arXiv:2301.10906v1 [cs.CV] 26 Jan 2023

Facial Emotion Recognition

Arpita Vats
Santa Clara University
avats@scu.edu

Aman Chadha
Stanford University
aman@amanchadha.com

Abstract

We present a facial emotion recognition framework, built upon Swin vision Transformers jointly with squeeze and excitation block (SE). A transformer model based on an attention mechanism has been presented recently to address vision tasks. Our method uses a vision transformer with a Squeeze excitation block (SE) and sharpness-aware minimizer (SAM). We have used a hybrid dataset, to train our model and the AffectNet dataset to evaluate the result of our model.

1. Introduction

Facial Emotion recognition is one of the major areas of research. Faces analysis indicates recognizing the angle and expression of a human being independently of the immersive environment it could be, and ambiguous emotions are the cornerstone of the problem. Understanding human emotion also plays a vital role in emotional intelligence. Facial expression is one of the most natural, powerful, and universal signals for human beings to convey their emotional states and intentions. [4] [15] The focus of this project is to analyze how the `Swin` transformer performs on this task, comparing our model with the State-of-Art models on hybrid datasets, taking into account the lack of inductive bias proper for Vision Transformer's configuration introduced in [10]. Generally, Transformers are data-hungry, and they need a considerable amount of data to be efficient as State-of-Art models. So, the challenge is to define a good FER model based on the `Swin` configuration with the capacity to detect facial emotions using a small amount of data. We will explain: the data composition given by different datasets with high data variables, data integration to merge them into a unique dataset, data analysis which defines the features of each subset of data and defines some attributes and metadata to change for normalized samples, and data prepossessing for the data manipulation and augmentation to create a dataset split into three subsets with some features in common (image format, size, number of channels). In conclusion, we will discuss model configurations for face detection and cropping procedures and fine-tuned transformers for Facial Emotion Recognition related to an evaluation analysis of results models.Table 1 In this work we present a key facial emotion recognition, named FER.

Table 1. ActiViTy Classes

ID	Class
0	Fear
1	Sadness
2	Happy
3	Anger
4	Disgust
5	Surprise
6	Neutral

2. Related Works

Deng *et al*. [5] proposed a multi-task learning method to learn from missing labels. They used a data balancing technique for the dataset. First, they used the ground truth labels of all three tasks to train a teacher model. Secondly, they used the output of the teacher model as the soft labels. They used the soft labels and the ground truth labels to train their student models.

Kuhnke and Rumberg *et al*. [9] proposed a two-stream aural visual model. Audio and image streams are first proposed separately and fed into a CNN network. Then they use temporal convolutions to the image stream. They use additional features extracted during facial alignment and correlations between different emotional representations to boost their performance.

Thinh *et al*. [3] proposed a model, in which they used ResNet50 [7] as the backbone of their deep learning neural network, to accelerate and enhance the training process they used pre-trained weights of ImageNet [5]. They used VGGFace2 for emotion recognition.

Zhang *et al*. [17] proposed Despite the different psychological philosophies of the three emotional representations, it is widely agreed that the representations are intrinsically associated with each other. One of the pieces of evidence is that similar facial muscle movements (action units) mostly

indicate similar inner statements, and so do the perceived facial emotions. However, most previous research works on multi-task emotion recognition omit this fact and they just model the different tasks in parallel branches. Inspired by the observation above, they serially design the recognition process from local action units to global emotion states. The streaming structure is useful to adjust the hierarchical distributions on the different feature levels.

Wen *et al*. [16] proposed a facial recognition model named Distract your Attention Network(DAN). Which consists of Feature Clustering Network (FCN), Multi-head cross Attention Network (MAN), and Attention Fusion Network (AFN). Where FCN is responsible for extracting robust features by adopting large-margin learning objectives to maximize class separability, the MAN instantiates several attention heads to simultaneously attend to multiple facial areas and build an attention map of these regions. Further, the AFN distracts this attention to multiple locations before fusing the attention maps into a comprehensive one.

The current state of the art for emotion recognition of Affectnet dataset is [14], proposed face detection, tracking, and clustering techniques which are applied to extract the sequences of faces from each frame. Next, a single efficient neural network is used to extract emotional features in each frame.

3. Methodology

Figure 1 depicts our framework. The main idea is `Swin` transformer with a Squeeze excitation layer added before the `Swin` Transformer. The model predicts the different basic facial emotions of humans. `Swin` Transformer is a hierarchical Transformer whose representation is computed with Shifted windows. The shifted windowing scheme brings greater efficiency by limiting self-attention computation to non-overlapping local windows while also allowing for cross-window connection. This hierarchical architecture has the flexibility to model at various scales and has linear computational complexity with respect to image size.

3.1. Swin Transformer

We have used vision transformers *et al*. [12] by fine-tuning the pre-trained model on ImageNet, to classify eight human emotions: anger, contempt, disgust, fear, happiness, neutral, sadness, and surprise. Transformers are becoming the standard for NLP tasks. The core component in this kind of model is the attention mechanism, which can extract valuable features from the input with a standard query, key, and value structure, where the matrix multiplication between queries and keys, pulls the similarity between them. Then, the softmax function applied to the result is multiplicities on the value, obtaining our 'attention' mechanism. Our transformer architecture is based on a stack of eleven encoders preceded by a hybrid patch embedding architecture. The improvement is made by considering the lack of inductive bias problem. Vision Transformer has much less image-specific inductive bias than CNNs.`Swin` Transformer block is built by replacing the standard multi-head self-attention (MSA) module in a Transformer block with a module based on shifted windows, with other layers kept the same. A `Swin` Transformer block consists of a shifted window-based MSA module, followed by a 2-layer MLP with GELU nonlinearity in between. A LayerNorm (LN) layer is applied before each MSA module and each MLP, and a residual connection is applied after each module.`Swin` Transformer first splits an input RGB image into non-overlapping patches by a patch-splitting module, like `ViT`. Each patch is treated as a "token" and its feature is set as a concatenation of the raw pixel RGB values. A patch size of `4×4` is used and thus the feature dimension of each patch is texttt4×4×3=48. A linear embedding layer is applied to this raw-valued feature to project it to an arbitrary dimension C.

3.2. Datasets

One of the problems that we met, it's the availability of data. Many datasets are protected only for research uses and are not available completely to students [13]. So, we will use only samples available on Kaggle or other open-source data platforms. Transformers need a good amount of samples to retrieve hidden patterns during the training phase and the few data in our hands are not enough to satisfy this requirement. So, we have the plan to manipulate our small amount of samples to increase the size of the final datasets using data augmentation. The final dataset will have eight different classes integrated by three different subsets: FER-2013: It contains approximately 40,000 facial RGB images of different expressions with a size restricted to 48×48, and the main labels can be split into seven types: `0 = Fear, 1 = Sadness, 2 = Happy, 3 = Anger, 4 = Disgust,5 = Surprise, 6 = Neutral`. The Disgust expression has a minimal number of 600 samples, while other labels have nearly 5,000 samples each. CK+: The Extended Cohn-Kanade (CK+) dataset contains images extrapolated from 593 video sequences from 123 different subjects, ranging from 18 to 50 years of age with a variety of genders and heritage. Each video shows a facial shift from a neutral expression to a targeted peak expression, recorded at 30 frames per second (FPS) with a resolution of either 640x490 or 640x480 pixels. Unfortunately, we do not have all generated datasets, but we stored only 1000 images with high variance from a Kaggle repository. AffectNet: It is a sizeable facial expression dataset with 60.000 images classified in eight categories(neutral, happy, angry, sad, fear, surprise, disgust, contempt) of facial expressions along with the intensity of valence and arousal. Each dataset

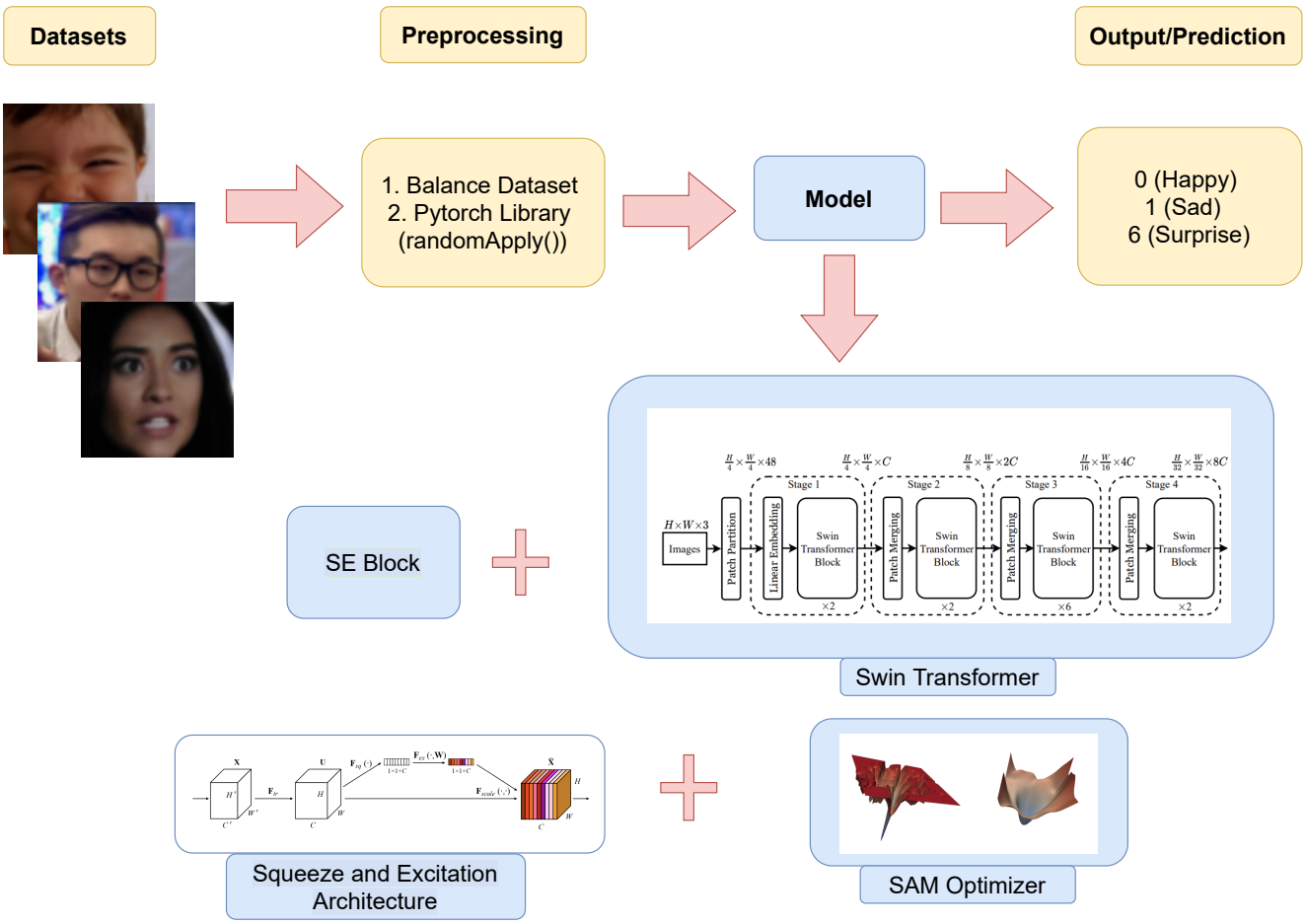


Figure 1. Facial Emotion Detection Using Swin Transformer with SE Block.

focuses on RGB channels for the coloring and has different sizes and image extensions entirely stored (the total amount of data is around 2 GB). So, we need to establish a standard format to manage them simultaneously. Finally, we have the final sections interested in the fine-tuning phase and training on a few models that can be saved and used in an external application for real-time classification on an ad-hoc application. More information about modeling is in the following sections.

3.3. Preprocessing

Once we have collected the data from all the different sources, then we integrated the dataset. We have validation and testing sets balanced with the same number of samples for each class; meanwhile, the training set has a minimum amount of samples for every class of values but is not balanced. We will readjust the training set through data augmentation to reach a sufficient number of samples for each class. We will then eliminate the excessively generated images and create a dataset with the same number of samples for each class of the problem domain. It is correct to say that the amount of data for contempt and disgust is really low, even after the integration with available open-source data; we can try to increase the variance of pixel matrices without using oversamples techniques but only data augmentation, which increases the number of the minor classes on the training set to obtain the same value of samples distribution and make the dataset balanced with generated images with similar features. This section will contain different data manipulation and merges of different datasets and different data augmentation involved in preprocessing the dataset and making it ready for training. As mentioned earlier we have used different datasets, so for them to be used as input for the model, We had to integrate the all different datasets into one dataset, with the same dimensions and configuration. Due to unbalanced class distribution, we decide to do various augmentation techniques. We used set of following techniques:

- **Image Rotation** It is one of the widely used augmentation techniques and allows the model to be more di-

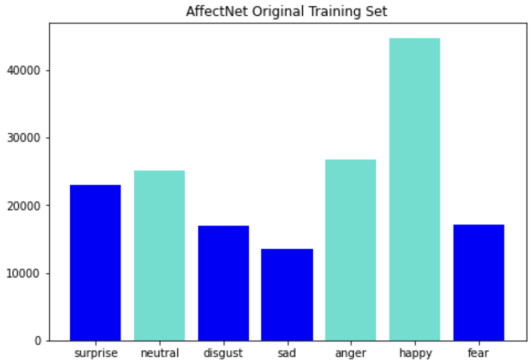


Figure 2. Final datasets for training after balancing the Unbalance datasets

versified. The value is between 0 and 360. We rotate images until 10 grades to adapt frontal images of FER-2013 and CK+48 on a similar face orientation to AffectNet faces and do not cripple already rotated images.

- **Augmentation** As we know Transformers require more data, since there was a lack of data we used different augmentation techniques to increase the sample size. We performed various augmentation methods using `pytorch` library like `RandomRotation`, `RandomAutocontrast`. This method helped the model get familiar with more data and hence improve the performance of model.

Figure 2 shows the Unbalanced and balanced dataset used for training after preprocessing and integration of different datasets of facial emotions mentioned above.

3.4. Model

In this section, we will introduce the use of `swin Transformer` [11] Single-Step Detector model for, respectively, emotions classification and face cropping with their adaptations. We resized the images to `224*224*3`shape for it to be used as Transformer input. Then the final step is to normalize it with the same values as the `swin` fine-tuning phase: 0.5 of mean and 0.5 as standard deviation along all channels. We have used `swin` transformer to recognize eight different facial emotions anger, contempt, disgust, fear, happiness, neutral, sadness, and surprise. `Swin Transformer` first splits an input RGB image into non-overlapping patches by a patch-splitting module, like ViT. Each patch is treated as a "token" and its feature is set as a concatenation of the raw pixel RGB values. A patch size of `4×4` is used and thus the feature dimension of each patch is `4×4×3=48`. A linear embedding layer is applied to this raw-valued

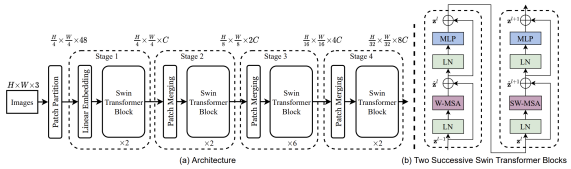


Figure 3. (a) The architecture of a Swin Transformer (Swin-T); (b) two successive Swin Transformer Blocks W-MSA and SW-MSA are multi-head self-attention modules with regular and shifted windowing configurations, respectively. [12]

feature to project it to an arbitrary dimension C. Several Transformer blocks with modified self-attention computation (Swin Transformer blocks) are applied to these patch tokens. The Transformer blocks maintain the number of tokens (H/4×W/4), and together with the linear embedding are referred to as "Stage 1".To produce a hierarchical representation, the number of tokens is reduced by patch-merging layers as the network gets deeper. The first patch merging layer concatenates the features of each group of 2×2 neighboring patches and applies a linear layer on the 4C-dimensional concatenated features. This reduces the number of tokens by a multiple of `2×2 = 4` (2 downsampling of resolution) The output dimension is set to 2C, and the resolution is kept at H/8×W/8. This first block of patch merging and feature transformation is denoted as "Stage 2".The procedure is repeated twice, as "Stage 3" and "Stage 4", with output resolutions of H/16×W/16 and H/32×W/32, respectively. These stages jointly produce a hierarchical representation, with the same feature map resolutions as those of typical convolutional networks, such as VGGNet and ResNet, which can conveniently replace the backbone networks in existing methods for various vision tasks.

3.5. Squeeze and Excitation (SE)

The Squeeze and Excitation block is also an attention mechanism. It contains widely fewer parameters than the self-attention block where two fully connected layers are used with only one operation of point-wise multiplication. It is firstly introduced in [8] to optimize CNN architecture as a channel-wise attention module, concretely we use only the excitation part since the squeeze part is a pooling layer built to reduce the dimension of the `2d-CNN` layers [1]. The SE is introduced on top of the Transformer encoder more precisely on the classification token vector. Different from the self-attention block which is used inside the Transformer encoder to encode the input sequence and extract features through class token the SE is applied to recalibrate the feature responses by explicitly modeling interdependencies among class token channels.

3.6. Transformer with Sharpness-Aware Minimizer

Sharpness-Aware Minimizer (SAM) uses the connections between the geometry of the loss landscape of deep neural networks and their generalization ability. It is also used for transformers to smooth the loss landscape and simultaneously minimize loss value as well as loss curvature thereby seeking parameters in neighborhoods having uniformly low loss value and generalizing, following a more linear curvature on the loss values. This is indeed different from traditional SGD-based optimization that seeks parameters having low loss values on an individual basis and not on neighbor relations. So, we can modify the Vision Transformer described in the previous section by adapting the optimizer on SAM. This should generalize the minimization of the loss value but increment the training time of the model. [2] The SAM optimization function can also tackle the 'noisy- label' problem present by obtaining a high degree of robustness to label noise datasets [6] which indeed is part of the problem taking into account the Affectnet dataset. Furthermore, according to final considerations of [2], SAM works better with small datasets, mostly on Vision transformers. In conclusion, we have considered two assumptions: first, SAM incurs another round of forwarding and backward propagation to update neuron weights, which will lead to around 2x computational cost per update but get better performances on small datasets. Second, we notice that the effect of SAM diminishes as the training dataset becomes larger, but we can't have the not augmented data and obtain a balanced dataset at the same time; due to the low number of contempt and disgust samples.

The proposed model starts from pre-trained parameters given by `timm` library available in Python. It allows us to download a pre-trained model with specific transformer configurations based on the dimension of the last layer for fine-tuning purposes. For each of their structure, it provides a random weighted-based version without a pre-training phase. Our model achieved an F1 score of 0.5452. As we present our experimental evaluation, we will also discuss the potential pitfalls of another method we try. We executed all our experiments on a system running Ubuntu Linux version 20.04 and equipped with a 12-core Intel(R) Core(TM) i9-7920X CPU @ 2.90GHz, 128 GB RAM, and 4 NVIDIA RTX 3090 24G GPUs.Models implementation uses PyTorch components as the main framework. During the preprocessing phase, we redefined the size of the images to adapt the dimension to 224x224 on three different channels (corresponding to the RGB channels). we normalized the input data and prepared samples for the training phase. The normalization phase applies a mean and a standard deviation of 0.5 for each channel. The best validation accuracy taken from the set of epochs during the training phase defines the final model weights set. Fine-tuning phase adapts the model parameters to the FER task using stochastic gradient descent or sharpness-aware minimizer adaptation with a cross-entropy loss function. The learning rate follows a scheduler that adjusts the initial value for every ten epochs by multiplying it by 0.1. Finally, we applied a simple momentum of 0.9 to increase the speed of training and variable learning according to the optimizer chosen in the experiment. We carried out different experiments with different configurations in this environment. The `swin` architectures enable better separation of classes compared to CNN baseline architecture. In addition, the SE block enhances `swin` model robustness, as the intra-distances between clusters are maximized. Interestingly, the features before the SE form more compact clusters with inter-distance lower than

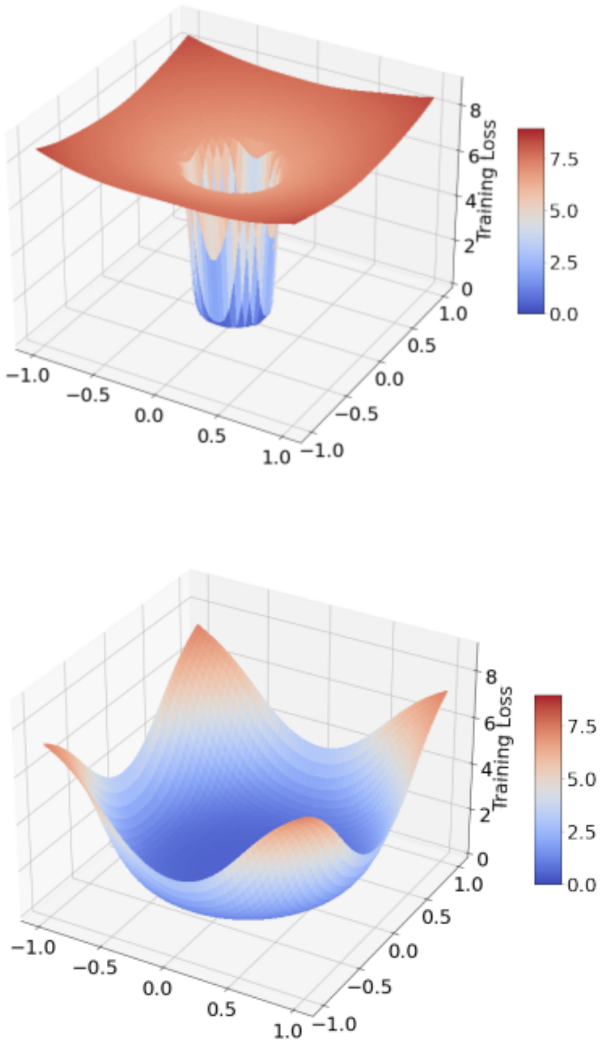
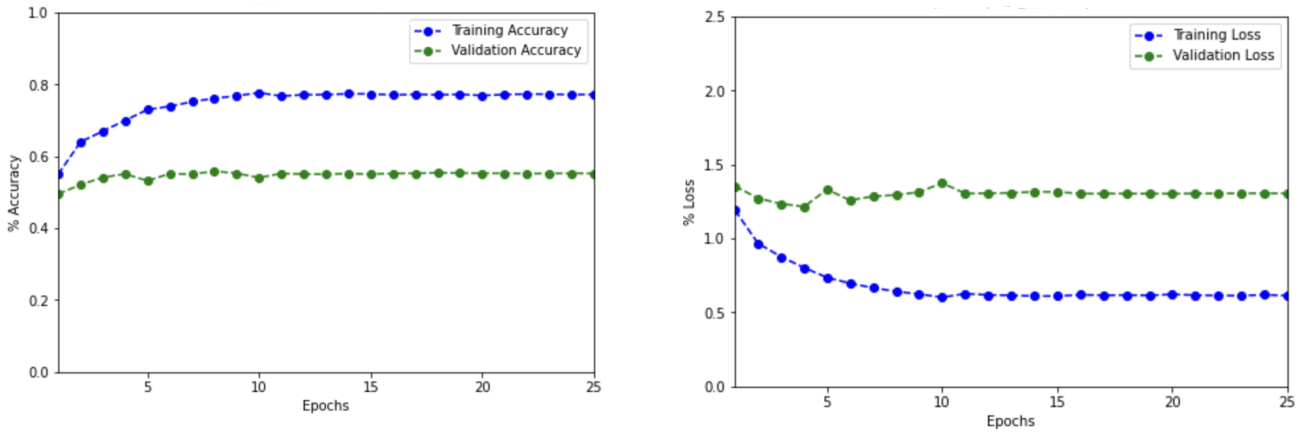


Figure 4. Cross-entropy loss landscape on ViT (top) and the same smoothed landscape with the application of SAM(bottom) during the training on ImageNet [2]



(a) (b)

Figure 5. Training and Validation Accuracy `Swin+SE+SAM` (b) Training and Validation Loss `Swin+SE+SAM`.

the features after the SE, which may interpret the features before SE are more robust than those after the SE. We have tested the performances of three different model which includes `swin`, `Swin`with SE, and lastly `Swin` with SE and SAM, and after testing we were able to conclude that `swin` with SE and SAM outperforms the other architectures.

3.7. Metrics Evaluation

We tested models on 4000 different samples of AffectNet without data augmentation with training and validation sets.

Table 2. Testing Accuracy (with approximation to 7 classes), Weighted Average Precision, Recall, and F1-Score on project models tested on AffectNet.

Metrics	Swin	Swin-SE	Swin-SE-SAM
7 Classes Accuracy	0.4921	0.5104	**0.5310**
Weighted Avg. Precision	0.5090	0.5470	**0.5485**
Weighted Avg. Recall	0.5000	0.5225	**0.5410**
Weighted Avg. F1-Score	0.4943	0.5169	**0.5420**

We trained our model using `Swin+SE+SAM` model for 25 epochs. The testing dataset is formed by 4000 samples equally distributed (500 samples per class). The plot above shows the training and Validation accuracy, Training accuracy was 0.832 and Validation accuracy was 0.5784. Also as the training reached closer to 25 epochs we can see training loss reduced similar to validation loss. Table 2 shows different metrics results for three different models we choose to compare against, which include `Swin`, `Swin+SE`, `Swin+SE+SAM`, We can see that the performance of`Swin+SE+SAM` seems to outperforms rest of the model used. Due to a lack of data for contempt class, we evaluated models on AffectNet considering only the 7 augmented classes. Finally, for a more detailed evaluation, we have written precision, recall, and F1. We tried some different configurations about the use of SAM and the gradual learning rate on the `Swin` configuration with the objective to find the best configuration, avoiding overfitting or underfitting, and obtaining acceptable performances using a small dataset. The current state of the art for Affectnet dataset F1 score is 0.6629 for 7 classes of emotions using Multi-task Efficient Net-B2.On the other hand, our model is one of the first approaches using `swin` Transformer for facial emotions recognition, and we were able to achieve an F1 score of 0.5420.

4. Conclusion

We have explored the direct application of Transformers to image recognition and test robustness on noisy datasets like AffectNet. We interpret an image as a sequence of patches and process it by a standard Transformer encoder as used in NLP. Our challenge was to test and obtain a model with the capability to recognize eight classes of emotions with the constraints of data availability for the FER task; we used only a subset of AffectNet, FER-2013, and CK+ to train and validate models. we also used the `swin`+SE, a simple scheme that optimizes the learning of the `swin` by an attention block called Squeeze and Excitation. It performs impressively well in improving the performance of `swin` in the FER task, Additionally, we also used SAM optimizer to further enhance the model performance to avoid the loss due to noisy data.

References

[1] Mouath Aouayeb, Wassim Hamidouche, Catherine Soladie, Kidiyo Kpalma, and Renaud Seguier. Learning vision transformer with squeeze and excitation for facial expression recognition, 2021. 4

[2] Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pre-training or strong data augmentations, 2021. 5

[3] Phan Tran Dac Thinh, Hoang Manh Hung, Hyung-Jeong Yang, Soo-Hyung Kim, and Guee-Sang Lee. Emotion recognition with sequential multi-task learning technique. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 3586–3589, 2021. 1

[4] Charles Darwin. *The Expression of the Emotions in Man and Animals*. Cambridge Library Collection - Darwin, Evolution and Genetics. Cambridge University Press, 2013. 1

[5] Didan Deng, Zhaokang Chen, and Bert Shi. Multitask emotion recognition with incomplete labels. pages 592–599, 11 2020. 1

[6] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization, 2020. 5

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 1

[8] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks, 2017. 4

[9] Felix Kuhnke, Lars Rumberg, and Jorn Ostermann. Two-stream aural-visual affect analysis in the wild. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*. IEEE, nov 2020. 1

[10] Hanting Li, Mingzhe Sui, Feng Zhao, Zhengjun Zha, and Feng Wu. Mvt: Mask vision transformer for facial expression recognition in the wild, 2021. 1

[11] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 4

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021. 2, 4

[13] Ali Mollahosseini, Behzad Hasani, and Mohammad H. Mahoor. AffectNet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, jan 2019. 2

[14] Andrey V. Savchenko, Lyudmila V. Savchenko, and Ilya Makarov. Classifying emotions and engagement in online learning based on a single facial expression recognition neural network. *IEEE Transactions on Affective Computing*, pages 1–12, 2022. 2

[15] Y.-I. Tian, T. Kanade, and J.F. Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):97–115, 2001. 1

[16] Zhengyao Wen, Wenzhong Lin, Tao Wang, and Ge Xu. Distract your attention: Multi-head cross attention network for facial expression recognition, 2021. 2

[17] Hanzhong Zhang, Jibin Yin, and Xiangliang Zhang. The study of a five-dimensional emotional model for facial emotion recognition. *Mobile Information Systems*, 2020:1–10, 12 2020. 1