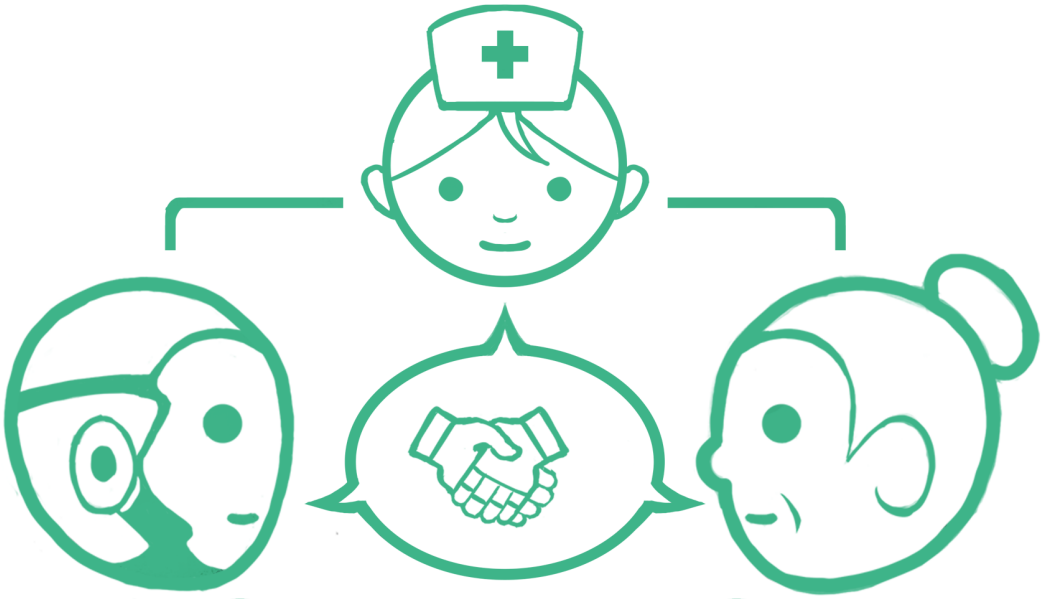


Deliverable D4.3: Multi-modal behaviour recognition in realistic environments

Due Date: 31/05/2022

Main Author: UNITN

Contributors: BIU, INRIA

Dissemination: Public Deliverable

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

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

DOCUMENT FACTSHEET

Deliverable	D4.3: Multi-modal behaviour recognition in realistic environments
Responsible Partner	UNITN
Work Package	WP4: Multi-Modal Human Behaviour Understanding
Task	T4.2: Individual & Group Behaviour Recognition; T4.1: Describing Humans
Version & Date	31/05/2022
Dissemination	Public Deliverable

CONTRIBUTORS AND HISTORY

Version	Editor	Date	Change Log
1	UNITN	11/05/2022	First Draft
2	BIU	12/05/2022	BIU contribution was added
3	INRIA	13/05/2022	INRIA contribution was added
4	BIU	26/05/2022	The document was reviewed by Sharon Gannot
5	UNITN	27/05/2022	The document was reviewed by Elisa Ricci
6	INRIA	09/06/2022	The document was reviewed by Matthieu Py
7	INRIA	14/06/2022	The document was modified according to the feedback of Elisa Ricci, Sharon Gannot and Matthieu Py
8	INRIA	22/06/2022	The document was reviewed by Matthieu Py
8	INRIA	23/06/2022	The document was modified according to the feedback of Matthieu Py

APPROVALS

Authors/editors	UNITN, INRIA, BIU
Task Leader	UNITN
WP Leader	UNITN

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

Contents

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

Executive Summary

This deliverable, namely, D4.3 is part of WP4 of the H2020 SPRING project. The main aim of this document is to present the first prototype implementation of tools for (a) single target behaviour recognition and (b) group-level behaviour analysis. Additionally, we also present an audio-based emotion recognition method, which can further be merged with behaviour recognition models in order to understand the behaviors and emotions simultaneously.

Regarding single task target behaviour recognition, this document includes the description of a novel method, which performs unsupervised domain adaptation as well as presenting a new spatial transformer. The results tested on several action recognition datasets, confirm the effectiveness of this method. The code is available in: `https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/single_target_da_action_recognition.git`. This method will be integrated to ARI and tested on human-robot interaction scenarios as a future work.

We approach the group-level behaviour analysis, in this deliverable, from two perspectives: (a) gaze target detection and (b) group detection. Gaze target detection is related to T4.1: Describing Humans since face detection is performed within that task and the faces of person is input to the proposed method to further detect the gaze target of that person. This module will predict the gaze target of each person in the scene captured by the head camera of ARI. Gaze target detection is an important module to understand who is interacting with whom and consequently, one can provide better group detection and group activity detection module. We present a novel multimodal method to address gaze target detection, showing better results than the state-of-the-art. This method was already integrated into ARI. It will be tested on wider set of scenarios for human-robot interaction as well as will be tested on the dataset collected by the SPRING project. The gaze target detection code can be found in: `https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/gaze-cnn.git`.

Group detection stands for assigning people that were detected in the surrounding of ARI to conversational groups. By correctly detecting groups, one can detect which group is interacting with ARI, identify the people ARI has to consider to interact with. Moreover, group detection improves the human aware navigation as it allows to model group spaces, which are used to navigate ARI to correctly join the groups and to avoid interrupting others while joining. As a solution, SPRING partners use the Graph-Cuts for F-formation (GCFF) algorithm by Setti et al. [43].

The last but not the least, this deliverable also includes a speech emotion recognition (SER) algorithm which is a variant of the system proposed in [19]. In the proposed 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, and a fully-connected layer. The proposed method is tested on publicly available datasets, showing promising results. The code can be found in: `https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/ser`.

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

1 Contributions

1.1 Introduction

This deliverable **D4.3** is part of **WP4** of the H2020 SPRING project, targeting to present the result of T4.2: Individual & Group Behaviour Recognition, while also utilising the results of T4.1: Describing Humans.

In this context we present:

- The first prototype implementation of tools for **single target behaviour recognition** (see Sec. 1.2),
- The first prototype implementation of tools for **group-level behaviour analysis**, which is addressed though **gaze target detection** (see Sec. 1.3.1), integrating information about faces derived from visual data (T4.1: Describing Humans),
- The first prototype implementation of tools for **group-level behaviour analysis** through **proxemics features** (Sec. 1.3.2)),
- The first prototype implementation for audio-based emotion recognition (Sec. 1.4). This can be further merged with behaviour recognition models in order to understand the behaviors and emotions simultaneously.

1.2 Single-target Behaviour Recognition

In this document, we include the description and code of the method we developed for single-target domain adaptation for action recognition. The details of this method is given in Sec. 1.2.1. We present our set of experiments in Sec. 1.2.2. Finally, we report our results in Sec. 1.2.2. The code of this approach is publicly available on GitLab[1].

1.2.1 Proposed method

Our proposal tackles the problem of Unsupervised Domain Adaptation (UDA) for single-target action recognition. Given a source dataset $\mathcal{S} = \{X_i^S, y_i^S\}_{i=1}^{N_S}$ of videos and associated annotations, and an unlabelled target dataset $\mathcal{T} = \{X_i^T\}_{i=1}^{N_T}$, where $X_i \in \mathcal{X}$ and $y \in \mathcal{Y}$, $\mathcal{Y} = \{1, 2, \ldots, K\}$ (K denotes the number of action categories), we aim to learn a function $F_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ with parameters θ that maps an input video X to a class label y and perform well on the target data. Note that this is not a trivial task, since source and target data are sampled from two different distributions, $P^S(X) \neq P^T(X)$. To handle this problem, we propose a novel approach that combines two main components: (a) a spatio-temporal transformer architecture and (b) a novel distribution alignment scheme derived from the $IBprinciple$ [48].

An overview of our proposed method is shown in Fig. 1.1. We propose a two-stage training pipeline where the model is first trained with source data and subsequently adapted using source and target data. Our model is defined as $F_\theta = C \circ H$, where H represents a video transformer encoder [44] and C represents a linear classifier. H is composed of two main parts, a spatial transformer H_s that extracts frame-level feature representations and a temporal transformer H_t that aggregates the frame-level features to produce video-level representations. In particular, H_s is the vision transformer ViT [12], whereas H_t is a simple multi-layer transformer as in [50]. An auxiliary MLP projection head P is also used in the second stage. Finally, the complete model also has a queue Q that is responsible for keeping the most recent feature representations of source data. The two main phases of our approach are described in Annex 3.1.

[1] **GitLab repository**: `https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/single_target_da_action_recognition.git`

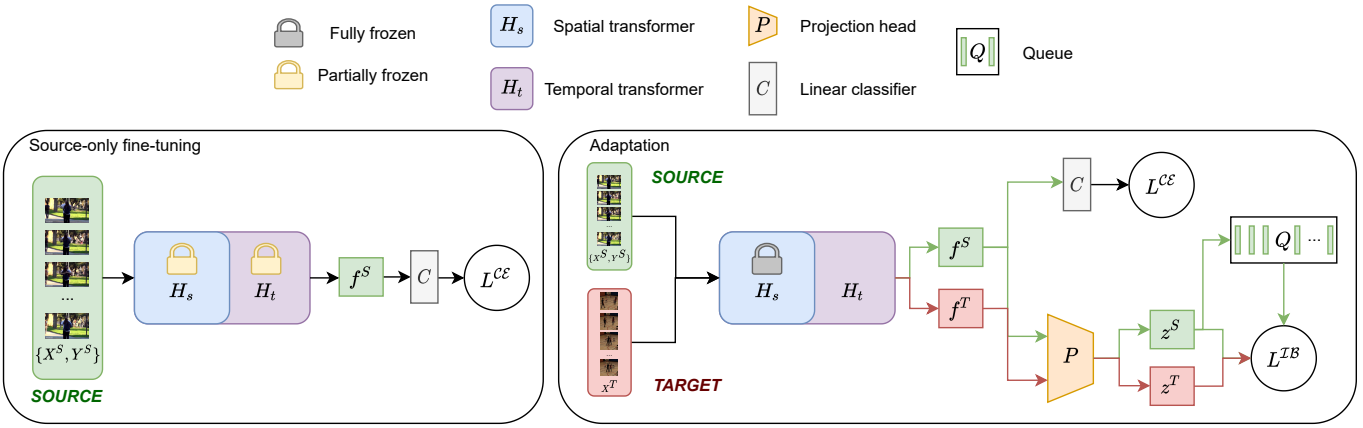


Figure 1.1: **Overview of our method** Our approach is articulated in two steps. In phase 1 (left), source data are fed to a video transformer H (composed of a spatial transformer H_s and a temporal transformer H_t) followed by a classifier C. The overall model is fine-tuned with a supervised cross-entropy loss $L^{\mathcal{CE}}$ similarly to [32]. In phase 2 (right), the weights of H_s are frozen, while H_t is fine-tuned. Source and target data are fed to the backbone and the proposed IB-based loss $L^{\mathcal{IB}}$ performs domain alignment, while $L^{\mathcal{CE}}$ further trains the action classifier. A queue Q is added in order to increase the number of source instances considered while computing $L^{\mathcal{IB}}$.

1.2.2 Experiments and Results

Datasets. We conduct an extensive evaluation of our approach on two benchmarks for UDA in action recognition, namely *HMDB↔UCF* [5] and *Kinetics→NEC-Drone* [6]. The former setting comprises videos from the *HMDB51* [23] and *UCF101* [45] action recognition datasets, which both contain real videos downloaded from Youtube. In this case, the domain shift is therefore present, but limited. *Kinetics→NEC-Drone*, consists of videos from the large scale *Kinetics* dataset [4], that contains sequences from Youtube, and the *NEC-Drone* [6] dataset, which consists of video sequences taken from moving drones in an indoor environment. Furthermore, the video sequences of *NEC-Drone* comprise high-resolution frames (1920x1080), and the action is often relegated to the corner of the frame and in many cases, the view is extremely slanted. Understandably, this setting is characterised by a significantly more challenging domain shift that consequently induces all the tested state-of-the-art methods to perform poorly on the original data. To alleviate this problem, we employed a pre-processing step exploiting a pretrained YOLO-based [40] human detection model using AlphaPose [15] to identify and locate the human actor(s) and then crop around the humans with a minimal resolution of 224x224. In the following experiments, we used the cropped version of the *NEC-Drone* dataset.

Baselines. We compare our results with those obtained by state-of-the-art methods for UDA in video action recognition, namely TA^{3}N [5], TCoN [36], SAVA [52] and CO^{2}A [10]. For a fair comparison, the results of TA^{3}N are also reported after replacing their ResNet backbone with I3D. Also, as a transformer-based baseline, we report the results obtained by replacing our proposed loss with three different domain alignment strategies, namely: a Maximum Mean Discrepancy (MMD) domain alignment component [30], an adversarial approach relying on a domain classifier as in [17] and a Maximum Classifier Discrepancy (MCD) based component [41]. MCD aligns domains by employing task-specific decision boundaries that maximise the discrepancy between the output of two distinct classifiers to detect target samples lying far from source support and minimise the discrepancy of the transformer, so it learns how to produce target features closer to source support. The adversarial-based approach [17] consists of adding an MLP-based domain classifier that is responsible for predicting the domains of the instances given their video-level feature representations. We added a target cross entropy based on the pseudo-labels to all baselines was we found that this improved performance.

Results. Table 1.1 presents the results on *HMDB↔UCF*. Along with the scores achieved with our proposed method, we report the ones obtained by previous approaches on the same settings. As it can be observed, all transformer-based models significantly outperform previous methods (except for CO^{2}A [10]) in both directions, suggesting that transformer-based methods are more robust to domain shift even without any domain adaptation strategy. In particular, we achieve an accuracy of 96.8% and 92.3% in the two directions, outperforming the current best competitor (CO^{2}A [10]) by 1% and 4.5%, respectively. Results also show that our method outperforms MMD with a transformer-based architecture. Also, the proposed MCD and adversarial-based baselines are outperformed (in just one of the two directions for the case of MCD). Finally, we report, as upper bounds, the scores obtained with the supervised version of the method, i.e., the case where ground truth target labels are used instead of pseudo-labels to compute the

Table 1.1: Results on *HMDB↔UCF*.

Method	Encoder	H→U	U→H
Baselines			
Source only [5]	ResNet	71.7	73.9
DANN [18]		76.3	75.2
JAN [31]		74.7	79.6
AdaBN [25]		72.2	77.4
MCD [41]		73.8	79.3
TA^{3}N [5]		81.8	78.3
Target only [5]		*82.8*	*94.9*
TCoN [36]	2D/3D CNN	89.1	87.2
Source only [52]	I3D	88.8	80.3
TA^{3}N [5]		90.5	81.4
SAVA [52]		91.2	82.2
CO^{2}A [10]		95.8	87.8
Target only [52]		*95.0*	*96.8*
Transformer-based			
Source only	Transformer	93.7	86.9
MMD [30]		96.5	87.9
MCD [41]		**97.2**	87.9
Adversarial [17]		96.6	87.6
UDAVT (ours)		96.8	**92.3**
UDAVT (ours) - *supervised*		*97.2*	*94.4*
Target only		*97.9*	*95.8*

Table 1.2: Results on *NEC-Drone*.

Method	Encoder	Top-1 Acc
Baselines		
Source only	Resnet	15.8
TA^{3}N [5]		28.0
Source only	I3D	32.0
TA^{3}N [5]		44.7
SAVA [52]		42.5
CO^{2}A [10]		45.8
Transformer-based		
Source only	Transformer	29.4
MMD [30]		54.4
MCD [41]		38.1
Adversarial [17]		40.8
UDAVT (ours)		**65.3**
UDAVT (ours) - *supervised*		*78.1*
Target only		*82.9*

cross-correlation matrix. Table 1.2 reports the scores obtained on the *Kinetics→NEC-Drone* benchmark. This setting corresponds to a more significant domain shift since the target video sequences are shot by drones in a specific indoor environment. For this reason, it is easy to observe that the absolute value of all the reported scores is significantly lower when compared to the accuracy obtained in the previous benchmarks. However, the results clearly show how the transformer-based approaches strongly outperform the baselines achieving a score of 65.3%, which is about 17 points more than the best competitor. In addition, the proposed loss achieves more than 10 points when compared to the MMD-based transformer. The gap is wider when it comes to the MCD and adversarial-based baselines, which are outperformed by 27 and 25 points. These experiments show that (i) the transformer-based backbone proves effective when applied to cases where a higher domain shift is present and (ii) the proposed alignment method addresses the domain gap more efficiently leading to a significant increase in accuracy on the target domain.

Next Steps. The proposed method will be tested on human-robot interaction scenarios particularly on the data collected by the Spring Project.

1.3 Group-level Behaviour Recognition

This deliverable presents various solutions, which eventually result in group-level behavior recognition. We first describe the gaze target detection module (Sec. 1.3.1), which is developed to detect the gaze of each individual in the scene captured by the head camera of ARI. This module allows us to better understand who is interacting with whom and consequently, we can provide a better group detection module. On the other hand, we also present a F-formation detection methodology in 1.3.1, which relies on the position of the individuals as well as their body orientations.

1.3.1 Gaze Target Detection

Human-beings have a remarkable capability to detect the gaze direction of others, understand whether a person is gazing them, follow other's gaze to identify their target and determine the attention of others [7]. However, automatically performing and quantifying these remains as a challenging problem. Gaze target detection (also referred as *gaze-following* [8, 16]) is to inferring where each person in the scene (2D or 3D) is looking at [39, 47, 28].

Proposed Method. We aim to predict the gaze of a person in an RGB scene image, captured by the head camera of ARI. To do so, we propose a method, whose inputs are: (a) an RGB scene image, which contains the field of the view of head camera of ARI; (b) an RGB head image, which is cropped from the RGB scene image, corresponding to the person whose gaze is going to be estimated, and (c) a scene depth image obtained from monocular depth estimation network of Ranftl et al. [37] (which will be further replaced by our method, whose development and optimization in process). The output of the proposed method is the gaze heatmap, i.e., a 1-channel 2D matrix whose peak value represents the gaze coordinates. In other words, we can predict the image coordinates of each person's gaze and predict the probability that each person is gazing at an object inside or outside the image. The proposed method is illustrated in Fig. 1.2. The code of this approach is publicly available on GitLab[2].

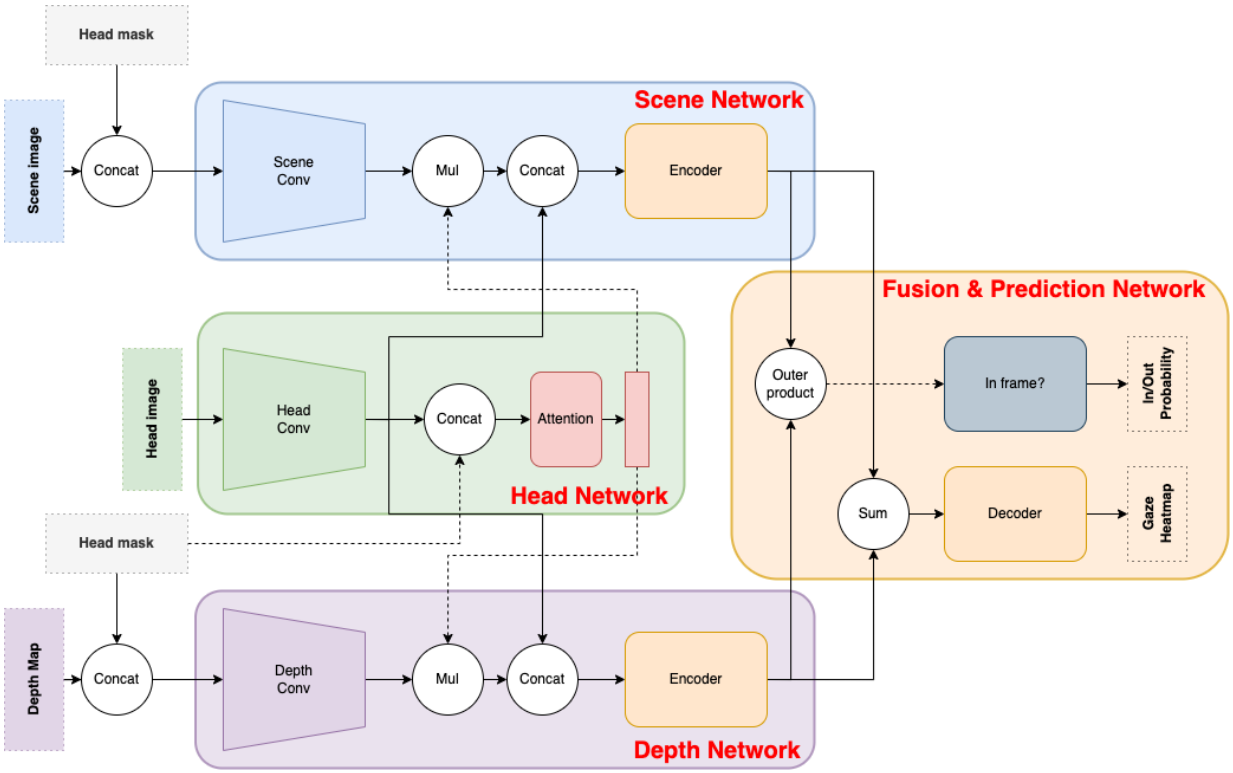


Figure 1.2: An illustration of the proposed method. Mul, Concat, Outer product and Sum stand for multiplication, concatenation, channel-wise outer product [46], summation operators, respectively.

As seen, the proposed network is composed of: *scene and depth network* processes the RGB and depth of the scene, while the *head network* processes the head independently and produces an attention map that is then multiplied by the RGB and depth embeddings. The *fusion and prediction module* concatenates scene, depth, and head features to obtain the two final outputs of the proposed method: a 2D gaze heatmap that encodes the region in which gaze happens, and the probability of the gaze target being inside or outside the scene. The details of each component are given in Annex 3.2.

Evaluation on Realistic Environments. The proposed method is evaluated on two benchmark datasets: GazeFollow [38] and VideoAttentionTarget [8]. We follow the standard training/testing split of each dataset for fair comparisons with the prior art. GazeFollow dataset [38] includes more than 120K images from various classification and detection datasets (i.e., SUN [53], COCO [27], Actions-40 [55], PASCAL [14], and Places [59]), with more than 130K annotations of head locations and the corresponding gaze points. VideoAttentionTarget [8] is a collection of 1331 video clips from various sources on YouTube. The annotations include more than 160K frame-level head bounding boxes and 110K gaze targets inside the scene. The following metrics were adopted to evaluate the performance of the proposed model in line with the prior art [38, 8, 16]. **Heatmap Area Under Curve (AUC %)** [20] is to asses the confidence of the predicted heatmap with respect to the ground-truth. **Average distance (Avg.Dist.)** stands for the Euclidean distance between the predicted gaze location and the ground-truth gaze point.

[2]**GitLab repository**: `https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/gaze-cnn.git`

Table 1.3: Evaluation on benchmark datasets. The best results (the higher the AUC and the lower the average distance ($Avg.Dist.$) is better) are shown in bold.

	GazeFollow [38]		**VidAttTrgt [8]**	
	AUC	Avg.Dist.	AUC	Avg.Dist.
[38]	87.8	0.190	-	-
[7]	89.6	0.187	83.0	0.193
[26]	90.6	0.145	-	-
[8]	92.1	0.137	86.0	0.134
[16]	92.2	**0.124**	90.5	**0.108**
Ours	**92.7**	0.141	**94.0**	0.129
Human	92.4	0.096	92.1	0.051

Results. We compare our approach with several prior art in Table 1.3. Our method achieves better results compared to all of them, and becomes SOTA for all datasets in terms of AUC. It surpasses even the human performance in GazeFollow [38] and VideoAttentionTarget [8] datasets. In particular, its relative performance improvements in VideoAttentionTarget [8] dataset is obtrusive. In terms of Avg.Dist., our method falls behind [16] while performing better than other methods.

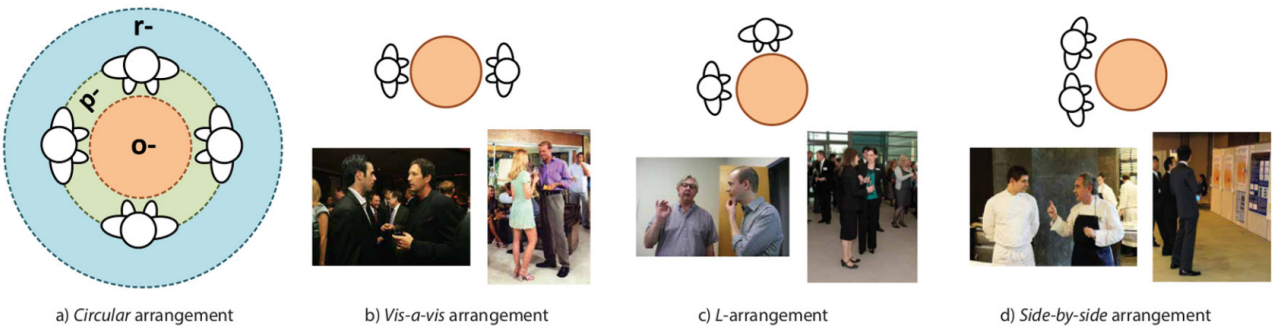
Next Steps. The proposed method will be tested on human-robot interaction scenarios particularly on the data collected by the Spring Project.

1.3.2 Group Detection

Group detection assigns people that where detected in the surrounding of ARI to conversational groups. Detection of such groups is an essential element of SPRING due to two reasons. 1) It defines the groups with which ARI can interact and identifies the people ARI has to consider when interacting with a group. 2) It improves the human aware navigation as it allows to model group spaces. Group spaces are used to correctly join groups and to avoid interrupting them by navigating, for example, through them.

As a solution we use the Graph-Cuts for F-formation (GCFF) algorithm by Setti et al. [43]. It is based on the concept of F-formations by Kendon [22]. F-formations describe the arrangement of individuals of a group with respect to their positions and orientations. They are defined by three social spaces: o-space, p-space, and r-space (Fig. 1.3). The o-space is an empty space around which the individuals of a group are positioned. The p-space is the space in which the members of a group are positioned. The r-space is the space outside the group.

GCFF identifies the o-spaces of different groups. It uses for this the concept of transactional segments [9]. These describe the space in front of a person in which it can perform actions and its sensing (vision and hearing) are best. GCFF identifies areas in which the transactional segments of people overlap (see Algorithm 1). These overlapping spaces are potential o-spaces of groups in which people interact with each other. The details of this method is given in Section 3.3.



a) *Circular* arrangement b) *Vis-a-vis* arrangement c) *L*-arrangement d) *Side-by-side* arrangement

Figure 1.3: Overview of different F-formations that describe group formations. Each formation is defined by a o-space (orange), the space in front of all people of a group where they interact, a p-space (green) where they are located, and a r-space (blue), the space surrounding the group. Figure taken from [43].

Algorithm 1 GCFF algorithm for group detection

Initialise with $O_{G_i} = TS_i \forall i \in [1, \ldots, n]$
old_cost $= \infty$
while $J(O_G, TS) <$ old_cost **do**
  old_cost $\leftarrow J(O_G, TS)$
  run graph cuts to minimize cost Eq. 3.6
  for $\forall g \in [1, \ldots, M]$ **do**
    if g is not empty **then**
      update $O_G \leftarrow \frac{1}{|G|} \sum_{i \in G} TS_i$
    end if
  end for
end while

Conclusion. The GCFF algorithm detects groups of people and their o-space center. Its hyper-parameters D, the distance between a person and its transactional segment center, and σ, used to control the MDL that punishes the number of detected F-formations, are chosen manually based on experiments performed with ARI. Empirical results of the GCFF detection performance can be found in [43].

1.4 Audio-based Emotion Classification

The ability to perceive the emotional state of a person is of crucial importance in the design of socially pertinent robots, as it supports the higher level decision on the preferred way to proceed with the interaction between the robot and the human.

We are proposing a SER algorithm which is a variant of the system proposed in [19]. In the proposed scheme, the acoustic features are extracted from the audio utterances and fed to a neural network that consists of CNN layers, BLSTM combined with an attention mechanism layer, and a fully-connected layer. We evaluated our model using Ryerson audio-visual database of emotional speech and song (RAVDESS) [29] and interactive emotional dyadic motion capture (IEMOCAP) [3] databases achieving weighted accuracy, of 80% and 64%, respectively. The code of the SER is publicly available on [3].

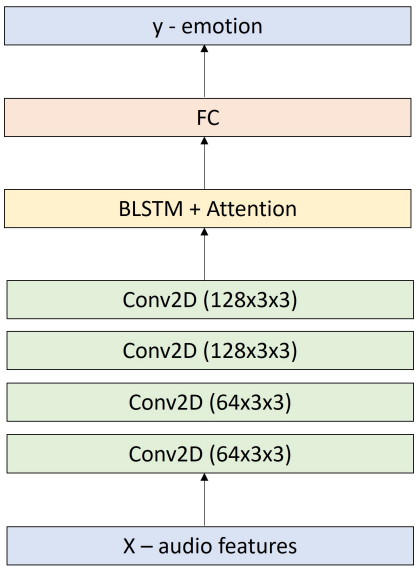


Figure 1.4: Architecture of the network.

Method. Our starting point is the SER model presented in [19] with several modifications adopted from [34]. We use a convolutional long- short term deep neural network (CLDNN) model that comprises three parts. The first part of the scheme consists of 4 layers of a Conv2D network with a kernel size of 3×3 and stride of 1. The first two layers

[3] https://gitlab.inria.fr/spring/wp4_behavior/non-integrated-contributions/ser

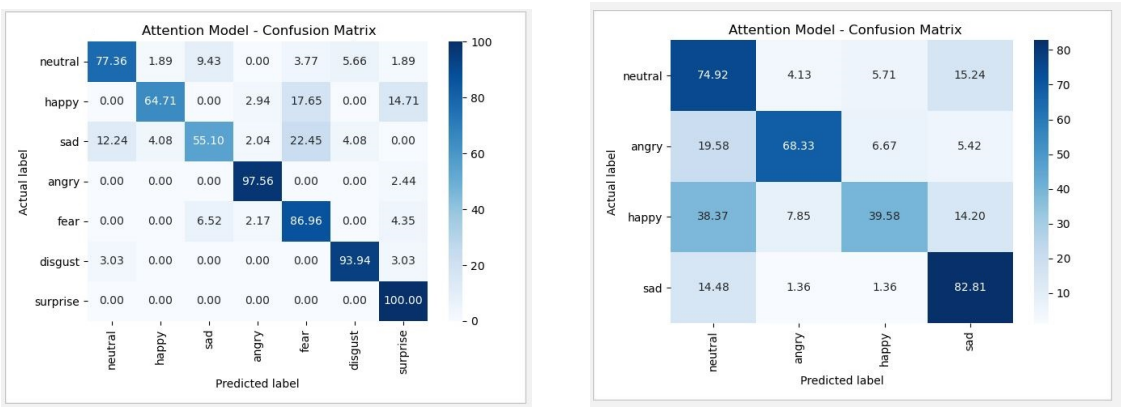


Figure 1.5: Results on RAVDESS database (left), IEMOCAP database (right).

comprise 64 filters and the following two layers comprise 128 filters each. The layers combine BatchNormalization, relu activation, MaxPooling2D with size of 4×4, stride of 4×4 and a dropout rate of 0.2. The two dimensions of the convolutional layers corresponds to the time and frequency axes, respectively. The second part of the scheme is a BLSTM+attention layer. The BLSTM is implemented with a size of 256, and outputs the hidden states, the hidden forward state, and the hidden backward state that are fed into an attention layer. We implemented the attention mechanism using the architecture proposed by Bahdanau et al. [2]. The third and last part of the system is an fully-connected (FC) layer with a softmax activation function, indicating the probabilities of each emotion. The entire architecture of the proposed SER is depicted in Fig. 1.4.

The model was trained with a categorical cross-entropy loss function,. The overall number of parameters was about 1.180M for the RAVDESS database and about 1.4420M for the IEMOCAP database.

After trying different combinations of features, the best results for the RAVDESS database were obtained by using the mel-spectrogram feature [42] with 128 mel bands. For the IEMOCAP database we found that the best results are obtained by concatenating several common audio features, namely mel frequency cepstral coefficientss (MFCCs) and their derivatives, spectral properties (centroid, contrast, bandwidth, and roll-off), zero-crossing rate (ZCR), and root mean square (RMS).

Results. RAVDESS [29] is a publicly available audio-visual database (we use only the audio modality). The database comprises 24 actors, evenly distributed between male and female speakers, each uttering 60 English sentences. Overall, there are 1,440 utterances in total, expressing 8 different emotions: 'sad', 'happy', 'angry', 'calm', 'fearful', 'surprised', 'neutral', and 'disgust'. In the experiments, we combined the emotions 'calm' and 'neutral', as they are very similar. All utterances are transcribed in advance. Consequently, the emotions are more artificially expressed as compared with spontaneous conversation. Another drawback of the database is the small number of utterances. The network classified the data into seven different emotions and obtained a weighted accuracy of 80%. The results are also presented as a confusion matrix as depicted in Fig. 1.5. It is important to note that the classes 'happy' and 'sad' have significantly lower accuracy than the other classes.

The IEMOCAP [3] database comprises approximately 12 hours of audio-visual data, including video, speech, motion capture of the face, and text transcriptions (again we only use the audio modality). The database consists of conversations of two people that are either improvised or played according to a pre-determined transcript that was chosen to evoke different emotions. The database consists of 10 actors, evenly distributed between male and female speakers. The utterances are classified into 9 different emotions: 'neutral', 'happiness', 'sadness', 'anger', 'surprise', 'fear', 'disgust', 'frustration', and 'excited'. Following [33, 35, 34, 54, 13, 57, 51], only the emotions 'neutral', 'happiness + excited' (denoted 'happy'), 'sadness', and 'anger' are used while training and evaluating the performance of a SER. The network classified the data into four different emotions and obtained a weighted accuracy of 64%. The results are represented as a confusion matrix as depicted Fig. 1.5. It is important to note that the class 'happy' has lower accuracy than the other classes. similar issues were reported in [33, 58, 35, 34] and it may indicate that the 'happy' emotion is more difficult to characterize.

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

2 Conclusions

We have presented the first prototype implementation of tools for (a) single target behaviour recognition and (b) group-level behaviour analysis. We also present an audio-based emotion recognition method, which can further be merged with behaviour recognition models in order to jointly understand behaviors and emotions.

Regarding the single task target behaviour recognition, we have introduced a novel method, which is based on spatial transformers and apply unsupervised domain adaptation, which is important to handle the domain-shift problem, that can occur when the trained and test domains are coming from different distributions. The performance of this module was evaluated in publicly available datasets, which are corresponding to realistic environments. This method will be integrated into ARI in the future and will be tested on human-robot interaction scenarios, most importantly, importantly on the dataset collected by the SPRING project.

Regarding group-level behaviour analysis, we have presented two methods. One concerns gaze target detection and the other concerns group detection. In the future, both will serve as important cues to understand and detect the group membership and its joint behavior. Gaze target detection, which is implemented as a multi-modal network using depth and scene images, was tested on publicly available datasets. This method was already integrated into ARI. It will be further improved to handle the domain-shift problem, and then will be tested on human-robot interaction scenarios as well, and most importantly on the dataset collected by the SPRING project. Group detection module is adapted from a state-of-the-art method. It will particularly, allow us to correctly detect the conversational groups, detect which group is interacting with ARI, identify the people ARI has to consider interaction with. Also, we expect that it will improve the human-aware navigation, as it should allow to model group spaces, which are used to navigate ARI to correctly join the groups and to avoid interrupting others while joining. The group and individual activity detection modules will be improved by including the detected objects in the scene.

The SER method presented in this study uses the acoustic features extracted from the audio utterances. It is composed of a neural network that consists of CNN layers, BLSTM combined with an attention mechanism layer, and a fully-connected layer. This method showed promising results on publicly available datasets. As future work, it will be integrated into ARI and tested on human-robot interaction scenarios, and most importantly on the dataset collected by the SPRING project.

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

3 Annex

3.1 Details of Single-Target Behavior Recognition

The two main phases of our approach are described as follows.

Phase I: Source-only fine-tuning The training process of this phase starts from a model H pretrained on the Kinetics dataset [21] and consists of fine-tuning the entire model F_θ using only the source data $\mathcal{S}$. As in [44], we consider 16 frames uniformly sampled to represent a source video X^S as input to F_θ. The first part of the model H_s divides each frame into 16x16 patches that are then projected into feature vectors. H_s consists of ViT, which receives as input the projected patches together with a classification token $[CLS]^S$ as in [11]. Each frame is processed individually, extracting feature representations $f^S_{H_s} = [CLS]^S$ that consists of the classification token linked to that specific frame. During the forward pass, $[CLS]^S$ will collect all important information from the image patches. The frame-level features $f^S_{H_s}$ are then forwarded through H_t together with a new classification token $[CLS]^T$ which, after processing, produces the video-level feature representations $f^S = [CLS]^T$. In this step, H_s and H_t are fine-tuned following the strategy proposed in [32], which consists of freezing all parameters except the positional encoding, the input embeddings, the classification tokens and the affine transformations inside the layer normalisations [1]. While [32] studied the problem of partially fine-tuning a transformer for handling different modalities, in this work, we show that this strategy can be successfully applied to the problem of domain adaptation. Finally, the video level features are then fed to a linear classifier C. The entire model is trained with a supervised cross-entropy loss $L^{\mathcal{CE}}$, defined as:

$$L^{\mathcal{CE}} = -\mathbb{E}_{(X,y)\in\mathcal{S}} \sum y_k \log \sigma(F_\theta(X)), \tag{3.1}$$

where σ is the softmax operation. Due to lack of space, the reader is referred to [44] and [12] for more details about the transformer architecture.

Phase II: Target Adaptation In this phase, the spatial transformer H_s is frozen, while the parameters of H_t are trained to exploit both labelled source and unlabelled target data. This choice is motivated by the need of reducing computational resources, while still performing adaptation at the temporal level. Freezing part of the model enables us to increase the batch size, which is fundamental for the proposed domain alignment strategy (Eqn. 3.3). To train our model, 16 frames are sampled from both source X^S and target X^T videos, as in the previous phase. Video-level feature representations f^S and f^T are then produced for videos of both domains. Subsequently, the temporal features of source videos f^S are provided as input to the linear classifier C. To perform adaptation, we rely on the Information Bottleneck (IB) principle [48, 49]. Fig. 3.1 shows how the IB principle is applied to our problem. First, we assume that there exists a domain transformation $g \sim \mathcal{G}$ that maps a target instance X^T to a source instance $\overline{X}^S$ that has the same label. Unlike [56], which considers that one instance is mapped to a perturbed version of the same instance via some type of data augmentation, we map a single target instance X^T to multiple different $\overline{X}^S$ in the same iteration. We experimentally show that this is indeed beneficial since by increasing the number of source instances via the usage of a queue, and consequently the number of pairs, we observed a large boost in performance. As annotations are not provided for the target domain, we resort to pseudo-labels for matching source and target instances. The model H maps $\overline{X}^S$ to the feature representation $\overline{f}^S$. According to the IB principle, we want the model H to learn a representation $\overline{f}^S$ which encodes as much information as possible about the original instance X^T. This objective is carried out by maximising the Mutual Information $I(\overline{f}^S, X^T)$. Then, the second objective consists of minimising $I(\overline{f}^S, \overline{X}^S)$ to make the model H invariant to the transformation of the sample X^T into a different domain. The overall loss function can be written as:

$$L^{\mathcal{IB}} = I(\overline{f}^S, \overline{X}^S) - \beta\, I(\overline{f}^S, X^T) \tag{3.2}$$

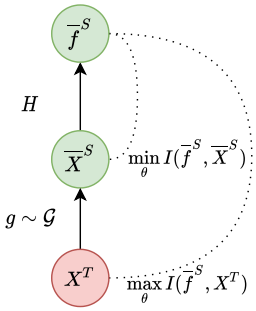


Figure 3.1: Information Bottleneck diagram showing the proposed flow of information to perform adaptation.

Since optimising for mutual information in a high dimensional space is difficult, previous works have proposed different ways to approximate Eqn. 3.2. In this work, we derive a loss function similar to that used in the Barlow Twins method [56], where it was proved that, under certain conditions, Eqn. 3.2 can be approximated as:

$$L^{\mathcal{IB}} = \sum_{i}^{d}(1 - C_{ii})^2 + \lambda \sum_{i}^{d}\sum_{j \neq i}^{d}(C_{ij})^2, \tag{3.3}$$

where $\boldsymbol{C}$ is a cross-correlation matrix computed over a batch of B data obtained through a feature extractor $\{z_1, \ldots, z_B\}$ and their corresponding transformed version $\{z'_1, \ldots, z'_B\}$, where i is the feature index and d is the total number of features. Each element of $\boldsymbol{C}$ is defined as $C_{ij} = \frac{\sum_b z_{i,b} z'_{j,b}}{\sqrt{\sum_b (z_{i,b})^2}\sqrt{\sum_b (z'_{i,b})^2}}$, where z_i and z'_i are mean centred. While in [56] the cross-covariance matrix is computed considering the original images and their augmented versions, we propose to re-purpose it for domain alignment using corresponding samples across the two domains. The loss in Eq.3.3 is a trade-off between two objectives, the first term that pushes the learned representation to be domain invariant and a second term that decorrelates the different components of the embedding. To build $\boldsymbol{C}$, we introduce a projection head P, similar to the one in [56], mapping f^S and f^T to z^S and z^T. Then, each source instance representation z_i^S is paired with all target instance representations z_j^T where the label of instance i and the pseudo-label of instance j are equal. Note that the same instance i or j can appear in more than one pair. We also introduced a queue Q to keep recent z^S, effectively increasing the number of possible instances that are paired with z^T in the minibatch. After forming this list of pairs, the cross-correlation matrix can be computed between the source instances and the target instances of all pairs. This process makes the model invariant to instances of different domains and enables tackling the domain adaptation setting. Our final loss, introducing a weighting factor α, is then defined as follows:

$$\mathcal{L} = L^{\mathcal{CE}} + \alpha \mathcal{L}^{\mathcal{IB}}. \tag{3.4}$$

3.2 Details of Gaze Target Detection

Each component of gaze target detection method are explained as follows.

Head Network. Given the RGB scene image S_i, we crop the head H_i of the person of interest. The head image is processed by the head network's backbone (ResNet50) that maps the original representation $\mathbf{H}_i$ into a feature embedding $\mathbf{e}_i^h$. Such features are average pooled and processed by a set of linear layers that outputs an attention map. The outcome of the attention map $\mathbf{attn}_i^h$ is multiplied by the scene and depth feature embeddings.

Scene Network. The scene network shares the same backbone structure as the head and depth networks. However, the input to this module is the concatenation of the RGB scene image S_i and the binary head mask M_i that encodes the position of the person's head in the image. Each channel of the feature embedding of the scene network $\mathbf{e}_i^s$ is multiplied by the attention map $\mathbf{attn}^h$ generated by the head network. By multiplying the output of the scene network's backbone with the attention map, we force the network to focus on objects in the scene that are relevant w.r.t. the person of interest and its head orientation.

Depth Network. The depth network shares the same backbone structure and input shape of the scene network. This module receives as the input the depth map $\mathbf{e}_i^d$ of the scene and a binary head mask. The feature embeddings from the depth backbone are multiplied by the head attention map.

Fusion and Prediction Network. The feature embeddings from the head network $\mathbf{e}_i^h$ and the *attended* scene and depth embeddings $\mathbf{e}_i^s \bigotimes \mathbf{attn}_i^h, \mathbf{e}_i^d \bigotimes \mathbf{attn}_i^h$ are sent as input to the fusion and prediction network. Such components creates a lower-dimensional feature space of scene and depth by concatenating each of those embeddings with the output of the head network's backbone. To obtain the 2D gaze heatmap H_i this module uses a multi-layer decoder that takes as input the summation of scene and depth embeddings. Moreover, the channel-wise outer product [46] between scene and depth embeddings is input to a small encoder that produces the *in/out of frame* output $InOut_i$.

3.3 Details of Group Detection

Graph-Cuts for F-formation (GCFF) uses for each person $P_i = (x_i, y_i, \theta_i)$ (with $i \in [1, \ldots, n]$) their position (x_i, y_i) in the 2D map and their body orientation θ_i which is determined by their feet orientation (Fig. 3.2). The transactional segment of a person is modeled by a Gaussian $TS_i \sim \mathcal{N}(\mu_i, \Sigma_i)$ where the center $\mu_i = [x_{\mu_i}, y_{\mu_i}] = [x_i + D\cos\theta_i, y_i + D\sin\theta_i]$ is located a distance D in front of the person. The Gaussian is circular: $\Sigma_i = \sigma * \mathbf{I}$ where $\mathbf{I}$ is the 2D identity matrix. The transactional segment represents a circular area in front of the person that should overlap with the F-formations' o-space the person is part of. $O_g = [u_g, v_g]$ is the position of a candidate o-space centre for F-formation $g \in \{1, M\}$. G_i is the F-Formation that contains person i. Thus, $O_{G_i} = [u_{G_i}, v_{G_i}]$ represents the position of the o-space center to which person i is assigned to. Single persons are assigned to a F-formation that has only them as members.

Given these definitions, we can define the probability of a person being part of a candidate F-formation. The GCFF is then finding the maximum-likelihood solution that identifies groups o-spaces and their members. We start with defining the likelihood of an individual i's transactional segment centre $C_i = [u_i, v_i]$ given the a priori variable TS_i:

$$\Pr(C_i|TS_i) \propto \exp\left(-\frac{||C_i - \mu_i||_2^2}{\sigma^2}\right) = \exp\left(-\frac{(u_i - x_{\mu_i})^2 + (v_i - y_{\mu_i})^2}{\sigma^2}\right)$$

Thus, the probability that a person i shares an o-space centre O_{G_i} is given by:

$$\Pr(C_i = O_{G_i}|TS_i) \propto \exp\left(-\frac{(u_{G_i} - x_{\mu_i})^2 + (v_{G_i} - y_{\mu_i})^2}{\sigma^2}\right)$$

The posterior probability of the overall assignment of all people is given by:

$$\Pr(C = O_G|TS) \propto \prod_{i\in[1,n]} \exp\left(-\frac{(u_{G_i} - x_{\mu_i})^2 + (v_{G_i} - y_{\mu_i})^2}{\sigma^2}\right) \tag{3.5}$$

where C is a random variable that models the possible joint location of all o-space centres, O_G is an instance of such a joint location, and TS the collection of transactional segments of all people. Just based on (3.5)the maximum a posteriori probability (MAP) solution is to assign to each person its own group with the o-space center at the exact location of its transactional segment: $O_{G_i} = TS_i$. To avoid this solution a minimum description length prior (MDL) that punishes the number of F-formations is added to (3.5):

$$\Pr(C = O_G|TS) \propto \prod_{i\in[1,n]} \exp\left(-\frac{(u_{G_i} - x_{\mu_i})^2 + (v_{G_i} - y_{\mu_i})^2}{\sigma^2}\right) \cdot \exp(-|O_G|)$$

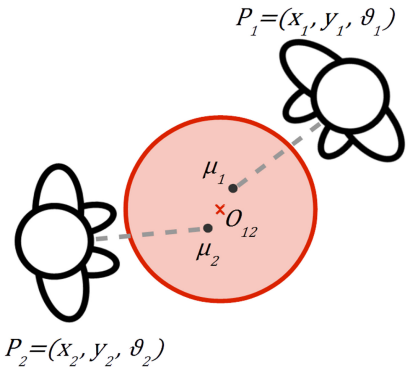


Figure 3.2: Inputs and outputs of the GCFF group detection algorithm [43]. Two persons (P_1, P_2) that face each other with their transactional segment centers μ_1 and μ_2. They are are categorized to be part of group O_{12} with its o-space depicted as a red circle. Figure taken from [43].

where $|O_G|$ is the number of F-formations. The MAP solution to this problem can be found by taking the negative log-likelihood and discarding normalising constants resulting in the objective:

$$J(O_G|TS) = \sum_{i \in [1,n]} (u_{G_i} - x_{\mu_i})^2 + (v_{g_i - y_{\mu_i}})^2 + \sigma^{-1}|O_G| \tag{3.6}$$

To solve (3.6), the GCFF procedure (Algorithm 1) starts by assigning to each person a possible o-space centre. It then uses a hill-climbing optimisation alternating between assigning individuals to o-space centres using a graph-cut based optimisation [24] that directly minimises the cost (3.6). It then minimises the least squares component by updating o-space centres to the mean of O_g, for all the individuals i currently assigned to the F-formation. The whole process is iterated until convergence.

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

Bibliography

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

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

[3] 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.

[4] João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. In *CoRR*, 2019.

[5] Min-Hung Chen, Zsolt Kira, Ghassan AlRegib, Jaekwon Yoo, Ruxin Chen, and Jian Zheng. Temporal attentive alignment for large-scale video domain adaptation. In *ICCV*, 2019.

[6] Jinwoo Choi, Gaurav Sharma, Manmohan Chandraker, and Jia-Bin Huang. Unsupervised and semi-supervised domain adaptation for action recognition from drones. In *WACV*, 2020.

[7] 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.

[8] 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.

[9] T Matthew Ciolek. The proxemics lexicon: A first approximation. *Journal of Nonverbal Behavior*, 8(1):55–79, 1983.

[10] Victor G. Turrisi da Costa, Giacomo Zara, Paolo Rota, Thiago Oliveira-Santos, Nicu Sebe, Vittorio Murino, and Elisa Ricci. Dual-head contrastive domain adaptation for video action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1181–1190, January 2022.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2019.

[12] 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. *arXiv preprint arXiv:2010.11929*, 2021.

[13] Mehmet Bilal Er. A novel approach for classification of speech emotions based on deep and acoustic features. *IEEE Access*, 8:221640–221653, 2020.

[14] Mark Everingham, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vision*, 88(2):303–338, jun 2010.

[15] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[16] 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.

[17] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[18] Yaroslav Ganin and Victor Lempitsky. Unsupervised Domain Adaptation by Backpropagation. In *ICML*, 2015.

[19] 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.

[20] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2106–2113, 2009.

[21] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017.

[22] Adam Kendon. *Conducting interaction: Patterns of behavior in focused encounters*, volume 7. CUP Archive, 1990.

[23] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, 2011.

[24] L'ubor Ladickỳ, Chris Russell, Pushmeet Kohli, and Philip HS Torr. Inference methods for crfs with co-occurrence statistics. *International journal of computer vision*, 103(2):213–225, 2013.

[25] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. In *ICLR Workshop*, 2017.

[26] 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.

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

[28] Meng Liu, Youfu Li, and Hai Liu. 3d gaze estimation for head-mounted eye tracking system with auto-calibration method. *IEEE Access*, 8:104207–104215, 2020.

[29] 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.

[30] Mingsheng Long and Jianmin Wang. Learning Transferable Features with Deep Adaptation Networks. In *ICML*, 2015.

[31] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, 2017.

[32] Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. Pretrained transformers as universal computation engines. *arXiv preprint arXiv:2103.05247*, 2021.

[33] Xi Ma, Zhiyong Wu, Jia Jia, Mingxing Xu, Helen Meng, and Lianhong Cai. Emotion recognition from variable-length speech segments using deep learning on spectrograms. In *Interspeech*, pages 3683–3687, 2018.

[34] Mustaqeem and Soonil Kwon. CLSTM: Deep feature-based speech emotion recognition using the hierarchical ConvLSTM network. *Mathematics*, 8(12):2133–2151, 2020.

[35] Mustaqeem, Muhammad Sajjad, and Soonil Kwon. Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM. *IEEE Access*, 8:79861–79875, 2020.

[36] Boxiao Pan, Zhangjie Cao, Ehsan Adeli, and Juan Carlos Niebles. Adversarial cross-domain action recognition with co-attention. In *AAAI*, 2020.

[37] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[38] 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.

[39] Adrià Recasens, Carl Vondrick, Aditya Khosla, and Antonio Torralba. Following gaze in video. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1444–1452, 2017.

[40] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[41] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 2018.

[42] Minji Seo and Myungho Kim. Fusing visual attention CNN and bag of visual words for cross-corpus speech emotion recognition. *Sensors*, 20(19):5559, 2020.

[43] Francesco Setti, Chris Russell, Chiara Bassetti, and Marco Cristani. F-formation detection: Individuating free-standing conversational groups in images. *PloS one*, 10(5):e0123783, 2015.

[44] Gilad Sharir, Asaf Noy, and Lihi Zelnik-Manor. An image is worth 16x16 words, what is a video worth? *arXiv preprint arXiv:2103.13915*, 2021.

[45] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[46] Zhongkai Sun, Prathusha Sarma, William Sethares, and Yingyu Liang. Learning relationships between text, audio, and video via deep canonical correlation for multimodal language analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8992–8999, 2020.

[47] Sanket Kumar Thakur, Cigdem Beyan, Pietro Morerio, and Alessio Del Bue. Predicting gaze from egocentric social interaction videos and imu data. In *Proceedings of the 2021 International Conference on Multimodal Interaction*, pages 717–722, 2021.

[48] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[49] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. *arXiv preprint arXiv:1503.02406*, 2015.

[50] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[51] Xianfeng Wang, Min Wang, Wenbo Qi, Wanqi Su, Xiangqian Wang, and Huan Zhou. A novel end-to-end speech emotion recognition network with stacked transformer layers. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6289–6293, 2021.

[52] Jin woo Choi, Gaurav Sharma, S. Schulter, and Jia-Bin Huang. Shuffle and attend: Video domain adaptation. In *ECCV*, 2020.

[53] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[54] Mingke Xu, Fan Zhang, Xiaodong Cui, and Wei Zhang. Speech emotion recognition with multiscale area attention and data augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6319–6323, 2021.

[55] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *2011 International conference on computer vision*, pages 1331–1338. IEEE, 2011.

[56] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

[57] Chenghao Zhang and Lei Xue. Autoencoder with emotion embedding for speech emotion recognition. *IEEE Access*, 9:51231–51241, 2021.

[58] Jianfeng Zhao, Xia Mao, and Lijiang Chen. Speech emotion recognition using deep 1d & 2D CNN LSTM networks. *Biomedical signal processing and control*, 47:312–323, 2019.

[59] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *Advances in neural information processing systems*, 27, 2014.