Multi Modal Emotion Recognition

Final Year Endsem project submitted in partial fulfilment of the requirements for the degree of B.Tech.

by

Student B. Bavesh.
(Roll No: COE18B007)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,
DESIGN AND MANUFACTURING, KANCHEEPURAM

May 2022

Certificate

I, **B Bavesh**, with Roll No: **COE18B007** hereby declare that the material presented in the Endsem Project Report titled **Multi Modal Emotion Recognition** represents original work carried out by me in the **Department of Computer Science and Engineering** at the **Indian Institute of Information Technology, Design and Manufacturing, Kancheepuram** during the year **2022**. With my signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property. I have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I have understood that any false claim will result in severe disciplinary action.
- I have understood that the work may be screened for any form of academic misconduct.

Date: 8/5/2022 Student's Signature: Bavesh

In my capacity as supervisor of the above-mentioned work, I certify that the work presented in this Report is carried out under my supervision, and is worthy of consideration for the requirements of endsem project work during the period Jan 2022 to May 2022.

Advisor's Name: Dr. V Masilamani Advisor's Signature

Abstract

Human ideas and sentiments are reflected in facial expressions. It gives the viewer a plethora of social cues, such as the focus of attention, intention, motivation, and emotion. It is regarded as an effective means of quiet communication. The analysis of these expressions provides a far more in-depth understanding of human behaviour. In recent years, AI-based Facial Expression Detection (FER) has emerged as a critical research issue, with applications in dynamic analysis, pattern recognition, interpersonal interaction, mental health monitoring, and many other areas. However, with the global shift to online platforms as a result of the Covid-19 pandemic, there has been a compelling need to develop and provide a new FER analysis framework with the increased visual data provided by videos and images, as well as other data such as EEG and ECG signals. This study focuses on creating a novel and adept AI-based solution for facial emotion recognition on different modes of data, such as images, videos and EEG signals. To address this problem statement on visual data, we employ a transformer-based network and study its effects on the data. Further, the solution is extended to different modes of data.

Acknowledgements

I would like to empress my utmost gratitude to my external guide Dr. Partha Pratim Roy and IIT Roorkee for the opportunity and their sincere guidance. I would also like to convey my sincere thanks to Dr. Masilamani for being an encouraging and edifying guide throughout the entirety of the Project. I would also like to thank IIITDM Kancheepuram for enabling me to work under a different atmosphere and for the appropriate time period.

Additionally, I would like to thank my parents, teachers and my friends for being supportive and helpful.

Contents

List of Figures

List of Tables

Abbreviations

FER	**F**acial **E**motion **R**ecognition
MLP	**M**ulti **L**ayer **P**erceptron
ViT	**Vi**sion **T**ransformer
CNN	**C**onvolutional **N**eural **N**etwork
RNN	**R**ecurrent **N**eural **N**etwork
LSTM	**L**ong **S**hort **T**erm **M**emory
ResMask	**Res**idual **Mask**ing network
DeiT	**D**ata **E**efficient **I**mage **T**ransformer
MSA	**M**ulti head **S**elf **A**ttention

Chapter 1

Introduction

1.1 Background

People are surrounded with human facial expressions that they can see. They are natural signs that assist them in understanding emotions from someone in front of them, as well as from photographs or films. These emotions are extremely complicated and difficult for machines to comprehend, yet they are simple for humans to comprehend. Mehrabian, a famous psychologist, discovered from his studies that the emotional data that humans define as emotions is arranged in portions, which helped him understand how humans might grasp such feelings. He discovered that language only transmits 7% of overall emotional data, whereas our language auxiliary, which varies by culture, transports 38%, such as speech rhythm, tone, pitch, and so on. So far, the largest proportion of emotional data exhibited by facial expression is 55%. This suggests that numerous useful emotional data may be gathered by identifying facial emotions, which successfully comprehend any human's state of mind and activities that are directly related to emotions. As a result, it is critical to delve deeper into this research domain, as less accurate systems impede large - scale production.

In recent years, the study of various brain signals and their patterns has also been subjected to a lot of research. Studies in the field of neuroscience has found a link between emotional patterns and brain functional regions, suggesting that dynamic

interactions between different brain regions are an important aspect in emotion detection as measured by electroencephalography (EEG). The activations of different parts of brain during certain trigger actions are determined by closely monitoring these signals. As a result, it is essential to conduct substantial research in this direction and find the effects of these signals in other tasks that depict the mental state of humans, such as emotion recognition.

1.2 Motivation

Human face expression detection has been widely employed in a wide range of human-computer interactions, including smartphones, affective computing, intelligent control systems, psychological and behavioural research, pattern searching, defence, social media, robotics, and other domains. By assessing these feelings, it is possible to provide maximum user happiness and feedback to enhance present technology. This is only possible in the fields of computer vision and deep learning. Because of the global shift towards online platforms such as online education to teach or gain knowledge virtually globally to all remote areas, IoT enabled health monitoring systems and temperature setters in cars and households, robotics, psychiatric evaluation based on violent behaviours of criminals or those mentally disturbed, mood swings study on adolescents to help guide them mentally, deep-learning In addition, numerous studies on Facial Emotion Recognition using Computer Vision have been conducted due to its practicality in intelligent robotics, health-related treatment, IoT, security surveillance, criminal psychological analysis, driver exhaustion observation, and other human computer interface mechanisms.With greater virtual connectedness via videos and photos, the necessity to implement cutting-edge technology based on people's emotions is now a vital aspect in maximising user-friendliness and happiness.

1.3 Objectives of the work

- Improve emotion recognition on visual data, i.e, images and videos.

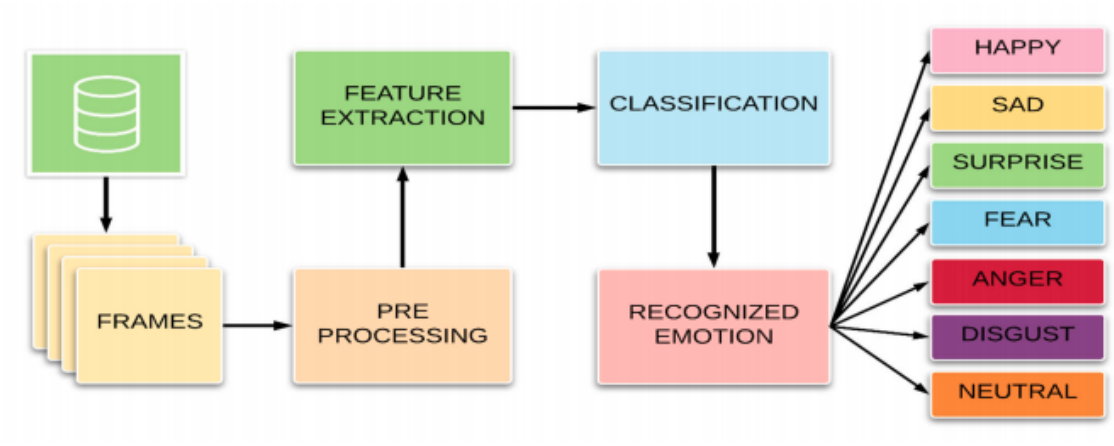

FIGURE 1.1: Facial Emotion Classification Process

- Extend the solution to other signals such as EEG.

Chapter 2

Related Works

Many image processing methods for feature extraction and clustering techniques like K-Nearest Neighbours have been employed in earlier studies.

Attributed to the advancements in Machine Learning and Deep Learning techniques and computation power, and the availability of numerous FER datasets that are open sourced, the old methods are being updated by various machine learning architectures. [1, 2] implement CNN-based architectures, while [3] implements U-Net inspired Residual Masking network. This consists of an encoder decoder system just like the U-Net. However, that whole block is used as a mask and is combined with normal residual blocks to capture more accurate features after every residual block.

Fan et al. 2016[4] uses RNN, initially employed to process text, along with CNN's to classify emotions from images. Further, Bi-LSTMs, a special-kind RNN that is capable of reading the input sequences from either side, is also used for emotion recognition[5]. However, RNNs-based models, especially Bi-LSTM, frequently have issues because to their Billion-number of parameters and gradient-related problems in training. Furthermore, an RNN design processes time series sequentially, making parallel inference a tough task.

CNN-based architectures do not encounter the aforementioned challenges, and do perform pretty well. However, none of the above models use the concept of attention and the relationship between different parts of an image to classify emotions. Pecoraro et al.[6] 2021 uses the concept of self-attention, but on the different features obtained from convolutional

neural networks. None of the existing methods for emotion recognition have used self-attention on the image itself.

To overcome the problems of RNN and LSTM-based models and to improve the accuracy produced by the existing SOTA, this research focuses on employing vision transformers to recognize emotions. Vision Transformers, developed by Kolesnikov et al. 2021[7], is a simple tweak on the actual transformers encoder mode(Vaswani et al. 2016)[8] designed primarily for NLP applications. Here, we split an image into multiple patches and embed them and find the relationships between different patches using self attention.

With regards to emotion recognition using EEG data, various studies have been performed over the years to solve EEG emotion recognition using machine learning. Nie et al. [9] used a linear dynamic system to smoothen the extracted features from EEG, and then used a SVM(support vector machine) to classify. Lin et al. [10] extracted five-band power spectrum features from EEG data, and then used MLP and SVM to classify.

Over the recent years, with the improved performance of deep learning on tasks such as target detection and speech recognition, many of them have started to use the same for EEG emotion recognition. Alhagry et al. [11] proposed a LSTM-based network to capture the temporal features and classify EEG data. Xing et al. [12] applied a stack auto encoder(SAE) and a LSTM model to build a hybrid model, which achieved excellent performance on the DEAP dataset. With the advent of newer types of neural networks such as graph neural networks, researchers have also mapped different parts of brain and represented them in the form of a graph [13].

There have been different methods of feature extraction from EEG [14]. For instance, Li et al. [15] use the method of mapping the electrodes to a 2-dimensional matrix, thereby mapping all the brain signals to 2 dimensional data. Wang et al. [16] create EEG spectrograms using short-time fourier transform(STFT). However, limited research has been conducted in using topography for mapping the brain signals to images. Hence, we would like to focus our research in that direction and study the effects of this mapping technique.

Chapter 3

Work Done

3.1 Data

The dataset used for emotion recognition on images is the FER2013 dataset. The data consists of 48x48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centered and occupies about the same amount of space in each image. The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples. The public test set used for the leaderboard consists of 3,589 examples. The final test set, which was used to determine the winner of the competition, consists of another 3,589 examples. A Kaggle forum discussion held by the competition organizers places human accuracy on this dataset is 68±5%.

3.1.1 Class Imbalance

Upon plotting the number of images of each emotion, it is evident that the dataset is imbalanced towards the 'happy' emotion. There are approximately 9000 happy images. On the other hand, there are only 900 images of the 'disgust' emotion. This makes the dataset difficult to work with.

FIGURE 3.1: FER2013 dataset

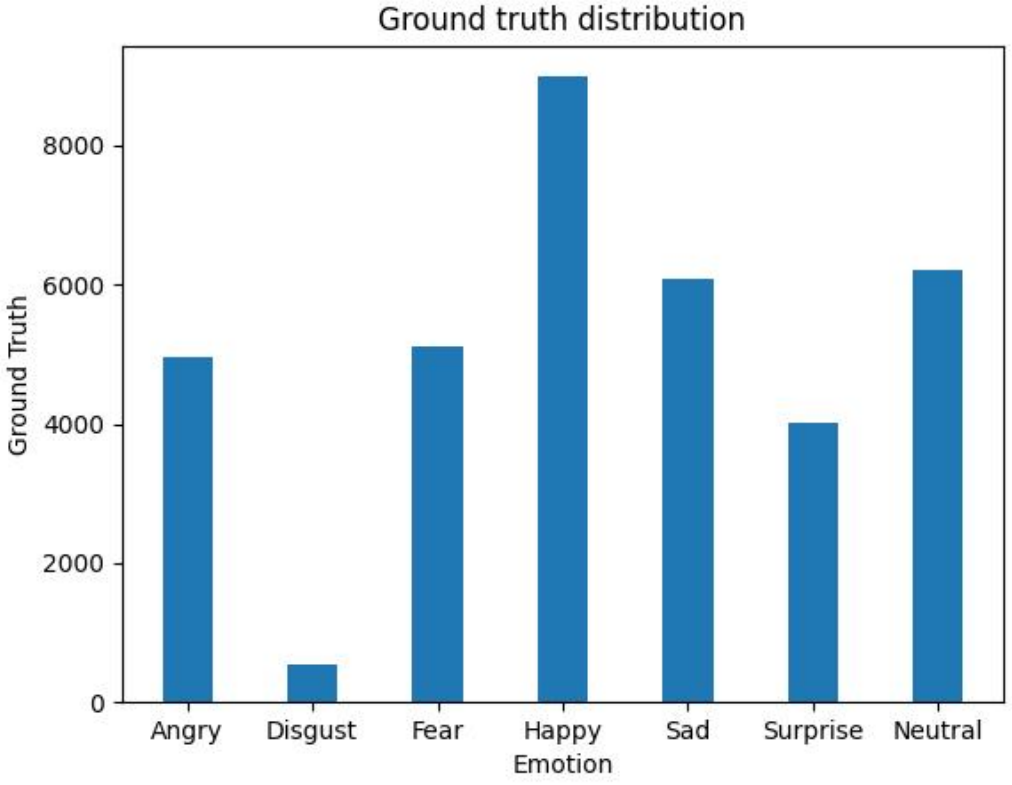

FIGURE 3.2: count of images of each class

3.1.2 Age Bracket

This dataset primarily consists of images of adults. Hence, this dataset is not holistic in regards to age.

3.2 Implementing Existing State-of-the-art

In an attempt to understand the subtleties and nuances behind the state-of-the-art models, we built the Residual Masking Model[3] and the hyperparameter tuned VGG-16 network from scratch and trained them. This helped us understand the reason behind the success of these models and the key takeaways these solutions had to offer.

3.2.1 Residual Masking Network

The residual masking network is a network that consists of multiple residual and masking blocks. The purpose of the residual blocks is to find out important features with the help of residual connections, so that no extra compute resources are used. The masking block follows a U-Net based encoder-decoder network which reconstructs the entire feature map, giving attention to the more important features while reconstructing.

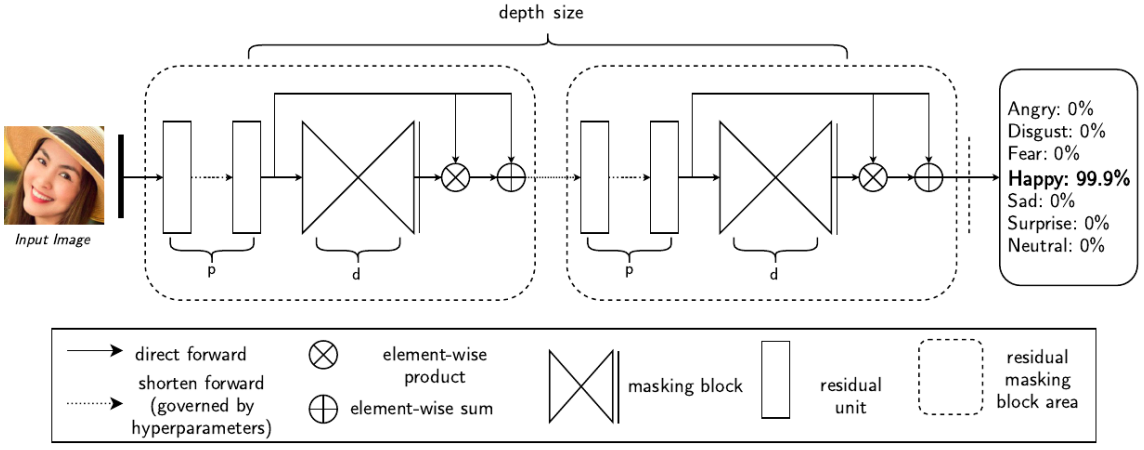

FIGURE 3.3: Architecture of ResMasking Network

3.2.1.1 Results

data_path	data
image_size	(224, 224)
num_classes	7
in_channels	3
architecture	resnet34
optimizer	SGD
loss function	Cross Entropy
learning rate	0.0001
weighted_loss	0
momentum	0.9
weight_decay	0.001
batch_size	48
epochs	50
plateau_count	8
plateau_patience	2
stepLR	50

TABLE 3.1: ResMasking training hyperparameter details

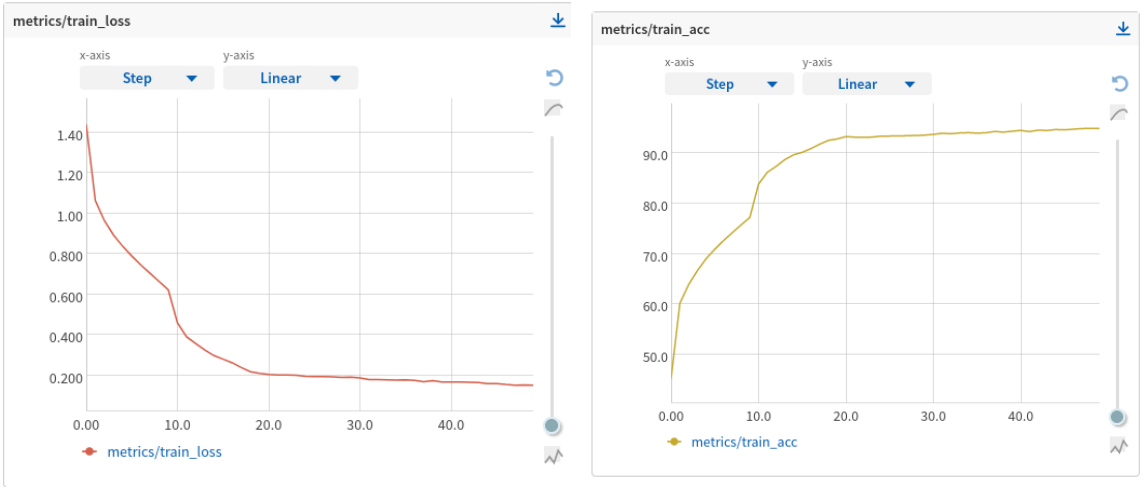

FIGURE 3.4: Training Loss and Accuracy

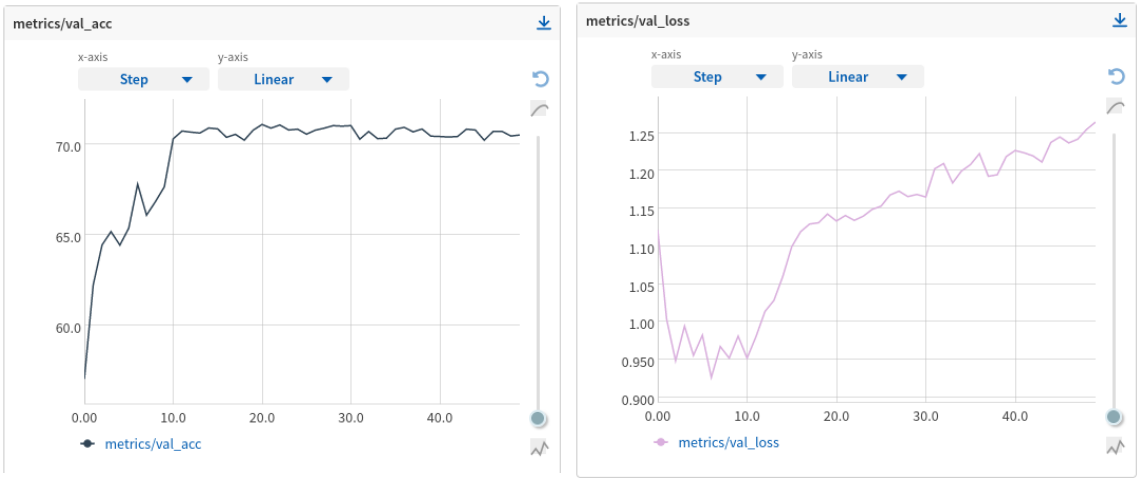

FIGURE 3.5: Validation Loss and Accuracy

3.3 Our Solution

Given the recent success of the Vision transformers on computer vision tasks, and the soft inductive bias that it brings with itself, we decided to use transformer-based models for our downstream task, facial emotion recognition. We used several different models such as Vision transformers [7], Data-Efficient Image Transformers [17]. We also tried out models that combined both convolutional neural networks and transformers, such as ConvNeXt [18] and ConViT [19].

3.3.1 Vision Transformer

This is a transformer model inspired from the Vaswani et al [8] transformer. Here, every image is split into square patches of a particular size, and the relationship between different patches are captured using self-attention. Unlike the encoder-decoder model used in Vaswani et al, Vision transformer uses only the encoder layer, followed by a multilayer perceptron.

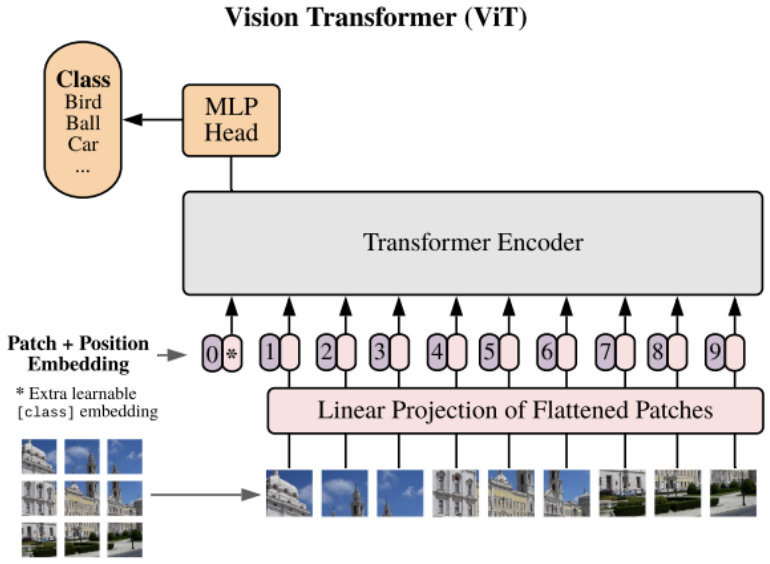

FIGURE 3.6: Architecture of Vision Transformer

3.3.1.1 Training Details

In this section, we provide details the training strategies that we used to train vision transformers. We build upon PyTorch and the timm library. We provide hyper-parameters and information on all the different techniques that were employed in our training experiments.

Initialization and hyper-parameters We had tried out various initialization techniques such as Xavier initialization, He initialization etc. After testing them and some of them not converging, we followed Hanin and Rolnick's recommendation of using truncated normal distribution. As for hyper-parameters, table 3.2 indicates the hyper-parameters used while training our ViT model.

Data Augmentation Transformers, in comparison to CNN based models, require larger amounts of data. Hence, extensive reliance on data augmentation is required while training. We used various types of transformations such as RandomCrop, Cutout, MixUp and translate. RandomHorizontalFlip, Random Rotation and random erasing

worked the best on our model. All the data augmentations were performed using Pytorch's transforms library.

Regularizers and optimizers Numerous regularization techniques like Dropout and MixUp were tried and tested. However, none of them, other than the LayerNorm used in the vanilla Vision Transformer gave us improved results. With regards to optimizers, all the experiments were predominantly conducted with SGD and Adam. Even though SGD gave us quicker convergence, we found Adam to give us better performance for this downstream task. Sharpness-aware minimization was also used, but was not found to give any tangible result.

Training Time Training the model from scratch for 85 epochs approximately took 8 hours, when trained in a distributed manner across 2 V100 GPUs. PyTorch's distributed data parallel training was used to train the model using multiple GPUs.

data_path	/data/Bavesh Balaji/ViT/utils/fer2013.csv
image_size	(224, 224)
patch size	(16, 16)
num_classes	7
in_channels	1
architecture	Vision Transformer
optimizer	Adam
loss function	Cross Entropy
learning rate	0.0003
weighted_loss	0
momentum	0.9
weight_decay	0.0001
batch_size	64
epochs	85
plateau_factor	0.75
plateau_patience	5
scheduler	ReduceLROnPlateau
embedding dimension	768
checkpoint directory	/data/Bavesh Balaji/ViT/checkpoints

TABLE 3.2: Vision Transformer training details

3.3.1.2 Results

The ViT model was trained on different data augmentations, weight initialization and regularization techniques. The baseline model with no augmentations or regularziers gave us a 40% validation accuracy. However, after incorporating the essential data augmentation and initialization techniques, the model was able to give us a 52% validation accuracy and a 54% test accuracy.

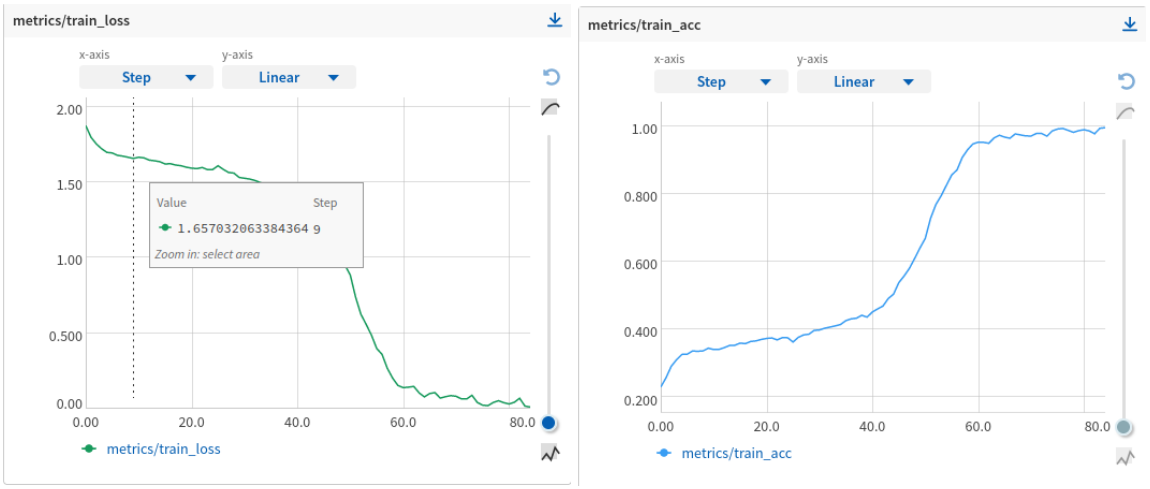

FIGURE 3.7: Training Loss and Accuracy

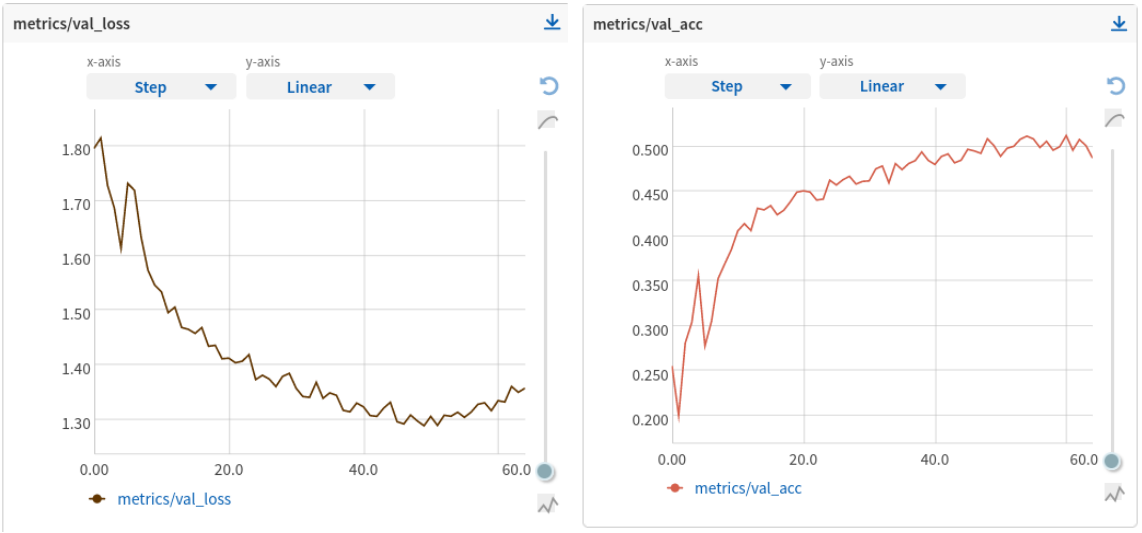

FIGURE 3.8: Validation Loss and Accuracy

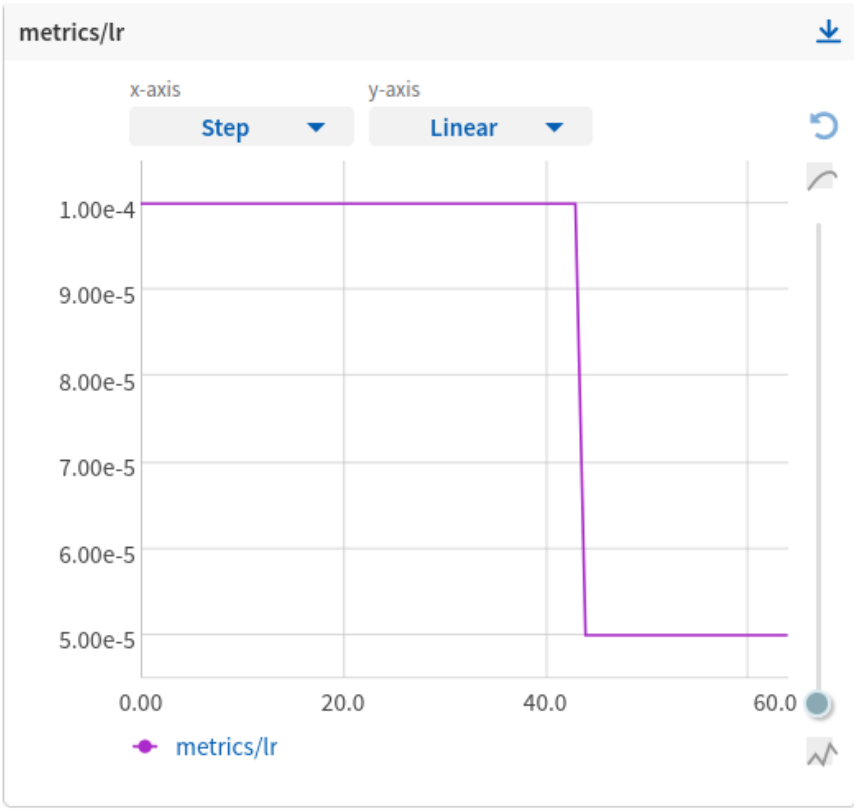

FIGURE 3.9: Learning Rate

3.3.2 Data Efficient Image Transformers - DeiT

The use of Vision transformers for our downstream task proved to be ineffective, producing a mere 54% test accuracy, in comparison to the state-of-the-art(74% accuracy). One of the reasons for this poor performance could be the insufficient training amounts of data. Vision transformer are known to not generalize well to small amounts of data. Hence, we decided to use the Data Efficient Image transformer. DeiT is a modified version of ViT, which is optimized to work well for smaller amounts of data.

3.3.2.1 Training Details

In this section, we provide details the training strategies that we used to train Data efficient image transformers. We build upon PyTorch and the timm library. We provide

hyper-parameters and information on all the different techniques that were employed in our training experiments.

Initialization and hyper-parameters We had tried out various initialization techniques such as Xavier initialization, He initialization etc. After testing them and some of them not converging, we followed Hanin and Rolnick's recommendation of using truncated normal distribution. As for hyper-parameters, table 3.3 indicates the hyper-parameters used while training our DeiT model.

Data Augmentation We used various types of transformations such as RandomCrop, Solarize and shear. RandomHorizontalFlip, Random Rotation, CutMix and random erasing worked the best on our model. All the data augmentations were performed using Pytorch's transforms library.

Regularizers and optimizers Numerous regularization techniques like Dropout and MixUp were tried and tested. Stochastic Depth and Layer Norm gave us improved results. With regards to optimizers, all the experiments were predominantly conducted with SGD, Adam, AdamR, AdamW and Sharpness-aware minimization(SAM). Even though SGD gave us quicker convergence, we found AdamW to give us better performance for this downstream task.

Training Time Training the model from scratch for 145 epochs approximately took 12 hours, when trained in a distributed manner across 2 V100 GPUs. PyTorch's distributed data parallel training was used to train the model using multiple GPUs.

data_path	/data/Bavesh Balaji/ViT/utils/fer2013.csv
image_size	(224, 224)
patch size	(16, 16)
num_classes	7
in_channels	1
architecture	Data Effecient Image Transformer
optimizer	AdamW
loss function	Cross Entropy
learning rate	0.00015
weighted_loss	0
momentum	0.9
weight_decay	0.0001
batch_size	64
epochs	145
plateau_factor	0.75
plateau_patience	5
scheduler	ReduceLROnPlateau
embedding dimension	768
checkpoint directory	/data/Bavesh Balaji/ViT/checkpoints

TABLE 3.3: Data Efficient Image Transformer training details

3.3.2.2 Results

The model was trained for 145 epochs on 2 GPUs. The training accuracy was constantly increasing and reached 90% in the end. The validation accuracy initially showed a smooth and monotonically increasing curve, for the first 10 epochs. However, it then started increasing slowly, and finally started fluctuating near 50%. We should have trained for another 50 epochs for the model to be trained fully, but we decided to stop the training since the validation accuracy was not improving for a long period of time.

Our DeiT model achieved a validation accuracy of 51.6% and test accuracy of 52.1%.

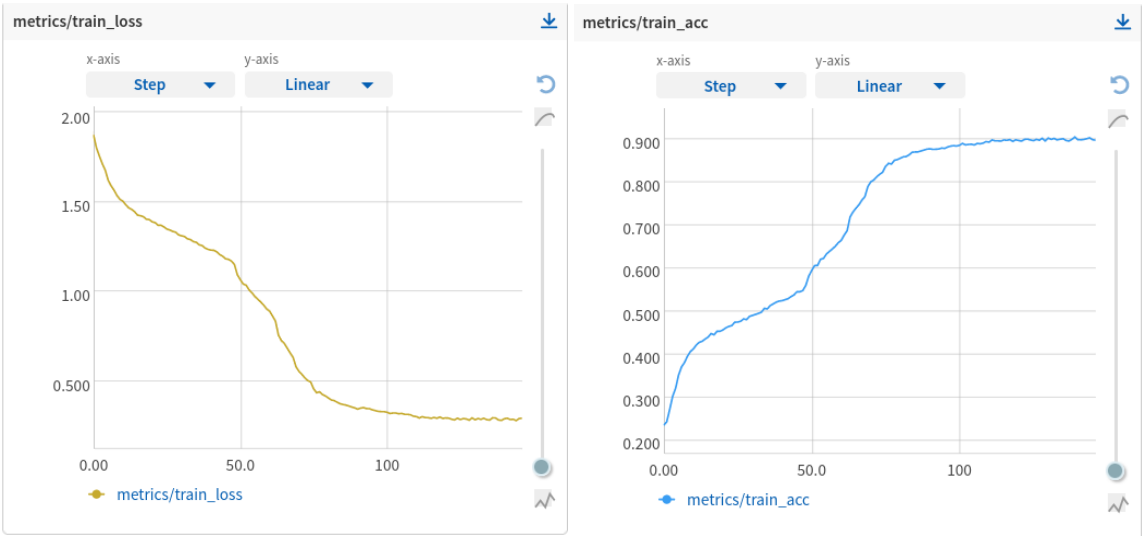

FIGURE 3.10: Training Loss and Accuracy

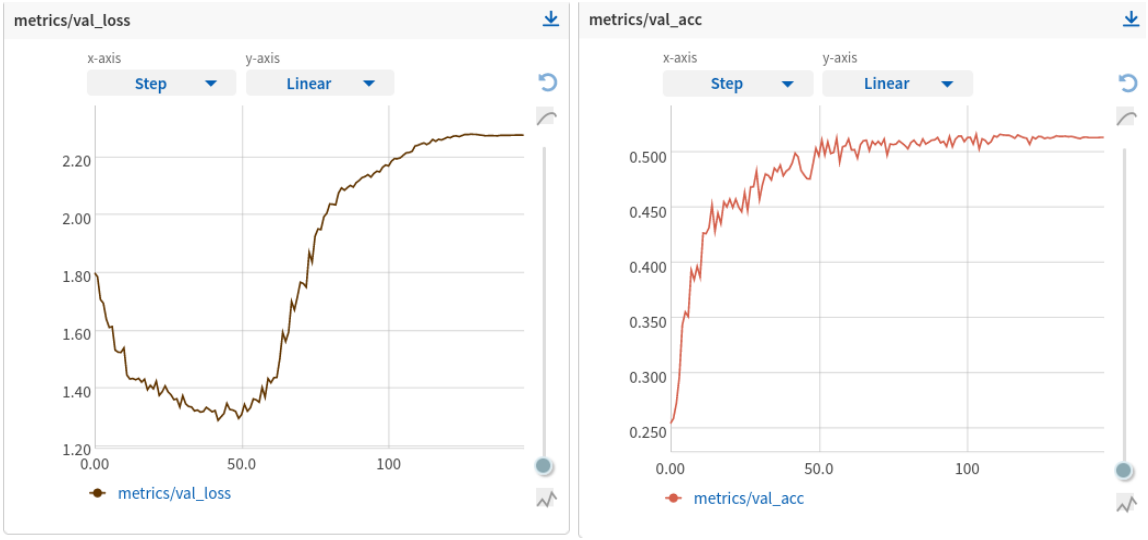

FIGURE 3.11: Validation Loss and Accuracy

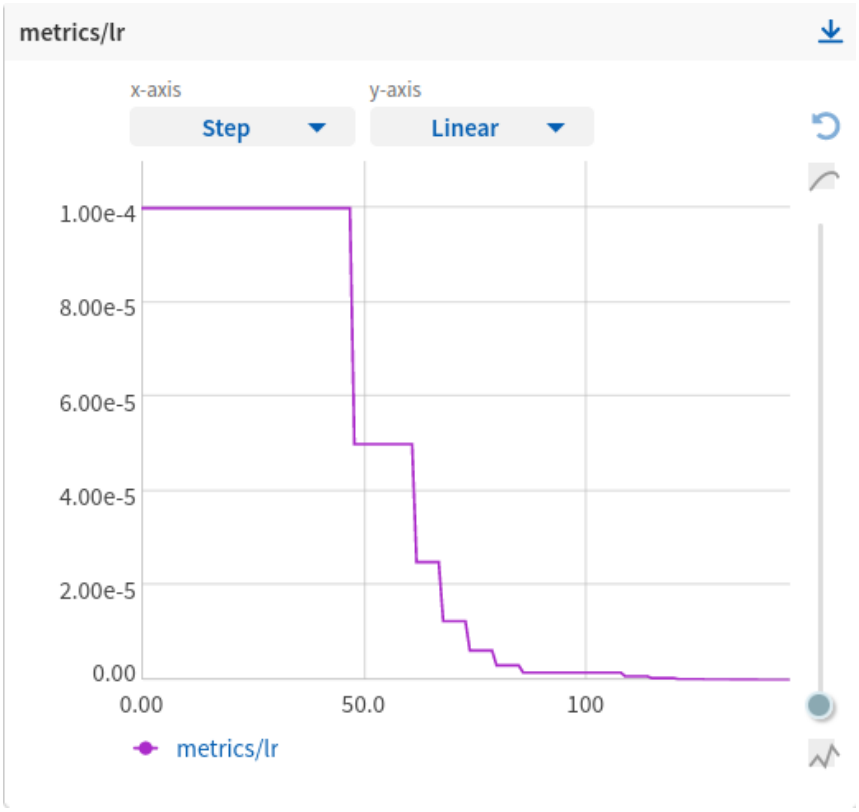

FIGURE 3.12: Learning Rate

3.3.3 CNNs vs Transformers

The low performance of ViT and DeiT in our emotion recognition task indicated the ineptitude of transformer-based models in comparison to convolution neural network based models on our downstream task. As a result, we decided to use the inherent advantage that the CNN-based models provide along with the unique and distinctive ideas that the transformers have used for better performance. The hard inductive bias that CNNs provide will allow the model to learn the features on small amounts of data quickly, while the ideas incorporated from transformers improves the generalizability and the robustness of the model.

3.3.4 ConvNeXt

In this section, we explain the model architecture that we have used for incorporating both CNNs and transformers into one model. We continue by providing the training strategies for this model and the results on our dataset.

ConvNeXt(Liu et al) is an entirely CNN based network that uses the key components of vision transformers to improve the robustness of the model. It is a modernized version of the ResNet model, where the design is changed to bring its design closer to hierarchical transformers(such as Swin Transformer). Some of the features incorporated are the non-overlapped convolutions used in the stem(similar to the only convolutional layer in transformers) and moving the positin of depthwise convoltuion layers up(similar to the pacement of MLP above the MSA module in transformers).

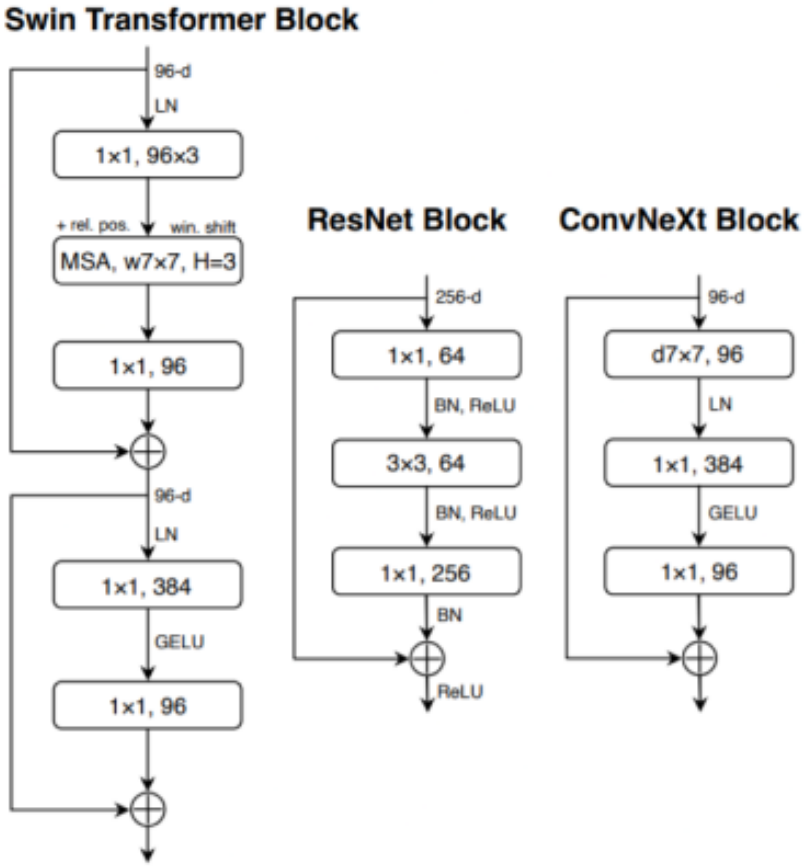

FIGURE 3.13: Architecture of Swin Transformer, ResNet and ConvNeXt

3.3.4.1 Training Details

In this section, we provide details the training strategies that we used to train Data efficient image transformers. We build upon PyTorch and the timm library. We provide hyper-parameters and information on all the different techniques that were employed in our training experiments.

Initialization and hyper-parameters For this model, we used the default initialization method in the paper, truncated normal distribution with a standard deviation of 0.02. As for hyper-parameters, table 3.4 indicates the hyper-parameters used while training our ConvNeXt model.

Data Augmentation We used various types of transformations such as RandomCrop, Solarize and shear. RandomHorizontalFlip, Random Rotation, Rand Auto Augment, CutMix and random erasing worked the best on our model. All the data augmentations were performed using Pytorch's transforms and the timm library.

Regularizers and optimizers In this model, the default Layer Norm was used as this was one of the features that was incorporated into the CNN-based model from a transformer. With regards to optimizers, all the experiments were predominantly conducted with SGD, Adam, AdamR, AdamW and Sharpness-aware minimization(SAM). Even though SGD gave us quicker convergence, we found AdamW to give us better performance for this downstream task.

Training Time Training the model from scratch for 110 epochs approximately took 13 hours, when trained in a distributed manner across 2 V100 GPUs. PyTorch's distributed data parallel training was used to train the model using multiple GPUs.

data_path	/data/Bavesh Balaji/ViT/utils/fer2013.csv
image_size	(224, 224)
patch size	(16, 16)
num_classes	7
in_channels	1
architecture	ConvNeXt
optimizer	AdamW
loss function	Cross Entropy
learning rate	0.0001
weighted_loss	0
momentum	0.9
weight_decay	0.0001
batch_size	64
epochs	110
plateau_factor	0.75
plateau_patience	5
scheduler	ReduceLROnPlateau
embedding dimension	768
checkpoint directory	/data/Bavesh Balaji/ViT/checkpoints

TABLE 3.4: ConvNeXt training details

3.3.4.2 Results

The model was trained for 110 epochs on 2 GPUs. The training accuracy was steadily increasing and reached 99.18% in the end. The validation accuracy initially showed a smooth and monotonically increasing curve, for the first 15 epochs. However, it then started increasing slowly, and finally started plateauing near 66%.

Our ConvNeXt model achieved a validation accuracy of 67.01% and test accuracy of 70.2%.

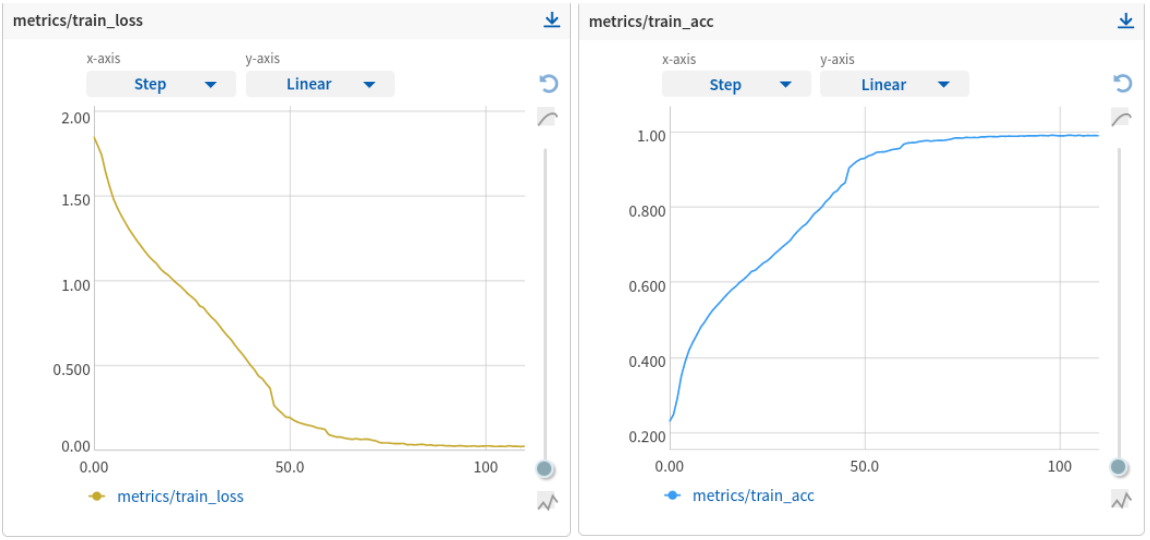

FIGURE 3.14: Training Loss and Accuracy

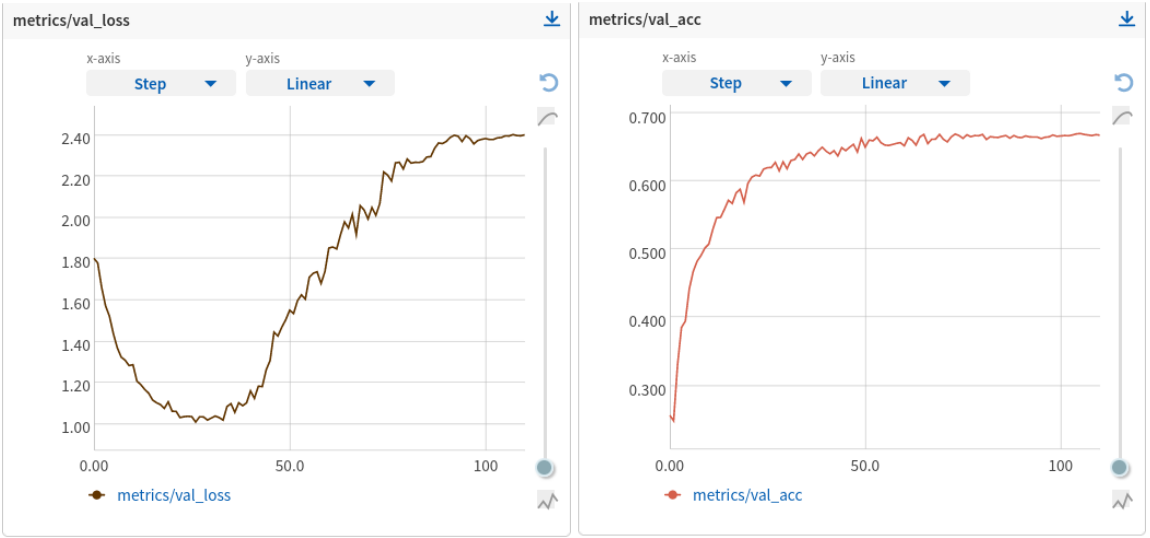

FIGURE 3.15: Validation Loss and Accuracy

3.3.5 EEG Emotion Recognition

Electroencephalography is a method to record the electrical activity of the scalp, which has been to represent the macroscopic activity of the surface layer of brain underneath. It essentially records the electrical fluctuations within the neurons of the brain. These fluctuations are shown to represent the activity of the brain, and can be used for diagnostic purposes.

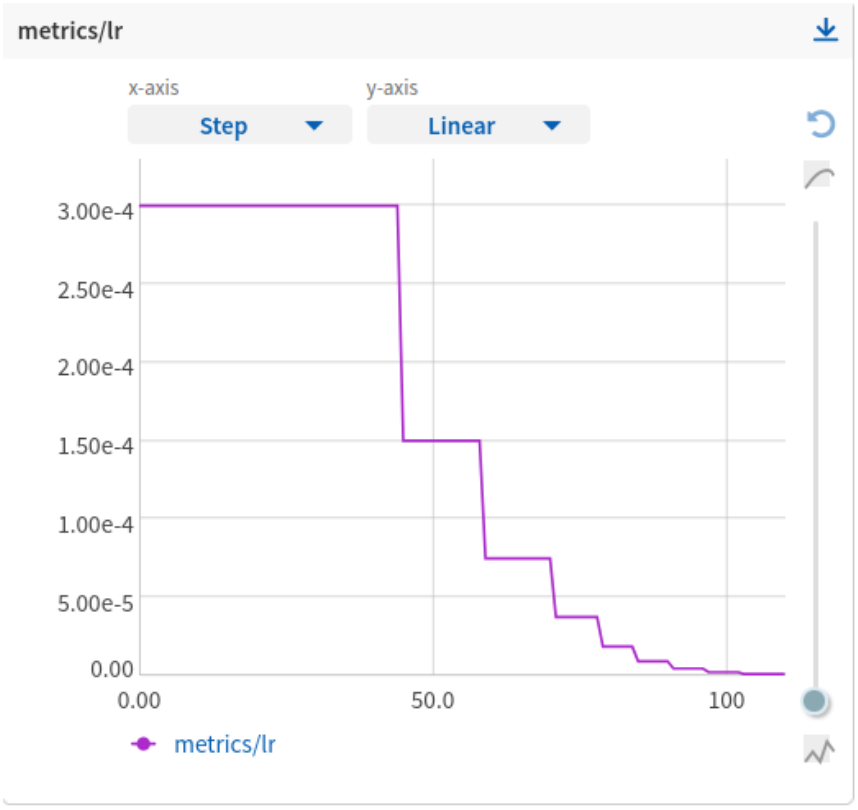

FIGURE 3.16: Learning Rate

Evoked potentials are signals in the EEG induced by sensory stimuli. It is used to study the function of the sensory cortex. We use these evoked potentials from the EEG to perform our downstream task of emotion classification.

We use the method of topography to represent these evoked potentials as two-dimensional images. The resulting topographic maps, or topo maps represent the brain's activity in a much-more understandable and efficient manner. We then employ deep learning techniques to classify emotions from these topo maps. We use **MNE** to visualize the EEG data and convert them to topo maps.

3.3.5.1 Data

The data that we use is a private dataset collected by ourselves. After converting it to topo maps, the dataset approximately consists of 1,500 images, comprising of just 2 emotions

for now:- happy and not happy. We are collecting more and more samples so as to get sufficient images for all the emotions.

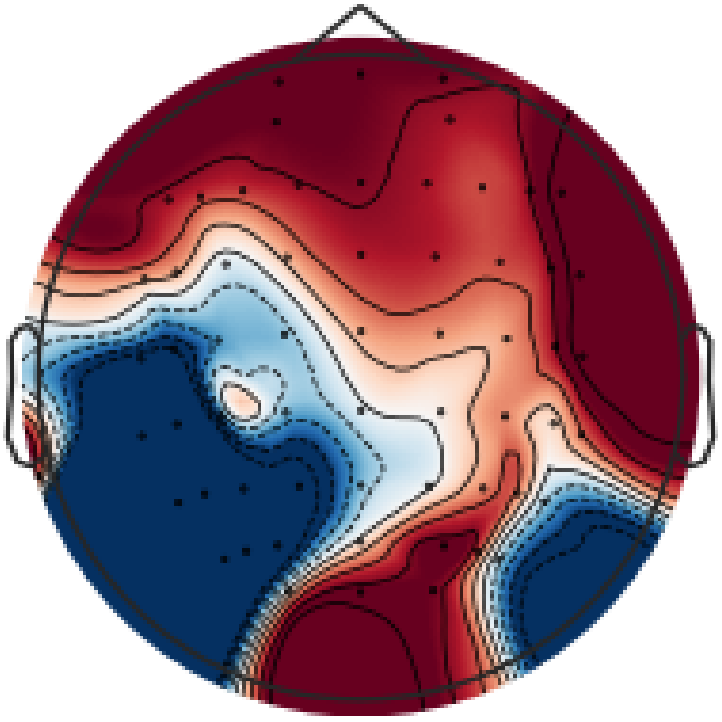

FIGURE 3.17: Topographical map

3.3.5.2 Results

In order to create a baseline model which serves as the reference for our future experiments, we tried to use different ResNet models without any augmentations. We have simply used dropout for regularization and He initialization. We trained the model for 15 epochs using adam. However, the model predominantly overfit towards the larger class(happy).

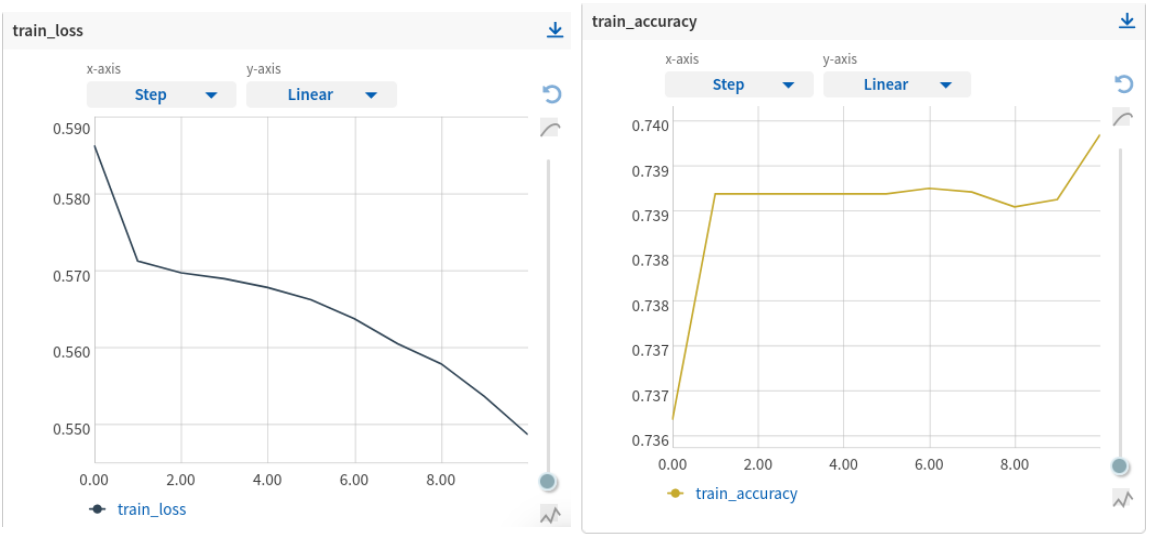

FIGURE 3.18: Training Loss and Accuracy

FIGURE 3.19: Validation Loss and Accuracy

3.3.6 Repeated Knowledge Distillation

For emotion recognition from images, building on the idea of mixing CNN and transformer-based models, we are currently employing a teacher-student strategy to improve the performance of Image Transformers, where we essentially use the predictions of the teacher model as pseudo-labels to optimize the student model. In this strategy, we aim at distilling or imparting knowledge from a strong teacher model to the student

transformer model using self-attention. To do this distillation, we make use of a distillation token. This distillation token is a learnable parameter like the class token, but its major aim is to reproduce the labels produced by the teacher model, instead of true labels. Hence, the global loss function will consist of two copmponents:- one component for the class token, which aims at reproducing the true labels, and the distillation component.

We aim at using a convolutional neural network based model as a teacher model, to inculcate the hard inductive biases of CNNs to the transformer model. This, we think, can give us performance at par, if not better than CNNs.

We also aim to use a fully trained student model and train another transformer using that as the teacher. On using a really strong teacher initially, and continuing the same process repeatedly for a finite number of times, we could get a new model that outperforms all the existing models out there, as it is learning from the best models out there.

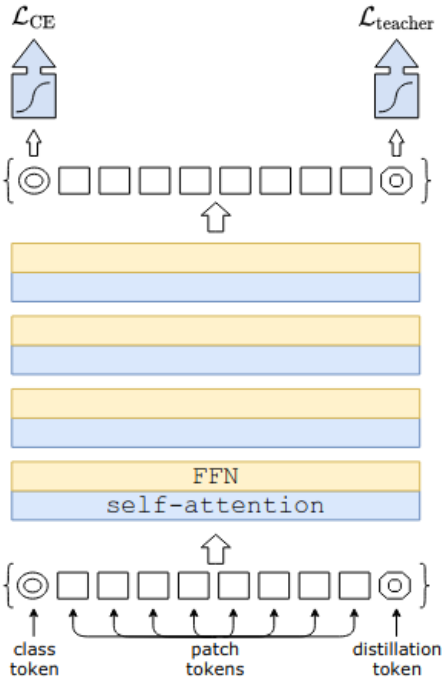

FIGURE 3.20: Knowledge Distillation in transformer

Chapter 4

Conclusion and Plan of Action

The effects of transformer-based models on facial emotion recognition has been studied thoroughly to our fullest extent. With the ConvNeXt model, that incorporates key components of transformers in CNN-based model, we have achieved a test accuracy of 70.2% accuracy and 67.01% validation accuracy, which is comparable to the state-of-the-art(74.4% test accuracy and 70.6% validation accuracy).

Our future plan of action with regards to this project is as follows:-

- **Continue work on knowledge distillation**:- Finish implementing the knowledge distillation strategy and test it on various different settings.
- **EEG emotion recognition**:- data pre-processing to be done to the topo maps and try out all the different data augmentations, regularizations, optimizers etc.

Bibliography

[1] Y. Khaireddin and Z. L. Chen, "Facial emotion recognition: State of the art performance on fer2013," *ArXiv*, vol. abs/2105.03588, 2021.

[2] C. Pramerdorfer and M. Kampel, "Facial expression recognition using convolutional neural networks: State of the art," *ArXiv*, vol. abs/1612.02903, 2016.

[3] L. Pham, T. H. Vu, and T. A. Tran, "Facial expression recognition using residual masking network," *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 4513–4519, 2021.

[4] Y. Fan, X. Lu, D. Li, and Y. Liu, "Video-based emotion recognition using cnn-rnn and c3d hybrid networks," *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, 2016.

[5] T.-H. S. Li, P.-H. Kuo, T.-N. Tsai, and P.-C. Luan, "Cnn and lstm based facial expression analysis model for a humanoid robot," *IEEE Access*, vol. 7, pp. 93 998–94 011, 2019.

[6] R. Pecoraro, V. Basile, V. Bono, and S. Gallo, "Local multi-head channel self-attention for facial expression recognition," *ArXiv*, vol. abs/2111.07224, 2021.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2021.

[8] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *ArXiv*, vol. abs/1706.03762, 2017.

[9] D. Nie, X.-W. Wang, L.-C. Shi, and B.-L. Lu, "Eeg-based emotion recognition during watching movies," *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pp. 667–670, 2011.

[10] Y.-P. Lin, C.-H. Wang, T.-P. Jung, T.-L. Wu, S.-K. Jeng, J.-R. Duann, and J.-H. Chen, "Eeg-based emotion recognition in music listening," *IEEE Transactions on Biomedical Engineering*, vol. 57, pp. 1798–1806, 2010.

[11] S. Alhagry, A. A. Fahmy, and R. A. El-Khoribi, "Emotion recognition based on eeg using lstm recurrent neural network," *International Journal of Advanced Computer Science and Applications*, vol. 8, 2017.

[12] X. Xing, Z. Li, T. Xu, L. Shu, B. Hu, and X. Xu, "Sae+lstm: A new framework for emotion recognition from multi-channel eeg," *Frontiers in Neurorobotics*, vol. 13, 2019.

[13] Y. Zhou, F. Li, Y. Li, Y. Ji, G. Shi, W. Zheng, L. Zhang, Y. Chen, and R. Cheng, "Progressive graph convolution network for eeg emotion recognition," *ArXiv*, vol. abs/2112.09069, 2021.

[14] N. R. Waytowich, V. J. Lawhern, J. O. Garcia, J. Cummings, J. Faller, P. Sajda, and J. M. Vettel, "Compact convolutional neural networks for classification of asynchronous steady-state visual evoked potentials," *Journal of neural engineering*, vol. 15 6, p. 066031, 2018.

[15] Y. Li, J. Huang, H. Zhou, and N. Zhong, "Human emotion recognition with electroencephalographic multidimensional features by hybrid deep neural networks," *Applied Sciences*, vol. 7, no. 10, p. 1060, 2017.

[16] Z.-M. Wang, S.-Y. Hu, and H. Song, "Channel selection method for eeg emotion recognition using normalized mutual information," *IEEE Access*, vol. 7, pp. 143 303–143 311, 2019.

[17] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. J'egou, "Training data-efficient image transformers & distillation through attention," in *ICML*, 2021.

[18] Z. Liu, H. Mao, C. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," 2022.

[19] S. d'Ascoli, H. Touvron, M. L. Leavitt, A. S. Morcos, G. Biroli, and L. Sagun, "Convit: Improving vision transformers with soft convolutional inductive biases," in *ICML*, 2021.

[20] C. Dalvi, M. sahebrao Rathod, S. Patil, S. Gite, and K. Kotecha, "A survey of ai-based facial emotion recognition: Features, ml & dl techniques, age-wise datasets and future directions," *IEEE Access*, vol. 9, pp. 165 806–165 840, 2021.

[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9992–10 002, 2021.