

**Emotion Recognition on Static Images
Using Transfer Learning and Ensembling**

M.Sc. THESIS

Hüseyin ABANOZ

Department of Computer Engineering

Computer Engineering Programme

JUNE 2018

**Emotion Recognition on Static Images
Using Transfer Learning and Ensembling**

M.Sc. THESIS

**Hüseyin ABANOZ
(504141545)**

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Prof. Zehra ÇATALTEPE

JUNE 2018

**Transfer Öğrenme ve Topluluk Öğrenmesi Kullanarak
Durağan Görüntüler Üzerinde Duygu Tanıma**

YÜKSEK LİSANS TEZİ

**Hüseyin ABANOZ
(504141545)**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof. Zehra ÇATALTEPE

HAZİRAN 2018

Hüseyin ABANOZ, a M.Sc. student of ITU Graduate School of Science Engineering and Technology 504141545 successfully defended the thesis entitled “Emotion Recognition on Static Images Using Transfer Learning and Ensembling”, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Zehra ÇATALTEPE**
Istanbul Technical University

Jury Members : **Prof. Mustafa Ersel KAMAŞAK**
Istanbul Technical University

Assoc. Prof. Songül VARLI ALBAYRAK
Yıldız Technical University

.....

Date of Submission : **4 May 2018**

Date of Defense : **8 June 2018**

To my spouse and children

FOREWORD

Many thanks to my teachers and friends for their support on thesis study.

June 2018

Hüseyin ABANOZ
(Computer Engineer)

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD.....	ix
TABLE OF CONTENTS.....	xi
ABBREVIATIONS	xiii
SYMBOLS.....	xv
LIST OF TABLES	xvii
LIST OF FIGURES	xix
SUMMARY	xxi
ÖZET	xxiii
1. INTRODUCTION	1
1.1 Emotion Recognition.....	1
1.2 Convolutional Neural Networks	2
1.3 Purpose of Thesis	5
1.4 Literature Review	6
2. Methodology.....	9
2.1 Transfer Learning	9
2.2 Implementation of Transfer Learning on CNN's	10
2.3 Models for Transfer Learning.....	10
2.4 Dataset	11
3. Experimental Setup	13
3.0.1 Experiment Evaluation	14
4. Transfer Learning.....	15
4.0.1 Transfer Learning Results	17
4.0.2 Decreasing Model Complexity	19
5. Ensembling.....	21
5.0.1 Stacking with Expert Classifiers.....	21
6. CONCLUSIONS AND RECOMMENDATIONS.....	27
REFERENCES.....	29
APPENDICES.....	31
APPENDIX A.1	33
APPENDIX A.2	33
APPENDIX A.3	34
APPENDIX A.4	34

1.0.1 Box Plot of Simplified VGGFace Experiment vs Stacked Experts	
Experiment Accuracies.....	34

ABBREVIATIONS

CPU	: Central Processing Unit
GPU	: Graphical Processing Unit
CNN	: Convolutional Neural Network
AVEC	: Audio/Visual Emotion Challenge
EMOTIW	: Emotion Recognition in the Wild
FER13	: Facial Expression Recognition 2013
MLP	: Multi Layer Perceptron

SYMBOLS

V	: Output Area Size
W	: Input Area Size
P	: Zero Padding Amount
S	: Stride
P	: Kernel Size

LIST OF TABLES

	<u>Page</u>
Table 2.1 : Samples from Dataset.....	12
Table 2.2 : Number of Instances Per Class.....	12
Table 4.1 : FER13 Validation Results After Retraining	15
Table 4.2 : Number of Convolutional Layers Experiment	17
Table 5.1 : Best FER13 Validation Results of Experimented Models	25

LIST OF FIGURES

	<u>Page</u>
Figure 1.1 : Convolution Input and Output Volumes	3
Figure 1.2 : Max Pooling	4
Figure 1.3 : Fully Connected Layer	5
Figure 2.1 : Implementing Transfer Learning	10
Figure 4.1 : VGGNet Architecture for Emotion Classification.....	16
Figure 4.2 : VGGFace Training&Validation Accuracy Chart on FER13 Dataset. (Steeper line represents the training).....	18
Figure 4.3 : VGGFace Confusion Matrix of Validation Data.	18
Figure 5.1 : Fear Expert Model Training Using Fear-Restructured FER13 Dataset.....	22
Figure 5.2 : Ensembling Diagram	24
Figure 5.3 : Accuracy Scatter Plot of 3 Different Experiments. Circle: VGGFace, Dot: Simplified VGGFace, x: Ensembled	25
Figure A.1 : VGG16 Confusion Matrix	33
Figure A.2 : Simplified VGGFace Confusion Matrix.....	33
Figure A.3 : ResNet50 Confusion Matrix.....	34

Emotion Recognition on Static Images Using Transfer Learning and Ensembling

SUMMARY

Emotion recognition task involves assigning a label to a sample from a set of emotions. Emotion set contains following emotions: Angry, Disgust, Fear, Happy, Sad, Surprise and Neutral. A model trained for this purpose should be able choose a single emotion which best describes the sample. Emotion recognition may be useful in any area where human and computer interacts which covers many areas. As a result, emotion recognition problem is studied by many researches recently. There are two popular contests to boost researches: Audio/Visual Emotion Challenge (AVEC) and Emotion Recognition in the Wild (EMOTIW).

Convolutional Neural Networks are formed using many stacked convolutional layers. Each convolutional layer consists of varying number of filters. Learning for a CNN is the action of learning the filter values. After convolutional layers, fully connected layers are added. Final layer in every CNN architecture is a fully connected layer with number of outputs equal to number of classes present in the problem. In emotion recognition problem there are 7 output nodes because there are 7 emotions.

CNNs are known to be good at computer vision tasks. In recent years, use of CNN models in emotion recognition contests has become dominant. Some researchers are trying to come up with architectures which are suitable for emotion recognition problem. Some researchers are trying to use combinations of existing architectures to make them suitable for the problem.

CNNs are powerful classifiers; however, CNNs are difficult to train. Training of a CNN model requires large amount of data and computing power. In case of small amount of data, CNNs tend to overfit the data. Convolutions done on images require a lot of calculations which takes days of execution time. Parallel processing is needed to speed up the process. Parallel computation using CPUs is not enough and expensive GPU hardware is needed.

Transfer learning emerges as a cheap and efficient way of making use of CNN classifiers. In transfer learning a CNN model which is pre-trained on for a different problem. Similarity between previous problem and current problem is important for success of transfer learning. CNN models contain many convolutional layers. Early layers in a model extracts low level features while later layers extract high level features that are specific to current problem. Transfer learning makes use of information at early layers which are applicable to many problems. Only late layers are re-trained to make the model suitable for the current problem.

Dataset is the most important resource for machine learning. FER13 dataset is used for the study. The dataset contains 32K grayscale face images. Even tough, dataset contains low quality images, which are 48X48 pixels, it is widely used due to number of samples present.

This study firstly tries to find a base model to use for transfer learning. For this purpose, different CNN architectures and models trained using different datasets are investigated. Popular VGG16, ResNet50, InceptionV3 models which are trained using Imagenet dataset are experimented. VGGFace model which is trained using Oxford Face dataset is experimented. VGG16 and VGGFace models showed most promising results. VGGFace model found to be most suitable base model for transfer learning because dataset used for pre-training resembles the dataset for emotion recognition.

Later on, a study for improving model performance using stacking is done. Expert models are trained. Each expert model is expert at classification of single emotion. The base model is ensembled with expert models to create a final classifier. A two layer MLP is used as meta learner. Experiments done on the ensembling did not give the best classifier accuracy but did give lower accuracy variance and better mean.

Experiments show that our use of ensembling together with transfer learning helps to create a good classifier. Best classifier shows 69.49 % accuracy on FER13 validation set.

Transfer Öğrenme ve Topluluk Öğrenmesi Kullanarak Durağan Görüntüler Üzerinde Duygu Tanıma

ÖZET

Duygu tanıma problemi, bir örneği duygu kümesinde bulunan duygulardan birisiyle etiketlemeyi amaçlar. Duygu kümesi 7 adet duygu barındırır: Kızgınlık, İğrenme, Korku, Üzgün, Şaşkın ve herhangi bir duygu içermeyen yalın. Duygu tanıma için eğitilmiş bir modelin, eldeki bir örnek için, duygu kümesi içinden örneği en iyi tanımlayan tek bir duyguyu seçmesi beklenir. Duygu tanıma bilgisayar ve insanların etkileşime girdiği her alanda faydalı olabilir. Bu da duygu tanıma eyleminin bir çok alanda kullanılabilieceği anlamına gelir. Duygu tanıma problemi bazı zorluklar barındırır. Bazı duygular için mikro ifadeler benzerlikler göstermektedir, bu durum duyguların birbiriyle karıştırılmasıyla sonuçlanabilmektedir. Ayrıca duygu tanıma için kullanılabiliecek veri kümeleri de sınırlıdır. Varolan veri kümeleri ya kötü kaliteli örnekler içermektedir ya da az sayıda örnek barındırmaktadır. Çoğu veri kümesi laboratuvar ortamında üretilmiştir ve gerçek hayattan örnekler ile karşılaşıldığı zaman sınıflandırıcılar kötü performans sergileyebilmektedir. Yaygın uygulama alanı ve hala geliştirmeye açık olması nedeniyle, duygu tanıma problemi bir çok araştırmacı tarafından çalışılmaktadır. Araştırmaların sayısını ve niteliğini artırmak için yarışmalar düzenlenmektedir. Popüler yarışmalar arasında Audio/Visual Emotion Challenge (AVEC) ve Emotion Recognition in the Wild (EMOTIW) yarışmaları gösterilebilir.

Duygu tanımda probleminde farklı bilgi türlerine başvurulabilir. Bu bilgi türleri sabit görüntüler, hareketli video görüntüler veya ses olabilir. Bu bilgi türleri tek tek kullanılabileceği gibi birlikte de kullanılabilir. Birlikte kullanıldığı durumlarda, her bilgi türü için ayrı bir model eğitilmekte, sonra bu modeller topluluk öğrenmesi uygulanarak birleştirilmektedir. Bu çalışmada tek bir bilgi türü kullanılmıştır. Duygu tanıma sabit görüntüler üzerinde uygulanmıştır.

Evrişimsel Sinir Ağları, CNN olarak adlandırılmaktadır, bir çok evrişimsel katmandan oluşur. Daha fazla evrişimsel katman içeren ağların yani daha derin ağların, daha başarılı oldukları gözlemlenmiştir. Bu yüzden araştırmacılar derin ağlara odaklanmıştır. Her evrişimsel katman değişen sayıda filtrelerden oluşmaktadır. Bir CNN eğitme, o CNN de bulunan filtrelerin değerlerinin öğrenilmesi eylemidir. Evrişimsel katmanları tam bağlı katmanlar izler. Her CNN mimarisi, çıkış düğümü sayısı problemde bulunan sınıf sayısına eşit olan tam bağlı katmanla biter. Duygu tanıma probleminde 7 adet sınıf bulunduğu için, eğitilen CNN modellerinin son katmanında 7 adet çıkış düğümü bulunur.

CNN ağlarının bilgisayarla görü görevlerinde başarılı olduğu bilinmektedir. Son yıllarda, duygu tanıma yarışmalarında CNN ağlarının kullanılma sıklığının arttığı görülmüştür. Bazı araştırmacılar duygu tanıma problemine uygun mimariler geliştirmeye çalışırken, bazı araştırmacılar da mevcut mimarilerin duygu tanıma problemine uygun kombinasyonlarını geliştirmeye çalışmaktadır.

CNN ağırları güçlü sınıflandırıcılardır. Fakat, CNN ağlarının eğitilmesi oldukça zordur. Bir CNN ağının eğitilmesi büyük miktarda veri ve hesaplama gücü gerektirir. Eğer az sayıda veri ile model eğitilirse, CNN ağı veriyi ezberler ve veri kümesi dışındaki örneklerde kötü performans sergiler. Evrişim işlemleri çok sayıda hesaplama gerektirir ve hesaplamaların tamamlanması günlerce sürebilir. Paralel işlemler kullanılarak bu sürecin azaltılması gerekir. Fakat merkezi işlem birimi olan CPU'lar paralel işlemler için yeterince uygun değildirler. Paralel işlemlerde etkili olan pahalı grafik işlemcileri yani GPU'lar kullanmak gerekir.

Transfer öğrenme CNN ağlarının eğitilmesindeki güçlüklerle karşı ucuz ve etkili bir yöntem olarak ortaya çıkar. Transfer öğrenmede, başka bir problem için eğitilmiş bir CNN modeli kullanılır. Önceki problemin mevcut probleme benzer olması, transfer öğrenme uygulamasının başarısını artırır. Bir CNN modeli, bir çok katmandan oluşur. İlk katmanlar piksel özniteliklerini ayırıştırırken, sonlardaki katmanlar probleme özgü nesnelerle ilgili öznitelikleri ayırıştırır. İlk katmanlardaki bilgiler, bir çok problem için benzerdir. Transfer öğrenme, bir çok problem için kullanılabilen ilk katmanlardaki bilgilerden faydalanır. Sadece sonraki katmanlar yeniden eğitilerek, model eldeki probleme uygun hale getirilir.

Veri kümesi, makine öğrenmesinde en önemli ve gerekli kaynaktır. Bu çalışmada FER13 veri kümesi kullanılmıştır. Veri kümesi, 32 bin adet yüz görüntüsünden oluşur. Görüntülerin 48X48 piksel boyutunda olması içerdiği bilgi miktarının düşük olması anlamına gelir. FER13 veri kümesinin tek sorunu görüntü kalitesinin düşük olması değildir, ayrıca sınıflar arasında örnek sayısı bakımından dengesizlikler vardır. Veri kümesi insanlar tarafından etiketlendiği için etiketler gürültü içermektedir. Bütün bu sorunlara rağmen, veri kümesi içerdiği örnek sayısının diğer kümelerle kıyaslanınca fazla olması nedeniyle yaygın olarak kullanılmaktadır.

Bu çalışma, veri kümesinin küçük olması nedeniyle transfer öğrenmenin duygu tanıma problemine uygunluğunu araştırır. Çalışmada, öncelikle, transfer öğrenmeye uygun bir baz model bulmaya çalışır. Bu amaçla, farklı CNN mimarileri ve farklı veri kümeleriyle eğitilmiş CNN modelleri incelenmiştir. Imagenet veri kümesinden eğitilmiş VGG16, ResNet50 ve InceptionV3 modelleriyle, Oxford Face veri kümesi kullanılarak eğitilmiş VGGFace modelleri üzerinde deneyler yapılmıştır. VGG16 ve VGGFace modellerinin iyi sonuçlar verdiği görülmüştür. Kullanılan veri kümesinin, duygu tanıma için kullanılan veri kümesine benzerliğinden dolayı, VGGFace modelinin en iyi sonuç verdiği görülmüştür.

Daha sonra, bulunan baz modelin performansının iyileştirilmesi için topluluk öğrenmesi kullanılmıştır. Topluluk öğrenmesinde farklı özellikteki sınıflandırıcılar birlikte kullanılarak, daha güçlü bir sınıflandırıcının elde edilmesi amaçlanır. Bu amaçla uzman modeller eğitilmiştir. Her uzman model, bir duygunun sınıflandırılması için eğitilmiştir. Baz model topluluk öğrenmesi yöntemiyle, uzman modellerle birleştirilmiştir. Bu yöntem kullanılarak yapılan deneylerde, elde edilen geçerleme isabet oranları önceki deneylerdeki en iyi geçerleme isabet oranını geçememiştir. Fakat elde edilen geçerleme isabet oranlarının daha düşük varyans ve daha iyi ortalama değeri verdiği görülmüştür.

Deneyler sonucunda, transfer öğrenme ve topluluk öğrenmesinin birlikte kullanılmasının, güçlü bir sınıflandırıcının eğitilmesine yardım ettiği görülmüştür. Transfer öğrenmede kullanılan modelin, problem için uygun olması, model eğitme için kullanılan veri kümesinin eldeki veri kümesi ile benzer olmasının önemi görülmüştür.

Farklı özelliklerdeki sınıflandırıcıların birlikte kullanıldığı durumda, daha güçlü bir duygu tanıma sınıflandırıcısının elde edilebileceği görünmüştür. Deneyler sonucunda elde edilen en iyi geçerleme isabet oranı % 69.40 olarak ölçülmüştür.

1. INTRODUCTION

In this theses, research is conducted to build an efficient emotion classifier using static facial expressions. Due to their strength on computer vision tasks [1–3], Convolutional Neural Networks are used as the classifier. Due to limited data and power resources, transfer learning method is experimented. Later in the study, further experiments are executed on the obtained models. Experimental results show the success of applied methods.

In this chapter information about the emotion recognition followed by an introductory information about convolutional neural networks is given. Purpose of the thesis is briefly explained. Last section in this chapter is devoted to literature review.

In chapter 2, information about applied methods is given. Main methodology applied is transfer learning. How transfer learning works and how it is implemented is described. The chapter is conclude with details on the dataset used.

In chapter 3, the details on how experiments are executed is given.

In chapter 4, transfer learning experiments are done using several well known pre-trained CNN models. Experiment results are presented and interpreted.

In chapter 5, ensembling experiments are executed. Results are presented and interpreted.

In the last chapter, conclusions about experiments done are given. Where this work can be useful and how it can be improved is briefly described.

1.1 Emotion Recognition

Emotion recognition is one of the popular research topics in computer vision domain. Emotion recognition task involves assigning a label to a sample from a set of emotions. In this thesis we use the following set of emotions as in other studies. (e.g. [4–8]) A model trained for this purpose should be able to choose a single emotion which best describes the sample.

Emotion recognition is important because it is useful in any area where human and computer interacts which covers many areas. As a result, emotion recognition problem is studied by many researchers recently. There are two popular contests to boost research: Audio/Visual Emotion Challenge (AVEC) and Emotion Recognition in the Wild (EMOTIW).

1.2 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are formed using stacked layers. A CNN may contain convolutional layers, pooling layers and fully connected layers. Number, type and order of layers used in a CNN forms the architecture. Also, the number of filters used in convolutional layers are specific to the architecture. Each layer takes a 3D input volume and produces a 3D output volume. Initial input volume contains the raw image pixels. Last output volume contains the class scores.

Each convolutional layer consists of varying number of filters. A filter is a 3-dimensional vector of wights. Width and height dimensions of the filter (receptive field size) are designated and specific to the architecture. The depth of a filter is always equal to the depth of the input volume.

A filter is convolved with input volume regions starting from top left of the volume. After each convolution operation, filter frame or kernel is slid to new regions such that all input volume is convolved with the filter. In the end, a 2-dimensional activation map is created. All filters create their own activation maps. All activation maps are stacked to form a 3-dimensional output volume. As a result, depth of an output volume is determined by the number of filters used in the layer. Figure 1.1 illustrates the formation of an output volume. Note that input to a convolution using a single filter is a 3-dimensional volume while resulting output is a 2-dimensional area in the output volume.

Spatial size of the output volume (V) depends on following parameters:

- **Input Field Size (W):** Height and width dimensions of the input volume.
- **Filter Size (F):** Height and width dimensions of the filter or the kernel.
- **Stride (S):** Number of pixels to slide the filter or the kernel after each convolution.

Input Volume

Output Volume

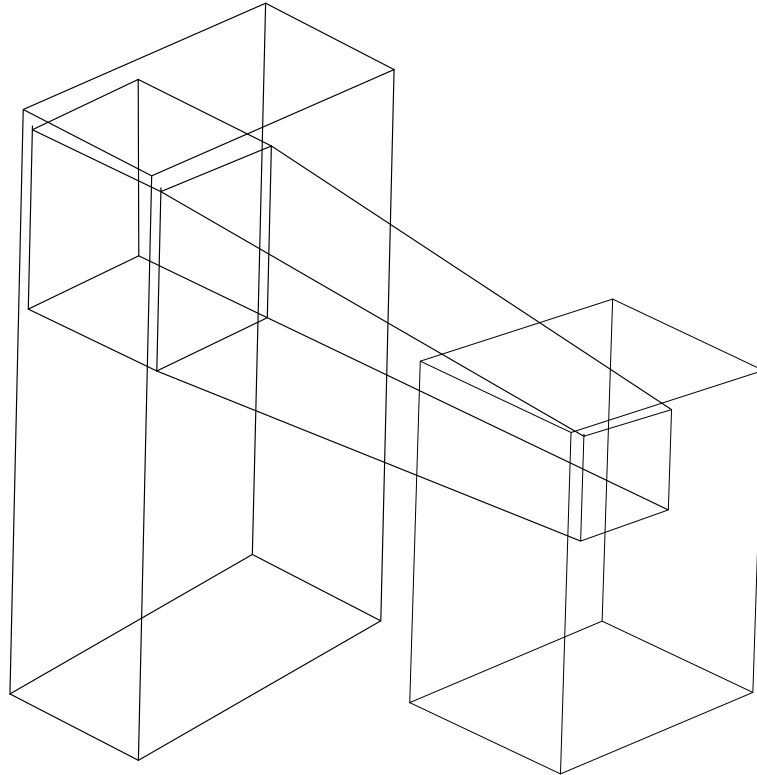


Figure 1.1 : Convolution Input and Output Volumes

- **Zero-Padding Amount (P):** Number of zero paddings to add around input field.

Spatial size of the output volume can be calculated using the following formula 1.1:

$$V = \frac{W + 2P - F}{S} + 1 \quad (1.1)$$

If an input image with 224X224 pixels is convolved with a kernel of size 3X3, stride 5 and zero padding 2, the output dimension will be:

$$\frac{224 + 2 \times 2 - 3}{5} + 1 = 46 \quad (1.2)$$

In the end, the output volume size will be 46X46. The depth of the volume will be determined by the number of filters used. Note that output volume dimensions are not related to input volume depth.

Convolutional layers are usually followed by a RELU layer which applies element-wise activation function. An example activation function can be **max(0,x)**.

3	1	7	1
3	5	9	8
4	7	2	5
7	5	5	8

5	9
7	8

Figure 1.2 : Max Pooling

While training CNN's, backpropagation algorithm is used to calculate gradients. The gradients are used to update network weights according to the error produced by the network. If the network is deep, as it is the case for most CNN's, gradients becomes extremely small for early layer such that early layer weights are not updated. This effectively stops learning for early layers. This phenomenon is known as Vanishing Gradient Problem. When the input is positive, derivative of RELU is 1 which removes the diminishing effect. Thus RELU activation function is resilient to Vanishing Gradient Problem, as opposed to the sigmoid activation function.

Pooling layers are used to downsample convolutional layer outputs. A pooling layer can apply either max pooling or average pooling. Figure 1.2 shows an example of max pooling using a 2X2 window.

Fully connected layers resemble multilayer perceptrons. In a fully connected layer, all nodes are connected to all other nodes in the previous layer. Final fully connected layer in every CNN classifier is output layer with the number of nodes equal to the number of classes present in the problem. In emotion recognition problem there are 7 output nodes because there are 7 emotions. Figure 1.3 shows an example fully connected layer with 2 outputs.

CNN's are known to be good at computer vision tasks. In recent years, use of CNN models in emotion recognition contests has become popular [4–8]. Some researchers are trying to come up with architectures which are suitable for emotion recognition problem [6, 7, 9]. Some researchers are trying to use combinations of existing architectures to make them suitable for the problem [4,5,8].

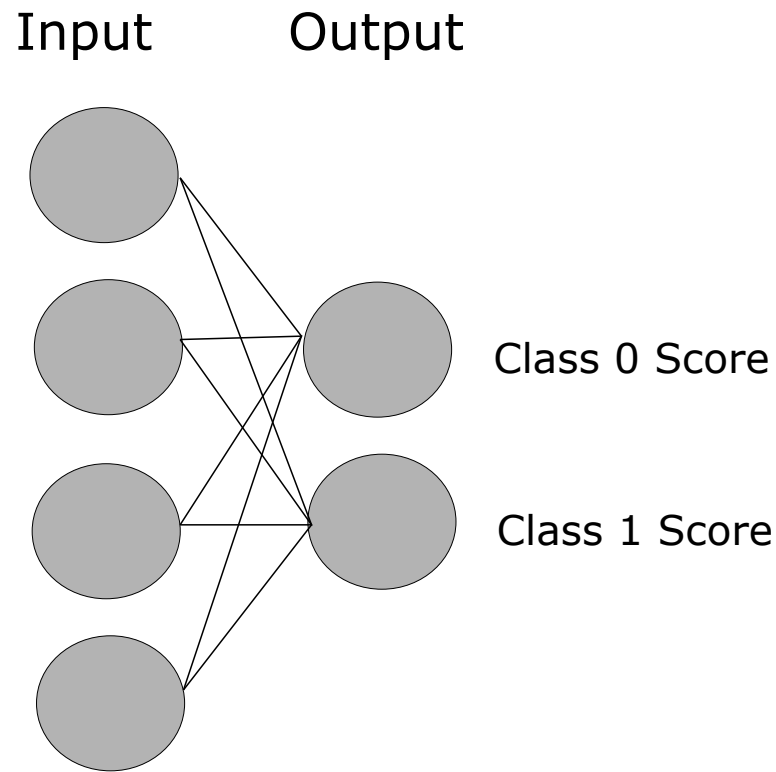


Figure 1.3 : Fully Connected Layer

CNN's are powerful classifiers; however, CNN's are difficult to train because training a CNN model requires a large amount of data [4] and computing power. In case of small amount of data, CNN's tend to overfit. Convolutions done on images require a lot of calculations which takes days of execution time. Parallel processing is needed to speed up the process. Parallel computation using CPUs is not enough and expensive GPU hardware is needed.

1.3 Purpose of Thesis

Purpose of this these is to do the required research to build an efficient emotion classifier using static facial expressions.

Emotion recognition is an important problem because it is useful in any case where computer and human interacts. Solving this problem makes it possible to create machines that can alter their behaviors according to human subject.

Another purpose of this theses is to show the effectiveness of transfer learning and ensembling on emotion recognition problem. Due to limited data and power resources, transfer learning method is experimented. Later in the study, obtained models are improved using ensembling method.

Additionally, obtained models can be used in future studies to create efficient emotion recognition systems. It is possible to use the models as a basis for video emotion recognition studies.

1.4 Literature Review

Emotion recognition problem is widely studied. There are community contests to boost the studies. One such contest is EmotiW contest. In EmotiW 2015, three winners of the contest relied on the CNN's to improve state of the art results.

Hong-Wei et al. [4] use a pre-trained model. AlexNet and VGG-CNN-M-2048 models are used for fine-tuning. Their focus was to increase the size of the training set for better generalization and avoiding overfitting. In addition to image dataset provided by contest organizers, they used FER2013 image dataset, which is publicly available, as additional training images. They reported 48.5 % accuracy for validation and 55.6 % accuracy for test on contest dataset.

Zhiding et al. [5] used CNN's with identical architectures but trained with differently seeded initial weights. For training, they used FER2013 dataset and then finetuned the CNN's with SFEW dataset which is provided by challenge organizers. Finally, they used different loss functions to find the best ensemble of CNN's at hand. Their best result is 55.96 % for validation and 61.29 % for test on contest dataset.

Bo-Kyeong et al. [6, 7] use a rule-based hierarchical ensemble of different CNN's which they call hierarchical committees. For different CNN's, they used differently seeded initial weights with identical architectures and completely different architectures. They also experimented using a different number of hierarchy levels. Their 3-level hierarchy yielded test accuracy of 61.6 % on contest dataset.

[10] used transfer learning and ensembling techniques. Transfer learning is applied using models which are trained using FER13 dataset. In ensembling part, the same CNN model is used to extract features. In the end, multiple logistic regression based classifiers are trained using the deep features which are produced by the same CNN network.

This study has similarities with [10] but methods are applied in a different way. This study tries to find most suitable models for transfer learning for emotion recognition.

Models which are trained on huge datasets like Imagenet are used for transfer learning. The amount of data used to train the model boosts feature quality. In ensembling part, restructured datasets are used to train expert classifiers for each class, again using transfer learning. Restructured databases help expert classifiers generate best features to discriminate related classes. While applying ensembling, instead of using the output of all expert models, a subset of models is selected by looking at the confusion matrix.

2. Methodology

The depth of a convolutional neural network has a direct impact on model performance [1]. As the number of layers increase, classification accuracy tends to increase.

Each layer extract different level of information. Filters on first convolutional layer of the network activate on simple structures like edges and colors. Filters on higher levels activate on more complex structures like circular shapes. As we go higher on layers, filters activate on more complex shapes like objects [11].

For example, if the model is trained for animal classification, top layers will activate on animal parts. In a broader sense, only top layers on a CNN network contain filters which are specific to problem at hand. This fact brings the idea to re-use early layer weights for new problems.

2.1 Transfer Learning

Early layers in a CNN network contains filters which activate on simple building blocks which are used to build complex objects. Those weights are similar for every object since they share the same building blocks. Thus it is a good idea to leverage early layer weights for new classification problems. This approach is called Transfer Learning.

In transfer learning, a previously trained model is used. To transfer the weights, the same architecture must be used in the new classifier. After weight transfer, it is necessary to use the problem-specific dataset to re-train the new classifier. After re-training, new classifier weights are adjusted to fit the new data.

During retraining, early convolutional layers are usually not touched. Only higher level layer weights are slightly adjusted. Learning the slight adjustments are possible with a relatively small amount of data. The small amount of data, in turn, makes it possible to train models in relatively small amount of time. As a result, cost of using convolutional neural networks for classification tasks reduces and applicability of CNN's increases.

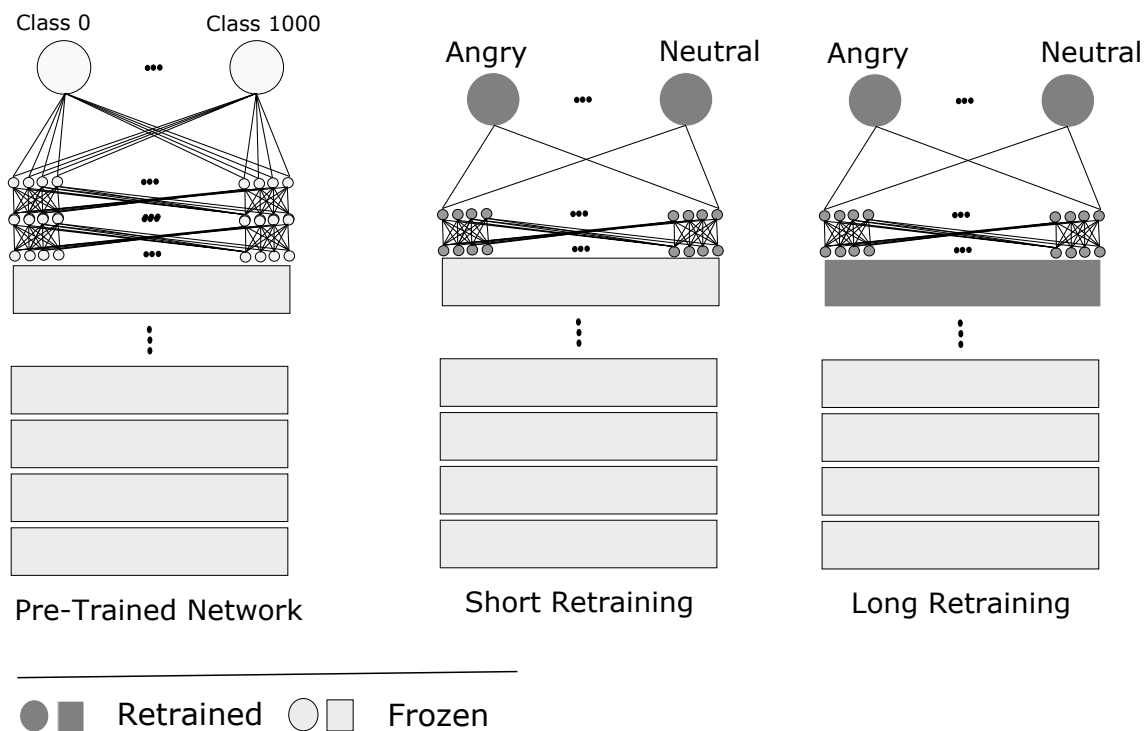


Figure 2.1 : Implementing Transfer Learning

2.2 Implementation of Transfer Learning on CNN's

In transfer learning, last fully connected layers and softmax layers of a model is removed and new layers are added. The number of nodes in softmax layer is determined according to output classes needed for a new problem. In emotion recognition problem, 7 softmax nodes are used. The new model is re-trained using new problem dataset.

Transfer learning technique is applied in two phases. Before actual training which we will refer as long re-training, a preliminary training which we will refer as short re-training is applied. Short re-training procedure is run to set initial weights to newly added layers. During short re-training, only fully connected layers and softmax layer is trained. Other layers are frozen. Later, in long re-training, last convolutional layers together with fully connected and softmax layers are trained. The number of convolutional layers to train depends on the size of the dataset and should be determined empirically. Basically, as the size of the dataset grows, the number of convolutional layers to should be increased. Figure 2.1 illustrates this process.

2.3 Models for Transfer Learning

There are popular CNN models which are trained using huge datasets. VGGNet [1], ResNet50 [3] and Inception [2] networks are trained on Imagenet [12] dataset. Imagenet dataset consists of over ten million images which belong to 1000 different classes. Imagenet Large Scale Visual Recognition Competition (ILSVRC) is organized annually.

VGGNet has the second place in ILSVRC 2014 classification challenge. It contains 16-19 layers with small convolutional filters (3x3). Inception network is the winner of ILSVRC 2014 classification challenge. It introduced the idea of modular CNN architecture (Inception module) which is not a sequential stack of layers. ResNet50 network is the winner of ILSVRC 2015. It is a 152 layers network which introduced the idea of residual blocks in which the input is added to the activation function.

VGGFace [8] model is basically a VGGNet network with 16 layers. It is trained using VGG Face dataset [8] which contains 2.6 Million facial images.

2.4 Dataset

Some datasets are generated in lab environments with perfect lighting and background conditions. Subjects are asked to act the expressions. Subjects exaggerate while trying to mimic emotion expressions. Because finding subjects is not very easy, it is difficult to create huge datasets. Models trained using such data perform poorly on real-life images.

Fer2013 [13] dataset is not created in lab conditions. The dataset contains 32K gray-scale images. Samples are collected over the internet using related keywords. Images are processed and image labels are validated by human labelers.

Image dimensions are 48X48. Low resolution images contain less information about facial expressions. Low dimension could be good in terms of ease of calculation. However, models used in transfer learning requires image dimensions to be 224X224.

Reliability of crowdsourcing is always under question. [14, 15]. According to [15] correctness of labels in FER13 dataset is around 65 % plus minus 5%. Table 2.1 can give an idea on this. The fear sample looks like an angry sample. The sad sample looks like a neutral sample. The disgust sample looks like an angry sample. It may even be regarded as a surprise sample.

Despite the non-perfect quality, the dataset is widely used in emotion recognition studies [4–8] because it contains a considerable number of images. Samples are not structured. Lighting and background conditions are close to the real life. The dataset contains natural expressions from many subjects. Such attributes important for creating models which generalize well to out-of-dataset images.

Before training, samples are not preprocessed because they are gray-scale facial images. Since faces are mostly centered similarly, no normalization is needed. Some non-facial samples are removed from the dataset after a manual observation. A total of 72 samples are removed which can be neglected compared to the size of the dataset.

Table 2.1 : Samples from Dataset







Angry	Fear	Disgust	Happy	Sad	Surprise
					

Table 2.2 : Number of Instances Per Class

	ANG.	DISG.	FEAR	NEUT.	HAPPY	SAD	SURP.
Training	3979	435	4090	7198	4951	4827	3163
Validation	467	56	496	895	607	653	415

Table 2.1 shows sample images from FER13 dataset. Table 2.2 shows the number of instances per class. A quick glance reveals that disgust class is under represented while happy class is over represented.

3. Experimental Setup

Selection of learning rate is quite important to avoid local minima when training a neural network. For full training, generally, 0.1 is used. However, for fine-tuning much smaller learning rate is preferred. Similar to [4] learning rate is selected as 0.001. Experiments with different learning rates showed that this is the optimal value. Momentum value of 0.9 is used to speed up learning. To learn smaller weights 0.0005 is used as weight decay. Stochastic gradient descent is used as the optimizer.

Dropout is one useful technique to avoid overfitting the data [16]. Dropout with probability p means in each pass a node is put off, not updated, with probability p . This way data is not memorized by nodes and network becomes more robust to overfitting. In our work drop out with probability 0.5 is used at last softmax layer.

Perturbation or image augmentation is a method which applies linear transformations to images to create additional training instances. Perturbation makes the network more robust to overfitting. Applied transformations are random combinations of rotation, vertical flipping and zooming.

For preprocessing, all images are resized to 224x224 pixels. Augmented images are not saved. Perturbations are applied during training procedure in CPU parallel with multiple threads. Pixel intensity values are rescaled to fit range $[1,0]$.

All training images do not fit into CPU memory. Even if they fit CPU memory, they certainly won't fit GPU memory. Thus, images are read from disk, augmented and fed to the network for backpropagation in small batches. In our experiments, random batches of size 64 are used.

Training convolutional networks involves great deal of random processes. Running the same experiment consecutively may result in slightly different models. Those models will yield different accuracy values on the same data. In order to evaluate the method success multiple experiment executions are needed.

3.0.1 Experiment Evaluation

To choose, the most suitable base model for transfer learning, experiments are executed once because there are considerable amount of difference between accuracy scores. Experiments afterwards are executed multiple times and their statistics are evaluated. In order to compare two methods, methods are executed multiple times. Statistics of each method is calculated. While comparing accuracy means, variances, minimum and maximum accuracy values are used. Statistical significance test is conducted to check if there is a statistically significant difference between accuracy results. Two tailed T-Test is used as statistical significance test.

4. Transfer Learning

Table 4.1 : FER13 Validation Results After Retraining

Model	Accuracy	Precision	Recall
Inception	0.5668	0.7248	0.7223
ResNet50	0.6120	0.7654	0.7533
VGG16	0.6464	0.8138	0.7586
VGGFace	0.6779	0.8089	0.8072

Figure 4.2 shows training and validation accuracy chart of VGGFace model retraining procedure. Dashed red line shows the boundary between short and long retrainings. Since this is a pre-trained network accuracy increases rapidly. Figure 4.3 shows confusion matrix for the model.

At the beginning, training and validation accuracies are very close which is expected. As the training proceeds training and validation accuracies increase together. At some point, the rates training and validation accuracies increase start to differ and the gap between training and validation accuracies starts to expand until the network weights converge. Once the network convergences, the gap remains the same. At that point, the training is stopped using an early stopping criterion.

Sudden drop effect on the boundary is a result of combining two separate training phases in one chart. When long retraining phase starts, accuracy drops due to weight updates on convolutional layers. Curve on the left of the dashed line represents the short retraining procedure. Curve on the right side of the dashed line represents the long retraining procedure. As one may observe, short retraining period starts to converge just before the boundary line. When long retraining starts, slope of the line increases and model converges at higher accuracies. This is the result of taking convolutional layers into training.

Confusion matrix summarizes the weaknesses and strengths of the trained classifier. Low accuracy of disgust class can be attributed to the low number of disgust samples found in the dataset. Despite the high frequency of samples, Fear and Sad classes also show low accuracy. Confusion matrix tells us that they are confused with each other.

VGGFace for Emotion Classification
Input(224X224X3)
Conv 64 (3X3)
Conv 64 (3X3)
Max Pooling (2X2)
Conv 128 (3X3)
Conv 128 (3X3)
Max Pooling (2X2)
Conv 256 (3X3)
Conv 256 (3X3)
Conv 256 (3X3)
Max Pooling (2X2)
Conv 512 (3X3)
Conv 512 (3X3)
Conv 512 (3X3)
Max Pooling (2X2)
Conv 512 (3X3)
Conv 512 (3X3)
Conv 512 (3X3)
Max Pooling (2X2)
FC6-1024
FC7-1024
FC8-7

Figure 4.1 : VGGNet Architecture for Emotion Classification.

Table 4.2 : Number of Convolutional Layers Experiment

# of Convolutional Layers Trained	Validation Accuracy
2	0.6712
5	0.6779
8	0.6676

This is an inherent problem with emotion recognition. Micro-expressions for different emotions may show similarities which makes it difficult to classify emotions just using still images.

Another fact which limits performance is the amount of noise present in the dataset. Quality of the dataset labels is discussed in section 2.4. According to [15] correctness of labels in FER13 dataset is around 65 % plus minus 5%. Dealing with noise present in the dataset is not in the scope of this thesis.

The number of convolutional layers to train during long retraining is important. It depends on the size of the dataset at hand. If there are plenty of data, more convolutional layers can be trained. If there is a tiny amount of data, it may be better not to train convolutional layers at all. Number of convolutional layers to train is among the hyperparameters and should be determined empirically.

Table 4.2 shows experiment results when searching for the number of layers to train. The best result is obtained when 5 convolutional layers are trained. Training only two convolutional layers is too few compared to the amount of data present. We can benefit more from the data by training more layers. On the other hand, experiment results show that we do not have enough data to train eight convolutional layers.

4.0.1 Transfer Learning Results

VGG16, InceptionV3, ResNet50 and VGGFace models are used as the base models. Validation results after retraining with FER13 dataset is shown in Table 4.1.

VGG16, InceptionV3 and ResNet50 models are trained using Imagenet dataset. InceptionV3 and ResNet50 models performed poorly compared to VGG16 model. InceptionV3 and ResNet50 models may be too complex for emotion recognition problem considering the amount of data available for training.

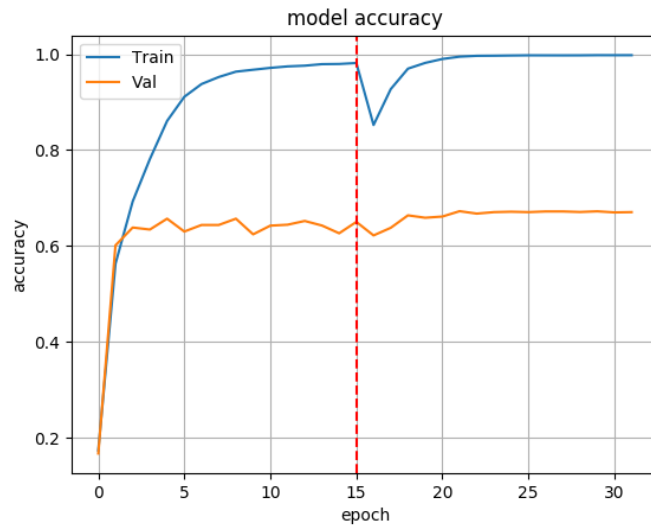


Figure 4.2 : VGGFace Training&Validation Accuracy Chart on FER13 Dataset.
(Steeper line represents the training)

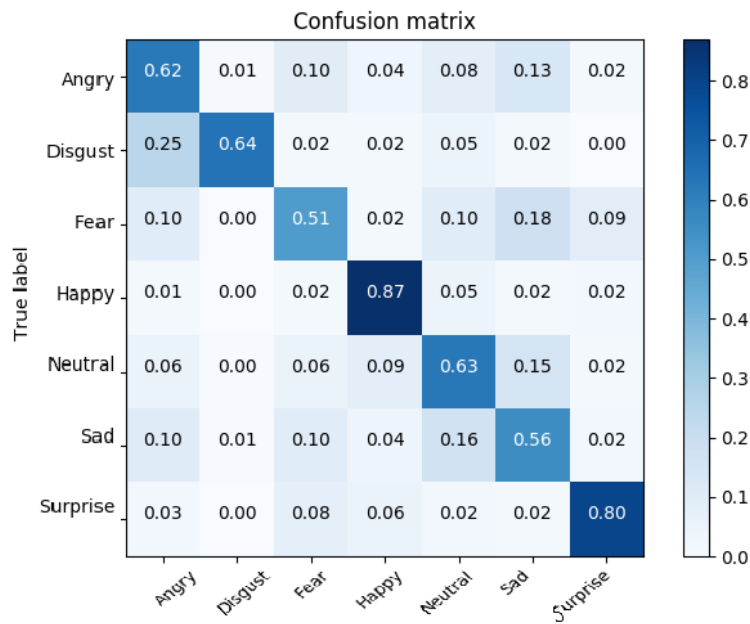


Figure 4.3 : VGGFace Confusion Matrix of Validation Data.

Best accuracies are obtained using VGG16 and VGGFace models. VGGFace and VGG16 have the same architecture thus close results are not surprising. Since VGGFace model is trained on face dataset it appears to be more suitable for emotion recognition problem. Later experiments are conducted using VGGFace model. Structure of the VGGFace model we trained is shown in Figure 4.1.

Results shown in Figure 4.1 for VGGFace belongs to the best model obtained during training. After five experiments accuracy is described with mean 0.6711 and standard deviation 0.005257 with the minimum being 0.6662 and maximum being 0.6779. Reader may check figure 5.3 to see how results are scattered. Note that even the minimum accuracy is better than the accuracy of VGG16, which is the second best model.

4.0.2 Decreasing Model Complexity

VGGFace model is trained using Oxford Face dataset. Its layer weights are adjusted for face images which is very important for Emotion Recognition problem. It is possible to decrease model complexity without removing any layers from the model.

After the last max pooling layer, a Global Average Pooling layer is added to VGGFace model (Check 4.1). A Global Average Pooling layer takes the average of each channel in the previous layer. If input vector is $N \times N \times C$ dimensional, the output will be a C dimensional vector.

Results are improved using this approach. Accuracy after 10 experiments is described with mean 0.6883 and standard deviation 0.003213. Highest accuracy is 0.6949 and lowest accuracy is 0.6835. (See figure 5.3 and Table 5.1)

T-test can be used to investigate statistical significance of the difference. Calculated t-value is -7.94697 and calculated p-value is smaller than 0.01 which means that differences are significant.

In rest of the document, we will refer to VGGFace model with global average pooling as the Simplified VGGFace model. Rest of the experiments unless specified otherwise are conducted using the simplified VGGFace model.

5. Ensembling

Stacking is an ensembling method which combines multiple simple classifiers with a meta-classifier to create a more powerful classifier. It is tempting to use classifiers which are created using transfer learning in the previous section.

The simplified VGGFace classifier is combined with VGG16 and ResNet50 based classifiers. As meta-classifier, a multi-layer perceptron (MLP) with two layers and softmax activation is trained. The first hidden layer contains 1024, the second hidden layer contains 512 nodes.

5 experiments are conducted to obtain reliable results. Calculated accuracy mean is 0.6636 and standard deviation is 0.002007. The worst accuracy is 0.6606 while the best accuracy is 0.6662. Even the best accuracy is worse than the accuracy of the Simplified VGGFace model. See Table 5.1.

This result can be attributed to similar capabilities of the classifiers used. When confusion matrices of the 3 models are inspected it can be seen that even though they show different accuracies they have similar strength and weaknesses. For example, all classifiers perform poorly for sad emotion. All classifiers perform well on happy emotion. All classifiers confuse fear with sad. See Appendices A.1, A.2 and A.3 to investigate corresponding confusing matrices. Thus for an input image, all classifiers votes similarly. Classifiers with different capabilities are needed to improve the results.

5.0.1 Stacking with Expert Classifiers

In our experiments, different models are finetuned: VGG16, VGGFace, ResNet50 and Inception. However, their confusion matrices show similar results. Thus, they are not very suitable for ensembling.

In order to create models with different capabilities, multiple datasets with two classes are created using FER13 dataset. For each dataset, one of 7 classes is selected as the base class. In the new dataset, samples belonging to the base class are labeled as

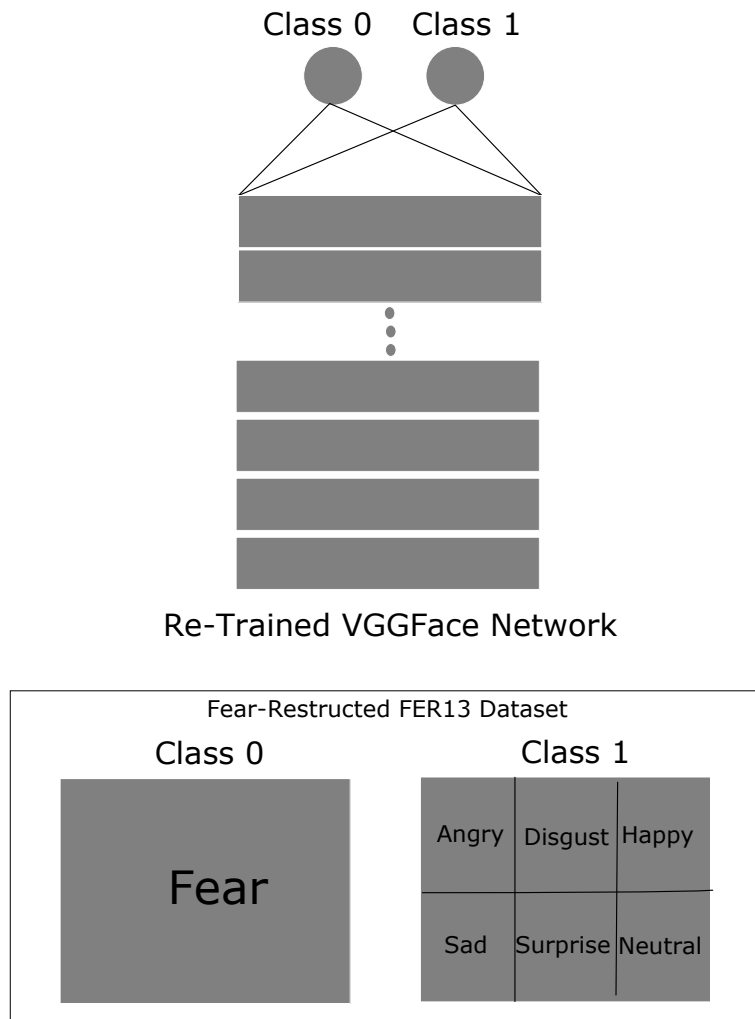


Figure 5.1 : Fear Expert Model Training Using Fear-Restructured FER13 Dataset

first class. Samples belonging to rest of the classes are labeled as second class. Such datasets were created for all 7 emotion classes.

Using each new dataset, a new pre-trained VGGFace model is retrained. Each resulting classifier is expert for classifying the base class of the dataset it is trained. For example, if the fear class is taken as the base class, in the new dataset, all fear samples are labeled as first class. Samples belonging to rest of classes are labeled as second class. Resulting classifier will be expert at classifying fear class. The trained model will be able to find patterns that best differentiates fear class from other classes. Figure 5.1 illustrates how the dataset is structured and new binary model looks like. Class 0 output of the model represents the class FEAR, Class 1 represents the non-FEAR class.

One caveat here is, if all remaining samples are used in the second class new dataset will be very unbalanced. Thus, a random set of samples are drawn from rest of the

classes such that the total number of samples belonging to the first class is equal to second class.

Expert models can be ensembled with other models to increase accuracy. The accuracy of our VGGFace model can be improved this way. A quick observation at confusion matrix shows that accuracy of fear and sad is low and those classes are confused with each other (Figure 4.3). A successful ensembling of VGGFace model with fear-expert VGGFace model and sad-expert VGGFace model should result in better performance.

For feature extraction, empiric results show that layer FC6 is the best option among FC6, FC7 and FC8 layers (See figure 4.1).

A multi-layer perceptron (MLP) with two layers and softmax activation is trained. The first hidden layer contains 1024, the second hidden layer contains 512 nodes. The MLP has 1024 inputs from the base model, 1024 inputs from the fear-expert model and 1024 inputs from the sad-expert model. In total, the MLP has 3072 inputs. Figure 5.2 shows the ensembled model architecture.

10 experiments are executed for reliable results. Accuracy mean is 0.6896 and standard deviation is 0.0006893. The worst accuracy is 0.6885 and the best accuracy is 0.6905. At first glance mean and standard deviation is improved while the best accuracy result still belongs to the simplified VGGFace model. Check Figure 5.3 and Table 5.1.

Merged model and VGGFace model results are compared using t-test. t-value is calculated as 1.289 and p-value is 0.214 which means that differences are not statistically significant.

Figure 5.3 shows accuracy results of experiments. First column shows accuracy values from finetuned VGGFace model. Second column shows accuracy values from finetuned VGGFace model with global average pooling. Third column shows accuracy scores from experts stacking experiment. In Appendix A.4 there is a box plot for Simplified VGGFace validation accuracies versus Stacked Expert models validation accuracies.

Even though accuracy results are not improved, ensembling with expert models produces more reliable classifiers without compromising from performance.

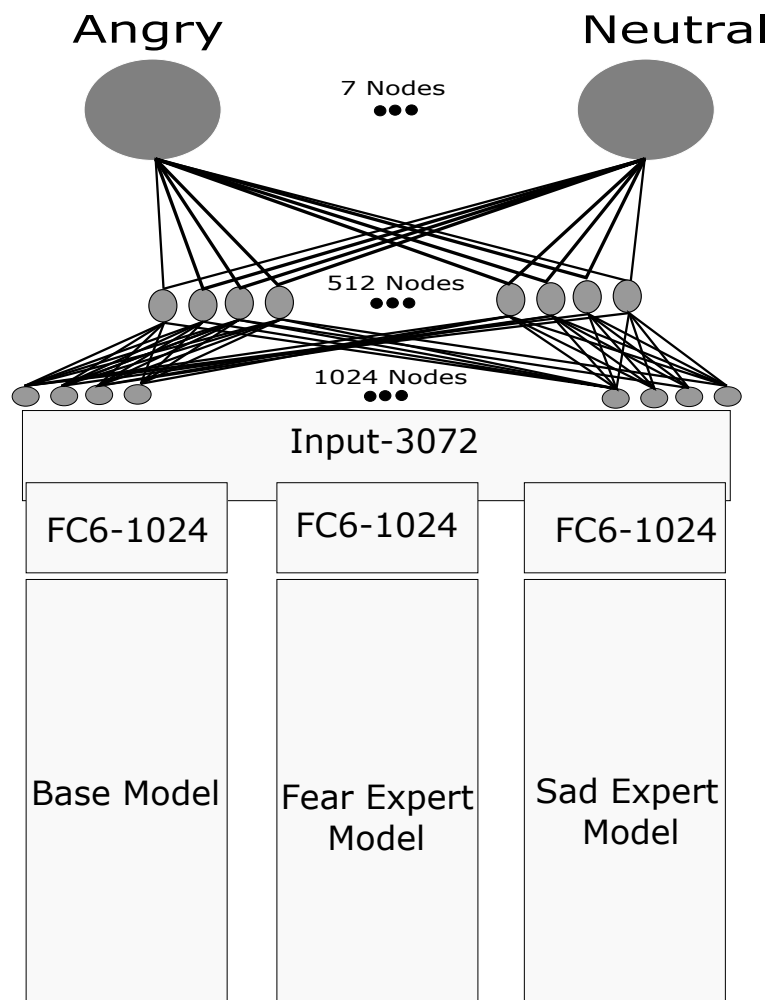


Figure 5.2 : Ensembling Diagram

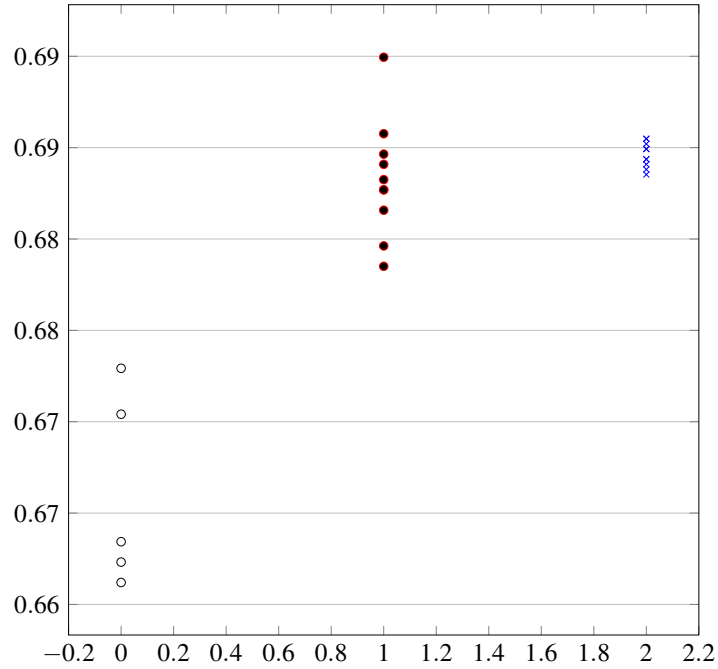


Figure 5.3 : Accuracy Scatter Plot of 3 Different Experiments. Circle: VGGFace, Dot: Simplified VGGFace, x: Ensembled
(Horizontal axis is for separating marks, does not have any meaning.)

Table 5.1 : Best FER13 Validation Results of Experimented Models

Model	Accuracy	Precision	Recall
VGGFace	0.6779	0.8089	0.8072
Simplified VGGFace	0.6949	0.8253	0.8148
Stacked Base Models	0.6662	0.8061	0.7932
Stacked Experts	0.6905	0.8206	0.8133

6. CONCLUSIONS AND RECOMMENDATIONS

Convolutional neural networks are difficult to train from scratch. They require a lot of data and computational resources. Transfer learning is a cheap and efficient way of making use of pre-trained CNN classifiers.

For transfer learning, using the right model suitable to the problem at hand is crucial. For example, in emotion recognition, a model like VGGFace which is pre-trained using a face dataset should be chosen. The number of layers to train is an important hyper parameter and should be chosen empirically according to the amount of data present for training. For transfer learning, choice of learning rate is also important. Choosing 0.001 as learning rate should suffice. Too complex models may not be very suitable for transfer learning if amount of data is limited.

Multiple CNN classifiers with different capabilities can be used to create a more powerful classifier. Dataset set can be structured into multiple binary datasets to create binary classifiers. Those binary classifiers are expert for classification of single emotion. They can be ensembled together to obtain a better classifier. The resulting classifier may not produce the best accuracy but the classifier may perform better in terms of variance.

Choice of the dataset for the problem is very important. All classes in the dataset should be represented well otherwise classification performance is affected negatively. Crowd sourcing enables access to large datasets in cost of noisy labels.

This study can be improved by incorporating more architectures into model selection phase. Another improvement path could involve elimination of noise present in the dataset. More datasets can be used to improve generalization capability of the trained models. Use of more datasets bring the complexity of different face alignments which should be unified using appropriate face alignment techniques.

Results of this study can be used in further studies. Created models can be used in video classification tasks. The base model can be used as a feature extractor to feed video classification models.

REFERENCES

- [1] **Simonyan, K. and Zisserman, A.** (2014). Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556*.
- [2] **Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. et al.** (2015). Going deeper with convolutions, *Cvpr*.
- [3] **He, K., Zhang, X., Ren, S. and Sun, J.** (2016). Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.770–778.
- [4] **Ng, H.W., Nguyen, V.D., Vonikakis, V. and Winkler, S.** (2015). Deep learning for emotion recognition on small datasets using transfer learning, *Proceedings of the 2015 ACM on international conference on multimodal interaction*, ACM, pp.443–449.
- [5] **Yu, Z. and Zhang, C.** (2015). Image based static facial expression recognition with multiple deep network learning, *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ACM, pp.435–442.
- [6] **Kim, B.K., Roh, J., Dong, S.Y. and Lee, S.Y.** (2016). Hierarchical committee of deep convolutional neural networks for robust facial expression recognition, *Journal on Multimodal User Interfaces*, 10(2), 173–189.
- [7] **Kim, B.K., Lee, H., Roh, J. and Lee, S.Y.** (2015). Hierarchical committee of deep cnns with exponentially-weighted decision fusion for static facial expression recognition, *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ACM, pp.427–434.
- [8] **Parkhi, O.M., Vedaldi, A., Zisserman, A. et al.** (2015). Deep Face Recognition., *BMVC*, volume 1, p. 6.
- [9] **Yao, A., Cai, D., Hu, P., Wang, S., Sha, L. and Chen, Y.** (2016). HoloNet: Towards Robust Emotion Recognition in the Wild, *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ICMI 2016, ACM, New York, NY, USA, pp.472–478, <http://doi.acm.org/10.1145/2993148.2997639>.
- [10] **Savoiu, A. and Wong, J.** Recognizing Facial Expressions Using Deep Learning.
- [11] **Zeiler, M.D. and Fergus, R.** (2014). Visualizing and understanding convolutional networks, *European conference on computer vision*, Springer.

- [12] **Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L.** (2009). Imagenet: A large-scale hierarchical image database, *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, pp.248–255.
- [13] **Carrier, P.L. and Courville, A.**, (2013), FER-2013 face database, Technical report, 1365, Université de Montréal, 2013.
- [14] **Barsoum, E., Zhang, C., Ferrer, C.C. and Zhang, Z.** (2016). Training deep networks for facial expression recognition with crowd-sourced label distribution, *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, ACM, pp.279–283.
- [15] **Goodfellow, I.J., Erhan, D., Carrier, P.L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.H. et al.** (2013). Challenges in representation learning: A report on three machine learning contests, *International Conference on Neural Information Processing*, Springer, pp.117–124.
- [16] **Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R.** (2014). Dropout: A simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, 15(1), 1929–1958.

APPENDICES

APPENDIX A.1 : VGG16 Confusion Matrix

APPENDIX A.2 : Simplified VGGFace Confusion Matrix

APPENDIX A.3 : ResNet50 Confusion Matrix

APPENDIX A.4 : Box Plot Simplified VGGFace Experiment vs Stacked Experts
Experiment Accuracies

APPENDIX A.1

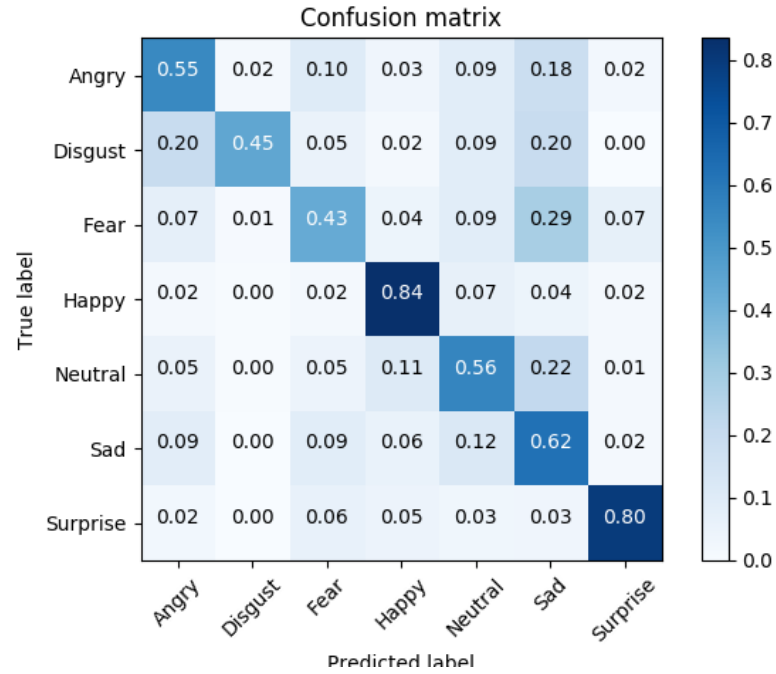


Figure A.1 : VGG16 Confusion Matrix

APPENDIX A.2

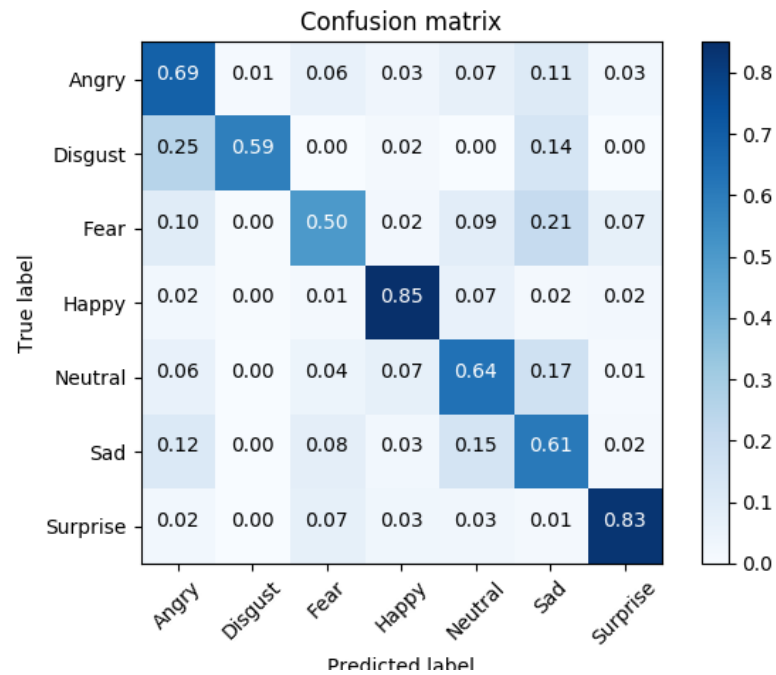


Figure A.2 : Simplified VGGFace Confusion Matrix

APPENDIX A.3

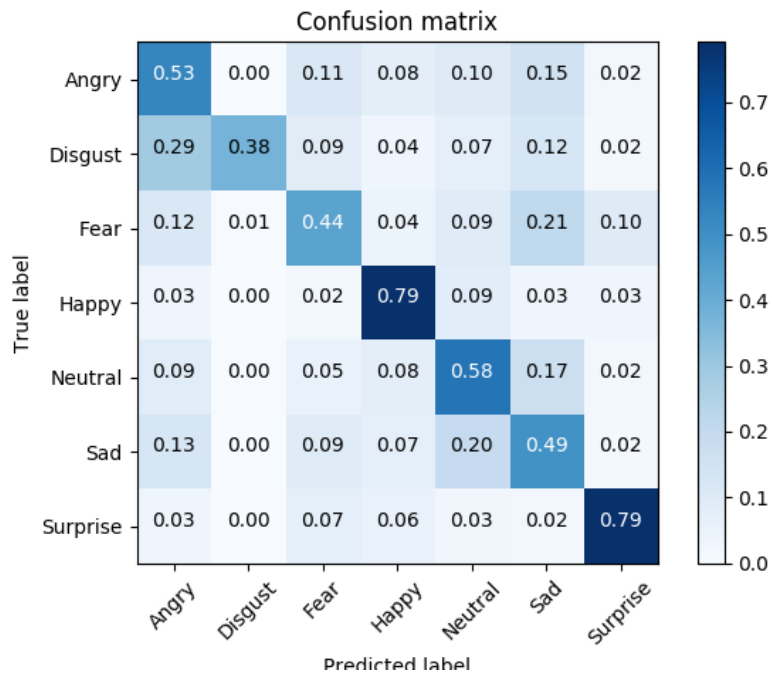
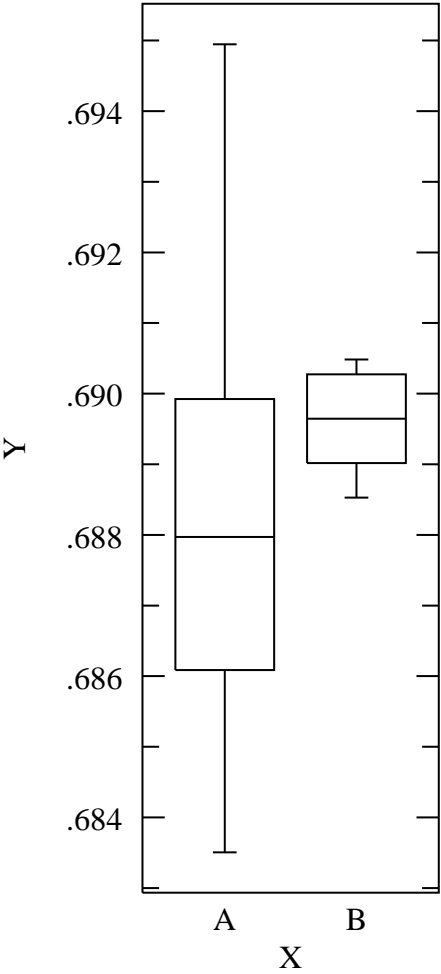


Figure A.3 : ResNet50 Confusion Matrix

APPENDIX A.4

1.0.1 Box Plot of Simplified VGGFace Experiment vs Stacked Experts Experiment Accuracies

0F49



CURRICULUM VITAE



Name Surname: Hüseyin ABANOZ

Place and Date of Birth: Samsun 1987

E-Mail: abanozh@itu.edu.tr

EDUCATION:

- **B.Sc.:** 2010, Marmara University, Engineering Faculty, Computer Science and Engineering Department

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2010-2013 Alcatel-Lucent, Software Engineer
- 2013-2016 Alcatel-Lucent, Senior Software Engineer
- 2016-2018 Nokia, Senior Software Engineer

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- Abanoz H., Çataltepe Z. (2018) TRANSFER ÖĞRENME VE TOPLULUK ÖĞRENMESİ KULLANARAK DURAĞAN GÖRÜNTÜLER ÜZERİNDE DUYGU TANIMA, 26. *IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU-2018)*