Emotion Recognition using Facial Expression

Submitted as Research Report in

SIT723/SIT724

SUBMISSION DATE

T2-2021

Tithra Chap

STUDENT ID 220051603

COURSE - Master of Data Science (S777)

Supervised by: Assc. Prof. Richard Dazeley, Dr. Bahareh Nakisa, Dr. Sunil Aryal

Abstract

Predicting emotions of people via their facial expressions helps improve the future of AI devices to understand and communicate the need of human without any verbal interaction. This concept has a wide application across industries where verbal communication is suppressed by the inconvenient nature of the interaction, e.g., student's satisfaction of a lecture session. Inspiring by this potential benefit, this research study focuses on developing a machine learning model that can effectively detect emotions using human facial expression by employing attention technique in combination of convolutional neural network. More specifically, this study applies Convolutional Block Attention Module (CBAM), that has been proposed for image classification by Woo et al. (2018), to further resolve a more challenging task in identifying the subtle distinctions of human facial expressions. The study incorporates ResNest architecture with CBAM to objectively detect seven basic human emotions: "Anger", "Disgust", "Fear", "Happiness", "Sadness", "Surprise", and "Neutral" (or "Contempt" in CK+) in FER2013, CK+ and JAFFE datasets. In addition, our study will discuss some pre-processing techniques such as face detection, cropping, and global gamma equalization to eliminate bad data samples and to minimize the variation of illumination. The study experiments are conducted under a range of finetuning settings of major hyperparameters to ensure the optimization of the model's performance. A proper selection of hyperparameters will also be elaborated to a certain extend. Lastly, we present experimental results of the accuracy performance of our model in identifying the seven basic emotions and compare them against existing literatures which use similar neural network architectures and datasets. The study also discusses the practical factors that contribute to the model's performance. Based on the results, our proposed model shows superior performance against other models in some datasets, i.e., JAFFE and CK+ datasets, but shows some level of compromise on other dataset, i.e., FER2013.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. Introduction

1. Aim and Objectives

The aim of this research study is to develop a machine learning model that is capable in classifying human emotions based on Facial Expression Recognition (FER). Since there are existing models out there that have been proposed to resolve the same problem, this research project is looking to discover a new model architecture that can leverage the emotion detection performance to another level.

To achieve this, we:

- conduct a research on existing literatures and identify a potential model architecture that never been used to resolve the problem in FER,
- select multiple datasets for experimenting with our model,
- explore and utilize some useful pre-processing techniques to eliminate outliers, especial in unconstraint dataset like FER2013,
- work on a series of hyper-tuning setting to reveal the optimum performance of the model,
- benchmark the experiment results with existing literatures, and
- evaluate and discuss the potentiality of the model and the drawback in a certain situation.

2. Structure

The rest of this report is organized as the following. The next section (**Section II**) describes the existing literatures. **Section III** explores the research design. The detail of artifact development is presented in **section IV**, followed by a discussion of experiment results in **section V**. Finally, we wrap up our work and share our view of possible future work in **section VI**.

II. Literature Review

Human emotion detection has a wide range of applications in many areas including business, medical, education, and even in government. For instance, it may be used by business corporation to identify customers' emotional satisfaction (Kang, 2012), by educator to extract level of student understanding of English classes (Cui et al., 2021), by automaker to warn driver's fatigue (Zhang and Hua, 2015), or by medical center to learn mental health of in-hospital patients (Chen et al., 2019), by government to study soldiers' social behavior (Anaki et al., 2012), etc. The usefulness of human emotion detection has brought a significant interest in research area. Variety of resources have been utilized to recognize the state of human emotions. The most common resources of emotion detection are facial images, facial videos, texts and voices (Yalamanchili et al., 2021). Other less common resources may include human pulse signals produced in the form of electrocardiogram (Pan and Li, 2020), and electroencephalographical signals produced by the electrical activity of neurons inside the human brain (Mehmood et al., 2017).

The growing power of computing and the upsurging uses of multimedia contents of today smart devices drive this area of emotion detection to focus more on the utilization of images and videos as the main source of data. Based on these visual resources, detection algorithms must be able to analyze and recognize the visual expression on the subject faces and predict the different state of emotions. According to (Ko, 2018), the seven basic human emotions of facial expression are happiness, surprise, anger, sadness, fear, disgust, and neutral. The classification of these emotions can be further enhanced by the support of preprocessing tasks, such as face alignment, augmentation, and normalization of the image data before training process (Li and Deng, 2020). FER is more challenging than other image classification because of the subtle variation of expressions (Kimura and Yachida, 1997), face angles or poses, and illuminations (Khatri et al., 2014). To mitigate these distracting factors, some studies included pre-processing stage where image data was regularized before fitting to the training model. Most of pre-processing tasks involve face detection and cropping as the first stage. In further stage, Khemakhem and Ltifi (2019) equalized the contrast level of the cropped face images, adjust their gamma level, and added embossed effect before fitting them to a 5-layers CNN. Another method, called Local Binary Patterns (LBP), was also employed by Li et al. (2020) as the second stage of pre-processing task to enhance the facial feature.

The core processing of detecting emotions in FER is the machine learning algorithm that is designed to pinpoint the differences in the images. Earlier, traditional machine learnings such as Multi-layer

Perceptron, Support Vector Machines, or k-Nearest Neighbours, and the like, were often utilized to handle this classification task of FER. However, recent state-of-art of image classification has a new trend into convolutional neural network (CNN). CNN can eliminate the process of manual feature extraction such as histogram of oriented gradients, gradient feature mapping, eigen vectors, etc. (Vyas et al., 2019), and thus speed up the work and minimize errors. CNN also goes through the feature extraction, but it is done by its natural mechanism during the learning process, said Vyas. This automatic feature extraction boosts the performance of CNN in FER to another level beyond the historical performance of machine learning. Its popularity gives rise to a large amount of variants such as Deep Belief Network, Deep Autoencoder, Recurrent Neural Network, and Generative Adversarial Network (Li and Deng, 2020). For instance, Deep CNN with ten layers was employed to classify facial emotions (Khaireddin and Chen, 2021). Other researcher made use of existing framework of CNN such as VGGNet to resolve the same problem (Kumar et al., 2017).

More recent studies have been found to utilize Attention Technique (AT) in CNN to capture the distinct expressions on human face. AT is the independent block or layer which can be blended into existing structure of CNN. AT is known as an auxiliary mechanism to emphasize the extraction of useful feature within a given facial expression (Sun et al., 2020). There are wide variety of uniquely crafted ATs, though the typical and effective types of ATs suggested by Sun et al. (2020) are Spatial attention, Self-attention, and Channel attention mechanism. Spatial attention mechanism is known to identify the invariance of image feature despite the difference of angle variation such as rotation, re-scaling, and other deformation. Self-attention helps resolve the loss of global information during the process of convolution and pooling, while Channel attention spots the important kernel's channel that contains useful information of the input image. The combined benefit of Channel and Spatial attention has led to an introduction of a new AT module called Convolutional Block Attention Module (CBAM) which was proposed by Woo et al. (2018). Woo also noted the powerful framework of Residual Network (ResNet) which can be integrated with CBAM to boost the performance even further. The key merit of ResNet comes from its ability to remember relevant feature information even after a long propagated convolutional and pooling transformation. Resnet prevents information loss caused by gradient explosion or vanishing by integrating a shortcut connection in every other convolutional block (Zhang et al., 2021, Wen et al., 2020).

By learning the potential of CBAM and ResNet as being one of the most promising image classification mechanisms, Sun (2021) applied CBAM + ResNet18 model to resolve the FER on FER2013 and CK+ dataset. Similarly Wen et al. (2020) also used ResNet architecture in combination with CBAM to

classify facial emotions on FER2013 and JAFFE datasets, by employing 17 convolutional blocks. Sun placed a CBAM block at the last block, just before Average-Pool layer of Resnet, while Wen placed 4 ResNet + CBAM layers at the consecutive location after the first convolutional block of the ResNet. **Figure 1** and **2** shows the CBAM integrated architecture of Sun and Wen respectively.

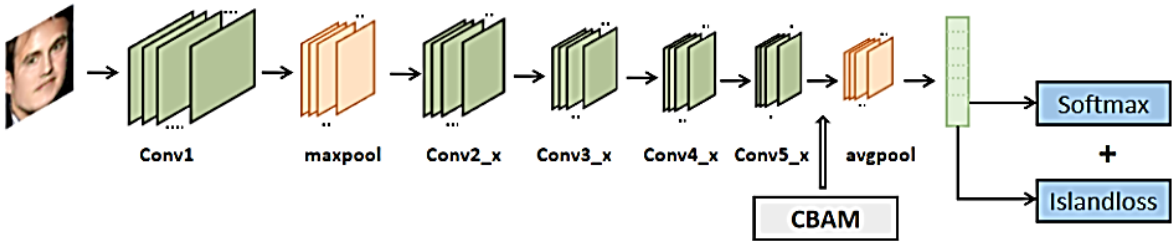

Figure 1. The proposed CBAM integrated architecture in ResNet18 by Sun (2021).

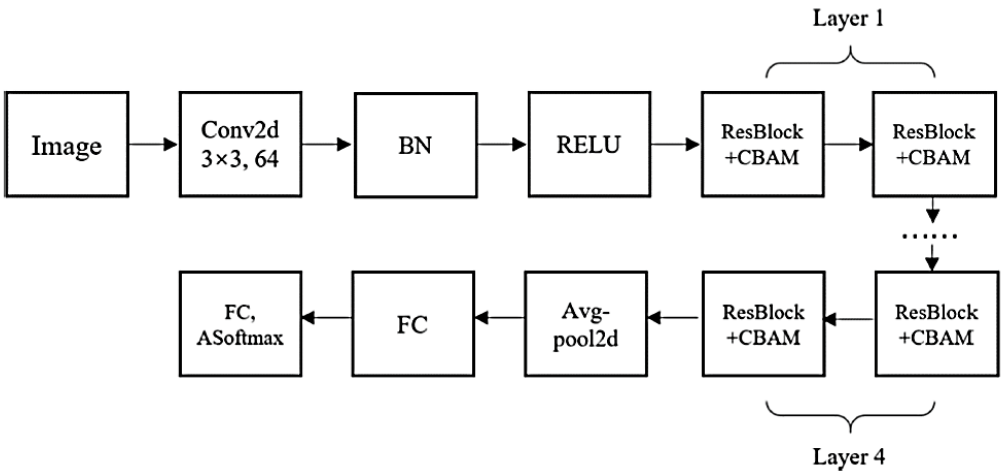

Figure 2. The proposed CBAM integrated architecture in ResNet by Wen et al. (2020).

Based on the experimental results, Wen's model outperformed Sun's on FER2013 dataset about **2%**. Another study, by Zhang et al. (2021), implemented CBAM + ResNet34 model. Unlike the earlier studies, he located 2 CBAM blocks before and after the complete structure of ResNet34. **Figure 3** presents his network architecture design and the location of CBAM blocks and ResNet34. The experimental results of his model indicated a better accuracy in detecting emotion than the earlier models either on FER2013 or CK+ dataset.

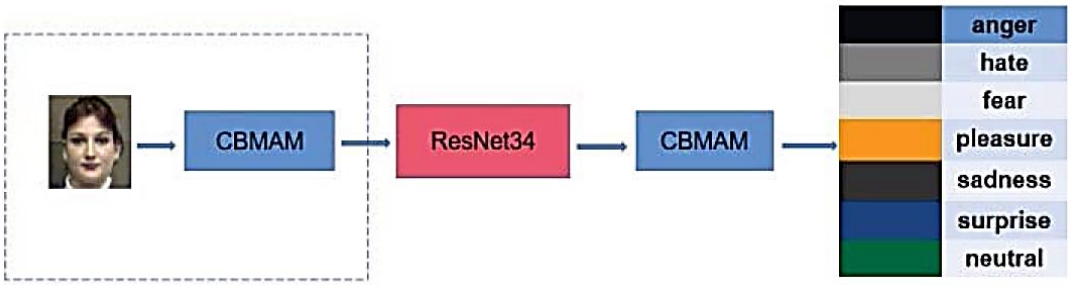

Figure 3. The proposed CBAM integrated architecture in ResNet34 by Zhang et al. (2021).

Another attempt was found to use CBAM + VGG19 model in FER (Cao et al., 2020). Cao inserted the CBAM block after every max pooling block of the VGG19 network in the attempt to reduce the complexity of the already-heavy VGG19 structure. **Figure 4** shows his CBAM + VGG19 network architecture. His experiments were conducted using FER2013 and CK+ dataset. VGG19 has owned a great classification reputation in many papers, however his model of CBAM + VGG19 showed inferior accuracy performance compared to the forementioned studies.

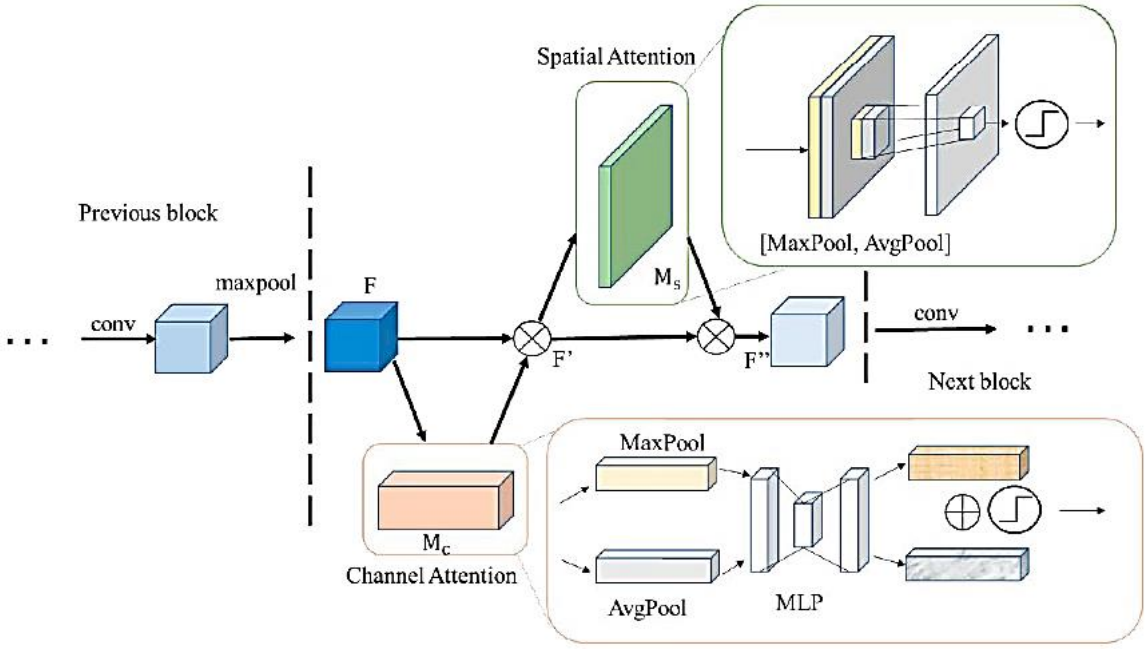

Figure 4. The proposed CBAM integrated architecture in VGG19 by Cao et al. (2020).

Inspired by the effectiveness of CBAM and ResNet in FER, our study will focus on the gap of CBAM + ResNet model that has never been explored by the existing papers. Different from the forementioned studies, CBAM + ResNet structure proposed by Woo et al. (2018) has never been used to detect the basic emotions from facial expression images of FER2013 or CK+ dataset. In a distinct design, Woo applied each CBAM block at every output of convolutional block of ResNet. **Figure 5** illustrates the location of CBAM block within his network architecture. So far, he only experimented the model using general images classification such as ImageNet-1K, MS COCO, and VOC 2007 dataset. We will use this

architecture design to resolve problem of FER on FER2013, CK+, and JAFFE and benchmark its performance against other models mentioned earlier.

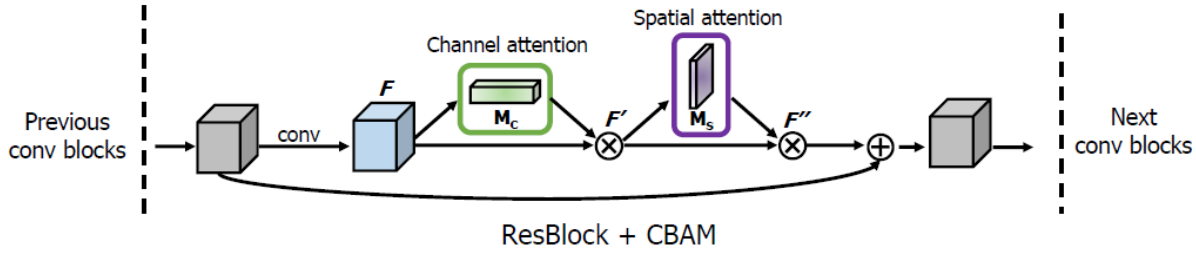

Figure 5. CBAM integrated with a ResBlock in ResNet by Woo et al. (2018).

Beside design and training data, it is also worth to mention that the model performance is also contributed by other factors by such as hyper-parameter setting and sampling technique of the model. It is noticeable that previous mentioned papers which employed CBAM had different setups of the hyper parameters of their experiments including number of epochs, batch size, learning rate, etc. Some of them used fixed ratio sampling between training and validation set, while others used cross-validation sampling. Taking this variation in mind, our study will consider all feasible hyper-tuning parameters and sampling techniques to ensure the optimum performance of the model.

III. Research Design

1. Datasets

FER2013 dataset

The FER2013 is a wild dataset of facial expression images that was taken randomly from internet. FER2013 database contains images with facial emotions and was developed by Goodfellow et al. (2013) for a Kaggle competition on FER in 2013. It includes a set of 35887 images in 8-bit grayscale format measuring 48x48 pixels, with facial emotions, divided into 3 sets: 28709 training data, 3589 test data and 3589 validation data. The 35887 images are categorized into: anger – 4953 images, disgust – 547 images, fear – 5121 images, happiness – 8989 images, sadness – 6077 images, surprise – 4002 images, and neutral – 6198 images. **Figure 6** shows five random sample images of FER2013 database.

Figure 6. Five sample images of FER2013 dataset

CK+ dataset

The Extended Cohn-Kanade (CK+) dataset contains 593 video sequences from a total of 123 different subjects, ranging from 18 to 50 years of age with a variety of genders and heritage. It contains 8-bit grayscale images with size of 48x48. The total sample size of the dataset is 981. It is divided into seven expression classes: anger – 135 images, contempt – 54 images, disgust – 177 images, fear – 75 images, happiness – 207 images, sadness – 84 images, and surprise – 249 images. The CK+ database is widely regarded as the most extensively used laboratory-controlled FER database available and has been used in the majority of FER performance comparisons. **Figure 7** presents five random sample images of CK+ dataset.

Figure 7. Five sample images of CK+ dataset

JAFFE dataset

The JAFFE dataset consists of 213 images of different facial expressions from 10 different Japanese female subjects. Each subject was asked to do about seven facial expressions (six basic facial expressions and a neutral). This dataset contains 8-bit grayscale images with size of 256x256. The sample contains a total of 213 images which further subdivided into anger – 30 images, disgust – 29 images, fear – 32 images, happiness – 31 images, sadness – 31 images, surprise – 30 images, and neutral – 30 images. This dataset is considered as a lab dataset with well control of class distribution. **Figure 8** illustrates five random sample images of JAFFE dataset.

Figure 8. Five sample images of JAFFE dataset

2. Pre-processing

Face detecting and cropping

Since facial expressions are difficult to detect in noisy background images, all the datasets used in our study are pre-processed to emphasize the facial feature and minimize the unwanted features. The first stage of pre-processing method is to detect the faces and crop them from the existing

background. We employ two library packages in python namely CV2 and FACE_DETECTION to detect faces and, if necessary, crop them. CK+ and JAFFE datasets still maintain their original sample size after face-detection and cropping, but FER2013 is only left with a total sample of 27,783. The process filtered out about 25% of the FER2013 dataset. The reason is that this dataset contains a lot of noises that were not detected as faces. Those noises include no faces found in the images or the shape of faces are partially missing, sided view, faces covered by disruptive object, watermark or hands, and poor resolution or blur. **Figure 9** illustrates sample images of JAFFE after face-detection and cropping.

Figure 9. Five sample images of JAFFE dataset that have been cropped and resized to 64x64.

Global gamma equalization

The facial images contain distinctive conditions of lighting or illumination, while others have different skin color of the subjects. These variations increase the noise of the sample images and reduce machine learning efficiency in training the facial features. To mitigate this issue, we normalized the images using global gamma equalization that could enhance the uniformity of the images. **Figure 10** presents sample images of JAFFE dataset after applying global gamma equalization.

Figure 10. Five sample images of JAFFE dataset after equalizing gamma effect.
The equalization fades away the dark shades on the faces and helps uniform either lighting or skin tone.

3. Model Architecture

ResNet Architecture

ResNet is known a as winner platform of deep learning model in Imagenet competition. The advancement of this architecture is the ability to quickly speed up the training of neural network with small kernel sizes in its ResNet modules. In addition, each ResNet module carries forward the

memory of the input image to prevent the gradient from disappearing or exploding due to the deepening of network layers. Its main function is to use residual learning to add identity short connection in the network and connect the original input directly to the later network layer. **Figure 11** embraces a residual module within the ResNet. As seen in the figure, each ResNet residual module contains two residual blocks. The input *x* is transformed via the residual module to produce the output *F(x)*. This output forms a part of the information that is later combined with the previous input *x*. This mechanism prevents information loss for a long propagation throughout the network.

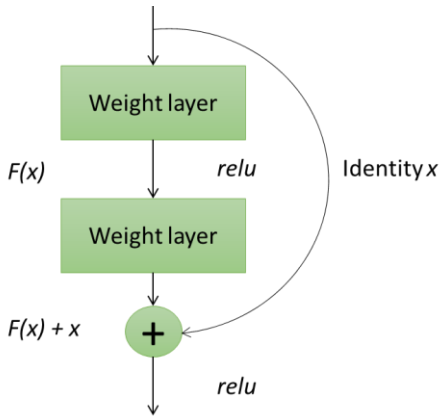

Figure 11. A block of ResNet structure with identity x carrying forward

Convolutional Block Attention Module (CBAM)

CBAM represents a combined block of attention technique proposed by Woo et al. (2018). This combined block embraces two attention modules, namely Channel Attention Module (CAM) and Spatial Attention Module (SAM). CAM is designed to emphasize the important channel and suppress the unimportant ones of the input image, while SAM is built to spot the informative regions of the input image. **Figure 12** shows the diagram of the CBAM block.

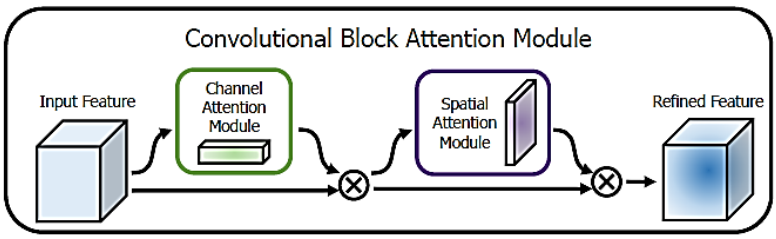

Figure 12. CBAM block with its sub-sequential CAM and SAM modules

CAM aggregates spatial information of a feature map of the input image by using both average-pooling and max-pooling operations, generating two different spatial context descriptors (one-dimension vectors). These descriptors are then forwarded to a shared network (multi-layer perceptron) for processing. Later, the two descriptors are combined using element-wise summation before being activated by sigmoid function to produce the final one-dimension vector. **Figure 13** visualizes the process of CAM.

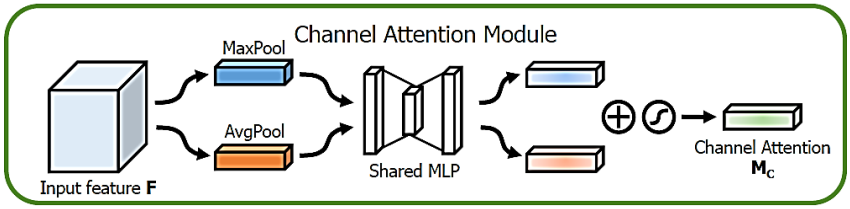

Figure 13. CAM processing pipeline

In SAM, average-pooling and max-pooling operations are applied on the feature map of the input image along the channel axis and concatenate them to generate an efficient feature descriptor (a two-dimension vector). The concatenated feature descriptor is later processed via a convolution layer to generate a spatial attention map which encodes the location of image feature that can tell whether to emphasize or suppress the information. **Figure 14** shows the diagram of SAM's process.

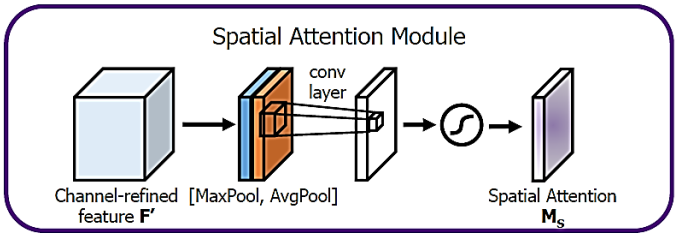

Figure 14. SAM processing pipeline

Model Architecture

In this study, we pick ResNet as the main architecture to build the machine learning model. Since our problem is to classify only 7 categories of target labels, it is reasonable to employ a ResNet that contains a depth of 18 blocks or ResNet18. We place the CBAM in each of the residual module, specifically after the second sub-module, as suggested by Woo et al. (2018). **Figure 5** elaborates graphically the placement of CBAM module in the ResNet.

IV. Experiments

The model construction is built using Keras packages in Python scripts. The simulation process is run under computing service provided by Google Colab with GPU power. The experiments are conducted on three datasets: FER2013, CK+, and JAFFE. Due to the distinct nature of each dataset; sampling method, pre-processing procedure, and parameters of the experiments are also arranged differently.

1. Sampling method

FER2013 dataset comes with stratified and shuffled sets of training, validating, and testing component. For that reason, we keep its original proportion when experiment the model. However, CK+ and JAFFE are not originally categorized, and therefore we shuffle and randomly split them into 80% of training, 10% of validating, and 10% of testing set. To generalize the sample

better, we also incorporate real-time data augmentation during the training process. The artificial augmentation includes zca_epsilon = 1e-06, width and height shift range = 0.1, and horizontal flip.

2. Pre-processing steps

FER2013 is a wild, un-constrained dataset. We do not crop this dataset as most of the images are properly done so. Another reason is that FER2013 images involve hand gestures as part of the facial expression, thus cropping will reduce the uniqueness of the images. CK+ dataset is also properly cropped originally, however JAFFE dataset contains large background images and it is necessary to crop them to further emphasize the unique expression of those faces. Unlike FER2013 and CK+, JAFFE images come with 256x256 pixels. After cropping, it is necessary to resize them to 64x64 for a convenient training purpose, while keeping adequate clarity of the images.

3. Parameter Settings

To optimize the model performance, it requires a finetuning process that tries through a range of relevant parameters such as optimizer, learning rate (LR), batch size, and epoch size. These settings also depend on the nature of the datasets. **Table 1** below lists the best settings found in our study.

Table 1. Best parameters setting for individual dataset

Datasets	FER2013	CK+	JAFFE
Optimizer	Adam	Adam	SGD
Learning Rate	Epoch-based schedule: 0.01, 0.001, 0.0001	Epoch-based schedule: 0.01, 0.005, 0.001	Epoch-based schedule: 0.05, 0.01, 0.005
Epoch size	200	200	80
Batch size	512	50	14

There are other settings of parameters that have been experimented but found to be less effective for the model performance. They include the use of: RMSprop optimizer, constant LR, one-cycle LR proposed (Smith, 2017), super-converge LR (Smith and Topin, 2018), smaller epoch size, and other inappropriate batch sizes. Their impacts on the model performance will be discussed later in the results and evaluations section.

4. Model Summary

The model contains 18 convolutional blocks. A residual module consists of two subsequential convolutional blocks, and thus it makes 9 residual modules in a ResNet18 architecture. CBAM is added to the second convolutional block of each residual module. The model ends with a flatten

layer, followed by a Dense layer as the output layer. **Figure 15** provides a snapshot of a compiled section of residual module.

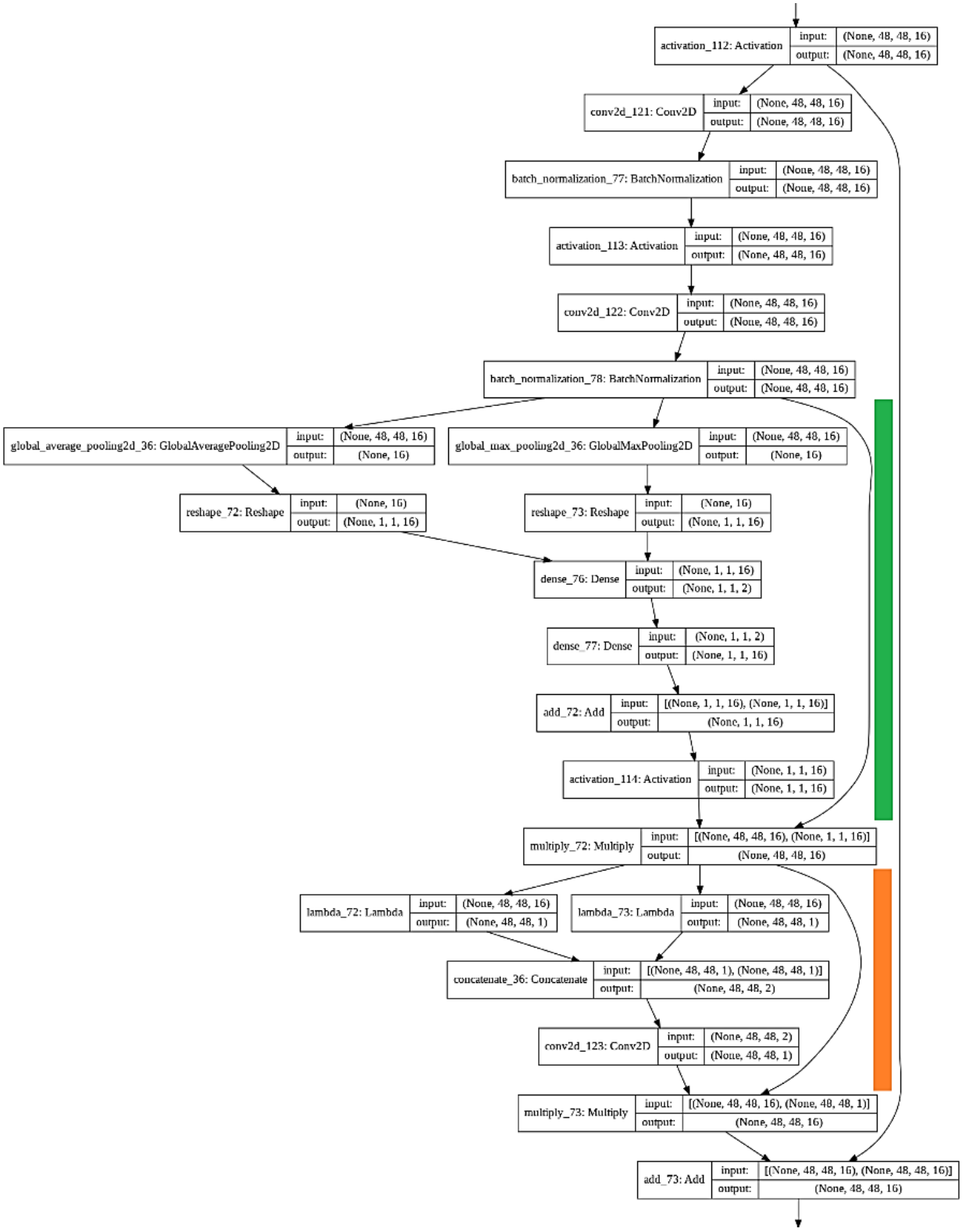

Figure 15. A residual module that contains CAM block (green bar) and SAM block (orange bar)

V. Results and Evaluations

Pre-processing strategies

The model is primarily experimented using different pre-processing techniques such as global gamma equalization with and without emboss effect, and with LBP. Having proved by Khemakhem and Ltifi (2019), emboss effect would add approximately **3%** of accuracy performance to the performance of gamma equalization alone. In another proven claim, LBP has shown a significant improvement of accuracy performance on FER2013 dataset (Li et al., 2020). However, we experiment these pre-processing techniques and find out that neither of them helps improve the classification accuracy in identifying emotion on FER2013 dataset, but rather lowers the performance further. Using Global gamma equalization alone can improve the accuracy rate by **1%** from the original images. The most significant impact on model performance is the noise of the sample. Using face-detection to eliminate of bad data points reveals a remarkable improvement between original dataset (**67.23%**) and a pruned dataset (**70.89%**). **Table 2** reveals the accuracy performance comparison of the above-mentioned pre-processing strategies using ResNet18 + CBAM and FER2013.

Table 2. Performance comparison between pre-processing strategies in FER2013 dataset

Pre-processing method	Accuracy
Original image	67.23%
Original images (with face-detection process)	70.89%
Global gamma equalization	71.43%
Global gamma equalization + Emboss	69.26%
Global gamma equalization + LBP	65.05%

Augmented data sample

Besides the effect of pre-processing strategies, augmenting data sample also contributes a significant improvement to the overall accuracy performance. In FER2013, data augmentation boosts the accuracy performance about **10%** in addition to the non-augmented sample of only **59.52%,** according to our experiment using ResNet18+CBAM. **Figure 16** compares the validation accuracy of augmented and non-augmented data sample on FER2013 during training. It is seen that without data augmentation, the model is dramatically overfitting. With the same number of training epochs, model without augmentation reaches the peak of training accuracy quick, while leaving validation accuracy consistently at a low rate. Data augmentation helps generate artificial variation of sample images on each epoch during training. This artificial variation, in return, generalizes the sample better and reduces the extend of outliers.

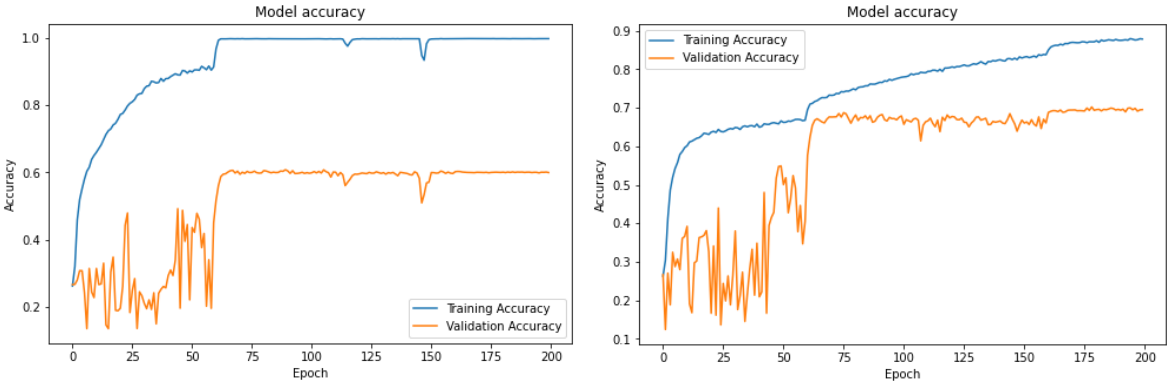

Figure 16. Comparison of validation accuracy of non-augmented (on the left) and augmented (on the right) data sample on FER2013 dataset using ResNet18+CBAM

Resnet18 + CBAM's hyperparameters on different datasets

The model is tested using three datasets: FER2013, CK+, and JAFFE. We found that optimum setting of hyperparameters for each dataset is different. The main contribution to this variation is the sample size and heterogeneity of the sample distribution. Variation in sample distribution can be mitigated via relevant pre-processing strategies mentioned earlier, while sample size issue is can be responded by the setting of LR, batch size, and epoch size during training. A good selection of constant LR can produce optimum accuracy performance but it needs much larger epoch size to converge and requires time consuming finetuning. On another hand, One-cycle LR and Super-converge LR by Smith may also be a quick solution. However due to our experiment, One-cycle LR only eliminates the need for heavy finetuning, while Super-converge LR shortens the training time remarkably. But both do not guarantee the optimum performance of the model. **Figure 17** presents training and validation accuracy of both LRs by Smith using ResNet18+CBAM on FER2013. Based on experiment results, One-cycle LR reaches its highest validation accuracy of **65.99%** within 200 epochs, and super-converge LR learns its best at **65.37%** within 50 epochs. These performances are still far below the true capability of our model.

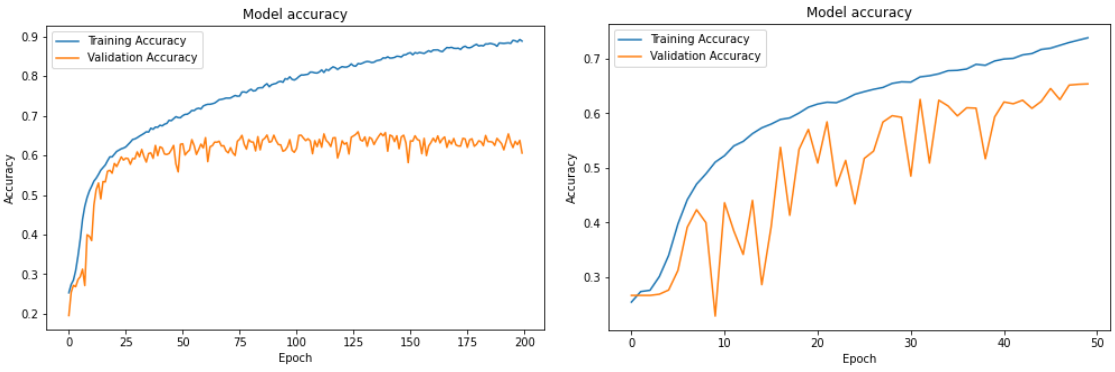

Figure 17. Training and validation accuracy of One-cycle LR (on the left) and Super-converge LR (on the right) using ResNet18+CBAM on FER2013

To resolve the issue, we use schedule LR which reduces the burden of heavy finetuning and can be easily maneuvered to optimize the model. The schedule LR ranges from 0.01 to 0.001, and up to

0.0001 for FER2013 (**Table 1** provides the detail of these settings, including other hyperparameters). Based on the optimum performances obtained, it seems to explain that bigger range of LR is needed for a smaller dataset, i.e., JAFFE, followed by a medium range on CK+, and the smallest range on FER2013. Following this nature, the number of epochs should also be adaptive to the LR ranges. In this case, the bigger range requires less epoch number for training, and it is the opposite for the smaller LR. We also discover a recognizable pattern of batch size setting. An effective batch size tends to be proportionally opposite to the sample size. In our experiments, the batch size is set to an approximate **70%**, **50%**, **20%** of the validation set of JAFFE, CK+, and FER2013 respectively based on the fact that JAFFE has very small sample size, followed by a larger CK+, and the largest FER2013. Too small batch size causes the model to learn from a small portion of data and easily overfit, resulting in random fluctuations in validation accuracy during training. While too big batch size imposes an over generalization from large portion of data which often prevents the model from learning effectively, and thus resulting in underfit. After an extensive finetuning, we achieve the optimum performances of ResNet18 + CBAM on the three datasets. **Figure 18**, **19**, and **20** depict the performance plots of training and validation loss and accuracy. We save the best model of each experiment and evaluate them with the test sets that has never involved in the training phase. We obtain the accuracy results of **71.43%**, **100%**, and **100%** on the test sets for FER2013, CK+, and JAFFE respectively.

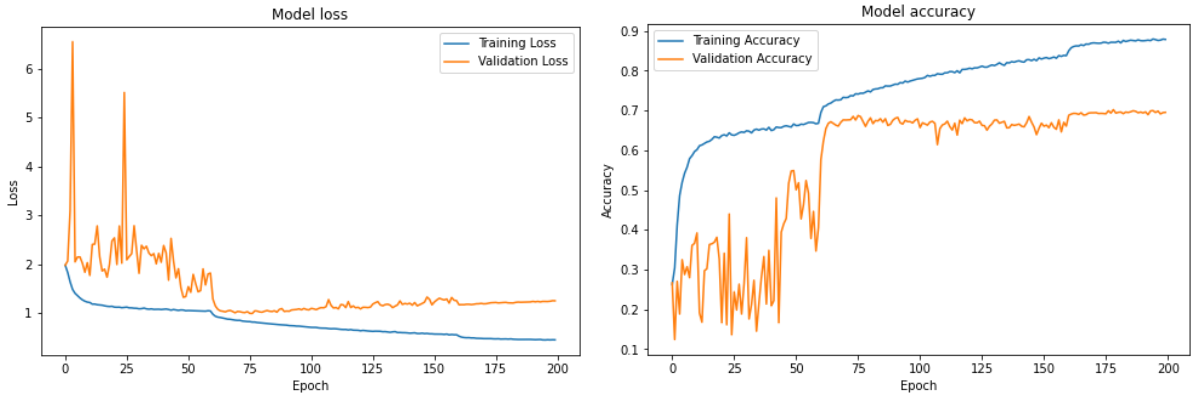

Figure 18. Plot of the model training and validation loss and accuracy on FER2013 dataset

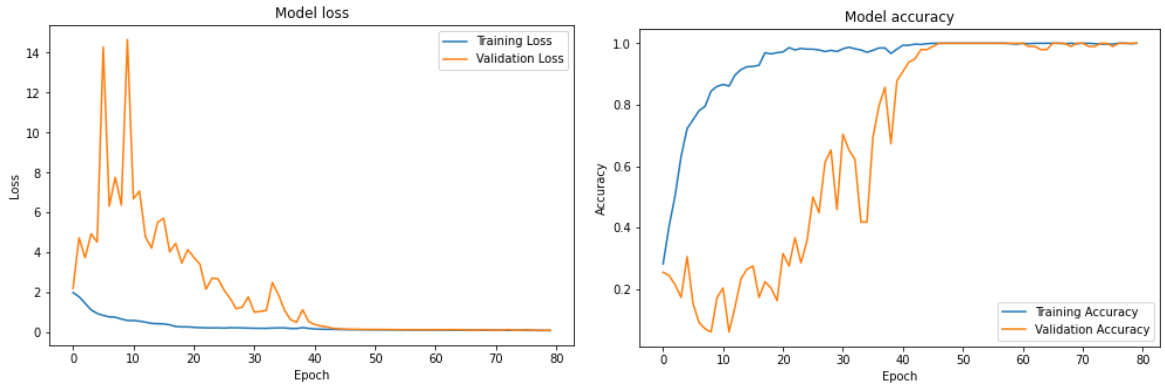

Figure 19. Plot of the model training and validation loss and accuracy on CK+ dataset

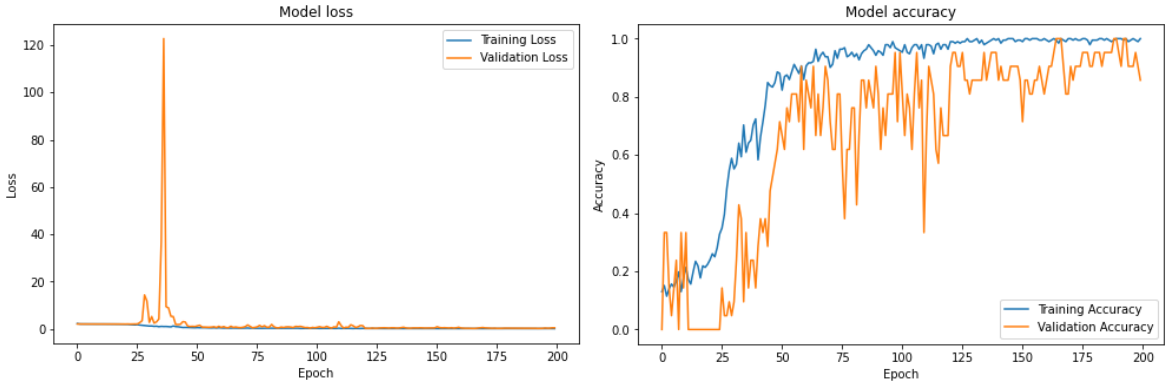

Figure 20. Plot of the model training and validation loss and accuracy on JAFFE dataset

Resnet + CBAM in different depths of convolutional layers

In addition to finetuning hyperparameters, we also study the effect of network depth on the model performance. We additionally attempt the depth of 34 and 50 of convolutional blocks to form ResNet34 and ResNet50 respectively. **Figure 21** and **22** demonstrate the training and validation loss and accuracy on FER2013 using ResNet34 and ResNet50. As seen, the validation accuracy of deeper network does not provide any concrete evidence of improvement. Best validation accuracy recorded is ResNet18 = **70.21%**, ResNet34 = **69.92%**, and ResNet50 = **70.21%**. We can notice that the validation loss and accuracy trend is more fluctuating when the network depth is smaller. More importantly, we learn that the deeper the network, the validation loss increases faster. This shows the sign of a rapid overfitting since the image features have gone through more filters (feature extraction). Based on this evidence, the depth of the network has no positive impact on FER with small number of target labels, in such case of FER2013, but rather pressurizes on computing efficiency.

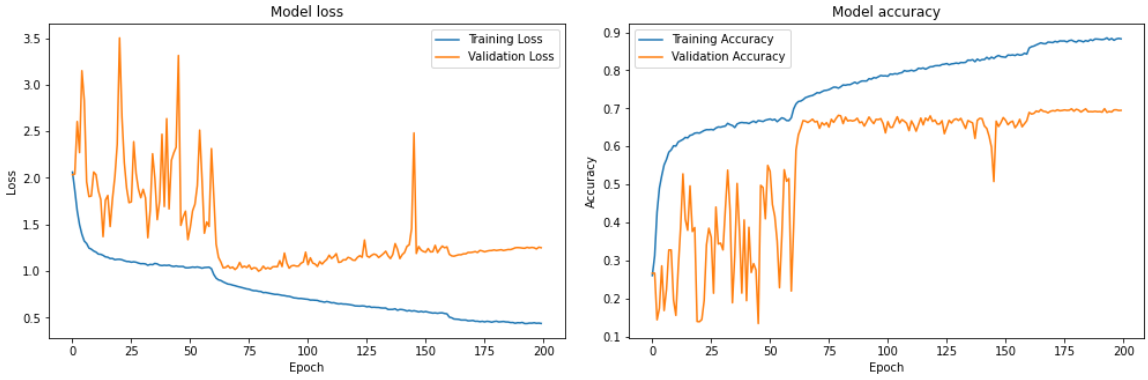

Figure 21. Plot of the model training and validation loss and accuracy on FER2013 dataset using ResNet34

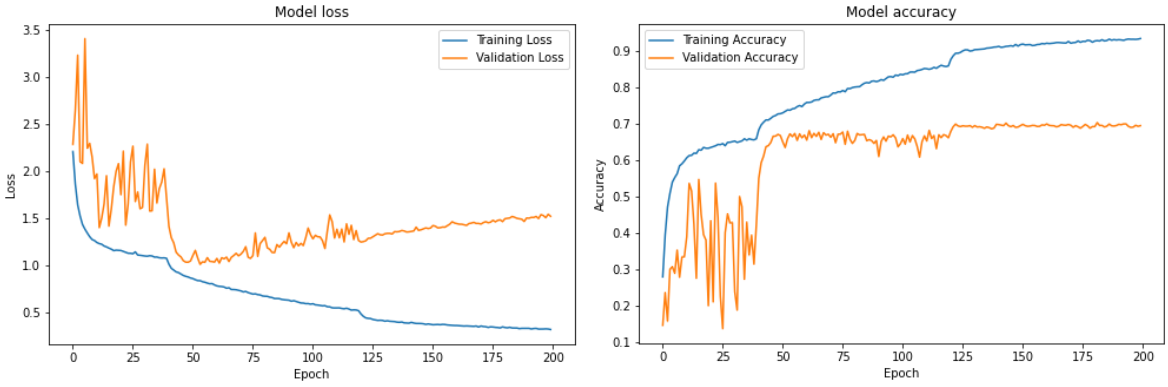

Figure 22. Plot of the model training and validation loss and accuracy on FER2013 dataset using ResNet50

ResNet18 + CBAM vs. other studies

To see how effective the model in emotion detection in FER, we compare our experiment performances in FER2013, CK+, and JAFFE with other authors who recently worked on FER using CBAM or powerful network architecture like VGG16 and using the same datasets. Some of the authors used the same ResNet architecture but had different design when integrating CBAM module to form their final network architectures. **Table 3** summarizes ResNet + CBAM architectures designed by Wen et al. (2020), Sun (2021), and Zhang et al. (2021).

Table 3. Existing ResNet + CBAM architecture designs and their summary description

Author(s)	Models	Descriptions
Wen et al. (2020)	ResNet+CBAM	Each "residual module + CBAM" block is placed after the first convolutional block of the network. There are 8 "residual module + CBAM" blocks placed consecutively till the end of network.
Sun (2021)	ResNet18+CBAM	The CBAM is integrated into the last convolutional block of the Resnet18.
Zhang et al. (2021)	ResNet34+CBAM	The CBAM is integrated into the first and the last convolutional block of the ResNet34

Based on the experiment results, our model outperformance all existing studies by producing the maximum accuracy at **100%** in both CK+ and JAFFE datasets. On FER2013 dataset, the performance of the model is slightly better than Cao et al. (2020), and Sun (2021) who used VGG19+CBAM and ResNet18+CBAM model respectively. However it is still inferior to ResNet+CBAM model of Wen et al. (2020) and VGG16 model of Li et al. (2020). **Table 4** provides the summary of these performance comparisons.

Table 4. Performance comparison between recent existing works and ours

Author(s)	Best proposed model	Accuracy (%)		
		FER2013	CK+	JAFFE
Wen et al. (2020)	ResNet+CBAM	73.50	-	98.90
Cao et al. (2020)	VGG19+CBAM	71.00	92.00	-
Li et al. (2020)	VGG16	**75.82**	98.68	98.52
Sun (2021)	ResNet18+CBAM	71.25	94.57	-
Zhang et al. (2021)	ResNet34+CBAM	-	95.10	-
Ours (ResNet18+CBAM)		71.43	**100**	**100**

Detail performance on FER2013 dataset

We produce the confusion matrix of the model performance on FER2013 dataset. **Figure 23** presents the plot of confusion matrix that shows the detail of accuracy performance with FER2013 test set. It is noticeable that the majority of the errors in prediction is under "Anger", "Fear", and "Sadness" emotion categories, where the most accurate prediction is "Happiness", followed closely by "Disgust", and "Surprise" emotions. "Anger", "Fear", and "Sadness" are the most confusing categories since they are all same negative feeling, except that they differ in emotional degree. Based on real observation into the dataset images, many labels in these categories are wrongly assigned between them. For example, a sad face is labeled as "fear" and vice versa. It is also worth to know that human accuracy of FER2013 dataset is only **65±5%** due to the problem of missed labeling (Goodfellow et al., 2013).

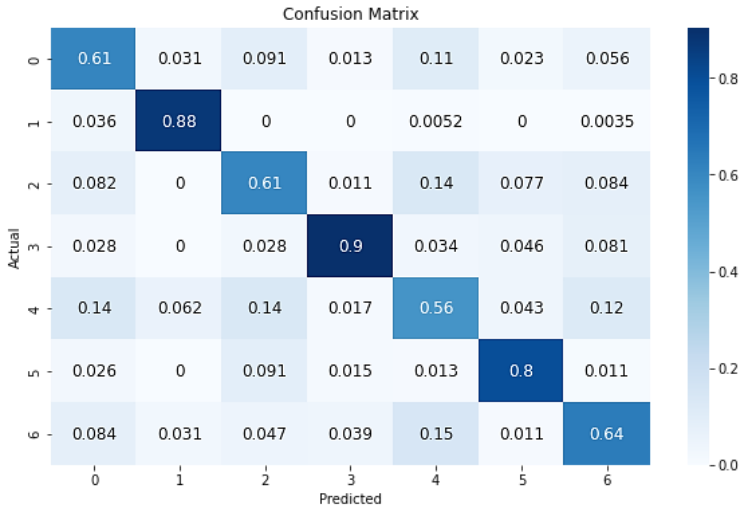

Figure 23. Confusion matrix of accuracy performance on FER2013 test set
(where 0 = "anger", 1 = "disgust", 2 = "fear", 3 = "happiness", 4 = "sadness", 5 = "surprise", and 6 = "neutral")

VI. Conclusion and future work

This study introduces a new way of employing existing architecture of CNN, i.e., ResNet in combination of attention block, called CBAM, to resolve the emotion recognition using facial expression images. By benchmarking with recent works of other studies, we can conclude that our model stands at the top rank for lab or strictly constraint datasets like CK+ and JAFFE. Although it shows compromise in performing classification on wild dataset, i.e., FER2013, comparing to some existing models, it still exceeds the human capability to recognize the emotions on that dataset. The model takes the advantage of memory efficiency in ResNet structure that tackles gradient explosion or vanishing, and efficient feature extraction of attention module to accurately pinpoint the subtle distinction of human facial expressions. The model's overall performance also shares a significant contribution to data pre-processing strategies and augmentation of sample. It is not wise to compromise the idea of "garbage in – garbage out". The model depends entirely on the reliability of the data sample. At another corner, face-detection, cropping, and global gamma equalization during the pre-processing steps have shown their remarkable efficiency in normalizing the images and enhancing the feature of the facial expressions, while data augmentation enlarges the intra-boundary of classes and thus improve generalization of the model. Another crucial factor in sharpening the model performance is the optimization of hyperparameters such as optimizer, LR, and batch size. We have learned that Adam and SGD are the most promising optimizers for FER task. Schedule LR is found to be effective in optimizing the accuracy. It also reduces the complexity of the finetuning task and can be customized to meet the challenge of sample variation easily. Lastly, we are aware that batch size can cause overfitting or underfitting if not adjusted correctly. The batch size should be set in relation to the size of the training sample.

Despite the fruitful findings, we have experienced challenges during the project that could be informative for any related future work. First of all, we have conducted the study experiments using Google Colab engine that has limitation for user in term of computing power and duration of experiment. This constraint limited our capability to have an extensive experiment in training the model. Any future study should be conducted using local GPU power to guarantee the uninterrupted experience. Additionally, our time shortage did not allow a wide exploration into many other FER datasets. Model should be further tested with larger epochs, more numbers of datasets, and possibly verified through cross-validation sampling technique to concretely confirm its realistic performance. We also suggest that when using existing concepts from existing literature, ones should consider the possible errors that may happen due to the different contextual use or other unknown reasons. Such a case of false positive claim of ***emboss effect*** and ***LBP*** on data pre-processing steps shows an example of this conceptual errors. We hope that readers find our work informatively and practically useful and contributive to the knowledge of future machine learning.

VII. References

ANAKI, D., BREZNIAK, T. & SHALOM, L. 2012. Faces in the face of death: Effects of exposure to life-threatening events and mortality salience on facial expression recognition in combat and noncombat military veterans. *Emotion,* 12**,** 860.

CAO, W., FENG, Z., ZHANG, D. & HUANG, Y. 2020. Facial Expression Recognition via a CBAM Embedded Network. *Procedia Computer Science,* 174**,** 463-477.

CHEN, X., QIAN, Y., FU, S. & SONG, Q. Real-time patient facial expression recognition using convolutional neural network. 2019 International Conference on Image and Video Processing, and Artificial Intelligence, 2019. International Society for Optics and Photonics, 113210R.

CUI, Y., WANG, S. & ZHAO, R. 2021. Machine Learning-Based Student Emotion Recognition for Business English Class. *International Journal of Emerging Technologies in Learning,* 16.

GOODFELLOW, I. J., ERHAN, D., CARRIER, P. L., COURVILLE, A., MIRZA, M., HAMNER, B., CUKIERSKI, W., TANG, Y., THALER, D. & LEE, D.-H. Challenges in representation learning: A report on three machine learning contests. International conference on neural information processing, 2013. Springer, 117-124.

KANG, M.-S. 2012. A Study on Efficient Facial Expression Recognition System for Customer Satisfaction Feedback. *Convergence Security Journal,* 12**,** 41-47.

KHAIREDDIN, Y. & CHEN, Z. 2021. Facial Emotion Recognition: State of the Art Performance on FER2013. *arXiv preprint arXiv:2105.03588*.

KHATRI, N. N., SHAH, Z. H. & PATEL, S. A. 2014. Facial expression recognition: A survey. *International Journal of Computer Science and Information Technologies (IJCSIT),* 5**,** 149-152.

KHEMAKHEM, F. & LTIFI, H. Facial Expression Recognition using Convolution Neural Network Enhancing with Pre-Processing Stages. 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), 2019. IEEE, 1-7.

KIMURA, S. & YACHIDA, M. Facial expression recognition and its degree estimation. Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997. IEEE, 295-300.

KO, B. C. 2018. A brief review of facial emotion recognition based on visual information. *sensors,* 18**,** 401.

KUMAR, G. R., KUMAR, R. K. & SANYAL, G. Facial emotion analysis using deep convolution neural network. 2017 International Conference on Signal Processing and Communication (ICSPC), 2017. IEEE, 369-374.

LI, J., JIN, K., ZHOU, D., KUBOTA, N. & JU, Z. 2020. Attention mechanism-based CNN for facial expression recognition. *Neurocomputing,* 411**,** 340-350.

LI, S. & DENG, W. 2020. Deep facial expression recognition: A survey. *IEEE transactions on affective computing*.

MEHMOOD, R. M., DU, R. & LEE, H. J. 2017. Optimal feature selection and deep learning ensembles method for emotion recognition from human brain EEG sensors. *Ieee Access,* 5**,** 14797-14806.

PAN, S. T. & LI, W. C. 2020. Fuzzy-HMM modeling for emotion detection using electrocardiogram signals. *Asian Journal of Control,* 22**,** 2206-2216.

SMITH, L. N. Cyclical learning rates for training neural networks. 2017 IEEE winter conference on applications of computer vision (WACV), 2017. IEEE, 464-472.

SMITH, L. N. & TOPIN, N. 2018. Super-convergence: Very fast training of residual networks using large learning rates.

SUN, J., JIANG, J. & LIU, Y. An Introductory Survey on Attention Mechanisms in Computer Vision Problems. 2020 6th International Conference on Big Data and Information Analytics (BigDIA), 2020. IEEE, 295-300.

SUN, L. Research on Expression Recognition Algorithm Based on Attention Mechanism. Journal of Physics: Conference Series, 2021. IOP Publishing, 042069.

VYAS, A. S., PRAJAPATI, H. B. & DABHI, V. K. Survey on face expression recognition using CNN. 2019 5th international conference on advanced computing & communication systems (ICACCS), 2019. IEEE, 102-106.

WEN, P., DING, Y., WEN, Y., DENG, Z. & XU, Z. Facial expression recognition method based on convolution neural network combining attention mechanism. International Conference on Artificial Intelligence and Security, 2020. Springer, 136-147.

WOO, S., PARK, J., LEE, J.-Y. & KWEON, I. S. Cbam: Convolutional block attention module. Proceedings of the European conference on computer vision (ECCV), 2018. 3-19.

YALAMANCHILI, B., DUNGALA, K., MANDAPATI, K., PILLODI, M. & VANGA, S. R. Survey on Multimodal Emotion Recognition (MER) Systems. Machine Learning Technologies and Applications: Proceedings of ICACECS 2020, 2021. Springer, 319-326.

ZHANG, X., CHEN, Z. & WEI, Q. Research and Application of Facial Expression Recognition Based on Attention Mechanism. 2021 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), 2021. IEEE, 282-285.

ZHANG, Y. & HUA, C. 2015. Driver fatigue recognition based on facial expression analysis using local binary patterns. *Optik,* 126**,** 4501-4505.