing 5.4, where a method transmits three different data types over varied middleware, such as a YARP native object message comprising a NumPy image and an audio chunk, a ROS image (OpenCV [37] compatible), and a ZeroMQ audio chunk. This scheme supports the simultaneous reception of different data types. Should an environment lack support for a specified middleware, a `'None'` type object is returned instead. Channeling is especially useful for handling multiple sensory inputs from different sources, allowing selective acquisition and disregard of unnecessary sensory input. The channeling scheme can be considered a combination of both mirroring and forwarding. However, we provide this scheme to avoid forcing developers to adapt their existing methods to accommodate Wrapyfi's model, which adopts a non-opinionated design.

Listing 5.4: Demonstration of Channeling with one method reading multiple returns of different data types through multiple middleware.

```
class ChannelCls(MiddlewareCommunicator):
  @MiddlewareCommunicator.register('NativeObject',
   'yarp', 'ChannelCls', '/example/native_yarp_msg',
   carrier='mcast', should_wait=True)
  @MiddlewareCommunicator.register('Image',
   'ros', 'ChannelCls', '/example/image_ros_msg',
   carrier='tcp', width='$img_width',
   height='$img_height', rgb=True, queue_size=10)
  @MiddlewareCommunicator.register('AudioChunk',
   'zeromq', 'ChannelCls', '/example/audio_zmq_msg',
   carrier='tcp', rate='$aud_rate',
   chunk='$aud_chunk', channels='$aud_chann')
  def read_mulret_mulmware(self,
   img_width=200, img_height=200,
   aud_rate=44100, aud_chunk=8820, aud_chann=1):
    ros_img = np.random.randint(256,
     size=(img_height, img_width, 3), dtype=np.uint8)
    zeromq_aud = (np.random.uniform(-1,1, aud_chunk),
     aud_rate,)
    yarp_native = [ros_img, zeromq_aud]
    return yarp_native, ros_img, zeromq_aud
```

5.6 Experimental Setup

Wrapyfi encodes and decodes data types that are employed by commonly used scientific computing, image processing, and deep-learning frameworks. To assess the overhead introduced by the encoding and decoding mechanisms, we transmitted these object types using the supported middleware—ROS, ZeroMQ, YARP, and ROS 2. The evaluation was carried out in the publish-subscribe mode on the same machine with an Intel Core i9-11900 running at 2.5 GHz, with 64 GB RAM and an NVIDIA GeForce RTX 3080 Ti GPU with 12 GB VRAM.

5.7 Results

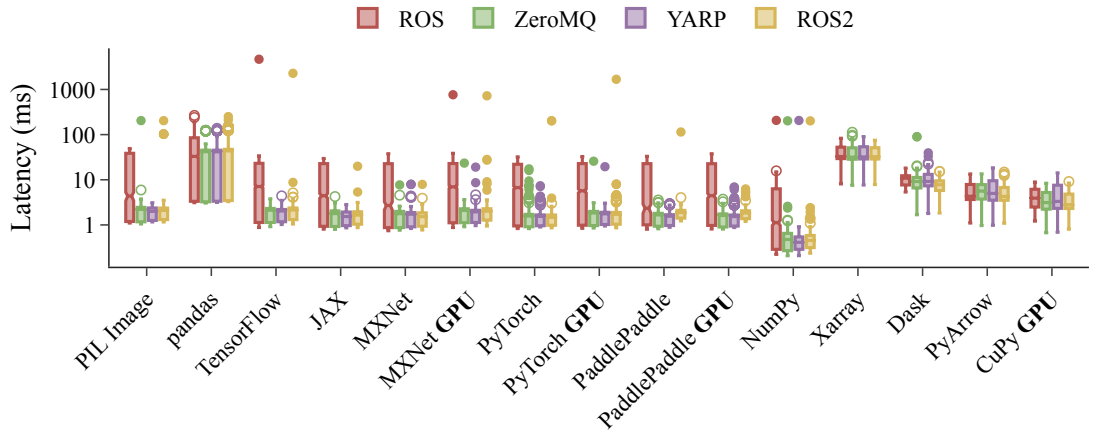


Figure 5.2: Latency between publishing and receiving 200×200 tensors of ones transmitted using each middleware independently with blocking methods. 2000 trials are conducted with a publishing rate of 100 Hz for each middleware and plugin combination. Latency indicates the time difference between transmission and reception including de/serialization.

The results in Figure 5.2 show that NumPy array transmission resulted in the lowest latency compared to other data types. Variances in performance were most significant with the ROS middleware, which also resulted in the highest latency on average. The ROS Python bindings serialize messages natively, resulting in additional overhead. GPU tensor mapping to memory showed insignificant delay compared to memory-mapped counterparts in the case of MXNet, PyTorch, and PaddlePaddle. pandas data frames were transmitted with the highest latency[5]. Compared to NumPy, pandas provides a vast range of tools for statistical analysis and data filtration, making it a more favourable framework for data scientists. However, since pandas relies either on NumPy or pyArrow as a backend, the latency of encoding and decoding its data types is constrained by the backend.

[5]pandas version 1 with NumPy as a backend

Additionally, the data structures of pandas objects have to be encoded into a structures compatible with those of the backend, accumulating the computational overhead.

5.8 Discussion

Wrapyfi is a framework that simplifies data transfer across different middleware platforms. Two of Wrapyfi's key strengths are the transmission of custom data types and support for multiple middleware. We introduced three communication schemes—mirroring, forwarding, and channeling—each serving a different set of applications. The framework currently supports two common communication patterns: publish-subscribe and request-reply. In future work, we plan to extend Wrapyfi to support more communication patterns available in some middleware platforms, such as actions in ROS 2, which are similar to asynchronous request-reply. We also aim to provide interfaces for custom messages and middleware-specific data types.

Wrapyfi's non-opinionated design provides the flexibility required to construct complex pipelines, extending from parallelized scripts to robotic interfaces. In Chapter 6, we demonstrate how the mirroring, forwarding, and channeling schemes can be applied in practice. In Chapter 7, we further demonstrate how Wrapyfi can be used to conduct robotic experiments that require precise scheduling and communication between multiple deep-learning models. Having a framework that can encode messages from all the platforms needed meant that our models and robot interfaces were not tied to a specific middleware. Also, distributing the deep-learning models that interacted with the robots was made simple, as the scripts could be mirrored, while some functionality could be switched off on machines with limited computational capability and delegated to more powerful systems.

Chapter 6

Social Human-Robot Interaction

In this chapter, we present two human-robot-interaction studies addressing the display of social cues on robots. Both studies are developed using Wrapyfi (detailed in Chapter 5). The first study explores the influence of a robot's social cue expression, namely gaze direction and affect, on human-human collaboration. In the second study, we assess humans' perception of a robot mirroring their gaze direction and affective signals in real-time settings. Our primary objective is to adopt the methods by which social robots are commonly evaluated as evidence of Wrapyfi's applicability and utility in practice. Moreover, we aim to study how robot social cue expression is perceived by humans, and whether the integration of robots into social settings affects the course of interactions.

6.1 Human-Human-Robot Collaboration

6.1.1 Introduction

Collaboration is a fundamental aspect of human social behavior, which plays an important role in achieving common goals and solving problems [193]. However, collaboration can be compromised by conflicts that may arise [123]. Conflicts could be task-related, where differences arise in viewpoints and ideas between collaborators [122]. Another form of conflict could be relationship-based, arising from tension or animosity between collaborators or team members [122]. One potential mitigation to task and relationship conflict is the integration of humanoid robots into human collaboration settings. Humanoid robots can potentially assist humans in enhancing their collaboration skills [128]. For instance, they can be useful in engaging children with autism spectrum disorders [238] or in reducing task conflicts during collaboration [226]. However, while robots can be helpful in such interaction scenarios, they may only have limited influence due to the complexity of social dynamics, where personal traits and cultural differences could lead to different interpretations of an expressed social cue.

These social cues provide vital information that allows us to understand social norms, establish trust, and form positive relationships. Humans are more likely to collaborate and work together toward common goals when they perceive positive

social cues. Negative social cues, on the other hand, can lead to mistrust, conflict, and a breakdown in collaboration [253]. As a result, social cue expression is essential for shaping human behavior and influencing collaboration [242]. To examine the validity of such influences in real-world scenarios where robots are involved, we design a human-human robot collaboration game. Rather than a typical Human-Robot Interaction (HRI) setting, where humans tend to adjust their behaviors and expectations to accommodate the robot, we instead design our study to prioritize the human-human collaboration, involving the robot as an additional social entity. This is known as *triadic HRI*, and through it, we maintain the naturalness of the interaction while still influencing the collaboration through the robot's social cue expression.

Eye gaze and facial expressions are particularly important social cues in HRI and collaboration. Humanoid robots are designed to simulate human behaviors and express emotions through movement and facial expressions. When robots display positive social cues like maintaining eye contact, nodding, and smiling, they can establish a connection with humans and gain their trust [18]. This can lead to more effective human-robot collaboration in a variety of tasks, including manufacturing, healthcare, and education [247]. Furthermore, social robots can facilitate human-human or human-robot collaboration by signaling humans—signaling is the act of displaying indications or performing gestures to guide behavior—and acting as mediators during the interaction [15]. Overall, social cues in HRI can enhance communication, making their integration into robots a positive design choice.

In collaborative settings, a humanoid robot that expresses social cues can influence human interaction by reducing personal biases, managing conflicts, and improving efficiency, encouraging constructive discussion and collaboration [188]. Robot gaze direction significantly influences human decision-making and perception. Kompatsiari *et al.* [142] study the effects of mutual and non-mutual robot gaze. Their findings reveal that participants attribute greater engagement and humanlike traits to a robot that establishes eye contact. Another study shows mutual gaze between robots and humans to influence the latter's decision-making time [25]: Participants were slower at making decisions when the iCub [171] robot established eye contact with them. Neural activity in the brain evoked by the robot's gaze direction draws similarity to gaze influences observed during social interactions with other humans, indicating that the robot gaze direction has a similar effect as human gaze direction [25]. Moreover, eye contact with robots elicits physiological changes associated with positive affect and higher attention allocation [136]. Gillet *et al.* [98] investigate how a social robot could use adaptive gaze behavior to balance the participation of native speakers and second language learners in a game. These results show that the robot's gaze direction could influence interaction among players, leading to an even (equal) contribution in participation between them.

Robot emotional cues, whether through speech, gestures, facial expressions, or other indications of affect, alter humans' perception of the robotic agent [225]. Their mental states are also influenced through emotion contagion [187]. Reyes *et al.* [211] study how a humanlike robot's negative facial expressions on failing to complete the task, affects human-robot collaboration. The task involves placing ten objects

within a container in collaboration with a robot. The authors [211] show that the robot's expression of sadness signaled a need for human help and consequently improved task performance. In follow-up work [212], the authors suggest that negative facial expressions signaling failure attract humans' attention and lead them to collaborate better.

Unlike previous studies that focus on direct HRI, our main objective is to evaluate the influence of non-verbal cues, namely, robot facial expressions and gaze behavior, on triadic HRI. We hypothesize that this can lead to a better understanding of how a robot can facilitate social dynamics among humans without interfering with or interrupting their verbal communication. A robot, as an observer, can also elicit different responses from humans depending on how they perceive the robot's humanlikeness, intelligence, and intentionality. Therefore, studying how a robot can use non-verbal cues to modulate human-human interaction guides us in designing social robots that can improve human collaboration.

Although previous studies indicate that robot gaze and facial expressions have an impact on HRI, limited research has explored the impact of humanoid robots in triadic—human-human-robot — collaboration scenarios. Moreover, few studies investigated the interaction effect between gaze direction and facial expressions from robots on human collaboration behaviors. Thus, in this study, we design a collaborative game between two human participants, with the objective of inserting objects into a shape sorter [92]. One participant serves as a guide, giving instructions to the other participant, who acts as an actor by placing occluded objects in the sorter. A humanoid robot is incorporated into the setting, displaying facial expressions while directing its gaze toward either the actor or guide.

This research explores robots' potential as collaborators in human-human team settings and their ability to communicate effectively through non-verbal cues. However, we do not aim to study whether a robot's presence can influence human-human interaction. Instead, we assume it to be present and investigate how, and to what extent its non-verbal social cues—especially facial expressions and gaze communication—influence human-robot triadic collaboration and human perception. More concretely, we aim to answer the following Research Questions (RQ):

RQ6.1.1 *How can a robot's facial expressions and gaze communication impact triadic collaboration?*

RQ6.1.2 *How do humans perceive the intelligence of a robot during triadic collaboration? Is it consistent with humans' general impressions of the robot?*

from which we derive the following hypotheses:

H6.1.1 The robot's positive facial expressions will improve human-human collaboration performance compared with neutral facial expressions. A mutual gaze between the guide and the actor could impact the performance of the task differently. There may be an interaction effect on human collaboration between facial expressions and gaze direction.

H6.1.2 The robot's positive facial expressions will make individuals perceive the robot as more intelligent than the neutral and negative facial expressions. A mutual gaze between the guide and the actor could elicit participants to have different impressions of the robot. There could be an interaction effect on human perception of the robot between facial expressions and gaze.

6.1.2 Study Design

To investigate our research questions and examine our hypotheses, we tasked pairs of participants with playing a collaboration game while the iCub robot observed their interaction and detailed the task they should perform after each interaction. We measured participants' completion time of the game and recorded their evaluation of the robot's intelligence during the game—participants pushed buttons on game round completion to indicate whether they thought the robot behaved intelligently or randomly. Participants were also asked to fill in the Godspeed questionnaire after the game to report on their impression of the iCub robot.

Task and Procedures

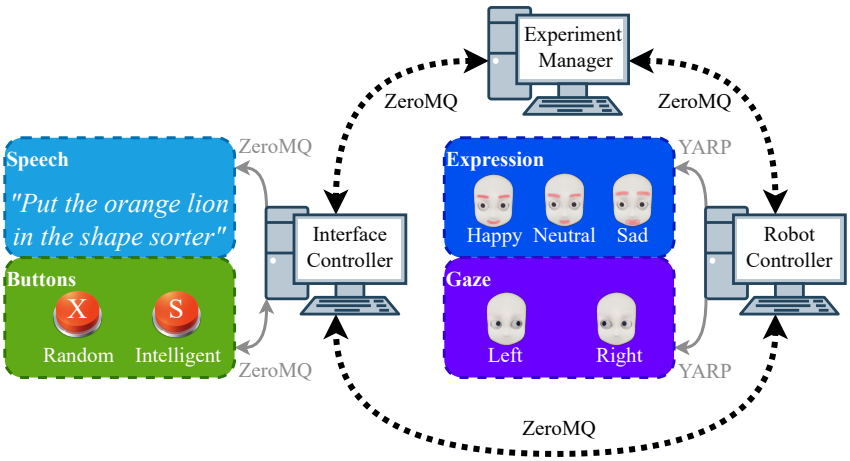


Figure 6.1: Experiment manager defines the game flow and communicates with controllers (represented by dotted arrows). Controllers connect unidirectionally to devices except for buttons since they register the user input and transmit it to the controller, requiring bidirectional communication.

In our study, we randomly matched participants in pairs. Each pair played multiple rounds of a triadic collaboration game while the iCub robot observed their interaction. Additionally, the iCub robot adopted the role of an instructor, requesting participants to place a particular object in its corresponding hole on a shape sorter. One of the two participants played the role of an actor and was capable of manipulating the objects and the shape sorter, which were obscured from their view. The other participant, having an unobstructed view of the objects and shape sorter, guided the actor in placing the selected object into its designated

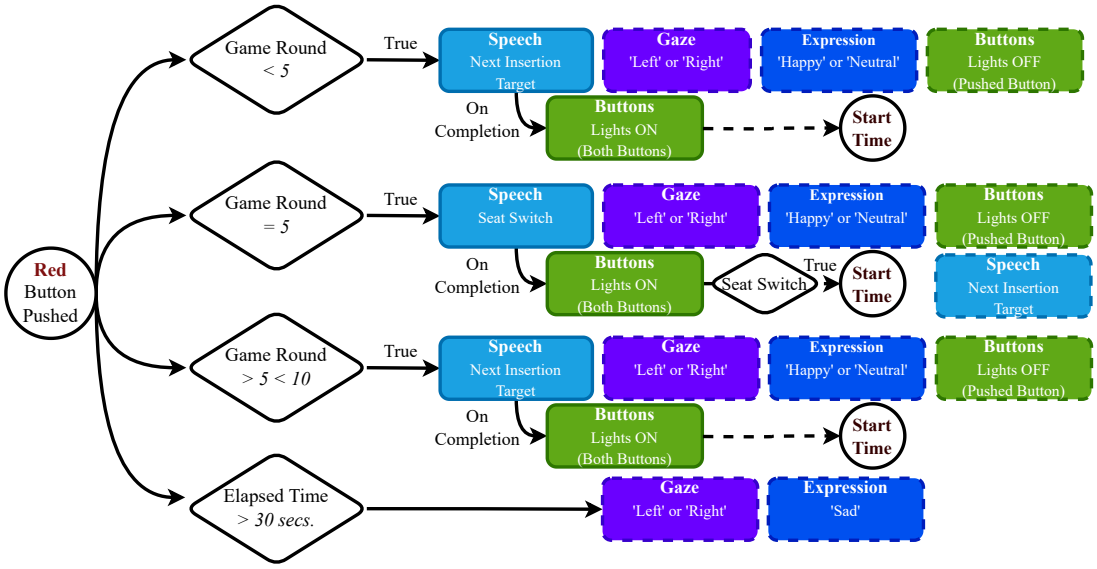


Figure 6.2: Game flow under different conditions after pressing either rating button. Speech is uttered initially on fulfilling a condition. Once the utterance is completed, follow-up actions are executed (e.g., switching on the button lights). Action blocks with dashed borders are executed in parallel after follow-up actions are completed, or previous conditions are fulfilled.

hole. Guidance was restricted to non-physical contact, conveyed mainly through verbal instructions and physical gestures.

The participants played a total of 10 rounds. Before each round started, the iCub robot displayed an initial facial expression, which could be neutral or happy, as depicted in Figure 6.1. During the game, they were tasked with the successful insertion of an object into the shape sorter. After 5 rounds of gameplay, participants changed seats, consequently switching their roles as actors and guides. During each round, the iCub robot displayed its final facial expressions. The participants were asked to guess the intention behind the robot's facial expressions. In doing so, the participants would distribute their attention between the task at hand and the iCub robot, rather than primarily focusing on completing the task quickly and disregarding the robot in the process. Each round completion time was recorded as the collaboration time for each pair. On round completion, participants were requested to rate the iCub robot as either intelligent or random according to its gaze behavior and facial cues during the round. The robot rating was performed by the participant assuming the role of a guide. Using two separate buttons to categorize the iCub robot as either intelligent or random, we acquired their responses following each round, along with their round completion time. If the participants finished one round within 30 seconds, the robot maintained its initial facial expression and gaze direction. If the participants failed to complete the task within 30 seconds, the iCub robot displayed a sad final facial expression as depicted in Figure 6.1 and shifted its gaze either toward the actor or the guide. After completing all 10 rounds, the participants filled in the Godspeed questionnaire [23] to report their impression

of the iCub robot based on its appearance and behavior. The overall flow of the game is illustrated in Figure 6.2.

Experimental Setup

Figure 6.3: The two participants adopting the roles of a guide *(left)* and an actor *(center)* while the iCub [171] robot *(right)* observes their gameplay. The actor inserts an object into the sorter following the guide's instructions.

The experimental setup shown in Figure 6.3 consisted of a round table where the iCub head was placed and at which two human participants were seated. The distance between the iCub's head and each participant was approximately 140 cm. A shape sorter with 12 holes, each with a color—6 colors in total—corresponding to an animal and a basic-shaped object, was placed on the table. The sorter and objects were occluded from the actors' view by opaque surfaces covering the sides of a plastic wireframe. The guide had a clear view of the shape sorter in order to guide the actor in inserting an object specified by the iCub robot.

After each trial, the guide rated the iCub robot's intelligence by pressing one of two labeled red buttons with lights. The 'X' labeled button indicated the iCub robot's observed behavior followed an unspecified pattern that did not correlate with the participants' actions. The 'S' labeled button signified an intelligent pattern of the iCub robot's behavior, which is associated with the participants' gameplay. The button lights signaled the ongoing running of the experiment round. Pressing either button momentarily switched it off until instructions for the next round were verbally delivered through a loudspeaker situated behind the iCub robot. Instructions were simply structured phrases to convey the target for each round, e.g., 'Put the orange lion in the shape sorter'. These instructions were uttered

using Amazon Polly speech synthesis, spoken with a child voice labeled as 'Justin' to match the iCub robot's appearance.

We define the game flow as the full experimental pipeline, beginning with the iCub robot introducing the game to participants, followed by providing instructions on which objects to insert, and eventually thanking the participants for taking part in the experiment. Further processes involved in the game flow include providing instructions on switching seats and filling in the questionnaire, keeping track of the participants' game completion time, performing gaze movements, and displaying facial expressions. The game flow involves three computers with different roles as depicted in Figure 6.1:

1. The **Experiment Manager (PC:EM)** runs the main script, which coordinates the tasks of controllers that interact with external devices and sensors. PC:EM receives feedback from the controllers and delegates actions to them, such as moving the iCub's head in either direction, changing the iCub robot's facial expression, or uttering instructions. It communicates over the ZeroMQ [113] middleware using Wrapyfi (detailed in Chapter 5), a Python wrapper with multi-middleware support for exchanging native Python objects, tensors, and arrays.

2. The **Interface Controller (PC:IC)** awaits button presses by the participants who had to rate the iCub robot's behavior as intelligent or random. It also controls the embedded button lights and sends audio signals to the speech interface via ZeroMQ. The buttons are connected to an Arduino AT-Mega 2560 microcontroller that communicates with PC:IC over USB serial. PC:IC also uses Wrapyfi for communicating over ZeroMQ.

3. The **Robot Controller (PC:RC)** sends control signals to the iCub robot to direct its gaze toward the guide or the actor based on predefined estimated positions. It also sends emotion templates to the robot through the emotion interface. Since the iCub robot runs YARP [170], we utilize both YARP and ZeroMQ on PC:RC to communicate with the iCub robot and PC:EM, respectively.

Data Analyses

Completion time (CT) and rating of the robot are measurements of participants' game performance and perception of the robot, respectively. We conducted a two-factor repeated measures ANOVA with facial expressions (neutral vs. happy) and gaze direction (actor vs. guide) on the game completion time to examine the impact of the robot's **initial** facial expressions and gaze direction on triadic collaboration. The **final** facial expressions and the gaze from the iCub robot were displayed while the participant played each round of the game.

To investigate how the robot's initial facial expression and gaze direction influence participants' perception of the robot's intelligence, a two-factor repeated measures ANOVA with facial expressions (neutral vs. happy) and gaze direction

(actor vs. guide) was performed on participants' ratings. We encoded participants' 'intelligent' rating of the robot with a value of '1', and 'random' with a value of '0'. Thus, the higher the robot rating, the more intelligent participants perceived it.

To measure the impact of the robot's final expression and gaze direction on participants' ratings of the robot, one paired t-test was performed between sad and happy expressions—neutral expressions were excluded considering their rare occurrence as final facial expressions. Another paired t-test was performed between the ratings of the actors and the guides. We did not analyze the interaction effect between the final expressions and gaze direction. This is due to participants observing more negative expressions than happy and neutral expressions since their completion time was usually longer than 30 seconds. Under the sad expression condition, gaze direction was balanced. However, under the happy and neutral expression conditions, gaze direction was not balanced, resulting in a majority of participants experiencing only a subset of the condition combinations.

Additionally, we also investigated whether there would be any differences between the first and last 5 rounds of participants' game performances and robot ratings by using paired-samples t-tests, given that participants switched roles after 5 rounds. Eventually, we analyzed the correlation between completion time, robot rating, and five sub-dimensions of the Godspeed questionnaire to study the relationship between participants' general impression of the robot and their perception of it during the game. All post hoc tests in the current study used Bonferroni correction.

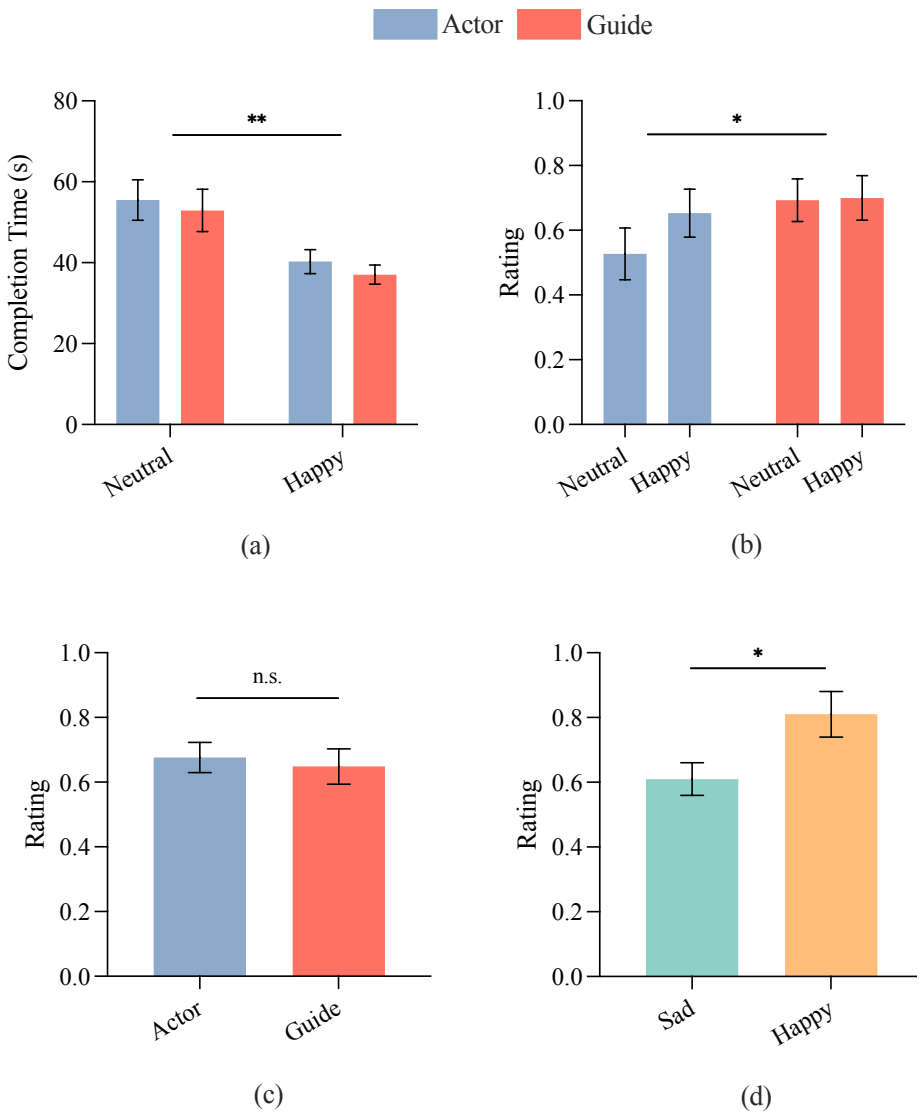
Participants

50 participants (female = 13, male = 37) took part in this experiment. Participants were between 21 and 55 years of age, with a mean age of 29.02 ± 5.60 years. All participants reported no history of neurological conditions (seizures, epilepsy, stroke, etc.) and had either normal or corrected-to-normal vision and hearing. This study was conducted following the principles expressed in the Declaration of Helsinki. Each participant signed a consent form approved by the Ethics Committee of the Department of Informatics, University of Hamburg.

6.1.3 Results

Initial Facial Expressions and Gaze on Collaboration

To evaluate the impact of the initial facial expressions and gaze direction on the collaborative game performances, a repeated measures ANOVA with a Greenhouse-Geisser correction was applied. Results displayed in Figure 6.4a showed that the main effect of facial expressions was significant. The participants' RT differs significantly between different facial expression conditions, $F\,(1, 23) = 15.73, p < .01, \eta_p^2 = .40$. Post hoc tests show that the participants finished the game significantly faster under the happy condition (mean ± SE = 38.66 ± 2.04 ms) than the neutral condition (mean ± SE = 54.22 ± 4.15 ms). However, the main effect of the initial gaze direction was not significant. There was no significant difference



CT: Round completion time; Robot Rating: participants' evaluation of the robot after each game round.

$*$ denotes $.01 < p < .05$, $**$ $.001 < p < .01$, and *n.s.* denotes no significance.

Figure 6.4: Participants' **(a)** completion time of the game and **(b)** their rating of the robot's intelligence under its different initial facial expression and gaze direction conditions. Their rating of **(c)** the robot's intelligence given its different final gaze directions, and **(d)** facial expressions.

in participants' game performances, whether or not the robot's initial gaze was toward the actor or the guide, $F\,(1, 23) = .93, p = .35, \eta_p^2 = .04$. There was no significant interaction effect between the initial facial expressions and the initial gaze direction, $F\,(1, 23) = .01, p = .94, \eta_p^2 = 0$.

Initial Facial Expressions and Gaze on Rating the Robot

To evaluate the impact of the initial facial expressions and gaze direction on rating the iCub robot, a repeated measures ANOVA with a Greenhouse-Geisser correction was applied. Results presented in Figure 6.4b showed that the main effect of facial expressions was not significant. There were no significant differences in participants' ratings of the robot between neutral (mean ± SE = .61 ± .06 ms) and happy conditions (mean ± SE = .68 ± .06 ms), $F(1, 23) = .79, p = .38, \eta_p^2 = .03$. However, the main effect of the initial gaze direction was significant, $F(1, 23) = 4.94, p < .05, \eta_p^2 = .17$. Post hoc tests show that participants perceived the iCub robot as significantly more intelligent when the robot initially looked at the guide (mean ± SE = .70 ± .05) than when looking at the actor (mean ± SE = .59 ± .06). There was no significant interaction effect between the initial facial expression and the initial gaze direction on participants' ratings of the robot, $F(1, 23) = 1.18, p = .29, \eta_p^2 = .05$.

Final Facial Expressions and Gaze on Rating the Robot

Paired-samples t-tests were conducted to study the influence of the final emotion on rating the robot. Results in Figure 6.4d showed that participants rated the iCub robot significantly more intelligent when the robot displayed happiness (mean ± SE = .81 ± .07) than sadness (mean ± SE = .61 ± .05), $t(24) = -2.46$, $p < .05$.

Paired-samples t-tests were also conducted to investigate how the final gaze direction impacted the robot's rating. No significant difference was found between the two conditions (actor: mean ± SE = .68 ± .05, guide: mean ± SE = .65 ± .05, $t(24) = .52$, $p = .61$), as shown in Figure 6.4c.

Learning Effects in the Collaborative Game

Figure 6.5 shows that there was a reduction in completion time for the first 5 rounds of the game. After switching roles, participants' completion time in the last 5 rounds also decreased, indicating that learning effects persisted throughout the game. Furthermore, we conducted a paired-samples t-test showing that participants took significantly less time completing the last 5 rounds (mean ± SE = 38.38 ± 2.12 s) than the first 5 rounds (mean ± SE = 51.62 ± 3.32 s), $t(24) = 3.94$, $p < .01$. These findings imply that repeated exposure to the collaborative game and increased familiarity with the partner's role improved performance, emphasizing the importance of experience and practice in enhancing collaborative skills. However, robot ratings did not follow a consistent trend. The paired-samples t-test performed on robot ratings indicated no significant difference between the first 5 rounds (mean ± SE = .64±.05) and the last 5 rounds (mean ± SE = .67±.06), $t(24) = -.44$, $p = .67$. These results suggest that participants' perception of the robot did not change with more practice of the game (rating counts after different rounds are shown in Figure 6.6).

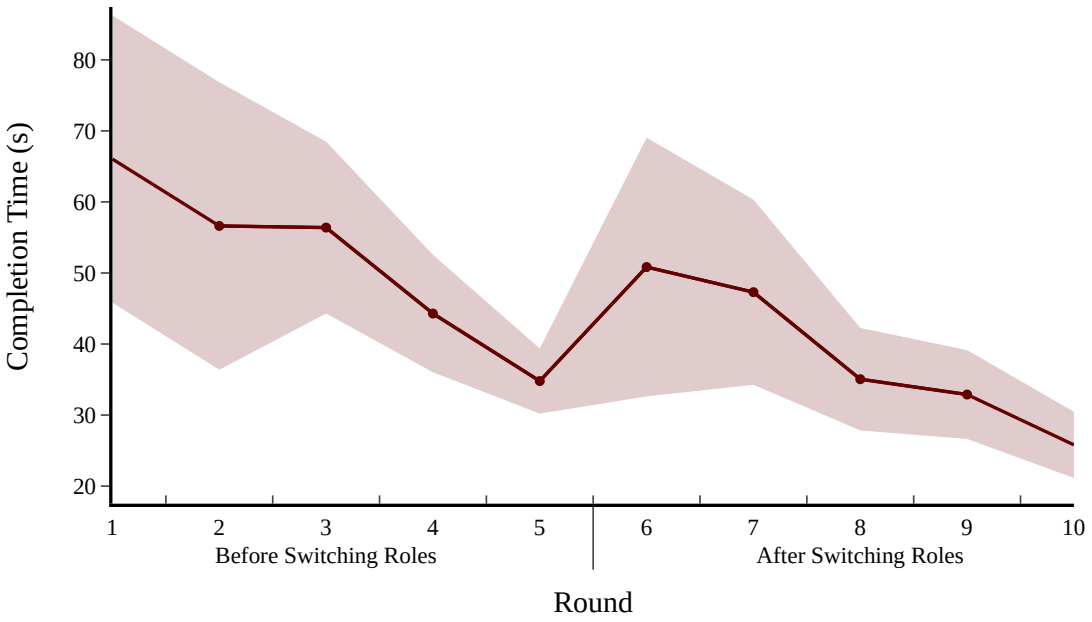


Figure 6.5: The mean completion time and the standard error per game round decline as participants gain collaboration and gameplay experience.

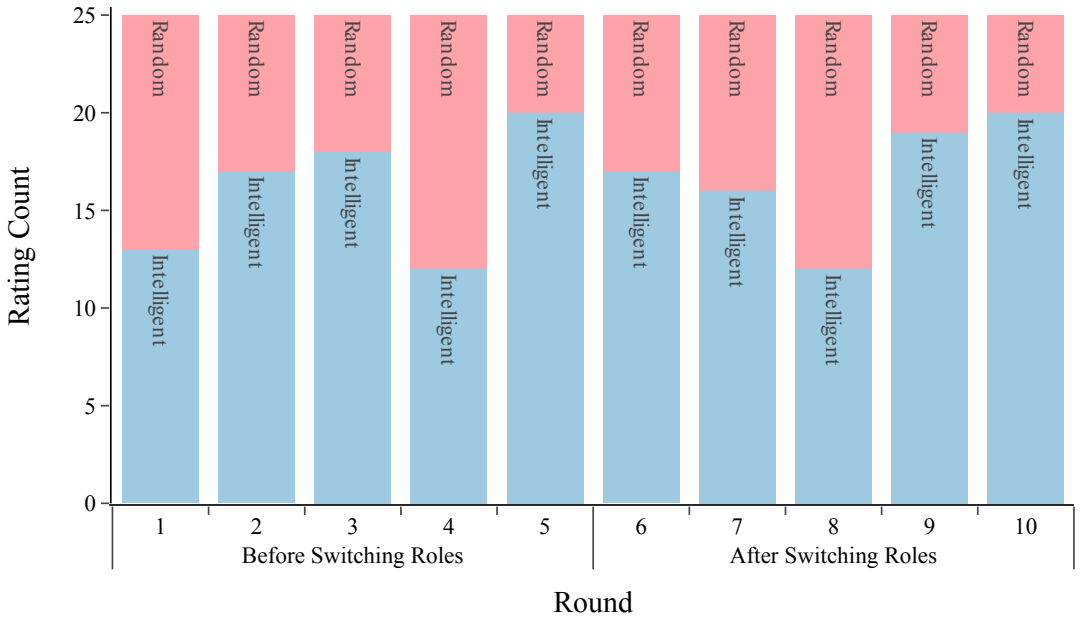


Figure 6.6: Robot rated more often as intelligent than random by the guide. Intelligence rating increases from 16 to 16.8 times after switching roles (guide|actor).

Godspeed Questionnaire

Table 6.1: Statistics and correlation matrix of ratings and Godspeed dimensions.

Dimension	Mean ± SD	Rating	AT	AN	LI	IN	SF
Rating	0.66 ± 0.29	1					
AT	2.48 ± 0.86	0.24	1				
AN	2.74 ± 0.83	0.22	0.82***	1			
LI	3.51 ± 0.86	0.27	0.61***	0.66***	1		
IN	3.10 ±0.84	0.41**	0.74***	0.68***	0.71***	1	
SF	3.66 ± 0.72	-0.08	0.36**	0.48***	0.56***	0.27	1

Rating: Participants' evaluation of the robot after each game round; AT: Anthropomorphism; AN: Animacy; LI: Likeability; IN: Intelligence; SF: Safety.
$*$ denotes $.01 < p < .05$, $**$ $.001 < p < .01$, $***$ $p < .001$, and *n.s.* denotes no significance.

Means and standard deviations for completion time, robot rating, five sub-dimensions (Anthropomorphism, Animacy, Likeability, Perceived Intelligence, Safety) of the Godspeed questionnaire, as well as the correlation coefficients between them, are displayed in Table 6.1. The rating of the robot during the game was positively correlated to Perceived Intelligence ($r = .41, p < .01$). The Completion time was not significantly correlated with any other measurements ($ps > .05$). Additionally, a weak positive correlation was measured between robot rating and Likeability ($r = .27, p = .063$). Within the sub-dimensions of the Godspeed questionnaire, only the association between Perceived Intelligence and Safety was marginally significant ($r = .27, p = .056$). Associations between other dimensions reached significance ($ps < .05$).

6.1.4 Discussion

Our study shows that a robot displaying a positive (happy) facial expression on initiating interaction improves collaboration between humans: participants complete the task within a shorter period of time—less than 30 seconds—when the iCub robot appears happy. We hypothesize that emotional contagion plays a role in altering the participants' emotions. The iCub robot's expression of happiness reflects positively on the participants' mood, resulting in them being more productive and collaborative. This hypothesis is supported by studies examining the relationship between emotional states and productivity [194], indicating that happy individuals tend to have better performance.

Participants completing the task within 30 seconds also rated the iCub robot as more intelligent, even though the robot followed the same strategy in every interaction. One influencing factor could be that the iCub robot displays a negative (sad) facial expression when the participants take longer than 30 seconds to complete the task. A robot that displays a happy facial expression may be perceived as more friendly, trustworthy, and competent than a robot that displays a negative

emotion [46]. However, we were unable to examine the effect of deferred facial expressions—shown after 30 seconds—due to the limited sample size. Examining whether a display of negative emotions has an effect on the robot's intelligence rating would only be possible if we were to vary the facial expressions when participants took longer than 30 seconds to complete a round. Given the infrequent occurrence of the event, the two conditions would not result in sufficient samples for a statically sound comparison.

On establishing mutual gaze with the guide, the iCub robot is regarded as more intelligent than when looking at the actor. Given that the guide rates the robot, we compared their ratings under the condition of mutual gaze—the iCub robot looking at the guide—and looking elsewhere. Our results aligned with prior findings, showing that mutual gaze caused participants to perceive the robot as more engaged, humanlike, and attentive, eliciting them to attribute higher intelligence to it [142, 25].

Several limitations in the current study could be addressed in future research. First, the gaze directions and the final facial expressions should be balanced to study the interaction effect on rating the robot. Second, more measurements could be conducted on participants' personality traits and their trust in the robot. This could lead to a more in-depth understanding of the current results. Finally, involving more emotions and increasing human-human interaction rounds could yield more concrete findings. Addressing these limitations would lead to an even more comprehensive understanding of the impact of non-verbal social cues on triadic collaboration.

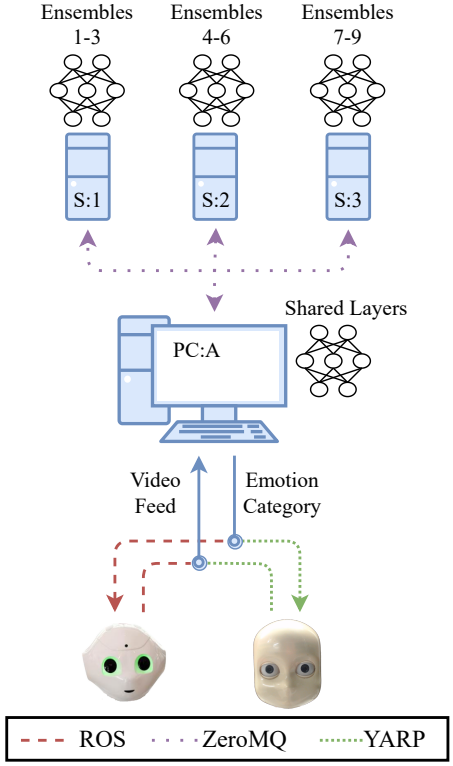
6.2 Social Cue Mirroring

6.2.1 Introduction

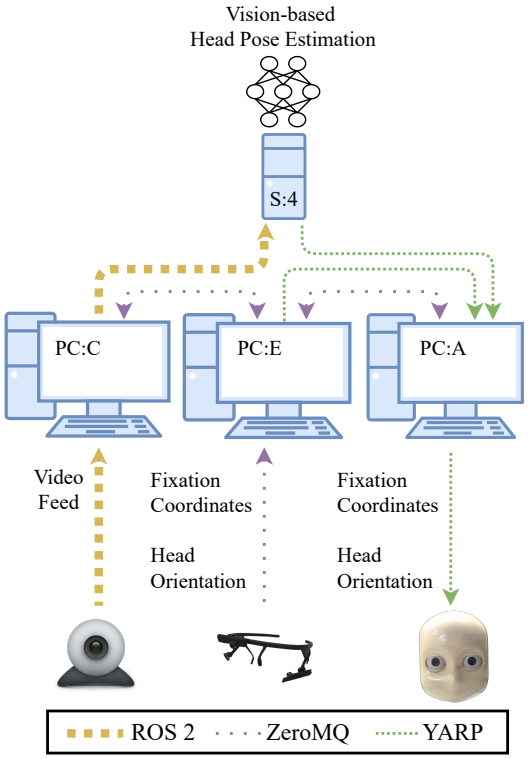
The mirror neuron system (MNS) has inspired many computational methods for intelligent robotics [265]. The MNS in humans facilitates the understanding of others by simulating their behaviors via sensorimotor processes [49]. Mirroring, a fundamental element of social interaction, involves subconsciously imitating another individual's nonverbal cues, such as gestures, expressions, and postures [96]. It can reflect an adaptive integration and utilization of social cues within the social context [258]. This mechanism often leads individuals to collaborate with those who exhibit similar and familiar behaviors [77]. Moreover, mirroring plays a significant role in human-robot social interaction. By mimicking non-verbal social cues, humans feel socially closer to the robot and perceive the robot as more aware of the intentions behind their social behaviors [151].

Humans mirror facial expressions as a means of social bonding and communication. Mirroring assists in expressing emotions, empathy, and intentions, consequently enhancing understanding among individuals [87]. Similarly, for robots to be part of social environments, they must recognize and replicate natural affective signals. Affective mirroring leads to the perception that the robot is capable of

conveying internal states, displaying social intelligence, and expressing humanlike characteristics [70, 38]. Current research in this area focuses on using deep-learning approaches to enable robots to recognize and mimic emotions [58]. However, a significant challenge is the robots' limited ability to mirror human affect in real-time accurately. Most robots can replicate basic emotions but the means of those expressions could lead to a lowering in perception of intuitiveness and relatability. To examine whether the means by which robots express affect influence the outcomes of a study, we conduct an experiment with two different robots. The iCub [171] robot is capable of expressing affect through light patterns assimilating human facial expressions, whereas the Pepper [235] robot can convey facial expressions through color changes. We illustrate the technical overview in Figure 6.7a.



(a) Facial expression recognition for updating the affective cues on the Pepper and iCub robots demonstrating the forwarding and mirroring schemes.



(b) Head orientation and gaze estimation for robotic control demonstrating the channeling and mirroring schemes.

Figure 6.7: Technical overview of the two tasks demonstrating the **(a)** forwarding and mirroring schemes, and **(b)** the channeling and mirroring schemes.

Studies also indicate that gesture imitation during human conversations leads to decreased social distance and increased prosocial behavior [87]. In human-robot social interaction, movement mirroring enhances robots' sociability during human-robot interactions, making them more humanlike, empathetic, and socially intelligent [40]. Two primary methods of enabling robots to mirror human move-

ments include Inertial Measurement Unit (IMU)-based and vision-based control. IMU-based control refers to teleoperated robots that are controlled by readings arriving from IMU sensors attached to the bodies of human operators, offering precision and adaptability [275]. They also allow for real-time transmission of orientation readings, facilitating a more rapid response. We define vision-based control mirroring as a computer vision-driven approach, relying on external sensors to capture and estimate the movements of a human operator. To assess the preferred means of movement capture and how it influences the perception of the robot, we conduct an experiment on the iCub robot, which is capable of moving both its eyes and head. The iCub robot mirrors a participant's head movements either through IMU-based readings or vision-based estimates. We illustrate the technical overview in Figure 6.7b.

The performance of robots' mirroring behaviors is measured from multiple dimensions during human-robot interaction. For instance, some studies reported that affective mirroring makes the robot more socially intelligent, humanlike, and less mechanical [41]. This observation stems from the robots' capabilities to adapt by understanding and responding to human emotions and social signals [131]. Metrics such as humanlikeness and responsiveness are used to assess how well robots' emotional responses align with human anticipations. In the context of movement mirroring, the mechanical and responsiveness criteria rate the precision, fluidity, and adaptability of robots as perceived by humans [93]. In this study, we evaluated robot performance based on four impression dimensions: social intelligence, mechanical attributes, responsiveness, and humanlikeness, as described by Seifert *et al.* [228].

Social robots are designed with the goal of assisting and adapting to humans. However, very often the opposite occurs, where individuals find themselves adapting to the robots instead. This issue arises since the robot or agent is not always built with human preferences and interactive needs in mind [155, 222]. Therefore, we also focus on improving real-time human-robot communication by reducing latency to make interactions more fluent and natural. We created a mirroring framework for easily interchanging robots and sensors, and selected specific neural models and communication methods based on their potential to address these issues. We conduct two experiments to compare the performance of different robot platforms on mirroring tasks and assessed the impact of using different control methods on the same robot platform. Our goal is to better understand these variations and improve how robotic design aligns with human expectations. We aim to address these problems by investigating the following research questions (RQ):

RQ6.2.1 *How do different robotics platforms, specifically the iCub and Pepper robots, compare in during affective mirroring?*

RQ6.2.2 *How do various robotic control methods, especially vision-based control and IMU-based control methods, impact the iCub robot's performance in movement mirroring tasks?*

6.2.2 Study Design

Affective Mirroring Task

In this experiment, participants were asked to make eight facial expressions—*Anger*, *Fear*, *Happiness*, *Disgust*, *Sadness*, *Neutral*, *Surprise*, and *Contempt*—in front of the Pepper or iCub robots. The expressions were to be performed within one minute in any order. The robot mirrored participants' expressions either through *affective signaling*—by changing the Pepper robot's eye and shoulder LED colors [157, 125]—or *robotic facial expressions*—by changing the iCub robot's eyebrow and mouth LED patterns [16]. Next, participants were asked to match the colors displayed on the Pepper robot and the facial expressions on the iCub robot to emotion categories. The experiment conducted on the physical iCub and Pepper robots is depicted in Figure 6.8.

Upon task completion, participants were asked to scan a QR code appearing on the Pepper robot's tablet using their cell phones to complete a three-item questionnaire, evaluating their experiences with either robot. In both questionnaires, participants were requested to rate their interaction with the robots on a 5-point Likert scale (Q1-1 and Q1-2):

1. How precise was the robot in mirroring your facial expressions? (1 = very imprecise, 5 = very precise)

2. Did the robot mirror your expressions with major delay? (1 = no significant delay, 5 = significant delay)

Participants rated their impression of the robots on four dimensions —*Socially Intelligent*, *Mechanical*, *Responsive*, and *Humanlike*—on a 5-point Likert scale (1 = not at all, 5 = yes, a lot).

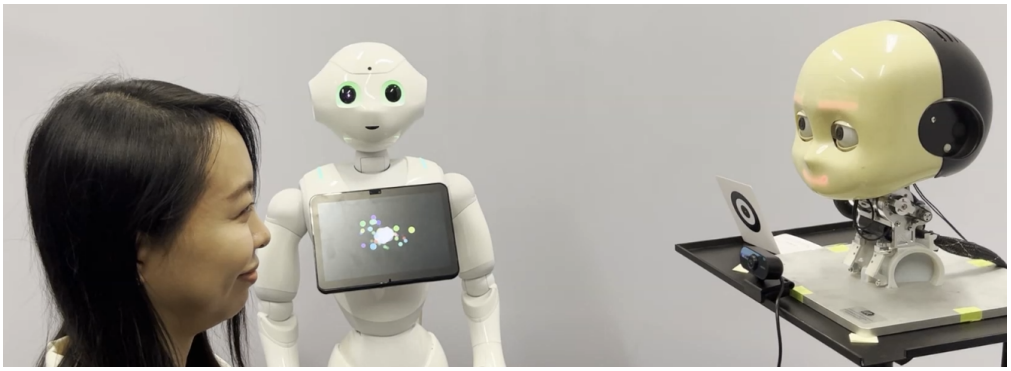


Figure 6.8: Facial expression imitation on the Pepper and iCub robots.

For recognizing facial expressions, we relied on the emotion categories inferred by the ESR9 [234] model. Siqueira *et al.* [234] present a neural model called ESR9 for facial expression recognition, composed of an ensemble of convolutional branches with shared parameters. The model provides inference in real-time settings, owing to its relatively small number of parameters across the ensemble branches, unimodal

visual input, and non-sequential structure. For the last timestep n, a majority vote is cast on the output categories resulting from each ensemble branch e_i:

$$\begin{aligned} \mathbf{c}(f)_n &= \sum_{i=1}^{E} [\mathrm{e}_i = f] \\ \mathrm{c}_n &= \arg\max_f \mathbf{c}(f)_n \end{aligned} \tag{6.1}$$

where $E = 9$ is signifying the number of ensemble branches. The emotion category index is denoted by $f \in [1, 8]$. The resulting $\mathbf{c}(f)_n$ holds counts of the ensemble votes for each emotion category f at n.

Given the model's sole reliance on static visual input, falsely recognized facial expressions lead to abrupt changes in the inferences. To mitigate sudden changes in facial expressions, we apply a mode smoothing filter to the last N discrete predictions—eight emotion categories—where $N = 6$ corresponding to the number of visual frames acquired by the model per second:

$$\begin{aligned} \mathbf{k}(f)_n &= \sum_{i=n-N+1}^{n} [\mathrm{c}_i = f] \\ \mathrm{k}_n &= \arg\max_f \mathbf{k}(f)_n \end{aligned} \tag{6.2}$$

resulting in the emotion category k_t being transmitted from the inference script running the facial expression recognition model to the managing script executed on **PC:A** as shown in Figure 6.7a. The managing script forwards data to and from the model and robot interfaces. We executed the inference script on four machines. The shared layer weights were loaded on an NVIDIA GeForce GTX 970 (denoted by PC:A in Figure 6.7a) with 4 GB VRAM. Machines **S:1**, **S:2**, and **S:3** shown in Figure 6.7a, shared similar specifications, each with an NVIDIA GeForce GTX 1050 Ti having 4 GB VRAM. We distributed nine ensemble branches among the three machines in equal proportions and broadcasted their latent representation tensors using ZeroMQ. The PyTorch-based inference script was executed on PC:A, S:1, S:2, and S:3, all having their tensors mapped to a GPU.

Depending on the experimental condition, images arrived directly from each robot's camera:

1. The iCub robot image acquired from the left eye camera having a size of 320×240 px and transmitted over YARP at 30 FPS.

2. The Pepper robot image acquired from the top camera having a size of 640×480 px and transmitted over ROS at 24 FPS.

The image was directly forwarded—the *forwarding scheme* is detailed in Section 5.5.2—to the facial expression model, resulting in a predicted emotion returned to the corresponding robot's LED interface.

Gaze and Head Movement Mirroring Task

In this experiment, participants interacted with the iCub robot given two conditions. Under the vision-based control condition, the iCub robot's movements were actuated by a vision-based head pose estimation model. Under the Inertial Measurement Unit (IMU)-based control condition, the orientation readings arrived instead from an IMU attached to a wearable eye tracker. Participants wore the eye tracker and were asked to look at the iCub robot, freely moving their eyes and head. Participants observed the movements of the iCub robot to evaluate the interaction. The experiment demonstrated on a simulated iCub robot is depicted in Figure 6.9.

Participants were requested to rate their interaction with the iCub robot on a 5-point Likert scale (Q2-1 – Q2-5):

1. How precise was the robot in mirroring your head movements? (1 = very imprecise, 5 = very precise)
2. Did the robot mirror your head movements with major delay? (1 = no significant delay, 5 = significant delay)
3. Did the robot move its eyes? (yes/no)
4. How precise was the robot in mirroring your eye movements? (1 = very imprecise, 5 = very precise)
5. Did the robot mirror your eye movements with major delay? (1 = no significant delay, 5 = significant delay)

Participants rated their impression of the iCub robot on four dimensions—*Socially Intelligent*, *Mechanical*, *Responsive*, and *Humanlike*—on a 5-point Likert scale (1 = not at all, 5 = yes, a lot).

For vision-based control, we relied on the orientation coordinates inferred by the 6DRepNet [110] model. Hempel *et al.* [110] present 6DRepNet, a novel end-to-end neural network model for head pose estimation. The authors proposed a unique solution that leverages a 6D rotation matrix representation and a geodesic distance-based loss function. The 6D rotation matrix utilized in their approach is highly efficient for representing the orientation of objects in three-dimensional space by encoding six parameters $\mathrm{p}_{[1,6]}$ instead of the typical nine:

$$\mathbf{p}_x = [\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3] \qquad \mathbf{p}_y = [\mathrm{p}_4, \mathrm{p}_5, \mathrm{p}_6] \tag{6.3}$$

resulting in a rotation matrix $\mathbf{R}$:

$$\begin{aligned} \mathbf{r}_x &= \frac{\mathbf{p}_x}{\sqrt{\sum_{n=1}^{3} \mathrm{p}_{x,n}^2}} \\ \mathbf{r}_z &= \frac{\mathbf{r}_x \times \mathbf{p}_y}{\sqrt{\sum_{n=1}^{3} \left(\mathrm{r}_{x,n} \times \mathrm{p}_{y,n}\right)^2}} \\ \mathbf{r}_y &= \frac{\mathbf{r}_z \times \mathbf{r}_x}{\sqrt{\sum_{n=1}^{3} \left(\mathrm{r}_{z,n} \times \mathrm{r}_{x,n}\right)^2}} \end{aligned} \tag{6.4}$$

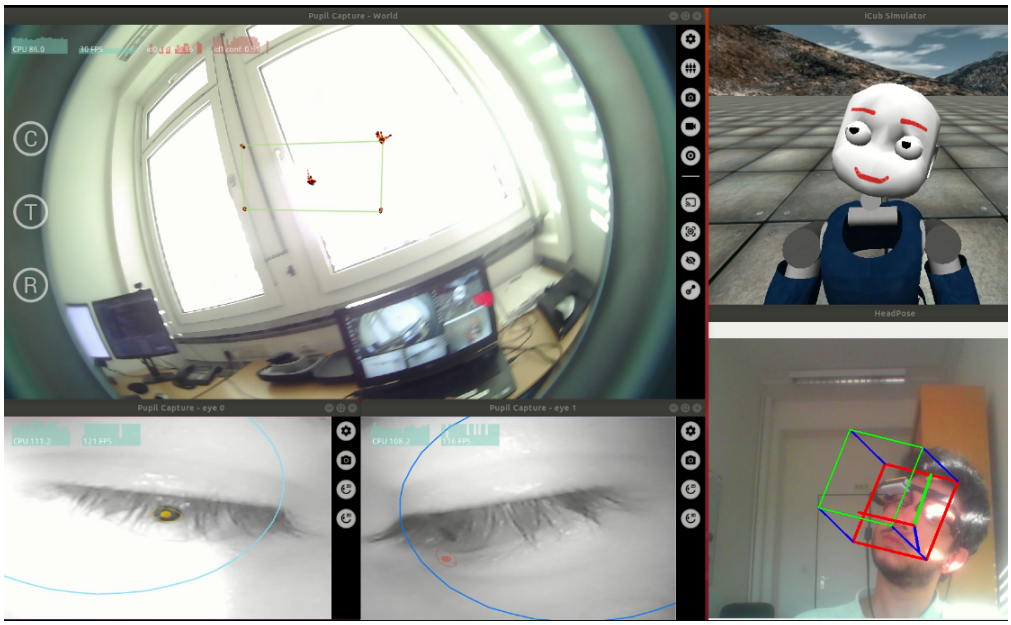


Figure 6.9: Head and eye movement imitation using either an IMU-fitted eye tracker for IMU-based readings or vision-based head pose estimation model coordinates to control a robot. The method is demonstrated on the simulated iCub robot using *iCubSim* [250] *(top right)*. A fixed camera facing the participant transmits the image *(bottom right)* to 6DRepNet [110], which in turn estimates the head orientation, displayed as a wireframe cube (*bottom right*). Simultaneously, the IMU readings are received and displayed as coordinate axes (*bottom right*). The eye tracker's eye camera views *(bottom left)* and world view *(top left)* are used to estimate the participant's fixation point, which controls the robot's eye movements.

$$\mathbf{R} = \begin{bmatrix} | & | & | \\ \mathbf{r}_x^\intercal & \mathbf{r}_y^\intercal & \mathbf{r}_z^\intercal \\ | & | & | \end{bmatrix} \equiv \begin{bmatrix} \mathrm{R}_{11} & \mathrm{R}_{12} & \mathrm{R}_{13} \\ \mathrm{R}_{21} & \mathrm{R}_{22} & \mathrm{R}_{23} \\ \mathrm{R}_{31} & \mathrm{R}_{32} & \mathrm{R}_{33} \end{bmatrix} \tag{6.5}$$

which is utilized to acquire the Euler angles following the standard order (roll ϕ, pitch θ, yaw ψ):

$$\begin{aligned} \alpha &= \sqrt{\mathrm{R}_{11}^2 + \mathrm{R}_{12}^2} \quad \beta = \begin{cases} 1, & \text{if } \alpha \geq 10^{-6} \\ 0, & \text{otherwise} \end{cases} \\ \phi_M &= (1-\beta) \cdot \operatorname{atan2}(\mathrm{R}_{12}, \mathrm{R}_{11}) \\ \theta_M &= (1-\beta) \cdot \operatorname{atan2}(\mathrm{R}_{23}, \mathrm{R}_{33}) + \beta \cdot \operatorname{atan2}(-\mathrm{R}_{32}, \mathrm{R}_{22}) \\ \psi_M &= \operatorname{atan2}(-\mathrm{R}_{13}, \alpha) \end{aligned} \tag{6.6}$$

where ϕ_M, θ_M, ψ_M define head orientation when the 6DRepNet model is used as the source for controlling the iCub robot.

The gaze coordinates were inferred from the Pupil Core [129] eye tracking glasses, worn by the participant. We attached a Waveshare 9-DOF ICM-20948

IMU to a Raspberry Pi Pico RP2040 microcontroller mounted on the upper-left rim of the Pupil Core. The eye tracker readings were inverted on the y and z axes to mirror the eye movements of the participants. We performed a single-marker eye tracker calibration prior to conducting the experiment with each participant. The two experimental conditions involved eye movements, but the orientation estimation source varied between vision-based and IMU-based control:

1. The 6DRepNet [110] model for vision-based head orientation estimation. The model is implemented in PyTorch and runs with GPU support. We executed the inference script on an NVIDIA GeForce GTX 1050 Ti (denoted by **S:4** in Figure 6.7b) with 4 GB VRAM, receiving 320×240 px images over ROS 2 captured using a Logitech C920 webcam with 30 FPS. The head orientation coordinates were inferred at a rate of 20 Hz.

2. The ICM-20948 attached to the Pupil Core eye tracker. Readings from the IMU were filtered using the Mahony algorithm [164] running on the RP2040 with a sampling frequency of 50 Hz. Both the IMU and the Pupil Core were connected to PC:E running the IMU interface, and the Pupil Capture software interfacing directly with the eye tracker. Since the Pupil interface communicates directly over ZeroMQ, we chose ZeroMQ for transmitting the IMU readings as well. In order to mirror the participant's head movement, we inverted the values of the roll ϕ and yaw ψ. However, given the IMU returns the yaw angle relative to the true north, this leads to an offset between orientation as measured by the IMU and the orientation of the participant relative to the robot. To account for this offset, we asked the participant to look straight at the robot before initiating the experiment. We then used the initial readings from 6DRepNet to shift the readings from the IMU.

The managing script running on **PC:A** as shown in Figure 6.7b, initialized the experiment by transmitting a trigger over ZeroMQ. A direct connection between machines **PC:C** and S:4 was established, as shown in Figure 6.7b, where PC:C received a trigger, starting the video feed which was directly transmitted to S:4 over ROS 2. To select the most suited middleware for this task, we evaluated the transmission latency of the 6DRepNet and IMU orientation coordinates with all four middleware. Two participants conducted five trials each, performing cyclic head rotations on θ, ψ, and ϕ—corresponding to the x,y, and z axes—independently.

The orientations inferred from the 6DRepNet and IMU were recorded for six seconds and channeled concurrently in real-time. Figure 6.10 shows the best-of-five attempts with the Euclidean distance being used as a measure of alignment when performing dynamic time warping between the two orientation estimation sources. YARP presents the lowest latency since it was configured to acquire the last message. Due to the differing sampling rates between the 6DRepNet and IMU, the accumulation of messages in the ZeroMQ subscriber resulted in a bottleneck leading to increasing latency between transmission and acquisition. With ROS and ROS 2, we set the subscriber queue size to 600 messages, allowing the subscribers to maintain all transmitted orientation coordinates without discardment. Setting

their queue sizes to 1 led to behavior matching that of YARP. However, the rate of dropped messages superseded YARP significantly, with YARP dropping approximately 2% of the messages, whereas ROS and ROS 2 exceeded 11% and 9%, respectively. The lowest distance offset between the two orientation estimation sources was achieved using ROS, however, we set YARP as the middleware of choice due to its consistent latency and relatively synchronized transmission of coordinates compared to other middleware in the *channeling scheme* mode—the channeling scheme is described in Section 5.5.3.

Next, **PC:E** as denoted in Figure 6.7b forwarded the head and fixation coordinates over YARP to the mirrored script in PC:A. Depending on the task at hand—head orientation coordinates arriving from the vision-based 6DRepNet model or IMU-based sensor—the channeling scheme method set either of the orientation estimation sources to `'None'` and transmitted the other along with the fixation

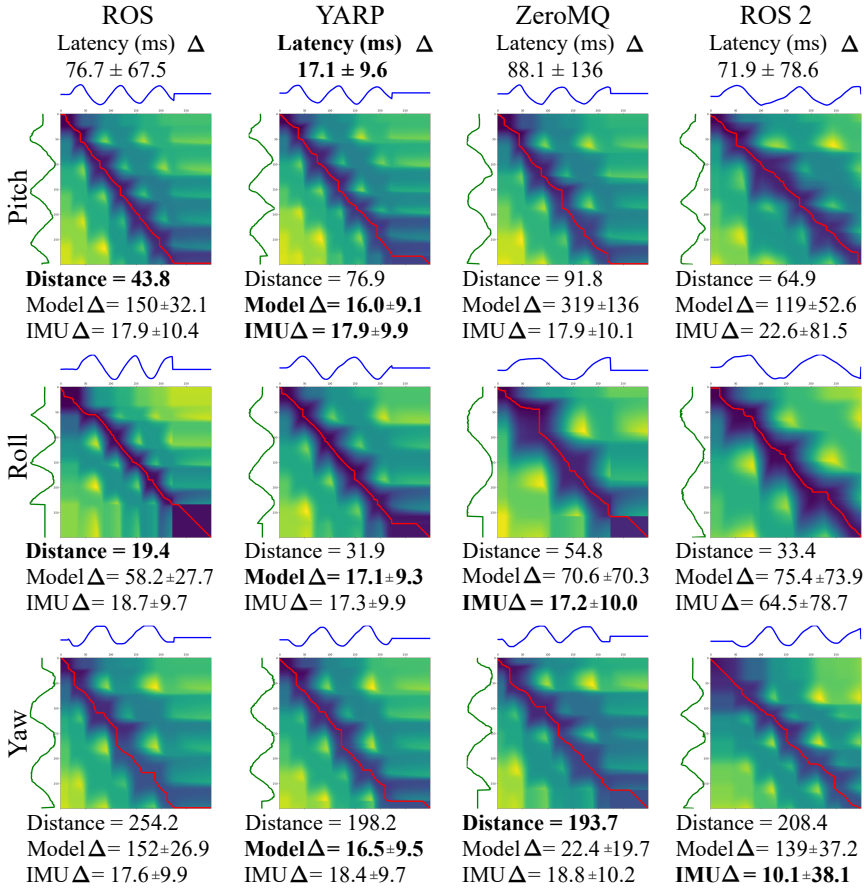
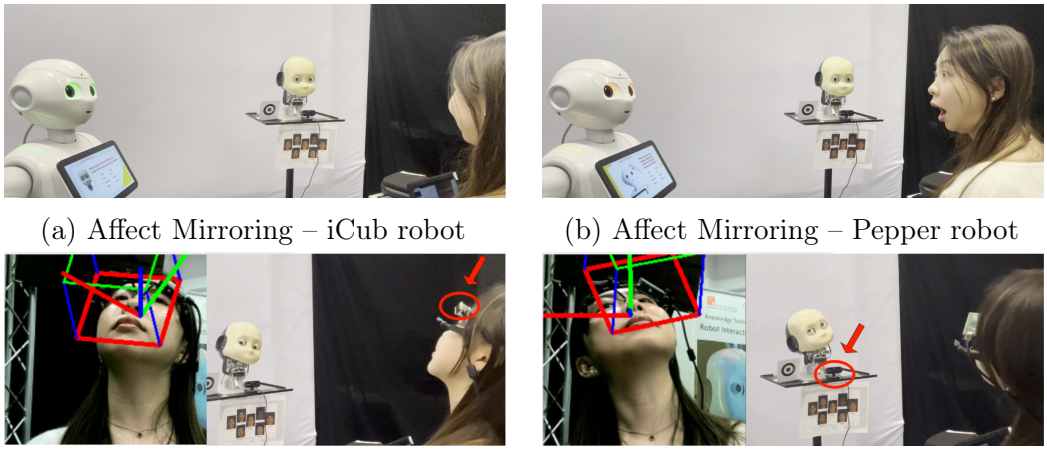


Figure 6.10: Channeling orientation coordinates received from the IMU *(horizontal)* and 6DRepNet model *(vertical)* to a non-blocking subscriber. Latency between de/serialization of IMU-based and vision-based model coordinates is measured for the best-of-five attempts using each middleware. The diagonal lines display the dynamic time-warping distances between the orientation estimation sources. **Bold** denotes the best (minimum) latency across middleware.

coordinates to the iCub robot over YARP.

Experimental Setup



(a) Affect Mirroring – iCub robot (b) Affect Mirroring – Pepper robot

(c) Movement Mirroring – IMU-based (d) Movement Mirroring – Model-based

Figure 6.11: The experimental setup showing a participant performing the four mirroring tasks in random order: **(a)** The iCub [171] robot mirroring facial expressions, **(b)** the Pepper [235] robot affectively signaling through LED color changes, **(c)** the iCub robot mirroring head movement based on an inertial measurement unit (IMU) readings with a red circle surrounding the IMU, and **(d)** the iCub robot mirroring head movement according to a vision-based model with a red circle surrounding the camera.

The participants were seated ∼80 cm away from the iCub robot's head, adjusting its height to match their eye level. A circular marker was placed beside the iCub robot to calibrate the Pupil Core eye tracker. Situated in front of the iCub robot was a Logitech C920 webcam facing the participants to perform tasks requiring a fixed view of their faces while the iCub robot moved its head and eyes. The Pepper robot stood facing the participants at an angle of 45° with a distance of 1.2 m. The Pepper robot displayed an illustration of the ongoing task on its tablet and communicated the instructions verbally. The interaction was one minute long per task condition and the condition order was randomized. The task conditions are shown in Figure 6.11. We used the Wrapyfi (detailed in Chapter 5) framework for managing the task order, transmitting data between models and robots using multiple middleware, and orchestrating the experimental pipeline.

Participants

30 participants (female = 7, male = 22, preferred not to say = 1) took part in both studies. Participants were between 24 and 41 years of age, with a mean age of 28.7. All participants reported no history of neurological conditions—seizures, epilepsy, stroke, etc.—and had normal or corrected-to-normal vision and hearing.

One participant's data was excluded from the Pepper robot's affective mirroring experiment because of self-reported color blindness. Another participant's data was excluded from the iCub robot's movement mirroring experiment due to technical issues. This study adhered to the principles expressed in the Declaration of Helsinki. Participants signed consent forms approved by the Ethics Committee at the Department of Informatics, University of Hamburg.

6.2.3 Results

We evaluated the results of both mirroring tasks, studying the perceived impression of the robot in each separate condition, as well as comparing the paired conditions within each respective task. Normality tests were conducted on the participants' answers to each dimension of the questionnaires. Results showed that their responses were normally distributed. In addition, all Post hoc tests in this study used Bonferroni correction.

Affective Mirroring

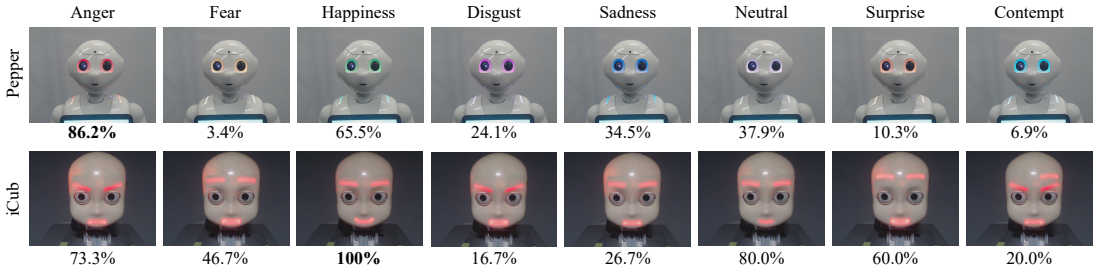


Figure 6.12: Eight emotion categories mimicked on the Pepper *(top)* and iCub *(bottom)* robots in the form of affective signaling and robotic facial expressions, respectively. Results of the human study are reported below each image in terms of the average accuracy in matching each affective signal or facial expression to an emotion category. **Bold** denotes the highest accuracy across emotion categories.

For the affective mirroring task on either robot, the recognition accuracy is listed in Figure 6.12. For the Pepper robot, participants were most accurate in recognizing anger (86.2%) and least accurate in recognizing fear (3.4%). Overall, the Pepper robot's affective signals were correctly matched to the corresponding emotion categories with a 33.6% accuracy. For the iCub robot, participants were most accurate in recognizing happiness (100%) and least accurate in recognizing disgust (16.7%). Overall, the iCub robot's facial expressions were correctly recognized with a 52.9% accuracy.

For participants' rating of interaction with the robots, results of paired-samples t-tests displayed no significant difference in precision (Q1-1) between the Pepper (mean $\pm$ SE $= 2.79 \pm .18$) and iCub (mean $\pm$ SE $= 2.90 \pm .15$) robots, ($t\,(28) = .46$, $p = .65$). No significant difference in delay (Q1-2) was found between the Pepper (mean $\pm$ SE $= 2.38 \pm .18$) and iCub (mean $\pm$ SE $= 2.48 \pm .20$) robots, ($t\,(28) = .52$,

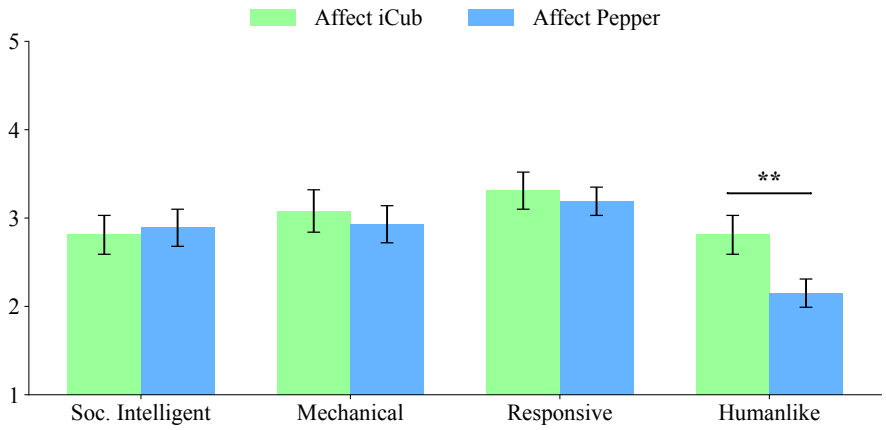
Table 6.2: Impressions of the robots on a 5-point Likert scale (1 = not at all, 5 = yes, a lot) under different task conditions (Mean ± SE).

Dimension	**Affect** *iCub* [171]	**Affect** *Pepper* [235]	**Mov. (Model)** *iCub* [171]	**Mov. (IMU)** *iCub* [171]
Soc. Intelligent	2.81 ± 0.22	2.89 ± 0.21	2.65 ± 0.20	2.46 ± 0.22
Mechanical	3.08 ± 0.24	2.93 ± 0.21	3.65 ± 0.24	3.85 ± 0.18
Responsive	3.31 ± 0.21	3.19 ± 0.16	3.65 ± 0.15	3.23 ± 0.22
Humanlike	2.81 ± 0.22	2.15 ± 0.16	2.46 ± 0.19	2.39 ± 0.21

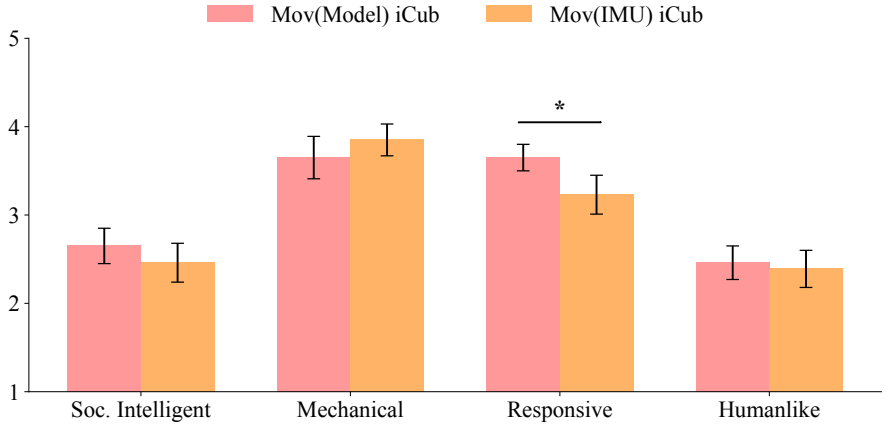
$p = .61$). For participants' rating of the impression of the robots, results of paired-samples t-tests displayed that the iCub (mean ± SE = 2.86 ± .20) robot was rated significantly more humanlike than the Pepper (mean ± SE = 2.10 ± .16) robot, ($t\,(28) = 3.45$, $p < .01$). No significant differences were found for the other three dimensions—*Socially Intelligent*, *Mechanical*, and *Responsive*—between the two robots ($ps > .05$) as shown in Table 6.2.

Movement Mirroring

A paired-samples t-tests showed that participants rated the vision-based controlled robot (mean ± SE = 3.55 ± .24) significantly more precise (Q2-1) than the IMU-based controlled robot (mean ± SE = 2.90 ± .19), ($t\,(26) = 2.19$, $p < .05$). The vision-based controlled robot (mean ± SE = 2.00 ± .17) was rated significantly less delayed (Q2-2) than the IMU-based controlled robot (mean ± SE = 2.66 ± .21), ($t\,(26) = -3.09$, $p < .01$). Under the vision-based controlled condition, all participants observed that the robot mirrored their eye movements, whereas two did not under the IMU-based condition (Q2-3). Therefore, we only analyzed data from 27 participants who reported observing eye movement under both conditions. The paired-samples t-test showed no significant difference in the precision rating of the eye movement between the vision-based controlled robot (mean ± SE = 2.48 ± .19) and the IMU-based controlled robot (mean ± SE = 2.37 ± .19) ($p > .05$) (Q2-4). Also, no significant difference was found in the delay rating of the eye movement between the vision-based controlled robot (mean ± SE = 3.07 ± .23) and the IMU-based controlled robot (mean ± SE = 3.48 ± .24) ($p > .05$) (Q2-5). For the impression of the robot, participants reported that the vision-based controlled robot (mean ± SE = 3.66 ± .22) robot was significantly more responsive than the IMU-controlled robot (mean ± SE = 3.17 ± .21), ($t\,(26) = 2.39$, $p < .05$). However, no significant differences were found in the remaining dimensions —*Socially Intelligent*, *Mechanical*, and *Humanlike*—between the two conditions ($ps > .05$) as shown in Table 6.2.



(a) Affective Mirroring



(b) Movement Mirroring

$*$ denotes $.01 < p < .05$, and $**$ $.001 < p < .01$.

Figure 6.13: Participants' impressions (5-point Likert scale) of robots under different affective and movement mirroring conditions.

6.2.4 Discussion

Participants associated the iCub robot's facial expressions with emotions more than the Pepper robot's affective signaling and found the iCub robot to be more humanlike. Another observation relates to the accuracy of recognizing different affective signals conveyed by either robot. Participants could accurately associate *Anger* with the color red and *Happiness* with green on the Pepper robot. This is complemented by findings associating exposure to different colors with physiological and psychological responses [267, 236]. Participants more accurately identified expressions of *Happiness*, *Neutral*, and *Surprise* on the iCub robot compared to the Pepper robot. This can be attributed to humans primarily relying on observing the mouth and eyebrows to recognize these facial expressions [103], features that the Pepper robot lacks.

On comparing the movement mirroring methods, we found that the vision-based control method resulted in smoother, more precise, and more responsive movements

than the IMU-based control method, as shown in Figure 6.13. The IMU-based control method transferred the IMU readings at a faster rate, however, this caused jittery movements due to hardware limitations. In our study, both methods were perceived as equally humanlike, suggesting that lower responsiveness doesn't negate humanlikeness.

Several limitations could be addressed and investigated in future research. We were unable to compare movement mirroring on the two humanoid robots. This is due to the Pepper robot's inability to roll its head or move its eyes, unlike the iCub robot. Our iCub robot does not have a full body, hence, we cannot compare the limb movement mirroring between the iCub and Pepper robots. Future studies could address the interaction effect between affective and movement mirroring. Moreover, researchers could investigate how different humanoid robots and control methods are received by children with autism spectrum disorders, and whether it affects their social functions [289].

Chapter 7

Cognitive Robotic Simulation

This chapter covers cognitive robotic simulation studies. The first study addresses multimodal audiovisual social cueing, where the social and auditory cues are congruent (matching direction) or incongruent (opposing direction). This study relies on the GASP model (DAM + LARGMU) developed in Chapter 3 deployed on a physical robot, that is exposed to congruent and incongruent social stimuli. The second study details an engineered solution for evaluating a scanpath model on a physical robot. The unified late integration variant of the GASP model (DAM + LARGMU) with fixation history developed in Chapter 4, is used to compare the performance of individuals to those of the model. This approach allows us to evaluate scanpath models on a physical robot without having to conduct separate human-robot interaction studies to validate every modification applied to those models. Moreover, these studies show that although the models developed in the course of this thesis do not yet measure up to human performance, they are utilizable under real-world conditions. This is evident from the trends in the models' predictions, closely matching those observed in human behaviors when performing the same tasks, under similar conditions.

7.1 Audiovisual Social Cueing

7.1.1 Introduction

Conflicting cues often occur during social interactions e.g., speaking to one person while looking at another, pointing at something unrelated to the subject of conversation, illusions such as the ventriloquism effect, where speech appears to emit from a different speaker due to lip movement [12], etc. How we process these conflicts is shaped by our attentional tendencies, preconditioned on the forms of stimuli, their conspicuity, and our physiology. When presented with conflicting cues, humans learn to pay attention to one or either depending on the task presented to them. The Posner cueing task [202] is a psychological test that assesses cognitive visual attention by presenting participants with arrows—located in the center of the monitor—pointing left or right, toward or opposing the direction of a target cue. Trials are considered valid when the arrow points in the same direction as the

target cue that appears shortly after it. Invalid trials denote incongruence, that is, when the arrow points in one direction, and the cue appears on the opposite side of the monitor. A third trial condition, described as neutral, refers to the stimulus appearing on either side of the monitor, however, two arrows appear pointing both left and right. The typical study takes place by having participants seated in front of a monitor, and tasked with immediately reporting the direction of the target stimuli. This is done by pressing a key on a computer keyboard, representing the direction of the target cue. In this study, we devise a similar task to the Posner cueing task. This task is designed to measure the reaction time between the appearance of the target cue and keyboard press under three conditions. In studies presented as part of this thesis, we disregard the reaction time, as we employ models that infer the output at constant time intervals. Instead, we focus solely on the difference in accuracy between a robot and humans performing our variant of the Posner cueing task. We replace the arrows and target stimuli utilized in the original Posner cueing task with multimodal social cues. This allows us to evaluate the ability of humans in localizing sound direction of arrival, in the presence of non-verbal visual social cues.

Social cues attract attention more prominently than salient (bottom-up) objects. Studies show that head orientation is a primary social cue for triggering the reflexive attention of an observer [191]. Studies by Parisi *et al.* [198] and Fu *et al.* [90] on audiovisual social cueing show that lip movements are more salient to humans than arm movements when tasked with localizing sound. This is due to the physical association between lip movement and speech [283]. Consequently, this imposes a strong bias on the participants, directing their attention toward the auditory target's location. Therefore, we conduct a study similar to that introduced by Parisi *et al.* [198] while obscuring lip movements and replacing the deictic social gestures used in their study with head orientation. We present the participants with short videos of three virtual avatars, two standing on either side facing each other, and one avatar in the middle facing the observer. The avatars wear medical masks to conceal their lip movements. The avatar in the middle orients its head toward the avatar on either side, while the word 'hello' is transmitted through stereo speakers. In Figure 7.1, we illustrate the conditions to which the observers (participants) are exposed. When the direction of the avatar's gaze and sound location match, this is what we describe as the congruent condition. A mismatch in direction is referred to as the incongruent condition and the neutral condition does not involve gaze movement, while the sound arrives from either side.

To assess whether a social attention model exhibits similar cognitive attention, we adapt the GASP model introduced in Chapter 3 to the social cueing task and mount it on a robotic platform. The GASP model integrates social cues to predict saliency. The social cues required for our task match those integrated by the GASP model, namely, gaze-following, which indicates the observed social actor's target of attention, and gaze estimation, which informs the model on the actor's gaze direction. The GASP model additionally integrates an audiovisual saliency predictor with the aforementioned social cues. However, the saliency predictor operates on monaural auditory input, which is insufficient to localize sound. We,

(a) Left Gaze

(b) Right Gaze

(c) Neutral Gaze

Figure 7.1: The six combinations covering the three congruence conditions—congruent, incongruent, and neutral. The avatar is the middle either **(a)** looks to the left while the audio arrives from the left (congruent) or right (incongruent) speaker, **(b)** looks to the right while the audio arrives from the right (congruent) or left (incongruent) speaker, and **(c)** looks straight-ahead while the audio arrives from either speaker (neutral).

therefore, modify the DAVE [245] saliency predictor used by the GASP model, by extending it with an additional auditory input channel to encode stereo audio. We describe it as the binaural DAVE model hereafter. Binaural DAVE is an audiovisual Sound Source Localization (SSL) model that attends to regions in the observable visual field from which the sound is estimated to arrive.

Our objective in this study is to adapt our social attention model for predicting the target of attention rather than exclusively localizing sound. For instance, using a unimodal SSL model will likely yield better results than an audiovisual SSL, since the former receives auditory input that is undistorted by visual cues. However, we rely on a binaural audiovisual model to evaluate the effect of incongruency between visual and auditory social cues on our social attention model. Additionally, by using the binaural audiovisual model, we can evaluate how our social attention model measures in comparison to humans under similar conditions. Finally, we assess the efficacy of using the GASP model in tandem with binaural DAVE on a physical robot. To achieve that, we employ the iCub [171] robot, a humanoid robot that is compatible with our study design due to its anthropomorphic and biomechanically-inspired structure—binaural microphones mounted on either ear and cameras attached to the pupils of its eyes.

7.1.2 Sound Source Localization

Recent studies have shown that the typical human brain partially processes natural sounds in the visual cortex [261]. Interestingly, blind individuals who rely on echolocation also localize sounds in the visual cortex [189] as evidenced by the observation of retinotopic-like[1] maps in the early visual cortical processing areas, on exposure to binaural sound recordings. These findings show that the human processes auditory information in the same regions dedicated to processing vision, and that brain regions are organized by task rather than sensory modalities. More importantly, we infer that auditory stimuli can influence our visual perception, and consequently, our visual attention.

By relying solely on visual stimuli, we can, for instance, attend to active speakers by observing their lip movements, or know with great certainty that a musician is playing an instrument by simply watching them perform. Inferring that a sound results from motion is reasonable, yet is not a sufficient assumption. However, knowing that a sound is produced by a source at a given location, would no longer require us relying exclusively on such inductive reasoning. We can do so by jointly localizing the sound source and attending to the movement. To localize sound, humans rely on several cues, including the interaural time and level differences between the auditory stimuli arriving at either ear. In other words, perceiving sound from both ears (binaurally) is a prerequisite for accurate sound localization. However, measuring the time or level difference only allows us to estimate the azimuth of sound. The shape of the pinnae, the Head-Related Transfer Function (HRTF) [182], learned experiences, spectral features, and cues from other modalities such as vision, all assist in estimating the elevation of sound.

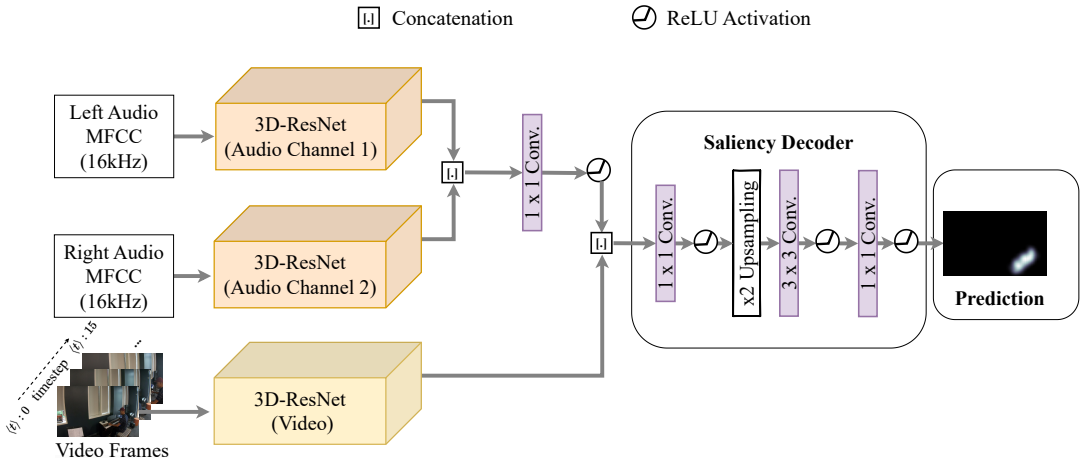
Researchers have more recently sought to integrate auditory information into computational models, designed for predicting visual attention [245, 256, 121, 270]. However, these approaches rely on monaural audio, which does not allow for the accurate localization of sound in videos. To create robotic or computational systems that can localize binaural sound sources in the vertical and horizontal planes, the most prevalent techniques involve estimating the HRTF [116], conducting spectral feature extraction [85], and integrating audio and visual information [207]. Wu *et al.* [268] propose a binaural audiovisual model for localizing sound. Their training procedure involves applying spatial perturbations to the auditory signal. The auditory stream encoders are gated with the visual streams at multiple stages, maintaining sequential and spatial knowledge by utilizing ConvLSTM [232] blocks. Finally, the model predicts the sound location map indicating the sound intensity at various locations within the visible field of view. Rachavarapu *et al.* [207] design an end-to-end model that localizes sound based on synthesized binaural audio. The synthesis model is supervised by the localization model, which in turn relies on the generative synthesis model. Such a closed-loop approach allows the model to constantly refine the predictions as it is further trained. These models either apply

[1]Retinotopic maps are fMRI representations matching visual input from the retina to the visual cortical neurons. Retinotopic-like mapping measures eccentricity in sound and matches it to visual eccentricity in the visual cortex.

multiple transformations to images and audio [268] or involve multiple stages [207]. Our approach uses a two-stage inference pipeline. Integrating it with existing binaural models would significantly increase execution time, making it unsuitable for real-time applications.

To assess the influence of auditory stimuli on visual attention and localization, we construct a deep neural model that predicts visual attention in videos with stereo sound, primed by binaural audio to enable localization. We extend a saliency prediction model [245], trained to predict the attention of a group of observers watching audiovisual content, under the free-viewing condition. Our extension involves the addition of an auditory stream to learn features arriving from two separate audio channels instead of one. Given the feed-forward structure and relatively low parameter count of the model, it allows for the parallelization of the auditory and visual stream encoders during inference. This makes it suitable for near-real-time applications, as the latency is negligible compared to previous binaural audiovisual localization models.

7.1.3 Binaural Deep Audiovisual Embedding Model



MFCC: Mel-Frequency Cepstral Coefficients.

Figure 7.2: Our binaural DAVE model architecture for multimodal audiovisual sound source localization. The model receives two audio channels and a video sequence as input, and encodes each stream using a separate 3D-ResNet [105].

Our architecture is based on the Deep Audiovisual Embedding (DAVE) [245] model. In its original form, the audiovisual monaural DAVE model encodes input from one video and one audio stream, which are projected onto a feature space of 3D-ResNets [105] (one for each input stream). 3D-ResNet extends the ResNet model [109] to operate on multiple frames by replacing 2D convolutional layers with their 3D counterparts. Its encoder is followed by a convolutional saliency decoder that upscales the latent representation and provides the corresponding saliency map.

In this work, we extend the DAVE model to accept binaural input. This binaural extension follows a similar structure to the monaural DAVE model, as shown in Figure 7.2. However, the main difference is that we employ two 3D-ResNets to process the auditory modality rather than the original single 3D-ResNet for processing monaural audio. The output of each stream is concatenated, encoded, and downsampled using a two-dimensional 1×1 convolutional layer. This layer is responsible for guaranteeing that the dimension of the feature produced by this branch of our architecture matches that of the features produced by the original DAVE model's audio-stream 3D-ResNet. Our extension of the DAVE model not only introduces another source of input but enables sound localization as it allows the model to learn patterns that distinguish the audio signals arriving at either auditory stream.

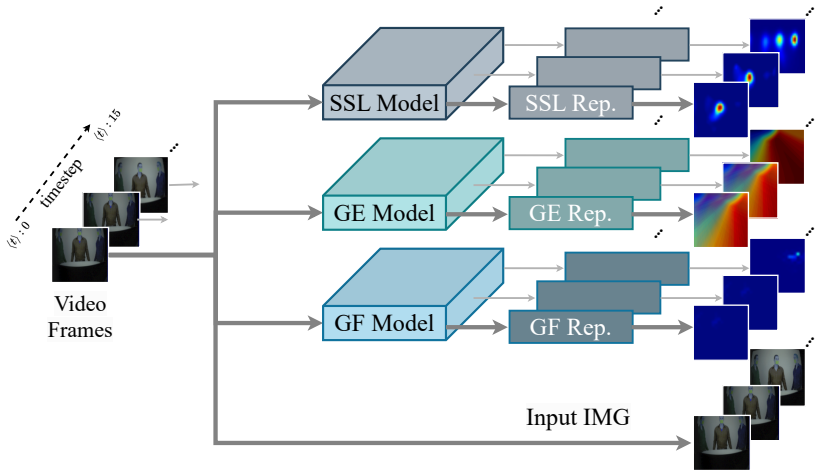
We initialize the binaural DAVE with the pre-trained parameters of the audiovisual DAVE [245]. The left and right auditory streams are initialized with identical parameter weights extracted from the 3D-ResNet auditory stream of the monaural variant. The 1×1 convolutional layer that encodes the concatenated audio features is initialized using the normalization method proposed by He *et al.* [108]. All model parameters are optimized except for the video 3D-ResNet's, which are frozen throughout optimization following the DAVE model's training procedure [245].

7.1.4 Dynamic Saliency Prediction

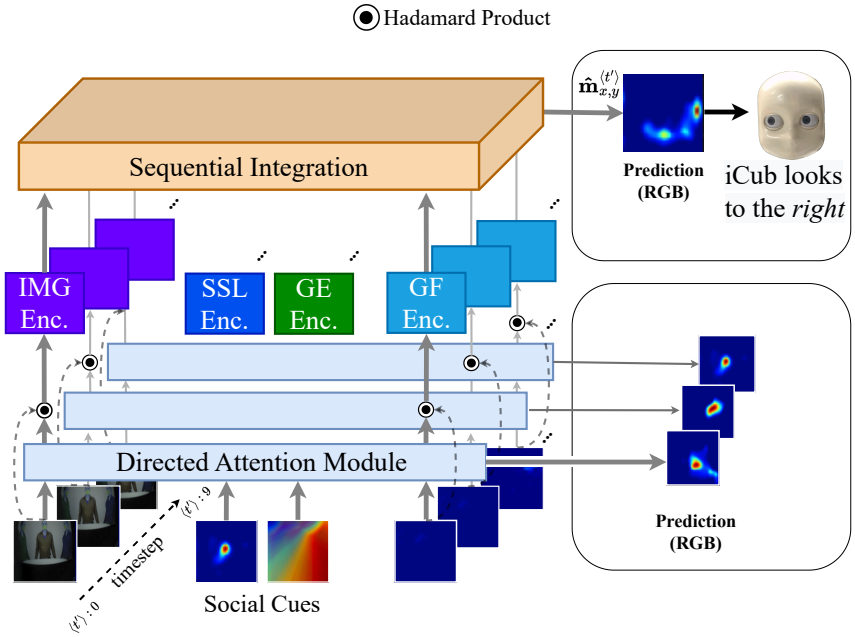
The process of predicting social attention in GASP (introduced in Chapter 3 is separated into two stages. The first stage, Social Cue Detection (SCD), is responsible for extracting social cue representations from a given audiovisual sequence. Figure 7.3a depicts the architecture of the SCD stage. Given a sequence of images and their corresponding representations, the GASP saliency prediction model then infers the corresponding salient region by integrating the social cue representation sequences. The overall integration pipeline followed by the GASP model is shown in Figure 7.3b and detailed in Chapter 3.

Based on the implementation detailed in Section 3.2, the SCD stage comprises four modules, three of which are dedicated to extracting a different social cue and one for audiovisual saliency prediction. Those modules include gaze following, gaze estimation, and facial expression recognition. For the current task, however, the facial expression recognition module is not integrated into the SCD pipeline. This is due to the virtual avatar faces being partially occluded and therefore, not displaying complete facial expressions. In order to closely replicate the human study described by Fu *et al.* [88], the iCub robot receives auditory stimuli from both of its microphones, simulating binaural hearing in humans. The audiovisual saliency prediction module integrated into the SCD stage is designed with a monaural audio stream. To operate on binaural stimuli, we replace the saliency prediction module with a binaural audiovisual Sound Source localization (SSL) model, denoted by the SSL model in Figure 7.3a. The binaural SSL model architecture is shown in Figure 7.2 following the structure described in Section 7.1.3.

The video streams used as input are split into frames and their corresponding



(a) The social cue detection stage pipeline.



(b) The GASP model for sequential integration.

IMG: Input Image; SSL: Sound Source Localization; GE: Gaze Estimation;
GF: Gaze Following.
Rep.: Representation Transformation; Enc.: Modality Encoder.

Figure 7.3: The SCD **(a)** is the social cue detection stage in which the representations of the sound source localization (SSL) and social cues are extracted, whereas GASP **(b)** is our social attention model receiving input from the SCD stage. $\hat{\mathbf{m}}_{x,y}^{\langle t' \rangle}$ represents the fixation density map predicted by the model at timestep t'. For illustration purposes only, the prediction is colorized (RGB jet colormap), with the color red indicating the peak on the fixation density map.

auditory chunk. For every video frame and corresponding audio chunk, the SCD stage extracts social cue and SSL representations, which are then propagated to the GASP model. The Directed Attention Module (DAM) weighs the representation feature map channels to emphasize those that represent high unexpectedness with respect to their predictions. Convolutional layers further encode those weighted feature map channels. In Figure 7.3b, these layers are denoted by *Enc.* (for encoder). The encoded feature maps of all video frames are then integrated using a recurrent extension of the convolutional Gated Multimodal Unit (GMU) [17]. The GMU's mechanism weighs the features of its multimodal input. The convolutional variant of the GMU which we employ accounts for learning spatial properties of the input features. The recurrent integration module of the GASP model considers the entire sequence of frames by performing the gated integration at every timestep.

In this work, we rely on the Late Attentive Recurrent Gated Multimodal Unit (LARGMU) GASP variant due to its better performance compared to other model variations as shown in Section 3.5.2. The LARGMU's recurrent structure allows it to integrate sequential features. Adding a soft-attention mechanism based on the convolutional Attentive Long Short-Term Memory (ALSTM) [63] prevents gradients from vanishing as feature sequences get sufficiently large. LARGMU is a late integration model, meaning that the gated integration is performed after the input channels are concatenated and, in sequence, propagated to the ALSTM. Additionally, we retain the Directed Attention Module (DAM) of LARGMU to avoid the GASP model relying exclusively on the binaural DAVE SSL model.

7.1.5 Binaural Gated Attention for Saliency Prediction

We employ the pretrained DAM+LARGU variant of the GASP model excluding the facial expression recognition input stream. We replace the audiovisual saliency detector with the sound localization variant of the DAVE model, binaural DAVE. In Section 3.5.4, we show that replacing saliency predictors does not require re-training GASP, allowing us to plug in the SSL model in place of the saliency prediction model without fine-tuning the sequential integration model's parameters.

GASP receives four sequences of data as input, one sequence of consecutive frames of the original video, and three sequences of feature maps, one for each model in the SCD stage. In our study, we capture sequences of 10 frames (timesteps $t' : 0$ to $t' : 9$ as shown in Figure 7.3b). The number of frames received as input by each model in the SCD stage varies due to dissimilarity in their expected inputs. The sound localization model receives a sequence of 16 frames as input, whereas the gaze estimation and following models receive sequences of 7 frames each. A more detailed explanation of how the frames are selected based on the timestep being processed is provided in Chapter 3. The auditory input is captured as a one-second chunk and propagated to each audio 3D-ResNet of the SSL model. In this study, the GASP model is embodied in the iCub robot which is exposed to the same series of one-second videos as were the participants. The one-second chunk used as input to the binaural SSL model corresponds to the entire audio recording per video.

7.1.6 iCub Eye Movement Determination

The social cue detectors and saliency predictor extract features for the previously acquired audiovisual frames during the auditory and visual acquisition phase. Following the detection and generation of spatiotemporal maps, the GASP model predicts a fixation density map $\hat{\mathbf{m}}^{\langle t \rangle} : \mathbb{Z}^2 \to [0, 1]$ for a given frame. The peak is registered in pixel coordinates and remapped to scalar values within the range of $\in \{-1, 1\}$ in both x and y axes, such that:

$$\hat{\mathbf{p}}_{x,y} = -1 + \frac{2 \cdot \arg\max_{x,y} \hat{\mathbf{m}}(x, y)}{\hat{\mathbf{m}}_{X,Y}}, \tag{7.1}$$

where $\hat{\mathbf{p}}_{x,y}$ represents the peak location in the normalized range and $\hat{\mathbf{m}}_{X,Y}$ are the width and height of the predicted fixation density map in pixels. We actuate the robot to look toward the peak. For simplicity, we assume the camera view to be independent of its location relative to the playback monitor. For all experiments, we control the eye movements of the iCub, disregarding vergence effects, microsaccades, and fixation duration. The positions are expressed in Cartesian coordinates, assuming the monitor to be at a distance of $\sim \delta_z$ from the image plane. We scale $\hat{\mathbf{p}}_{x,y}$ by a factor of $\alpha_{x,y} = \{.35, .3\}$ to limit the viewing range of the eyes. We then convert the Cartesian coordinates to spherical coordinates:

$$\begin{aligned} \hat{\mathbf{p}}_{\phi} &= \operatorname{atan}\left(\frac{\alpha_y \cdot \hat{\mathbf{p}}_y}{\delta_z}\right), \\ \hat{\mathbf{p}}_{\theta} &= \operatorname{atan}\left(\frac{\alpha_x \cdot \hat{\mathbf{p}}_x}{\sqrt{{\delta_z}^2 + (\alpha_y \cdot \hat{\mathbf{p}}_y)^2}}\right), \end{aligned} \tag{7.2}$$

where $\hat{\mathbf{p}}_{\phi}$ and $\hat{\mathbf{p}}_{\theta}$ are the pitch and yaw angles respectively. These angles are used to actuate the eyes of the iCub such that they tilt $\sim 24°$ and pan $\sim 27°$ at most[2].

7.1.7 Study Design

Task and Procedure

The participants began the experiment with 30 practice trials and entered into the formal test when their accuracy of practice trials reached 90%. Each condition was repeated 96 times, with a total of 288 trials separated into four blocks. There was a 1-minute rest break between every two blocks. The time duration for each trial was set to 1900-2300 ms, and the formal test lasted for 12 minutes per participant.

In each trial, participants watched short video clips with congruent, incongruent, or neutral social cues. The trials were repeated with the iCub robot as shown in Figure 7.4. The videos showed three virtual avatars, with one in the middle looking to the right or left. Virtual avatars were chosen over recordings of humans, as the experiment requires strict control over the avatar's behavior, both in terms of timing and exact motion. By using synthetic data as the experimental stimuli, it can

[2]The iCub can tilt and pan its eyes in ranges of $\in \{-40°, 40°\}$ and $\in \{-45°, 45°\}$ respectively.

(a) Human Study (b) Robot Study

Figure 7.4: The experimental setup shows a participant **(a)** engaging in the formal test with headphones to hear the auditory stimulus, and a keyboard to input whether the sound arrived from the left or right speaker. Similarly, the iCub robot is shown **(b)** engaging in the test having headphones playing back the auditory signal and responding to the target by gazing toward it.

be ensured, for instance, that looking to the left and right are exactly symmetrical motions, thus avoiding any possible bias. Moreover, using three identical avatars that are only different in terms of clothing color also alleviates a bias towards individuals in a real setting. The avatars were created using a data generation framework for research on shared perception and social cue learning [134]. The localized sounds were created using a head-related transfer function[3] that modifies the left and right audio channels to simulate different latencies and damping effects for sounds arriving from different directions.

During the experiment, the participants were asked to determine as soon and as precisely as possible, whether the auditory stimulus originated from the avatar on the left or the right. The participants decided on the direction by pressing the keys 'F' and 'J' on the keyboard, corresponding to the left and right avatars, respectively. The participants' responses during the display of the auditory target and the second fixation were then recorded. The stimulus display and response recording were both managed by *E-prime 2.0*[4].

We reported the Stimulus-Response-Compatibility (SRC) in our results. The SRC effect measures performance in the stimulus-response by comparing the error rates under the congruent and incongruent conditions:

$$SRC = ER_{INC} - ER_{CON}, \tag{7.3}$$

where SRC represents the difference between the error rates under the incongruent ER_{INC} and congruent ER_{CON} conditions. The larger SRC effect may be accompanied by the weaker top-down control, dysfunction, or immaturity of conflict control [60, 169].

[3] https://sound.media.mit.edu/resources/KEMAR.html
[4] https://pstnet.com/products/e-prime/

Experimental Setup

The binaural DAVE was fine-tuned on a subset of the FAIR-Play dataset [94]. The FAIR-Play dataset consists of 1,871 video clips of single or multiple individuals playing musical instruments indoors. Auditory input is binaural with the sound source location maps provided by Wu *et al.* [268]. We fine-tuned the DAVE variants using the 5-fold splits with five training trials per model variant and fold—monaural, binaural, and visual-only—each trial with a random seed used for parameter initialization. Eventually, each model variant was trained for a total of 25 trials (5-folds $\times$ 5 random initialization of parameters).

Similar to the monaural DAVE, the loss of the binaural DAVE model was computed as the Kullback-Leibler divergence between the predicted and ground-truth sound location maps at the last timestep of the 16-frame sequence. The input frames, sound channels, and ground-truth maps were together flipped at random during training as an augmentation transform. We used the Adam optimizer with $\beta_1 = .9$, $\beta_2 = .999$, and a learning rate of .001. The model was trained for five epochs with mini-batches containing four sequences of 16 visual frames, each corresponding to one-second stereo recordings of audio. An NVIDIA RTX 3080 Ti with 11 GB VRAM and 32 GB RAM was used to train all DAVE variants.

Given the close resemblance of audiovisual sound localization to the saliency modeling task, we relied on metrics commonly used to evaluate the latter [43]. We measured Pearson's Correlation Coefficient (CC) and similarity (SIM) between the ground-truth and predicted maps, to quantify the performance of our model. CC measures the linear correlation between two normalized variables, whereas SIM signifies the similarity between two distributions. A similarity of 1 indicates that any two distributions are identical.

Participants watched the short, 3-avatar videos under normal indoor light conditions. Auditory noise in their surroundings was minimal, and the room acoustic effects were negligible since the sound was played directly through on-ear headphones. In our 3-avatar scenario, the auditory directions were frontal left and frontal right at 60°, corresponding to the positions where the peripheral avatars stand. During the experiment, the participants sat positioned 55 cm from the monitor at a desk and wore headphones, as depicted in Figure 7.4a The human and robot experimental setups closely resembled each other as shown in Figure 7.4b, allowing us to simulate the environmental setting experienced by the participants.

However, some adjustments were required to replicate the human experiments on the iCub head as closely as possible. First, the iCub head was placed at a distance of $\sim$30 cm from a 24-inch monitor (1920×1200 px resolution), as depicted in Figure 7.4b. This distance is, however, shorter than the 55 cm distance the participants sat from the computer monitor. The distance reduction was performed so that the iCub robot's field of vision covers a larger portion of the monitor. Since the robot lacks foveated vision, the attention is distributed uniformly to all visible regions, causing the robot to attend to irrelevant environmental changes or visual distractors. Second, the previous robot's eye fixation position needed to be retained as a starting point for the next trial to provide scenery variations

to the model. Direct light sources also needed to be switched off to avoid glare. Once the experimental setup was ready, the pipeline started the video playback in full-screen mode, simultaneously capturing a 30-frame segment of the video using a single iCub camera[5] along with one-second audio recordings from each microphone[6] mounted on the iCub robot's ears. The video segments were propagated directly from the iCub robot to the neural model pipeline using the YARP [170] middleware. Since YARP transmits messages with a low latency and is supported by the iCub robot, we chose YARP for image transfer. The audio chunks on the other hand were transmitted using the ZeroMQ [113] middleware given its lower packet drop rate compared to YARP. Switching between different middleware was facilitated using *Wrapyfi* (detailed in Chapter 5), a Python wrapper with multi-middleware support, to distribute computing and integrate robots with deep neural models.

The iCub head shifted its eyes toward the auditory target. This differs from how participants responded to the stimuli. The participants provided feedback by pressing a key indicating direction. The robot's direction of gaze on the monitor, whether closer to its leftward or rightward edges, is analogous to humans pushing a button indicating the side, allowing us to compare humans and the robot on this basis. ER can be adequately measured and analyzed as the robot response. One-way repeated measures ANOVA is used to test the SRC effects of the robot's response under the three congruency conditions (congruent, incongruent, and neutral). All post hoc tests in the current study use Bonferroni correction. Additionally, an independent t-test is conducted to compare the difference in SRC effects between humans and the robot.

Participants

37 participants (female = 20, male = 17) took part in this experiment. Participants were between 18 and 29 years of age, with a mean age of 22.89 years. All participants reported no history of neurological conditions (seizures, epilepsy, stroke, etc.) and had either normal or corrected-to-normal vision and hearing. This study was conducted following the principles expressed in the Declaration of Helsinki. Each participant signed a consent form approved by the Ethics Committee of the Institute of Psychology, Chinese Academy of Sciences.

7.1.8 Results

Binaural Sound Source Localization

We compared the predicted sound location maps against the ground-truth maps for all video frames. The input consisted of the preceding 15 frames of a given video's final frame at timestep $t : 15$ including the final frame. The evaluation results were reported following the fifth training epoch, given that the validation loss increased after the fifth epoch. The binaural DAVE outperformed both the audiovisual and

[5] http://wiki.icub.org/wiki/Cameras
[6] http://wiki.icub.org/wiki/Microphones

Table 7.1: Average 5-fold cross-validation results on the FAIR-Play dataset using our binaural audiovisual DAVE model for sound source localization. **Bold** denotes the best scores.

Model Architecture	CC↑	SIM↑
Visual Only DAVE	0.5030 ± 0.0032	0.3972 ± 0.0018
Audiovisual DAVE	0.6068 ± 0.0027	0.4398 ± 0.0017
Binaural Audiovisual DAVE *(Ours)*	**0.6411** ± 0.0016	**0.5050** ± 0.0009

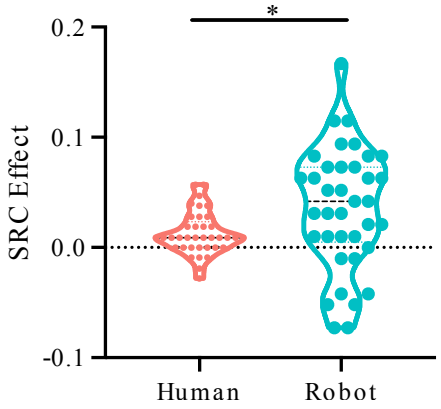
visual-only variants of DAVE, as shown in Table 7.1. The variance across five trials was also lower for the binaural DAVE indicating higher stability given the implicit information about the source locations.

Error Rates in Robot Prediction

A repeated measures ANOVA with a Greenhouse-Geisser correction revealed that the robot's ER differed significantly between different congruency conditions, $F\,(2, 34) = 8.02$, $p < .01$, $\eta_p^2 = .18$ (see Figure 7.6c and Figure 7.6d). Post hoc tests showed that the robot presented significantly lower ER under the congruent condition (mean ± SE = $.37 \pm .01$) than the incongruent condition (mean ± SE = $.41 \pm .01$), $p < .01$. However, there was no statistical significance in the difference between the neutral condition (mean ± SE = $.38 \pm .01$) and the two other congruency conditions, $p > .05$ in both cases.

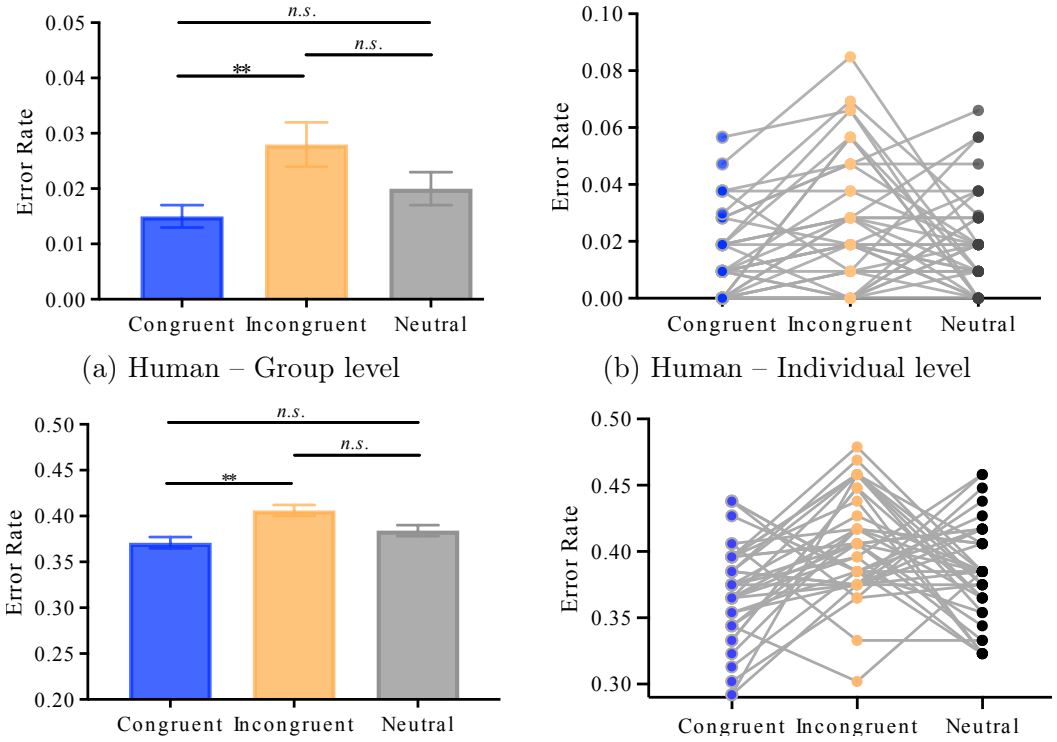
Human-Robot Comparison

Results of the t-test displayed that the robot had a significantly larger SRC effect (mean ± SE = $.04 \pm .001$) than humans (mean ± SE = $.01 \pm .01$), $t\,(72) = 2.35$, $p < .05$ (see Figure 7.5).



∗ denotes $.01 < p < .05$.

Figure 7.5: SRC effects comparison between humans and the iCub robot.



(a) Human – Group level

(b) Human – Individual level

(c) iCub – Group level

(d) iCub – Individual level

$*$ denotes $.01 < p < .05$, $**$ $.001 < p < .01$, $***$ $p < .001$, and *n.s.* denotes no significance.

Figure 7.6: Error rates of **(a)** participants under different congruency conditions – group level, **(b)** participants under different congruency conditions – individual level, **(c)** the iCub robot under different congruency conditions – group level, and **(d)** the iCub robot under different congruency conditions – individual level.

These results verify that similarly to humans, the robot's response accuracy is significantly better ($p < .01$) in a congruent condition than in an incongruent one. This similarity is further corroborated by the lack of significant difference ($p > .05$) in both the humans' and the robot's ER in the neutral condition compared to the other two conditions (see Figure 7.6a and Figure 7.6c).

7.1.9 Discussion

In Section 7.1.8, we observed a significant gap in SIM, but not in CC, between the binaural DAVE model and other variants. The SIM metric is highly sensitive to false negatives [43]. Given the objective of localizing sounds in the visual stream, saliency prediction models would produce maps uncorrelated with regions having high sound activity. In the case of audiovisual and video-only variants, the models are unaware of the sound location and rely on the activity observed in the audiovisual and visual streams, respectively. This implies that those model variants behave like saliency predictors rather than sound localizers.

In Figure 7.7, we observe that the predictions highly corresponded to the ground-

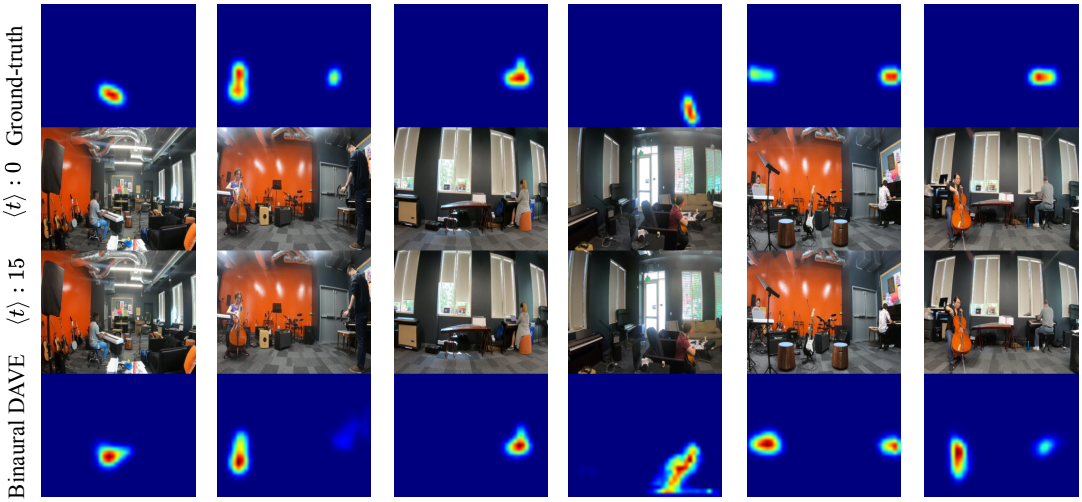


Figure 7.7: Qualitative examples showing the binaural audiovisual DAVE predictions on the FAIR-Play [94] test subset.

truth maps, with the exception of incorrect predictions displayed in the last two columns. We note that such false predictions often occur due to the labels being provided as constant sound location maps for entire video clips [268]. Changes during the video in which one musician begins playing at a later time beyond the start of the clip were ignored, as seen from the example shown in the last column of Figure 7.7. The musician played the cello as indicated by the hand movement in transition between the timesteps $t : 0$ and $t : 15$. The model accurately inferred the location of the sound source (the cello), given that a majority of the training samples did not present transitions between sources along the entire clip. We hypothesize that datasets labeled with higher granularity across time would lead to better performance.

Compared to the results reported by Wu *et al.* [268] on the FAIR-Play dataset, our binaural DAVE model did not perform as well. However, our training and evaluation schemes are not directly comparable since we have opted for a 5-fold cross-validation scheme, whereas Wu *et al.* [268] trained and evaluated on predefined data splits. Our choice was driven by the fact that all frames in any video clip were labeled according to the sound source midway through the clip, resulting in a non-representative ground-truth map for some videos. Using predefined train and test sets as those proposed by Gao and Grauman [94] for the FAIR-Play dataset would not have been suitable for this task specifically, given the labels provided by Wu *et al.* [268]. A worse score achieved on those sets is not indicative of the model's generalization capability, since more videos with final-midway frame mismatch—in terms of sound source—could hypothetically belong to the test set. This could lower the model's score but does not necessarily indicate worse performance, since the model could accurately localize sound, even when the ground-truth is incorrect.

In Section 7.1.8, the robot showed significantly larger SRC effects with higher variability than humans. Although minimal, the iCub robot's ego noise makes audio localization more challenging than it is for a human, who is capable of adjusting

to the surrounding noise in a relatively short period. In contrast, the iCub robot relied solely on the learned behaviors of its pretrained model, without any form of adaptation. Moreover, the form of response differed between humans and the robot. The human participants responded to the stimuli by pressing a key that corresponds to the auditory cue's direction, whereas the iCub robot shifted its gaze toward the target. The social attention model running on the robot predicts a fixation density map, representing the most likely region a human would tend to fixate in a multimodal audiovisual scenario. However, the difference between the tasks presented to the participants and for which the robot was trained led to a higher difference in the SRC effects.

We additionally attribute the lower performance of the robot to the difference in training and evaluation environments. The model was trained on streamed videos—original video files from the datasets on which our models were trained and evaluated—that covered the entire perceptual field and were of higher quality than what the robot perceived with its sensors. This includes the robot's camera resolution and its distortion effects, the quality of the headphones and the robot's microphones in terms of signal-to-noise ratio and sampling rate, aliasing effects arising from capturing the display monitor, and the display monitor's resolution and refresh rate. Physical environmental factors, such as the lighting conditions, reflections on the monitor, and distractors such as the edges of the monitor or items within the robot's view, place the robot at a disadvantage compared to humans.

The trends in error rate difference across the three congruency conditions were similar for humans and the robot as shown in Section 7.1.8. This indicates that when social and auditory cues are in conflict, human attention to the target audio reduces, which is a trait that is learned by the social attention model as well. Moreover, it is evident that the social attention model attends to sounds and that the sound source is often identified by the SSL model. When the visual cue guides attention to a target opposing the sound location inferred by the SSL model, it leads to lower accuracy in prediction. Therefore, we deduce that the social attention model does consider the information arriving from the SSL model. Otherwise, changing the location of the sound would not affect the model, and more importantly, the congruent condition would not result in a better performance. However, our model performs better under congruent and neutral conditions, indicating that the direction of the social cue and the sound location, guide our model's attention.

The purpose of our study was to examine whether conscious decisions made in solving an audiovisual social cueing task are reflected in the attention behaviors learned with a social attention saliency model. However, such models represent the attention patterns of a group of observers, yet humans express different attention behaviors on an individual level [126]. Therefore, we design a study in Section 7.2, to evaluate robots deploying models that consider these differences.

7.2 Personalized Social Attention

7.2.1 Introduction

Scaling social robot studies is challenging since most depend on human perception arising from interactions with the robots. Recruiting a large number of human participants for conducting these studies is generally impractical in terms of time and resource investment. Robotics simulators have emerged as a solution to the scaling problem in the social robotics sphere. However, although the physics and aesthetic realism of robotics simulators has been advancing rapidly, the fidelity of social cue (socialness) simulation is still limited. For instance, automating the testing of embodied social models, such as those employed for the social navigation task [50], is made possible with large generative language and multimodal models. Such approaches rely on simulating social behavior in the form of abstract action primitives using generative models [166]. However, the level of abstraction and quality of generated outputs could misrepresent real-world conditions under which the robot operates, potentially leading to inaccurate and unsafe behavior [167].

Moreover, the spectrum of social cues displayed by humans during social interactions is broad and context-dependent. Such social cues include facial expressions and gaze direction among other [262]. Considering social cues is especially relevant when evaluating social attention models. Social attention—saliency and scanpath prediction—models predict human gaze by integrating social cues. To address the limitations of social cue simulation, we present a cognitive robotic simulation scheme that allows us to evaluate social attention models in physical environments. We simplify the problem while still maintaining similarity to the human data collection setup. Additionally, we assume that the physical environment provides a means for allowing the robot to perceive the pre-recorded stimuli. Our approach, although tailored specifically for social attention models, can be applied to other social tasks with varying degrees of complexity.

In more detail, we evaluate our dynamic scanpath prediction model described in Chapter 4, which infers priority maps, indicating the attention region of the individual observer. The peak of the priority map defines the gaze target. Actuating a robot to gaze toward that target, as well as the targets to follow in sequence would effectively simulate human scanpaths. Our approach involves projecting the ground-truth priority maps to a simulated environment. The map is projected to a monitor within the simulator at a distance from the simulated robot, approximately equivalent to the distance of a real monitor from the physical robot. By controlling the gaze of the simulated robot to match the physical robot, the view of the ground-truth priority map resembles the view of the physical robot's predicted map. This allows us to compare the ground-truth to the predicted map using common saliency metrics [43].

Scanpath prediction in near-real-time settings requires models that are robust to intrinsic factors such as camera resolution, focal length, microphone sampling, and sensitivity. Additionally, they must be resilient to extrinsic factors like lighting conditions, background clutter, motion blur, and auditory reverberation. Moreover,

structuring the acquisition and execution pipelines greatly influences the performance of a model, since the movement duration of a robot has to be factored in. To evaluate the applicability of our scanpath prediction models in physical settings, we closely mimic the experimental setup in which the datasets—Findwho [272] and MVVA [158]—were collected, replacing the human observers with the iCub [171] robot. These datasets are composed of social videos that were watched under the free-viewing condition [257, p. 26] by multiple human observers, whose eye movements were collected using an eye tracker. The iCub robot is chosen since it is capable of moving both its head and eyes, with cameras attached to its pupils and microphones mounted on both its ears.

7.2.2 Cognitive Robotic Simulation of Scanpath Prediction

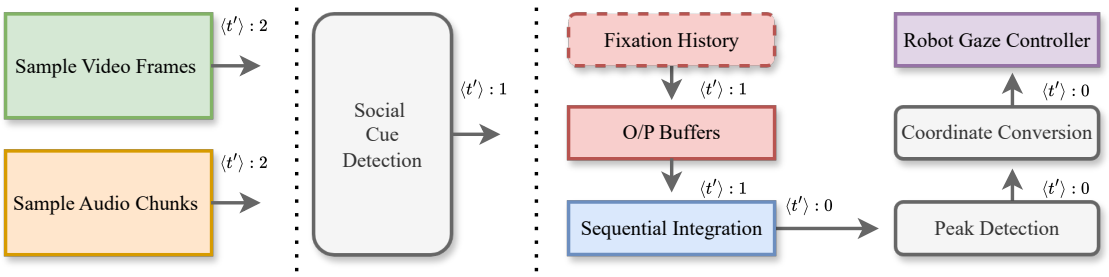


Figure 7.8: The scanpath control pipeline for actuating the robot. Assuming the current timestep at $\langle t' \rangle : 0$, we show the output availability for each component at the relative timestep. Video and audio sampling are performed in parallel, blocking all other components to avoid interrupting the atomicity of real-time capture.

Videos in the MVVA [158] and FindWho [272] datasets are played on a monitor facing the iCub robot [171]. The robot captures those videos with its camera and microphones. Following capture, the social cue and saliency prediction modalities are executed and their representations are generated. These representations are queued in the output buffer along with the fixation history—the preceding fixations of the observer under test. Concurrently, the sequential integration model operates on the representations of previous timesteps and predicts an individual observer's priority map. The predicted map is propagated to the peak detector. The peak coordinates are converted to yaw and pitch which are then used to actuate the physical and simulated robot simultaneously using *YARP* [170]. The ground-truth priority map for the last video frame of a given context is channeled to a simulated monitor. Finally, the metrics are computed and the pipeline is looped until all videos in the evaluation set are completed. An overview of the execution pipeline is shown in Figure 7.8. The pipeline defines the steps taken to evaluate our unified scanpath models in real-time.

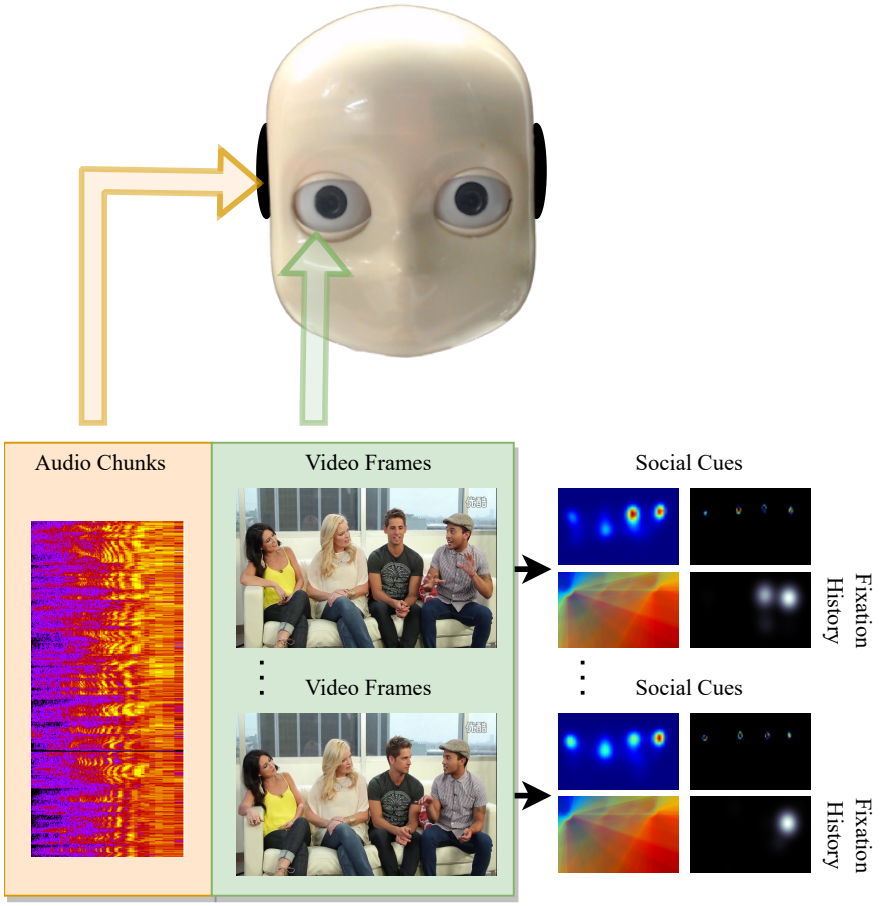


Figure 7.9: The videos are played back in segments on a monitor facing the iCub robot. Audio chunks and video frames are captured through the iCub's sensors. The social cue and saliency features are represented as 2D maps and propagated to the unified scanpath model, which predicts a priority map.

Audio and Video Sampling

Initially, the videos are played on the physical monitor in one-second chunks and are paused until the pipeline repeats. During playback, the iCub robot facing the monitor captures a sequence of images at 10 FPS and audio at 16 kHz. The video playback is executed as a separate process that awaits a signal to resume. This signal is transmitted before the iCub begins capturing one-second chunks of audiovisual frames using its integrated sensors. The communication between the sampling and playback processes is handled by *Wrapyfi* (detailed in Chapter 5). This is performed in a blocking manner to avoid interrupting the capture process.

Social Cue Detection

We utilize the social cue detectors proposed in Chapter 4. This includes the facial expression [234] and gaze estimation [130] cue modalities, along with the saliency prediction [245] modality. The cues are detected, transformed, and represented as shown in Figure 7.9, following the procedure detailed in Section 3.2. The cue detectors extract the representations sequentially and maintain frames and chunks from previously sampled video and audio. As long as the same video is playing, the frames are queued and processed by the detectors according to their context lengths. At the beginning of a video, the frames collected are not sufficient to cover the context length of all detection models. For instance, the DAVE saliency prediction model requires 16 video frames, however, our samplers return 10 frames only. The remaining 6 frames would be padded with the last acquired frame and shifted as more samples are collected. At every timestep, the detected representations are propagated to the output buffers in the form of a single 2D representation per modality.

Fixation History

The fixation history is the sequence of fixations that precede the one being predicted by our sequential integration model (detailed in Chapter 3 and Chapter 4) in the form of a priority map. The fixation history serves the purpose of providing context to our model, in order to inform it on the observer priority map to be predicted. Moreover, the next fixation depends on the previous fixation positions. Without representing the previous scanpath—sequence of fixations—, the predictions would be arbitrary. The fixation history module extracts the ground-truth priority map for a given timestep t' and propagates it to the output buffers.

Output Buffers

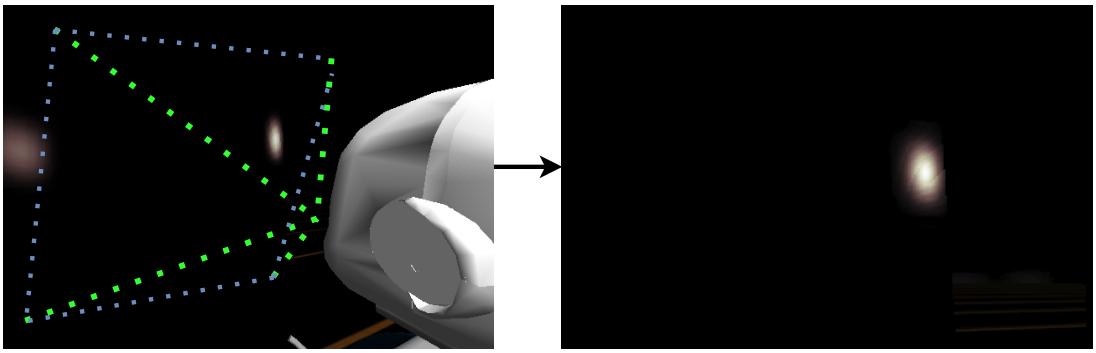
The output buffer represents all queues storing the latest state representations for each modality, agnostic to the input sampling mechanism. At every output timestep t', each modality-specific buffer is queued with a single 2D feature map. The maximum size for all queues is governed by the context size of the sequential integration model.

Sequential Integration

We use the scanpath model developed in Chapter 4. We employ two models trained with the FindWho [272] and MVVA [158] observer data. More specifically, we evaluate the unified integration model. The unified integration model is similar in structure to the sequential integration GASP variants, additionally extended with the fixation history module. The Directed Attention Module (DAM) is trained on the fixation density maps of a group of observers, whereas the Late Attentive Recurrent Gated Multimodal Unit (LARGMU) is trained on the priority maps of all observers individually.

Peak Detection and Coordinate Conversion

(a) Physical environment showing video playback on a monitor facing the iCub robot.



(b) Projection of priority map in simulation.

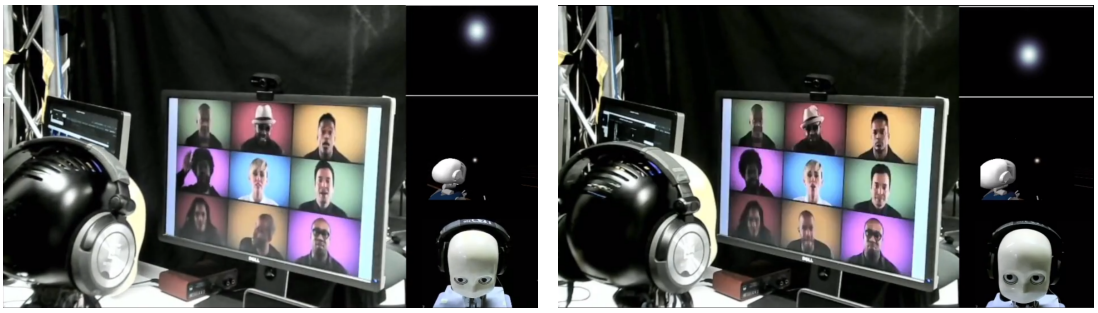
Figure 7.10: Environmental setup where the **(a)** external view of the physical environment *(left)* and the region of capture *(right)* upon which the priority map is inferred, and **(b)** captured camera view from the simulated environment *(right)*, in which the ground-truth priority map is projected on a virtual monitor followed by color-correction and evaluation against the inferred map.

We detail the peak detection and coordinate conversion schemes in Section 7.1.6, replacing the GASP model with our scanpath model predicting a priority map in place of the fixation density map. On extracting the priority map from the scanpath prediction model's output, the peak of the priority map is registered as the target of gaze. The robot captures the images and audio from the environment, applies the scanpath prediction model to the captured stimuli, and directs the robot's gaze toward the peak. Simultaneously, the ground-truth priority map is projected to a monitor within a simulated environment as shown in Figure 7.10, and the peak of the priority map is detected relative to the monitor. Finally, the predicted priority map is evaluated against the simulator-projected ground-truth map.

Robot Gaze Controller

The iCub [171] robot is used in all experiments for evaluating performance on the MVVA [158] and FindWho [272] datasets as shown in Figure 7.11. The MVVA data

collection procedure does not enforce fixing the head pose. To accommodate the influence of the head rotation, we utilize the *iKin* [217] library. More specifically, we aim to evaluate gaze shifts by relying on the iCub robot's vestibulo-ocular reflex functionality to compensate for the head movements resulting from fixating on a target location. The integration of such an effect is necessary due to its impact on stimuli capture as well as the fixations following the current one at any given timestep. For the FindWho evaluation trials, the head pose is fixed such that the iCub's line-of-sight is perpendicular to the monitor. We, therefore, control the eyes directly by specifying the target of gaze as the peak of the predicted priority map in the visible pixel space.



(a) MVVA Evaluation (b) FindWho Evaluation

Figure 7.11: The iCub robot executing the evaluation pipeline based on the **(a)** FindWho [272] dataset allowing for eye movements only, and **(b)** MVVA [158] dataset allowing for head and eye movements. The physical robot *(bottom right)* observes clips *(left)* and predicts a priority map *(top right)* according to the observer under test. The ground-truth priority map is projected to a monitor in simulation *(center right)*, after which the projected ground-truth map and predicted map are compared in terms of the NSS and AUCJ metrics.

7.2.3 Study Design

Mapping Prediction to Ground-Truth Gaze

Videos displayed on a monitor would naturally require a different ground-truth mapping methodology to direct streamed video comparisons. To avoid ambiguity in mapping fixation positions, we project the ground-truth priority map onto a monitor within the iCub simulator, as shown in Figure 7.10b. Knowing the robot's distance from the monitor within the physical environment, we mirror the head and eye movements registered on the physical robot within the simulator, providing an approximate position of the intended fixation. We adjust the ground-truth priority maps to match the size of the monitor in the physical environment from the perspective of the observer. Given the distance from the monitor δ_z during the data collection phase, we can approximate the width and height of the projected ground-truth map by repositioning the simulated monitor at a distance of δ_z from the robot. Next, the simulated monitor is resized to match the size of the physical

Table 7.2: Experimental setup and dataset properties collected under the free-viewing condition.

Property	*MVVA* [158]	*FindWho* [272]
Distance to monitor	∼55 cm	∼60 cm
Monitor resolution	1280×720 px (16:9)	1280×720 px (16:9)
Monitor size	23-inch	23.8-inch
Video duration	10-30s	∼20s
Frames per second	30	25
Audio channels	Stereo	Monaural
Head-pose	Free	Fixed
Eye tracker	EyeLink1000 Plus[7]	Tobii X2-60[8]
No. training videos	210 (70%)	46 (70%)
No. validation videos	30 (10%)	-
No. test videos	60 (20%)	19 (30%)
No. observers	34 (1 excl.)	39

one and the view from the robot's left eye is captured. Finally, the simulated capture is compared to the predicted priority map from the physical environment.

Experimental Setup

In this study, we utilized the pretrained unified scanpath model with the best performance. The integration architecture (DAM + LARGMU, context size $T' = 10$), yielded the best results for a majority of the experiments on both the MVVA [158] and FindWho [272] datasets. The training pipeline is detailed in Section 4.4.

Given the procedural differences in the collection of the MVVA and FindWho datasets, we considered the properties shown in Table 7.2. However, accounting for the robot's visual field and camera resolution, we did not fully align our setup with those properties. For evaluating the MVVA dataset, the corresponding integration model was deployed on the robot. We placed the robot at a distance of ∼30 cm from a 23-inch monitor. For the FindWho dataset evaluation, we moved the robot further from the monitor to a distance of ∼35 cm and deployed the integration variant of our scanpath prediction model, trained on the FindWho dataset. In alignment with the datasets' collection protocols, we set the robot to move its eyes only when evaluating the FindWho dataset. As for the MVVA dataset evaluation, we used the *iKin* [217] library to direct the robot's gaze shift through head and eye movements. Both datasets were evaluated separately. The stimuli videos were replayed a number of times equivalent to the number of individual observers. For each observer, the fixation history consisted of the preceding ground-truth fixations on observing the specific video frames and audio chunk. The videos were played

back at 1 s intervals and captured using the iCub robot's left camera. Audio was played back also for 1 s intervals through on-ear headphones, placed on the iCub robot's microphones.

During the physical evaluation, the social cue detectors and the GASP model were distributed among two NVIDIA GeForce GTX 970 GPUs with a total of 8GB VRAM and 32GB RAM.

7.2.4 Participants

Experiments on the physical robot, evaluating all observers individually required $\sim$13 hours in total for the FindWho dataset (39 observers, 19 videos), and $\sim$42 hours for the MVVA dataset (33 observers, 60 videos). We provide more details about the participants in Section 4.4.1.

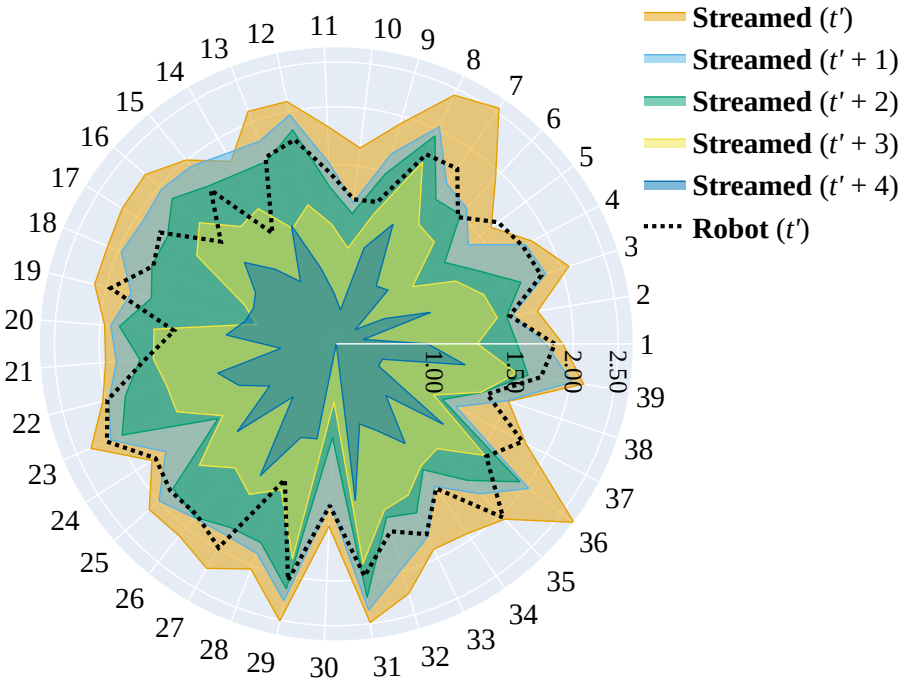
7.2.5 Results

Pearson correlation analyses were conducted to measure the alignment between robot-captured metrics and direct streamed video metrics over one-step and multi-step-ahead time intervals. The robotic experiments were only conducted for one-step-ahead predictions. Multi-step-ahead predictions refer to evaluations extending over multiple future steps. This describes feeding the predicted priority map back into the fixation history module as future samples are collected, where every additional step into the future is denoted by $t' + N$. Here, t' refers to one-step-ahead prediction, and N to the number of additional steps ahead.

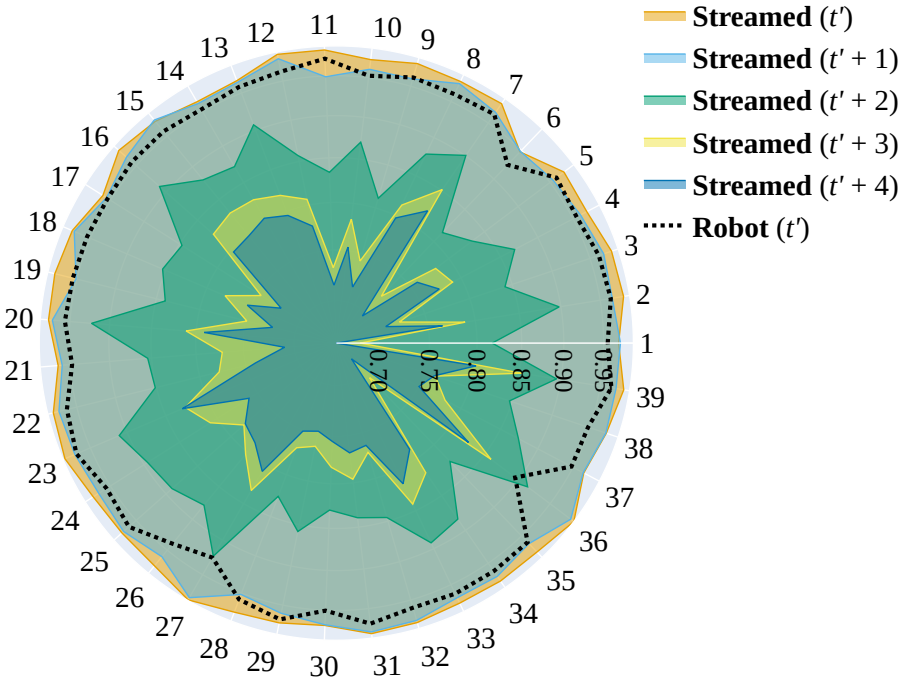
We evaluated the FindWho [158] dataset on the robot, and measured its performance in terms of the NSS [43] and AUCJ [43] (described in Section 4.3) metrics against one-step-ahead and multi-step-ahead streamed video predictions as shown in Figure 7.12. For the NSS metric, moderate correlations were observed between the robot-captured and direct streamed videos (r = 0.498), which decreased with the addition of the steps ahead (r = 0.442 at $t' + 1$, r = 0.401 at $t' + 2$, r = 0.279 at $t' + 3$, and r = 0.336 at $t' + 4$). In contrast, the AUCJ metric exhibited a weak initial correlation (r = 0.165) that turned negative for future predictions (r = -0.142 at $t' + 1$ through r = -0.098 at $t' + 4$), indicating a divergence in attention distribution metrics with step-ahead increments. The full results are listed in Table A.2.

We evaluated the MVVA [158] dataset on the robot, and measured its performance in terms of the NSS and AUCJ metrics against one-step-ahead and multi-step-ahead streamed video predictions as shown in Figure 7.13. For the NSS metric, strong correlations were observed between the robot-captured and direct streamed videos, starting at r = 0.76 for one-step-ahead, with a gradual decrease through the steps ahead (r = 0.74 at $t' + 1$, r = 0.68 at $t' + 2$, and r = 0.63 at $t' + 3$). For the AUCJ metric, a moderate initial correlation (r = 0.48) was observed, which

[8] https://www.sr-research.com/eyelink-1000-plus
[9] https://connect.tobii.com/s/x2-downloads

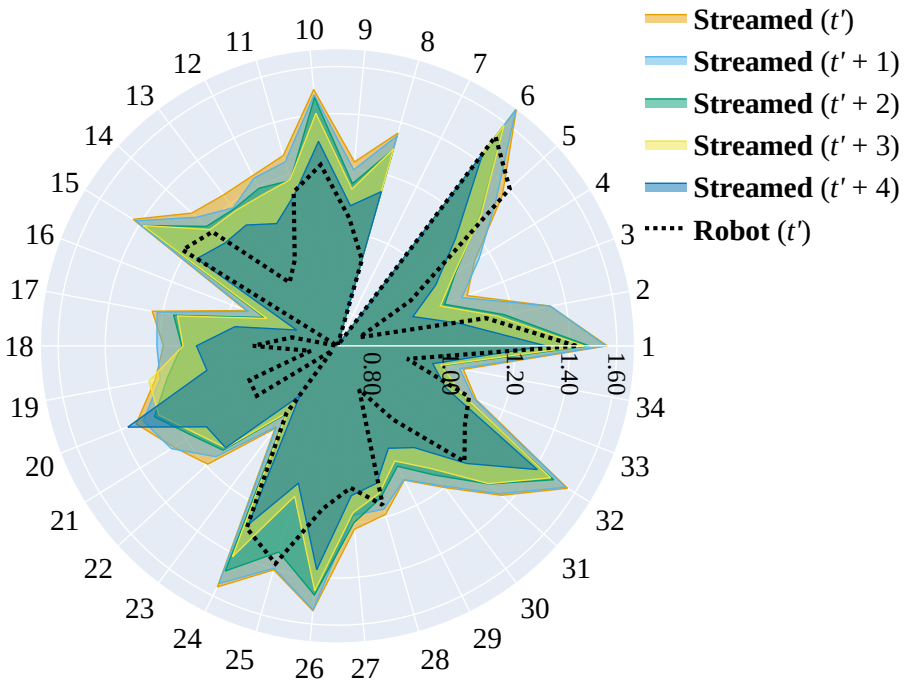


(a) NSS

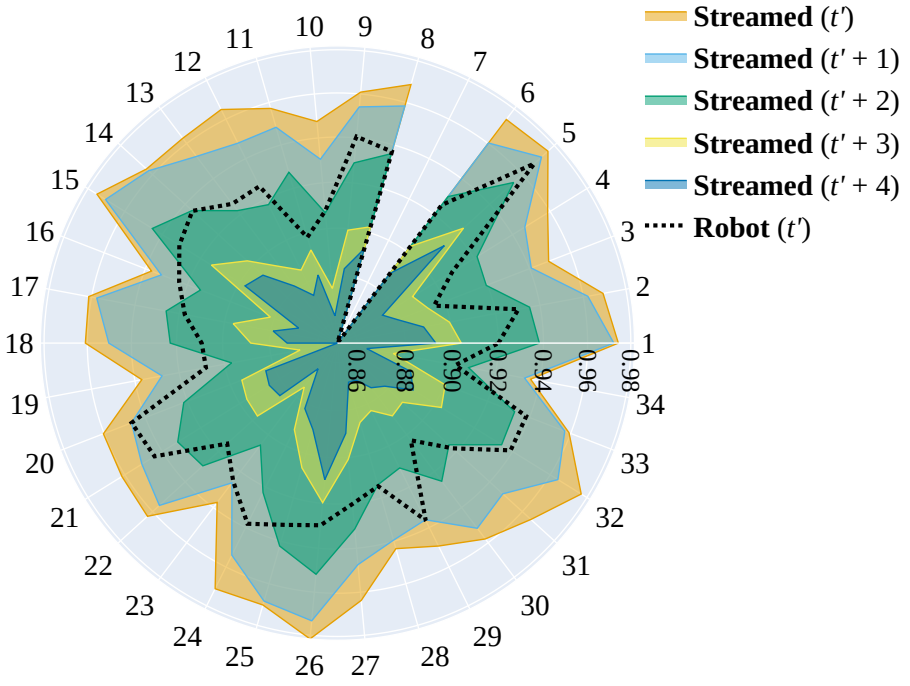


(b) AUCJ

Figure 7.12: Robot one-step-ahead predictions using the unified late integration model (DAM + LARGMU, context size $T' = 10$) trained on the FindWho [272] dataset, compared to the streamed multi-step-ahead predictions in terms of the **(a)** NSS and **(b)** AUCJ metrics. The angular axis indicates the observer identifier, whereas the radial axis shows the metric score.



(a) NSS



(b) AUCJ

Figure 7.13: Robot one-step-ahead predictions using the unified late integration model (DAM + LARGMU, context size $T' = 10$) trained on the MVVA [158] dataset, compared to the streamed multi-step-ahead predictions in terms of the **(a)** NSS and **(b)** AUCJ metrics. The angular axis indicates the observer identifier, whereas the radial axis shows the metric score.

gradually increased for future predictions (r = 0.52 at $t'+1$, r = 0.57 at $t'+2$, and r = 0.59, $t'+3$, and r = 0.59 at $t'+4$), suggesting a strengthening of predictive alignment in visual attention metrics with step-ahead increments. The full results are listed in Table A.3.

7.2.6 Discussion

Expectedly, the robot scored lower than the streamed video evaluation in terms of the NSS and AUCJ metric scores. We found that the observer scores were correlated for streamed and robot-captured videos, when evaluating on the MVVA dataset. The correlation was weaker on the FindWho dataset, even tending towards negative values for the AUCJ correlation, as the number of steps ahead was increased. The AUCJ metric is sensitive to false-positive predictions. When training and evaluating on a relatively small dataset, the mean scores are higher, however, the variance is significantly larger. The unified model trained on the FindWho dataset observed fewer patterns that are universal (universal attention), biasing the model more toward the scanpaths of the individual (personalized attention). This resulted in fewer erroneous predictions overall as evident from the higher NSS score. However, given the small size of the dataset, some of the observers' scanpaths were not learned sufficiently, while others were more similar to the average among the group of observers, and therefore, predicted accurately.

As for the MVVA dataset, which is approximately three times as large as the FindWho dataset, the unified model was exposed to more universal attention patterns. We saw that the robot's predicted gaze was robust to noise, as the scores of all participants were highly correlated with one-step-ahead and multi-step-ahead predictions, both in terms of NSS and AUCJ. Moreover, the larger size of the dataset meant that the evaluation was a better representative of the model's performance in comparison to the smaller FindWho dataset.

We conclude that the unified model is robust to noise, since the input arriving from the robot's camera and microphones differed to a large degree from the streamed videos. The lighting effects, distractors, and lower resolution were not very detrimental to the robot's performance, suggesting that our social attention model can be used in real-world settings to predict personalized scanpaths. We presume, however, that increasing the number of steps ahead during the robot evaluation would degrade its performance further. This reduction in performance was observed for the streamed videos under multi-step-ahead evaluation, suggesting a similar pattern for the robot as well. Our cognitive robotic simulation approach makes it possible to improve our social attention models and evaluate their performances on a physical robot or possibly several robotic platforms, without needing to conduct Human-Robot Interaction (HRI) studies. This has significant advantages, including the enabling of reproducible experiments and scaling of experiments beyond what is possible through HRI. Evaluating the humanlikeness or naturalness of an interaction with a robot would still require conducting HRI studies. However, our approach makes it possible to conduct such studies after the models have been refined and were shown to work sufficiently well in the physical world.

Chapter 8

Discussions and Conclusions

Throughout this thesis, we have tackled challenges extending over several domains, including machine and deep learning, middleware communication systems, Human-Robot Interaction (HRI), robotics, and psychology. We primarily focused on designing an artificial neural network model that integrated information from multiple social cue modalities to assimilate human social attention. We then embodied the model architecture and its variants in robotic platforms to evaluate the robustness of such systems in response to real-world environments and changes. In order to embody our models, we created a multi-middleware wrapper called *Wrapyfi* for establishing communication across different robots, deep-learning models, and devices. This wrapper allowed us to realize and implement two HRI studies, providing us with insights into human responses to robotic gaze and social cue behaviors. Moreover, Wrapyfi made it possible for us to easily replace neural models according to the task under study, allowing us to evaluate our neural models in real-world environments on two cognitive simulation studies using a physical robot. This capability allowed for comparison between human and robot performance on social attention-related tasks.

We believe our approaches have brought value and expanded the body of science relating to social robotics. Our social attention models surpassed other saliency prediction models at the time of development, indicating that the integration of social cues helped improve saliency model predictions in social interaction contexts. However, transferring such models to the real world is still challenging. Although we have circumvented many hurdles that come along with such a transfer, a few fundamental issues remain to be solved. In this chapter, we discuss challenges that were addressed and those yet to be tackled, in relation to social attention modeling and robotic gaze control implementation. We also frame our work within the scope of the research questions posed in Section 1.3, discuss the application potentials of our models, and the future directions for extending and improving the methods elaborated on in this thesis.

8.1 Modeling Social Attention

In the first part of this thesis, we studied common mechanisms of integrating and fusing auxiliary social cue representations using neural attention and gating mechanisms. The objective was to predict the attention target of groups (saliency) and individuals (scanpaths) under the free-viewing condition upon observing dynamic audiovisual social scenes. We observed that training our saliency and scanpath prediction models, given all factors included—gating, attention, integration method, audiovisual input, dynamic input (video), social cues, and fixation history—contributed to our social attention models' state-of-the-art performance. Our novel procedure in representing social cues as images not only allowed for the replacement of social cue detectors without retraining the social attention model but also increased the interpretability of our models. Additionally, we are able to retain the information resulting from the visual social cue representations.

An alternative approach would have been to project the social cue representations to a modality other than vision, for instance, describing facial expressions as text categories or gaze direction in the form of numerical coordinates would hinder spatial attention and, at best, the training phase would require significantly more iterations to learn spatial associations. In other words, the location of an event would potentially be irrelevant given that the shared neural representation of the map—fixation density map (attention map), sound location map, priority map—and the social cue category (or coordinates) do not align spatially or correlate across dimensions.

Another approach, which is more conventional in computer vision and deep learning, relies on the extraction of features from the latent representations of backbone models. These backbones would be the social cue detection models for the purpose of this work. Although the information contained within such representations could potentially deliver more features than our approach, they are likely to have different shapes with varying numbers of neural units. This would therefore require separate encoders for each social cue modality, rendering the different encoders incomparable to each other. Measuring the contribution of each modality to the final prediction is not possible, given that the encoder shapes and unit counts do not match. Moreover, encoding often implies a reduction in dimension. Although we encoded the saliency and social cue representations, our encoders are identical in architecture. Hence, the reduction ratio of encoder input to output is the same, allowing us to replace the saliency and auxiliary social cue modalities with other models trained for the tasks while reusing the trained encoder parameters without further fine-tuning.

We note that while our model architectures are suited for representing social interactions, they underperform when predicting social attention in dynamic scenes not involving humans (non-social interactions). Although intended by design—our thesis is aimed at social interactions—such a limitation becomes more prominent with longer video sequences. Humans could appear in the scene at some point in time but not during the full interaction. Our approaches employed neural gating and attention mechanisms that allowed for the propagation of relevant social cue

and saliency features to the social attention model. This would result in our model greatly relying on the saliency prediction model fed into it as a prior among other features, for non-social interactions. However, we also introduced the Directed Attention Module (DAM) described in Section 3.3.1 within the context of saliency prediction and its relevance to predicting scanpaths in Section 4.2.3. The DAM served as a mechanism to avoid shortcut learning [95], which was inevitable given the approach we followed, where our social attention models relied on pretrained saliency prediction models. This mechanism, nonetheless, also distorts the saliency representation, which is detrimental in the case of non-social interactions.

A necessary consideration when adopting an approach such as ours—GASP variants described in Chapter 3 and extended the variants to operate as a unified scanpath predictor in Chapter 4—is the reliance on visual representations of features. In this thesis, we focused on integrating a limited number of nonverbal social cues. For multimodal social cue detectors or saliency predictors, where the output of each detector could be represented visually, GASP would be an appropriate choice of model. However, once features extend beyond the topographical or spatiotemporal representation domain, projecting those features to the visual domain becomes challenging. For instance, representing an audio spectrogram as a 2D visualization is possible, yet the magnitude at each frequency bin is not tied to the location of a sound source. Although the gating and attention mechanisms in the GASP model are not restricted to operations on spatiotemporal or 2D representations, we lose the homogeneity across the modality encoders and the target maps of the DAM. Due to the reliance of the DAM on the Squeeze-and-Excitation [118] model, which applies channel-wise attention to the different representations—in the GASP model, those channels are the social cue modality and saliency prediction representations, separated by their RGB channels—, learning the saliency based on the inverse (the inverted stream) of the features as performed by the DAM would potentially assign arbitrary weights to each channel. Therefore, excluding the DAM from the GASP model and adapting the modality encoders according to the input shapes would make GASP suitable for non-spatial representations. However, caution must be taken when discarding the DAM to avoid propagating a prior to the model that is highly correlated with the final output representation.

8.2 Controlling Social Robots

In the second part of this thesis, we conducted HRI experiments and performed cognitive simulations on the iCub [171] robot. We also developed a Python framework for cross-communicating between different neural models, robots, applications, and platforms using common message-oriented and robotics middleware. Our objective was to create a tool that would simplify the setting of robotic experiments, enabling us to evaluate our saliency and scanpath prediction models on physical robots.

We conducted two HRI studies detailed in Chapter 6, which were developed using our multi-middleware Python framework—Wrapyfi is described in Chapter 5. In one study described in Section 6.1, we found that the facial expression displayed

by a robot influenced the collaboration between two humans. Humans completed the task at a faster rate and attributed higher intelligence to the iCub robot when the robot expressed 'happiness' before initiating the collaborative task. Additionally, the iCub robot was perceived to be more intelligent on initially establishing mutual gaze—eye contact with the participant on starting the collaborative game task with another participant—with the guide, whose role was to verbally assist an acting participant—human performing the physical action—in completing a collaborative task and to evaluate the robot on task completion.

In another study described in Section 6.2, we measured the influence on human perception of having different robots and sensors for accomplishing the same tasks. We observed that the robot's interface used for conveying affect impacted the outcomes of the experiment. For instance, the Pepper [235] robot lacking an interface for facial expressions, only allowing for LED color changes, reduces the interpretability of the affective signals in comparison to the robotic facial expressions expressed by the iCub robot. Moreover, acquiring head poses from a vision-based model or Inertial Measurement Unit (IMU) readings to orient the iCub robot's head, also affected the participants' impressions of the robot in terms of responsiveness.

These studies were used as a testbed to evaluate Wrapyfi in real-world settings. Additionally, we demonstrated that with Wrapyfi, sensors, robots, and deep-learning models can be easily exchanged according to experimental requirements. We note that the communication between the different components and the number of interdependent modules required for the operation of such studies demands fine calibration. Without Wrapyfi, conducting and implementing such studies would consume significantly more time and effort. Replacing sensors like cameras with IMUs, allowing an eye tracker with a ZeroMQ [113] API to communicate with a robot interface running on YARP [170], or readily distributing deep-learning models across multiple machines, demands careful engineering and major code duplication, which is what Wrapyfi was designed to simplify.

In Chapter 7, we utilized Wrapyfi to embody the models described within Chapter 3 and Chapter 4 in the iCub robot. These models targeted social attention tasks, all of which can be expressed through overt attention, namely gaze toward salient or prioritized regions. We chose the iCub robot as the platform of embodiment for its advanced gaze capabilities, expressed through humanlike head and eye movements.

In Section 7.1, we studied human attention responses under social audiovisual incongruence—incongruence occurs when gaze gestures oppose the speech sound source in direction, congruence occurs when both gaze and speech sound source directions match, whereas the neutral condition refers to nondirectional gaze (no head movements) and speech sound direction (monaural audio). Our GASP model was embedded into the iCub robot in tandem with our binaural audiovisual model based on the DAVE [245] model. This binaural model allowed for the localization of sound in audiovisual videos, a task closely related to saliency prediction, differing only in the training data. The latter receives binaural audio input as opposed to monaural audio, and its ground-truth maps indicate the location of sound rather than the attention maps of a group of observers. The robot was exposed to similar congruency conditions as humans by observing videos of three animated

humanlike avatars wearing masks. In these videos, one avatar looked straight ahead, to the left, or the right, while a short speech sound arrived from the location of either the left, right, or center avatar; under free-viewing on a monitor, the robot listened to binaural audio and gazed toward a target that was not explicitly defined. Humans significantly surpassed the robot in recognizing the direction of sound under all congruency conditions. However, the robot and humans exhibited similar localization trends under all three conditions. This finding indicates that our social attention models, although trained on human data for a different task (saliency prediction), and the participants in this experiment had to simply identify the direction of sound arrival, the patterns of the cognitively simulated eye movements on the robot closely matched the human goal-driven decision making patterns.

In Section 7.1, we simulated human responses on the iCub robot by directing its gaze toward the peak of the attention map predicted by the GASP model. However, the attention patterns of the individuals differ from those of a group of observers. Therefore, we adapted our GASP saliency prediction model to predict the scanpaths of individual observers instead. Following the adaptation, we proceeded to embed this model on the iCub robot in Section 7.2. In this study, we placed the robot in front of a monitor, playing videos from the two datasets on which our scanpath prediction model was trained, and let the robot attend toward the prioritized region for each observer, independently. We observe that the reduction in the robot's performance in comparison to the individual humans' performance was not as prominent as it was in Section 7.1. This indicates that accounting for differences in gaze patterns is critical, as it avoids convergence of gaze points toward the spatial mean point as was the case in Section 7.1, which is generally distributed around the center of the monitor (central bias [219]).

The approach followed in Section 7.2 requires a simulated environment that very closely resembles the physical environment. To evaluate saliency maps in the physical world, one could realize a pipeline that projects a priority map from the physical monitor, to the coordinates of the ground-truth map's plane. This is known as homography, a technique that has evolved from local image feature mapping [159] to deep learning-based view synthesis [156, 54]. Such methods might render more accurate transformations that reflect the current setting of an environment, rather than an approximated simulation. However, such an approach does not account for the robot's state and the camera's orientation as the robot actively views the environment. Therefore, we would additionally need to consider the camera's orientation in order to accurately reproject the predicted priority map. This would eventually result in an approach similar to ours, with the distinction being that our approach does not distort the predicted priority map through any transformations. In Section 7.2, we instead overlaid the predicted priority map upon the robot's view in simulation, then evaluated the match between the ground-truth—priority map displayed on a simulated monitor at a distance similar to that between the physical monitor and the robot—and predicted priority maps.

8.3 Relevance to Research Questions

RQ1.1 *Does integrating social cues, like gaze direction and facial expressions, with saliency models improve the models' performances?*

In Chapter 3, we created a social attention model that integrates social cues with audiovisual saliency representations, to eventually predict saliency. To investigate the relevance of each social cue to the task, we performed an ablation study in Section 3.5.3. We observed that the inclusion of all three cue representations—gaze estimation, gaze following, facial expression recognition—results in the highest performance as per all saliency metrics [43], excluding the AUCJ metric. The AUCJ metric is sensitive to false positives. Therefore, a model predicting fixation density maps with uniformly distributed magnitudes—interpreted as more fixation points predicted than the number of observers—would be scored lower in terms of AUCJ, which equates to worse performance. The gaze following [210] task is the most difficult cue detection task among all three. The model has to estimate the gaze direction, which could point outside the visual field. Following this estimation, it additionally detects the target of fixation, which could lie at any distance from the gazer. Moreover, it generates a probability density that indicates the likelihood that a region will be gazed upon. These limitations result in the model estimating noisy maps. Since our social attention model with the DAM included should weigh the cue modality contributions uniformly, a noisy representation will contribute to a noisy prediction and, thus, more false positives.

Our hypothesis concerning the noisiness of the gaze following representation is supported by Table 3.3, given that it scores lowest on all metrics, even compared to the model excluding all social cues. We also observed that the facial expression and gaze estimation representations independently lead to worse performance. However, the combination of the two leads to an improvement. We deduce that the quality of the representation plays a major role in valuing the contribution of each social cue. Moreover, the introduction of social cues, namely gaze and facial expressions, improves a saliency model's performance in the presence of social stimuli.

RQ1.2 *How can non-verbal social cues be integrated into social attention models?*

Feature engineering—extracting features from raw data that are known to be informative to a task, based on expert knowledge of that task—is the most prevalent approach to raw data processing in machine learning. However, as deep-learning models were developed further, feature engineering became less necessary as the tasks increased in complexity, and the former led to powerful feature extraction capabilities. Most modern deep-learning models trained for one or multiple tasks rely on backbones [290] or have their parameters initialized with those of other pretrained models. These backbones are usually

trained for simple tasks on large amounts of data with high variability. A latent layer in the backbone feeds its representations into a downstream model during training, allowing the model to extract features from the processed backbone representations rather than raw data. Backbones can significantly speed up the training of a model for a new task. However, once a model is trained, the backbone cannot be replaced without further fine-tuning.

Deep-learning models are rapidly improving. Enhancing existing downstream models by integrating them with the latest and best-performing backbones is costly. We therefore developed a solution in Chapter 3 that combines the strengths of feature engineering and deep-learning modeling. We used existing deep-learning models that detect and extract social cues. These are analogous to backbones, however, we did not extract their latent representations. Instead, we transformed their outputs into spatiotemporal maps as shown in Figure 3.2, which can be both interpretable and flexible, in that they can represent any output of a model that can perform a similar task. Such representations limit the features available to our models. However, we know from existing literature that humans rely on social cues when directing their attention in social settings. Equipped with this knowledge, we augmented the models with additional features that can improve their performance, as is commonly done when engineering features.

On representing social cues, we proceeded to address the problem of integrating those representations. We evaluated different integration and approaches on images (static) in Section 3.5.1 and sequences of frames (dynamic) in Section 3.5.2. In general, we found dynamic models to outperform static models given the additional context information, which is necessary for predicting saliency. We compared dynamic fusion and integration approaches and observed that integration, namely the LARGMU (described in Section 3.3.3) resulted in the best performance. The difference between integration and fusion lies in the order of gating and attention. For fusion models, gating precedes attention, whereas integration applies attention before gating. Fusion models, unlike integration models, maintain separable representations for each modality, allowing us to assess the independent contributions of the modalities to the task. We hypothesize that integration models outperform fusion models since the attention mechanism allows for the propagation of more distributed features, arriving from all modalities. On the other hand, fusion models first apply gating, allowing or disallowing certain modality representations through. Attention being more granular than gating, it can be seen as a filter that emphasizes the most relevant features propagated to the gate. When the gate precedes attention, some modality features are weighted to be more relevant, limiting the operation of attention to the weighted features only.

RQ1.3 *How can social attention models be personalized?*

Our main aim in this thesis was to develop models that can simulate human gaze on robots. In Section 7.1, we evaluated a saliency prediction model on a robot. This model, which we developed in Chapter 3, was trained on the attention maps of all observers watching social videos. We extracted the peaks of the attention maps predicted by the model and controlled the robot to gaze toward that peak. The peak represents the region to which most observers would likely look, upon perceiving the same stimuli as the robot. Our goal in Section 7.1 was to assess whether social attention models implicitly derive patterns that correspond to those of humans when presented with conflicting social and auditory cues. This experiment was aimed at assessing the performance of our social attention model in physical settings, and whether it infers attention cues from the visual and auditory stimuli as humans would. However, this approach was not designed to, nor would it accurately simulate human gaze patterns.

Gaze patterns are sequential, meaning that previous fixations affect the ones to follow. Consequently, accurate gaze modeling inherently implies that a scanpath must be modeled. Therefore, in Chapter 4, we developed a scanpath prediction model that extends our saliency prediction model, developed in Chapter 3. This extension involved the integration of a fixation history module, that retains the previous fixations of an observer. The fixation history serves two purposes, one being the retention of previous fixations to predict the ones to follow and another being the specification of the gaze pattern to be followed. The latter is especially important for our approach. In Chapter 4, we developed a single unified scanpath prediction model, and compared it to individual models, each trained exclusively on the scanpaths of a single observer. The individual models served as baseline models, following the reasoning that the best prediction of scanpaths that our model can achieve is made so by training the model on one task only, that is the prediction of a single observer's gaze patterns. However, this does not take into account that universal attention, or the attention traits that are common among all observers, is not sufficiently represented in an individual model, due to the limited variability in its training samples. This was confirmed in Section 4.5.2, where we showed that the unified model significantly outperformed individual models in terms of the AUCJ and NSS metric scores. By integrating the fixation history, we were able to transform our saliency model into a scanpath prediction model. Moreover, the fixation history acts as a prompt that personalizes the unified model, by enabling us to choose the gaze pattern that we would want the model to mimic. Without the fixation history module, our unified model cannot predict scanpaths, since it has no other prior that informs it on how to personalize the gaze predictions.

RQ1.4 *Which methods are needed to embody social attention models in robots?*

In this thesis, we conducted four studies on physical robots, each comprising multiple tasks and experiments. The heterogeneity in the sensors and robots used in these experiments, their data types, their communication packages and middleware, their APIs, and their sampling rates, made it necessary to have a software package that can handle and abstract the complexity resulting from these differences. As a solution to fulfill all these requirements, we developed the Wrapyfi framework in Chapter 5 that wraps all the middleware required for exchanges within a single API, allowing us to switch middleware without having to rewrite our scripts. Moreover, most of our studies relied upon several deep-learning models, all of which had to communicate data to the robots, acquire readings from their sensors, and even exchange tensors or signals with other deep-learning models. Our framework was designed to provide plugins for many popular and niche Python frameworks, whether they were developed for deep-learning-based applications, image and audio processing, or numerical analysis. We also introduced three communication schemes with this framework, by which scripts and models can be easily distributed across different machines according to their capabilities and supported libraries. Since our social attention models relied on several social cue detectors, each a deep-learning model on its own, distributing them was a necessity. Some machines were dedicated to sensory data acquisition at high sampling rates, others were dedicated to the running of one or several models, while a few were managing the experiment pipelines and scheduling communication between other machines.

The main purpose of our framework was to run our models on robots, without tying a model's implementation to a certain robot or middleware. We provided examples as part of our open-source repository, such as scripts enabling the control and sensor acquisition from specific robots. Thus, demonstrating how our framework could also lead to standardizing interfaces. By standardizing the conventions and interface formats, we were able to easily replace social cue detectors and sensors depending on the study, without rewriting or modifying our robot interfaces.

RQ1.5 *How can we assess the performance of a physical robotic gaze implementation?*

Social robotic tasks are usually evaluated by conducting Human-Robot Interaction (HRI) studies, by which the responses of participants are analyzed. These responses are either provided directly by the participants in the form of questionnaires or inferred from their behaviors. We conducted two HRI studies in Chapter 6 on human responses to robots displaying social cues. Our studies combined the two forms of analyses, where we asked participants to fill out questionnaires in Section 6.1 and Section 6.2, and measured their task completion times in Section 6.1. The questionnaire-based and human-inference-based approaches are the conventional forms of evaluation in social

scenarios since measuring a robot's performance based on a 'humanlikeness metric' is not practically realizable. However, conducting HRI studies requires a number of participants to perform a task that is time-consuming, subject to confounding and bias effects, prone to failed or illegible trials, difficult or impossible to scale, and not easily reproducible or replicable.

To mitigate these limitations, we performed what we termed as cognitive robotic simulation in Chapter 7. This allowed us to test our models in physical environments to observe their robustness to sensor noise and feasibility as the cameras of our robot were active—active vision refers to cameras that are attached to actuators and can move within their environments as they apply a task that allows them to explore those environments. Our first study in Section 7.1 addressed a relatively simple cognitive simulation, where we compared a robot running the models developed in Chapter 3 and Section 7.1) to humans selecting the direction of sound arrival as it aligns (congruent) or conflicts (incongruent) with the direction of a visual (non-verbal) social cue. The model predicted an attention map, from which the peak was extracted and the robot was actuated to look toward. The human participants and the robot watched the same videos under the same conditions and with the same set of aligning or conflicting stimuli. After which, we compared the direction chosen by the participants, as key presses indicating sound arrived from the right or left, to the robot gazing upon the monitor, and assuming it is looking toward the left or right side of the monitor in correspondence with humans pressing the directional keys. We measured the success and failures of the robot and the humans in locating sound under congruent and incongruent conditions in Section 7.1.8. We found the robot and humans to locate sound more accurately when the social and auditory cues were congruent, compared to the incongruent and neutral—auditory cue without a gaze direction as a social cue—conditions.

Our second study conducted in Section 7.2 addressed a more complex evaluation scheme. We designed a physical setup similar to that in Section 7.1, with different stimuli and evaluation procedures. The stimuli consisted of social videos viewed by participants whose gaze data was collected while watching the videos. In this setup, we embodied a scanpath prediction model developed in Chapter 4 on the robot. Instead of comparing the accuracy of the robot and humans in making binary decisions as was the case in Section 7.1, we measured the match between the priority maps predicted by the model when streaming the videos and the videos acquired by letting the robot watch the videos on a monitor and acquiring the stimuli from its camera and microphones. We were able to utilize the AUCJ and NSS metrics for this comparison by projecting the ground-truth priority maps to a simulated monitor, such that its distance and angle from the robot matched the physical setup. Next, we compared the reprojected map in simulation to the predicted map, based on the physical stimuli. We then measured the model's performance between streamed video and physical robot video

input. In Section 7.2, we showed that the robot displayed patterns similar to the streamed input in terms of the NSS and AUCJ metric scores. This indicated that although distractors and low-fidelity sensor data from the physical environment degraded the model's performance, given that scores followed similar trends for each observer, the model was inferring patterns that aligned with those on streaming the input. This finding verified that our scanpath model is robust to noise and was able to attend to stimuli correctly.

8.4 Limitations and Future Work

Unlike virtual and augmented reality models of gaze that receive a full 360° view of the simulated or real environment [216], our models were developed to operate on salient stimuli within the perceivable visual field. However, in real-world scenarios, attractors could lie beyond that field. For instance, an alerting sound occurring behind the perceiver would trigger a reaction, such as a head rotation toward the alert. Peyrache *et al.* [201] show that the neural activity of the head direction neurons in animals conveys true spatial information, which, along with egocentric information, can be translated to spatial code. Following this finding, we will extend our models to translate their multimodal integration representations to a *universal map* that is encoded relative to the environment rather than being limited to the perceivable visual field of the observer. This would require gaze data that is not only accompanied by binaural audiovisual data but also includes videos with a sufficiently wide angle of the visual-input view, such that stimuli beyond the local visual field are visible. A wide-angle view is necessary so that we would be able to incorporate elements that are outside the perceivable visual field into our model predictions. This approach will allow us to cognitively simulate the way humans and animals navigate and interact with their environments, taking into account not only what is directly visible but also what can be inferred or anticipated from auditory cues and wide peripheral vision. Embodying such a model in a robot might also play a role in influencing the perception of its personality [181]. In preparation for this task, we provide an overview of the data collection and HRI evaluation procedures [92].

Another limitation arises with our choice of learning paradigm. Our scanpath models were trained in a supervised fashion, which consequently resulted in them yielding deterministic gaze predictions. This simplifies the evaluation process since the predicted priority or attention map similarity can be measured against the ground-truth maps. However, even though the scanpaths of an individual are *idiosyncratic* on repeated views of stimuli, they are still unique [84]. Adapting our approaches to integrate variability into the learning process, for instance, through Inverse Reinforcement Learning (IRL) could result in more realistic scanpath prediction. Yang *et al.* [281] propose a static scanpath prediction model that learns goal-directed scanpaths through IRL using Generative Adversarial Imitation Learning (GAIL) [114]. The model extracts semantic segmentation features from the input image, along with an encoding of the goal. It is then trained for visual

search, with the goal defined by the target object in the scene. The GAIL model generates scanpaths while its discriminator must distinguish generated scanpaths from those of humans tasked with finding specific objects in an image. Another approach by Chen *et al.* [53] addresses scanpath prediction in a Reinforcement Learning (RL)-based visual question-answering framework, predicting the fixations of human observers for attending to certain regions in an image based on a proposed question. The authors introduce a mechanism that maximizes the distinction between scanpaths resulting from different questions, as well as a loss function that measures the inconsistency between training and inference phase contexts. Moreover, the loss function is also designed to maximize the ScanMatch [68] score of the predicted trajectories. We note that Yang *et al.* [281] and Chen *et al.* [53] address goal-directed scanpath prediction in images only. However, both could be adapted to benefit from our social attention model architectures by replacing their visual encoders with our sequentially encoded cue and saliency representations, their attention modules with our sequential integration modules, and finally, their task embeddings with the fixation history representations.

Nonetheless, in typical IRL and RL settings, the reward is either learned by the model or shaped specifically for achieving a predefined objective, respectively. Although these approaches are practical in structured environments with minimal noise, adapting them to our models might result in suboptimal performance when embodying the models in robots. This is due to noisy signal readings that distort the model's ability to learn accurate behaviors and patterns in physical environments. As a remedy to noisy reward signals, Li *et al.* [152] propose to separate the reward model into external and internal components. The external reward is acquired from the environment and may be susceptible to noise, while the internal reward is generated by an internal model that reflects the RL agent's intrinsic motivation. This allows the reward and policy models to be learned separately, mitigating the model's performance degradation due to noise. By integrating this paradigm into our future IRL- or RL-based social attention models, we can further improve their robustness in real-world settings.

Appendix A

Additional Tables and Figures

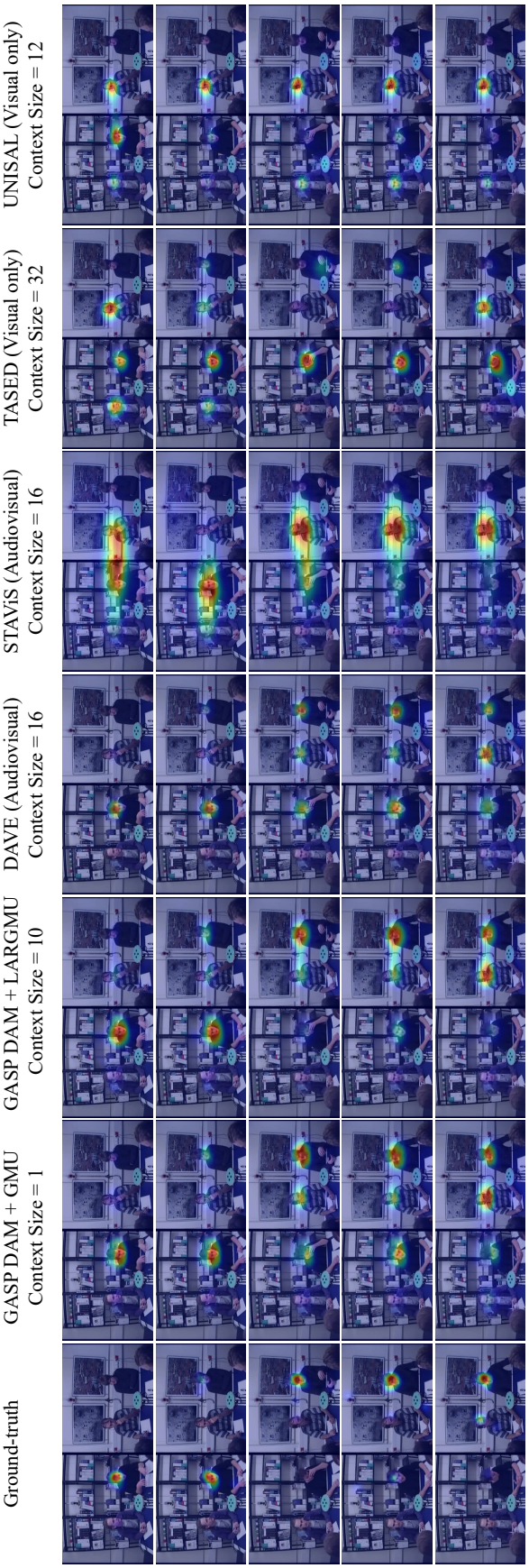


Figure A.1: Frame predictions on the Coutrot Database 2 [66] for comparison with state-of-the-art models. Our GASP models employ the audiovisual DAVE model (Context Size = 16) as the saliency predictor in the SCD stage.

Table A.1: Best model results for all context sizes with the standard deviation for each metric score provided, along with the training duration and number of parameters. All models employ the audiovisual DAVE model (Context Size = 16) as the saliency predictor in the SCD stage.

Model Architecture	AUCJ ↑	sAUC ↑	CC ↑	NSS ↑	SIM ↑	**No. Parameters**	**Training Time**	**Context Size**
DAM + GMU	0.8845 ± 0.1389	0.6397 ± 0.0831	0.6620 ± 0.2324	2.77 ± 0.53	0.5233 ± 0.1628	0.77M	25 mins	1
DAM + LARGMU	0.8789 ± 0.0420	0.6437 ± 0.0742	0.6703 ± 0.1068	2.78 ± 0.32	0.5354 ± 0.0675	4.28M	185 mins	2
LARGMU	0.8860 ± 0.0396	0.6460 ± 0.0778	0.6698 ± 0.1093	2.80 ± 0.32	0.5191 ± 0.0634	4.28M	140 mins	4
LARGMU	0.8818 ± 0.0400	0.6433 ± 0.0732	0.6556 ± 0.1027	2.76 ± 0.30	0.5205 ± 0.0607	4.28M	130 mins	6
DAM + LARGMU	0.8872 ± 0.0374	0.6529 ± 0.0703	0.6903 ± 0.1167	2.84 ± 0.31	0.5520 ± 0.0729	4.28M	280 mins	8
DAM + LARGMU	0.8830 ± 0.0451	0.6527 ± 0.0785	0.6980 ± 0.1164	2.87 ± 0.34	0.5566 ± 0.0753	4.28M	315 mins	10
DAM + LARGMU	0.8775 ± 0.0430	0.6418 ± 0.0747	0.6612 ± 0.1115	2.74 ± 0.32	0.5328 ± 0.0716	4.28M	330 mins	12

Table A.2: Multi-step-ahead predictions for each observer using the unified integration model fine-tuned on the small FindWho [272] dataset. All models are based on the DAM + LARGMU ($T' = 10$) GASP [6] variant with the additional fixation history module. Multi-step-ahead experiments conducted only on streamed input and one-step-ahead on the robot-acquired input.

ID	t'		$t'+1$		$t'+2$		$t'+3$		$t'+4$		Robot (t')	
	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑
1	0.970	1.898	0.972	1.744	0.817	1.512	0.683	1.241	0.661	0.967	0.955	1.823
2	0.981	1.695	0.966	1.495	0.898	1.450	0.789	1.383	0.765	0.704	0.964	1.473
3	0.980	2.082	0.969	1.844	0.841	1.619	0.723	1.332	0.709	1.007	0.962	1.801
4	0.970	1.847	0.961	1.780	0.869	1.351	0.790	1.203	0.774	0.805	0.955	1.753
5	0.973	1.612	0.955	1.398	0.832	1.206	0.782	0.987	0.758	0.691	0.960	1.684
6	0.947	1.930	0.948	1.568	0.814	1.500	0.722	1.242	0.696	0.890	0.924	1.473
7	0.981	2.557	0.967	1.618	0.901	1.471	0.851	1.266	0.822	0.872	0.964	1.773
8	0.979	2.419	0.976	2.032	0.878	1.933	0.813	1.672	0.798	1.187	0.961	1.747
9	0.981	1.920	0.960	1.645	0.811	1.485	0.742	1.205	0.715	1.010	0.962	1.281
10	0.974	1.641	0.961	1.247	0.870	1.181	0.783	0.993	0.754	0.727	0.953	1.268
11	0.984	1.816	0.949	1.514	0.834	1.341	0.732	1.115	0.715	0.789	0.973	1.444
12	0.986	2.105	0.980	1.967	0.857	1.823	0.807	1.245	0.777	0.902	0.963	1.734
13	0.967	2.118	0.965	1.807	0.907	1.625	0.819	1.136	0.795	1.154	0.959	1.659
14	0.965	1.753	0.962	1.835	0.871	1.635	0.827	1.337	0.804	0.878	0.953	1.160
15	0.975	2.003	0.977	1.925	0.880	1.692	0.830	1.306	0.797	0.991	0.958	1.651
16	0.980	2.190	0.967	1.963	0.911	1.829	0.826	1.526	0.796	1.131	0.958	1.324
17	0.962	2.169	0.956	1.922	0.847	1.663	0.746	1.397	0.722	0.989	0.954	1.733
18	0.976	2.119	0.973	1.968	0.854	1.667	0.779	1.010	0.754	0.962	0.955	1.654
19	0.980	2.122	0.951	1.759	0.839	1.586	0.749	0.925	0.723	0.981	0.955	1.955
20	0.978	1.955	0.973	1.896	0.922	1.816	0.811	1.527	0.791	1.065	0.957	1.381
21	0.965	1.946	0.960	1.843	0.854	1.639	0.772	1.533	0.709	0.810	0.947	1.595
22	0.982	2.035	0.974	1.973	0.851	1.813	0.779	1.466	0.740	1.129	0.963	1.988
23	0.986	2.318	0.972	2.082	0.910	1.958	0.826	1.457	0.830	1.040	0.970	2.126
24	0.977	1.820	0.968	1.678	0.895	1.233	0.809	1.199	0.760	0.910	0.956	1.783
25	0.975	2.133	0.972	1.998	0.890	1.820	0.782	1.529	0.779	1.186	0.963	1.853
26	0.976	2.113	0.961	1.879	0.878	1.872	0.803	1.363	0.787	0.863	0.939	1.844
27	0.988	2.234	0.985	1.844	0.923	1.766	0.833	1.460	0.808	1.310	0.925	1.985
28	0.977	2.038	0.952	1.875	0.826	1.771	0.769	1.335	0.751	1.011	0.959	1.265
29	0.974	2.511	0.961	2.267	0.858	2.133	0.763	1.833	0.747	0.995	0.969	2.047
30	0.970	1.525	0.969	1.358	0.829	0.981	0.782	0.820	0.755	0.614	0.951	1.375
31	0.983	2.486	0.981	2.337	0.839	2.192	0.796	1.850	0.768	1.348	0.970	1.970
32	0.981	2.243	0.978	1.975	0.846	1.510	0.771	1.453	0.764	0.930	0.961	1.624
33	0.977	1.905	0.969	1.777	0.892	1.560	0.842	1.414	0.816	0.993	0.963	1.754
34	0.978	1.936	0.971	1.456	0.884	1.316	0.818	1.289	0.788	1.122	0.963	1.473
35	0.979	2.061	0.964	1.730	0.826	1.576	0.705	1.266	0.681	0.878	0.961	2.037
36	0.989	2.698	0.984	2.036	0.915	1.927	0.859	1.550	0.827	1.196	0.895	1.563
37	0.966	1.782	0.965	1.205	0.874	1.116	0.781	1.062	0.750	0.779	0.948	1.750
38	0.973	1.517	0.972	1.504	0.847	1.319	0.761	1.315	0.760	0.782	0.948	1.356
39	0.981	2.140	0.971	2.017	0.895	1.616	0.853	1.525	0.801	1.176	0.965	1.724

Table A.3: Multi-step-ahead predictions for each observer using the unified integration model fine-tuned on the small MVVA [158] dataset. All models are based on the DAM + LARGMU ($T' = 10$) GASP [6] variant with the additional fixation history module. Multi-step-ahead experiments conducted only on streamed input and one-step-ahead on the robot-acquired input.

ID	t'		$t'+1$		$t'+2$		$t'+3$		$t'+4$		**Robot** (t')	
	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑
1	0.974	1.558	0.972	1.551	0.938	1.480	0.904	1.458	0.893	1.309	0.920	1.427
2	0.969	1.339	0.962	1.339	0.935	1.173	0.900	1.149	0.888	1.028	0.930	1.112
3	0.949	1.075	0.941	1.058	0.919	1.005	0.892	0.991	0.879	0.912	0.895	0.775
4	0.959	1.145	0.946	1.163	0.921	1.074	0.889	1.070	0.874	1.006	0.909	0.926
5	0.976	1.365	0.972	1.330	0.955	1.251	0.924	1.258	0.913	1.141	0.968	1.404
6	0.974	1.683	0.960	1.677	0.930	1.585	0.903	1.587	0.888	1.436	0.925	1.525
8	0.969	1.357	0.958	1.346	0.936	1.293	0.903	1.295	0.893	1.142	0.937	0.934
9	0.961	1.224	0.954	1.196	0.929	1.150	0.899	1.133	0.883	1.080	0.941	1.043
10	0.947	1.504	0.930	1.481	0.907	1.470	0.875	1.406	0.863	1.297	0.908	1.214
11	0.957	1.270	0.948	1.247	0.927	1.180	0.892	1.183	0.881	1.132	0.898	1.141
12	0.965	1.243	0.948	1.233	0.917	1.194	0.886	1.169	0.874	1.068	0.926	0.951
13	0.963	1.239	0.953	1.184	0.922	1.179	0.892	1.171	0.882	1.113	0.926	0.908
14	0.964	1.267	0.963	1.245	0.936	1.198	0.903	1.183	0.894	1.117	0.936	1.171
15	0.975	1.433	0.970	1.421	0.945	1.386	0.914	1.385	0.897	1.159	0.931	1.215
16	0.937	0.956	0.932	0.950	0.914	0.905	0.882	0.899	0.869	0.821	0.924	0.725
17	0.961	1.237	0.958	1.219	0.926	1.163	0.896	1.149	0.879	0.974	0.918	0.827
18	0.961	1.187	0.950	1.210	0.923	1.127	0.888	1.123	0.873	1.081	0.909	0.921
19	0.937	1.227	0.928	1.209	0.897	1.184	0.868	1.249	0.852	1.056	0.908	0.783
20	0.960	1.337	0.947	1.308	0.922	1.265	0.895	1.248	0.884	1.371	0.947	0.947
21	0.962	1.251	0.951	1.261	0.932	1.176	0.896	1.161	0.885	1.121	0.944	0.949
22	0.963	1.193	0.956	1.156	0.929	1.124	0.897	1.113	0.884	1.113	0.915	0.739
23	0.937	0.972	0.926	0.966	0.906	0.924	0.875	0.920	0.865	0.866	0.925	0.922
24	0.971	1.555	0.954	1.539	0.923	1.479	0.892	1.414	0.882	1.276	0.938	1.291
25	0.970	1.403	0.968	1.396	0.942	1.331	0.906	1.129	0.889	1.087	0.932	1.377
26	0.982	1.540	0.973	1.538	0.952	1.473	0.920	1.456	0.910	1.371	0.930	1.154
27	0.964	1.222	0.947	1.174	0.931	1.200	0.901	1.166	0.890	1.110	0.920	1.086
28	0.943	1.192	0.940	1.173	0.914	1.135	0.886	1.119	0.869	1.087	0.915	1.159
29	0.949	1.109	0.936	1.107	0.911	1.062	0.883	1.044	0.869	1.003	0.936	0.832
30	0.958	1.199	0.952	1.192	0.925	1.148	0.890	1.120	0.875	1.040	0.903	0.942
31	0.965	1.359	0.948	1.351	0.916	1.296	0.888	1.294	0.878	1.189	0.918	1.180
32	0.977	1.561	0.964	1.552	0.934	1.491	0.903	1.475	0.889	1.411	0.939	1.109
33	0.959	1.105	0.957	1.098	0.933	1.048	0.900	1.058	0.885	1.008	0.938	1.082
34	0.936	1.042	0.933	1.031	0.908	0.981	0.875	0.978	0.864	0.954	0.902	0.887

Table A.4: Multi-step-ahead predictions using unified models fine-tuned on the small FindWho [272] and large MVVA [158] datasets. All models are based on variants of GASP [6] with the additional fixation history modules. The context size T' for each model is shown in parentheses. **Bold** denotes the best scores.

Model Architecture	**Dataset**	t'		$t'+1$		$t'+2$		$t'+3$		$t'+4$	
		AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑	AUCJ↑	NSS↑
Fusion: DAM + ARGMU ($T'=8$)	FindWho	0.969	1.252	0.941	1.190	0.853	1.074	0.703	0.893	0.693	0.657
Integration: DAM + LARGMU ($T'=10$)	FindWho	**0.976**	**2.035**	**0.967**	**1.788**	0.866	**1.603**	0.787	**1.327**	0.763	0.969
Fusion: DAM + ARGMU ($T'=8$)	MVVA	0.952	1.352	0.931	1.305	0.893	1.206	0.843	1.176	0.827	1.049
Integration: DAM + LARGMU ($T'=10$)	MVVA	0.960	1.283	0.951	1.271	**0.926**	1.214	**0.894**	1.202	**0.881**	**1.116**

Appendix B

Resulting Publications

B.1 Publications and Workshop Articles Associated with this Dissertation

Parts of the following manuscripts were included in this thesis[1]:

[6] **© 2021 IJCAI organization http://www.ijcai.org. Reprinted,** Fares Abawi, Tom Weber, and Stefan Wermter. "GASP: Gated Attention for Saliency Prediction". In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI Organization, 2021, pp. 584–591. DOI: 10.24963/ijcai.2021/81

[88] **Creative Commons Attribution International 4.0 License.** Di Fu, Fares Abawi, Hugo Carneiro, Matthias Kerzel, Ziwei Chen, Erik Strahl, Xun Liu, and Stefan Wermter. "A Trained Humanoid Robot can Perform Human-Like Crossmodal Social Attention and Conflict Resolution". In: *International Journal of Social Robotics* 15 (2023), pp. 1325–1340. DOI: 10.1007/s12369-023-00993-3

[89] **© 2023 IEEE. Reprinted, with permission,** Di Fu, Fares Abawi, and Stefan Wermter. "The Robot in the Room: Influence of Robot Facial Expressions and Gaze on Human-Human-Robot Collaboration". In: *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2023, pp. 85–91. DOI: 10.1109/RO-MAN57019.2023.10309334

[4] **Creative Commons Attribution International 4.0 License.** Fares Abawi, Philipp Allgeuer, Di Fu, and Stefan Wermter. "Wrapyfi: A Python Wrapper for Integrating Robots, Sensors, and Applications Across Multiple Middleware". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. ACM, 2024, pp. 860–864. DOI: 10.1145/3610977.3637471. URL: https://wrapyfi.readthedocs.io

[1] Grammarly (https://grammarly.com) and Writefull (https://writefull.com) were used to assist with grammar and language editing of the manuscripts adapted to this thesis

[91] **Creative Commons Attribution International 4.0 License.** Di Fu*, Fares Abawi*, Philipp Allgeuer, and Stefan Wermter. "Human Impression of Humanoid Robots Mirroring Social Cues". In: *Companion of the ACM/IEEE International Conference on Human-Robot Interaction (HRI Companion)*. ACM, 2024, pp. 458–462. DOI: 10.1145/3610978.3640580

[5] Fares Abawi, Di Fu, and Stefan Wermter. "Unified Dynamic Scanpath Predictors Outperform Individually Trained Neural Models". In: *(Under Review)* abs/2405.02929 (2024). DOI: 10.48550/arXiv.2405.02929. arXiv: 2405.02929

[7] Fares Abawi and Stefan Wermter. "HRI-Free Evaluation of Embodied Social Attention Models through Cognitive Robotic Simulation". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) Workshop on Interactive AI for Human-Centered Robotics*. 2024

B.2 Other Related Publications and Workshop Articles

[181] Hadi Beik Mohammadi, Nikoletta Xirakia, Fares Abawi, Irina Barykina, Krishnan Chandran, Gitanjali Nair, Cuong Nguyen, Daniel Speck, Tayfun Alpay, Sascha Griffiths, Stefan Heinrich, Erik Strahl, Cornelius Weber, and Stefan Wermter. "Designing a Personality-Driven Robot for a Human-Robot Interaction Scenario". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4317–4324. DOI: 10.1109/ICRA.2019.8793770

[132] Matthias Kerzel, Manfred Eppe, Stefan Heinrich, Fares Abawi, and Stefan Wermter. "Neurocognitive Shared Visuomotor Network for End-To-End Learning of Object Identification, Localization and Grasping on a Humanoid". In: *Proceedings of the IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL-Epirob)*. IEEE. 2019, pp. 19–24. DOI: 10.1109/DEVLRN.2019.8850679

[135] Matthias Kerzel*, Fares Abawi*, Manfred Eppe, and Stefan Wermter. "Enhancing a Neurocognitive Shared Visuomotor Model for Object Identification, Localization, and Grasping with Learning from Auxiliary Tasks". In: *IEEE Transactions on Cognitive and Developmental Systems* 14.4 (2022), pp. 1331–1343. DOI: 10.1109/TCDS.2020.3028460

[92] Di Fu*, Fares Abawi*, Erik Strahl, and Stefan Wermter. "Judging by the Look: The Impact of Robot Gaze Strategies on Human Cooperation". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) Workshop on Machine Learning for HRI: Bridge the Gap between Action and Perception*. 2022. DOI: 10.48550/arXiv.2208.11647

∗ *indicates equal contribution*

Appendix C

Resources: Videos and Code

[6] In Chapter 3 we present the GASP saliency prediction model.
The associated model parameters and code (training and inference) are available at:
http://software.knowledge-technology.info#gasp.
Video presentation:
https://www.youtube.com/watch?v=e4HFTmEgirk.

[5] In Chapter 4 we develop a unified scanpath prediction model integrating fixation history to enable scanpath personalization.
The associated model parameters and code (training and inference) are available at:
http://software.knowledge-technology.info#gasp.

[4] In Chapter 5 we present a Python framework for integrating message-oriented and robotics middleware.
The associated code is available at:
http://software.knowledge-technology.info#wrapyfi.
Documentation:
https://wrapyfi.readthedocs.io.
Video demos:
https://raw.githubusercontent.com/fabawi/wrapyfi/main/assets/tutorials/vid_demo_ex2-1.mp4,
https://raw.githubusercontent.com/fabawi/wrapyfi/main/assets/tutorials/vid_demo_ex1-1.mp4.

[89] In Chapter 6 (Section 6.1) we study the influence of robot social cues on human-human-robot collaboration.
Video presentation:
https://fares.abawi.me/assets/video/robotintheroom_video.mp4.

[91] In Chapter 6 (Section 6.2) we study the difference in perception on displaying affective signals on the Pepper and iCub robots. We also study the influence of using different sensory modalities to control the same robot.
Video presentation:
https://www.youtube.com/watch?v=Qn0xo4JyG9c.

[88] In Chapter 7 (Section 7.1) we present our audiovisual sound localization model and we integrate it with the GASP model to deploy it on the iCub robot for studying the correlation between audiovisual congruency in humans and the robot.
Video presentation:
https://www.youtube.com/watch?v=bjiYEs1x-7E.

[7] In Chapter 7 (Section 7.2) we embody this model on the iCub robot to evaluate its robustness to noisy real-world conditions, simulating the environment to which the human participants were exposed.
Video demo:
https://fares.abawi.me/assets/video/usp_eval_video.mp4

Appendix D

Acknowledgements

I would like to thank all who have contributed to the successful completion of my doctoral studies. First and foremost, I want to express my deepest gratitude to Prof. Stefan Wermter for his guidance and support. His expertise, insightful feedback, and innovative ideas have significantly shaped my research. Moreover, his enthusiasm and dedication have inspired me to push the boundaries of my work. I would also like to sincerely thank Prof. Frank Steinicke and Prof. Timo Gerkmann, for investing the time and effort to review my thesis.

I am very grateful to Dr. Cornelius Weber for his mentorship throughout my studies. His constructive feedback has been immeasurable to my research. Thank you as well for taking the time to carefully review my thesis and for your contributions that greatly improved its quality. Your insights and willingness to help have been inspirational, and I am deeply appreciative of everything you have done for me so far. I also greatly appreciate Dr. Antonio Andriella and Dr. Di Fu for reviewing my thesis and providing very helpful feedback.

Thank you to all the CML project and the Knowledge Technology group members for creating a collaborative and intellectually stimulating environment that helped to form my academic identity and improve my research. A special thanks to Erik Strahl for always being there to promptly resolve every technical issue I encountered. Your expertise made my experience much smoother and less stressful. To Dr. Matthias Kerzel, thank you for your guidance, collaboration, and the discussions that inspired many of my ideas. Thank you Katja Kösters for your exceptional editorial aid and assistance with various matters. Your thoughtfulness and constant willingness to help have left a lasting impact. To all my collaborators, thank you for your phenomenal work and contributions.

I wish to express my heartfelt appreciation to Di, whose influence on my research and support have been invaluable during my studies. Your encouragement and friendship have enriched my life in many ways and I am truly grateful for the countless hours you've dedicated to help me succeed. To my parents, Mirvat and Ebrahim, your love, guidance, and belief in me have driven me forward. Your sacrifices inspired and motivated me every step of the way. To my sisters, Deena and Maya, your support and understanding have been a source of strength and joy. I could not have reached this milestone without you.

Bibliography

[1] Sathyanarayanan N. Aakur. and Arunkumar Bagavathi. "Unsupervised Gaze Prediction in Egocentric Videos by Energy-Based Surprise Modeling". In: *Proceedings of the International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP)*. SciTePress, 2021, pp. 935–942. DOI: 10.5220/0010288009350942.

[2] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2015. DOI: 10.5281/zenodo.4724125. URL: https://www.tensorflow.org.

[3] Fares Abawi. *Intermediate Representations in Deep Multimodal Neural Networks*. https://fares.abawi.me/assets/pdf/Fares_Abawi_MSc_Thesis.pdf. Universität Hamburg Fachbereich Informatik, 2019.

[4] Fares Abawi, Philipp Allgeuer, Di Fu, and Stefan Wermter. "Wrapyfi: A Python Wrapper for Integrating Robots, Sensors, and Applications Across Multiple Middleware". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. ACM, 2024, pp. 860–864. DOI: 10.1145/3610977.3637471. URL: https://wrapyfi.readthedocs.io.

[5] Fares Abawi, Di Fu, and Stefan Wermter. "Unified Dynamic Scanpath Predictors Outperform Individually Trained Neural Models". In: *(Under Review)* abs/2405.02929 (2024). DOI: 10.48550/arXiv.2405.02929. arXiv: 2405.02929.

[6] Fares Abawi, Tom Weber, and Stefan Wermter. "GASP: Gated Attention for Saliency Prediction". In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI Organization, 2021, pp. 584–591. DOI: 10.24963/ijcai.2021/81.

[7] Fares Abawi and Stefan Wermter. "HRI-Free Evaluation of Embodied Social Attention Models through Cognitive Robotic Simulation". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) Workshop on Interactive AI for Human-Centered Robotics*. 2024.

[8] ABi, ed. *Open-Source Robotics Projects*. ABi Research, 2019. URL: https://www.abiresearch.com/market-research/product/1029218-open-source-robotics-projects/.

[9] Henny Admoni and Brian Scassellati. "Social Eye Gaze in Human-Robot Interaction: A Review". In: *Journal of Human-Robot Interaction* 6.1 (2017), pp. 25–63. DOI: 10.5898/JHRI.6.1.Admoni.

[10] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E. Hinton. "Neural Additive Models: Interpretable Machine Learning with Neural Nets". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 34. https://api.semanticscholar.org/CorpusID:216641712. Curran Associates, Inc., 2021, pp. 4699–4711.

[11] Samer Al Moubayed, Jonas Beskow, Gabriel Skantze, and Björn Granström. "Furhat: A Back-Projected Human-Like Robot Head for Multiparty Human-Machine Interaction". In: *Proceedings of the Conference on Cognitive Behavioural Systems (COST)*. Springer, 2012, pp. 114–130. DOI: 10.1007/978-3-642-34584-5_9.

[12] David Alais and David Burr. "The Ventriloquist Effect Results from Near-Optimal Bimodal Integration". In: *Current Biology* 14.3 (2004), pp. 257–262. DOI: 10.1016/j.cub.2004.01.029.

[13] Robert G. Alexander and Susana Martinez-Conde. "Fixational Eye Movements". In: *Eye Movement Research: An Introduction to its Scientific Foundations and Applications*. Springer, 2019, pp. 73–115. DOI: 10.1007/978-3-030-20085-5_3.

[14] Tayfun Alpay, Fares Abawi, and Stefan Wermter. "Preserving Activations in Recurrent Neural Networks Based on Surprisal". In: *Neurocomputing* 342 (2019), pp. 75–82. DOI: 10.1016/j.neucom.2018.11.092.

[15] Antonio Andriella, Henrique Siqueira, Di Fu, Sven Magg, Pablo Barros, Stefan Wermter, Carme Torras, and Guillem Alenya. "Do I Have a Personality? Endowing Care Robots with Context-Dependent Personality Traits". In: *International Journal of Social Robotics* 13 (2020), pp. 2081–2102. DOI: 10.1007/s12369-020-00690-5.

[16] Motonobu Aoki, Karthikeyan Kalyanasundaram Balasubramanian, Diego Torazza, Francesco Rea, Doreen Jirak, Giulio Sandini, Takura Yanagi, Atsushi Takamatsu, Stephane Bouet, and Tomohiro Yamamura. "A Novel Wire-driven 3D Eyebrow Design for Communication with Humanoid Robot iCub". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2022, pp. 8248–8254. DOI: 10.1109/IROS47612.2022.9981954.

[17] John Arevalo, Thamar Solorio, Manuel Montes-y-Gómez, and Fabio A González. "Gated Multimodal Networks". In: *Neural Computing and Applications* 32 (2020), pp. 10209–10228. DOI: 10.1007/s00521-019-04559-1.

[18] Franziska Babel, Johannes Kraus, Linda Miller, Matthias Kraus, Nicolas Wagner, Wolfgang Minker, and Martin Baumann. "Small Talk with a Robot? The Impact of Dialog Content, Talk Initiative, and Gaze Behavior of a Social Robot on Trust, Acceptance, and Proximity". In: *International Journal of Social Robotics* (2021), pp. 1–14.

[19] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. arxiv.org, 2015. DOI: 10.48550/arXiv.1409.0473.

[20] Junjie Bai, Fang Lu, Ke Zhang, et al. *ONNX: Open Neural Network Exchange*. Version 1.14.1. 2019. URL: https://onnx.ai.

[21] Cagdas Bak, Aysun Kocak, Erkut Erdem, and Aykut Erdem. "Spatio-Temporal Saliency Networks for Dynamic Saliency Prediction". In: *IEEE Transactions on Multimedia* 20.7 (2017), pp. 1688–1698. DOI: 10.1109/TMM.2017.2777665.

[22] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. "Multimodal Machine Learning: A Survey and Taxonomy". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.2 (2018), pp. 423–443. DOI: 10.1109/TPAMI.2018.2798607.

[23] Christoph Bartneck, Elizabeth Croft, and Dana Kulic. "Measuring the Anthropomorphism, Animacy, Likeability, Perceived Intelligence and Perceived Safety of Robots". In: *International Journal of Social Robotics* 1 (2009), pp. 71–81.

[24] Frederic Bartumeus and Jordi Catalan. "Optimal Search Behavior and Classic Foraging Theory". In: *Journal of Physics A: Mathematical and Theoretical* 42.43 (2009), p. 434002. DOI: 10.1088/1751-8113/42/43/434002.

[25] Marwen Belkaid, Kyveli Kompatsiari, Davide De Tommaso, Ingrid Zablith, and Agnieszka Wykowska. "Mutual Gaze with a Robot Affects Human Neural Activity and Delays Decision-Making Processes". In: *Science Robotics* 6.58 (2021), eabc5044. DOI: 10.1126/scirobotics.abc5044.

[26] Giovanni Bellitto, Federica Proietto Salanitri, Simone Palazzo, Francesco Rundo, Daniela Giordano, and Concetto Spampinato. "Hierarchical Domain-Adapted Feature Learning for Video Saliency Prediction". In: *International Journal of Computer Vision* 129.12 (2021), pp. 3216–3232. DOI: 10.1007/s11263-021-01519-y.

[27] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. "Learning Long-Term Dependencies with Gradient Descent Is Difficult". In: *IEEE Transactions on Neural Networks and Learning Systems* 5.2 (1994), pp. 157–166. DOI: 10.1109/TNNLS.2015.2496306.

[28] James Bergstra, Daniel Yamins, and David Cox. "Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures". In: *Proceedings of the International Conference on Machine Learning (ICML)*. https://dl.acm.org/doi/10.5555/3042817.3042832. PMLR, 2013, pp. 115–123. URL: https://hyperopt.github.io/hyperopt.

[29] Jonnathan Berrezueta-Guzman, Vladimir Espartaco Robles-Bykbaev, Iván Pau, Fernando Pesántez-Avilès, and María-Luisa Martín-Ruiz. "Robotic Technologies in ADHD Care: Literature Review". In: *IEEE Access* 10 (2022), pp. 608–625. DOI: 10.1109/ACCESS.2021.3137082.

[30] Markus Bindemann. "Scene and Screen Center Bias Early Eye Movements in Scene Viewing". In: *Vision Research* 50.23 (2010), pp. 2577–2587. DOI: 10.1016/j.visres.2010.08.016.

[31] Elina Birmingham, Walter F. Bischof, and Alan Kingstone. "Saliency Does Not Account for Fixations to Eyes Within Social Scenes". In: *Vision Research* 49.24 (2009), pp. 2992–3000. DOI: 10.1016/j.visres.2009.09.014.

[32] Elina Birmingham and Alan Kingstone. "Human Social Attention: A New Look At Past, Present, and Future Investigations". In: *Annals of the New York Academy of Sciences* 1156.1 (2009), pp. 118–140. DOI: 10.1111/j.1749-6632.2009.04468.x.

[33] Giuseppe Boccignone, Vittorio Cuculo, Alessandro D'Amelio, Giuliano Grossi, and Raffaella Lanzarotti. "Give Ear to My Face: Modelling Multimodal Attention to Social Interactions". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. Vol. 11130. Springer, 2018, pp. 331–345. DOI: 10.1007/978-3-030-11012-3_27.

[34] Giuseppe Boccignone, Vittorio Cuculo, Alessandro D'Amelio, Giuliano Grossi, and Raffaella Lanzarotti. "On Gaze Deployment to Audio-Visual Cues of Social Interactions". In: *IEEE Access* 8 (2020), pp. 161630–161654. DOI: 10.1109/ACCESS.2020.3021211.

[35] Ali Borji. "Saliency Prediction in the Deep Learning Era: Successes, Limitations, and Future Challenges". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.8 (2019), pp. 679–700. DOI: 10.1109/TPAMI.2019.2935715.

[36] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. *JAX: Composable Transformations of Python+NumPy Programs*. Version 0.4.15. 2018. URL: https://jax.readthedocs.io.

[37] Gary Bradski. "The OpenCV Library". In: *Dr. Dobb's Journal of Software Tools* (2000). URL: https://opencv.org.

[38] Cynthia Breazeal, Kerstin Dautenhahn, and Takayuki Kanda. "Social Robotics". In: *Springer Handbook of Robotics* (2016), pp. 1935–1972. DOI: 10.1007/978-3-319-32552-1_72.

[39] Neil D.B. Bruce and John K. Tsotsos. "Saliency, Attention, and Visual Search: An Information Theoretic Approach". In: *Journal of Vision* 9.3 (2009), p. 5. DOI: 10.1167/9.3.5.

[40] Nicoleta Bugnariu, Carolyn Young, Katelyn Rockenbach, Rita M. Patterson, Carolyn Garver, Isura Ranatunga, Monica Beltran, Nahum Torres-Arenas, and Dan Popa. "Human-Robot Interaction as a Tool to Evaluate and Quantify Motor Imitation Behavior in Children with Autism Spectrum Disorders". In: *Proceedings of the International Conference on Virtual Rehabilitation (ICVR)*. IEEE. 2013, pp. 57–62. DOI: 10.1109/ICVR.2013.6662088.

[41] Martin Buss et al. "Towards Proactive Human-Robot Interaction in Human Environments". In: *Proceedings of the International Conference on Cognitive Infocommunications (CogInfoCom)*. IEEE. 2011, pp. 1–6.

[42] Martin Buss et al. "Multimodal Conversational Interaction with a Humanoid Robot". In: *Proceedings of the International Conference on Cognitive Infocommunications (CogInfoCom)*. IEEE. 2012, pp. 667–672. DOI: 10.1109/CogInfoCom.2012.6421935.

[43] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Frédo Durand. "What Do Different Evaluation Metrics Tell Us About Saliency Models?" In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.3 (2019), pp. 740–757. DOI: 10.1109/TPAMI.2018.2815601.

[44] Zoya Bylinskii, Adrià Recasens, Ali Borji, Aude Oliva, Antonio Torralba, and Frédo Durand. "Where Should Saliency Models Look Next?" In: *European Conference on Computer Vision (ECCV)*. Springer. Springer, 2016, pp. 809–824. DOI: 10.1007/978-3-319-46454-1_49.

[45] Daniele Caligiore, Magda Mustile, Daniele Cipriani, Peter Redgrave, Jochen Triesch, Maria De Marsico, and Gianluca Baldassarre. "Intrinsic Motivations Drive Learning of Eye Movements: An Experiment with Human Adults". In: *PLoS One* 10.3 (2015), pp. 1–15. DOI: 10.1371/journal.pone.0118705.

[46] Natalia Calvo-Barajas, Giulia Perugia, and Ginevra Castellano. "The Effects of Robot's Facial Expressions on Children's First Impressions of Trustworthiness". In: *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2020, pp. 165–171. DOI: 10.1109/RO-MAN47096.2020.9223456.

[47] Roser Cañigueral and Antonia F. de C. Hamilton. "The Role of Eye Gaze During Natural Social Interactions in Typical and Autistic People". In: *Frontiers in Psychology* 10 (2019), p. 560. DOI: 10.3389/fpsyg.2019.00560.

[48] Giorgio Cannata, Mirko D'Andrea, and Marco Maggiali. "Design of a Humanoid Robot Eye: Models and Experiments". In: *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*. IEEE, 2006, pp. 151–156. DOI: 10.1109/ICHR.2006.321377.

[49] Evan W. Carr and Piotr Winkielman. "When Mirroring Is Both Simple and "Smart": How Mimicry Can Be Embodied, Adaptive, and Non-representational". In: *Frontiers in Human Neuroscience* 8 (2014), p. 505. DOI: 10.3389/fnhum.2014.00505.

[50] Konstantinos Charalampous, Ioannis Kostavelis, and Antonios Gasteratos. "Recent trends in social aware robot navigation: A survey". In: *Robotics and Autonomous Systems* 93 (2017), pp. 85–104. DOI: 10.1016/j.robot.2017.03.002.

[51] Sneha Chaudhari, Varun Mithal, Gungor Polatkan, and Rohan Ramanath. "An Attentive Survey of Attention Models". In: *ACM Transactions on Intelligent Systems and Technology* 12.5 (2021), pp. 2157–6904. DOI: 10.1145/3465055.

[52] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. "MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems". In: *CoRR* abs/1512.01274 (2015). DOI: 10.48550/arXiv.1512.01274. arXiv: 1512.01274. URL: https://mxnet.apache.org.

[53] Xianyu Chen, Ming Jiang, and Qi Zhao. "Predicting Human Scanpaths in Visual Question Answering". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 10876–10885. DOI: 10.1109/CVPR46437.2021.01073.

[54] Xianyu Chen, Ming Jiang, and Qi Zhao. "Local-to-Global Registration for Bundle-Adjusting Neural Radiance Fields". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 8264–8273. DOI: 10.1109/CVPR52729.2023.00799.

[55] Yizhou Chen, Mark Van der Merwe, Andrea Sipos, and Nima Fazeli. "Visuo-Tactile Transformers for Manipulation". In: *Proceedings of the Conference on Robot Learning (CoRL)*. Vol. 205. https://proceedings.mlr.press/v205/chen23d.html. PMLR, 2023, pp. 2026–2040.

[56] Chris Chesher and Fiona Andreallo. "Eye Machines: Robot Eye, Vision and Gaze". In: *International Journal of Social Robotics* 14 (2022), pp. 2071–2081. DOI: 10.1007/s12369-021-00777-7.

[57] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL, 2014, pp. 1724–1734. DOI: 10.3115/v1/D14-1179.

[58] Felipe Cid, Jose Moreno, Pablo Bustos, and Pedro Núnez. "Muecas: A Multi-Sensor Robotic Head for Affective Human Robot Interaction and Imitation". In: *Sensors* 14.5 (2014), pp. 7711–7737. DOI: 10.3390/s140507711.

[59] Roberto Cipolla, Yarin Gal, and Alex Kendall. "Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 7482–7491. DOI: 10.1109/CVPR.2018.00781.

[60] Jonathan D. Cohen, Kevin Dunbar, and James L. McClelland. "On the Control of Automatic Processes: A Parallel Distributed Processing Account of the Stroop Effect." In: *Psychological Review* 97.3 (1990), p. 332. DOI: 10.1037/0033-295x.97.3.332.

[61] Charles E. Connor, Howard E. Egeth, and Steven Yantis. "Visual Attention: Bottom-Up Versus Top-Down". In: *Current Biology* 14.19 (2004), pp. 748–752. DOI: 10.1016/j.cub.2004.09.041.

[62] Macario O. Cordel, Shaojing Fan, Zhiqi Shen, and Mohan S. Kankanhalli. "Emotion-Aware Human Attention Prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 4026–4035. DOI: 10.1109/CVPR.2019.00415.

[63] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. "Predicting Human Eye Fixations via an LSTM-Based Saliency Attentive Model". In: *IEEE Transactions on Image Processing* 27.10 (2018), pp. 5142–5154. DOI: 10.1109/TIP.2018.2851672.

[64] Enrique Coronado, Shunki Itadera, and Ixchel G. Ramirez-Alpizar. "Integrating Virtual, Mixed, and Augmented Reality to Human–Robot Interaction Applications Using Game Engines: A Brief Review of Accessible Software Tools and Frameworks". In: *Applied Sciences* 13.3 (2023), p. 1292. DOI: 10.3390/app13031292.

[65] Antoine Coutrot and Nathalie Guyader. "An Audiovisual Attention Model for Natural Conversation Scenes". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE. 2014, pp. 1100–1104. DOI: 10.1109/ICIP.2014.7025219.

[66] Antoine Coutrot and Nathalie Guyader. "An Efficient Audiovisual Saliency Model to Predict Eye Positions When Looking at Conversations". In: *Proceedings of the European Signal Processing Conference (EUSIPCO)*. IEEE, 2015, pp. 1531–1535. DOI: 10.1109/EUSIPCO.2015.7362640.

[67] Antoine Coutrot, Janet H. Hsiao, and Antoni B. Chan. "Scanpath Modeling and Classification with Hidden Markov Models". In: *Behavior Research Methods* 50.1 (2018), pp. 362–379. DOI: 10.3758/s13428-017-0876-8.

[68] Filipe Cristino, Sebastiaan Mathôt, Jan Theeuwes, and Iain D. Gilchrist. "ScanMatch: A Novel Method for Comparing Fixation Sequences". In: *Behavior Research Methods* 42.3 (2010), pp. 692–700. ISSN: 1554-3528. DOI: 10.3758/BRM.42.3.692.

[69] Alessandro D'Amelio and Giuseppe Boccignone. "Gazing at Social Interactions between Foraging and Decision Theory". In: *Frontiers in Neurorobotics* 15 (2021), p. 31. DOI: 10.3389/fnbot.2021.639999.

[70] Luisa Damiano, Paul Dumouchel, and Hagen Lehmann. "Towards Human–Robot Affective Co-Evolution Overcoming Oppositions in Constructing Emotions and Empathy". In: *International Journal of Social Robotics* 7 (2015), pp. 7–18. DOI: 10.1007/s12369-014-0258-7.

[71] Richard Dewhurst, Marcus Nyström, Halszka Jarodzka, Tom Foulsham, Roger Johansson, and Kenneth Holmqvist. "It Depends on How You Look at It: Scanpath Comparison in Multiple Dimensions with MultiMatch, a Vector-Based Approach". In: *Behavior Research Methods* 44.4 (2012), pp. 1079–1100. DOI: 10.3758/s13428-012-0212-2.

[72] Richard Droste, Jianbo Jiao, and Alison J. Noble. "Unified Image and Video Saliency Modeling". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Vol. 12350. Springer, 2020, pp. 419–435. DOI: 10.1007/978-3-030-58558-7_25.

[73] Jaime Duque-Domingo, Jaime Gómez-García-Bermejo, and Eduardo Zalama. "Gaze Control of a Robotic Head for Realistic Interaction with Humans". In: *Frontiers in Neurorobotics* 14 (2020). DOI: 10.3389/fnbot.2020.00034.

[74] Janick Edinger, Dinesh K. Pai, and Miriam Spering. "Coordinated Control of Three-Dimensional Components of Smooth Pursuit to Rotating and Translating Textures". In: *Eye Movements, Strabismus, Amblyopia and Neuro-Ophthalmology* 58 (2017), pp. 698–707. DOI: 10.1167/iovs.16-21038.

[75] Ayssam Elkady and Tarek Sobh. "Robotics Middleware: A Comprehensive Literature Survey and Attribute-Based Bibliography". In: *Journal of Robotics* 2012 (2012). DOI: 10.1155/2012/959013.

[76] Enchanted Tools. *Mirokaï*. 2023. URL: https://enchanted.tools/robot.

[77] Hinke M. Endedijk, M. Meyer, H. Bekkering, A.H.N. Cillessen, and Sabine Hunnius. "Neural Mirroring and Social Interaction: Motor System Involvement During Action Observation Relates to Early Peer Cooperation". In: *Developmental Cognitive Neuroscience* 24 (2017), pp. 33–41. DOI: 10.1016/j.dcn.2017.01.001.

[78] Tair Faibish, Alap Kshirsagar, Guy Hoffman, and Yael Edan. "Human Preferences for Robot Eye Gaze in Human-to-Robot Handovers". In: *International Journal of Social Robotics* 14.4 (2022), pp. 995–1012. DOI: 10.1007/s12369-021-00836-z.

[79] Pietro Falco, Shuang Lu, Andrea Cirillo, Ciro Natale, Salvatore Pirozzi, and Dongheui Lee. "Cross-Modal Visuo-Tactile Object Recognition Using Robotic Active Exploration". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5273–5280. DOI: 10.1109/ICRA.2017.7989619.

[80] Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. "SalUn: Empowering Machine Unlearning via Gradient-based Weight Saliency in Both Image Classification and Generation". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=gn0mIhQGNM. openreview.net, 2024.

[81] Shaojing Fan, Zhiqi Shen, Ming Jiang, Bryan L. Koenig, Juan Xu, Mohan S. Kankanhalli, and Qi Zhao. "Emotional Attention: A Study of Image Sentiment and Visual Attention". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 7521–7531. DOI: 10.1109/CVPR.2018.00785.

[82] Tom Foulsham, Jason JS Barton, Alan Kingstone, Richard Dewhurst, and Geoffrey Underwood. "Modeling Eye Movements in Visual Agnosia with a Saliency Map Approach: Bottom–up Guidance or Top–Down Strategy?" In: *Neural Networks* 24.6 (2011), pp. 665–677. DOI: 10.1016/j.neunet.2011.01.004.

[83] Tom Foulsham and Lucy Anne Sanderson. "Look Who's Talking? Sound Changes Gaze Behaviour in a Dynamic Social Scene". In: *Visual Cognition* 21.7 (2013), pp. 922–944. DOI: 10.1080/13506285.2013.849785.

[84] Tom Foulsham and Geoffrey Underwood. "What Can Saliency Models Predict About Eye Movements? Spatial and Sequential Aspects of Fixations During Encoding and Recognition". In: *Journal of Vision* 8.2 (2008), p. 6. DOI: 10.1167/8.2.6.

[85] Andrew Francl and Josh H McDermott. "Deep Neural Network Models of Sound Localization Reveal How Perception Is Adapted to Real-World Environments". In: *Nature Human Behaviour* 6 (1 2022), pp. 111–133. DOI: 10.1038/s41562-021-01244-z.

[86] Simone Frintrop and Patric Jensfelt. "Attentional Landmarks and Active Gaze Control for Visual SLAM". In: *IEEE Transactions on Robotics* 24.5 (2008), pp. 1054–1065. DOI: 10.1109/TRO.2008.2004977.

[87] Chris Frith. "Role of Facial Expressions in Social Interactions". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 364.1535 (2009), pp. 3453–3458. DOI: 10.1098/rstb.2009.0142.

[88] Di Fu, Fares Abawi, Hugo Carneiro, Matthias Kerzel, Ziwei Chen, Erik Strahl, Xun Liu, and Stefan Wermter. "A Trained Humanoid Robot can Perform Human-Like Crossmodal Social Attention and Conflict Resolution". In: *International Journal of Social Robotics* 15 (2023), pp. 1325–1340. DOI: 10.1007/s12369-023-00993-3.

[89] Di Fu, Fares Abawi, and Stefan Wermter. "The Robot in the Room: Influence of Robot Facial Expressions and Gaze on Human-Human-Robot Collaboration". In: *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2023, pp. 85–91. DOI: 10.1109/RO-MAN57019.2023.10309334.

[90] Di Fu, Pablo Barros, German I. Parisi, Haiyan Wu, Sven Magg, Xun Liu, and Stefan Wermter. "Assessing the Contribution of Semantic Congruency to Multisensory Integration and Conflict Resolution". In: *IROS 2018 Workshop on Crossmodal Learning for Intelligent Robotics*. IEEE. 2018. DOI: 10.48550/arXiv.1810.06748.

[91] Di Fu, Fares Abawi, Philipp Allgeuer, and Stefan Wermter. "Human Impression of Humanoid Robots Mirroring Social Cues". In: *Companion of the ACM/IEEE International Conference on Human-Robot Interaction (HRI Companion)*. ACM, 2024, pp. 458–462. DOI: 10.1145/3610978.3640580.

[92] Di Fu, Fares Abawi, Erik Strahl, and Stefan Wermter. "Judging by the Look: The Impact of Robot Gaze Strategies on Human Cooperation". In: *IEEE International Conference on Robot and Human Interactive Communication (RO-MAN) Workshop on Machine Learning for HRI: Bridge the Gap between Action and Perception*. 2022. DOI: 10.48550/arXiv.2208.11647.

[93] Luis A. Fuente, Hannah Ierardi, Michael Pilling, and Nigel T. Crook. "Influence of Upper Body Pose Mirroring in Human-Robot Interaction". In: *Proceedings of the International Conference on Social Robotics (ICSR)*. Springer. Springer, 2015, pp. 214–223. DOI: 10.1007/978-3-319-25554-5_22.

[94] Ruohan Gao and Kristen Grauman. "2.5D Visual Sound". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 324–333. DOI: 10.1109/CVPR.2019.00041.

[95] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. "Shortcut Learning in Deep Neural Networks". In: *Nature Machine Intelligence* 2.11 (2020), pp. 665–673. DOI: 10.1038/s42256-020-00257-z.

[96] György Gergely. "The Social Construction of the Subjective Self: The Role of Affect-Mirroring, Markedness, and Ostensive Communication in Self-Development". In: *Developmental Science and Psychoanalysis*. Routledge, 2018, pp. 45–88. DOI: 10.4324/9780429473654-4.

[97] Davide Ghiglino, Cesco Willemse, Davide De Tommaso, Francesco Bossi, and Agnieszka Wykowska. "At First Sight: Robots' Subtle Eye Movement Parameters Affect Human Attentional Engagement, Spontaneous Attunement and Perceived Human-Likeness". In: *Paladyn, Journal of Behavioral Robotics* 11.1 (2020), pp. 31–39. DOI: 10.1515/pjbr-2020-0004.

[98] Sarah Gillet, Ronald Cumbal, André Pereira, José Lopes, Olov Engwall, and Iolanda Leite. "Robot Gaze Can Mediate Participation Imbalance in Groups with Different Skill Levels". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 2021, pp. 303–311. DOI: 10.1145/3434073.3444670.

[99] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. "ImageBind: One Embedding Space to Bind Them All". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2023, pp. 15180–15190. DOI: 10.1109/CVPR52729.2023.01457.

[100] Christoph Goller and Andreas Kuchler. "Learning Task-Dependent Distributed Representations by Backpropagation through Structure". In: *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*. Vol. 1. IEEE, 1996, pp. 347–352. DOI: 10.1109/ICNN.1996.548916.

[101] Michael Görner, Robert Haschke, Helge Ritter, and Jianwei Zhang. "MoveIt! Task Constructor for Task-Level Motion Planning". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2019, pp. 190–196. DOI: 10.1109/ICRA.2019.8793898.

[102] Albert Gu, Karan Goel, and Christopher Ré. "Efficiently Modeling Long Sequences with Structured State Spaces". In: *Proceedings of the International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=uYLFoz1vlAC. openreview.net, 2022.

[103] Maria Guarnera, Zira Hichy, Maura Cascio, Stefano Carrubba, and Stefania L. Buccheri. "Facial Expressions and the Ability to Recognize Emotions From the Eyes or Mouth: A Comparison between Children and Adults". In: *The Journal of Genetic Psychology* 178.6 (2017), pp. 309–318. DOI: 10.1080/00221325.2017.1361377.

[104] Léa Haefflinger, Frédéric Elisei, Silvain Gerber, Béatrice Bouchot, Jean-Philippe Vigne, and Gérard Bailly. "On the Benefit of Independent Control of Head and Eye Movements of a Social Robot for Multiparty Human-Robot Interaction". In: *Proceedings of the International Conference on Human-Computer Interaction (HCII)*. Springer. 2023, pp. 450–466. DOI: 10.1007/978-3-031-35596-7_29.

[105] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. "Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet?" In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 6546–6555. DOI: 10.1109/CVPR.2018.00685.

[106] Charles R. Harris et al. "Array Programming with NumPy". In: *Nature* 585.7825 (2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://www.numpy.org.

[107] Anna-Katharina Hauperich, Laura K. Young, and Hannah E. Smithson. "What Makes a Microsaccade? A Review of 70 Years of Research Prompts a New Detection Method." In: *Journal of Eye Movement Research* 12.6 (2019). DOI: 10.16910/jemr.12.6.13.

[108] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving Deep Into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 1026–1034. DOI: 10.1109/ICCV.2015.123.

[109] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Identity Mappings in Deep Residual Networks". In: *European Conference on Computer Vision (ECCV)*. Springer. 2016, pp. 630–645. DOI: 10.1007/978-3-319-46493-0_38.

[110] Thorsten Hempel, Ahmed A. Abdelrahman, and Ayoub Al-Hamadi. "6D Rotation Representation for Unconstrained Head Pose Estimation". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 2496–2500. DOI: 10.1109/ICIP46576.2022.9897219.

[111] Bernhard J.M. Hess and Jakob S. Thomassen. "Kinematics of Visually-Guided Eye Movements". In: *PLoS One* 9.5 (2014), e95234. DOI: 10.1371/journal.pone.0095234.

[112] Roy S. Hessels. "How Does Gaze to Faces Support Face-to-Face Interaction? A Review and Perspective". In: *Psychonomic Bulletin & Review* 27.5 (2020), pp. 856–881. DOI: 10.3758/s13423-020-01715-w.

[113] Pieter Hintjens. *ZeroMQ: Messaging for Many Applications*. https://api.semanticscholar.org/CorpusID:190148604. O'Reilly Media, Inc., 2013. URL: https://zeromq.org.

[114] Jonathan Ho and Stefano Ermon. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems (NeurIPS)*. https://dl.acm.org/doi/10.5555/3157382.3157608. Curran Associates, Inc., 2016, pp. 4572–4580.

[115] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[116] Jonas Hornstein, Manuel Lopes, José Santos-Victor, and Francisco Lacerda. "Sound Localization for Humanoid Robots – Building Audio-Motor Maps Based on the HRTF". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2006, pp. 1170–1176. DOI: 10.1109/IROS.2006.281849.

[117] Anthony Hu and Seth Flaxman. "Multimodal Sentiment Analysis To Explore the Structure of Emotions". In: *Proceedings of the ACM International Conference on Knowledge Discovery & Data Mining (SIGKDD)*. ACM, 2018, pp. 350–358. DOI: 10.1145/3219819.3219853.

[118] Jie Hu, Li Shen, and Gang Sun. "Squeeze-and-Excitation Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 7132–7141. DOI: 10.1109/TPAMI.2019.2913372.

[119] Yifei Huang, Minjie Cai, Zhenqiang Li, and Yoichi Sato. "Predicting Gaze in Egocentric Video by Learning Task-Dependent Attention Transition". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Vol. 11208. Springer, 2018, pp. 754–769. DOI: 10.1007/978-3-030-01225-0_46.

[120] Laurent Itti and Christof Koch. "Computational Modelling of Visual Attention". In: *Nature Reviews Neuroscience* 2.3 (2001), pp. 194–203. DOI: 10.1038/35058500.

[121] Samyak Jain, Pradeep Yarlagadda, Shreyank Jyoti, Shyamgopal Karthik, Ramanathan Subramanian, and Vineet Gandhi. "ViNet: Pushing the Limits of Visual Modality for Audio-Visual Saliency Prediction". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 3520–3527. DOI: 10.1109/IROS51168.2021.9635989.

[122] Karen A. Jehn. "A Multimethod Examination of the Benefits and Detriments of Intragroup Conflict". In: *Administrative Science Quarterly* 40.2 (1995), pp. 256–282. DOI: 10.2307/2393638.

[123] Karen A. Jehn. "A Qualitative Analysis of Conflict Types and Dimensions in Organizational Groups". In: *Administrative Science Quarterly* 42.3 (1997), pp. 530–557. DOI: 10.2307/2393737.

[124] Ming Jiang, Xavier Boix, Gemma Roig, Juan Xu, Luc Van Gool, and Qi Zhao. "Learning to Predict Sequences of Human Visual Fixations". In: *IEEE Transactions on Neural Networks and Learning Systems* 27.6 (2016), pp. 1241–1252. DOI: 10.1109/TNNLS.2015.2496306.

[125] David O. Johnson, Raymond H. Cuijpers, and David van der Pol. "Imitating Human Emotions with Artificial Facial Expressions". In: *International Journal of Social Robotics* 5 (2013), pp. 503–513. DOI: 10.1007/s12369-013-0211-1.

[126] Tilke Judd, Frédo Durand, and Antonio Torralba. "A Benchmark of Computational Models of Saliency to Predict Human Fixations". In: *CSAIL Technical Reports* (2012). https://dspace.mit.edu/bitstream/handle/1721.1/68590/MIT-CSAIL-TR-2012-001.pdf?sequence=1&isAllowed=y.

[127] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. "Learning to Predict Where Humans Look". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE. 2009, pp. 2106–2113. DOI: 10.1109/ICCV.2009.5459462.

[128] Malte F. Jung, Nikolas Martelaro, and Pamela J. Hinds. "Using Robots to Moderate Team Conflict: The Case of Repairing Violations". In: *Extended Abstracts of the ACM/IEEE International Conference on Human-Robot Interaction (HRI Extended Abstracts)*. ACM. 2015, pp. 229–236. DOI: 10.1145/2701973.2702094.

[129] Moritz Kassner, William Patera, and Andreas Bulling. "Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-Based Interaction". In: *Adjunct Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UBICOMP)*. ACM, 2014, pp. 1151–1160. DOI: 10.1145/2638728.2641695.

[130] Petr Kellnhofer, Adrià Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. "Gaze360: Physically Unconstrained Gaze Estimation in the Wild". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2019, pp. 6912–6921. DOI: 10.1109/ICCV.2019.00701.

[131] Matthias Kerzel, Jakob Ambsdorf, Dennis Becker, Wenhao Lu, Erik Strahl, Josua Spisak, Connor Gäde, Tom Weber, and Stefan Wermter. "What's on Your Mind, NICO? XHRI: A Framework for eXplainable Human-Robot Interaction". In: *KI-Künstliche Intelligenz* 36.3-4 (2022), pp. 237–254. DOI: 10.1007/s13218-022-00772-8.

[132] Matthias Kerzel, Manfred Eppe, Stefan Heinrich, Fares Abawi, and Stefan Wermter. "Neurocognitive Shared Visuomotor Network for End-To-End Learning of Object Identification, Localization and Grasping on a Humanoid". In: *Proceedings of the IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL-Epirob)*. IEEE. 2019, pp. 19–24. DOI: 10.1109/DEVLRN.2019.8850679.

[133] Matthias Kerzel, Erik Strahl, Sven Magg, Nicolás Navarro-Guerrero, Stefan Heinrich, and Stefan Wermter. "NICO – Neuro-Inspired COmpanion: A Developmental Humanoid Robot Platform for Multimodal Interaction". In: *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2017, pp. 113–120. DOI: 10.1109/ROMAN.2017.8172289.

[134] Matthias Kerzel and Stefan Wermter. "Towards a Data Generation Framework for Affective Shared Perception and Social Cue Learning Using Virtual Avatars". In: *Workshop on Affective Shared Perception, ICDL 2020 (WASP)*. IEEE International Conference on Development and Learning. https://whisperproject.eu/images/WASP2020/submissions/9_ICDL_Workshop_WASP-KerzelWermter.pdf. IEEE. 2020.

[135] Matthias Kerzel, Fares Abawi, Manfred Eppe, and Stefan Wermter. "Enhancing a Neurocognitive Shared Visuomotor Model for Object Identification, Localization, and Grasping with Learning from Auxiliary Tasks". In: *IEEE Transactions on Cognitive and Developmental Systems* 14.4 (2022), pp. 1331–1343. DOI: 10.1109/TCDS.2020.3028460.

[136] Helena Kiilavuori, Veikko Sariola, Mikko J. Peltola, and Jari K. Hietanen. "Making Eye Contact with a Robot: Psychophysiological Responses to Eye Contact with a Human and with a Humanoid Robot". In: *Biological Psychology* 158 (2021), p. 107989. DOI: 10.1016/j.biopsycho.2020.107989.

[137] Yuki Kinoshita, Masanori Yokoyama, Shigeo Yoshida, Takayoshi Mochizuki, Tomohiro Yamada, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. "Transgazer: Improving Impression by Switching Direct and Averted Gaze Using Optical Illusion". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. ACM, 2017, pp. 53–62. DOI: 10.1145/2909824.3020235.

[138] Ahmad F. Klaib, Nawaf O. Alsrehin, Wasen Y. Melhem, Haneen O. Bashtawi, and Aws A. Magableh. "Eye Tracking Algorithms, Techniques, Tools, and Applications with an Emphasis on Machine Learning and Internet of Things Technologies". In: *Expert Systems with Applications* 116 (2021), p. 114037. DOI: 10.1016/j.eswa.2020.114037.

[139] Aysun Kocak, Erkut Erdem, and Aykut Erdem. "A Gated Fusion Network for Dynamic Saliency Prediction". In: *IEEE Transactions on Cognitive and Developmental Systems* 14.3 (2022), pp. 995–1008. DOI: 10.1109/TCDS.2021.3094974.

[140] Nathan Koenig and Andrew Howard. "Design and Use Paradigms for Gazebo, an Open-Source Multi-Robot Simulator". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2004, pp. 2149–2154. DOI: 10.1109/IROS.2004.1389727.

[141] Michael Koller, Astrid Weiss, Matthias Hirschmanner, and Markus Vincze. "Robotic Gaze and Human Views: A Systematic Exploration of Robotic Gaze Aversion and Its Effects on Human Behaviors and Attitudes". In: *Frontiers in Robotics and AI* 10 (2023), p. 1062714. DOI: 10.3389/frobt.2023.1062714.

[142] Kyveli Kompatsiari, Vadim Tikhanoff, Francesca Ciardo, Giorgio Metta, and Agnieszka Wykowska. "The Importance of Mutual Gaze in Human-Robot Interaction". In: *Proceedings of the Ninth International Conference on Social Robotics (ICSR)*. Springer. 2017, pp. 443–452. DOI: 10.1007/978-3-319-70022-9_44.

[143] Joshua Kramer. "Advanced Message Queuing Protocol (AMQP)". In: *Linux Journal* 2009.187 (2009).

[144] Jay Kreps, Neha Narkhede, and Jun Rao. "Kafka: A Distributed Messaging System for Log Processing". In: *NetDB'11 Workshop*. https://www.microsoft.com/en-us/research/wp-content/uploads/2017/09/Kafka.pdf. ACM. 2011. URL: https://kafka.apache.org.

[145] Alap Kshirsagar, Melanie Lim, Shemar Christian, and Guy Hoffman. "Robot Gaze Behaviors in Human-to-Robot Handovers". In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 6552–6558. DOI: 10.1109/LRA.2020.3015692.

[146] Matthias Kümmerer and Matthias Bethge. "State-of-the-Art in Human Scanpath Prediction". In: *CoRR* abs/2102.12239 (2021). DOI: 10.48550/arXiv.2102.12239. arXiv: 2102.12239.

[147] Matthias Kümmerer and Matthias Bethge. "Predicting Visual Fixations". In: *Annual Review of Vision Science* 9 (2023), pp. 269–291. DOI: 10.1146/annurev-vision-120822-072528.

[148] Guohao Lan, Tim Scargill, and Maria Gorlatova. "EyeSyn: Psychology-Inspired Eye Movement Synthesis for Gaze-Based Activity Recognition". In: *Proceedings of the ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2022, pp. 233–246. DOI: 10.1109/IPSN54338.2022.00026.

[149] Stéphane Lathuilière, Benoit Massé, Pablo Mesejo, and Radu Horaud. "Neural Network Based Reinforcement Learning for Audio-Visual Gaze Control in Human-Robot Interaction". In: *Pattern Recognition Letters* 118 (2019), pp. 61–71. DOI: 10.1016/j.patrec.2018.05.02.

[150] Aoqi Li and Zhenzhong Chen. "Individual Trait Oriented Scanpath Prediction for Visual Attention Analysis". In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. IEEE. 2017, pp. 3745–3749. DOI: 10.1109/ICIP.2017.8296982.

[151] Jamy Li, Wendy Ju, and Cliff Nass. "Observer Perception of Dominance and Mirroring Behavior in Human-Robot Relationships". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. ACM, 2015, pp. 133–140. DOI: 10.1145/2696454.2696459.

[152] Mengdi Li, Xufeng Zhao, Jae Hee Lee, Cornelius Weber, and Stefan Wermter. "Internally Rewarded Reinforcement Learning". In: *Proceedings of the International Conference on Machine Learning (ICML)*. https://proceedings.mlr.press/v202/li23ax. PMLR, 2023, pp. 20556–20574.

[153] Mu Li, Kanglong Fan, and Kede Ma. "Scanpath Prediction in Panoramic Videos via Expected Code Length Minimization". In: *CoRR* abs/2305.02536 (2023). DOI: 10.48550/arXiv.2305.02536. arXiv: 2305.02536.

[154] Yin Li, Alireza Fathi, and James M. Rehg. "Learning to Predict Gaze in Egocentric Video". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2013, pp. 3216–3223. DOI: 10.1109/ICCV.2013.399.

[155] Velvetina Lim, Maki Rooksby, and Emily S. Cross. "Social Robots on a Global Stage: Establishing a Role for Culture During Human-Robot Interaction". In: *International Journal of Social Robotics* 13.6 (2021), pp. 1307–1333. DOI: 10.1007/s12369-020-00710-4.

[156] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. "BARF: Bundle-Adjusting Neural Radiance Fields". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2021, pp. 5721–5731. DOI: 10.1109/ICCV48922.2021.00569.

[157] Pei-Chun Lin, Patrick CK Hung, Ying Jiang, Carolina Padilla Velasco, and Marco Antonio Martínez Cano. "An Experimental Design for Facial and Color Emotion Expression of a Social Robot". In: *The Journal of Supercomputing* 79.2 (2023), pp. 1980–2009. DOI: 10.1007/s11227-022-04734-7.

[158] Yufan Liu, Minglang Qiao, Mai Xu, Bing Li, Weiming Hu, and Ali Borji. "Learning to Predict Salient Faces: A Novel Visual-Audio Saliency Model". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Vol. 12365. Springer, 2020, pp. 413–429. DOI: 10.1007/978-3-030-58565-5_25.

[159] David G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In: *International Journal of Computer Vision* 60 (2004), pp. 91–110. DOI: 10.1023/B:VISI.0000029664.99615.94.

[160] Xinhui Luo, Zhi Liu, Weijie Wei, Linwei Ye, Tianhong Zhang, Lihua Xu, and Jijun Wang. "Few-Shot Personalized Saliency Prediction Using Meta-Learning". In: *Image and Vision Computing* 124 (2022), p. 104491. DOI: 10.1016/j.imavis.2022.104491.

[161] Thang Luong, Hieu Pham, and Christopher D. Manning. "Effective Approaches to Attention-Based Neural Machine Translation". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL, 2015, pp. 1412–1421. DOI: 10.18653/v1/D15-1166.

[162] Yanjun Ma, Dianhai Yu, Tian Wu, and Haifeng Wang. "PaddlePaddle: An Open-Source Deep Learning Platform from Industrial Practice". In: *Frontiers of Data and Computing* 1.1 (2019), pp. 105–115. DOI: 10.11871/jfdc.issn.2096.742X.2019.01.011. URL: https://www.paddlepaddle.org.cn.

[163] Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. "Robot Operating System 2: Design, Architecture, and Uses in the Wild". In: *Science Robotics* 7.66 (2022), eabm6074. DOI: 10.1126/scirobotics.abm6074. URL: https://www.ros.org.

[164] Robert Mahony, Tarek Hamel, and Jean-Michel Pflimlin. "Nonlinear Complementary Filters on the Special Orthogonal Group". In: *IEEE Transactions on Automatic Control* 53.5 (2008), pp. 1203–1218. DOI: 10.1109/CDC.2005.1582367.

[165] Anthony Mallet, Cédric Pasteur, Matthieu Herrb, Séverin Lemaignan, and Félix Ingrand. "GenoM3: Building Middleware-Independent Robotic Components". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 4627–4632. DOI: 10.1109/ROBOT.2010.5509539.

[166] Shashank Rao Marpally, Pranav Goyal, and Harold Soh. "Towards Automated Scenario Testing of Social Navigation Algorithms". In: *Unsolved Problems in Social Robot Navigation Workshop at RSS2024* (2024). https://unsolvedsocialnav.org/papers/Marpally.pdf.

[167] Christoforos Mavrogiannis, Francesca Baldini, Allan Wang, Dapeng Zhao, Pete Trautman, Aaron Steinfeld, and Jean Oh. "Core Challenges of Social Robot Navigation: A Survey". In: *Journal of Human-Robot Interaction* 12.3 (2023). DOI: 10.1145/3583741.

[168] Wes McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the Python in Science Conference (SciPy)*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 56–61. DOI: 10.25080/Majora-92bf1922-00a.

[169] Heather E. McNeely, Robert West, Bruce K. Christensen, and Claude Alain. "Neurophysiological Evidence for Disturbances of Conflict Processing in Patients with Schizophrenia." In: *Journal of Abnormal Psychology* 112.4 (2003), p. 679. DOI: 10.1037/0021-843X.112.4.679.

[170] Giorgio Metta, Paul Fitzpatrick, and Lorenzo Natale. "YARP: Yet Another Robot Platform". In: *International Journal of Advanced Robotic Systems* 3.1 (2006), p. 8. DOI: 10.5772/5761. URL: https://www.yarp.it.

[171] Giorgio Metta, Lorenzo Natale, Francesco Nori, Giulio Sandini, David Vernon, Luciano Fadiga, Claes Von Hofsten, Kerstin Rosander, Manuel Lopes, José Santos-Victor, Alexandre Bernardino, and Luis Montesano. "The iCub Humanoid Robot: An Open-Systems Platform for Research in Cognitive Development". In: *Neural Networks* 23.8-9 (2010), pp. 1125–1134. DOI: 10.1016/j.neunet.2010.08.010.

[172] Kyle Min and Jason J. Corso. "TASED-Net: Temporally-Aggregating Spatial Encoder-Decoder Network for Video Saliency Detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 2019, pp. 2394–2403. DOI: 10.1109/ICCV.2019.00248.

[173] Xiongkuo Min, Guangtao Zhai, Jiantao Zhou, Xiao-Ping Zhang, Xiaokang Yang, and Xinping Guan. "A Multimodal Saliency Model for Videos with High Audio-Visual Correspondence". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 3805–3819. DOI: 10.1109/TIP.2020.2966082.

[174] Takashi Minato, Michihiro Shimada, Shoji Itakura, Kang Lee, and Hiroshi Ishiguro. "Evaluating the Human Likeness of an Android by Comparing Gaze Behaviors Elicited by the Android and a Person". In: *Advanced Robotics* 20.10 (2006), pp. 1147–1163. DOI: 10.1163/156855306778522505.

[175] Biswajeeban Mishra and Attila Kertesz. "The Use of MQTT in M2M and IoT Systems: A Survey". In: *IEEE Access* 8 (2020), pp. 201071–201086. DOI: 10.1109/ACCESS.2020.3035849.

[176] Chinmaya Mishra and Gabriel Skantze. "Knowing Where to Look: A Planning-based Architecture to Automate the Gaze Behavior of Social Robots". In: *Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2022, pp. 1201–1208. DOI: 10.1109/RO-MAN53752.2022.9900740.

[177] Parag K. Mital, Tim J. Smith, Robin L. Hill, and John M. Henderson. "Clustering of Gaze During Dynamic Scene Viewing is Predicted by Motion". In: *Cognitive Computation* 3.1 (2011), pp. 5–24. DOI: 10.1007/s12559-010-9074-z.

[178] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. "Recurrent Models of Visual Attention". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 2. https://dl.acm.org/doi/abs/10.5555/2969033.2969073. Curran Associates, Inc., 2014, pp. 2204–2212.

[179] Volodymyr Mnih et al. "Human-Level Control through Deep Reinforcement Learning". In: *Nature* 518.7540 (2015), pp. 529–533. DOI: 10.1038/nature14236.

[180] Youssef Mohamed and Séverin Lemaignan. "ROS for Human-Robot Interaction". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3020–3027. DOI: 10.1109/IROS51168.2021.9636816.

[181] Hadi Beik Mohammadi et al. "Designing a Personality-Driven Robot for a Human-Robot Interaction Scenario". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4317–4324. DOI: 10.1109/ICRA.2019.8793770.

[182] Henrik Møller. "Fundamentals of Binaural Technology". In: *Applied Acoustics* 36.3-4 (1992), pp. 171–218. DOI: 10.1016/0003-682X(92)90046-U.

[183] Tirin Moore and Marc Zirnsak. "Neural Mechanisms of Selective Visual Attention". In: *Annual Review of Psychology* 68 (2017), pp. 47–72. DOI: 10.1146/annurev-psych-122414-033400.

[184] Bilge Mutlu, Jodi Forlizzi, and Jessica Hodgins. "A Storytelling Robot: Modeling and Evaluation of Human-like Gaze Behavior". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2006, pp. 518–523. DOI: 10.1109/ICHR.2006.321322.

[185] Si-Ahmed Naas, Xiaolan Jiang, Stephan Sigg, and Yusheng Ji. "Functional Gaze Prediction in Egocentric Video". In: *Proceedings of the International Conference on Advances in Mobile Computing & Multimedia (MoMM)*. ACM, 2020, pp. 40–47. DOI: 10.1145/3428690.3429174.

[186] Lorenzo Natale, Ali Paikan, Marco Randazzo, and Daniele E. Domenichelli. "The iCub Software Architecture: Evolution and Lessons Learned". In: *Frontiers in Robotics and AI* 3 (2016), p. 24. DOI: 10.3389/frobt.2016.00024.

[187] Roland Neumann and Fritz Strack. "'Mood Contagion': The Automatic Transfer of Mood between Persons." In: *Journal of Personality and Social Psychology* 79.2 (2000), p. 211. DOI: 10.1037/0022-3514.79.2.211.

[188] Tatsuya Nomura, Tomohiro Suzuki, Takayuki Kanda, and Kensuke Kato. "Measurement of Negative Attitudes toward Robots". In: *Interaction Studies. Social Behaviour and Communication in Biological and Artificial Systems* 7.3 (2006), pp. 437–454. DOI: 10.1075/is.7.3.14nom.

[189] Liam J. Norman and Lore Thaler. "Retinotopic-Like Maps of Spatial Sound in Primary 'Visual' Cortex of Blind Human Echolocators". In: *Proceedings of the Royal Society B: Biological Sciences* 286.1912 (2019), p. 219190. DOI: 10.1098/rspb.2019.1910.

[190] David Noton and Lawrence Stark. "Scanpaths in Saccadic Eye Movements While Viewing and Recognizing Patterns". In: *Vision Research* 11.9 (1971), pp. 929–942. DOI: 10.1016/0042-6989(71)90213-6.

[191] Lauri Nummenmaa and Andrew J. Calder. "Neural Mechanisms of Social Attention". In: *Trends in Cognitive Sciences* 13.3 (2009), pp. 135–143. DOI: 10.1016/j.tics.2008.12.006.

[192] Abraham Montoya Obeso, Jenny Benois-Pineau, Mireya Saraí García Vázquez, and Alejandro Álvaro Ramírez Acosta. "Visual vs Internal Attention Mechanisms in Deep Neural Networks for Image Classification and Object Detection". In: *Pattern Recognition* 123 (2008), p. 108411. DOI: 10.1016/j.patcog.2021.108411.

[193] Hisashi Ohtsuki, Christoph Hauert, Erez Lieberman, and Martin A Nowak. "A Simple Rule for the Evolution of Cooperation on Graphs and Social Networks". In: *Nature* 441.7092 (2006), pp. 502–505. DOI: 10.1038/nature04605.

[194] Andrew J. Oswald, Eugenio Proto, and Daniel Sgroi. "Happiness and Productivity". In: *Journal of Labor Economics* 33.4 (2015), pp. 789–822. DOI: 10.2139/ssrn.1526075.

[195] Matthew K.X.J. Pan, Sungjoon Choi, James Kennedy, Kyna McIntosh, Daniel Campos Zamora, Günter Niemeyer, Joohyung Kim, Alexis Wieland, and David Christensen. "Realistic and Interactive Robot Gaze". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 11072–11078. DOI: 10.1109/IROS45743.2020.9341297.

[196] Letitia Parcalabescu, Nils Trost, and Anette Frank. "What is Multimodality?" In: *Proceedings of the First Workshop on Multimodal Semantic Representations (MMSR)*. ACL. 2021, pp. 1–10. DOI: 10.48550/arXiv.2103.06304.

[197] Gerardo Pardo-Castellote. "OMG Data-Distribution Service: Architectural Overview". In: *Proceedings of the International Conference on Distributed Computing Systems Workshops (ICDCSW)*. IEEE. 2003, pp. 200–206. DOI: 10.1109/ICDCSW.2003.1203555.

[198] German I. Parisi, Pablo Barros, Di Fu, Sven Magg, Haiyan Wu, Xun Liu, and Stefan Wermter. "A Neurorobotic Experiment for Crossmodal Conflict Resolution in Complex Environments". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 2330–2335. DOI: 10.1109/IROS.2018.8594036.

[199] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems (NeurIPS)*. https://dl.acm.org/doi/10.5555/3454287.3455008. Curran Associates, Inc., 2019, pp. 8026–8037. URL: https://www.pytorch.org.

[200] Andre Pereira, Catharine Oertel, Leonor Fermoselle, Joe Mendelson, and Joakim Gustafson. "Responsive Joint Attention in Human-Robot Interaction". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2019, pp. 1080–1087. DOI: 10.1109/IROS40897.2019.8968130.

[201] Adrien Peyrache, Natalie Schieferstein, and Gyorgy Buzsàki. "Transformation of the Head-Direction Signal into a Spatial Code". In: *Nature Communications* 8.1 (2017), p. 1752. DOI: 10.1038/s41467-017-01908-3.

[202] Michael I. Posner and Yoav Cohen. "Components of Visual Orienting". In: *Attention and performance X: Control of language processes*. Psychology Press, 1984, pp. 531–556.

[203] Pooja Prajod, Matteo Lavit Nicora, Matteo Malosio, and Elisabeth André. "Gaze-Based Attention Recognition for Human-Robot Collaboration". In: *Proceedings of the International Conference on PErvasive Technologies Related to Assistive Environments (PETRA)*. ACM, 2023, pp. 140–147. DOI: 10.1145/3594806.3594819.

[204] Carla Pritsch, Silke Telkemeyer, Cordelia Mühlenbeck, and Katja Liebal. "Perception of Facial Expressions Reveals Selective Affect-Biased Attention in Humans and Orangutans". In: *Scientific Reports* 7.1 (2017), pp. 1–12. DOI: 10.1038/s41598-017-07563-4.

[205] Minglang Qiao, Yufan Liu, Mai Xu, Xin Deng, Bing Li, Weiming Hu, and Ali Borji. "Joint Learning of Audio-Visual Saliency Prediction and Sound Source Localization on Multi-Face Videos". In: *International Journal of Computer Vision* (2023), pp. 1–23. DOI: 10.1007/s11263-023-01950-3.

[206] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y. Ng. "ROS: An Open-Source Robot Operating System". In: *IEEE International Conference on Robotics and Automation Workshop on Open Source Software (ICRAOSS)*. Vol. 3.2. https://api.semanticscholar.org/CorpusID:6324125. IEEE. 2009, p. 5. URL: https://www.ros.org.

[207] Kranthi Kumar Rachavarapu, Vignesh Sundaresha, Aakanksha, and AN Rajagopalan. "Localize to Binauralize: Audio Spatialization from Visual Sound Source Localization". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE. 2021, pp. 1930–1939. DOI: 10.1109/ICCV48922.2021.00194.

[208] Timothy P. Racine, David A. Leavens, Colwyn Trevarthen, Peter Hobson, Jessica Hobson, Vasudevi Reddy, Malinda Carpenter, Kristin Liebal, Stephen V. Shepherd, Massimiliano L. Cappuccio, et al. *Joint Attention: New Developments in Psychology, Philosophy of Mind, and Social Neuroscience*. MIT Press, 2012.

[209] Keith Rayner. "Eye Movements in Reading and Information Processing: 20 Years of Research." In: *Psychological Bulletin* 124.3 (1998), p. 372. DOI: 10.1037/0033-2909.124.3.372.

[210] Adrià Recasens, Carl Vondrick, Aditya Khosla, and Antonio Torralba. "Following Gaze in Video". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 1435–1443. DOI: 10.1109/ICCV.2017.160.

[211] Mauricio E. Reyes, Ivan V. Meza, and Luis A. Pineda. "The Positive Effect of Negative Feedback in HRI Using a Facial Expression Robot". In: *International Workshop in Cultural Robotics*. Springer. 2016, pp. 44–54. DOI: 10.1007/978-3-319-42945-8_4.

[212] Mauricio E. Reyes, Ivan V. Meza, and Luis A. Pineda. "Robotics Facial Expression of Anger in Collaborative Human-Robot Interaction". In: *International Journal of Advanced Robotic Systems* 16.1 (2019), p. 1729881418817972. DOI: 10.1177/1729881418817972.

[213] Neal Richardson, Ian Cook, Nic Crane, Dewey Dunnington, Romain François, Jonathan Keane, Dragoș Moldovan-Grünfeld, Jeroen Ooms, and Apache Arrow. *Arrow: Integration to 'Apache' 'Arrow'*. 2023. URL: https://arrow.apache.org.

[214] Matthew Rocklin. "Dask: Parallel Computation with Blocked algorithms and Task Scheduling". In: *Proceedings of the Python in Science Conference (SciPy)*. Ed. by Kathryn Huff and James Bergstra. SciPy, 2015, pp. 130–136. DOI: 10.25080/Majora-7b98e3ed-013. URL: https://www.dask.org.

[215] Juan Antonio Rojas-Quintero and María del C. Rodríguez-Liñán. "A literature Review of Sensor Heads for Humanoid Robots". In: *Robotics and Autonomous Systems* 143 (2021), p. 103834. DOI: 10.1016/j.robot.2021.103834.

[216] Tim Rolff, Frank Steinicke, and Simone Frintrop. "Gaze Mapping for Immersive Virtual Environments based on Image Retrieval". In: *Frontiers in Virtual Reality* 3 (2022). DOI: 10.3389/frvir.2022.802318.

[217] Alessandro Roncone, Ugo Pattacini, Giorgio Metta, and Lorenzo Natale. "A Cartesian 6-DoF Gaze Controller for Humanoid Robots". In: *Robotics: Science and Systems (RSS)*. Vol. 2016. 2016. DOI: 10.15607/RSS.2016.XII.022.

[218] Miguel Fabián Romero Rondón, Lucile Sassatelli, Ramón Aparicio-Pardo, and Frédéric Precioso. "TRACK: A New Method from a Re-Examination of Deep Architectures for Head Motion Prediction in 360° Videos". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.9 (2022), pp. 5681–5699. DOI: 10.1109/TPAMI.2021.3070520.

[219] Keith H. Ruddock, David S. Wooding, and Sabira K. Mannan. "The Relationship between the Locations of Spatial Features and Those of Fixations Made during Visual Examination of Briefly Presented Images". In: *Spatial Vision* 10.3 (1996), pp. 165–188. DOI: 10.1163/156856896x00123.

[220] Kerstin Ruhland, Christopher E. Peters, Sean Andrist, Jeremy B. Badler, Norman I. Badler, Michael Gleicher, Bilge Mutlu, and Rachel McDonnell. "A Review of Eye Gaze in Virtual Agents, Social Robotics and HCI: Behaviour Generation, User Interaction and Perception: A Review of Eye Gaze". In: *Computer Graphics Forum* 34.6 (2015), pp. 299–326. DOI: 10.1111/cgf.12603.

[221] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning Internal Representations by Error Propagation". In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. MIT Press, 1986, pp. 318–362. DOI: 10.1016/B978-1-4832-1446-7.50035-2.

[222] Selma Šabanović. "Robots in Society, Society in Robots: Mutual Shaping of Society and Technology as a Framework for Social Robot Design". In: *International Journal of Social Robotics* 2.4 (2010), pp. 439–450. DOI: 10.1007/s12369-010-0066-7.

[223] Brenda Salley and John Colombo. "Conceptualizing Social Attention in Developmental Research". In: *Social Development* 25.4 (2016), pp. 687–703. DOI: 10.1111/sode.12174.

[224] Akanksha Saran, Elaine Schaertl Short, Andrea Thomaz, and Scott Niekum. "Understanding Teacher Gaze Patterns for Robot Learning". In: *Proceedings of the Conference on Robot Learning (CoRL)*. Vol. 100. https://proceedings.mlr.press/v100/saran20a.html. PMLR, 2020, pp. 1247–1258.

[225] Shane Saunderson and Goldie Nejat. "How Robots Influence Humans: A Survey of Nonverbal Communication in Social Human-Robot Interaction". In: *International Journal of Social Robotics* 11 (2019), pp. 575–608. DOI: 10.1007/s12369-019-00523-0.

[226] Brian Scassellati, Laura Boccanfuso, Chien-Ming Huang, Marilena Mademtzi, Meiying Qin, Nicole Salomons, Pamela Ventola, and Frederick Shic. "Improving Social Skills in Children with ASD Using a Long-Term, In-Home Social Robot". In: *Science Robotics* 3.21 (2018), eaat7544. DOI: 10.1126/scirobotics.aat7544.

[227] Jürgen Schmidhuber. "Deep Learning in Neural Networks: An Overview". In: *Neural Networks* 61.6 (2015), pp. 85–117. DOI: 10.1016/j.neunet.2014.09.00.

[228] Johanna Seifert, Orsolya Friedrich, and Sebastian Schleidgen. "Imitating the Human. New Human-Machine Interactions in Social Robots". In: *NanoEthics* 16.2 (2022), pp. 181–192. DOI: 10.1007/s11569-022-00418-x.

[229] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. "GRAD-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization". In: *Proceedings of the IEEE international Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 618–626. DOI: 10.1007/s11263-019-01228-7.

[230] Hae Jong Seo and Peyman Milanfar. "Static and Space-Time Visual Saliency Detection by Self-Resemblance". In: *Journal of Vision* 9.12 (2009), pp. 15–15. DOI: 10.1167/9.12.15.

[231] Stephen V. Shepherd. "Following Gaze: Gaze-Following Behavior as a Window into Social Cognition". In: *Frontiers in Integrative Neuroscience* 4 (2010), p. 5. DOI: 10.3389/fnint.2010.00005.

[232] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 28. https://dl.acm.org/doi/10.5555/2969239.2969329. Curran Associates, Inc., 2015, pp. 802–810.

[233] Masahiro Shiomi, Takayuki Kanda, Nicolas Miralles, Takahiro Miyashita, Ian Fasel, Javier Movellan, and Hiroshi Ishiguro. "Face-to-Face Interactive Humanoid Robot". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vol. 2. IEEE. 2004, pp. 1340–1346. DOI: 10.1109/IROS.2004.1389582.

[234] Henrique Siqueira, Sven Magg, and Stefan Wermter. "Efficient Facial Feature Learning with Wide Ensemble-Based Convolutional Neural Networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. AAAI, 2020, pp. 5800–5809. DOI: 10.1609/aaai.v34i04.6037.

[235] SoftBank Robotics Group. *Pepper the Humanoid and Programmable Robot*. 2014. URL: https://www.aldebaran.com/en/pepper.

[236] Sichao Song and Seiji Yamada. "Expressing Emotions through Color, Sound, and Vibration with an Appearance-Constrained Social Robot". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. ACM, 2017, pp. 2–11. DOI: 10.1145/2909824.3020239.

[237] Christoph Strauch, Alex J. Hoogerbrugge, and Antonia F. Ten Brink. "Gaze Data of 4243 Participants Shows Link between Leftward and Superior Attention Biases and Age". In: *Experimental Brain Research* (2024), pp. 1–11. DOI: 10.1007/s00221-024-06823-w.

[238] Sarah Strohkorb, Ethan Fukuto, Natalie Warren, Charles Taylor, Bobby Berry, and Brian Scassellati. "Improving Human-Human Collaboration between Children with a Social Robot". In: *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2016, pp. 551–556. DOI: 10.1109/ROMAN.2016.7745172.

[239] Florian Strohm, Mihai Bâce, and Andreas Bulling. "Learning User Embeddings from Human Gaze for Personalised Saliency Prediction". In: *CoRR* (2024). DOI: 10.48550/arXiv.2403.13653. arXiv: 2403.13653.

[240] Wanjie Sun, Zhenzhong Chen, and Feng Wu. "Visual Scanpath Prediction Using IOR-ROI Recurrent Mixture Density Network". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.6 (2019), pp. 2101–2118. DOI: 10.1109/TPAMI.2019.2956930.

[241] Yusuke Tanaka and Ankur Mehta. "REMS: Middleware for Robotics Education and Development". In: *CoRR* abs/2210.05784 (2022). DOI: 10.48550/arXiv.2210.05784. arXiv: 2210.05784.

[242] Martin Tanis and Tom Postmes. "Social Cues and Impression Formation in CMC". In: *Journal of Communication* 53.4 (2003), pp. 676–693. DOI: 10.1111/j.1460-2466.2003.tb02917.x.

[243] Karen Tatarian, Rebecca Stower, Damien Rudaz, Marine Chamoux, Arvid Kappas, and Mohamed Chetouani. "How does Modality Matter? Investigating the Synthesis and Effects of Multi-Modal Robot Behavior on Social Intelligence". In: *International Journal of Social Robotics* 14 (2022), pp. 893–911. DOI: 10.1007/s12369-021-00839-w.

[244] Gyan Tatiya and Jivko Sinapov. "Deep Multi-Sensory Object Category Recognition Using Interactive Behavioral Exploration". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7872–7878. DOI: 10.1109/ICRA.2019.8794095.

[245] Hamed R. Tavakoli, Ali Borji, Juho Kannala, and Esa Rahtu. "Deep Audio-Visual Saliency: Baseline Model and Data". In: *ACM Symposium on Eye Tracking Research and Applications (ETRA)*. ETRA '20 Short Papers. ACM, 2020, pp. 1–5. DOI: 10.1145/3379156.3391337.

[246] The pandas development team. *pandas-dev/pandas: Pandas*. 2020. DOI: 10.5281/zenodo.3509134.

[247] Yunus Terzioğlu, Bilge Mutlu, and Erol Şahin. "Designing Social Cues for Collaborative Robots: The Role of Gaze and Breathing In Human-Robot Collaboration". In: *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 2020, pp. 343–357.

[248] Kristina Tesch and Timo Gerkmann. "Multi-Channel Speech Separation Using Spatially Selective Deep Non-Linear Filters". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 32 (2024), pp. 542–553. DOI: 10.1109/TASLP.2023.3334101.

[249] Andrea Testa, Andrea Camisa, and Giuseppe Notarstefano. "ChoiRbot: A ROS 2 Toolbox for Cooperative Robotics". In: *IEEE Robotics and Automation Letters* 6.2 (2021), pp. 2714–2720. DOI: 10.1109/LRA.2021.3061366.

[250] Vadim Tikhanoff, Angelo Cangelosi, Paul M. Fitzpatrick, Giorgio Metta, Lorenzo Natale, and Francesco Nori. "An Open-Source Simulator for Cognitive Robotics Research: The Prototype of the ICub Humanoid Robot Simulator". In: *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems (PerMIS '08)*. ACM, 2008, pp. 57–61. DOI: 10.1145/1774674.1774684.

[251] Rebecca M. Todd, William A. Cunningham, Adam K. Anderson, and Evan Thompson. "Affect-Biased Attention As Emotion Regulation". In: *Trends in Cognitive Sciences* 16.7 (2012), pp. 365–372. DOI: 10.1016/j.tics.2012.06.003.

[252] Emanuel Todorov, Tom Erez, and Yuval Tassa. "MuJoCo: A physics Engine for Model-Based Control". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 5026–5033. DOI: 10.1109/IROS.2012.6386109.

[253] Michael Tomasello and Amrisha Vaish. "Origins of Human Cooperation and Morality". In: *Annual Review of Psychology* 64 (2013), pp. 231–255. DOI: 10.1146/annurev-psych-113011-143812.

[254] Claude Toussaint, Philipp T. Schwarz, and Markus Petermann. "Navel – a Social Robot with Verbal and Nonverbal Communication Skills". In: *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA)*. ACM, 2023. DOI: 10.1145/3544549.3583898.

[255] Anne M Treisman and Garry Gelade. "A Feature-Integration Theory of Attention". In: *Cognitive Psychology* 12.1 (1980), pp. 97–136. DOI: 10.1093/acprof:osobl/9780199734337.003.0011.

[256] Antigoni Tsiami, Petros Koutras, and Petros Maragos. "STAViS: Spatio-Temporal AudioVisual Saliency Network". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 4766–4776. DOI: 10.1109/CVPR42600.2020.00482.

[257] Roger P.G. Van Gompel. *Eye Movements: A Window on Mind and Brain*. Elsevier, 2007. DOI: 10.1016/B978-0-08-044980-7.X5000-9.

[258] Lyn M. Van Swol. "The Effects of Nonverbal Mirroring on Perceived Persuasiveness, Agreement with an Imitator, and Reciprocity in a Group Discussion". In: *Communication Research* 30.4 (2003), pp. 461–480. DOI: 10.1177/0093650203253318.

[259] Valentina Vasco, Arren Glover, Yeshasvi Tirupachuri, Fabio Solari, Manuela Chessa, and Chiara Bartolozzi. "Vergence Control with a Neuromorphic iCub". In: *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 732–738. DOI: 10.1109/HUMANOIDS.2016.7803355.

[260] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention Is All You Need". In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 30. https://dl.acm.org/doi/10.5555/3295222.3295349. Curran Associates, Inc., 2017, pp. 6000–6010.

[261] Petra Vetter, Łukasz Bola, Lior Reich, Matthew Bennett, Lars Muckli, and Amir Amedi. "Decoding Natural Sounds in Early "Visual" Cortex of Congenitally Blind Individuals". In: *Current Biology* 30.15 (2020), pp. 3039–3044. DOI: 10.1016/j.cub.2020.05.071.

[262] Shuo Wang and Ralph Adolphs. "Social Saliency". In: *Computational and Cognitive Neuroscience of Vision*. Springer, 2017, pp. 171–193. DOI: 10.1007/978-981-10-0213-7_8.

[263] Ziqiang Wang, Zhi Liu, Gongyang Li, Yang Wang, Tianhong Zhang, Lihua Xu, and Jijun Wang. "Spatio-Temporal Self-Attention Network for Video Saliency Prediction". In: *IEEE Transactions on Multimedia* (2021). DOI: 10.1109/TMM.2021.3139743.

[264] David Watson, David Wiese, Jatin Vaidya, and Auke Tellegen. "The Two General Activation Systems of Affect: Structural Findings, Evolutionary Considerations, and Psychobiological Evidence." In: *Journal of Personality and Social Psychology* 76.5 (1999), pp. 820–838. DOI: 10.1037/0022-3514.76.5.820.

[265] Stefan Wermter, Günther Palm, Cornelius Weber, and Mark Elshaw. "Towards Biomimetic Neural Learning for Intelligent Robots". In: *Biomimetic Neural Learning for Intelligent Robots: Intelligent Systems, Cognitive Robotics, and Neuroscience*. Springer, 2005, pp. 1–18. DOI: 10.1007/11521082_1.

[266] Ronald J. Williams and David Zipser. "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks". In: *Neural Computation* 1.2 (1989), pp. 270–280. DOI: 10.1162/neco.1989.1.2.270.

[267] Lisa Wilms and Daniel Oberfeld. "Color and Emotion: Effects of Hue, Saturation, and Brightness". In: *Psychological Research* 82.5 (2018), pp. 896–914. DOI: 10.1007/s00426-017-0880-8.

[268] Xinyi Wu, Zhenyao Wu, Lili Ju, and Song Wang. "Binaural Audio-Visual Localization". In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Vol. 35. 4. AAAI, 2021, pp. 2961–2968. DOI: 10.1609/aaai.v35i4.16403.

[269] Agnieszka Wykowska. "Robots As Mirrors of the Human Mind". In: *Current Directions in Psychological Science* 30.1 (2021), pp. 34–40. DOI: 10.1177/0963721420978609.

[270] Jiawei Xie, Zhi Liu, Gongyang Li, and Yingjie Song. "Audio-Visual Saliency Prediction with Multisensory Perception and Integration". In: *Image and Vision Computing* 143 (2024), p. 104955. DOI: 10.1016/j.imavis.2024.104955.

[271] Junwen Xiong, Ganglai Wang, Peng Zhang, Wei Huang, Yufei Zha, and Guangtao Zhai. "CASP-Net: Rethinking Video Saliency Prediction from an Audio-Visual Consistency Perceptual Perspective". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2023, pp. 6441–6450. DOI: 10.1109/CVPR52729.2023.00623.

[272] Mai Xu, Yufan Liu, Roland Hu, and Feng He. "Find Who to Look At: Turning from Action to Saliency". In: *IEEE Transactions on Image Processing* 27.9 (2018), pp. 4529–4544. DOI: 10.1109/TIP.2018.2837106.

[273] Mai Xu, Yuhang Song, Jianyi Wang, Minglang Qiao, Liangyu Huo, and Zulin Wang. "Predicting Head Movement in Panoramic Video: A Deep Reinforcement Learning Approach". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.11 (2019), pp. 2693–2708. DOI: 10.1109/TPAMI.2018.2858783.

[274] Peng Xu, Xiatian Zhu, and David A. Clifton. "Multimodal Learning with Transformers: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.10 (2023), pp. 12113–12132. DOI: 10.1109/TPAMI.2023.3275156.

[275] Wenbin Xu, Xudong Li, Wenda Xu, Liang Gong, Yixiang Huang, Zelin Zhao, Lujie Zhao, Binhao Chen, Haozhe Yang, Li Cao, and Chengliang Liu. "Human-Robot Interaction Oriented Human-in-the-Loop Real-Time Motion Imitation on a Humanoid Tri-Co Robot". In: *Proceedings of the International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE. 2018, pp. 781–786. DOI: 10.1109/ICARM.2018.8610806.

[276] Yanyu Xu, Yanbing Dong, Junru Wu, Zhengzhong Sun, Zhiru Shi, Jingyi Yu, and Shenghua Gao. "Gaze Prediction in Dynamic 360° Immersive Videos". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 5333–5342. DOI: 10.1109/CVPR.2018.00559.

[277] Yanyu Xu, Nianyi Li, Junru Wu, Jingyi Yu, and Shenghua Gao. "Beyond Universal Saliency: Personalized Saliency Prediction with Multi-task CNN". In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. IJCAI Organization, 2017, pp. 3887–3893. DOI: 10.24963/ijcai.2017/543.

[278] Yanyu Xu, Ziheng Zhang, and Shenghua Gao. "Spherical DNNs and Their Applications in 360° Images and Videos". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2022), pp. 7235–7252. DOI: 10.1109/TPAMI.2021.3100259.

[279] Fei Yan, Cheng Chen, Peng Xiao, Siyu Qi, Zhiliang Wang, and Ruoxiu Xiao. "Review of Visual Saliency Prediction: Development Process from Neurobiological Basis to Deep Models". In: *Applied Sciences* 12.1 (2022), p. 309. DOI: 10.3390/app12010309.

[280] Qin Yang, Yuqi Li, Chenglin Li, Hao Wang, Sa Yan, Li Wei, Wenrui Dai, Junni Zou, Hongkai Xiong, and Pascal Frossard. "SVGC-AVA: 360-Degree Video Saliency Prediction with Spherical Vector-Based Graph Convolution and Audio-Visual Attention". In: *Transactions on Multimedia* (2023), pp. 1–16. DOI: 10.1109/TMM.2023.3306596.

[281] Zhibo Yang, Lihan Huang, Yupei Chen, Zijun Wei, Seoyoung Ahn, Dimitris Samaras, Gregory Zelinsky, and Minh Hoai. "Predicting Goal-directed Human Attention Using Inverse Reinforcement Learning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 193–202. DOI: 10.1109/CVPR42600.2020.00027.

[282] Zhibo Yang, Sounak Mondal, Seoyoung Ahn, Gregory J. Zelinsky, Minh Hoai, and Dimitris Samaras. "Target-Absent Human Attention". In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 52–68. DOI: 10.1007/978-3-031-19772-7_4.

[283] H. Henny Yeung and Janet F. Werker. "Lip Movements Affect Infants' Audiovisual Speech Perception". In: *Psychological Science* 24.5 (2013), pp. 603–612. DOI: 10.1177/0956797612458802.

[284] Abolfazl Zaraki, Daniele Mazzei, Manuel Giuliani, and Danilo De Rossi. "Designing and Evaluating a Social Gaze-Control System for a Humanoid Robot". In: *IEEE Transactions on Human-Machine Systems* 44.2 (2014), pp. 157–168. DOI: 10.1109/THMS.2014.2303083.

[285] Gregory J. Zelinsky and James W. Bisley. "The What, Where, and Why of Priority Maps and Their Interactions with Visual Working Memory". In: *Annals of the New York Academy of Sciences* 1339.1 (2015), pp. 154–164. DOI: 10.1111/nyas.12606.

[286] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. "GaAN: Gated Attention Networks for Learning on Large and Spatiotemporal Graphs". In: *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*. https://auai.org/uai2018/proceedings/papers/139.pdf. 2018, pp. 339–349.

[287] Mengmi Zhang, Keng Teck Ma, Joo Hwee Lim, Qi Zhao, and Jiashi Feng. "Deep Future Gaze: Gaze Anticipation on Egocentric Videos Using Adversarial Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017, pp. 4372–4381. DOI: 10.1109/CVPR.2017.377.

[288] Shifeng Zhang, Xiangyu Zhu, Zhen Lei, Hailin Shi, Xiaobo Wang, and Stan Z. Li. "S^3FD: Single Shot Scale-Invariant Face Detector". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 192–201. DOI: 10.1109/ICCV.2017.30.

[289] Zhi Zheng, Eric M. Young, Amy R. Swanson, Amy S. Weitlauf, Zachary E. Warren, and Nilanjan Sarkar. "Robot-Mediated Imitation Skill Training for Children with Autism". In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 24.6 (2015), pp. 682–691. DOI: 10.1109/TNSRE.2015.2475724.

[290] Ce Zhou et al. "A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT". In: *CoRR* (2023). DOI: 10.48550/arXiv.2302.09419. arXiv: 2302.09419.

Erklärung der Urheberschaft

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Dissertationsschrift selbst verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Sofern im Zuge der Erstellung der vorliegenden Dissertationsschrift generative Künstliche Intelligenz (gKI) basierte elektronische Hilfsmittel verwendet wurden, versichere ich, dass meine eigene Leistung im Vordergrund stand und dass eine vollständige Dokumentation aller verwendeten Hilfsmittel gemäßder Guten wissenschaftlichen Praxis vorliegt. Ich trage die Verantwortung für eventuell durch die gKI generierte fehlerhafte oder verzerrte Inhalte, fehlerhafte Referenzen, Verstöße gegen das Datenschutz- und Urheberrecht oder Plagiate.

Hamburg, 15.01.2025

Ort, Datum

Unterschrift

Erklärung zur Veröffentlichung

Ich erkläre mein Einverständnis mit der Einstellung dieser Dissertation in den Bestand der Bibliothek.

Hamburg, 15.01.2025

Ort, Datum

Unterschrift