SPRING

# Deliverable D4.5: Emotional and robot acceptance analysis in relevant environments

Due Date: 31/01/2024

Main Author: BIU

Contributors: BIU, UNITN

Dissemination: Public Deliverable

This project has received funding from the European Union’s Horizon 2020 Research and Innovation Programme under Grant Agreement No. 871245.

DOCUMENT FACTSHEET

| | |
|---|---|
| **Deliverable** | D4.5: Emotional and robot acceptance analysis in relevant environments |
| **Responsible Partner** | BIU |
| **Work Package** | 4 |
| **Task** | Result of T4.3 on Broca Data. |
| **Version & Date** | 31/01/2024 |
| **Dissemination** | Public Deliverable |

CONTRIBUTORS AND HISTORY

| Version | Editor | Date | Change Log |
|---|---|---|---|
| 1 | UNITN | 15/12/2023 | First Draft |
| Final | UNITN,BIU | 15/01/2024 | Joint contributions and reviews towards final version |

APPROVALS

| | |
|---|---|
| **Authors/editors** | UNITN, BIU |
| **Task Leader** | BIU |
| **WP Leader** | UNITN |

# Contents

# Executive Summary

This deliverable, namely, D4.5, is part of WP4 of the H2020 SPRING project. The aim of this document is to present novel approaches for emotion recognition, gaze target detection, and automatic social acceptance of the robot. We present quantitative and qualitative results of the developed approaches related to T4.3 "*Multi-modal Affect and Robot Acceptance Analysis*".

Emotion recognition describes the ability to predict a person's emotional state in the robot's field view. The developed method leverages the power of unsupervised pre-training to mitigate the domain shift problem when deployed in different environments. The output of this module is a positive or negative emotion depending on the emotional state of the person. Furthermore, we implemented a single-microphone speech emotion recognition algorithm to evaluate the emotional state of people communicating with ARI using audio signals as well. Emotion recognition modules have been tested on several publicly available datasets, showing their robustness against several SOTA methods.

Gaze target detection aims to predict where the person is looking. The module here presented employs a novel end-to-end Transformer-based architecture able to simultaneously predict the *object class* and the *location* of the gazed-object, resulting in a comprehensive, explainable gaze analysis. Upon evaluation of the in-the-wild benchmarks, our method achieves state-of-the-art results on all metrics.

Social acceptance of the robot refers to the process of identifying and understanding the level of interaction, involvement, or connection between humans and robots in a given context [32]. Our proposed method concentrates on analyzing the gaze behavior of human agents. We leverage ARI's gaze target detection module to extract handcrafted features, drawing inspiration from [8], which has demonstrated promising results in analyzing multi-party conversations. Importantly, we evaluated the social acceptance module using data collected by the SPRING project in Broca Hospital.

The modules' source code are available in the SPRING repository[1]. As per European Commission requirements, the repository will be open to the public for at least four years after the end of the SPRING project.

[1] `https://gitlab.inria.fr/spring`

# 1 Introduction

This deliverable D4.5 is part of WP4 of the H2020 SPRING project, presenting the results of T4.3 "*Multi-modal Affect and Robot Acceptance Analysis*". The document presents frameworks for emotion recognition, gaze target detection, and automatic social acceptance with qualitative and quantitative results.

Facial expressions are essential to nonverbal communication and are major indicators of human emotions. Effective automatic Facial Emotion Recognition (FER) systems can facilitate comprehension of an individual's intention and prospective behaviors in Human-Computer and Human-Robot Interaction. Facial masks exacerbate the occlusion issue since these cover a significant portion of a person's face, including the highly informative mouth area from which positive and negative emotions can be differentiated. Conversely, the efficacy of FER is largely contingent upon the supervised learning paradigm, which necessitates costly and laborious data annotation. Our study centers on utilizing the reconstruction capability of a Convolutional Residual Autoencoder to differentiate between positive and negative emotions. The proposed approach employs Unsupervised Feature Learning and inputs facial images of individuals with and without masks as inputs. Our study emphasizes the transferability of the proposed approach to different domains compared to current state-of-the-art fully supervised methods. The comprehensive experimental evaluation demonstrates the superior transferability of the proposed approach, highlighting the effectiveness of unsupervised feature learning. Despite outperforming more complex methods in some scenarios, the proposed approach is characterized by relatively low computational expense. Furthermore, our framework for emotion recognition also incorporates information from audio-based emotion recognition, where a single-microphone speech emotion recognition algorithm can estimate the emotions of people talking with ARI.

Gaze target detection aims to predict the image location where the person is looking and the probability that a gaze is out of the scene. Several works have tackled this task by regressing a gaze heatmap centered on the gaze location; however, they overlooked decoding the relationship between the people and the gazed objects. We propose a Transformer-based architecture that automatically detects objects in the scene to build associations between every head and the gazed-head/object, resulting in a comprehensive, explainable gaze analysis composed of the gaze target area, gaze pixel point, the class, and the image location of the gazed-object. Upon evaluation of the in-the-wild benchmarks, our method achieves state-of-the-art results on all metrics.

Social acceptance detection involves identifying and comprehending the level of interaction, involvement, or connection between humans and robots within a specific context [32]. This encompasses the analysis of diverse cues, including verbal and non-verbal communication, gestures, facial expressions, and other social signals, to assess the extent to which a person actively engages with or responds to a robot [3]. In tackling this objective, our proposed method focuses on scrutinizing the gaze behavior of human agents. We utilize ARI's gaze target detection module to extract handcrafted features inspired by [8], which has exhibited promising results in analyzing multi-party conversations.

The rest of this deliverable is structured as follows: first, we describe the framework for emotion recognition using image and audio signals. Second, we describe the proposed module for gaze target detection, and third, we describe the social acceptance module. We conclude this deliverable with a summary of the results of the proposed modules.

# 2 Emotion Recognition

The emotion recognition module is capable of understanding the current emotional state of a person by predicting whether the face has a ***positive*** or ***negative*** attitude. We differentiate between these two discrete values mainly because of i) data availability, that is the available datasets provide valence values which determine whether a face is negative (unpleasant) or positive (pleasant), and ii) practicality, this scenario provides a more accessible setting for the deep learning architecture than training on several emotional classes thus increasing the model's performance. Additionally, we train the proposed model on masked and unmasked faces increasing the robustness of the model in case a masked person is approaching the robot. We decide to focus on the *transferability* of Facial Expression Recognition (FER) systems due to their unsupervised feature learning capabilities which provide a more robust adaptation to real-world applications, that is unsupervised feature learning does not require (labeled) re-training when the domain changes (e.g., hospital rooms, brightness), thus increasing the abilities of a robot employed in different social scenarios. To assess this, we examine the following cross-dataset settings and validate whether:

- an unsupervised feature learning-based approach (i.e., Ours) performs better than fully supervised methods (i.e., state-of-the-art (SOTA)) when the domains of the pre-training model and the classifier are the same, but the testing dataset is different, and
- an unsupervised feature learning-based method (i.e., Ours) performs better than fully supervised approaches (SOTA) when the pre-training domain is different from the domains that the classifiers are trained and tested on.

## 2.1 Implementation

The emotion recognition module is composed of two neural networks i) an autoencoder-based architecture used as a feature extractor (2.1-*top*) and ii) the classification head responsible for classifying whether the emotion is positive or negative (2.1-*bottom*).

**AutoEncoder (Unsupervised pre-training).** The employed Convolutional Autoencoder (AE), visualized in Fig. 2.1-*top* is composed of an encoder having three main residual blocks, each featuring three convolutions with 2D-kernels $3 \times 1$, $1 \times 3$, $3 \times 1$, ReLU as activation function and a max pooling operation. The input image of this network is of dimension $64 \times 64 \times 3$ while its output has a size of $2048$. The encoder employs residual connections particularly the first layer of each block is shared among the block itself and the skip connection, the output of the block is then summed with the output from the skip connection. The decoder is the transpose version of the encoder employing the same structure that takes as input the latent space from the encoder reconstructing the original image. Each decoder block uses a transpose-convolutional layer with ReLU and batch normalization.
This model is trained with Mean Squared Error (MSE):

$$\mathcal{L}_{\text{MSE}} = \frac{1}{2}\mathbb{E}_{\mathbf{X}\sim\mathcal{B}}\left[\|\mathbf{X} - \hat{\mathbf{X}}\|_F^2\right], \tag{2.1}$$

where $\mathbf{X}$ is the input image, and $\|\cdot\|_F$ denotes the Euclidean norm of the vector obtained after flattening the tensor $\mathbf{X}$. The MSE loss in (2.1) is minimized by using ADAM optimizer over mini-batches $\mathcal{B}$ and the reconstructed data are defined as:

$$\hat{\mathbf{X}} = \mathbf{D}_\theta \circ \mathbf{E}_\varphi(\mathbf{X}), \tag{2.2}$$

The MSE loss has the learnable parameters $\theta, \varphi$ updated by mini-batch gradient descent, where we estimate

$$\mathbb{E}_{\mathbf{x}\sim\mathcal{B}}\left[\mathcal{L}_{\text{MSE}}(\theta,\varphi)\right] = \mathbb{E}_{\mathbf{x}\sim\mathcal{B}}\left[\|\mathbf{x} - \mathbf{D}_\theta(\mathbf{E}_\varphi(\mathbf{x}))\|_F^2\right],$$

by averaging the MSE loss $\mathcal{L}_{\text{MSE}}$ over the mini-batch $\mathcal{B}$.

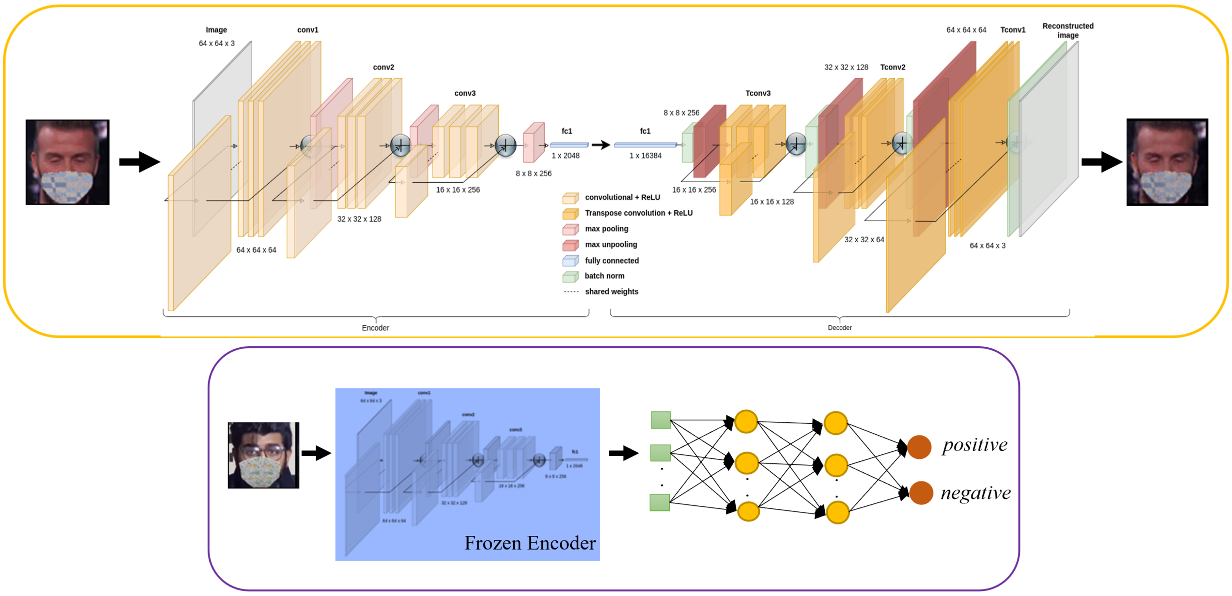

Figure 2.1: Proposed convolutional autoencoder trained with Mean Squared Error loss (top). Downstream task; positive/negative emotion classification learned with an MLP using the features extracted from the frozen encoder of our convolutional autoencoder trained unsupervised way (bottom).

**Classification Head.** Once the AE is trained with MSE, without using the labels of the data (aka unsupervised pretraining), following the representation learning literature, we freeze the AE and use it only to extract features for the training/testing data, which are used to train/test a linear classifier (see Fig. 2.1-bottom). The linear classifier is a Multilayer Perceptron (MLP) composed of two layers with parametrized ReLu as the activation function, trained to perform the classification of positive and negative emotions. The training of the MLP is performed with Focal Loss motivated by the fact that it could be able to better handle the class imbalance problem, if any.

## 2.2 Datasets

We employ three in-the-wild large-scaled FER datasets supplying valence annotations. These datasets are: AffectNet, Aff-wild2 and AFEW-VA. Notice that, FER datasets can show differences in terms of the range of the valence annotations as occurring between AffectNet/Aff-Wild2 versus AFEW-VA. However, the sign of the valence (i.e., whether it is positive or negative) is essential for FER. Given the cross-dataset analysis we perform in this study, it was necessary for us to apply discretization to the valence scores, and perform the FER as the classification of positive and negative emotions.

**AffectNet.** The AffectNet dataset is one of the largest image-based datasets for FER, including 287651 training, and 4000 validation images annotated manually. We use the validation set for model evaluation. The images from AffectNet have various sizes. The valence annotations have values in an interval between $-1$ and $+1$. We discretized the valence values in the way that the values smaller than zero refer to the negative class, and bigger than zero refers to the positive class.

**Aff-wild2.** The Aff-wild2 dataset is composed of 558 videos collected from Youtube including 458 subjects. The valence values are between $-1$ and $+1$. We discretized such values as described for AffectNet.

**AFEW-VA.** The AFEW-VA dataset contains 600 video clips selected from movies including indoor and outdoor scenes. This dataset provides a wide spectrum of facial expressions, captured in various circumstances with natural head pose movements, complex backgrounds, and severe occlusions. The valence annotations are per frame in a range between $-10$ to $10$. We discretized them such that the values smaller than zero refer to the negative class and bigger than zero refers to the positive class.

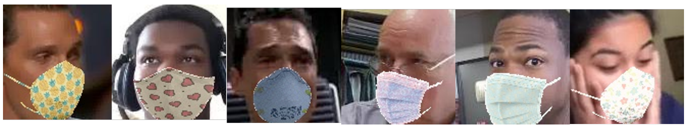

Figure 2.2: Samples of $F_m$ images obtained by applying Anwar and Raychowdhury's method [2] to the original unmaksed facial images.

| Dataset | Source | # of Training Images | | | # of Testing Images | | |
|---|---|---|---|---|---|---|---|
| | | Unmasked | Masked | Total | Unmasked | Masked | Total |
| 1 | AffectNet | 143825 | 130205 | 274030 | 1999 | 1903 | 3902 |
| 2 | Aff-Wild2 | 145920 | 105032 | 250952 | 31873 | 24624 | 56497 |
| 3 | AFEW-VA | 5658 | 7024 | 12682 | 631 | 781 | 1412 |

Table 2.1: Details of the datasets used in the experimental analysis.

**Facial masking method.** As explained earlier we increase the robustness of the system by training it on masked and unmasked facial images. The publicly available datasets lack masked faces with valence annotations thus we masked them using a facial masking method proposed in [2]. This method [2] provides five different mask types (surgical, N95, KN95, cloth, gas mask), in our setting we used all of them except the gas mask. Additionally, it provides 24 different patterns with different color intensities. When masking the datasets we randomly select the mask type, pattern, and color for each image in a dataset. We also randomly changed the intensity of the color. This resulted in 162 different facial masks. Since each mask type has multiple templates based on angle, they cover a wide range of face tilts, resulting in accurate masked facial images [2]. Still, we applied a manual visual inspection to discard the facial images of having the mask misplaced. Fig. 2.2 reports some images generated by the aforementioned method.

**Final datasets.** We used the above AffectNet, Aff-wild2 and AFEW-VA datasets and the masking method [2] to build the following datasets:

- **Dataset 1**: Its training and testing splits are composed of randomly selected 50% of the original (unmasked) images of AffectNet combined with the masked images generated from the other 50% of the dataset. The training and testing instances were kept the same as supplied by the original dataset.
- **Dataset 2**: After removing very similar faces (i.e., the ones appearing in the consecutive frames of the videos, and having the same emotion type) in the video clips of Aff-Wild2 to obtain an in-the-wild image-based dataset, we applied facial mask generation [2] to every remaining image. The instances of training and testing splits were kept as supplied by the original dataset. We ensured that if one type of image (between masked and unmasked) appears in training, its counterpart does not appear in the test set and vice versa. Moreover, the identities across training and testing splits are not overlapping.
- **Dataset 3**: We first removed the very similar faces (i.e., the ones appearing in the consecutive frames of the videos, and having the same emotion type) in order to obtain an in-the-wild image-based dataset from AFEW-VA. Following that, the mask generation [2] was applied to the remaining images. The facial images in which the mask generation misplaced the mask were discarded from the group of masked images, while their original correspondences were kept as unmasked images. Such images can be observed as relatively difficult ones, still, we argue that rather than omitting them totally from the evaluation (which is the case SOTA applies), involving them as unmasked images are still contributing. In that case, the same identities can appear in the training and testing splits while the head orientation, the emotion classes, and the image types masked or unmasked for the same identity are different.

Table 2.1 reports the sample distribution of the three datasets. Datasets 1-3 have slightly imbalanced numbers of masked and unmasked images in the training sets. This might bring an additional challenge for FER models. However, we did not manipulate the training splits to obtain balanced classes since imbalanced data is a frequently observed situation in real-world (FER) applications [9].

| | | F1 (↑) | | |
|---|---|---|---|---|
| Method | Feature Learning | Dataset 1 | Dataset 2 | Dataset 3 |
| Barros & Sciutti [7] | supervised | 48.8 | 26.9 | 75.2 |
| ResNet50 [21] | supervised | 66.2 | 41.2 | 79.2 |
| (Proposed) Know. Dist. | supervised | 70.3 | 44.1 | 83.8 |
| ViT [16] | supervised | 38.3 | 29.9 | 58.2 |
| ViT (w/ResNet50) [16, 29] | supervised | **71.0** | **65.7** | 87.7 |
| Proposed | unsupervised | 58.8 | 46.6 | **95.4** |

Table 2.2: Evaluation of the proposed method and the SOTA on Datasets (a) 1, (b) 2, and (c) 3 in terms of F1 score. The best results are indicated in **bold** and the second best results are given underlined. The symbol ↑ implies that a higher value is preferred.

## 2.3 Evaluation

We adopted several fully supervised State-Of-The-Art (SOTA) methods in order to compare their efficiency and effectiveness against the proposed approach. We implemented the FaceChannel [6] network with its last layer suitable for the binary classification task (i.e., softmax), and by using the search space applied by Barros & Sciutti [7] for the number of layers and unit per layer. We implemented a knowledge distillation approach as baseline between InceptionV3 (teacher) and MobileNet (student), this gives as a comparison on a low resource usage (as the developed AutoEncoder model). Another comparison approach is the usage of ResNet50 which is adapted by several SOTAs and, additionally, it employs residual connections as the proposed AE giving a direct comparison. We also included the Visual Transformers [16] into our comparisons. The effectiveness of the proposed method and SOTA are measured with F1-score $(F1)$.

### Results

Even though our main focus is to study the *transferability* of the proposed method with respect to other approaches, we first report a comparative study across our model and the prior art on the same-dataset setup to draw us an empirically validated comparative method out of all SOTA (see Table 2.2). The results highlight the better performance of ViT [16] used together with pre-trained ResNet50, on average. However, our approach surpasses ViT with ResNet50 when tested on datasets whose scalability is relatively smaller such as the case of Dataset 3. For relatively larger datasets such as Dataset 2, our model demonstrates the second-best performance after ViT with ResNet50 by surpassing all other fully supervised methods. Without using pre-trained ResNet50, ViT [16] underperforms in all datasets. The proposed Knowledge Distillation approach, overall, achieves better results compared to Barros and Sciutti [7] and ResNet50 even though its student component is much lightweight compared to both approaches.

### Cross-Dataset Analysis

The cross-dataset analysis includes two types of investigation. In the first one, we evaluate the models' performances when the datasets used in the pre-training and during the training of the classifier are the same, but the classifier's testing dataset is different. Such experiments are relevant given that there is often a domain gap between the training/validation data and the testing domain in real-world applications. Table 2.3 reports the results of this experiment showing that the majority of the time the proposed unsupervised feature learning-based model's transferability is superior to the proposed fully supervised knowledge distillation model. The only exception occurred when Dataset 1 was used as the training dataset and the testing is performed on Dataset 3. Still, even in the further case, the performance gap between the two models is lower than the former, i.e., the proposed unsupervised feature learning-based model surpasses the knowledge distillation. Overall, a drop in performance is possible due to the domain gap between the datasets. Particularly, training on either Dataset 1 or Dataset 2 significantly decreases the performance on Dataset 3 compared to both training and testing on Dataset 3.

The second type of cross-dataset analysis is to evaluate the models' performances when the pre-training dataset is different from the dataset the classifiers are trained and tested on. Such a setting simulates real-world applications in which one typically has models trained on one dataset (so-called pre-trained models) but further needs to be fine-tuned on another dataset whose distribution is the same as the testing dataset but different from the pre-training dataset.
We evaluated the performance of the knowledge distillation model in two settings:

| | | | Classifier | | |
|---|---|---|---|---|---|
| Method | Feature Learning | Pre-training Dataset | Training Dataset | Testing Dataset | F1 (↑) |
| Know. Dist. | supervised | - | 1 | 2 | 38.3 |
| Proposed | unsupervised | 1 | 1 | 2 | **44.7** |
| Know. Dist. | supervised | - | 2 | 1 | 44.0 |
| Proposed | unsupervised | 2 | 2 | 1 | **58.4** |
| Know. Dist. | supervised | - | 1 | 3 | **60.4** |
| Proposed | unsupervised | 1 | 1 | 3 | 53.2 |
| Know. Dist. | supervised | - | 2 | 3 | 46.8 |
| Proposed | unsupervised | 2 | 2 | 3 | **51.2** |

Table 2.3: Cross-dataset analysis when the testing dataset is different from the pre-training and training datasets. The best results of each metric are given in bold. Notice that the pre-training of the proposed method is unsupervised, i.e., without using the labels. The symbol ↑ implies that a higher value is preferred.

| | | | Classifier | | |
|---|---|---|---|---|---|
| Method | Feature Learning | Pre-training Dataset | Training Dataset | Testing Dataset | F1 (↑) |
| Know. Dist. (a) | supervised | 1 | 3 | 3 | 81.8 |
| Know. Dist. (b) | supervised | 1 | 3 | 3 | 69.5 |
| Proposed | unsupervised | 1 | 3 | 3 | **95.8** |
| Know. Dist. (a) | supervised | 2 | 3 | 3 | 84.7 |
| Know. Dist. (b) | supervised | 2 | 3 | 3 | 60.2 |
| Proposed | unsupervised | 2 | 3 | 3 | **94.5** |

Table 2.4: Cross-dataset analysis when the pre-training dataset is different from the dataset the classifier is fine-tuned and tested on. The best results of each metric are given in bold. Notice that the pre-training of the proposed method is unsupervised, i.e., without using the labels. See text for the description of (a) and (b). The symbol ↑ implies that a higher value is preferred.

- *(a)* The teacher model was trained on the pre-training dataset, and then the student network was trained on the same dataset. Furthermore, the student network was fine-tuned with the classifier's training dataset and tested with the classifier's test set. All layers of the student network were fine-tuned.
- *(b)* The teacher network was trained on the pre-training dataset, and then the student model was trained on the same dataset. Consequently, the student network was fine-tuned with the classifier's training dataset and tested with the classifier's test set. Only the last layer of the student was fine-tuned.

The corresponding results are given in Table 2.4. Herein, we used Dataset 1 and Dataset 2 in pre-training, and Dataset 3 was used for the classifier's training and testing. It is a common practice that model pre-training is performed on relatively larger datasets. In this vein, we did not perform pre-training on Dataset 3 given that it is the smallest dataset out of all (otherwise it is highly likely that a catastrophic forgetting would happen, therefore the transferability cannot be studied). Also in such cases, the proposed unsupervised feature learning-based approach surpasses the proposed knowledge distillation model for both settings (a) and (b), once again proving its better transferability. It is notable that pre-training on Dataset 1 slightly improves the results (from 95.4% to 95.8%) of our unsupervised feature learning-based method with respect to the one given in Table 2.2 (i.e., the same-dataset analysis) and pre-training on Dataset 2 improves the results of proposed knowledge distillation with respect to the same-dataset analysis (from 83.8% to 84.7%).

## 2.4 Audio-based Emotion Recognition

The emotional state of the person communicating with ARI can be inferred from the audio signal as well. For that, we have developed and implemented a single-microphone speech emotion recognition (SER) algorithm [33], which is a variant of the system proposed in [23]. In this scheme, the acoustic features are extracted from the audio utterances and fed to a neural network that consists of convolutional neural networks (CNN) layers, bidirectional long short-term memory (BLSTM) combined with an attention mechanism layer [4], and a fully connected layer. The architecture of the proposed SER is depicted in Fig. 2.3. Feature selection constitutes a pivotal facet of developing a resilient emotional system. In our work, we analyze several combinations of features, such as the mel-spectrogram feature and mel frequency cepstral coefficients (MFCC). The features used for the interactive emotional dyadic motion capture (IEMOCAP) database are described in Table 2.5.

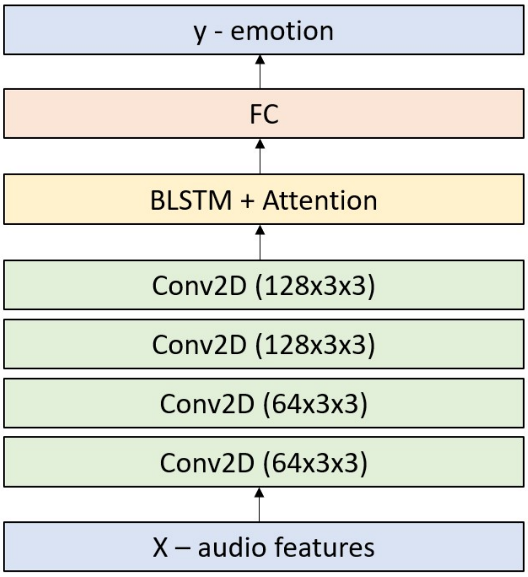

Figure 2.3: Architecture of the network.

Table 2.5: Proposed features for the IEMOCAP dataset

| Feature | Parameter Value | Description |
|---|---|---|
| MFCC | sr=16000, hop_length=512 | Spectrogram in mel-scale |
| MFCC Derivative | width=9, mode='interp', order=1, axis=-1 | Local estimates of the MFCC derivative |
| Spectral Centroid | sr=16000, hop_length=5120 | The frequency of the center of mass of the spectrum |
| Spectral Contrast | sr=16000, hop_length=512 | Ratio of the average power in the upper and lower quadrants |
| Spectral Bandwidth | sr=16000, hop_length=512 | 3dB Bandwidth |
| Spectral-roll off | sr=16000, hop_length=512 | Threshold frequency below which a specified percentage of the total spectral energy lies |
| zero-crossing rate (ZCR) | hop_length=512 | Zero-crossing rate of the time-domain signal |
| root mean square (RMS) | hop_length=512 | The root-mean-square value of the signal |

We evaluated our model using two popular databases, Ryerson audio-visual database of emotional speech and song (RAVDESS) [26] and IEMOCAP [10] datasets.

**Results for the RAVDESS dataset:** Given the similarities between the emotions 'calm' and 'neutral,' and considering that the number of utterances in 'neutral' is only half of those in the other emotion classes, a decision was made to merge both emotions under the label 'neutral.' In total, the network classified the data into seven different emotions and achieved a weighted accuracy of 80%. The results are also illustrated in the form of a confusion matrix shown in Fig. 2.4. It is noteworthy that the classes 'happy' and 'sad' exhibit significantly lower accuracy compared to the other classes.

**Results for the IEMOCAP dataset:** In many cases reported in the literature, only the emotions 'neutral,' 'happiness + excited,' 'sadness,' and 'anger' are used while training and evaluating the performance of a SER method on the IEMOCAP dataset since these classes are balanced in the number of their utterances. The emotions 'happiness' and 'excited' have a certain degree of similarity, and there are too few utterances of 'happiness.' Therefore, these emotions are combined together to create the 'happy' label with an approximately similar number of utterances as the other three emotions used for evaluation. In total, the network classified the data into four different emotions and obtained a weighted accuracy of 66%. The results are presented as a confusion matrix, as depicted in Fig. 2.5. It is important to note that the class 'happy' has lower accuracy than the other classes. Similar effects were reported in the literature, and it may indicate that the 'happy' emotion is more difficult to characterize. It remains a challenge to

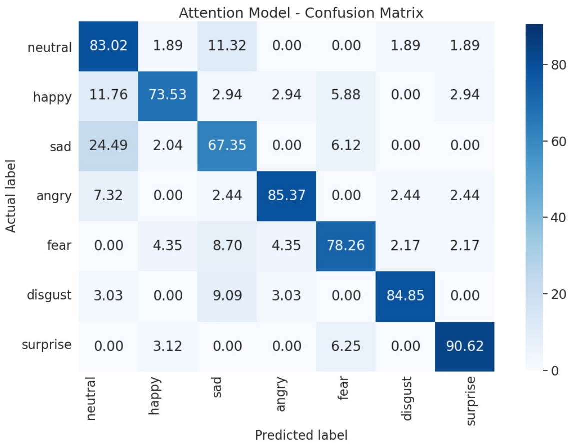

Figure 2.4: Confusion matrix of the results on RAVDESS dataset.

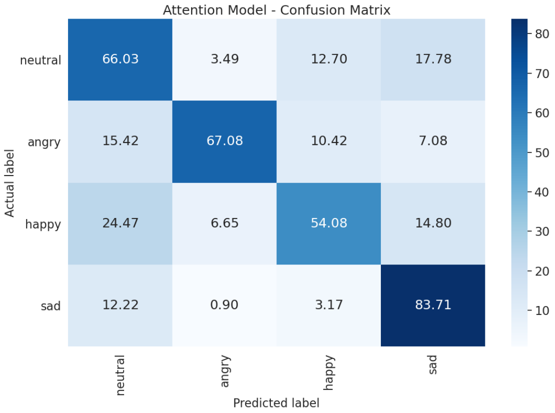

Figure 2.5: Confusion matrix of results on IEMOCAP dataset.

apply this algorithm to actual data recorded at the hospital.

# 3 Gaze Target Detection

Gazing is a powerful nonverbal signal, which indicates a person's visual attention and allows one to understand the interest, intention, or (future) action of people [18]. For this reason, gaze analysis has widely been used in several disciplines such as human-computer interaction [28, 37], neuroscience [14, 30], social and organizational psychology [11, 17], and social robotics [1] to name a few. The proposed method [35] is an end-to-end Transformer-based architecture. Given a scene image, we first extract all objects, including the ones classified as heads, with an *Object Detector Transformer*. Then, for each head, a gaze vector is predicted. Using this gaze vector, we build a *gaze cone* for each person individually, allowing the model to filter out objects not in a person's Field of View (FoV). Subsequently, a masked transformer (called *Gaze Object Transformer*) learns the interactions between the detected heads and objects, boosting the gaze target detection performance in terms of both heatmaps and gaze points (*i.e.*a single pixel in the scene). Furthermore, this architecture can predict whether a gaze target point is out of the frame.

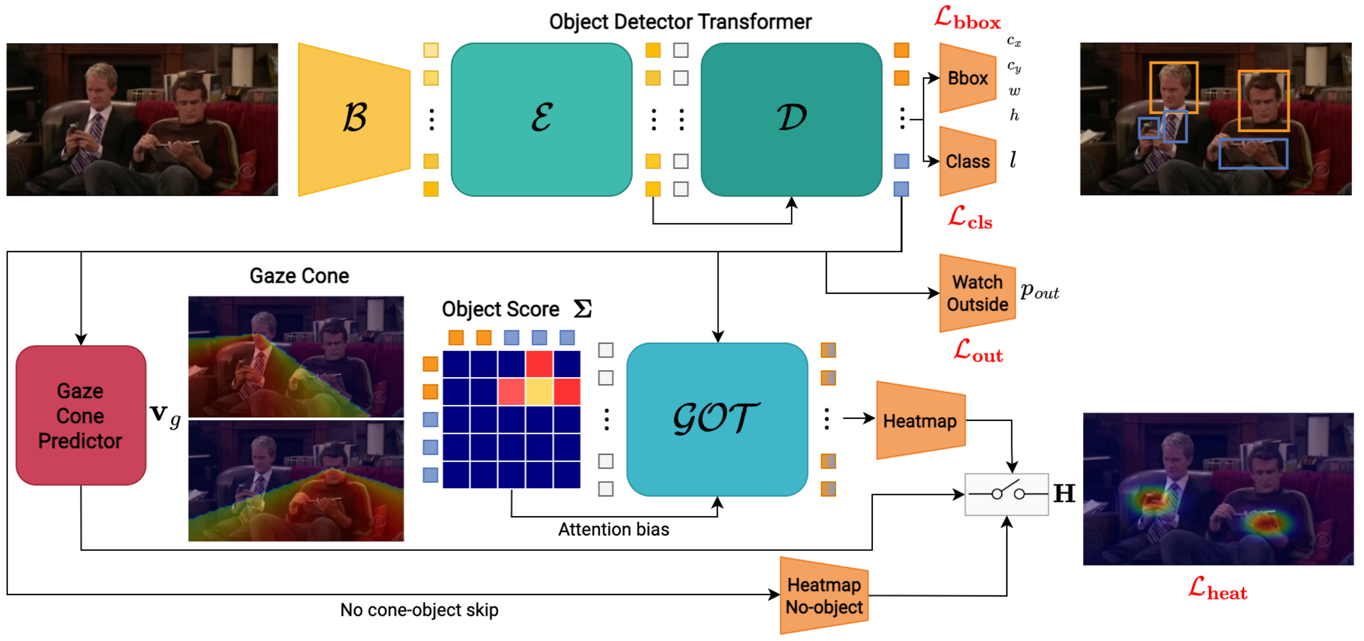

Figure 3.1: **Proposed method.** The encoder ($\mathcal{E}$) and decoder ($\mathcal{D}$) of the Object Detector Transformer operate on the features extracted by a backbone $\mathcal{B}$ to learn rich object features used to detect and localize objects (including heads) in the scene. Head features are used to build the **gaze cone**. Objects in the cone are extremely likely to be gaze-interesting. The *object score* matrix $\Sigma$ boosts attention scores in the *Gaze Object Transformer* ($\mathcal{GOT}$), whose output features are used to build the gaze heatmap. If no object lies in the cone, a skip-connection lets the network only predict the heatmap from head features.

## 3.1 Implementation

The proposed method is shown in Fig 3.1. Given an image, we first predict the set of objects $\mathbf{O} = \{(c_x, c_y, w, h, l)\}$ in it, where $(c_x, c_y, w, h)$ represent the center coordinates of a single object and its width and height, respectively, $l \in [0, CLS)$ is an object's label, and $CLS$ is the number of classes, including a special *no object* ($\emptyset$) class. To this end, after extracting the image features through a backbone $\mathcal{B}$, we use an *Object Detector Transformer* that reasons on the scene features with the encoder $\mathcal{E}$ and learns relevant object features with the decoder $\mathcal{D}$. Such features differentiate between heads $\mathbf{O_h}$ and other objects in the scene. For each head $\mathbf{O}^i_h$, we feed its features to the *Gaze Cone Predictor* to determine a gaze vector $\mathbf{v}^i_g$ that represents the gaze direction of the person. This gaze vector is used to build a gaze cone with an angle of $\alpha$ corresponding to the Field of View (FoV) and selectively maintain the objects inside the cone for each head. The *Gaze-Object Transformer* ($\mathcal{GOT}$) models the relationships between the detected objects and predicts the probability of them being the gaze target of any person, with a higher likelihood for the objects closer to the gaze vector. The gaze of each person is represented as a Gaussian heatmap $\mathbf{H}^i$ centered on the gaze point $\mathbf{p}^i_g$, and when no object is present inside the gaze cone, we use a *no cone-object skip* to compute the heatmap directly from the head features. We also use the head features to predict the probability of the gaze target being outside the frame. To sum up, our model consists of three major components: (a) Object Detector Transformer, (b) Gaze Cone Predictor, and (c) Gaze Object Transformer, which are described thoroughly in the following sections.

**Object Detector Transformer.** Given an RGB image $\mathbf{x} \in \mathbb{R}^{C \times H \times W}$, we aim to predict the bounding boxes and labels of objects. We start by extracting a feature map $\mathbf{f_b} \in \mathbb{R}^{C_b \times H_b \times W_b}$ with a convolutional backbone $\mathcal{B}$. Due to the high channel dimensionality, we linearly project the channel dimension to a lower space $C^{b'}$. We flatten the spatial dimensions and obtain $\mathbf{f}'_\mathbf{b} \in \mathbb{R}^{H_b W_b \times C'_b}$, which is fed to a transformer encoder $\mathcal{E}$ that enhances the coarse image features extracted by $\mathcal{B}$. $\mathcal{E}$ is designed as a stack of multi-head self-attention (MHSA) and feed-forward (FFN) layers. The projected output of $\mathcal{B}$, $\mathbf{f}'_\mathbf{b}$, forms the input queries $Q$, keys $K$, and values $V$ of $\mathcal{E}$. To retain the spatial information of the feature map, we add positional encodings for $Q$ and $K$. The output of the encoder, $\mathbf{f_e}$, forms the input $K$ and $V$ of the cross-attention module of the transformer decoder $\mathcal{D}$. $\mathcal{D}$ completes our Object Detector Transformer and introduces a multi-head cross-attention module to obtain object-relevant features. First, the decoder performs self-attention on a set of learnable embeddings $\mathbf{e_d} \in \mathbb{R}^{N \times C'_b}$, where $N$ is the maximum number of objects to be predicted. Like $\mathcal{E}$, we add the learnable embeddings $\mathbf{e_d}$ with a set of fixed positional embeddings. The output of the self-attention on $\mathbf{e_d}$ is then fed to a multi-head cross-attention module, where $\mathbf{e_d}$ are the queries, and $\mathbf{f_e}$ are the keys and values. The output features $\mathbf{f}_d$ of the transformer decoder are finally used by two multi-layer perceptrons (MLP) to predict the object bounding box (Bbox) and class, respectively.

**Gaze Cone Predictor.** An MLP takes as input the features of objects detected as heads $\mathbf{O}_h$ and estimates, for each of them, a 3D gaze vector $\mathbf{v}^i_g = (\theta^i, \phi^i, \rho^i)$. Each gaze vector uniquely identifies the orientation of the person's gaze with $\theta$, $\phi$, and $\rho$, which are the vector's polar angle, azimuthal angle, and magnitude, respectively. For each gaze vector $\mathbf{v}^i_g$, we design a 3D cone of angle $\alpha$ and apex ($c^i_x$, $c^i_y$, $c^i_z$) representing the FoV of a person, where $c^i_x$, $c^i_y$, and $c^i_z$ are the center coordinates of the head. The cone axis has the same direction as the gaze vector. The intensity of the cone, *i.e.*, the point saliency, is calculated as the cosine similarity between $\mathbf{v}^i_g$ and all vectors inside the cone starting from ($c^i_x$, $c^i_y$, $c^i_z$). In the 2D case, $\theta$ is not available, and we only have one angle $\phi$ and the magnitude $\rho$ for the gaze vector, while the 2D cone is still in the center of the apex but spans only in 2D instead of 3D. We adopt the discretized space of the same dimensionality of the predicted heatmap presented in [20], while we extend it to the 3D case, with $x$, $y$, and $z$ axis corresponding to the width, height, and depth of the image. For the 2D cone building, we follow the approach of [20], but we constrain the cone to be a fixed angle $\alpha$, which is in line with the FoV of human boundaries [22].

Formally, let $angle(\mathbf{v}_a, \mathbf{v}_b)$ be the absolute value of the angle between two vectors, and $\sigma(\mathbf{v}_a, \mathbf{v}_b)$ be the cosine similarity between two vectors $\mathbf{v}_a$ and $\mathbf{v}_b$ conditioned on the cone angle $\alpha$:

$$\sigma(\mathbf{v}_a, \mathbf{v}_b) = \begin{cases} \cos(\mathbf{v}_a, \mathbf{v}_b) & \text{if } angle(\mathbf{v}_a, \mathbf{v}_b) \leq \frac{\alpha}{2}, \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

The projected 3D gaze cone of a person $i$, $\mathbf{CD}^i_{3D}$, whose head center coordinates are $c^i_x, c^i_y, c^i_z$, and predicted gaze vector $\mathbf{v}^i_g$, is defined as:

$$\mathbf{CD}^i_{3D} = \{\sigma(\mathbf{v}^i_g, \mathbf{v}^{ijkl}_H)\} \forall j, k, l \in [0, w) \times [0, h) \times [0, d) \tag{3.2}$$

where $w$, $h$, and $d$ are the width, height, and depth of the space on which the 3D cone is computed, and $\mathbf{v}^i_H$ indicates the vectors in the discretized space starting from $(c^i_x, c^i_y, c^i_z)$.

The set of 3D cones $\mathbf{CD}_{3D}$ allows us to define the *object score* as a square matrix $\Sigma$ of size $N \times N$, where $N$ is the number of objects detected by the Object Detector Transformer. The object score matrix represents whether an object is in each person's visual cone and how close it is to their predicted gaze vector. Each row represents an object where the rows of objects not classified as heads are zero. For rows of *head* objects, the score for each other object is equivalent to the value of the gaze cone picked at the center coordinates of the object. When no object is in the gaze cone, the corresponding row becomes zero, and then we exploit the *no cone-object* skip to compute the gaze heatmap. The *object score* matrix $\Sigma$ is used by $\mathcal{GOT}$ as an additive bias in the attention module. The rationale behind the score matrix $\Sigma$ is to exploit the strong prior coming from the gaze vector and constrain the network to focus on relevant objects in the scene.

**Gaze Object Transformer** A stack of MHSA and FFN layers encodes a set of learnable embeddings $\mathbf{e}_g \in \mathbb{R}^{N \times C_b^{'}}$, where $N$ is the number of predicted objects. Unlike the object detector transformer's encoder, the multi-head self-attention includes an additive bias, *i.e.*our *object score* matrix $\Sigma$. Therefore, the new attention is defined as:

$$\text{BiasedAttention}(Q, K, V) = \text{softmax}\Big(\frac{QK^T + \Sigma}{\sqrt{d_k}}\Big)V \tag{3.3}$$

Additionally, we mask the learnable embeddings corresponding to objects not classified as heads. The masked features of the self-attention of $\mathcal{GOT}$ are the inputs to the cross-attention module. Likewise self-attention, the cross-attention module exploits the *object score* matrix as additive bias and performs binary masking on heads for $Q$ and other objects for $K$ and $V$. We exclude objects with low confidence prediction or classified as *no-object* ($\emptyset$).

The output features of the cross-attention form the input to the *heatmap* MLP to predict the gaze heatmap for each head. However, since we cannot assume that an object is always present, a second MLP (*heatmap no-object* in Fig. 3.1) predicts the heatmap from head features only when no object is inside the visual cone. The outputs of *heatmap* MLP and *heatmap no-object* MLP are fed to *a gated operator* that selects the heatmap based on the presence (or absence) of objects in the cone of each person. Finally, an additional *watch outside* MLP, only for head objects, predicts $\mathbf{p}_{out}$, the probability that the given head gaze lies outside the frame.

## 3.2 Datasets & Metrics

Our model is trained and tested on both GazeFollow [31] and VideoAttentionTarget [13] datasets. **GazeFollow** [31] is a large-scale *image* dataset containing over $122K$ images in total with more than $130K$ people. The test images include gaze and head location annotations performed by up to $10$ people for a single person in the scene. At the same time, the training set contains only one annotator's judgment indicating gaze and head locations. **VideoAttentionTarget** [13] is composed of YouTube *video* clips, each has a length of up to $80$ seconds. It includes $109,574$ in-frame and $54,967$ out-of-frame gaze annotations together with the head locations. Both the training and test sets contain one gaze annotation per person. Given that we do not use the *temporal* information in our model, we randomly select one image for every $5$ consecutive frames, allowing us to avoid overfitting. This setup is the same with SOTA [5, 19, 24, 34, 36].

**Evaluation Metrics.** We evaluate the performance of the proposed method in terms of **gaze target detection** and **object class detection and localization**. For the former task, we use all standard metrics [12, 13] described as follows. **AUC** assesses the confidence of the predicted gaze heatmap *w.r.t.* the gaze ground-truth. **Distance** (Dist.) is the $\mathcal{L}_2$ between the ground-truth gaze point and the predicted gaze location, which is the point with the maximum confidence on the gaze heatmap. In GazeFollow, it is a standard to declare both the minimum and average distances. **I/O gaze AP** is the average precision used to evaluate the *out-of-frame* probability of the gaze in VideoAttentionTarget. We use the standard metric **Mean Average Precision (mAP)** for object class detection and localization. In that case, a prediction is correct if the class label of the predicted bounding box and the ground truth bounding box are the same and the Intersection over Union ($IoU$) between them is greater than a $threshold$ value.

## 3.3 Results

Our method's gaze target detection performance is compared with the SOTA in Table 3.1. Recalling that the cropped head images and the head locations are required for traditional methods (i.e., SOTA except [36]) and these methods are evaluated when the ground-truth head locations are granted (referred to as "Head GT"), we proceed with the evaluation procedure of [36], summarized as follows. Tu et al. [36] employ additional head detectors to automatically obtain the head position given to the traditional models, providing their real-world application performance. We inherit the corresponding results from [36] and refer to them as "Real". For the methods whose "Real" results are not provided by [36], we obtain the results using RetinaFace [15] to detect head position. However, we can perform this only for the method whose code is publicly available: [34].

As we can see from the results, our method only with RGB data outperforms existing SOTA on all datasets for all metrics. Such a performance is important to emphasize since several SOTA perform relatively poorly even though they use multi-modalities [19, 24] or temporal data [13]. Notably, for VideoAttentionTarget [13] dataset, our method achieves better scores compared to many complex methods relying on several pretrained task-specific backbones (e.g., 2D-pose estimation) [5] or leveraging the temporal dimensionality of the data [27]. At the same time, both utilize

RGB and depth maps. Our better performance *w.r.t.* Transformer-based [36] is also conspicuous. Furthermore, when RGB and depth are considered, our method performance slightly improves on average. Recalling that we use depth information only during gaze cone production without requiring additional (pretrained) CNN to learn depth features as in [24, 34] or needing to detect the eyes as in [19], the corresponding results are noteworthy. Particularly, our minimum and average distance and mAP results are always the best whether or not others were evaluated within "Head GT" or "Real" settings. This shows that the proposed method is notably good at predicting if the gaze is located inside or outside the frame, the gaze heatmaps, and eventually, a single pixel gaze point that our model predicts per person is much closer to the ground truth-gaze point.

| Method | Modalities | Multiperson Gaze | GazeFollow [31] | | | | | | VideoAttentionTarget [13] | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | Distance ↓ | | | | *In frame* | | | | *Out of frame* | |
| | | | AUC ↑ | | Avg. | | Min. | | AUC ↑ | | Dist. ↓ | | AP ↑ | |
| | | | *Head GT* | *Real*$^\dagger$ | *Head GT* | *Real*$^\dagger$ | *Head GT* | *Real*$^\dagger$ | *Head GT* | *Real*$^\dagger$ | *Head GT* | *Real*$^\dagger$ | *Head GT* | *Real*$^\dagger$ |
| Random | | | 0.504 | 0.391 | 0.484 | 0.533 | 0.391 | 0.487 | 0.505 | 0.247 | 0.458 | 0.592 | 0.621 | 0.349 |
| Center | | | 0.633 | 0.446 | 0.313 | 0.495 | 0.230 | 0.371 | - | - | - | - | - | - |
| Fixed bias | | | - | - | - | - | - | - | 0.728 | - | 0.326 | - | 0.624 | - |
| Recasens et al. [31] | R | ✗ | 0.878 | 0.804 | 0.190 | 0.233 | 0.113 | 0.124 | - | - | - | - | - | - |
| Chong et al. [12] | R | ✗ | 0.896 | 0.807 | 0.187 | 0.207 | 0.112 | 0.120 | 0.830 | 0.791 | 0.193 | 0.214 | 0.705 | 0.651 |
| Lian et al. [25] | R | ✗ | 0.906 | 0.881 | 0.145 | 0.153 | 0.081 | 0.087 | 0.837 | 0.784 | 0.165 | 0.172 | - | - |
| Chong et al. [13] | R + T | ✗ | 0.921 | 0.902 | 0.137 | 0.142 | 0.077 | 0.082 | 0.860 | 0.812 | 0.134 | 0.146 | 0.853 | 0.849 |
| Fang et al. [19] | R + D | ✗ | 0.922 | - | 0,124 | - | 0.067 | - | 0.905 | - | 0.108 | - | 0.896 | - |
| Bao et al. [5] | R + D + P | ✗ | 0.928 | - | 0.122 | - | - | - | 0.885 | - | 0.120 | - | 0.869 | - |
| Jin et al. [24] | R + D | ✗ | 0.920 | - | 0.118 | - | 0.063 | - | 0.900 | - | 0.104 | - | 0.895 | - |
| Tonini et al. [34] | R + D | ✗ | 0.927 | 0.894 | 0.141 | 0.165 | - | - | 0.940 | 0.894 | 0.129 | 0.182 | - | - |
| Qiaomu et al. [27] | R + D + T | ✗ | 0.934 | - | 0.123 | - | 0.065 | - | 0.917 | - | 0.109 | - | 0.908 | - |
| Tu et al. [36] | R | ✓ | - | 0.917 | - | 0.133 | - | 0.069 | - | 0.904 | - | 0.126 | - | 0.854 |
| Tu et al. [36]* | R | ✓ | - | 0.915 | - | 0.104 | - | 0.055 | - | 0.891 | - | 0.229 | - | 0.809 |
| Our method | R | ✓ | - | **0.922** | - | 0.072 | - | 0.033 | - | 0.923 | - | **0.102** | - | **0.944** |
| Our method | R + D | ✓ | - | **0.922** | - | **0.069** | - | **0.029** | - | **0.933** | - | 0.104 | - | 0.934 |

Table 3.1: Evaluation on the GazeFollow [31] and VideoAttentionTarget [13] datasets. *Head GT* refers to using carefully labeled ground-truth head crops and locations in training and testing. *Real* indicated with $^\dagger$ is the implementation of [36], which applies an additional SOTA head detection network to predict the head location for real-world applications. We produce only [34]'s *Real* results (see text for details). $\star$ indicates our implementation. $R$, $D$, $T$, and $P$ stand for RGB, depth, temporal processing, and 2D-pose, respectively.

We visualize gaze heatmaps of our method and [36] in Fig. 3.2 on the GazeFollow dataset. Our predictions are more accurate compared to [36] in line with the quantitative results.

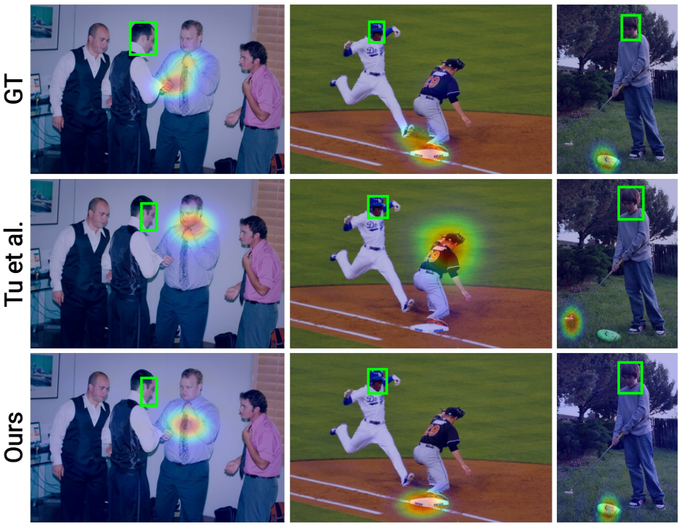

Figure 3.2: Qualitative results of our method (bottom) and Tu et al. [36] (middle) *w.r.t.* the ground-truth (top). For simplicity, we show only one person's gaze.

# 4 Automatic Social Acceptance Detection

Detecting human-robot engagement involves recognizing and understanding the level of interaction, involvement, or connection between people and robots in a specific situation [32]. This entails scrutinizing various cues like spoken and unspoken communication, gestures, facial expressions, and other social signals to gauge how actively engaged or responsive someone is to a robot [3]. To tackle this, our proposed method zeros in on analyzing how people look at things. We make use of ARI's previously discussed gaze target detection module to pull out specific features, taking inspiration from [8], which has shown promising outcomes in understanding conversations involving multiple people.

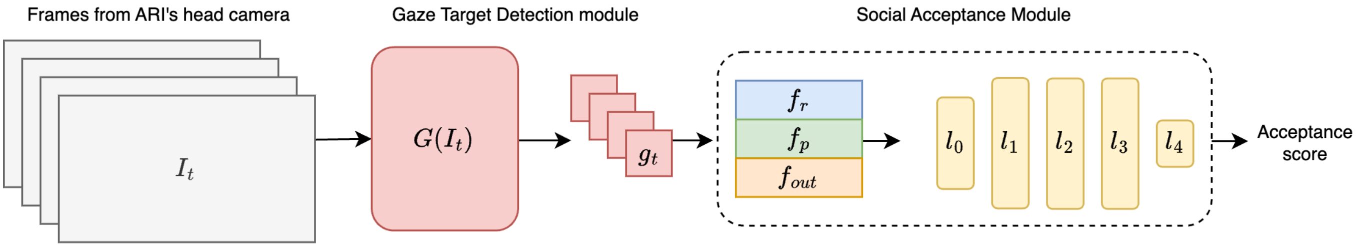

Figure 4.1: **Proposed pipeline** for automatic robot acceptance.

Our proposed pipeline for automatic robot acceptance is shown in Fig 4.1. Given a set of frames from ARI's head camera, we feed them to the gaze target detection module $G$ (Sec. 3) to produce the gaze heatmaps $g$, one per each frame and person. Given a set of gaze heatmaps and the information of the gaze location in the scene detected by our module, we define a gaze vector, which is a discretization of gaze location at each video frame. The instances of that vector can be as follows:

- Total number of frames where a person is looking at the robot $f_r$.
- Total number of frames where a person is engaging with another person in front of the robot $f_p$.
- The number of frames where a person is looking outside the field of view of the ARI head camera ($f_{out}$).
- The ratios between $f_r$ and $f_p$, $f_p$ and $f_{out}$, and $f_{out}$ and $f_r$.

These features are used to train an MLP composed of 5 densely connected layers, with hidden dimensions set to $32$ and ReLU activation between each layer, along with the engagement annotation for the corresponding video clip. The output of the social acceptance module is a continuous value from $0$ to $1$, with $0$ being *no-acceptance* and $1$ *acceptance*.

Our social acceptance module was trained using videos obtained from the SPRING project at Broca Hospital. The acceptance ground truth for these videos was annotated by a psychology student. Specifically, we annotated five videos featuring a total of seven individuals, totaling $55k$ annotated frames. Among these frames, $40k$ were labeled as *acceptance*, while $15k$ were labeled as *no-acceptance*.

| | No engagement | Engagement |
|---|---|---|
| No engagement | **84** | 506 |
| Engagement | 13 | **1234** |

Table 4.1: Confusion matrix of our social acceptance model. Rows are prediction, columns are ground-truth.

Tab. 4.1 displays the confusion matrix for the evaluated performance of the proposed social acceptance module on the Broca dataset. Additionally, when assessed on the Broca dataset, our method attains an accuracy of $71\%$.

# 5 Conclusions

This deliverable presented three modules for (a) emotion recognition, (b) gaze target detection, and (c) automatic social acceptance of the robot. We presented both qualitative and quantitative results of the approaches, which showed remarkable results. Emotion Recognition predicts whether a person has *positive* or *negative* emotion, the module leverages the power of unsupervised pre-training to mitigate the domain shift problem when deployed on different environments, moreover, the method achieves SOTA performance on various datasets. Furthermore, we propose a single-microphone speech emotion recognition algorithm to evaluate the emotional state of people interacting with ARI. Gaze Target Detection aims to predict where the person is looking. The module employs a novel end-to-end Transformer-based gaze target detector able to simultaneously predict the *object class* and the *location* of the gazed-object. The latter is advantageous *w.r.t.* existing methods as it improves explainability. Extensive experiments validate both modules showing comparable or better performance than SOTA methods. Lastly, we proposed a module for social acceptance of the robot that relies on ARI's gaze target detection module to extract handcrafted features to learn an *acceptance score*. The evaluation results of the social acceptance module on the Broca dataset reveal promising results.

This project has received funding from the European Union's Horizon 2020
Research and Innovation Programme under Grant Agreement No. 871245.

# Bibliography

[1] Henny Admoni and Brian Scassellati. Social eye gaze in human-robot interaction: a review. *Journal of Human-Robot Interaction*, 6(1):25–63, 2017.

[2] Aqeel Anwar and Arijit Raychowdhury. Masked face recognition for secure authentication, 2020.

[3] Salvatore M Anzalone, Sofiane Boucenna, Serena Ivaldi, and Mohamed Chetouani. Evaluating the engagement with social robots. *International Journal of Social Robotics*, 7:465–478, 2015.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[5] Jun Bao, Buyu Liu, and Jun Yu. Escnet: Gaze target detection with the understanding of 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14126–14135, 2022.

[6] Pablo Barros, Nikhil Churamani, and Alessandra Sciutti. The facechannel: a fast and furious deep neural network for facial expression recognition. *SN Computer Science*, 1(6):1–10, 2020.

[7] Pablo Barros and Alessandra Sciutti. I only have eyes for you: The impact of masks on convolutional-based facial expression recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1226–1231, 2021.

[8] Cigdem Beyan, Francesca Capozzi, Cristina Becchio, and Vittorio Murino. Prediction of the leadership style of an emergent leader using audio and visual nonverbal features. *IEEE Transactions on Multimedia*, 20(2):441–456, 2017.

[9] Cigdem Beyan and Robert Fisher. Classifying imbalanced data sets using similarity based hierarchical decomposition. *Pattern Recognition*, 48(5):1653–1672, 2015.

[10] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. IEMOCAP: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335–359, 2008.

[11] Francesca Capozzi, Cigdem Beyan, Antonio Pierro, Atesh Koul, Vittorio Murino, Stefano Livi, Andrew P Bayliss, Jelena Ristic, and Cristina Becchio. Tracking the leader: Gaze behavior in group interactions. *Iscience*, 16:242–249, 2019.

[12] Eunji Chong, Nataniel Ruiz, Yongxin Wang, Yun Zhang, Agata Rozga, and James M. Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[13] Eunji Chong, Yongxin Wang, Nataniel Ruiz, and James M Rehg. Detecting attended visual targets in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5396–5406, 2020.

[14] Kim M Dalton, Brendon M Nacewicz, Tom Johnstone, Hillary S Schaefer, Morton Ann Gernsbacher, Hill H Goldsmith, Andrew L Alexander, and Richard J Davidson. Gaze fixation and the neural circuitry of face processing in autism. *Nature neuroscience*, 8(4):519–526, 2005.

[15] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[17] S Gareth Edwards, Lisa J Stephenson, Mario Dalmaso, and Andrew P Bayliss. Social orienting in gaze leading: a mechanism for shared attention. *Proceedings of the Royal Society B: Biological Sciences*, 282(1812):20151141, 2015.

[18] Nathan J Emery. The eyes have it: the neuroethology, function and evolution of social gaze. *Neuroscience & biobehavioral reviews*, 24(6):581–604, 2000.

[19] Yi Fang, Jiapeng Tang, Wang Shen, Wei Shen, Xiao Gu, Li Song, and Guangtao Zhai. Dual attention guided gaze target detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11390–11399, 2021.

[20] Anshul Gupta, Samy Tafasca, and Jean-Marc Odobez. A modular multimodal architecture for gaze target prediction: Application to privacy-sensitive settings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5041–5050, 2022.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Ian P Howard, Brian J Rogers, et al. *Binocular vision and stereopsis*. Oxford University Press, USA, 1995.

[23] Che-Wei Huang and Shrikanth Shri Narayanan. Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition. In *IEEE international conference on multimedia and expo (ICME)*, pages 583–588, 2017.

[24] Tianlei Jin, Qizhi Yu, Shiqiang Zhu, Zheyuan Lin, Jie Ren, Yuanhai Zhou, and Wei Song. Depth-aware gaze-following via auxiliary networks for robotics. *Engineering Applications of Artificial Intelligence*, 113:104924, 2022.

[25] Dongze Lian, Zehao Yu, and Shenghua Gao. Believe it or not, we know what you are looking at! In *Asian Conference on Computer Vision*, pages 35–50. Springer, 2018.

[26] Steven R Livingstone and Frank A Russo. The Ryerson audio-visual database of emotional speech and song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. *PloS one*, 13(5):e0196391, 2018.

[27] Qiaomu Miao, Minh Hoai, and Dimitris Samaras. Patch-level gaze distribution prediction for gaze following. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 880–889, 2023.

[28] Borna Noureddin, Peter D Lawrence, and CF Man. A non-contact device for tracking gaze in a human computer interface. *Computer Vision and Image Understanding*, 98(1):52–82, 2005.

[29] Namuk Park and Songkuk Kim. How do vision transformers work? In *ICLR*, 2022.

[30] Keith Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin*, 124(3):372, 1998.

[31] Adria Recasens, Aditya Khosla, Carl Vondrick, and Antonio Torralba. Where are they looking? In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[32] Hanan Salam and Mohamed Chetouani. Engagement detection based on mutli-party cues for human robot interaction. In *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 341–347, 2015.

[33] Dalia Sherman, Gershon Hazan, and Sharon Gannot. Study of speech emotion recognition using BLSTM with attention. In *31st European Signal Processing Conference (EUSIPCO)*, Helsinki, Finland, September 2023.

[34] Francesco Tonini, Cigdem Beyan, and Elisa Ricci. Multimodal across domains gaze target detection. In *Proceedings of the 2022 International Conference on Multimodal Interaction*, pages 420–431, 2022.

[35] Francesco Tonini, Nicola Dall'Asen, Cigdem Beyan, and Elisa Ricci. Object-aware gaze target detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21860–21869, 2023.

[36] Danyang Tu, Xiongkuo Min, Huiyu Duan, Guodong Guo, Guangtao Zhai, and Wei Shen. End-to-end human-gaze-target detection with transformers. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2192–2200. IEEE, 2022.

[37] Lijun Yin and Michael Reale. Real time eye tracking for human computer interaction, November 11 2014. US Patent 8,885,882.