

ProxEmo: Gait-based Emotion Learning and Multi-view Proxemic Fusion for Socially-Aware Robot Navigation

Venkatraman Narayanan, Bala Murali Manoghar, Vishnu Sashank Dorbala, Dinesh Manocha, and Aniket Bera
University of Maryland, College Park, USA

Supplemental version including Code, Video, Datasets at <https://gamma.umd.edu/proxemo/>

Abstract— We present *ProxEmo*, a novel end-to-end emotion prediction algorithm for socially aware robot navigation among pedestrians. Our approach predicts the perceived emotions of a pedestrian from walking gaits, which is then used for emotion-guided navigation taking into account social and proxemic constraints. To classify emotions, we propose a multi-view skeleton graph convolution-based model that works on a commodity camera mounted onto a moving robot. Our emotion recognition is integrated into a mapless navigation scheme and makes no assumptions about the environment of pedestrian motion. It achieves a mean average emotion prediction precision of 82.47% on the Emotion-Gait benchmark dataset. We outperform current state-of-art algorithms for emotion recognition from 3D gaits. We highlight its benefits in terms of navigation in indoor scenes using a Clearpath Jackal robot.

I. INTRODUCTION

Recent advances in AI and robotics technology are gradually enabling humans and robots to coexist and share spaces in different environments. This is especially common in places such as hospitals, airports, and shopping malls. Navigating a robot with collision-free and socially-acceptable paths in such scenarios poses several challenges [1]. For example, in the case of a crowded shopping mall, the robot needs to be aware of the intentions of an oblivious shopper coming towards it for friendly navigation. Knowing the perceived emotional state of a human in such scenarios allows the robot to make more informed decisions and navigate in a socially-aware manner.

Understanding human emotion has been a well-studied subject in several areas of literature, including psychology, human-robot interaction, etc. There have been several works that try to determine the emotion of a person from verbal (speech, text, and tone of voice) [2], [3] and non-verbal (facial expressions, walking styles, postures) [4], [5] cues. There also exist multi-modal approaches that use a combination of these cues to determine the person’s emotion [6], [7].

In our work, we focus on emotionally-aware robot navigation in crowded scenarios. Here, verbal cues for emotion classification are not easily attainable. With non-verbal cues, facial expressions that are often occluded from the egocentric view of the robot and might not be fully visible. Besides, emotion analysis from facial features is a topic of debate in several previous works: these features are inherently unreliable caused by vague expressions emerging from a variety of psychological and environmental factors [8], [9]. As such, in our work, we focus on using “walking styles” or “gaits” to extract the emotions of people in crowds.

Obtaining perceived emotions from gaits is a challenging problem that has been well documented in the past. More recently, various machine learning solutions [10], [11] have been proposed to tackle this problem. However, these approaches suffer from the following drawbacks:



Fig. 1: ProxEmo: We present a gait-based emotion and proxemics learning algorithm to perform socially-aware robot navigation. The **red** arrow indicates the path of the robot without social awareness. The **green** arrow indicates the new path after an angry emotion is detected. Observe the significant shift away from the pedestrian when an angry gait is detected. This form of navigation is especially useful when the robot is expected to navigate safely through crowds without causing discomfort to nearby pedestrians.

- The training datasets used are singular in direction, i.e., there is motion capture only when a person is walking in a straight line towards the camera. This is a significant disadvantage for our task of socially-aware crowd navigation, where the robot often encounters people walking from several directions towards or away from the camera.
- Some approaches that are tailored towards using emotion for enhancing the task of robot navigation assume a static overhead camera that captures the trajectories of pedestrians. This is not ideal, as the overhead camera might not always be available in all scenarios.

To overcome these challenges, we propose *ProxEmo*, a novel algorithm for realtime gait-based emotion classification for socially-guided navigation. *ProxEmo* is tailored towards working with commodity RGB egocentric cameras that can

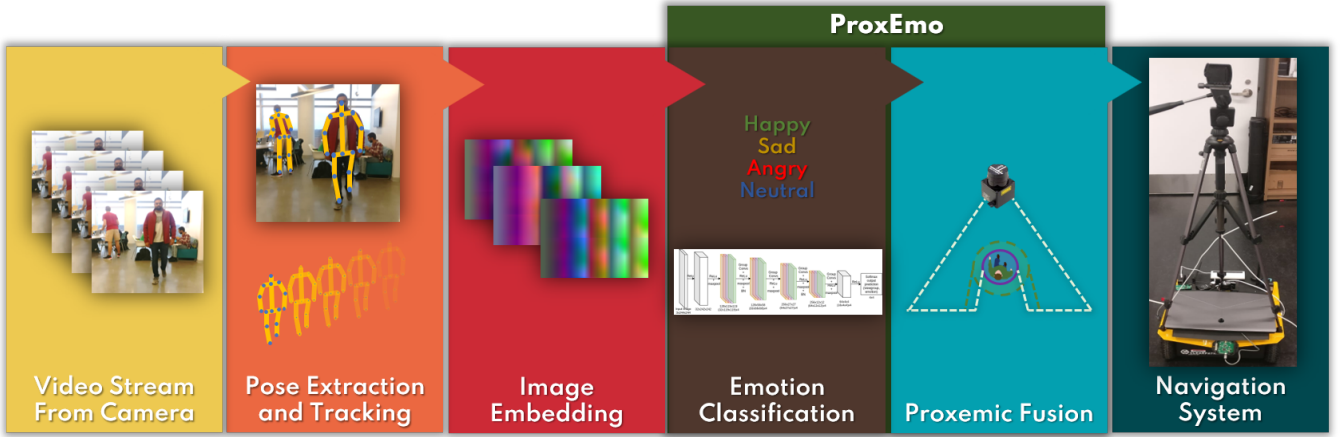


Fig. 2: Overview of our Pipeline: We first capture an RGB video from an onboard camera and extract pedestrian poses and track them at each frame. These tracked poses over a predefined time period are embedded into an image, which is then passed into our ProxEmo model for classifying emotions into four classes. The obtained emotions then undergo proxemic fusion with the LIDAR data and are finally passed into the navigation stack.

be retrofitted onto moving platforms or robots for navigating among pedestrians. The **major contributions** of our work can be summarized as follows:

- We introduce a novel approach using group convolutions to classify pedestrian emotions from gaits, which drastically improves accuracy compared to SOTA.
- Our method explicitly takes into consideration pedestrian behavior in crowds as we train our model on skeletal data of people approaching the robot from multiple directions, as opposed to approaching from a single view from the front.
- We present a new navigation scheme using **Proxemic Fusion** that accounts for pedestrian emotions.
- Finally, we introduce a **Variational Comfort Space**, which integrates into our navigation scheme, taking into account varying pedestrian orientations.

We note that identifying the true nature of a person’s emotion via only a visual medium can be difficult. Therefore in this work, we focus only on the *perceived* emotions from the point of an external observer as opposed to *actual* internal emotion.

II. RELATED WORK

In this section, we present a brief overview of social-robot navigation algorithms. We also review related work on emotion modeling and classification from visual cues.

A. Social Robotics and Emotionally-Guided Navigation

As robots have become more commonplace, their impact on humans’ social lives has emerged as an active area of research. Studies from multiple domains [12]–[15] have tried to quantify this impact in several ways. In [1], Kruse et al. present a comprehensive survey on navigation schemes for robots in social scenarios. They describe various social norms (interpersonal distances, human comfort, sociability) that the robot must consider not to cause discomfort to people around it. In [16], Satake et al. hypothesize strategies for robots to approach humans in crowded scenarios. Tai et al. [17] introduce a Generative Adversarial Imitation Learning (GAIL) based approach for socially compliant navigation.

Michaid et al. [18] discuss about how robots can attain *artificial* emotions for social interactions. Several classical [19]–[21] and deep learning [22] approaches tackle the problem of navigation through highly dynamic environments. More recently, reinforcement learning methods [23], [24] have been described for collision avoidance in such environments. For pedestrian handling, in particular, Randhavane et al. [25] make use of a pedestrian dominance model (PDM) to identify the dominance level of humans and plan a trajectory accordingly. In [26], Rios-Martinez et al. present a detailed survey on the proxemics involved with socially aware navigation. In [27], Kitazawa et al. discuss ideas such as *Information Process Space* of a human. In [28], Pandey et al. discuss a strategy to plan a socially aware path using milestones.

B. Emotion Modeling and Classification

There exists a substantial amount of research that focuses on identifying the emotions of humans based on body posture, movement, and other non-verbal cues. Ruiz-Garcia et al. [29] and Tarnowski et al. [30], use deep learning to classify different categories of emotion from facial expressions. Randhavane et al. [31], [32] classify emotions into four classes based on affective features obtained from 3D skeletal poses extracted from human gait cycles. Their algorithm, however, requires a large number of 3D skeletal key-points to detect emotions and is limited to single individual cases. Bera et al. [33] classify emotions based on facial expressions along with a pedestrian trajectory obtained from overhead cameras. Although this technique achieves good accuracy in predicting emotions from trajectories and facial expressions, it explicitly requires overhead cameras in its pipeline and does not work on a mobile robot equipped with only onboards cameras.

C. Action Recognition for Emotions

The task of action recognition involves identifying human actions from sequences of data (usually videos) [34]. A common task in many of these models is recognizing gait-based actions such as walking and running. Thus, the task of gait action recognition is closely related to the task of emotion recognition from gaits, as both perform classification

on the same input. Bhattacharya et al. [11], [35] use graph convolutions for the emotion recognition task, in a method similar to the action recognition model used in Yan et al. [36]. Ji et al. [37] propose a CNN-based method that gives state-of-the-art results on gait based action recognition tasks. Their model is invariant to viewpoint changes.

III. OVERVIEW AND METHODOLOGY

We propose a novel approach, *ProxEmo*, for classifying emotions from gaits that works with an egocentric camera setup. Our method uses 3D poses of human gaits obtained from an onboard robot camera to classify perceived emotions. These perceived emotions are then used to compute variable proxemic constraints in order to perform socially aware navigation through a pedestrian environment.

Figure 2 illustrates how we incorporate *ProxEmo* into an end-to-end *emotionally-guided* navigation pipeline.

The following subsections will describe our approach in detail. We first discuss the dataset and the augmentation details we used for training. Then, we briefly discuss our pose estimation model, followed by a detailed discussion of our emotion classification model, *ProxEmo*. Finally, we describe how socially-aware navigation can be performed using the obtained emotions.

A. Notations

In our formulation, we represent the human with 16 joints as shown in figure 4. Thus, a pose $P \in \mathbb{R}^{16 \times 3}$ of a human is a set of 3D positions of each joint j_i , where $i \in \{0, 1, \dots, 15\}$. For any RGB video V , we represent the gait extracted using 3D pose estimation as G . The gait G is a set of 3D poses P_1, P_2, \dots, P_τ where τ is the number of frames in the input video V .

B. Dataset Preparation

We make use of two labeled datasets by Randhavane et al. [38] and Bhattacharya et al. [11], containing time-series 3D joints of 342 and 1835 gait cycles each (a total of 2177 gait samples). Each gait cycle has 75 timesteps with 16 joints as shown in Figure 4. Thus, each sample in this new dataset has a dimension of $joints \times time \times dimensions = 16 * 75 * 3 = 3600$. These samples are labeled into 4 emotion classes: *angry*, *sad*, *happy*, and *neutral* with 10 labelers per video (to capture the perceptual difference between different labelers). In order to train our network for prediction from multiple views, we augment the dataset as follows. First, we consider a camera placed at a particular distance from the human, as shown in Figure 3. Then for different camera positions oriented towards the human, we perform augmentation by applying transformations given in equation 1.

$$j_{aug} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad (1)$$

where j_{aug} are the coordinates of the augmented joints, T_x, T_y, T_z are the translation vectors, and θ is the rotation along Y axis. For our experiments, we attain $72 \times 4 = 288$ augmentations for each sample by considering θ at gradients of 5° , with 4 translations of $[1m - 4m]$ along the Z axis (T_z). Thus, after augmentation, we have a total of $288 \times 2177 = 626,976$ gaits in our dataset.

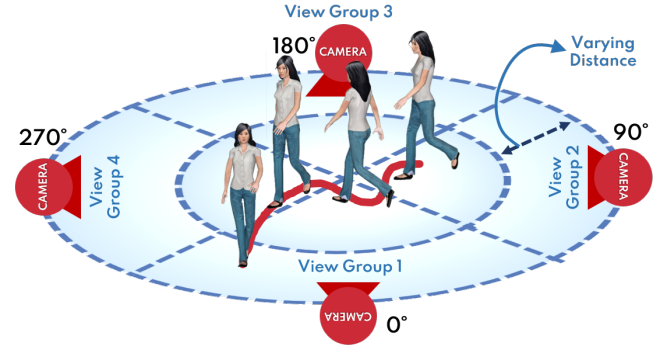


Fig. 3: Data Augmentation: By applying specific translations and rotations, we augment the data into different camera views. We divide the viewpoints into four view-groups based on the angle of approach to categorize the direction in which the person is walking. The augmentations also take into consideration varying distances of the camera from the origin point of the gait sequence.

C. Human-Pose Estimation

A pose estimation strategy for humans walking in a crowded real-world scenario has to be robust to noise coming from human attire or any items they might be carrying. To account for this, we employ a robust approach described in [39] for this task. Their paper describes a two-step network trained in a weakly supervised fashion. First, a *Structure-Aware PoseNet (SAP-Net)* trained on spatial information provides an initial estimate of joint locations of people in video frames. Later, a *Temporal PoseNet (TP-Net)* trained on time-series information corrects this initial estimate by adjusting illegal joint angles and joint distances. The final output is a sequence of well-aligned 3D skeletal poses P . Figure 4 is a representation of the skeletal output obtained.

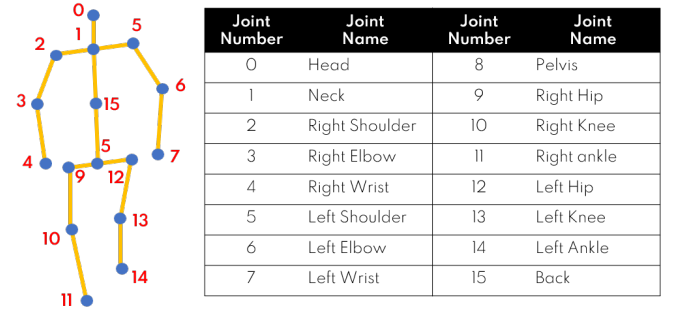


Fig. 4: Skeleton Representation: We represent a pedestrian by 16 joints (j_i). The overall pose of the pedestrian is defined using these joint positions.

D. Generating Image Embeddings

We observe that 2D convolutions are much faster and efficient as opposed to graph convolutions [37]. Hence, we embed the spatial-temporal skeletal gait sequence G as an image I , using the equations described in 2.

$$I = \{R_{(x,y)}; G_{(x,y)}; B_{(x,y)}\} \quad (2)$$

$$B_{(x,y)} = X_{(t,j)} \quad G_{(x,y)} = Y_{(t,j)} \quad R_{(x,y)} = Z_{(t,j)}$$

Here, R , B , and G are image color channels, x, y are the co-ordinates of image, and X_j, Y_j, Z_j are the co-ordinates

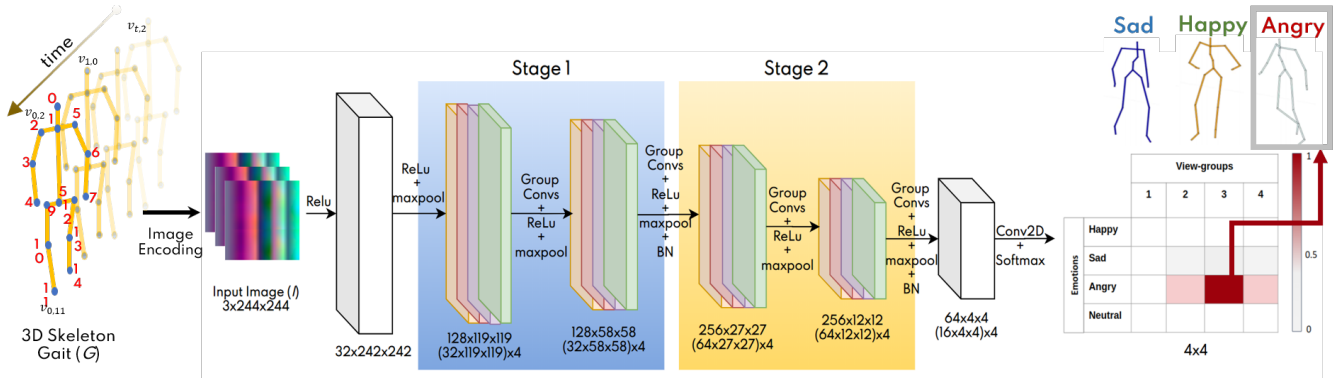


Fig. 5: ProxEmo Network Architecture: The network is trained on image embeddings of the 5D gait set G , which are scaled up to 244×244 . The architecture consists of four group convolution (GC) layers. Each GC layer consists of four groups that have been stacked together. This represents the four group convolution outcomes for each of the four emotion labels. The group convolutions are stacked in two stages represented by **Stage 1** and **Stage 2**. The output of the network has a dimension of 4×4 after passing through a softmax layer. The final predicted emotion is given by the maxima of this 4×4 output.

of skeletal joint j at time t . This image I is finally upsampled to $244 \times 244 \times 3$ for training our *ProxEmo* model.

E. ProxEmo: Classifying Emotions from Gaits

Figure 5 illustrates the architecture of our model for emotion classification. The image embeddings I obtained from the gaits are passed through two stages of group convolutions to obtain an emotion label.

1) *Group Convolutions*: We take inspiration from [40] and make use of group convolutional layers in designing our *ProxEmo* architecture. **Group Convolution Layers (GC)**, in essence, operate just like 2D convolution layers, except that they fragment the input into n_g groups and perform convolution operations individually on them before stacking the outputs together. The advantage of doing this is that the network learns from different parts of the input in isolation. This is especially useful in our case because we have a dataset that varies based on two factors, view-group and emotion labels. The variation in the view-groups is learned by the different convolution groups GC , and the emotions are learned by the convolutions taking place within each group. Group convolutions increase the number of channels in each layer by n_g times. The output (h_i) of each group in the convolution layer is,

$$h_i = x_i * k_i \quad (3)$$

$$h_{out} = [h_1 | \dots | h_{n_g}] \quad (4)$$

where, h_{out} is the output of the group convolution, x_i represents the input, and k_i represents the kernel for convolution. The output $[h_1 | \dots | h_{n_g}]$ is a matrix concatenation of all the group outputs along channel axis. In our case, we choose n_g as 4 because we have 4 view-groups.

2) *ProxEmo Architecture*: The network consists of seven convolution layers. The initial layer is a traditional 2D convolution layer, which performs channel up-sampling for the forthcoming group convolution operations. These operations take place in two stages -

Stage 1: This consists of two GC layers, each having 128 convolution filters (32 per group $\times n_g$).

Stage 2: This consists of two convolution GC layers, however, unlike stage 1, each GC 256 convolution filters (64 per view-group $\times n_g$).

Both traditional 2D convolution and GC layers are passed through a *ReLU* non-linear activation and max pooling layer. The outputs from **Stage 1** and **Stage 2** are represented by h_s where $s = 1, 2$. We also perform *batch normalization*. The output of each both the group convolution stages, h_s are given by,

$$\begin{aligned} p_s^* &= GC(x_s, k_s^1) \\ p_s &= \text{MaxPool}(\text{ReLU}(p_s^*)) \\ h_s^* &= GC(p_s, k_s^2) \\ h_s &= \text{BatchNorm}(\text{MaxPool}(\text{ReLU}(h_s^*))) \end{aligned} \quad (5)$$

where, s represents the two group convolution stages as described before, x_s is the input to the group convolution stage ' s ', k_s^1 and k_s^2 represent convolution kernels for first and second GC layers within a stage, p_s^* and h_s^* are the first and second GC layer outputs determined using equation 3.

After performing the group convolutions, the output h_2 is passed through two 2D convolution layers. These convolution layers help in gathering the features learned by the GC layers to finally predict both the view-group and emotion of the gait sequences.

Rather than using fully-connected layers for predicting the view-group, our method utilizes convolution layers to predict the $n_k \times n_g$ output, where n_k is the number of emotions and n_g is the number of view-groups. This makes our model considerably lighter (number of model parameters) and faster (run-time performance), compared to other state-of-the-art algorithms.

The final output of the classifier consists of multi-class softmax prediction, $E_{i,j}$, given by the equation 6. Here $e_{i,j}$ refers to the final hidden layer output of the network, where $i = 0, 1, \dots, (n_k - 1)$ is the emotion class and $j = 0, 1, \dots, (n_g - 1)$ is the view-group class.

$$E_{i,j} = \frac{\exp(e_{i,j})}{\sum_{i=0}^{n_k-1} \sum_{j=0}^{n_g-1} \exp(e_{i,j})} \quad (6)$$

$E_{i,j}$ can be considered as a 4×4 matrix containing 16 values corresponding to different view-groups and emotions.

F. Emotion-guided Navigation using Proxemic Fusion

We use the emotions $E_{i,j}$ predicted from *ProxEmo* to compute the comfort space (c) of a pedestrian, which

is the socially comfortable distance (in cm) around a person. We combine c along with the LIDAR data (L) to perform “proxemic fusion” (III-F.3), obtaining a set of points where it is permissible for the robot to navigate in a socially-acceptable manner. This is illustrated in figure 6.

1) *Comfort Space Computation*: In order to model the predicted emotions $E_{i,j}$ from *ProxEemo* into a comfort space distance c , we use the following equation:

$$c = \frac{\sum_{j=1}^4 c_j \cdot \max(E_j)}{\sum_{j=1}^4 E_j} \cdot v_g \quad (7)$$

Here, E_j represents a column vector of the *softmax* output, which corresponds to the group outcomes for each individual emotion. c_j is a constant derived from psychological experiments described in [41] to compute the limits on an individual’s comfort spaces and is chosen from a set $\{90.04, 112.71, 99.75, 92.03\}$ corresponding to the comfort spaces (radius in cm) for $\{happy, sad, angry, neutral\}$ respectively. We acknowledge that these distances depend on many factors, including cultural differences, environment, or a pedestrian’s personality, and restrict our claims to variations in comfort spaces due to the emotional difference. These distances are based on how comfortable pedestrians are while interacting with others. v_g is a view-group constant defined in the following subsection.

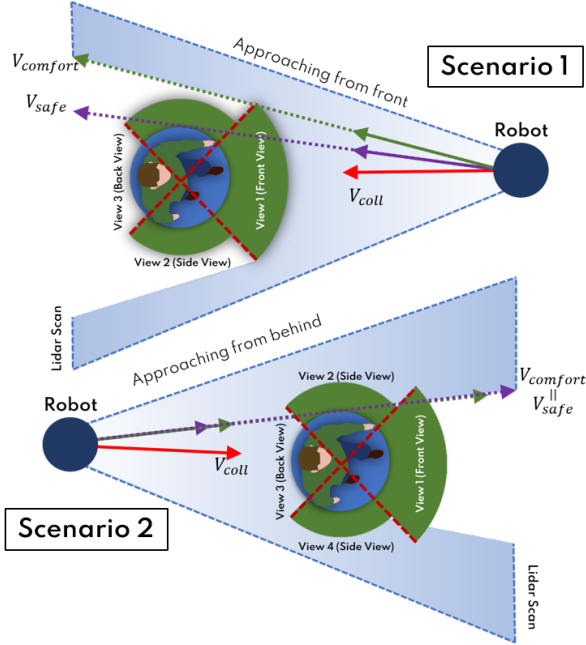


Fig. 6: Variational Comfort Space: We consider a varying comfort space c around a person based on their position (defined by the view-group g) in front of the robot. In scenario 1, the pedestrian approaches the robot from the front. Here, as the pedestrian is aware of the robot’s presence, it needs to be more respectful of the proxemic comfort space and thus takes action $V_{comfort}$ represented by the green arrow. In scenario 2, the robot is approaching the person from behind. An unaware pedestrian need not be disturbed by the robot, due to which it can be more liberal with its actions. The violet arrow representing the safe action V_{safe} coincides with $V_{comfort}$ in this case.

2) *Variational Comfort Space (v_g)*: We take inspiration from the *Information Process Space* defined by Kitazawa et

al. [27] to define our own Variational Comfort Space constant v_g . This constant acts as a scaling factor in the comfort space based on the orientation of the pedestrian in the robot’s view. This orientation is easily obtainable as *ProxEemo* also gives us a view-group output along with the emotion.

v_g is chosen from a set of $\{1, 0.5, 0, 0.5\}$ based on the view group g predicted. This is chosen based on the fact that people have varying personal space with respect to their walking direction, i.e., a pedestrian will care more about his/her personal space in front as compared to the sides. Also, the pedestrian might not care about personal and comfort space behind them since it does not lie in their field of view [42].

In figure 6, we look at two scenarios to illustrate how the robot handles pedestrians considering variational comfort spaces:

- **Scenario 1:** The robot is positioned in front of the person walking towards it. This is classified as view-group 1, having a v_g value of 1. As the robot is visible to the person, in this case, it should be more precautionary in safely maneuvering around the person. The comfort space around the pedestrian is larger in this case, and the robot takes a more skewed trajectory.
- **Scenario 2:** The robot is approaching the pedestrian from behind. This gait is classified as view-group 3 and has a v_g value of 0. As the robot is not in the person’s field of vision, in this case, it can afford to safely pass around the fixed space F_s of the person.

At any time instant, the velocity of the robot will be directed towards the goal, and if there is an obstacle, it will lead to a collision v_{coll} . If an obstacle avoidance algorithm is used, the navigation scheme avoids it with an action v_{safe} . However, for socially acceptable proximally-aware navigation, this is not sufficient, as this requires the robot to follow certain social norms. In order to adhere to these social norms, we incorporate the emotions predicted by *ProxEemo* to navigate in a socially acceptable manner represented by $V_{comfort}$.

3) *Proxemic Fusion*: We fuse the LIDAR data to include proxemic constraints by performing a Minkowski sum (M) of the set of LIDAR points L and a set containing the points in a circle Z defined by a radius r . The Minkowski sum M provides us with a set of all the admissible points where the robot can perform emotionally-guided navigation. This is formulated using the following equations.

$$\begin{aligned} L &= \{a \mid a - a_0 = d_{lidar}\} \\ Z &= \{b \mid dist(a, b) \leq r\} \\ M &= L + Z = \{a + b \mid a \in L, b \in Z\} \end{aligned} \quad (8)$$

Here, a_0 is a reference point on the LIDAR, and d_{lidar} is the distance measurement (in metres). r is the inflation radius and is defined using the comfort space c as:

$$r = c - [\max(dh) - \min(dh)] \quad (9)$$

where $dh \in L$ is a set of the LIDAR distances only for points where a human was detected. The maximum value of dh corresponds to the farthest distance from the person from their fixed inner space F_s , while the minimum value of dh corresponds to the closest distance of the person from this space. F_s is represented by the blue circle around the person

in the figure 6. In terms of mathematical morphology, the outcome of *proxemic fusion* is similar the dilation operation of the human, modelled as a obstacle, with the comfort space.



Fig. 7: Emotionally-Guided Navigation: We use the emotions detected by ProxEmo along with the LIDAR data to perform Proxemic Fusion. This gives us a comfort distance c around a pedestrian for emotionally-guided navigation. The green arrows represent the path after accounting for c while the violet arrows indicate the path without considering this distance. Observe the significant change in the path taken, especially in the sad case. Note that the overhead image is representational, and ProxEmo works entirely from a egocentric camera on a robot.

We encourage the readers to see detailed analysis, source code, and qualitative video results of our algorithm at <https://gamma.umd.edu/proxemo/>.

IV. EXPERIMENTS AND RESULTS

A. Evaluation Metrics

We evaluate our model using two metrics:

- **Mean Accuracy (%)** - $\frac{1}{n_k \times n_g} \sum_{i=0}^{n_k} \sum_{j=0}^{n_g} \frac{TP_{i,j}}{N_{i,j}}$
- **Mean F1 score** - $\frac{2}{n_k \times n_g} \sum_{i=0}^{n_k} \sum_{j=0}^{n_g} \frac{Pr_{i,j} * Rc_{i,j}}{Pr_{i,j} + Rc_{i,j}}$

where, n_k ($= 4$) is the number of emotion classes, n_g ($= 4$) is the number of view-groups, $TP_{i,j}$ is the number of true predictions for i^{th} emotion class and j^{th} view-group, $N_{i,j}$ is the total number of data samples for i^{th} emotion class and j^{th} view-group, $Pr_{i,j}$ and $Rc_{i,j}$ is the *precision* and *recall* for i^{th} emotion class and j^{th} view-group. All the metrics mentioned are derived from a *confusion matrix* generated by comparing actual vs predicted emotion and view-group for the data samples.

B. Implementation Details

For training, our dataset (III-B) has a train-validation split of 90%-10%. We generate a set of angles and translations that are different from the original dataset to formulate the test set.

We perform training using an ADAM [43] optimizer, with decay parameters of ($\beta_1 = 0.9$ and $\beta_2 = 0.999$). The

experiments were run with a learning rate of 0.009 and with 10% decay every 250 epochs. The models were trained with softmax multi-class cross-entropy loss, \mathcal{L} , represented in equation 10. The training was done on 2 Nvidia RTX 2080 Ti GPUs having 11GB of GPU memory each and 64 GB of RAM.

$$\mathcal{L} = \frac{1}{m} \sum_{m=1}^M \sum_{i=0, j=0}^{n_k, n_g} -y_{m,i,j} \log(E_{m,i,j}) \quad (10)$$

where, $y_{m,i,j}$ is the target one-hot encoded label representing emotion class $i \in \{0, 1, \dots, n_k\}$ and view-group $j \in \{0, 1, \dots, n_g\}$ for the data sample $m \in \{0, 1, \dots, M\}$. $E_{m,i,j}$ is the predicted *softmax* output probability for data sample m being emotion class i and view-group class j .

C. Comparing ProxEmo with other Emotion Classifiers

We evaluate the performance of our ProxEmo network, against two other emotion classification algorithms [11] [38]. Since the other emotion classification algorithms don't consider the arbitrary view scenario, we compare our results with just single-view data, i.e., skeletal gaits that are directly approaching the RGB-D camera. Table I presents these results. The accuracy metrics reported are generated by modifying the equations in IV-A, for a single view-group (i.e., $n_g = 1$).

Method	Accuracy (%)
Venture et al. [44]	30.83
Daoudi et al [45]	42.5
Li et al. [46]	53.7
Baseline (Vanilla LSTM) [38]	55.47
Crenn et al [47]	66.2
STEP [11]	78.24
ProxEmo (ours)	82.4

TABLE I: Comparison of ProxEmo with other state-of-the-art emotion classification algorithms: We compare the accuracy (%) of our ProxEmo network with existing emotion classification algorithms on single-view (facing the camera) data samples. We observe that our network outperforms the current state-of-the-art algorithm by 4%. Furthermore, our network outperforms the state-of-the-art algorithm across each emotion class.

D. Comparing ProxEmo with Action Recognition Models

As mentioned in section II-C, action recognition models and emotion recognition models that have inputs as gaits are closely related tasks. Thus, we can evaluate ProxEmo on pre-existing action recognition models by fine-tuning them on the emotion recognition task. We compare our model with two existing state-of-the-art action recognition models, (i) Spatial-Temporal Graph convolution networks (ST-GCN) [36], and (ii) VS-CNN [37]. These architectures were trained using the datasets [11], [38] (discussed in Section III-B).

1) *ST-GCN*: The spatial-temporal graph convolution networks [36] perform skeletal action recognition using undirected spatial-temporal graphs for hierarchical representation of skeleton gait sequences. In the original implementation, the spatial-temporal graphs are used in a graph convolution network to detect the action performed through the sequence.

We fine-tune ST-GCN to predict human emotion instead of the actions. The human emotions modeled as a class label for the implementation.

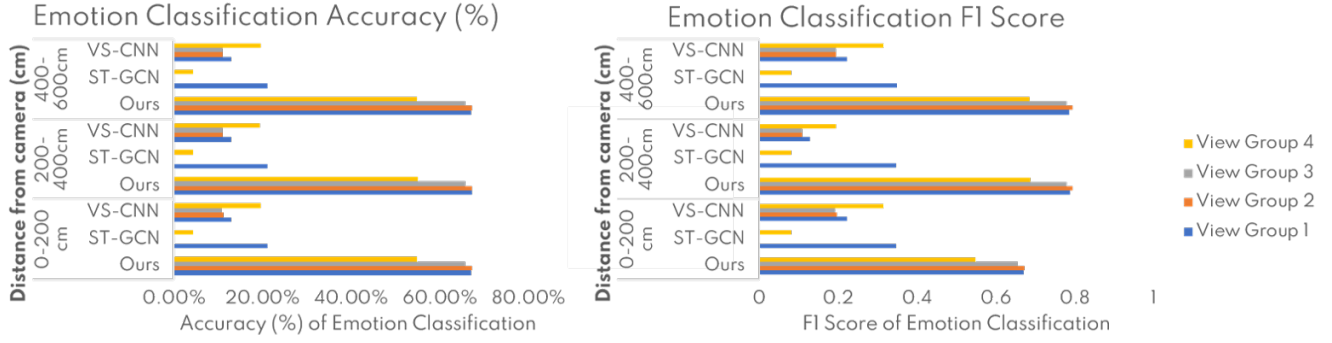


Fig. 8: Comparison of ProxEemo with other arbitrary view algorithms : Here we present the performance metrics (discussed in section IV-A) of our ProxEemo network compared to the state-of-the-art arbitrary view action recognition models. We perform a comprehensive comparison of models across multiple distances of skeletal gaits from the camera and across multiple view-groups. It can be seen that our ProxEemo network outperforms other state-of-the-art network by 50% at an average in terms of prediction accuracy.

		Predicted			
		Angry	Happy	Sad	Neutral
Actual	Angry	95.88%	1.19%	0.48%	2.46%
	Happy	1.76%	94.49%	0.91%	2.85%
	Sad	1.22%	3.50%	83.37%	11.91%
	Neutral	5.62%	6.71%	6.39%	81.28%

Fig. 9: Confusion Matrix: We show the percentage of gaits belonging to every emotion class that were correctly classified by our algorithm, ProxEemo.

2) **VS-CNN:** One of the major drawbacks of ST-GCN is that it is not tuned for multi-view/arbitrary-view skeletal gait sequences. View-guided Skeleton CNN (VS-CNN) [37] approaches this problem by building a dataset that multiple view-points with respect to the human reference frame. The multiple views are combined into four groups, each consisting of the one-quarter (90 degrees) of the view-points sequences. The action recognition is performed in three stages: (i) a *view-group predictor network* that predicts the view-group C (of 4 view-groups) of the sequence. (ii) a *view-group feature network* that consists of four individual networks, based on SK-CNN [48], for each view-group, and finally, (iii) a *channel classifier network* that combines (i) and (ii) to predict the action label for the skeletal gait sequence.

The VS-CNN also steers away from graph convolutions with an aim to increase the run-time performance of the network. 2D convolutions were observed to be much faster and efficient as opposed to graph convolutions. Hence, the spatial-temporal skeletal gait sequences are transformed into images. In our experiment, we tweak the final output of VS-CNN architecture using equation 6 to predict human emotions as opposed to actions. The network was trained with a *softmax* cross-entropy loss function, represented in equation 10.

The table I and figure 8 present a comparison of our model against VS-CNN and ST-GCN. We can observe that ProxEemo outperforms the state-of-the-art action recognition algorithms in both single-view and arbitrary-view skeletal gait sequences. Also, observe that in table II, ProxEemo takes up the least number of model parameters. This is because we perform group convolutions and eliminate Fully Connected layers in our network. Figure 9 is a confusion matrix of the predicted vs actual emotion classes of ProxEemo. We

can infer from this matrix that our model performs fairly well across all emotion classes with a high accuracy. Since, the evaluation metrics for *socially acceptable* is not well-defined, we don't report any evaluation on our *emotion-guided navigation planning*.

View-Groups	Model Parameters		
	ST-GCN [36]	VS-CNN [37]	ProxEemo(ours)
4	1.4M	63M	0.33M
6	1.4M	65M	0.5M
8	1.4M	68M	0.69M

TABLE II: Comparison of model parameters: Our ProxEemo model has significantly fewer parameters compared to ST-GCN [36] and VS-CNN [37]. This is due to the fact that we use Group Convolutions (GC) and eliminate Fully Connected (FC) layers in our network.

V. CONCLUSION, LIMITATIONS AND FUTURE WORK

We present ProxEemo, a novel group convolution-based deep learning network that takes 3D skeletal gaits of a human and predicts the perceived emotional states {happy, sad, angry, neutral} for *emotionally-guided* robot navigation. Our model specifically takes into consideration arbitrary orientations of pedestrians and is trained using augmented data comprising of multiple view-groups. We also present a new approach for socially-aware navigation that takes into consideration the predicted emotion and view-group of the pedestrian in the robot's field of view. In doing this, we also define a new metric for computing comfort space, that incorporates constants derived from emotion and view-group predictions. The limitation of our model during inference time is that it is reliant on real-time 3D skeletal tracking.

In the future, we plan to look at multi-modal cues for emotion recognition. We intend to dynamically compute proxemic constraints using continual feedback in a reward-based training scheme. We would also like to model other emotions like Excitement, Disgust, Confused, etc. or even other representations of emotions, like the continuous VAD model (Valence, Arousal, and Dominance) [49]. We also plan to add higher-level information, with regards to the environmental or cultural context that are known to influence human emotions, which can further improve our classification results.

REFERENCES

- [1] T. Kruse, A. K. Pandey, R. Alami, and A. Kirsch, "Human-aware robot navigation: A survey," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1726 – 1743, 2013.
- [2] Tin Lay Nwe, Foo Say Wei, and L. C. De Silva, "Speech based emotion classification," in *Proceedings of TENCON 2001*, Aug 2001.
- [3] A. Tawari and M. M. Trivedi, "Speech emotion analysis in noisy real-world environment," in *2010 20th International Conference on Pattern Recognition*, Aug 2010, pp. 4605–4608.
- [4] C. F. Benítez-Quiroz, R. Srinivasan, and A. M. Martínez, "Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild," *2016 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5562–5570, 2016.
- [5] A. Kleinsmith and N. Bianchi-Berthouze, "Affective body expression perception and recognition: A survey," *IEEE Transactions on Affective Computing*, vol. 4, no. 1, pp. 15–33, Jan 2013.
- [6] T. Mittal, U. Bhattacharya, R. Chandra, A. Bera, and D. Manocha, "M3er: Multiplicative multimodal emotion recognition using facial, textual, and speech cues," 2019.
- [7] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, and S. Narayanan, "Analysis of emotion recognition using facial expressions, speech and multimodal information," in *ICMI*. ACM, 2004.
- [8] L. Y. Mano, B. S. Façal, V. P. Gonçalves, G. Pessin, P. H. Gomes, A. C. de Carvalho, and J. Ueyama, "An intelligent and generic approach for detecting human emotions: a case study with facial expressions," *Soft Computing*, pp. 1–13, 2019.
- [9] L. F. Barrett, R. Adolphs, S. Marsella, A. M. Martinez, and S. D. Pollak, "Emotional expressions reconsidered: challenges to inferring emotion from human facial movements," *Psychological Science in the Public Interest*, vol. 20, no. 1, pp. 1–68, 2019.
- [10] M. Chiu, J. Shu, and P. Hui, "Emotion recognition through gait on mobile devices," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, March 2018, pp. 800–805.
- [11] U. Bhattacharya, T. Mittal, R. Chandra, T. Randhavane, A. Bera, and D. Manocha, "Step: Spatial temporal graph convolutional networks for emotion perception from gaits," *preprint arXiv:1910.12906*, 2019.
- [12] C. Breazeal, "Toward sociable robots," *Robotics and Autonomous Systems*, vol. 42, no. 3, 2003, socially Interactive Robots.
- [13] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and Autonomous Systems*, vol. 42, no. 3, pp. 143 – 166, 2003, socially Interactive Robots.
- [14] H. Yan, M. H. Ang, and A. N. Poo, "A survey on perception methods for human-robot interaction in social robots," *International Journal of Social Robotics*, vol. 6, no. 1, pp. 85–119, 2014.
- [15] A. Bera, S. Kim, T. Randhavane, S. Pratapa, and D. Manocha, "Glmpr: real-time pedestrian path prediction using global and local movement patterns," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 5528–5535.
- [16] S. Satake, T. Kanda, D. F. Glas, M. Imai, H. Ishiguro, and N. Hagita, "How to approach humans? strategies for social robots to initiate interaction," in *Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction*, ser. HRI '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 109–116.
- [17] L. Tai, J. Zhang, M. Liu, and W. Burgard, "Socially compliant navigation through raw depth inputs with generative adversarial imitation learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1111–1117.
- [18] F. Michaud, P. Pirjanian, J. Audet, and D. Létourneau, *Artificial Emotion and Social Robotics*. Tokyo: Springer Japan, 2000.
- [19] D. Wilkie, J. Van Den Berg, and D. Manocha, "Generalized velocity obstacles," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 5573–5578.
- [20] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Optimal reciprocal collision avoidance for multi-agent navigation," in *Proc. of the IEEE International Conference on Robotics and Automation, Anchorage (AK), USA*, 2010.
- [21] J. Van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 1928–1935.
- [22] A. Loquercio, A. I. Maqueda, C. R. Del-Blanco, and D. Scaramuzza, "Dronet: Learning to fly by driving," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1088–1095, 2018.
- [23] P. Long, T. Fanl, X. Liao, W. Liu, H. Zhang, and J. Pan, "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6252–6259.
- [24] T. Fan, X. Cheng, J. Pan, D. Manocha, and R. Yang, "Crowdmov: Autonomous mapless navigation in crowded scenarios," *arXiv preprint arXiv:1807.07870*, 2018.
- [25] T. Randhavane, A. Bera, E. Kubin, A. Wang, K. Gray, and D. Manocha, "Pedestrian dominance modeling for socially-aware robot navigation," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5621–5628.
- [26] J. Rios-Martínez, A. Spalanzani, and C. Laugier, "From proxemics theory to socially-aware navigation: A survey," *IJSR*, 2015.
- [27] K. Kitazawa and T. Fujiyama, "Pedestrian vision and collision avoidance behavior: Investigation of the information process space of pedestrians using an eye tracker," in *Pedestrian and Evacuation Dynamics 2008*, W. W. F. Klingsch, C. Rogsch, A. Schadschneider, and M. Schreckenberg, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 95–108.
- [28] A. K. Pandey and R. Alami, "A framework towards a socially aware mobile robot motion in human-centered dynamic environment," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2010, pp. 5855–5860.
- [29] A. Ruiz-Garcia, M. Elshaw, A. Altahhan, and V. Palade, "Deep learning for emotion recognition in faces," in *Artificial Neural Networks and Machine Learning*, A. E. Villa, P. Masulli, and A. J. Pons Rivero, Eds. Cham: Springer International Publishing, 2016, pp. 38–46.
- [30] P. Tarnowski, M. Kołodziej, A. Majkowski, and R. J. Rak, "Emotion recognition using facial expressions," *Procedia Computer Science*, vol. 108, pp. 1175 – 1184, 2017, international Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.
- [31] T. Randhavane, A. Bera, K. Kapsaskis, U. Bhattacharya, K. Gray, and D. Manocha, "Identifying emotions from walking using affective and deep features," *arXiv preprint arXiv:1906.11884*, 2019.
- [32] T. Randhavane, U. Bhattacharya, K. Kapsaskis, K. Gray, A. Bera, and D. Manocha, "The liar's walk: Detecting deception with gait and gesture," *arXiv preprint arXiv:1912.06874*, 2019.
- [33] A. Bera, T. Randhavane, R. Prinja, K. Kapsaskis, A. Wang, K. Gray, and D. Manocha, "The emotionally intelligent robot: Improving social navigation in crowded environments," *arXiv preprint arXiv:1903.03217*, 2019.
- [34] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and vision computing*, 2017.
- [35] U. Bhattacharya, C. Roncal, T. Mittal, R. Chandra, A. Bera, and D. Manocha, "Take an emotion walk: Perceiving emotions from gaits using hierarchical attention pooling and affective mapping," *arXiv preprint arXiv:1911.08708*, 2019.
- [36] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.
- [37] Y. Ji, F. Xu, Y. Yang, F. Shen, H. T. Shen, and W.-S. Zheng, "A large-scale varying-view rgb-d action dataset for arbitrary-view human action recognition," *arXiv preprint arXiv:1904.10681*, 2019.
- [38] T. Randhavane, U. Bhattacharya, K. Kapsaskis, K. Gray, A. Bera, and D. Manocha, "Learning perceived emotion using affective and deep features for mental health applications," in *2019 ISMAR*, Oct 2019.
- [39] R. Dabral, A. Mundhada, et al., "Learning 3d human pose from structure and motion," in *ECCV*, 2018.
- [40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [41] G. Ruggiero, F. Frassinetti, Y. Coello, M. Rapuano, A. S. Di Cola, and T. Iachini, "The effect of facial expressions on peripersonal and interpersonal spaces," *Psychological research*, 2017.
- [42] S. Kim, J. Choi, S. Kim, and R. Tay, "Personal space, evasive movement and pedestrian level of service," *Journal of advanced transportation*, vol. 48, 2014.
- [43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [44] G. Venture, H. Kadone, T. Zhang, J. Grèzes, A. Berthoz, and H. Hicheur, "Recognizing emotions conveyed by human gait," *International Journal of Social Robotics*, vol. 6, no. 4, pp. 621–632, 2014.
- [45] M. Daoudi, S. Berretti, P. Pala, Y. Delevoe, and A. Del Bimbo, "Emotion recognition by body movement representation on the manifold of symmetric positive definite matrices," in *International Conference on Image Analysis and Processing*. Springer, 2017, pp. 550–560.
- [46] B. Li, C. Zhu, S. Li, and T. Zhu, "Identifying emotions from non-contact gaits information based on microsoft kinetics," *IEEE Transactions on Affective Computing*, vol. 9, no. 4, pp. 585–591, 2016.
- [47] A. Crenn, R. A. Khan, A. Meyer, and S. Bouakaz, "Body expression recognition from animated 3d skeleton," in *2016 International Conference on 3D Imaging (IC3D)*. IEEE, 2016, pp. 1–7.
- [48] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, 2017.
- [49] A. Mehrabian, "Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament," *Current Psychology*, vol. 14, no. 4, 1996.