ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

REAL TIME EMOTION BASED MUSIC PLAYER USINGCNN ARCHITECTURES

Assistant Professor Mrs. Indumathi S K[1] Sireesha K[2], Kavan MC[2]
[1] *Professor of Dr Ambedkar Institute of Technology, Dept of MCA, Bangalore-560060, Karnataka, India*
[2] *Students of Dr Ambedkar Institute of Technology, Dept of MCA, Bangalore-560060, Karnataka, India*

ABSTRACT

Emotion detection is the process of detecting a human being's emotions based on various facial cues and visual information. This field has gained much traction since the popularity of deep learning. Emotion detection has also given rise to many applications that had not been thought of before. One of the areas that are heavily associated with emotions is music. Music can invoke particular emotions of the listener, and a person feeling a certain emotion would look for a similar song. We use our emotion detection model to associate these emotions with a music player that plays music that accompanies user experiences. The model we designed includes two convolutional neural networks (CNN) models: a five-layer model and a global average pooling (GAP) model. We combined these CNN models with transfer-learning models. For our transfer-learning models, we used three pre-trained models: ResNet50; SeNet50; VGG16. Our results are comparable with the state-of-the-art models; however, our models are more efficient in performance.

INDEX TERMS—***Class Weighting, Convolutional Neural Network, Emotion Detection, Ensemble, Global Average Pooling, Transfer Learning***

I. INTRODUCTION

Facial emotional recognition (FER) [1] is a rising field in deep learning. FER aims to predict the emotion of the face based on the visual face. Emotion detection is a complex process as extracting visual cues from the face is complicated since they are not always obvious. They can be very subtle, and even in some cases, non-existent. There exist models that can detect emotion with great accuracy, but they only do that in a controlled environment. In a real environment, the problem becomes much more challenging as we have to factor in lighting, different facial structures, occlusions, head pose, etc. However, the last decade has caused this field to improve to perform better than typical humans drastically. This can be mainly attributed to the popularity of deep learning algorithms and computer vision. This has led to the rise of various applications such as medical treatments, social robotics, driver fatigue surveillance, etc.

Our contribution mainly focuses on enhancing a FER model and using it for a real-world application. We designed a model that combines two vanilla CNN models and a few transfer learning models. We named the two vanilla CNN models as a five-layer model and a global average pooling (GAP) model described in detail later. The transfer learning models we utilized are ResNet50, SeNet50, and VGG16, described in detail later. The GAP model stands out from the rest as it reduces the parameters by almost 80% while maintaining a decent accuracy. This lightweight model helps us in the real world aspect as it is easily mountable on small devices. We also explored a wide variety of ideas to enhance the model further. The scope of our exploration included data augmentation, class weighting, adding auxiliary data, and ensembling. We want to investigate how well we can read emotions from the user and recommend music that matches the user's mood. Although facial recognition and music recommendation are well-investigated topics, the combination has not been explored, which is what we want to study.

The rest of the paper is arranged as follows. Section II reviews some existing models related to the problem of facial expression recognition. Section III gives a detailed account of our approach. Section IV outlines and discusses our experiments and the corresponding results obtained, followed by Section V's conclusion.

II. RELATED WORK

The FER2013 dataset was created by Goodfellow et al. [2] to make it a Kaggle competition to boost research in emotion detection. The top three teams had used some variant of convolutional neural network with image transformation. The winner Yichuan Tang [3] had achieved an accuracy of 71.2%. They had used an SVM loss function as well as an L2-SVM loss function. The idea of using these loss functions was novel at the time and resulted in par excellence performance. Also, the model performs better on benchmark datasets such as MNIST [4], CIFAR-10 [5], ICML-2013 [2]. The limitation of the paper is that it does not explore other multi-class SVM formulations.

W. Deng and S. Li [6] did a recent survey that goes deep into the present state of application of deep learning to FER. Another paper that we would like to discuss is that of D. V. Sang et al. [7], as this paper had much influence on the work carried out by us. This paper's central idea was to use convolutional neural networks to extract semantic information

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

from the face in an automated manner without putting in any extra effort in manually designing the feature descriptors. When applied to the dataset of the FER2013 competition, it far exceeds the winner of the competition. The paper explains this by saying that they tried various combinations of training tricks and loss functions. The paper also mentions that their method has far fewer parameters, making it more efficient, and this is important as it makes it suitable for real-time systems.

Pramerdorfer et al. [8] implemented an ensemble model of six states of the art Convolutional Neural Networks (CNN) based predictors that achieved a performance of 75.6%. The paper goes to great lengths to identify bottlenecks. However, the issue with the paper is that it does not provide any method for data augmentation. Krizhevsky et al. [9] built a deep CNN model that classified almost one million images of the ImageNet dataset into its different classes. It achieved exceptional performance just based on supervised learning. This model's issue is that the model drastically degrades in performance if we remove any convolutional layer. Z. Yu et al. [10] builds a face emotion detector model using 3 states of art predictors. This model can achieve the state of the art performance but randomly initializing a single model gives slightly worse than expected performance. B-K Kim et al.[11] has used the idea of considering aligned and non-aligned states of the face for increasing the accuracy of the prediction. However, this research is not useful for emotion detection. Instead, it is more suited for facial recognition.

III. METHODOLOGY

A. Models

While building the model architectures, our main goals were to maximize the accuracy of the FER2013 dataset's test set and not compromise on the real-time aspect. So, we tried to balance accuracy and the total number of parameters while building our models. We developed two CNN models and three transfer learning models, which we describe in detail below. The major hurdle in FER dataset was the class imbalance in the dataset. We addressed this problem by exploring class weighting, which led to some promising results. Finally, we achieved our highest accuracy of 76.12% by ensembling all of these individual models.

1) Model 1 - Five-Layer Model: This model, as the name suggests, consists of five layers. The first three stages consist of convolutional and max-pooling layers each, followed by a fully connected layer of 1024 neurons and an output layer of 7 neurons with a soft-max activation function. The first convolutional layers utilized 32, 32, and 64 kernels of 5*5, 4*4, and 5*5. These convolutional layers are followed by max-pooling layers that use kernels of dimension 3*3 and stride 2, and each of these used ReLu for the activation function. Batch normalization was added at every layer and 30% dropout after the last fully connected layer. This was done to improve performance further.

The visual representation of the model architecture is shown in Fig. 1. We trained the model for 350 epochs. Stochastic Gradient Descent (SGD) [12] was used as the optimizer and cross-entropy as the loss function. The learning rate and batch size were set to 0.01 and 256, respectively.

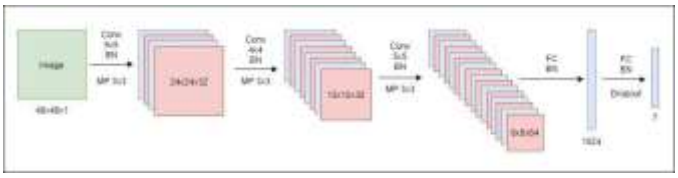

Fig. 1: 5-Layer Model Architecture

2) Model 2 - Global Average Pooling (GAP) Model: Our second proposed CNN architecture aims to reduce the vast amount of generally present parameters in a regular convolutional network while still maintaining a decent accuracy. This is necessary to develop a fast real-time CNN, reducing the gap between slow performances and real-time architectures. It reduces the number of parameters by eliminating the final fully connected layers. Most of the CNNs usually contain more than 80% of the total parameters in the fully connected layers at the end.

The model architecture is as described in Fig. 2. The model consists of a standard fully convolutional network with 9 Convolution layers, ReLu layers, Batch Normalization layers, and a final GAP layer. The name of the model we have given comes from this final layer. It contains 642,935 parameters, out of which 641,463 are trainable, much less than traditional deep learning models for such a task containing more than 2 million parameters. Dropout is used after each convolution step to regularize the network. The model uses GAP to eradicate the fully connected layers. This is achieved by having in the last convolution layer the same number of feature maps as the number of classes in the dataset we want to predict and finally applying the softmax activation function to each of those feature maps. The model is trained with an ADAM optimizer.

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

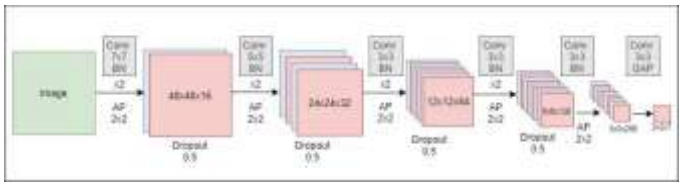

Fig. 2: GAP Model Architecture

A. Transfer Learning

The significant challenges of the FER2013 dataset were the small size of the dataset and imbalance in the classes.Transfer Learning helped in improving the accuracy of such a dataset. We utilized SeNet50 [13], ResNet50 [14], and VGG16 [15] as the pre-trained models, which are complex models with a large number of parameters, and they are known to give good results on image captioning tasks. So we utilized it for our task of facial expression recognition. We used the Keras library for the same. These models had specific input requirements to resize and recolor the 48*48 grayscale images of FER2013 to meet those requirements for each transfer learning model during training time.

1) *Model 3 - Fine-Tuning ResNet50*: ResNet50 was the first transfer learning model explored. ResNet stands for Residual Network. This model is fifty layers deep. This transfer learning model's input requirement was first met by resizing the images from the FER2013 dataset during training time. We started by replacing the original output layer and added two fully connected layers of 1024 and 4096. Finally, an output layer of size 7 (one for each class) was added with a softmax activation function. To further optimize the accuracy, the first few layers in ResNet was frozen, and the rest of the network was kept trainable. Stochastic Gradient Descent (SGD) [12] was used as the optimizer. Batch size and learning rate were fixed at 64 and 0.001, respectively. This model was trained for 100 epochs and achieved 73.22% accuracy on the test set.

2) Model 4 - Fine-Tuning SeNet50: SeNet50 was the second transfer learning model explored. SeNet stands for Squeeze and Excitation Network. This model is fifty layers deep. It resembles the structure of ResNet50. So, we trained the model on the same set of parameters used for ResNet50: the batch size of 64 and the learning rate of 0.001. This model was trained for 150 epochs and achieved 72.19% accuracy on the test set.

3) Model 5 - Fine-Tuning VGG16: VGG16 was the third and final transfer learning model explored. This model is much shallower than ResNet50 and SeNet50 as it consists of only 16 layers. However, VGG16 is more complicated in its architecture and has much more parameters. We froze all the pre-trained layers and added two fully connected layers of 1024 and 4096 with fifty percent dropout. Adam optimizer was used while training. Batch size and learning rate were fixed at 128 and 0.01. This model was trained for 120 epochs and achieved 69.30% accuracy on the test set.

B. Implementing The Models

1) Data Preparation: There are several different versions of the FER2013 dataset. These differ in labeling, image size, and directory structure. We tackle these differences by partitioning all the input datasets into seven directories, with each directory representing each class in the FER2013 dataset. During training, images were loaded in batches from disk, and the images were resized using Keras data generators.

2) Data Augmentation: We applied data augmentation to increase the size of the dataset and improve accuracy. Some of the data augmentation techniques applied were horizontal mirroring, image zooms, degree rotations, and horizontal/vertical shifting.

3) Class Weighting: One of the FER2013 dataset's major problems was the imbalance in the number of samples for different classes. We fix this by applying class weighting[16] inversely proportional to it. This considerably improved performance, especially with disgust class where the misclas- sification rate dropped from 62% to 35%.

w = nsamples j nj (1)

4) Ensembling: We performed ensembling with soft voting of the five models to improve our highest test accuracy to 76.12%.

C. Web-App

We have used the chrome browser application to run the Web-App. The flow of this application is as follows:

1) Run the capture.py file. It will then trigger the HTML file, which will show the CSS-HTML-based music player (webpage)
2) To play any music, click on the play button shown on the song or have a plus sign to add it to the queue.
3) Another option based on emotion will be shown on the right upper side, select it. JavaScript will trigger the python function.
4) Camera will start and record the back-end image and go for ten successful images that contain any face.
5) Generate emotion prediction on those images, get the aggregate result of those ten results, choose appropriate emotion, and forward it to JS script.
6) JS chooses a random song of that genre to play.
7) Whenever a song will go to an end, it will repeat the same back-end process so that the user will not be aware of it.

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

IV. EXPERIMENT AND ANALYSIS

A. Dataset

For this work, we have used the FER2013 dataset [17]. FER2013 is a popular and complex benchmark dataset used in many competitions and research. It has a human accuracy of 64±5%. It consists of around 36,000 images, which are normalized to 48x48 grayscale images. The images are divided into seven different classes, with each class representing a facial expression. The different classes available are Happy (8,988), Neutral (6,197), Sad (6,076), Angry (4,954), Surprise (4,001), Fear (5,120) and Disgust (548). The high imbalance in the classes makes it a very challenging dataset.

B. Accuracy

Table I shows the accuracy of the individual models and also the accuracy of the ensemble model. The accuracy of the transfer learning models are higher due to the complexity of their design. This comes at a price of a much higher number of parameters, which is not a desirable trade-off for real-world applications. We have further shown the accuracy obtained by implementing class weighting, where the individual accuracy of the models does not necessarily improve. However, it depicts the increase in the ensemble model's overall accuracy by handling the class imbalance problem in our dataset. We achieved our highest test accuracy of 76.12% for our ensemble model after applying class weighting.

TABLE I: Our Model's Accuracy on FER2013 dataset

Model	Accuracy	Class Weighted
ResNet50	73.22%	72.28%
SeNet50	72.19%	70.99%
VGG16	69.30%	69.15%
5-Layer Model	65.67%	-
GAP Model	66.54%	-
Ensemble	**74.84%**	**76.12%**

Table II compares the total number of parameters and accuracy of our model with other models for the FER2013 dataset. The results show that the GAP model had the least number of parameters (642,935) than all other models while achieving a decent accuracy. The GAP model's weight file occupies only 20 MB of space compared to other models that occupy over 200 MB of space. This makes the model mountable even on small devices, which will require such an application. We have also depicted results from Deep-Emotion [18], and Pramerdorfer et al. [8].

TABLE II: Comparison with Other Models

Model	Parameters (in Million)	Accuracy
Human-Level	-	**64±5%**
Deep-Emotion [18]	**43 M**	**70.02%**
Pramerdorfer et al. [8]	**5.3 M**	**75.2%**
5-Layer Model (Our Model)	**2.5 M**	**65.67%**
GAP Model (Our Model)	**0.64 M**	**66.54%**
Ensemble (Our Model)	-	**76.12%**

C. Confusion Matrix

The confusion matrix of the Ensemble model is shown in Fig. 3. The rows correspond to the true values, and the columns correspond to our predictions. As we can clearly see, Fear is the class where our network fares the worst, and Happy is the most successful class. Another interesting observation is that 18 percent of the images labeled as fear are predicted as sad by our model, which is similar to humans' mispredictions on the same image.

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

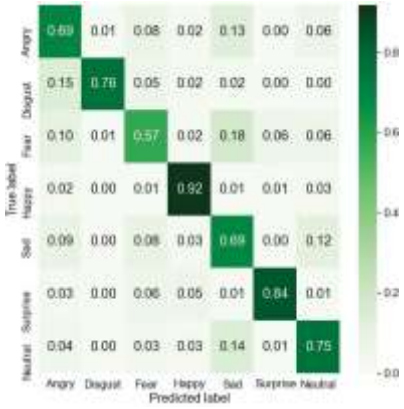

Fig. 3: Confusion Matrix

D. WebApp

Fig. 4 shows the emotion detection process in action andFig. 5 shows the frontend of the web app display

(a) Happy

(b) Angry

(c) Neutral

(d) Neutral

Fig. 4: Emotion Detection

Fig. 5: WebApp display

E. Feedback Analysis

Fig. 6 shows the ratings of 53 users, suggesting a positive outlook. Criticism included:

- More music variety
- Emotion recognition may not work 100% at times

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

• Needs to detect more emotions outside of the 7 utilized. Fig. 7 shows the different questions that were asked for the user feedback and the average rating based on the responsefrom 53 users

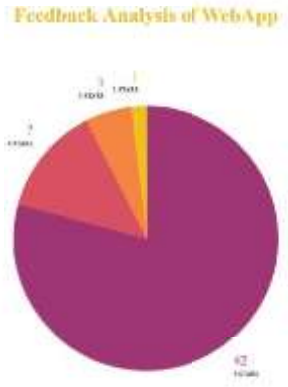

Fig. 6: Feedback of 53 users

Questions	Average Rating
The ease of using the interface	4.4
Song relevance to the emotion	3.75
Emotion detection response	3.8
Time delay in queuing the next song	4.8
Overall rating	4.42

Fig. 7: Questions Asked

V.CONCLUSION AND FUTURE WORK

In this paper, an emotion detection model is proposed to recommend music based on one's mood. Our work aims to achieve the highest possible accuracy while not compromising the real-time aspect to apply to the real-world scenario. We explored several models built differently, including vanilla CNNs and pre-trained networks based on ResNet50, SenNet50, and VGG16. One model that stood out was the GAP model that managed to achieve an accuracy of 66.54% while reducing the number of parameters by around 80%. This was a breakthrough as such a lightweight model is easily mountable on small devices, which adds to real-world scenarios' applicability. We further solved the challenging class imbalance problem of the FER2013 dataset by using class weighting and data augmentation. We achieved our best test accuracy of 76.12% by ensembling all of our models.

As part of future work, our models' accuracy could be further improved by applying landmark detection techniques that cancel out irrelevant facial features from the image during training. We could better handle images with multiple classes of emotion by utilizing a multi-label classification technique. We want to adapt our model in some other real-world scenarios, like a teaching-learning environment, where the teacher could improve his/her teaching based on the feedback received by utilizing the model or in Psychology, where it would help the psychologist analyze and study a person's behavior.

REFERENCES

1. *B. Ko, "A Brief Review of Facial Emotion Recognition Based on Visual Information," Sensors (Basel, Switzerland), vol. 18, 2018.*
2. *I. J. Goodfellow, D. Erhan, P. L. Carrier, A. C. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, Y. Zhou, C. Ramaiah, F. Feng, R. Li, X. Wang, D. Athanasakis, J. Shawe-Taylor, M. Milakov, J. Park, R. T. Ionescu, M. Popescu, C. Grozea, J. Bergstra, J. Xie, L. Romaszko, B. Xu, C. Zhang, and Y. Bengio, "Challenges in representation learning: A report on three machine learning contests," Neural networks : the official journal of the International Neural Network Society, vol. 64, pp. 59–63, 2015.*
3. *Y. Tang, "Deep Learning using Linear Support Vector Machines," arXiv: Learning, 2013.*
4. *L. Deng, "The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]," IEEE Signal Processing Magazine, vol. 29, pp. 141–142, 2012.*

ISSN (Online): 2455-3662
EPRA International Journal of Multidisciplinary Research (IJMR) - Peer Reviewed Journal
Volume: 8| Issue: 7| July 2022|| Journal DOI: 10.36713/epra2013 || SJIF Impact Factor 2022: 8.205 || ISI Value: 1.188

5. *A. Krizhevsky, "CIFAR-10 (Canadian Institute for Advanced Research)." [Online]. Available: http://www.cs.toronto.edu/ kriz/cifar.html*
6. *S. Li and W. Deng, "Deep Facial Expression Recognition: A Survey," ArXiv, vol. abs/1804.08348, 2018.*
7. *D. V. Sang, N. V. Dat, and D. P. Thuan, "Facial expression recognition using deep convolutional neural networks," 2017 9th International Conference on Knowledge and Systems Engineering (KSE), pp. 130– 135, 2017.*
8. *C. Pramerdorfer and M. Kampel, "Facial Expression Recognition us- ing Convolutional Neural Networks: State of the Art," ArXiv, vol. abs/1612.02903, 2016.*
9. *A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, pp. 84 – 90, 2012.*
10. *Z. Yu and C. Zhang, "Image based Static Facial Expression Recognition with Multiple Deep Network Learning," Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, 2015.*
11. *B.-K. Kim, S.-Y. Dong, J. Roh, G. min Kim, and S. Lee, "Fusing Aligned and Non-aligned Face Information for Automatic Affect Recognition in the Wild: A Deep Learning Approach," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1499–1508, 2016.*
12. *S. Ruder, "An overview of gradient descent optimization algorithms,"ArXiv, vol. abs/1609.04747, 2016.*
13. *J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7132–7141, 2018.*
14. *K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.*
15. *K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," CoRR, vol. abs/1409.1556, 2015.*
16. *A. Anand, G. Pugalenthi, G. Fogel, and P. Suganthan, "An approach for classification of highly imbalanced data using weighting and undersam- pling," Amino acids, vol. 39, pp. 1385–91, 11 2010.*
17. *Wolfram Data Repository, "FER 2013." [Online]. Available: https://datarepository.wolframcloud.com/resources/FER-2013*
18. *S. Minaee and A. Abdolrashidi, "Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network," ArXiv, vol. abs/1902.01019, 2019.*