Detecting Human Engagement in Social Robots

Aditi Vijay Gode
Indiana University-Bloomington
Bloomington, Indiana
adigode@iu.edu

Kowshik Selvam
Indiana University-Bloomington
Bloomington, Indiana
kselvam@iu.edu

Mihir Ravindra Patel
Indiana University-Bloomington
Bloomington, Indiana
mihipate@iu.edu

Abstract

Facial Emotion Recognition(FER) is a challenging Computer Vision task to reveal information of a person's emotion state. They have a wide range of applications, and one among them is detecting facial emotions of toddlers in classrooms for the social assistive robot sitting in front of them to make a decision based on the revealed emotion. We explore various models to first detect discrete emotions on the FER2013 dataset, starting from the state-of-the-art architectures, and later move on to output the subtle aspects of emotions, namely valence and arousal, using a wide ensemble model.

1. Introduction

Humans are extremely expressive, and they can convey thousands of emotions without uttering a single word. Facial expressions are like a window to the mental state of a person and that emotional state of a person can influence in making decisions, solving a problem, and also gives an estimate of the level of concentration a person is putting into a task. Also, emotions are universal and the expressions of sadness, fear, happiness, disgust are the same for all the people around the world. Facial Emotion Recognition(FER) aims to help computers recognize human emotions.

Our idea is to be a part of the larger social engagement algorithm in **"Detecting Human Engagement in Social Robots".** After having a detailed conversation with Leigh Levinson (PhD Student, Psychology), we came up with this idea of detecting human emotions while they are interacting with the robot. We are building a model that detects the emotions of children in the age group of 4 to 12 when they interact with the socially assistive robot. Below is the setup provided by the psychology department in which we will be testing our model:

Figure 1 : The Robot present in the Artificial Intelligence Lab at Indiana University which will be used to test the existing model.

2. Background and related work

The previous works in the field of facial expression recognition includes facial image preprocessing, appearance feature extraction, which are basically not done through programming but manually and finally such features are classified [1]. These perform better only in the in-lab datasets but with the wild dataset it used to perform poorly. With increase in the development in deep learning as stated in [2], the features learned by Convolutional Neural Networks that are trained for facial expression prediction actually reflect the results of

emotions that are suggested by many psychologists.
ResNets are state of the art architecture that have been used several times due to their high complexity and readily available pre-trained models that have been trained on huge number of images to learn a number of features that can be transferred. There have been several works on image classification using ResNets[11] .Moreover, they have also been used for emotion classification [12]. which has about 90% accuracy, but the models are very complex and have been trained on huge datasets

Attention convolutional networks for emotion recognition have also been implemented in order to make networks that focus only on the important features about the face. [13].However, all of these models have been implemented on large datasets and require great hardware.

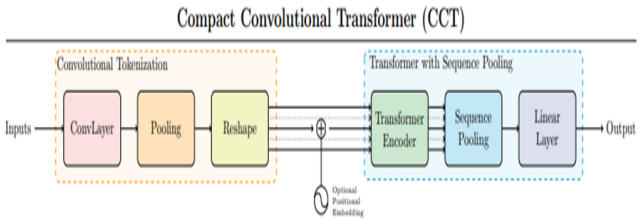

Figure 2: Compact Convolutional Transformer
Note: Figure adapted from paper[3]

Transformers in vision tasks give the benefit of global attention but they require big datasets and a large amount of pre-training. Also, they are difficult to train on machines with low computation power. Authors in paper[3] suggest that Compact Convolutional Transformers(CCT) can be used for datasets which are not as large in size as traditionally required by data-hungry transformers. They introduce convolutions in the tokenization stage and argue that introduction of convolutions eliminates the need for positional encoding. They also add a sequence pooling layer, by skipping the addition of class tokens before the sequence. The entire sequence of data is pooled over because there is important information across different parts of the image. The addition of sequence pooling makes the model compact. The result is the same 1 X D(classes) vector as obtained in the Vision Transformers(ViT). The authors were able to train CCT from scratch on CIFAR dataset with 60,000 images with an accuracy of 94.78% and with a lot less parameters. The authors were able to train ImageNet within 200 epochs with just 22.5 million parameters and get about 1% higher accuracy than the Vision Transformer.

The VGG16 architecture is one of the state-of-the-art architectures used in Computer vision. The input is a 224*224 image and the images are passed through a series of convolutional layers. There are 13 convolutional and 3 fully connected layers in the VGG16 architecture. As described by the authors of VGGNet in paper[vgg], VGG has smaller filters of size 3 X 3 with a stride of 1. After that, a max pooling layer is applied of 2 X 2 size with a size of 2 which reduces the height and width of the image to half. In this way, two convolutions for the first block, two for the second block, and three convolutions for the third, fourth, and fifth block are applied. Two fully connected layers follow this. Throughout the layers, the size of the image decreases but the number of feature maps increases. Each convolutional layer is followed by a ReLu(activation function), and a softmax function follows the final fully connected layer with the number of neurons the same as the required classes. The authors in paper[batchnorm] also mention how batch normalization can be used to train models with saturating nonlinearities, and how larger learning rates can be used with batch normalization to help the model train faster.

The AVCA approaches using the affective dimension dealt two ways of embedding the continuous affective states such as the valence and arousal range. This is done either using the features such as color, texture and motion that are being extracted from the dataset. Or from the emotional responses of the people while they are made to watch data stimuli. The paper [8] has implemented the first method where the continuous affective dimensions are represented using the features that are extracted from the data. To improve the AVCA's prediction accuracy, they try to correlate the relationship between the discrete emotion categories and the continuous affective emotions. By incorporating both the methods results in reducing the affective gap and this is one of the earliest papers to talk about this issue.

3. Methodology

In order to do the emotion classification, we have performed a study of different models on the FER dataset.

3.1 FER 2013

The dataset we have used for emotion classification is the FER-2013 dataset which is a dataset that consists of 48x48 pixel grayscale images of faces consisting of 7 different emotion categories. They have been labeled as: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral. The images have been cropped and adjusted such that the face is centered, and each face approximately occupies the same face in the image. FER2013 data was introduced by the ICML, and we use

the official training, validation, and testing sets. The total number of training images used are 28,708, whereas there are 3589 testing images and 3289 validation images. These images were in the form of CSVs having two columns, “emotion” and “pixels”, where the “emotion” column consists of numeric code from 0 to 7 indicating the emotion label and the “pixels” column consists of space separated strings indicating the pixel values of the images. The dataset is available on Kaggle[9] and was posted as part of a competition of facial emotion recognition. This dataset was prepared by Pierre-Luc Carrier and Aaron Courville, as part of an ongoing research project.

3.2. Implementation

3.2.1 Resnet18

The first model that we have implemented is the ResNet18[10] model which is an 18-layer deep model. This model has been implemented as a transfer learning[14] method that uses the pre-trained model available on PyTorch. The pre-trained model of ResNet has been trained for the ImageNet dataset consisting of 1000 classes. In order to work for our classification using transfer learning, we trained the last layer to classify the 7 classes of emotion. As the model has been pre-trained on a large dataset of more than a million images, the model has learned rich feature representations for different types of images leading it to be one of the state of the art architectures. Due to the increasing network, depth networks are very tough to train due to the problem of gradient vanishing due to the gradients reducing to extremely small values. ResNet avoids this problem by identifying shortcuts in the layers by skipping one or more layers in the network. Due to such advantages, we decided to try our classification using the pre-trained ResNet 18 model. A pre-trained model contains the weights and biases representing the features which are often transferable to different datasets. This helps us save time and lets us investigate where and how we can improve our model.

3.2.2 Attentional Convolutional Network

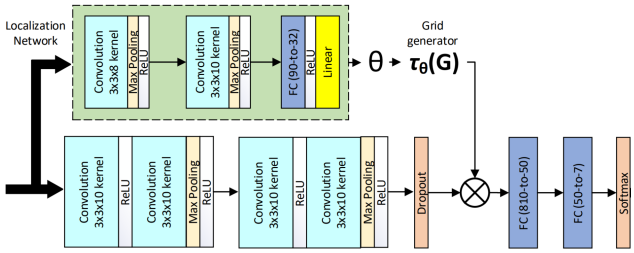

Figure 3 : Model Architecture[15]

The second model we have implemented is the attentional convolutional network for emotion classification.[15] Attentional Convolution Networks concentrate only on certain parts of the image that seem to be important and that help classify the images. This leads to a lesser number of parameters of the model leading to lesser memory and time required. Since emotions can be judged from specific parts of the face, there is no need for a model to look all over the image for the classification. In order to exploit this idea we have added an attention mechanism using spatial transformer network to focus on the main regions of the face. Figure 3 illustrates the architecture of the model. There are two parts to it, one of them is the feature extraction part whereas the other part is the fully connected network. The four convolution layers of the feature extraction part split into two where one of the parts goes through a localization network to learn the transformer matrix using the grid generator. This spatial transformer part focuses on the important parts of the image, essentially warping the input to the output using the grid $T(\theta)$. The transformer used is the affine transformation. The loss function to train the network used is Adam Optimizer, that consists of the regularization term and the classification loss. The implementation is similar to [16].

3.2.3. Compact Convolutional Transformer

Our FER2013 can be considered as a small sized dataset since it has just around 28,000 training samples with 3000 each in validation and testing. It was evident that a data-hungry Vision Transformer(ViT) would not have worked well in our case. We came across Compact Convolutional Transformer(CCT) which used less computational power and gave best results for small and mid sized datasets. The authors of the paper[3] were able to get 94.78% accuracy on the CIFAR-10 dataset with 6000 examples per class. Our dataset was smaller than that with as low as 500 examples for one class(emotion=Disgust) and with the maximum being 7000 examples for class 3(emotion=Happy). As far as we know, there is no record of Compact Convolutional Transformers(CCT) being applied to Facial Emotion Recognition(FER) tasks and we wanted to observe how efficiently does the CCT classifies emotions on small-sized datasets like FER2013.

3.2.4 VGG16 with additional layers

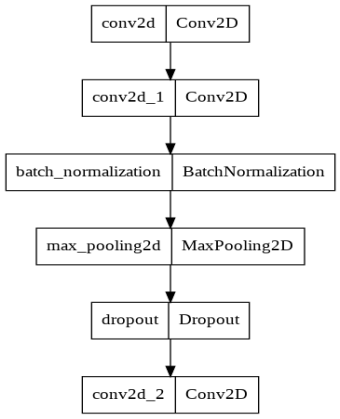

Figure 4 : First block of VGG net

We initially used VGG16 net as described in paper[4] but it gave accuracy of 52%. We, then, added a batch normalization layer after two convolution layers for the first two blocks, and then added a batch normalization layer after every convolutional layer from the third block. Additionally, we also experimented with different dropout rate after the max pooling and the fully connected layers(except the last layer) and chose the dropout rate of 30%.

3.3 Valence and Arousal

Since the subject here being toddlers, the other main research question would be what will be the emotion scale on a continuous range. Thus, adding a continuous emotion scale rather than providing the 7 discrete emotions(unlike in previously implemented models) will provide us with more insights regarding the subject. Hence to predict the valence and arousal range on the discrete emotion detection, we used the AffectNet database.

3.3.1. AffectNet

Affectnet database is considered to be the largest database that contains both the dimensional and categorical models. Also, this particular dataset of images are captured in the wild and not in the lab. There are about 450000 images with labels that contain both the discrete categories of the emotions and the continuous dimensional model which contains the valence and arousal range. In the discrete categorical emotion model, there are 11 categories defined which being Happy, Surprise, Sad, Neutral, Anger, Fear, Disgust, Contempt, Uncertain, Non-face and None. This includes 8 discrete emotions. There is annotation for valence and arousal level present. Valence indicates the negativity or positivity level of an image and arousal defines how exciting or soothing an event is.

3.3.2. Network Model

The Ensemble model that we have implemented utilizes the convolutional neural network properties. The main properties of convolutional neural networks are that they are translation invariant and by stacking up many layers of convolutional networks, they tend to learn the spatial hierarchies. In the case of emotion detection here, all the layers tend to learn the local visual patterns in the faces such as the colors and edges. Then, the layers that are following would learn the local features from these previous layers to detect mouth, eyes and nose. The last layer will predict the emotions. These are the main properties that are involved in the wide ensemble model that we have developed.

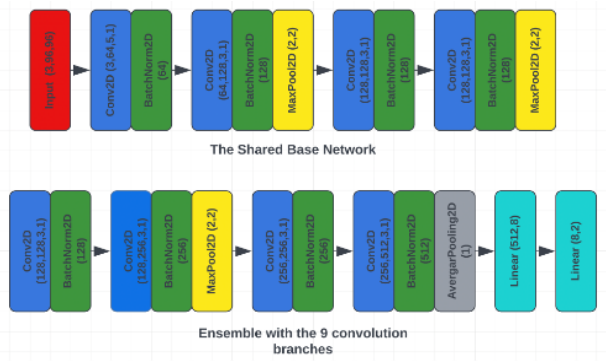

Figure 5: In the Figure on top, we have the base network, and, in the bottom, we have the ensemble with 9 convolution branches. Here after each batch normalization layer, the RELU activation function is applied. The last linear layer is added for the affect perception.

The main challenge that we faced was to choose the perfect branching level. As branching at the level 1 or level 2, that is early stages resulted in redundancy of low level facial features because the levels has to learn skin textures and so forth. But, if we do the branching at a very higher level then the features from these shared base layers don't correspond to the spatial facial features. So, we need to perform branching in the midway and hence we went with Level 3.

In our case, the wide ensemble model consists of 2 blocks where one being the network base Figureure 5 top block.

This is basically an order of convolutional layers for learning the features in the mid and the low level. The second block is an ensemble consisting of the independent convolutional branches. Here, in the ensemble of the convolutional branches each branch will learn distinctive features. The training in our model occurs by minimizing the loss function which is defined by:

$$L_{esr} = \sum_b \sum_I L [P (f(x_i) = y_i|x_i, \theta_{shared}, \theta_b), y_i] \quad (1)$$

In equation 1, (x_i, y_i) denotes the samples that are randomly taken from the training set, b points towards the index of the branch, q_{shared} is the parameter of the base of the network and q_b is the convolution branch parameter.
Here's how this algorithm works : First, the shared layers are initialized with q_{shared}, then we iterate over each branch of the ensemble where q_b is initialized to the convolutional branch, and then from the training dataset a subset is sampled. Now, iterating over a mini batch of the dataset we do the forward phase. Within each batch, we iterate over the existing branch, and we compute the loss using Equation 1. Finally, we do the backward phase and optimization.

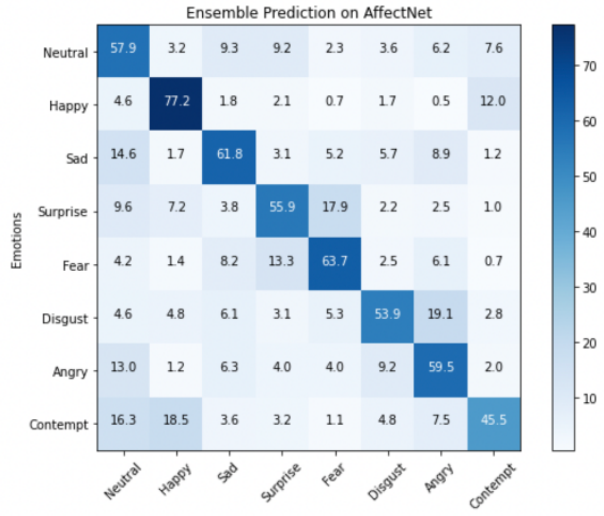

Figure 6 : The Confusion on the AffectNet dataset after performing Ensemble prediction

As you can see, from Figureure 1, each branching level consists of 8 convolutional levels, each followed by a batch normalization layer. The global average pooling layer transforms the last feature map into a vector and forwards it to a dense output layer for facial expression recognition.

Firstly, to detect the face from the images we used the dlib CNN face recognition model to put a bounding box and extract the face from any type of images. For now, we have implemented in such a way that if there are multiple faces detected then the algorithm will provide the face covering the largest pixel space in the image. While performing the discrete emotion detection, one of the main challenges that we faced was that the data for each distinct emotion was not balanced. So, each branch of the ensemble we trained with 6000 images from each emotion. To minimize the loss function we used Stochastic Gradient Descent(SGD). The learning rate that we used initially was 0.1 and the momentum of 0.9. After 10 epochs, we applied a learning decay of 0.5 multiplicative factor.

Now, in the case of the continuous affect perception, we predict the valence and arousal of the images of faces. Here, we fine-tuned the previous model to predict the discrete emotions to also predict the levels of the valence and arousal. We believed that rather than training the model from scratch, using the previous model would have already learned some facial features. and it would make the network to predict accurate results by learning at a faster rate. If you consider for example, if we train a face image which has a smiling feature then that particular dataset will correspond to positive arousal and valence levels. As in Figure 5, to include the valence and arousal predictions, instead of replacing the models output layer on top of each branch, we just add 2 neurons. The weights that are connected to those weights are only ones whc=ich are trained. Also, we do know that the relation between the continuous affect (valence and arousal), and the distinct emotions is non-linear. So, a ReLu function is applied to the second last layer.

For each branch of the ensemble, we train it on a balanced subset of around 5000 images for each emotion. Also, the valence and arousal are from the continuous domain and hence the problem here is a regression problem. Thus, using SGD we minimize the RMSE. To perform this, we use a learning rate of 0.01 and with a momentum of 0.9.

4. Results

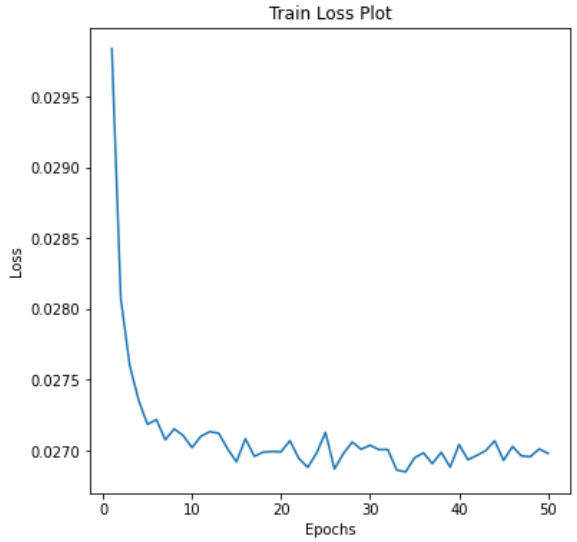

Figure 7 : Learning Curve for ResNet on FER2013

We achieved poor results for the ResNet 18 model during training as it plateaued at 50.45% accuracy during training and 40.23% accuracy for our test sets. We suspect that this is due to the high complexity of the network due to the increased number of layers and parameters(11M) of the pretrained network. So we decided to move on to other models where we could expect better results with lesser training time and complexity.

Attentional Convolutional network performed better than our previous ResNet18 model.After trying to change the hyperparameters various times we discovered that the optimal hyperparameter values were, 300 epochs with a learning rate of 0.005 and a batchsize of 128. This resulted in a training accuracy of 71.67% and testing accuracy of 54%.

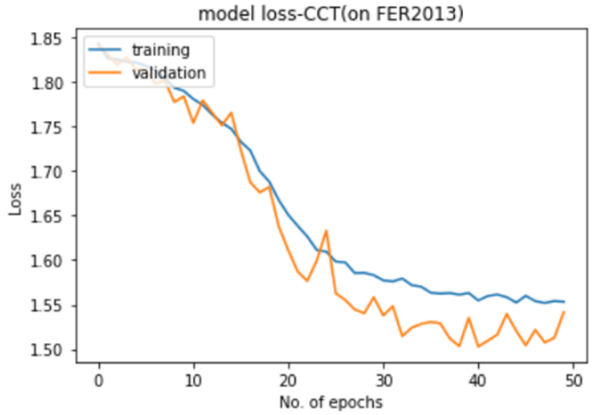

Figure 8: Learning curve for CCT on FER2013

We took the implementation of CCT from the Keras website and edited the code according to our requirements. We tried experimenting with different convolutional and transformer layers, tried removing positional embedding and checking the results. We also experimented while training the CCT on our data by trying different optimizers like RMSprop, SGD, different values of weight decay and learning rates. We tested our model by monitoring validation loss instead of validation accuracy and observed the results for different metrics for evaluation. The learning rate of 0.0001 and weight decay of 0.0001 gave the best results along with Adam optimizer and with categorical cross entropy as loss while monitoring on validation accuracy. We also tried adding a separate learning rate decay function because we saw that the training loss fluctuated a lot during the epochs. Because the dataset was extremely imbalanced, we also attempted different techniques to balance it. We assigned class weights to labels while training such that the classes which were underrepresented would be high weightage. This can be done manually, and also by a function from Sklearn. We tested our model with an oversampler from imblearn which brings all class labels to the same value by augmenting the classes with less number of labels. However, both of these methods did not help much for increasing the performance of our model. We got a training accuracy of 43.55%, validation accuracy of 47.2%, and testing accuracy of 46.81% for 50 epochs. We saw that the accuracy remained stagnant after 50 epochs.

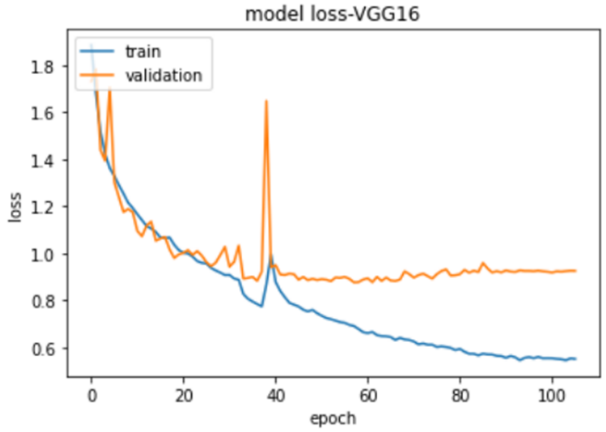

Figure 9: Learning curve for VGG16 on FER2013

The batch size for VGG16 was set to 64 and the augment data in a batch size of 64 was sent as the input. Furthermore, we opted for a learning rate decay by a factor of 0.5 when there is no increase in accuracy for 5 epochs. Our model was able to train using 33 million parameters using GPU on Google colab in 30 mins with the early stopping of 106 epochs. We were able to obtain a training accuracy of 79.71% and testing accuracy of

68%. The plot for loss on our customized VGG16 model shows that the model has started overfitting. The introduction of batch normalization layers in our VGG16 helped the training to speed up the training by a significant amount. We also used dropout after max pooling and dense layers which helped to overcome the problem of overfitting of the model to some extent.

Model	Training Accuracy	Validation Accuracy	Testing Accuracy	# of Parameters
VGGNet16	79.71%	70.23%	**68%**	33M
CCT	46.81%	47.21%	46.81%	406k
Attentional CNN	71.67%	52.58%	54%	100K
ResNet18	50.45%	45.50%	40.23%	11M

Figure 10: Comparative study of various models on FER2013 dataset for Facial Emotion Recognition(FER)

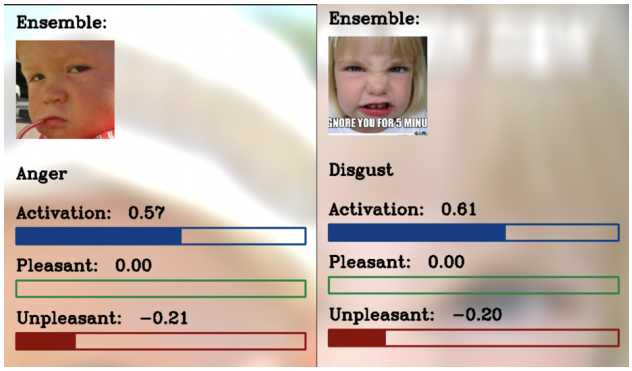

Figure 11: From the valence and arousal values obtained we initialized a threshold to obtain the Activation, Pleasant and Unpleasant value. The values are passed into a GUI to get this result.

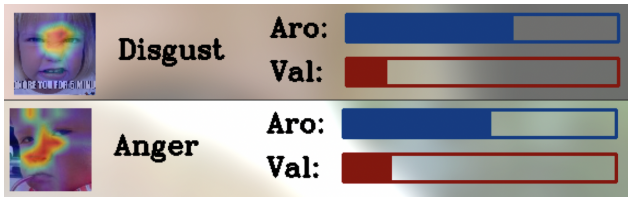

Figure 12: The Arousal and Valence Range in the last convolutional branch of the ensemble is passed through a GUI to get this result.

We found that the ensemble network with 9 convolutional branches achieved the highest accuracy with the AffectNet dataset. From Figureure 6 we can conclude that the ensemble's collective classification has reached an accuracy of 59.3%. The dimensions of valence and arousal achieved RMSEs of 0.33 and 0.36 respectively. Since we have only trained the output layer for the valence and arousal integration, this model can still perform the discrete emotion detection along with the continuous range of affect. This particular property results in a great reduction in the computation load and also huge reduction in redundancy. In Figure 10 and Figure 11 we passed the output data into an existing GUI to get those results showing the discrete emotion along with the Arousal and Valence range.

5. Discussion

The results obtained for the ResNet18 model were unsatisfactory. This was mostly due to the fact that the FER 2013 data was highly unbalanced as shown in Figure 13. The class1 images(disgust) consisted only of 400 images thus leading to bias towards other classes. We tried setting the class weights according to the number of images in the classes but couldn't improve on the accuracy. Due to this we decided to try other models, as the accuracy didn't seem to improve, it was consuming time, and other models implemented parallelly were yielding better results.

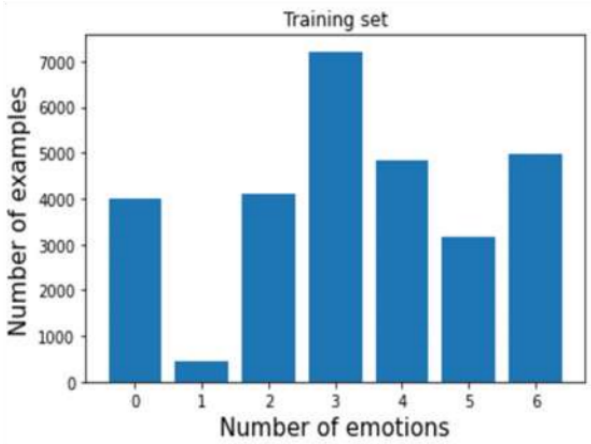

Figure 13: Distribution of Classes

The attentional Convolution Architecture was showing good results for the training dataset, but didn't perform as expected on the testing dataset. This may be due to the case of overfitting. However, this model consisted only of 100k parameters which was very low as compared to other models, thus requiring less memory and training time than others. This model can be used for small devices where there is not a lot of computation power required and less memory available and for purposes where certain errors are acceptable.

The plot for train and validation loss of CCT on FER2013 dataset shows that the model has no signs of overfitting. The testing and validation accuracy is greater than the training accuracy which says that the model is not

learning patterns from the training data very well and is becoming too generalized. As mentioned before, CCT was designed to work for small and mid-sized datasets and worked really well for CIFAR-10 with 60,000 samples but our dataset is almost half of that. This can be the reason why we are getting really low accuracy and our model is not able to perform better. To answer our research question if a Compact Convolutional Transformer can work well on face emotion recognition tasks, we observe that the model shows potential for good performance but requires far more data than we expected.

The plot for loss on our customized VGG16 model shows that the model has started overfitting. The introduction of batch normalization layers in our VGG16 helped the training to speed up the training by a significant amount. We also used dropout after max pooling and dense layers which helped to overcome the problem of overfitting of the model to some extent. Our experiment shows that simple state-of-the-art models like VGG16 with some modifications can also be used to recognize discrete emotions in humans.

6. Conclusion

We have trained, tested, and built various models to determine the emotions of toddlers when they interact with a social robot. Our main goal of this project is to develop the best model that not only determines the discrete emotions but also the continuous affect perception through detection of valence and arousal. We discovered that valence and arousal focus more on the subtle aspects of facial emotions are better metrics in determining the facial expressions of people. However, the datasets we trained our models on are images of people from all age groups and testing our models specifically on the dataset with images of just toddlers will give a more accurate measure of their performance.
Next, we are planning to collaborate with Leigh Levinson (PhD Student, Psychology) to build the complete application for the Social Robot where the toddlers could interact. According to the theory, thermal images are better inputs for recognizing involuntary expressions. Facial temperatures change when humans experience different emotions and they are also robust in absence of visible light. We are planning to integrate thermal images for detecting emotions in toddlers to keep them interactive with the robot. However, this is a challenging task since we will have to manually collect and label the data. Also, we believe our algorithm and model will serve as the first step toward detecting emotions using a thermal camera for the social robot to better understand the children in the autism spectrum. Also, to deal with such complex topics rather than just taking the cross model learning of the emotional expressions we should also take into consideration the contextual and also the temporal information during the phase of emotion detection.

7. References

[1] Tian, Y.-L.; Kanade, T.; and Cohn, J. F. 2005. Facial Expres-sion Analysis. New York, NY: Springer New York. 247–275. Ullman, S. 2019. Using neuroscience to develop artificial intelligence. Science 363(6428):692–693.

[2] Khorrami, P.; Paine, T. L.; and Huang, T. S. 2015. Do deep neural networks learn facial action units when doing expression recognition? In IEEE on CVPR - Workshops, 19–27.

[3] Hassani, A., Walton, S., Shah, N., Abuduweili, A., Li, J., & Shi, H. (2021). Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*.

[4] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[5] Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). PMLR.

[6] Siqueira, H., Magg, S., & Wermter, S. (2020, April). Efficient facial feature learning with wide ensemble-based convolutional neural networks. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 04, pp. 5800-5809).

[7] Mollahosseini, A., Hasani, B., & Mahoor, M. H. (2017). Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, *10*(1), 18-31.

[8] Zhang, L., & Zhang, J. (2017, July). Synchronous prediction of arousal and valence using LSTM network for affective video content analysis. In *2017 13th international conference on natural computation, fuzzy systems and knowledge discovery (ICNC-FSKD)* (pp. 727-732). IEEE.

[9] https://www.kaggle.com/competitions/challenges-in-representation-learning-facial-expression-recognition-challenge/data

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun," Deep Residual Learning for Image Recognition"

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages1097–1105,2012.

[12] "Image classification based on RESNET "September 2020Journal of Physics Conference Series 1634(1):012110 DOI:10.1088/1742-6596/1634/1/012110

[13] Bin Li, Dimas Lima,Facial expression recognition via ResNet-50,International Journal of Cognitive Computing in Engineering,Volume 2, 2021,Pages 57-64,ISSN 2666-3074, https://doi.org/10.1016/j.ijcce.2021.02.002.(https://www.sciencedirect.com/science/article/pii/S2666307421000073)

[14] Comprehensive Survey on Transfer Learning"
Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, Qing He

[15] “Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network” Shervin Minaee, AmiraliAbdolrashidi, https://doi.org/10.48550/arXiv.1902.01019
[16] https://github.com/omarsayed7/Deep-Emotion