ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

FACIAL EXPRESSION PAIR MATCHING

M.Sc. THESIS

Deniz ENGİN

Department of Electronics and Communication Engineering

Electronics Engineering Programme

OCTOBER 2017

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

FACIAL EXPRESSION PAIR MATCHING

M.Sc. THESIS

Deniz ENGİN
(504141205)

Department of Electronics and Communication Engineering

Electronics Engineering Programme

Thesis Advisor: Assoc. Prof. Dr. Sıddıka Berna Örs YALÇIN

OCTOBER 2017

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

YÜZ İFADESİ ÇİFTİ EŞLEŞTİRME

YÜKSEK LİSANS TEZİ

Deniz ENGİN
(504141205)

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Programı

Tez Danışmanı: Assoc. Prof. Dr. Sıddıka Berna Örs YALÇIN

EKİM 2017

Deniz ENGİN, a M.Sc. student of ITU Graduate School of Science Engineering and Technology 504141205 successfully defended the thesis entitled “FACIAL EXPRESSION PAIR MATCHING”, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Assoc. Prof. Dr. Sıddıka Berna Örs YALÇIN** ....
Istanbul Technical University

Jury Members : **Assoc. Prof. Dr. Hazım Kemal EKENEL** ....
Istanbul Technical University

Prof. Dr. Çiğdem Eroğlu ERDEM ....
Marmara University

Date of Submission : **8 September 2017**
Date of Defense : **6 October 2017**

To my family,

ACKNOWLEDGMENTS

I would like to thank my advisor Assoc. Prof. Dr. Sıddıka Berna Örs YALÇIN and my co-advisor Assoc. Prof. Dr. Hazım Kemal Ekenel for their guidance, advice, and helps.

I am also grateful to the members of SiMiT Lab for their support. Very special thanks to my family, my friends, and especially Sema Saraç, for their unconditional support during the preparation of this thesis.

In addition, this work was funded by TUBITAK with project no. 113E067 and Marie Curie FP7 Integration Grant in the 7th EU Framework.

October 2017

Deniz ENGİN
Electronics Engineer

TABLE OF CONTENTS

ABBREVIATIONS

AU	**:** Action Unit
CNN	**:** Convolutional Neural Network
DNN	**:** Deep Neural Network
FACS	**:** Facial Action Coding System
FER	**:** Facial Expression Recognition
HOG	**:** Histogram Of Gradients
LBP	**:** Local Binary Pattern
LFW	**:** Labeled Faces in the Wild
LOSO	**:** Leave One Subject Out
PCA	**:** Principal Component Analysis
ReLU	**:** Rectified Linear Unit
SIFT	**:** Scale Invariant Feature Transform
SURF	**:** Speeded Up Robust Features
SVM	**:** Support Vector Machines

LIST OF TABLES

LIST OF FIGURES

Page

FACIAL EXPRESSION PAIR MATCHING

SUMMARY

Facial expression plays an important role in the social communication of people. Therefore, Facial Expression Recognition (FER) methods which are used for Human-Machine Interaction and Human-Robot Interaction have been developed based on computer vision and machine learning techniques. Both basic emotions or Facial Action Coding System (FACS) can be used for analysis of facial expressions. FACS based on Action Unit (AU) defined as a movement of the human face can consist of one or more facial muscle which can be observed alone or in groups. Combination of action units refers to different emotions.

Human feelings are categorized into six basic emotions: happy, angry, disgust, fear, sadness, surprise. For facial expression recognition, there are two types of datasets which are controlled and uncontrolled (wild). Uncontrolled datasets are collected from everyday life images that include facial expressions, while controlled datasets are created from people acting out a facial expression or capturing their spontaneous facial expressions in controlled environments. In an uncontrolled environment, various problems are encountered such as illumination, viewpoint variations, and occlusion. Due to these problems, facial expression recognition becomes a difficult task. The annotation of facial expression also requires excessive work even though images can be easily downloaded from the web for the creation of significant amount of uncontrolled dataset.Creating controlled datasets is challenging, hence, these datasets often contain fewer images, conversely.In order to address these problems: (i) augmentation for small datasets, (ii) annotation for large datasets, we introduce a new pair matching problem for facial expression recognition. In this study, we are able to determine whether two facial expressions are the same or different without knowing any of the facial expressions. Thus, large datasets can be annotated by using the proposed method. In addition, by training a model from the pairs of a small dataset, and utilizing pair classification results, facial expressions can be recognized.

Automatic face analysis systems generally contain the following steps: face detection and alignment, feature extraction, and classification or similarity estimation. Feature extraction methods can be divided into two groups: hand-crafted features such as Histogram of Oriented Gradients, Gabor Filter, Scale-Invariant Feature Transform, Local Binary Pattern and learned features from data by using deep neural networks. Support Vector Machines is commonly used for classification, as well as deep neural networks.

In this thesis, The Extended Cohn-Kanade (CK+) and the JAFFE datasets are used for pair matching experiments. After the alignment process, pairs are created from these datasets and Local Binary Pattern for feature extraction and Support Vector Machine classification is used for the experiments. The Labeled Faces in the Wild (LFW) dataset

which includes a small number of pairs for face recognition, aims to use more data for the training and testing processes. Due to the small number of pairs on the CK+ dataset, the LFW format is used to split to the matched and mismatched pairs for the CK+ dataset for the first pair classification experiments, and we achieve high performance. The experimental results show that our pair matching formulation increased the facial expression recognition. In addition, deep learning methods conducted on the SFEW and the FER-2013 datasets are used the facial expression classification in the wild. Since VGG Face convolutional neural network model is trained on face images, it is chosen to fine-tune on facial expression dataset to understand facial expressions.

YÜZ İFADESİ ÇİFTİ EŞLEŞTİRME

ÖZET

Yüz ifadeleri insanların birbirleri ile kurdukları sosyal iletişimde önemli bir rol üstlenmektedir. Bu nedenle, insan makine etkileşimi ve insan robot etkileşimi alanlarında kullanılmak amacıyla, bilgisayarla görü ve makine öğrenmesi algoritmalarına dayalı yüz ifadesi tanıma yöntemleri uzun yıllardır çalışılmakta ve her geçen gün gelişmektedir. Yüz ifadeleri analiz edilirken, temel duygu tanımlarının yanı sıra, yüz hareketi kodlama sistemi de (Facial Action Coding System - FACS) kullanılmaktadır. Temel duygular mutluluk, üzüntü, korku, iğrenme, şaşırma ve kızgınlık olmak üzere altı temel ifade olarak tanımlanmıştır. Bu duyguların yanında nötr imgeler de kullanılmakta olup, bazı veri kümelerine bu temel ifadelere ek olarak küçümseme, çığlık gibi başka ifadeler de eklenmiştir. Yüz hareketi kodlama sistemi ise yüz hareketlerinin kişinin yüzündeki olağan değişimleri olarak adlandırılan eylem birimine dayanmaktadır. Eylem birimleri bir ya da birkaç yüz kasından oluşacak biçimde tanımlanmış olup, tek başına ve ya grup şeklinde gözlemlenebilir. Her yüz ifadesi için gözlenmesi ve/veya gözlenmemesi gereken eylem birimleri tanımlıdır. Kişilerin yüz ifadesinde bulunan eylem birimleri tespit edildikten sonra yüz ifadesine karar verilmektedir.

Veri kümeleri de imgelerin veya videoların etiketlenme türüne göre, yüz ifadesi veya eylem birimi etiketi olmak üzere iki grupta incelenebilir. Bunun yanı sıra videolardan oluşan dinamik ve resimlerden oluşan statik veri kümeleri bulunmaktadır. Videolardan oluşan yüz ifadesi veri kümelerinde video çerçeveleri “offset”, “onset” and “peak” olarak etkilenebilmektedir. “Peak frame” yüz ifadesinin belirgin olduğu çerçeveyi temsil etmektedir. “Offset frame” nötr olarak tanımlanırken, “onset frame” ifadenin başladığı çerçeveyi temsil etmektedir.

Yüz ifadesi veri kümeleri, kontrollü ve kontrolsüz olmak üzere iki grupta toplanabilir. İnsanların yüz ifadelerinin taklidiyle oluşan ya da kendiliğinden anlık olarak oluşan yüz ifadelerini içeren veri kümeleri kontrollü veri kümesi olarak adlandırılır. Kontrolsüz veri kümeleri ise internetten yüz ifadesi içeren imgelerin indirilmesiyle oluşturulmaktadır. Işık veya aydınlatma değişimleri, poz açılarının değişmesi, yüz ifadelerinin başka nesneler tarafından kapanması gibi problemler kontrolsüz veri kümelerinde yüz ifadesi tanınmasını zorlaştırmaktadır. Kontrolsüz veri kümeleri büyük ölçekli veri kümeleridir, bu nedenle bunların etiketlenmesi önemli bir problem oluşturmaktadır. Veri kümelerinin etiketlemesi alanında uzman kişiler tarafından yapılması gerekmektedir. Küçük ve orta ölçekli veri kümelerinin etiketlenmesi birkaç ay sürerken, büyük ölçekli veri kümeleri için çok uzun zaman gerekmektedir.

Bu tezde, ifade etiketi olmayan iki yüz ifadesi incelenerek, yüz ifadelerinin aynı ya da farklı olduğunu belirleme problemi tanımlanmıştır. Kişi tanıma ve yüz ifadesi analizi hakkında yapılan çalışmalar temel alınarak bir yöntem önerilmiştir. Uzmanlık

gerektiren yüz ifadesi etiketleme işleminin yapılabilmesi için etiketi olmayan iki yüz ifadesinin, aynı olup olmadığını belirleyebilmek bu problem tanımının yapılmasının temel amacıdır. Önerilen yöntemler ile oluşturulan eş olan ve eş olmayan çiftler sayesinde az imge içeren veri kümeleri için veri miktarının arttırılmış olur. Böylece yüz ifadesi tanıma doğruluğunu da arttırılır.

Otomatik yüz tanıma ve analizi sistemleri genel olarak yüz sezimi, öznitelik çıkarımı ve sınıflandırma aşamalarından oluşmaktadır. Seçilen veri kümesi için öncelikle yüz sezimi yapılmakta ve gerekli ise yüz hizalama işlemi de yapılabilmektedir. Ön işlemeden geçirilen yüz imgelerinden öznitelik çıkarımı yapıldıktan sonra ise istenilen duruma göre benzerlik tahmini veya sınıflandırma işlemi yapılmaktadır. Öznitelik çıkarma yöntemleri, temel elle öznitelik çıkarma yöntemi (hand-crafted) ve derin öğrenme metotları aracılıyla veriden öğrenilen öznitelik çıkarma yöntemi olmak üzere iki grupta toplanabilir. Bunun yanında temel öznitelikler geometrik, görünüş ve hareket olmak üzere üç ana başlıkta incelenebilir. Nirengi noktaları (Facial Landmarks) geometrik özniteliklere örnek olarak verilebilir. "Gabor Filtre", "Histogram of Oriented Gradients (HOG)", "Scale-Invariant Feature Transform (SIFT)", " Local Binary Pattern (LBP)", ve "Speeded-Up Robust Features (SURF)" ise geometrik tabanlı özniteliklere örnek olarak verilirken, "Optical flow" ise hareket bazlı öznitelik olarak tanımlanabilir ve dinamik veri kümelerinden öznitelik çıkarımı için kullanılır.

Yüz ifadesi tanıma için önerilen çeşitli kıyaslama veri kümelerinin yanında, yüz ifadesi tanıma yarışmaları için de veri kümeleri oluşturulmaktadır. Bu veri kümeleri kontrollü ve kontrolsüz olabilmektedirler. Extended Cohn-Kanade (CK+), JAFFE, MultiPIE, MMI, FERA, AFEW ve SFEW veri kümeleri yüz ifade analizinde yaygın olarak kullanılan yüz ifadesi veri kümleridir. LFW veri kümesi ise yüz doğrulama için oluşturulan ve yüz imge çiftlerinden oluşan bir veri kümesidir. Veri kümesi iki alt küme şeklinde yayınlanmıştır. İlk alt küme parametre ve algoritma seçimi için kullanılırken, ikinci alt küme, 10 alt kümeye ayrılarak çapraz doğrulama yöntemi ile performans raporlama için kullanılmaktadır. Yüz ifadesi çifteleri oluşturularak, bu çalışma için veri kümesi hazır hale getirilmiştir. Hazırlanan bu veri kümesi az sayıda imge çifti içerdiği için veri kümesini eğitim ve test olarak ayırmak yerine, verinin maksimum şekilde kullanımı amaçlanan LFW veri kümesi formatı temel alınarak, yüz ifadesi çiftleri bu formata uygun şekilde alt kümelere ayrılmıştır. Bu format CK+ veri kümesi için yapılan ilk deneylerde kullanılmıştır. Diğer deneyler için bir öznenin dışarıda bırakılması şeklinde eğitim ve test setleri oluşturulmuştur. Bu durumda her kişinin test edilmesi için oluşturulan modelin kişinin resimlerini içermeyen imge çiftleri ile eğitilmesi gerekmektedir.

Yüz ifade çifti problem tanımının deneyleri için yüz ifadelerini sınıflandırmada yaygın olarak kullanılan Extended Cohn-Kanade (CK+) veri kümesi ve JAFFE veri kümesi kullanılmıştır. İki veri kümesi de az veri içeren, kontrollü veri kümesidir. CK+ veri kümesi eylem birimi ve yüz ifadesi etiketlerini içeren video ve imgelerden oluşmaktadır. 327 imge yüz ifadesi etiketine sahiptir. Mutluluk, kızgınlık, küçümseme, iğrenme, şaşırma, üzüntü, korku olmak üzere altı temel duyguya ek olarak küçümseme yüz ifadesi de veri kümesinde yer almaktadır. CK+ veri kümesinde yüz ifadesi etiketi bulunan 118 kişi vardır. JAFFE veri kümesi ise 10 kişinin 213 imgesinden oluşmaktadır. İmge sayılarının yüz ifadesi dağılımı orantılıdır. Altı temel yüz ifadesine ek olarak, nötr yüz ifadesi de bu veri kümesinde yer almaktadır.

Yüz imgeleri hizalama işlemi nirengi noktalarına göre yapılmıştır. Her göz için altı nirengi noktası bulunmaktadır. Bu noktalar kullanılarak gözlerin merkez noktaları bulunmuş ve bu noktalara göre yüz hizalama ve kesme işlemi yapılarak ve ön işleme adımı tamamlanmıştır. Her yüz ifadesi için eşleşen bütün yüz ifade çiftleri oluşturulmuştur. Eş olmayan imge çiftleri ise eş olan çiftlerle aynı sayıda ve yaklaşık üç katı sayıda olacak şekilde seçilmiştir. Eş olmayan imge çiftleri seçilirken yüz ifadesi dağılımına dikkat edilmiştir. CK+ veri kümesinde, yüz ifadeleri için imge sayıları birbirinden çok farklı sayıda olduğu için bazı eş olmayan çiftlerin dağılımı orantılı olmamıştır. Küçümseme ifadesi az imge içerdiğinden, eş olmayan imge çiftlerinde küçümseme ifadesi de az bulunmaktadır. JAFFE veri kümesinde ise bütün duygular yaklaşık eşit sayıda olduğu için eş olmayan çiftlerin dağılımı daha orantılı olmuştur ve bu durum önerilen yöntemin başarısını daha çok arttırmıştır. Öznitelik çıkarımı için yüz tanıma problemi için önerilen yerel ikili örüntü (YİÖ) yöntemi seçilmiştir. Sınıflandırma için yaygın olarak kullanılan destek vektör makinesi (DVM) ise ikili sınıflandırma yapması için kullanılmıştır. İlk olarak LFW veri kümesi formatında, CK+ veri kümesi çiftleri oluşturularak çiftlerin aynı ya da farklı olduğuna karar verilen deneyler yapılmıştır. Çalışmanın diğer kısımlarında, çift oluşturmanın yüz ifadesi tanıma doğruluğunu artırdığını göstermek için yapılan deneyler kişiden bağımsız olarak yapılmış olup, bir kişinin imgeleri veya çiftleri ayrılarak eğitim verileri oluşturulmuştur. CK+ veri kümesinde 118 kişi olduğu için, her kişi ayrı bir eğitim setine sahiptir ve 118 model eğitilip her imge için ayrı test setleri oluşturulup test edilmiştir. CK+ veri kümesinde eş olmayan çift sayısı yaklaşık 3 kat arttırıldıktan sonra ise farklı eğitim veri kümesi hazırlanmıştır. Verinin fazla olması nedeniyle, kişiler 5 ayrı gruba ayrılarak, 5 ayrı eğitim kümesi oluşturulmuştur. Bu durumda 5 ayrı model eğitilmiş olup, yine bütün imgeler ayrı test edilmiştir. Çift sayısı arttırıldığında, öznitelik vektörlerinin boyutunu azaltmak için ana bileşenler analizi yöntemi ile öznitelik vektör boyutu indirgenmiştir. JAFFE veri kümesinde 10 kişi olduğu için deneyler kişiden bağımsız olarak 10 model eğitilmiş ve her imge ayrı test edilmiştir. Önerilen yöntemin sonuçları imge çiftleri kullanarak yüz ifadesi tanımanın doğruluk oranının arttığını göstermektedir.

Kontrollü veri kümeleri için yapılan deneylerin yanı sıra kontrolsüz veri kümeleri için de deneyler yapılmıştır. SFEW ve FER-2013 veri kümeleri derin öğrenme yöntemleri ile sınıflandırılmıştır. Hazır olan modeller üzerinden ince ayar yapılarak modellerin yüz ifadelerini üzerinde özelleşmesi sağlandıktan sonra test edilmiştir. Eğitim ve test veri kümeleri için veri kümelerinde belirtilen eğitim, doğrulama ve test kümeleri kullanılmıştır.

1. INTRODUCTION

The human feelings are important for communications. Accordingly, the human face has an important source for non-verbal communications. Facial expression recognition by using the human face is one the essential analysis in order to understand the human feeling and to be able to help them. Therefore, most of the researchers have been working on this topic in the field of the computer vision applications such as Human-Machine Interaction (HMI), Human-Robot Interaction (HRI). Many automatic facial expression recognition systems have been developed in recent years.

The fundamental approaches of facial expression recognition are mainly based on two expression definitions which are emotions and/or Facial Action Coding System - FACS [1]. Facial Motion Coding System (FACS) [1] based on Action Unit (AU) which defines as changes in the human face. Action Units can consist of one or more facial muscle, which can be observed alone or in a group. The intensity of AUs is assigned a range of A to E. A means trace, while E means maximum. Combination of AUs represents a facial expression. In addition to FACS, basic emotions were also defined as sadness, surprise, happy, angry, disgust, and fear [2]. In addition to basic emotions, some datasets have different emotions such as contempt, scream, and neutral.

According to collecting environments, facial expression datasets can be categorized into two groups: constrained and unconstrained datasets. Constrained datasets have been collected under controlled conditions such as light, background. Unconstrained datasets contain collected images from the web, which are belong to everyday life in variation such as pose, illumination, age, gender, race, ethnicity, and camera quality [3]. Constrained datasets are also called as lab dataset, whereas unconstrained dataset defined as the wild dataset. Due to the difficulties of collecting and labeling data from specific persons, there are a few images in the constrained datasets. Although images of facial expression can be easily obtained from the web, the labeling process of the images of unconstrained datasets is required lots of work [4].

Datasets can be divided into two categories: dynamic (sequence-based, video-based) or static (image-based) for lab and wild datasets. Sequences can be defined as an offset, onset, or peak in the sequence-based datasets. Image-based dataset generally includes peak frames which have the evident facial expression in expressions datasets. Also, some datasets include sequences and images, for instance, The Extended Cohn-Kanade dataset [5] is chosen for this study, has sequences and images which are labeled according to facial expressions and AUs.

FER systems are generally evaluated in subject-independent manner and cross-dataset approach. In order to split dataset for train and test, k-fold cross-validation and leave-one-subject-out (LOSO) approaches are applied in subject independent manner. In LOSO approach, one subject is chosen as the test set while other subjects are used for the train set. In addition, k folds are distinguished different subject for each fold. For cross-dataset experiments, one dataset or more than one dataset can be used for training and the different dataset is used for testing. By reason of the fact that different datasets have the various condition such as illumination and light, the cross-dataset task is not easier than the subject-independent task. In addition, performance on constrained datasets has better results than performance on unconstrained datasets because of quality of images and illumination, obstacle, etc.

Facial expression analysis generally employs three steps which are face alignment, feature extraction, and classification, sequentially. If there are different viewpoints in the dataset, face alignment can be required to obtain frontal faces and increase the performance of FER systems. Feature extraction methods are based on geometric, appearance and motion features. Facial landmarks are geometric features, while Gabor Filter [6], Histogram of Oriented Gradients (HOG) [7], Scale-Invariant Feature Transform (SIFT) [8], Local Binary Pattern(LBP) [9], and Speeded-Up Robust Features (SURF) [10] are hand-crafted and appearance based features. Optical flow [11] is an example of motion features. In addition, features can be learned from data by using a neural network such as deep learning methods. Traditional machine learning algorithms such as support vector machine which is widely used and deep neural networks can employ for classification problems.

In this thesis, we introduce a pair matching formulation for facial expression recognition and propose a solution to address this problem. According to the formulation, two unlabeled facial expression images should be defined as same or different. Figure 1.1 demonstrates the pipeline of the proposed method.

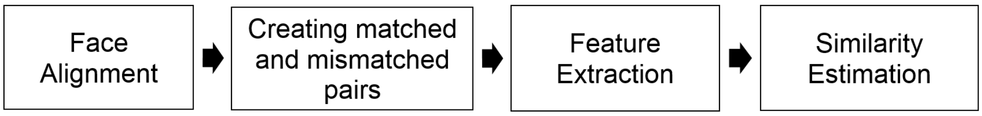


Figure 1.1 : The pipeline of the proposed method.

Facial expression images are aligned, firstly. Matched and mismatched pairs are created by considering the number of images for each facial expression. Features are extracted from each image by using the local binary pattern, and differences of features for each pair are calculated by using subtraction. These differences are labeled as zero for the mismatched pairs and labeled as one for the matched pairs, in order to apply the binary classification by using support vector machine (SVM). Facial expressions are predicted by utilizing the pair matching results. Firstly, we use high dimensional feature vector to train SVM model. After increasing number of training examples, we apply principal component analysis (PCA) for dimension reduction.

In addition to pair matching formulation, several experiments for facial expression recognition in the wild are performed without using pair matching. Since we have more training data in the wild datasets, convolutional neural network (CNN) which is defined as the end-to-end network is chosen. Feature extraction and classification is employed by using pre-trained VGG Face model [12]. This model is trained on face images, in this case, it is already learned face features. Thus, this pre-trained model is chosen.

The Extended Cohn-Kanade Dataset (CK+) [5] and The Japanese Female Facial Expression (JAFFE) Dataset [13] which are commonly used for facial expression recognition are chosen for this study. Since both of them have few images and contain posed expression in the controlled environment, the proposed method is performed on them. On the other hand, the Static Facial Expressions in the Wild (SFEW) [14], the Facial Expression Recognition 2013 [15] datasets are used for facial expression recognition in the wild.

1.1 Purpose of Thesis

The first main purpose of this thesis is to define facial expression matching problem and develop a novel pair matching system. The second aim is to be able to recognize facial expressions in the wild. The advantages of face recognition in pair matching problem definition:

- In order to be able to decide whether facial expression same or different for unlabeled face expression for two person
- The amount of data increased for facial expression recognition problems for small datasets
- The proposed methods for pair matching the field of face recognition for many years of studies can be used

1.2 Related Work

In this subsection section, related works of facial expression recognition in the constrained and unconstrained datasets which are the CK+, the JAFFE, the SFEW, the FER-2013 datasets are discussed. Results of the previous works are given in the Chapter 6.

Facial expression recognition (FER) has been studied for many years in various applications. As stated before, FER approaches can be performed on sequence-based or image-based datasets in the constrained and unconstrained environments. Facial expression recognition can be applied by using action units or facial expression labels.

In the earlier studies, most common methods for FER system utilize the hand-crafted features. In [16], 3D SIFT [17], HOE [18], LBP-TOP [19], HOG 3D [20] methods were applied and STM, STM_ExpLet methods have been proposed.

AU-aware Deep Networks (AUDN) with three module has been proposed in [21]. The first module includes two layers, convolutional layer, and max-pooling, the second module is called AU-aware receptive fields layer investigates over-complete representation to replicating the combinations of AUs. In the last module, Restricted Boltzmann Machine (RBM) is used to extract hierarchical features for linear SVM classification. Hand-crafted features and deep features are utilized and compared on the CK+ dataset for defined seven expressions and neutral, SFEW and MMI datasets. In addition to AUDN, hand-crafted features such as LBP, SIFT, HOG, Gabor are also used for comparison, these features have not performed as well as AUDN network.

In [22], facial expression recognition method based on Deep Neural Networks (DNNs) has been proposed, subject-independent and cross-dataset experiments have been performed on MultiPIE, MMI, DISFA, FERA, SFEW, CK+, and FER-2013 datasets. The proposed architecture consists of two convolutional with max pooling layer and four inception layers inspired by GoogLeNet [23].

The Boosted Deep Belief Network (BDBN) has been presented with three important contributions in [24]. The first contribution is that feature learning, feature selection, and classifier took part in the one network which is a loop process. Secondly, image patches are used as an input, and lastly, the boosting technique, and multiple DBNs are combined with proposed objective function. The proposed framework evaluated on the CK+ and the JAFFE datasets.

Three methods have been presented such as 2D Inception-ResNet, 3D Inception-ResNet, and 3D Inception-ResNet with facial landmarks based on 3D-CNNs and Long Short-Term Memory (LSTM) for FER videos in [25]. Facial landmarks have been used as an input and they improved the accuracy. Temporal relations are also used for image sequences. The proposed methods have been evaluated on the MMI, FERA and DISFA datasets in subject-independent manner and cross-database tasks.

In [26], DNN based two-part neural network has been presented for facial expression recognition from videos. To identify temporal relations in sequential frames, Conditional Random Field (CRF) was used for one part of the network. The other part of the network was inspired from Inception-ResNet [27]. Experiments are conducted on the CK+, MMI and FERA datasets in a subject-independent manner, moreover, cross-dataset experiments were performed.

Identities defined as subject attributes can be learned while learning the procedure for facial expression recognition. Two different facial expressions which belong to the same person is similar than two images which are same facial expression belong two different people. Therefore, facial expression recognition is challenge task when compared with the face recognition problems. There are a few studies to eliminate identity information for facial expression recognition systems. Identity-aware Convolutional Neural Network (CNN) with the contrastive loss to diminish inter-person by learning identity information has been proposed [28]. The proposed network has two CNN with shared parameters.

Liu et al. [29] have suggested that adaptive deep metric learning methods to close features space each other for two images which have the same expression from different individuals and detract from features of same person images for different expressions. Deep metric loss and softmax loss are combined in two fully-connected layers with optimized collectively. These proposed methods have been performed on the CK+, the MMI, and the SFEW datasets.

Dahmane et al. [30] have been proposed the prototype-based model for all expressions. Reference prototype of facial expression is produced by using a SIFT-flow registration. Oriented gradients are applied to all images in order to analysis appearance of facial expressions. Using SIFT-flow registration with HOG feature descriptors and SVM for classification have been evaluated on the JAFFE dataset [30].

Two hybrid system has been suggested for FER systems by Buciu et al. [31]. Independent component analysis (ICA) was combined with cosine similarity measure, maximum correlation classifiers, and SVM for the first system. A set of Gabor Wavelets (GWs) has been applied on the original images, and the same classifiers with the first system have been used for classification in the second hybrid system. The proposed methods have been evaluated on the JAFFE dataset.

A CNN architecture has been proposed the Linear support vector machines is used instead of softmax function at the end of the network in [32]. This study is the winner of the ICML Challenge [33] on the FER-2013 dataset.

Dhall et al. [14] have been introduced the new wild dataset which is called SFEW. This dataset has been obtained from AFEW dataset which includes videos from movies. Features are extracted by using LPQ and PHOG, SVM with RBF kernel is used to classify seven facial expressions in the proposed baseline method [14].

In [34], combination of local appearance features and global geometry information is used to get an intermediate face representation in the proposed model. The main of the proposed model based on hierarchical Bayesian is to recognition in the multipose dataset. The experiments were conducted on the SFEW datasets in [34].

2. AN OVERVIEW OF CONVOLUTIONAL NEURAL NETWORKS

Convolutional Networks was proposed by Yann Lecun in 1989 [35]. First CNN network which is known as LeNet proposed by Lecun et al. [36]. Convolutional Neural Network (CNN) is a specialized kind of neural network that performs convolution operation, therefore, named as "convolutional neural network" [37]. In recent years, CNNs have become the more successful variety of visual tasks such as localization, object detection, classification.

CNN basically includes neurons that receive some inputs, and conducts dot product and produce the output which is the input for next layers. CNNs contain several layers depends on the architecture, basically, these layers perform convolution, feature representation, and classification operations. Also, each convolutional neural network has loss function on the last layer. Early CNN layers learn more generic features, while later layers are specialized on datasets, and learn more task-specific features.

In this chapter, components of CNN architectures and well-known architectures are defined. Moreover, Transfer Learning method is briefly explained.

2.1 Components of CNN Architecture

Most important components of convolutional neural networks which are layers, activation functions, regularization method are summarized in this subsection.

Convolutional Layer

Convolutional neural networks mainly based on convolution operation, thus this layer is the core building block for CNNs. The convolutional layer parameter includes learnable filters. Convolution operation is given in Equation 2.1, where I is the two-dimensional image as an input, K is the two-dimensional kernel [37].

$$S(i,j) = (I * K)(i,j) = \sum_{m}\sum_{n} I(m,n)K(i-m,j-n) \tag{2.1}$$

Some neural network libraries implement the convolution operation without flipping the kernel that is known as cross-correlation but called as convolution. Equation 2.1 defines cross-correlation.

$$S(i,j) = (I*K)(i,j) = \sum_m \sum_n I(i+m, j+n)K(m,j) \tag{2.2}$$

The flipping kernel is necessary for the commutative property. In machine learning, convolution is used with other function. When combined convolution with other functions, we obtained functions which do not have commute property. Thus, it does not matter that convolutional operation flips its kernel or not [37].

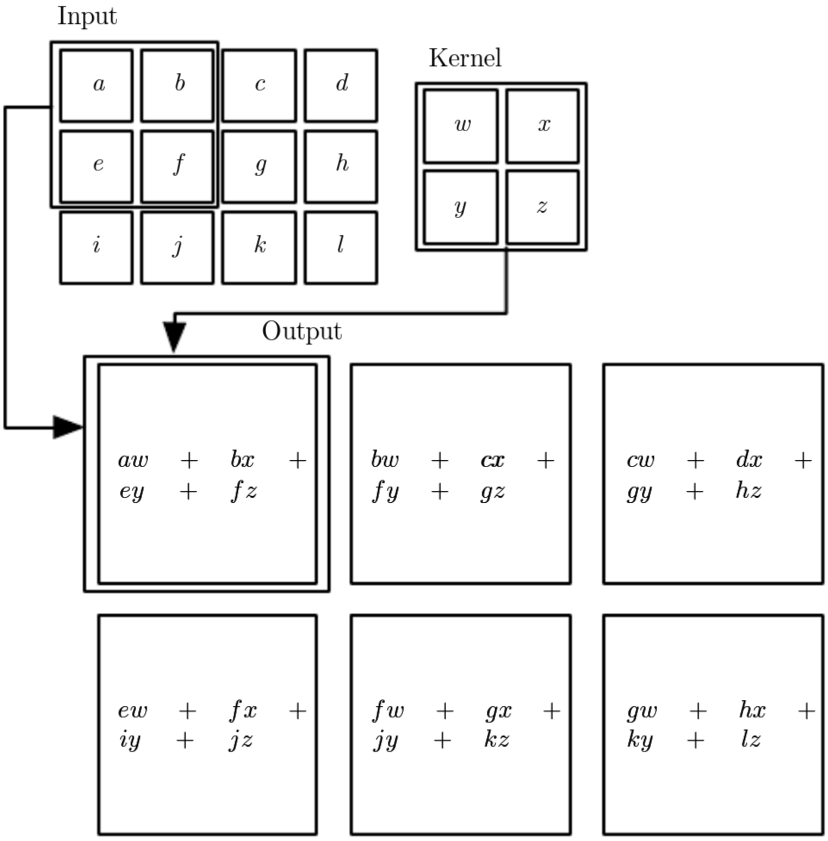


Figure 2.1 : An example of 2-D convolution operation [37].

Figure 2.1 illustrates that 2-D valid convolution without kernel flipping performed to a 2-D sensor [37]. In this figure, kernel size is given as 2x2, the pointed position of the kernel on input produces an output value calculated as described in the first box. This kernel lies in the entire image, each shift of kernel produce an output which can be shown in below part of the figure.

Pooling Layer

This layer reduces the dimension from one layer to next layer. It can be also called as subsampling layer. The number of parameters and complexity are reduced by using pooling. Thus, overfitting is prevented by losing information and this helps to generalize network.

There are several approaches for pooling operations such as max pooling, average pooling, L2-norm pooling, and weighted average. In generally, pooling layer takes place in between convolutional layers.

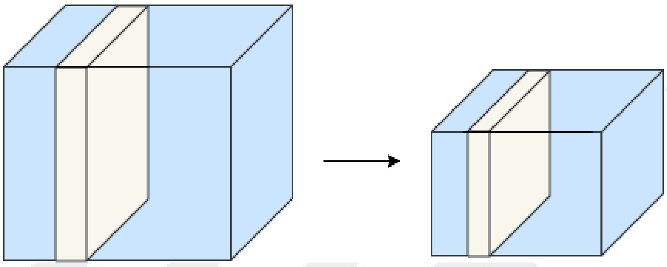


Figure 2.2 : Max pooling operation.

Max pooling [38] is more common pooling approaches, it replaced the NxN sub-area to their max value. Figure 2.2 shows max pooling operations, for instance, max pooling operation by using the 2x2 filter with stride 2 reduces dimension to half, for instance, WXH dimension reduces to (W/2)x(H/2).

Activation Functions

Every activation function takes an input as a single number and conducts defined the mathematical operation on the input. Activation functions are generally implemented after one pair of convolution and pooling layer. There are various activation functions such as tanh, Sigmoid, the Rectified Linear Unit (ReLU), Leaky ReLU, PReLU, ELU, and Maxout. In the previous studies, sigmoid and tanh functions were used, while ReLU activation function has become popular in recent years.

The sigmoid function takes a real number and produces an output in the range of 0 and 1. If the input value is the large negative number, the output is calculated as 0, while if the input value is the large positive number, the output is calculated as 1. On the other hand, the output of tanh activation function is in the range of [-1,1]. Sigmoid kills gradients, both sigmoid and tanh activations saturate.

Some activation functions are defined as follows:

Hyperbolic tangent function:

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.3}$$

Sigmoid function:

$$sigmoid(x) = \frac{1}{1+e^{-x}} \tag{2.4}$$

Rectified Linear Unit function:

$$ReLU(x) = max(0,x) \tag{2.5}$$

Leaky Rectified Linear Unit function:

$$LeakReLU(x) = max(0,1x,x) \tag{2.6}$$

ReLU activation function is very computationally efficient and it converges much faster than Sigmoid and tanh functions in empirically. ReLU activation function basically performs thresholding at zero and provides a non-linear operation. ReLU units can be die during training, thus, Leaky ReLU has been proposed to fix this problem.

Dropout

Deep neural networks have lots of parameters. While lots of parameters are beneficial, they cause to overfitting problem. In order to prevent overfitting, dropout has been proposed as a regularization method [39]. Basically, unit(s) is dropped with its connection randomly during training neural networks. Dropout also increases the performance on supervised learning [39].

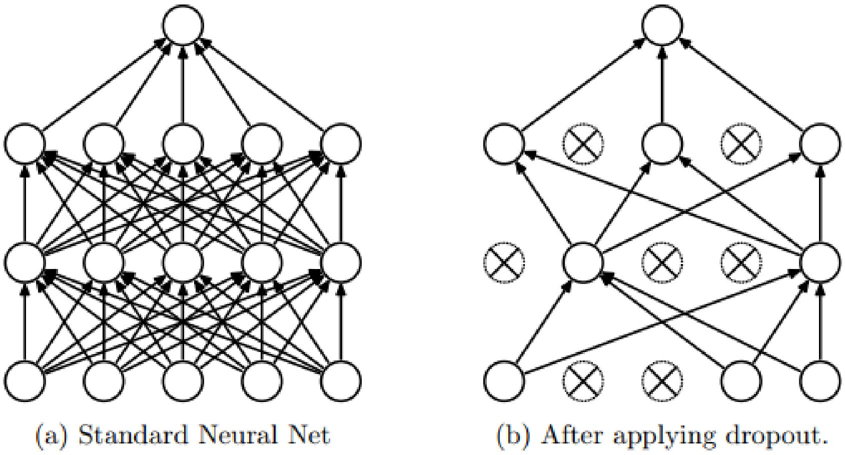


Figure 2.3 : Dropout neural net model [39].

Figure 2.3 demonstrates an example of the dropout neural net model, standard neural network is given in Figure 2.3 (a). When crossed units are dropped with its connections randomly *(i.e. two units are dropped in the first layer)*, the new neural net model is obtained as in Figure 2.3 (b).

Fully-Connected Layer

As stated in the name of this layer, neurons are fully pairwise connected between two consecutive layers like regular neural networks. Fully-connected layers are typically used in the last part of CNN architectures.

Loss Functions

Output layer usually represents the class scores for classification or the real-valued target for regression. Loss functions take place in the output layer and quantify the agreement between the predicted and the actual labels in the classification. CNNs try to minimize the loss function which becomes an optimization problem.

There are several loss functions such as SVM, Softmax, and Triplet Loss. The Softmax function is widely used as the output layer for image classification, it converts scores to probabilities. Each class has one neuron with the class probability in the output layer. Neural network predicts the one of classes which has the highest probability.

Softmax funtion is defined as:

$$softmax(x)_p = \frac{e^{x_p}}{\sum_{i=1}^{n} e^{x_i}} \tag{2.7}$$

2.2 CNN Architectures

First CNN architecture was proposed by LeCun et al., which is named as LeNet [36]. CNNs has become more popular since the first deep CNN architecture which is known as AlexNet [40] won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 challenge [41]. After AlexNet [40], well-known architectures such as VGG [42], GoogLeNet [23], ResNet [43] proposed for mainly computer vision tasks. The pre-trained models of these architectures are available online. A brief explanation about these architectures are given in this section.

AlexNet

AlexNet [40] is one of the deep convolutional neural network architecture and the winner of ILSVRC 2012 challenge [41]. AlexNet was trained on ImageNet [44] which has 1.2M images from 1000 classes. AlexNet [40] architecture has five convolutional layers and three fully-connected layers with 60 million parameters [40]. The last layer has probability distribution over 1000 different classes, since it was trained on ImageNet [44] dataset. Dropout [39] is used to prevent overfitting in this architecture.

VGG

VGG [42] has been proposed as the deep convolutional neural network in two versions and took place in the first and second in the localization and classification task in the ImageNet Challenge 2014, respectively. One of the VGG architectures has 16 layers, while another one has 19 layers. In this architecture, very small (3x3) convolution filters were used to improve accuracy. In the VGG-16, there are 13 convolutional layers and three fully connected layers. Five max-pooling layers follow some convolution layers. VGG architectures have around 140 million parameters.

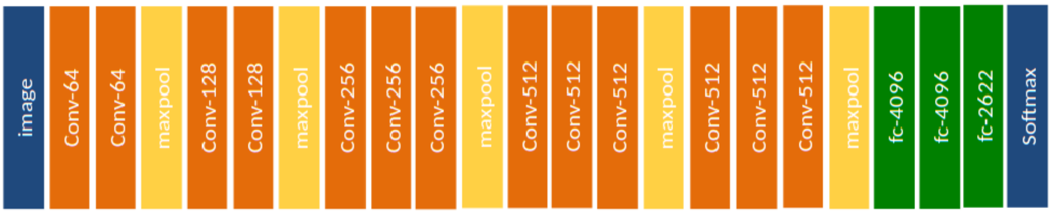


Figure 2.4 : VGG Face architecture [12].

Table 2.1 : Network configuration for VGG Face architecture [12].

Layer Type	Kernel Size	Stride	Number of Filter	FC Units
Conv	3x3	1x1	64	-
Conv	3x3	1x1	64	-
MaxPool	2x2	2x2	-	-
Conv	3x3	1x1	128	-
Conv	3x3	1x1	128	-
MaxPool	2x2	2x2	-	-
Conv	3x3	1x1	256	-
Conv	3x3	1x1	256	-
Conv	3x3	1x1	256	-
MaxPool	2x2	2x2	-	-
Conv	3x3	1x1	512	-
Conv	3x3	1x1	512	-
Conv	3x3	1x1	512	-
MaxPool	2x2	2x2	-	-
Conv	3x3	1x1	512	-
Conv	3x3	1x1	512	-
Conv	3x3	1x1	512	-
MaxPool	2x2	2x2	-	-
FC	-	-	-	4096
FC	-	-	-	4096
FC	-	-	-	2622

The new face dataset which consists of 2.6 million face images from 2622 celebrities has been introduced in [12]. Moreover, VGG Face, very deep CNN architecture, has been proposed for face verification problem which are inspired by VGG architecture [42]. Face recognition problem has been addressed as a classification problem in this architecture. The results for this study are comparable with the state-of-art results on benchmark face datasets like Labeled Faces In the Wild Dataset (LFW) [3] and YouTube Faces Dataset (YTF) [45]. Since VGG Face architecture pre-trained on this proposed dataset [12] which consists of face images, this pre-trained model was used to fine-tunne model on facial expression dataset in this thesis.

As demonstrated in Figure 2.4, VGG Face has 16 layers like VGG-16 [42]. There are 13 convolutional layers, some of them are followed by max-pooling layers. All the convolution layers are followed by ReLU activation function. There are three fully connected layers at the end of the network. The proposed dataset [12] has 2622 individuals, thus, last fully connected layer 2622 dimensional. The Softmax layer computes the class probabilities.

In Figure 2.4, the number of each convolution layer such as 64, 128, 256, and 512 indicates the number of the filters for each layer. In addition, Table 2.1 indicates the details of VGG Face CNN architecture. The filter size, the number of filters, stride, and padding are given for each convolution layer. In this configuration, fully-connected layers are also listed as convolution which defines as the special case of convolution [12].

GoogLeNet

GoogLeNet [23] has 22 layers, however it has about twelve times fewer parameters than AlexNet [40]. Besides, Inception module has been proposed for this architecture [23]. The main idea of the Inception module is that an input image is filtered with different size of filters which means multiple convolutions are performed on the same input. Also, pooling is conducted at the same time, then all results are concatenated. This module provides multi-level feature extraction from each input.

ResNet

ResNet (Residual Network) [43] was the winner of classification task in ILSVRC 2015 challenge. He et al. [43] suggested that the different number of layers which is up to 152 such as ResNet-50, ResNet-101. Although ResNet [43] is deeper than VGG [42], ResNet [43] has lower complexity than VGG [42]. The idea behind the ResNet [43] is that Residual Block which basically adds the original input to the output of convolution-relu-convolution series.

2.3 Transfer Learning

Training an entire CNN from scratch (starting with random initialization) requires more computation power, and takes so much time. Therefore, Transfer Learning methods are performed instead of training from scratch. Transfer Learning has mainly two approaches which depend on the size of the target dataset and similarity between source and target datasets [46]. These approaches are using CNN model for feature extraction and fine-tuning the CNN model. The first approach is employed when the target dataset contains a small number of samples and the target dataset is similar to the source dataset. In this approach, pre-trained CNN models on ImageNet dataset [44] *(e.g. ImageNet, which contains 1.2 million images with 1000 categories)* can be used as feature extractor. After extraction features which are also called as CNN codes, a linear classifier such as Linear SVM, and Softmax classifier is trained on the new dataset. Another approach for Transfer Learning is fine-tuning the CNN model on the new dataset. Not only retrain classifier but also weights of the pre-trained network are updated by using backpropagation during the process of fine-tuning. As explained before, earlier layers of CNN model have more generic features that are useful for many tasks, thus these layers can be fixed and last layers can be retrained to learn new task for new dataset. It can be used when the target dataset has sufficient amount of samples, otherwise, it leads to overfitting [47].

YÖK

3. FEATURE EXTRACTION

In this chapter, the local binary pattern for feature extraction is clarified. Also, feature normalization method and principal component analysis for dimension reduction are explained.

3.1 Feature Extraction by using Local Binary Pattern

The local binary pattern (LBP) operator was introduced for texture description based on texture unit which is represented by eight elements from neighbors of center pixel [9]. The LBP operator calculates a binary code for each unit of an image by using a center pixel and its neighborhoods. Center pixel is chosen as a threshold. If neighborhood pixel is smaller than the threshold, the pixel is labeled as zero. On the other side, if neighborhood pixel is bigger than the threshold, the pixel is labeled as one. Each binary code is converted to the decimal value which is used as a new pixel value.

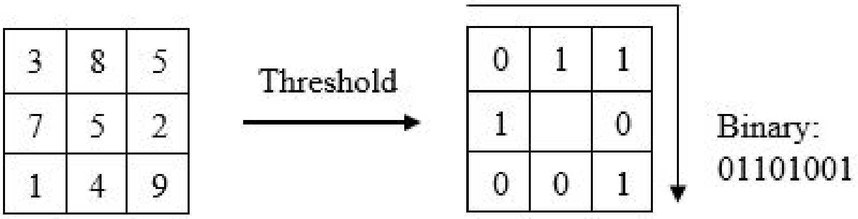


Figure 3.1 : The basic LBP operator.

The basic LBP operator which is defined for the 3x3 neighborhood is represented in Figure 3.1. According to this example 3.1, the center pixel of the 3x3 unit is chosen as the threshold which is equal to 5, neighborhood pixel values are converted the binary value. For instance, 3 is changed with 0, while 8 is changed with 1. Then, ’01101001’ binary code is obtained.

The histogram of labeled image ($f_1(x,y)$) is described in Equation 3.1 and this is used for texture descriptor.

$$H_{i,j} = \sum_{x,y} I\{f_l(x,y) = i\}, i = 0, ..., n-1,$$
$$I\{A\} = \begin{cases} 1, & \text{A is true} \\ 0, & \text{A is false} \end{cases} \tag{3.1}$$

The basic LBP operator leads to loss of feature for high-resolution images, therefore the LBP operator was improved in [48]. Thanks to the circular neighborhood and interpolating the pixel values, either different radius or size of the neighborhood can be chosen. Different neighborhood and radius parameters such as $LBP_{(8,1)}$, $LBP_{(12,1.5)}$, $LBP_{(16,2)}$, $LBP_{(24,3)}$ are illustrated in Figure 3.2.

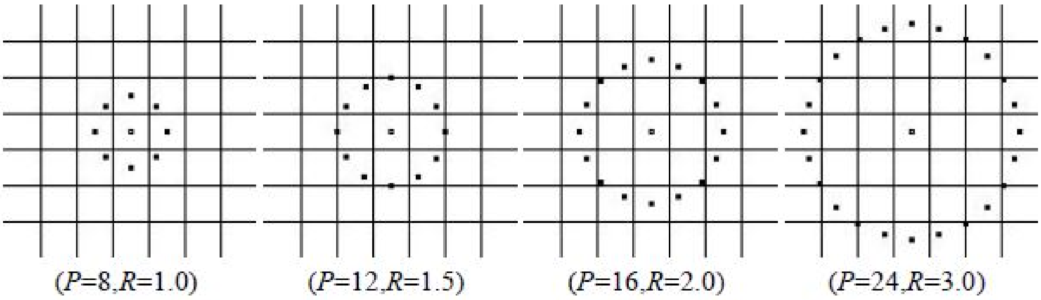


Figure 3.2 : Examples of different neighborhood parameters [48].

Uniform patterns is another extension of the LBP operator. According to researchers, the majority of images consists of the uniform pattern which means at most two transactions from 0-1 or 1-0 transitions in binary codes. For instance, 11000001 (2 transitions) is defined as a uniform pattern, while 10011001 (4 transitions) is not defined as a uniform pattern. The LBP operator with 8 neighborhoods has $2^8 = 256$ binary codes, 58 of them are a uniform pattern. The LBP operator has added a single label for other patterns. As a result, the dimension of the feature vector is found as 59-dimensional when neighborhood parameter is chosen as 8. Usage of uniform pattern can be indicated as $LBP_{(P,R)}^{(u2)}$.

The extended LBP operator [48] causes to loss of spatial information. Therefore, the spatially enhanced histogram was proposed for face recognition applications [49], and the proposed method for the histogram is defined in 3.2:

$$H_{i,j} = \sum_{x,y} I\{f_l(x,y) = i\} I\{(x,y) \in R_j\}, i = 0, ..., n-1, j = 0, ..., m-1, \tag{3.2}$$

In this approach [49], images are divided into regions, and then the histogram is calculated for each region, separately. After that, all histograms are concatenated. This feature extraction process with histogram concatenation is demonstrated in Figure 3.3.

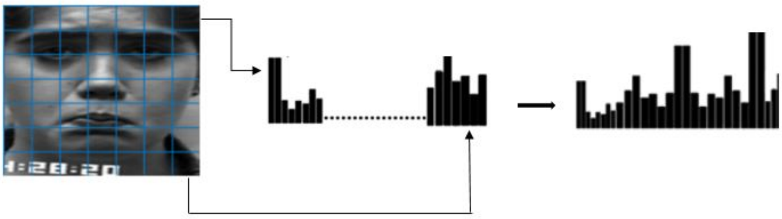


Figure 3.3 : Feature extraction process.

Assuming that the image is divided into m region, the dimension of new feature vector can be calculated by using Equation 3.3 where B_r means that dimension of the feature vector for each region. B_r is equal 59-dimensional when P is chosen as 8.

$$B = m * B_r \tag{3.3}$$

In this work, proposed the LBP version for face recognition [49] has been used for feature extraction. In [49], several LBP operators with different neighborhood and radius parameters were tried and results were reported. According to results, $LBP^{(u2)}_{(8,2)}$ operator was pointed as most efficient based on accuracy and complexity. In addition, 130×150 pixel images were divided into $7 \times 7 = 49$ regions in this previous work. Therefore, aligned images were resized to 130x150 pixel for all images. Then, the LBP operator was applied for 49 regions, separately. As mentioned before, the $LBP^{(u2)}_{(8,2)}$ operator produces 59-dimensional feature vector. According to this, each region has the 59-dimensional histogram. After histograms of all regions have been concatenated, $59 \times 49 = 2891$ dimensional feature vector has been obtained for each image.

The feature vector for each image in the pairs are subtracted from each other in the order of pair. If the pair consists of matched facial expressions, the new feature vector is labeled as one. On the other side, if the pair consists of mismatched facial expressions, the new feature vector is labeled as zero. These feature vector differences and labels have been used for binary classification.

3.2 Feature Normalization

L2 normalization is applied on each feature vector, before using features or applying dimension reduction. According to Equation 3.4, L2 norm is calculated where $\vec{v}$ is feature vector. Then, the normalized feature vector is calculated by using Equation 3.5.

$$\|\tilde{\mathbf{v}}\|_2 = \sqrt{\sum_{i=1}^{n} |v_i|^2} \tag{3.4}$$

$$v\_\vec{n}orm = \vec{v}/\|\tilde{\mathbf{v}}\|_2 \tag{3.5}$$

3.3 Principal Component Analysis

Principal component analysis (PCA) is widely used in many applications such as dimension reduction, data visualization, and loss data compression [50]. PCA is also known as Karhunen–Loève transform [51].

PCA is unsupervised and effective dimension reduction method, and it keeps as much information as possible. Maximization of variance is a significant criterion. Basically, PCA learns a representation of data, there is no linear correlation between elements of this representation. In other words, elements of transformed data are mutually uncorrelated [52].

Orthogonal and linear transformation of the input data is learned by PCA [37] that projects input ’x’ with assuming that input is already centered into lower dimension space on the direction of ’w’ in which there is maximum variance, the projected output ’z’ is calculated as following formula:

$$z = w^T x \tag{3.6}$$

$$
\begin{aligned}
Var(z) = Var(w^T x) &= E[(w^T x - w^T \mu)^2] \\
&= E[(w^T x - w^T \mu)(w^T x - w^T \mu)] \\
&= E[w^T (x-\mu)(x-\mu)^T w] \\
&= w^T E[(x-\mu)(x-\mu)^T] w \\
&= w^T \Sigma \, w
\end{aligned} \tag{3.7}
$$

PCA aims to find W such that Var(z) is maximize. According to Equation 3.3, Var(z) is calculated and it is maximized where $Cov(x) = \Sigma$. The covariance matrix of X with mean vector μ is defined in [52]:

$$Cov(x) = E[(x-\mu)(x-\mu)^T] = \Sigma \tag{3.8}$$

Eigenvalues and eigenvectors of Σ can be calculated by using singular value decomposition (SVD). The number of eigenvectors which has larger eigenvalues should be chosen for the reduced dimension. The columns of W consists of chosen eigenvectors of the Σ.

The covariance matrix of data will be $n \times n$ dimensional, W is $n \times k$ dimensional, where n is the number of features.

In order to choose reduced dimension, the proportion of variance is calculated as follows:

$$\frac{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + ... + \lambda_k}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + ... + \lambda_k + ... + \lambda_d} \tag{3.9}$$

In Equation 3.9, k is the dimension of the output, d is the dimension of the input and λ represents the eigenvalue. The proportion of variance can be calculated for different k values until this value is greater than 0.9.

Another method for choosing k values, plotting sorted descending eigenvalues. After a particular k value, eigenvalues are equal to almost zero. The particular k value can be selected.

In this thesis, PCA is utilized to dimension reduction in some experiments to reduce time and space cmplexity. Generally, around 99% of the variance is kept.

Steps of principal component analysis are briefly described as follows:

- Subtract the mean of input data from input data
- Find the covariance matrix of the centered input data
- Find the eigenvectors and eigenvalues of the covariance matrix
- Choose the reduced dimension by utilizing larger eigenvalues
- Transform the input to the new input space by using eigenvectors corresponding to the selected number of eigenvalues

4. CLASSIFICATION

In this chapter, SVM for classification and performance reporting methods are explained.

4.1 Support Vector Machines

The Support Vector Machine (SVM) has been used widely for classification and regression problems. SVM is defined as a maximum margin binary classifier [51].

Given N number of training data (X_i) and labels (y_i), where $x_i \in R_d$ is dimension of training data and $y_i \in \{-1, 1\}$, $\Phi(x)$ is feature transform and b is bias for linear separation problem in binary SVM classification.

$$y(x) = w^T \Phi(x) + b \tag{4.1}$$

The goal of SVM is to find hyperplane (decision boundary) which seperates two class by maximizing the margin with the optimizer such as Lagrange multipliers. The margin is defined as a distance between the separating decision boundary and the nearest instances which are called support vectors. Figure 4.1 shows an illustration for SVM. A few options for decision boundary can be demonstrated in Figure 4.1 (a), while optimal decision boundary with the maximum margin can be observed in Figure 4.1 (b).

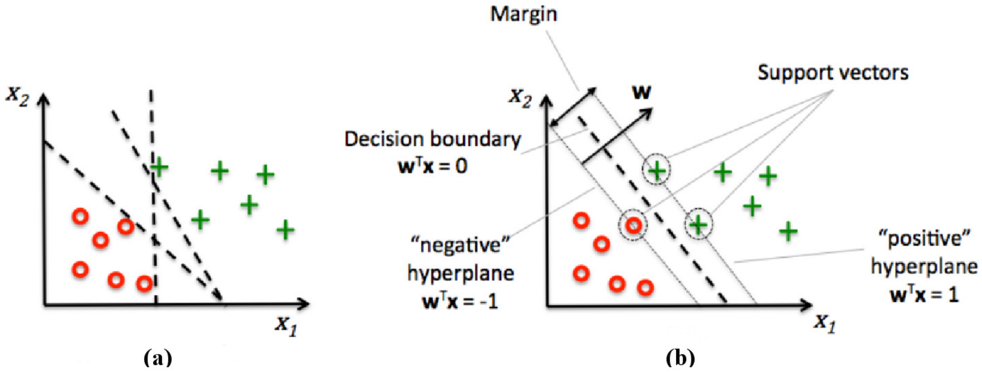


Figure 4.1 : SVM illustration for a two-class problem [53].

In order to maximize the margin, SVM aims to minimize:

$$C\sum_{n=1}^{N}\xi_n+\frac{1}{2}\|w\|^2 \tag{4.2}$$

where ξ_n is slack variable which is introduced by Schölkopf et al. [54], C > 0 is the cost parameters and arranges the balance between the margin and slack variables [51].

SVM is originally defined as a linear classifier, however, it also performs non-linear classification by using kernel method (kernel trick). Linear kernel and Gaussian kernel which is also known as the Radial Basis Function kernel are used in this study.

Linear Kernel:

$$k(x^{(i)},x^{(j)})=\langle x^{(i)},x^{(j)}\rangle \tag{4.3}$$

Gaussian (RBF) Kernel:

$$k(x^{(i)},x^{(j)})=exp\left(-\frac{\|x^{(i)}-x^{(j)}\|^2}{2\sigma^2}\right) \tag{4.4}$$

Gaussian kernel measures distance between pair of example and define as similarity function. σ parameter decides similarity metric go down whether fast or not.

C parameter for Linear kernel, C and σ^2 parameters for the Gaussian kernel is optimized. Large C value leads to lower bias and high variance which corresponds the overfitting, while small C value leads to higher bias and lower variance which corresponds the underfitting. On the other hand, large σ^2 value cause to higher bias and lower variance, while small σ^2 value cause to lower bias and high variance. In order to prevent overfitting, C should be decreased and σ^2 should be increased for the Gaussian kernel.

There are two approaches for multiclass SVM which are one-against-all and one-against-one. In one-against-one approach, K(K-1)/2 binary models can be produced, K is defined as the number of classes. In this approach, all possible binary SVMs are trained and which class has highest numbers of prediction is chosen as predicted class. In one-against-all approach, a model for each class is produced [51].

Publicly available LIBSVM library is used for SVM [55]. LIBSVM supports one-against-one classification for multiclass problems. As recommended in [55], grid searches should be applied to select optimized parameters for each train data. 110 parameters for C = 2^{-5}, 2^{-3}, 2^{-1}, ..., 2^{13}, 2^{15} and γ = 2^{-15}, 2^{-13}, 2^{-11}, ..., 2^{3} values are tried by using 5 fold cross-validation.

4.2 Performance Reporting

Several methods exist for performance reporting such as confusion matrix, precision, recall, overall accuracy, F1-score and receiver operator characteristics (ROC) curves. We used overall accuracy and confusion matrix which are generally used for classification, and these results can be compared with previous works.

Overall accuracy is calculated by using Equation 4.5. It defines as the number of correct prediction in all predictions.

$$overall\ accuracy = \frac{tp+tn}{tp+tn+fp+fn} \tag{4.5}$$

The confusion matrix (contingency table) shows that number of predictions for other classes for each true class. Table 4.1 is an example of confusion matrix for binary classification.

Table 4.1 : The confusion matrix of a two-class.

		Predicted	
		Negative	**Positive**
Actual	**Negative**	TN	FP
	Positive	FN	TP

True positive (TP), true negative (TN), false positive (FP), and false negative (FN) are defined as follows:

- **True Negative:** The number of negative instances predicted as negative
- **False Positive:** The number of negative instances predicted as positive
- **False Negative:** The number of positive instances predicted as negative
- **True Positive:** The number of positive instances predicted as positive

5. DATASET PREPARATION

A brief explanation of available facial expression datasets is given in this section. The CK+, the JAFFE, the FER-2013, and the SFEW datasets are chosen for experiments and explained in detail. The process of face alignment, besides creating pairs and experimental setup for different experiments are clarified consecutively.

5.1 Datasets

There are several well-known facial expression datasets in the controlled and the uncontrolled environments: the Extended Cohn-Kanade dataset (CK+) [5], the Japanese Female Facial Expression dataset (JAFFE) [13], the Multi-PIE [56], the MMI [57],the DISFA [58], the UNBC-McMaster Pain Shoulder Archive [59], the Facial Expression Recognition 2013 [15], the Acted Facial Expressions In The Wild (AFEW) [60], and the Static Facial Expressions in the Wild (SFEW) [14].

The Multi Pose, Illumination, and Expression Dataset: The Multi-PIE dataset [56] contains more than 750,000 images which are collected in four sessions under 15 different points of view and 19 lighting conditions. There are 337 subjects with different ethnics such as European-Americans, Asian, African-American, and others. Facial expressions in this dataset: neutral, smile, surprise, squint, disgust, and scream.

MMI Facial Expression Dataset: The MMI [57] has 740 static images and 848 videos of 19 individuals with different ethnics that are European, Asian, or South American. Static images and image-sequences are labeled with facial expressions, single AU and multiple AUs in frontal and profile views.

A Spontaneous Facial Action Intensity Dataset: The DISFA [58] contains 130,000 videos of 27 subjects (12 female, 15 male) in range of age between 18 to 50. 66-facial landmarks are given for each video. Each video frame is also coded for presence, absence, and intensity of AUs for spontaneous expressions.

The UNBC-McMaster Shoulder Pain Expression Archive Dataset: This pain dataset [59] consists of spontaneous facial expression for 200 videos from 25 individuals, 48,398 FACS coded frames, pain scores for all frames, self-reports and observer measures for sequences, and 66 points AAM landmarks. There are 129 subjects (66 female, 63 male) with having shoulder pain.

The Facial Expression Recognition 2013: The FER-2013 [15] dataset was released for ICML 2013 Challenges in Representation Learning. The dataset includes 35,887 images taken from the real world by using the Google image search. Thus, it is also called as a wild dataset.

The dataset consists of three parts: private, public and training which correspond test, validation and train set, respectively. Six basic expressions which are anger, disgust, fear, happiness, sadness, surprise besides neutral are labeled in this dataset. The number of images for each expression is given in Table 5.1.

Table 5.1 : Frequency of facial expressions in the FER-2013 dataset [15].

Facial Expression	The number of images
Anger (An)	4953
Disgust (Di)	547
Fear (Fe)	5121
Happiness (Ha)	8989
Sadness (Sa)	6077
Surprise (Su)	4002
Neutral (Ne)	6198

Figure 5.1 illustrates several sample images from the FER-2013 dataset [15]. As observed in this figure, the dataset contains lots of variations from the individuals from different ethnics, age, and gender besides many appearance variations in terms of view angle, illuminations, and occlusions (i.e. hands).

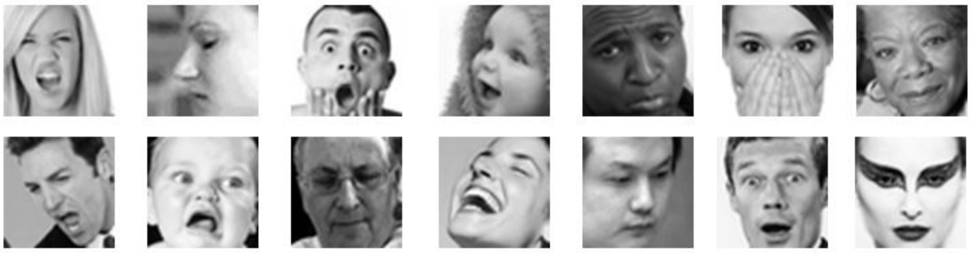


Figure 5.1 : Sample images from the FER-2013 dataset [15].

Acted Facial Expressions In The Wild: The AFEW [60] is dynamic temporal dataset. The dataset is collected from 37 movies which are chosen from realistic screenplays. Therefore, it can be assumed that facial expressions are taken from real life. 957 videos are labeled with six basic expressions and neutral. If video sequences have more than an actor, all of them are labeled with own facial expressions.

Static Facial Expressions in the Wild: The SFEW [14] has been created by choosing frames from AFEW, therefore, it is named as Static Facial Expressions. SFEW includes static images which are labeled with six basic expressions and neutral by two people. This dataset contains various pose angels, different resolutions, occlusions, thus, it is really close the real world images.

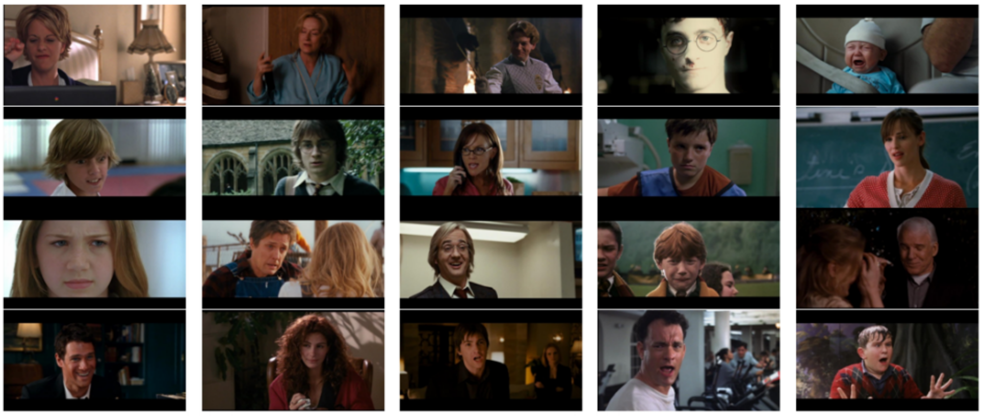


Figure 5.2 : Sample images from the SFEW dataset [14].

The Extended Cohn-Kanade Dataset: The CK+ [5] consists of 123 individuals who are between 18 to 50 age, 69% female, 81%, Euro-American, 13% Afro-American, and 6%other groups. There are 593 sequences in CK+ datasets. Each of sequence includes images from the neutral frame (onset) to the last frame which is defined as peak frame. This dataset includes posed and spontaneous expressions. In addition, peak expressions are fully FACS coded and some of them were labeled based on FACS Investigator Guide [61] and affirmed by facial expression researchers. Sample images from the CK+ dataset can be shown in Figure 5.3.

Each AU is defined as a movement such as AU1 is described as "Inner Brow Raiser" whereas it is shown in sadness expression. AU12 is defined as "Lip Corner Puller" must be presented for happy. The combination of AUs given in Table 5.2 describes a facial expression.

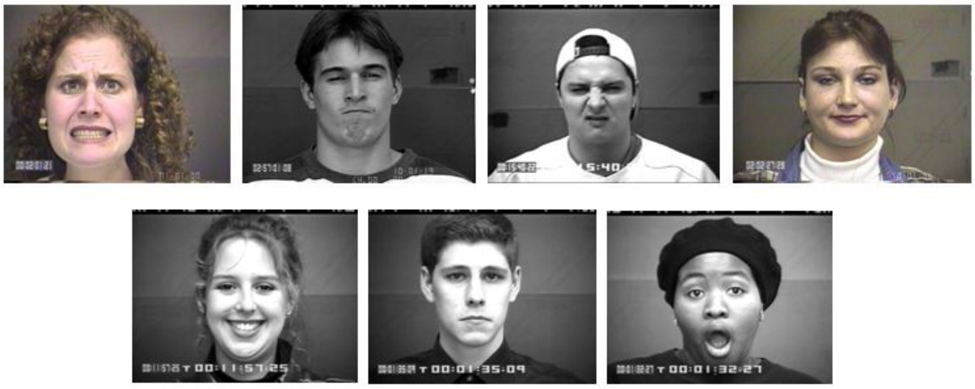


Figure 5.3 : Sample images from the CK+ dataset [5].

The process of labeling peak frames for this dataset has three steps: (1) According to AU combinations for each image, the facial expression is labeled. If the clip has AU(s) which is not in criteria or required AU(s) is absent, the clip is exempted. (2) If a clip has AU which is not included in different facial expression criteria, whether facial expression or spoiler is consistent are decided. (3) In this step, researchers have determined that target facial expression is correct or not. After this process, 327 images from 118 individuals are labeled with anger, contempt, disgust, fear, happy, sadness, and surprise.

Table 5.2 : Descriptions of facial expression in terms of FACS [5].

Facial Expression	Criteria
Angry	AU23 and AU24 must be present in the AU combination
Contempt	AU14 must be present (either unilateral or bilateral)
Disgust	Either AU9 or AU10 must be present
Fear	AU combination of AU1+2+4 must be present, unless AU5 is of intensity E then AU4 can be absent
Happy	AU12 must be present
Sadness	Either AU1+4+15 or 11 must be present. An exception is AU6+15
Surprise	Either AU1+2 or 5 must be present, the intensity of AU5 must not be stronger than B

Table 5.3 indicates the frequencies of facial expressions. There is no equal distribution between the number of images for each expression. For instance, contempt expression has only 18 images, while surprise expression has 83 images. Another important point is that each person has not image for all facial expressions. In the other words, if one facial expression exists for a person, this person has just one image for this facial expression [5].

Table 5.3 : Frequency of facial expressions in the CK+ dataset [5].

Facial Expression	The number of images
Angry (An)	45
Contempt (Co)	18
Disgust (Di)	59
Fear (Fe)	25
Happy (Ha)	69
Sadness (Sa)	28
Surprise (Su)	83

The Japanese Female Facial Expression dataset: The JAFFE Dataset [13] includes 213 images of 10 Japanese female models. Six basic expressions and neutral are posed by participants. Figure 5.4 shows sample images from the JAFFE dataset with the resolution of 213 grayscale images is 256x256. Angry, disgust, fear, happy, sad, surprise facial expression and neutral can be indicated, respectively.

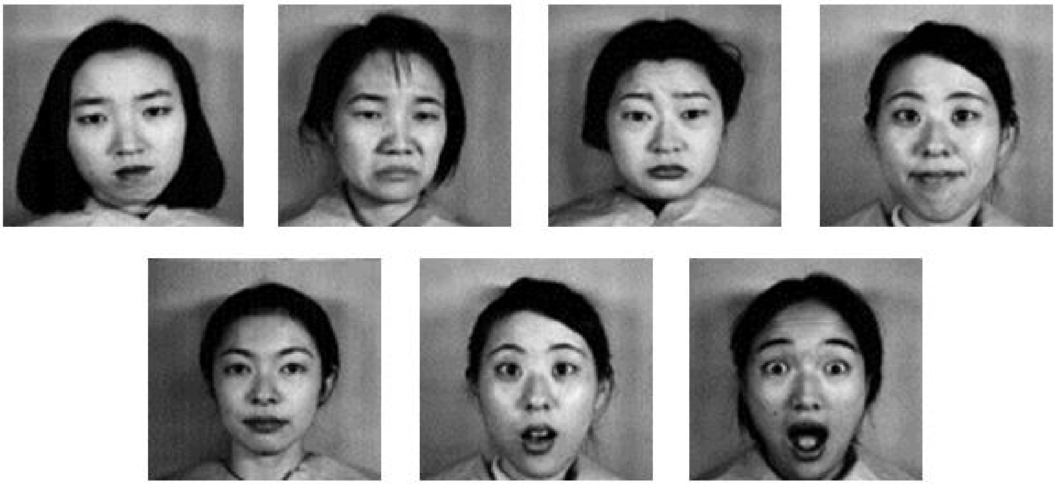


Figure 5.4 : Sample images from the JAFFE dataset [13].

As demonstrated in Table 5.4, frequencies of facial expressions are almost equal number.

Table 5.4 : Frequency of facial expressions in the JAFFE dataset [13].

Facial expression	The number of images
Angry (An)	30
Disgust (Di)	29
Fear (Fe)	32
Happy (Ha)	31
Sad (Sa)	31
Surprise (Su)	30
Neutral (Ne)	30

5.2 Face Alignment

5.2.1 Face alignment on the CK+ dataset

Face alignment is the important process for face recognition applications. Each image sequence consists of frontal views and 30-degree views in the CK+ dataset. Therefore, face alignment is required to frontalize images. In addition to this, regions of faces are cropped from original image sequences by using landmark points.

68 landmark points for each image are defined in the CK+ dataset. As shown in Figure 5.5, each eye has 6 landmark points which are used to find the center pixel of eyes. Face alignment is applied and extracted the face from images by using central pixel of eyes. Example of finding eye distance is illustrated in Figure 5.5, respectively.

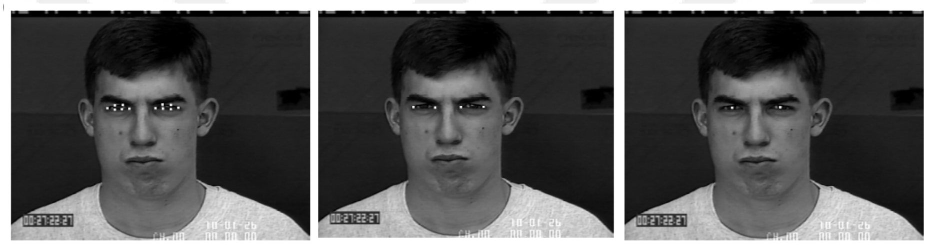


Figure 5.5 : Eyes of landmarks (a), eye distance (b), center of eyes (c) on the CK+ dataset.

In Figure 5.6 and 5.7, examples of face alignment can be shown for disgust and surprised expressions. In this process, irrelevant background parts of images are eliminated from the images. For instance, ears and hair can be assumed as irrelevant part of facial expression images are discarded. Choosing the size of the face region is really hard because of each person has different size of the face. Different pixel values have been tried to find best face region for all images. According to various pixel experiments, 350x430 pixels are chosen to obtain from either 640x490 or 640x480 pixel images with 8-bit grayscale or 24-bit color images.

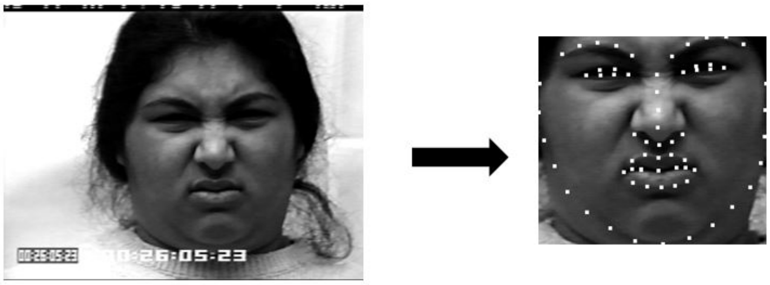


Figure 5.6 : An example of face alignment on the CK+ dataset (disgust expression).

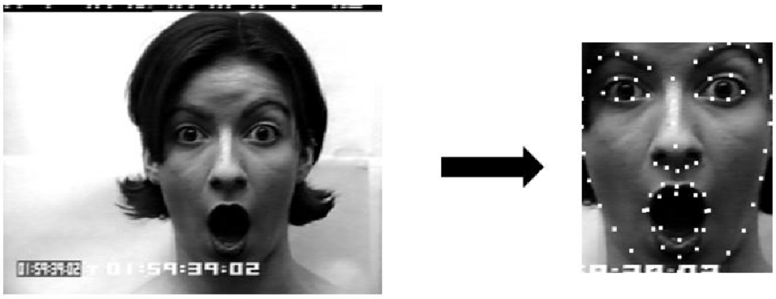


Figure 5.7 : An example of face alignment on the CK+ dataset (surprised expression).

5.2.2 Face alignment on the JAFFE dataset

Facial landmark points are not defined in the JAFFE dataset. Therefore, 68 facial landmark points are found for all images. Eye distances are calculated by utilizing six eye landmark points for each eye with the same procedure like the CK+ dataset before the centers of eyes are described. Figure 5.8 shows that steps of finding eye center for each image, respectively.

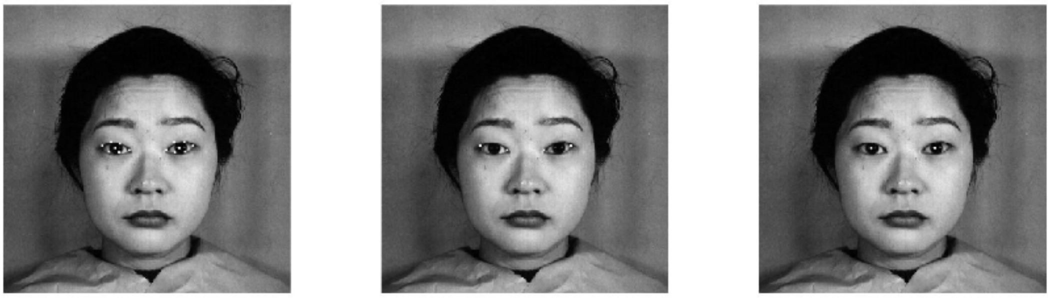


Figure 5.8 : Eyes of landmarks (a), eye distance (b), center of eyes (c) on the JAFFE dataset.

An example of face alignment process on the JAFFE dataset can be illustrated in Figure 5.9. The resolution of each images is preserved during face alignment process whereas the regions of faces are cropped and resized to the original resolution size 256x256 pixels. Hair, ears and background part of images are also eliminated for this dataset.

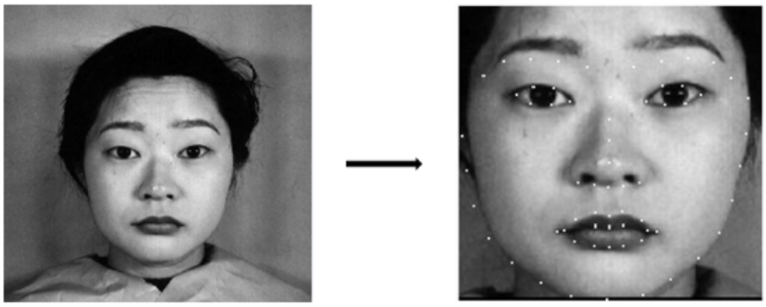


Figure 5.9 : An example of face alignment on the JAFFE dataset.

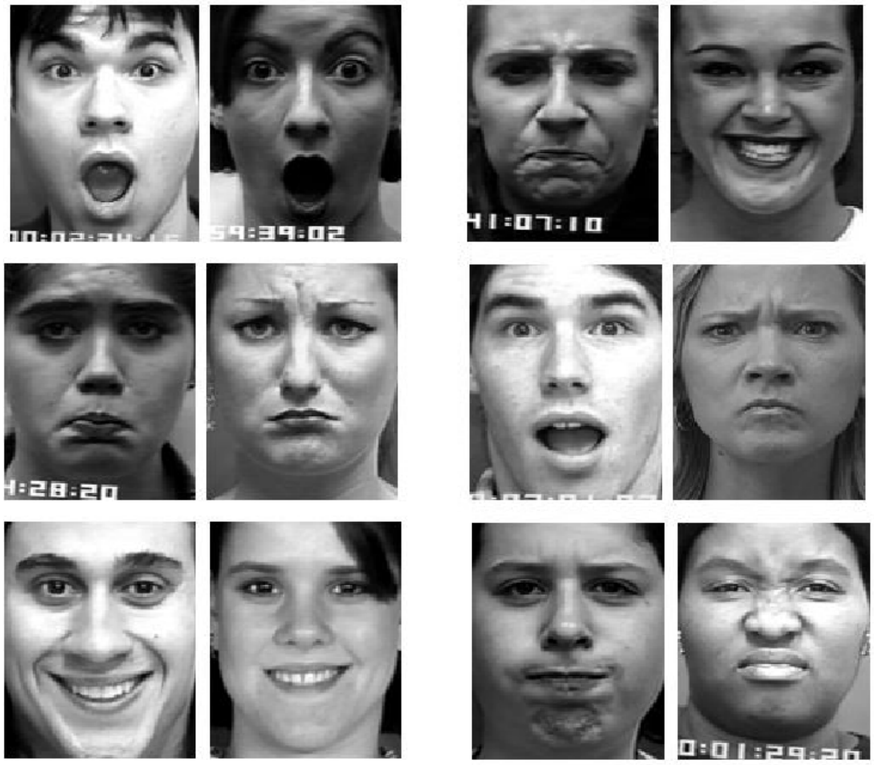
5.3 Creating Facial Expression Pairs

In this section, creating pairs on the CK+ dataset and the JAFFE dataset are explained in detail, respectively. Three different experiment sets are constituted for the CK+ database, while there is only one experiment set for the JAFFE dataset.

5.3.1 Creating pairs on the CK+ dataset

5.3.1.1 Creating pairs on the LFW format

The number of images for each facial expression is given in Table 5.3. All possible matched pairs for each expression are created. For instance, one image of angry expression is matched with the other images of angry expression, and 990 matched pairs are obtained. The number of matched facial expression can be represented in Table 5.5, according to the number of matched pairs calculated by using combination formula *(i.e. (n *(n-1))/2)*. In total, 9281 number of matched pairs are generated for all expressions. Figure 5.10 (a) and Figure 5.10 (b) indicate the examples of matched and mismatched pairs for the CK+ dataset, respectively.



(a) (b)

Figure 5.10 : Examples of matched (a) and mismatched (b) facial expressions on the CK+ dataset.

Table 5.5 : The number of matched pairs on the CK+ dataset.

Facial expression	**The number of matched pairs**
Angry (An)	990
Contempt (Co)	153
Disgust (Di)	1711
Fear (Fe)	300
Happy (Ha)	2346
Sadness (Sa)	378
Surprise (Su)	3403

Labeled Faces in the Wild (LFW) [3] is one of benchmark dataset for face verification which is a kind of pair matching problem, in the same way, we proposed pair matching formulation for facial expressions. Thus, the LFW dataset format is used to split our matched and mismatched pairs for our first experiments on the CK+ dataset.

The LFW dataset contains 13,233 unconstrained face images from 5749 individuals. 1680 of people have two or more images, while the others have just one image. The LFW dataset is split into two parts: View 1 and View 2, instead of splitting train, validation, and test sets. The aim of this approach, aggrandize data for training and testing pairs.

According to LFW technical report [3], View 1 is used for model development and parameter selection, while, View 2 is just used for performance reporting. Usage of View 1 and View 2 is explained in detail:

- **View 1: Model selection and algorithm development**
  - View 1 consists of train and test sets.
  - Train set is used for training desired model and parameters.
  - Retrain model until finding to best accuracy on test set.
- **View 2: Performance reporting**
  - View 2 includes 10 subsets.
  - 10 experiments are conducted independently by leave-one-out cross -validation.
  - One different subset is chosen as a test set for each experiment, while others are used for training.

- 10 different results are recorded and used for performance reporting.

Performance reporting formulations are also described in LFW technical report [3]. The estimated mean accuracy is calculated by using Equation 5.1.

$$\hat{\mu} = \frac{\sum_{i=1}^{10} p_i}{10} \tag{5.1}$$

In addition to mean accuracy, standard deviation must be reported for View 2 datasets. The standard error of the mean is defined as follows:

$$S_E = \frac{\hat{\sigma}}{\sqrt{10}} \tag{5.2}$$

The estimate of the standard deviation is also given in Equation 5.3.

$$\hat{\sigma} = \sqrt{\frac{\sum_{i=1}^{10} (p_i - \hat{\mu})^2}{9}} \tag{5.3}$$

9250 matched pairs are chosen and split into View 1 and View 2 in the LFW format to use for the first pair matching experiment. For the first experiment, the number of mismatched pairs is also chosen as the same number of matched pairs like the LFW dataset. 9250 mismatched pairs are chosen from all possible mismatched pairs, eventually. During creating and selecting pairs, subjects are not considered which means that one subject can appear in one of the pairs in one of the subsets in the LFW format, and the same subject can appear in another pair in the different subset. While the same pair is not presented in different subsets.

Table 5.6 : The number of pairs for the LFW format on the CK+ dataset.

Datasets	View 1		View 2		The Number of Pairs
	Train	Test	Train	Test	
LFW	2200	1000	600 x 9	600	9200
Proposed dataset	4000	2500	1200 x 9	1200	18500

The number of pairs in the LFW datasets and the proposed datasets can be observed in Table 5.6. View1 set of the LFW includes 1100 pairs of matched images and 1100 pairs of mismatched images for training, 500 pairs of matched images and 500 pairs of mismatched images for testing. Each View 2 subset has 300 pairs of matched images and 300 pairs of mismatched images. In each experiment for performance reporting, one subset used for testing has 600 pairs while other nine subsets used for training have 5400 pairs.

In our proposed dataset has approximately twice as large as the LFW dataset. View 1 of the proposed dataset has 2000 pairs of matched images and 2000 pairs of mismatched images for training, whereas 1250 pairs of matched images and 1250 pairs of mismatched images for testing. On the other side, View 2 has 10 subsets which consist of 600 pairs of matched images and 600 pairs of mismatched images.

5.3.1.2 Creating pairs for LOSO experiments

In addition to theLFW format, two experiment sets are created by using two different approaches for LOSO experiments that are performed to find facial expressions. Unlike the LFW format, pairs are created in the subject independent manner for both. One of them is divided into test and train sets, named as LOSO experiments while another one has five-fold to conduct five-fold cross-validation experiments which are called five-fold experiments in this study.

LOSO pairs: All possible matched pairs are created similar to the LFW format, and mismatched pairs are chosen the exact number of matched pairs for the train set. Due to creating pairs for leave-one-subject-out (LOSO) experiments, each train set should not be included a particular subject. The train set is used in training process for this particular subject. In the light of this information, after created all pairs, pairs which include images of one subjects are discarded, and train set of this particular subjects is constituted. Since 118 subjects exist in the CK+ dataset, 118 train sets are obtained in overall.

In order to find the facial expression for all images, each image should be own test set. A test set includes matched and mismatched pairs of a particular image, the other images without using the images of the same subject. For instance, happy image of a particular subject is matched and mismatched with the other images which belong to other subjects. Each test set has approximately 320 pairs. By utilizing results of test pairs, votes for all expressions are calculated. When the vote is considered as matched results, the facial expression with the highest number of votes is defined as the facial expression.

To sum up, 118 train sets for each subject and 327 test sets for all images are utilized to define the facial expression for each image.

Five-fold pairs: Order of pairs in a pair has a significant effect on training. Therefore, two combinations of each pair *(i.e. a-b, b-a)* are used in the train set. Due to the fact that lots of pairs are obtained for LOSO experiments, five-fold cross-validation is chosen to conduct another experiment.

118 subjects are divided into five set, four of them has 24 subjects while another one has 22 subjects. Each train fold includes pairs which consist of subjects of combinations different 4 subject sets, in this way, each train fold has around 24,000 pairs. Subjects of the remaining set are using to create test pairs. In the test sets, each image is matched or mismatched with all images from other sets of subjects. Similar to LOSO test set, each image has own test set. In this situation, each image has a test set which has approximately 250 pairs.

Briefly, five models are trained and 327 images are tested in this experiment. The test set of each image has tested with the trained model without includes this subject in the train set.

5.3.2 Creating pairs on the JAFFE dataset

As mentioned the JAFFE dataset explanation, this dataset consists of 10 subjects. One subject has more than one images for each facial expression. All possible matched pairs are created in subject independent manner. For instance, an image with a happy expression on one subject is matched with all happy expression images of other subjects. Images of the same subject are not matched or mismatched. Since the differences of features from matched and mismatched pairs are used to create training data, the order of images in a pair is important as mentioned before. Therefore, two combinations for each pair *(i.e. a-b and b-a)* are created and used to find feature differences. None of the matched or mismatched pairs consists of images from the same subject. The number of mismatched pairs is chosen as triple the number of matched pairs.

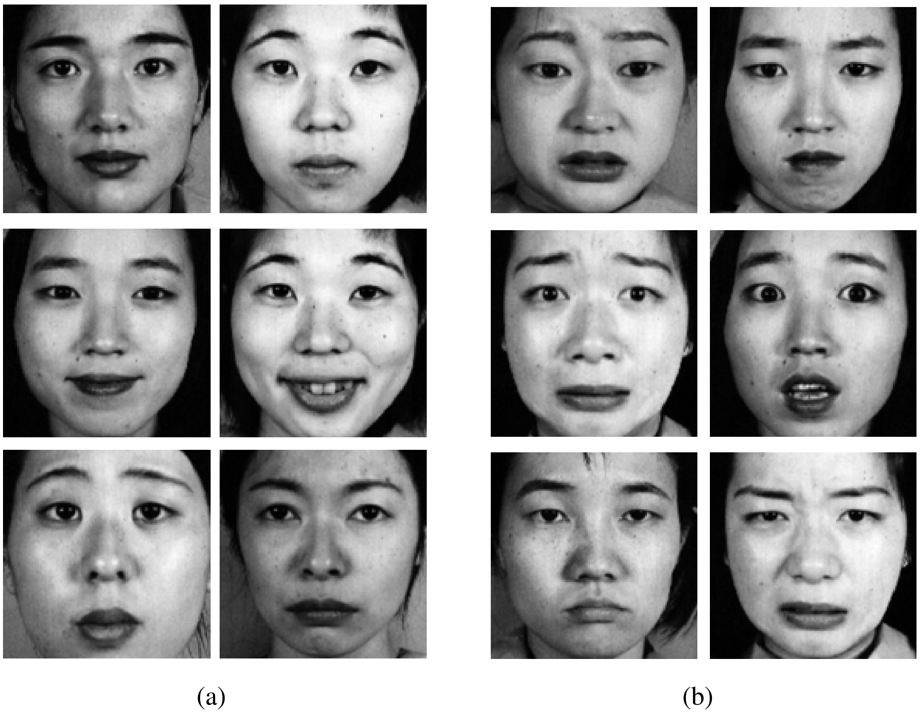
The JAFFE dataset has fewer subjects and accordingly fewer pairs when compared with the CK+ dataset. Therefore, five-fold experiments are not required for this dataset.

In the JAFFE dataset explanation, Table 5.4 demonstrates the frequency of facial expressions. The overall numbers of matched and mismatched pairs are represented in Table 5.7.

Table 5.7 : The number of pairs on the JAFFE dataset.

	The number of Pairs
Matched Pairs	5830
Mismatched Pairs	17494
Total	23324

Examples of matched pairs for neutral, happy and sad facial expressions can be demonstrated in Figure 5.11 (a), respectively. On the other hand, examples of mismatched pairs on the JAFFE dataset are also represented in Figure 5.11 (b). Fear and angry facial expressions can be mismatched in the first row. The second row, fear and surprise expressions are given, and angry and disgust facial expressions are placed in the last row, respectively.



(a) (b)

Figure 5.11 : Examples of matched (a) and mismatched (b) facial expressions on the JAFFE dataset.

The number of matched and mismatched pairs before creating the train sets for each subject in terms of facial expressions can be shown in Table 5.8. As analyzed from this table, a close the number of images for each facial expression yields a balanced combination of expression pairs. This balanced distribution of pairs improves facial expression accuracy. Trained model can be learned all combinations without bias.

Table 5.8 : The number of matched and mismatched pairs for each combination of facial expressions on the JAFFE dataset.

	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**	**Ne**
An	**810**	492	422	388	364	465	359
Di	492	**754**	504	353	366	344	348
Fe	422	504	**920**	575	387	382	374
Ha	388	353	575	**862**	367	340	524
Sa	364	366	387	367	**864**	535	521
Su	465	344	382	340	535	**810**	337
Ne	359	348	374	524	521	337	**810**

There is a constrained about the usage number of each image in all pairs, during the process of creating pairs. The number is chosen in range 54 to 85. For instance, one image can be presented in all pairs at least 54 times, and it can be up to 85 times. Otherwise, one subject can appear in more than other subjects such as two times or more. This situation can lead to the bias for this particular subjects.

Table 5.9 : Subject based training pair numbers on the JAFFE dataset.

Subjects	**The number of training pairs**
KA	18160
KL	18384
KM	18380
KR	18820
MK	18598
NA	18598
NM	18816
TM	18704
UY	19086
YM	19014

10 train sets are constrained for each subject after created matched and mismatched pairs. Matched and mismatch pairs which not include the image of one subject are selected for train set of this subject. Each subject has own training set according to discarding the matched and mismatched pairs which include the image from a particular subject.

The first column represents subjects in Table 5.9, these IDs are described in the dataset. To sum up, 10 train sets and 213 test sets are composed for the JAFFE experiments. Each subject has around between 18,000 and 19,000 pairs in their train sets, while each image has one test set which has around 190 pairs.

YK

6. EXPERIMENTAL RESULTS

We conducted extensive experiments for facial expression recognition by using pair matching formulation on the two well-known facial expression datasets: the CK+ and the JAFFE datasets which are widely used to evaluate facial expression recognition systems. On the other hand, several experiments for facial expression recognition in the wild were performed on the FER-2013 and the SFEW datasets.

6.1 Results on the CK+ Dataset

In this section, experimental results on the CK+ dataset are explained in detail.

6.1.1 Pair matching

As stated in previous sections, all images were aligned and converted to grayscale images. Pairs were created as matched and mismatched pairs, and the LFW format *(i.e. View 1 and View 2)* has been used for these experiments. Features were extracted for each image by using $LBP_{(8,2)}^{(u2)}$ operators. Feature differences between each pair of images are calculated by the subtraction operation. These feature differences were labeled as 1 for matched pairs and 0 for mismatched pairs. Then, binary SVM classifications by using Linear and Gaussian kernels were applied.

As explained in Section 5.3, pairs were divided into two views as defined as the LFW format. View 1 was used for model selection and algorithm development. While View 2 was used for performance reporting by using 10-fold cross-validation. In order to find accuracy, the proposed performance reporting method for the LFW benchmark was used.

According to our pairs distribution for training and testing, none of the same pairs take place in test and train test as well as View 1 and View 2. However, these experiments were performed without subject independent manner which means that same subject can appear in a pair of train and test set. Thus, we achieved higher accuracy.

As shown in Table 6.1, 97.36% and 99.28% accuracies were achieved by using Linear and Gaussian kernel, respectively.

Table 6.1 : Our results of the pair matching on the CK+ dataset.

Methods	**Accuracy (%)**
LBP + SVM (Linear kernel)	97.36 ± 0.0180
LBP + SVM (Gaussian kernel)	99.28 ± 0.0038

6.1.2 Facial expression recognition

Our baseline method was applied for facial expression recognition on the CK+ dataset. This experiment was conducted to analyze the effect of pair matching formulation. $LBP_{(8,2)}^{(u2)}$ operator was used for feature extraction, and SVM with leave-one-subject-out (LOSO) strategy was used for facial expression recognition. According to LOSO strategy, images of one subject is chosen as a test set, and the other images take part in the train test.

In Table 6.2, the confusion matrix indicates the facial expression accuracy according to facial expressions. As shown in this table, contempt-fear, sad-angry expression are confused with each other. In addition, contempt and fear expressions have fewer images than other expressions, thus recognition of accuracy for this facial expressions are lower than others. Except for contempt expression, our results are better than the baseline paper [5]. These results were compared with pair matching formulation in Section 6.1.3.

The overall accuracy of facial expression recognition was found as 91.44% by using Equation 4.5.

Table 6.2 : The confusion matrix for LOSO on the CK+ dataset.

	An	**Co**	**Di**	**Fe**	**Ha**	**Sa**	**Su**
An	**86.67**	2.22	6.67	2.22	0	2.22	0
Co	0	**77.78**	0	11.11	0	0	11.11
Di	1.69	0	**98.31**	0	0	0	0
Fe	4.00	4.00	0	**76.00**	12.00	0	4.00
Ha	0	0	0	0	**100.00**	0	0
Sa	17.86	0	3.57	3.57	0	**71.43**	3.57
Su	0	1.20	1.20	1.20	0	0	**96.40**

6.1.3 Pair matching for facial expression recognition

In these experiments, the proposed pair matching formulation was used for facial expression recognition. As explained in Section 5.3.1.2, two different setups are used for these experiments. One of them has 118 train sets and 327 test sets. Since the CK+ dataset has 118 individuals and experiments were conducted based on LOSO strategy, in this case, each subject has tested own SVM model. On the other hand, another experiment was performed on five-fold, therefore 5 train sets, 327 test sets are available for this setup. Five train sets were created in subject independent manner.

Each image has own test pairs, that is 327 test sets exist for both experiments. By utilizing pair matching results, the probability for each class are calculated for each image, and the highest probability is chosen as a predicted expression.

The confusion matrix for the LOSO experiments can be shown in Table 6.3. When compared the results with the facial expression recognition without using pair matching formulation in Section 6.1.2, we obtained higher or equal results, except for disgust expression. The overall accuracy of facial expression recognition was found as 92.35% by using Equation 4.5. According to results, the overall accuracy was also increased by using pair matching.

Table 6.3 : The confusion matrix of the proposed pair matching on the CK+ dataset.

	An	**Co**	**Di**	**Fe**	**Ha**	**Sa**	**Su**
An	**88.89**	2.22	2.22	0	0	6.67	0
Co	5.56	**88.89**	0	0	0	5.56	0
Di	3.39	0	**94.92**	0	0	0	1.69
Fe	0	4.00	0	**76.00**	12.00	8.00	0
Ha	0	0	0	0	**100.00**	0	0
Sa	21.43	0	3.57	0	0	**75.00**	0
Su	0	2.41	0	0	0	0	**97.59**

Another pair matching experiment was conducted on five-fold pairs setup which is described in Section 5.3.1.2. SVM parameters were optimized for each fold by using grid search and each fold has own parameters in these experiments. Results are given in Table 6.4. Since train sets have not included more individuals like LOSO pair setups which means that individual variations are fewer, the result of this experiment is not good as previous LOSO pair matching experiment.

Table 6.4 : Our results on the CK+ dataset.

Methods	Top-1 Accuracy (%)	Top-2 Accuracy (%)
LBP & SVM (5-fold)	90.21	97.85
LBP & SVM (LOSO)	**92.35**	**98.16**

Figure 6.1 demonstrates the examples of the wrong prediction. For instance, the predicted expression is contempt, while real expression is surprised in (a), the predicted expression is happy, the real expression is contempt in (b). Lastly, the predicted expression is angry, while real expression is sad in (c). According to the confusion matrix that is given in Table 6.3, angry and sad facial expressions are confused with each other.

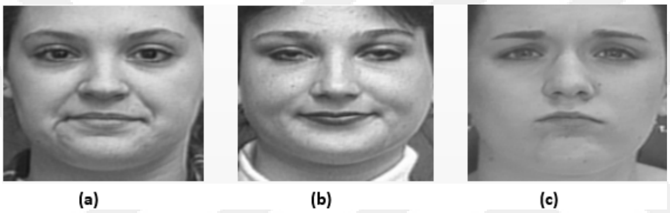


Figure 6.1 : Examples of the wrong prediction on the CK+ dataset.

Comparison with state-of-the-art methods

In Table 6.5 shows that comparison our result with state-of-art results on the CK+ 5.3 dataset. We would like to clearly identify that these methods have not used the same train set with our experiments. There are several constrained to compare our results with recent approaches. Briefly, the CK+ dataset contains sequence images and peak frames are labeled with six basic expressions and contempt for 327 images. Some studies use just six basic facial expressions, while some of them use neutral images instead of contempt expression by labeling the first frames of sequences as neutral. Defined seven facial expression labels in the dataset are used in this thesis.

Six basic facial expression and neutral have been used for recognition in [26]. Some neutral frames manually are labeled in this work, and there is no detail information about the number of images for neutral images. However, the dataset is split into train, validation, and test set in a subject-independent manner. Subject-independent train and test set were not used in most of studies.

Several data augmentation methods can be used in some previous works. In [28], last three frames are labeled with peak frame expression, thus 981 images were obtained. Dataset was divided into eight subsets in subject-independent and 8-fold cross-validation experiments are conducted. To predict facial expression for each image, highest accuracy from last three frames is chosen as a predicted facial expression. Because of these situations, we are not able to compare our results exactly with most of the recent results.

Table 6.5 : Performance comparison on the CK+ dataset in terms of seven expressions.

Methods	Accuracy (%)
2D Inception-ResNet (5-fold) [25]	85.77
3D Inception-ResNet (5-fold) [25]	89.50
3D Inception-ResNet + landmarks (5-fold) [25]	93.21
Inception-ResNet without CRF [26] (5-fold)	85.77
Inception-ResNet with CRF [26] (5-fold)	93.04
DTAN (10-fold) [62]	91.44
DTGN (10-fold) [62]	92.35
DTAGN (10-fold) [62]	97.25
DTGAN [63] (8-fold)	91.44
AUDN [21] (10-fold)	92.05
Going Deeper [22]	93.20
STM-ExpLet [16] (10-fold)	94.19
IACNN [28] (8-fold)	95.37
Ours (LOSO)	**92.35**

Six basic expression and neutral is used in a subject-independent manner by using 5-fold cross-validation for deep learning methods [22]. 3D-Inception ResNet with landmarks, 2D and 3D Inception-ResNet which have been proposed in [25]. In these experiments, seven expressions have been used in a subject-independent manner similar to our experiments.

In addition to seven expression that is defined on CK dataset, neutral which is collected first frames from sequences in [21]. Last three frames are also chosen and labeled with expression. Shortly, eight class classification is applied in a subject-independent manner. Hand-crafted features such as LBP, SIFT, HOG, Gabor are also used in [21], other accuracies are less than our results.

Results of the proposed methods are conducted with the same experimental setup [22, 25] is higher than our results around 1%, while the proposed methods with the higher results were applied to different experimental setup.

6.2 Results on the JAFFE Dataset

In this section results on the JAFFE dataset with and without pair matching formulation are given and compared with state-of-the-art results.

6.2.1 Facial expression recognition

This experiment was performed to analyze improvement in facial expression accuracy by using pair matching formulation like the CK+ dataset. LBP features and deep features were used in these experiments. The pre-trained VGG Face model was fine-tunned on the FER-2013 database. This model was utilized the feature extraction. Both features were combined and new feature vector dimension was obtained as 6987. After L2 normalization was applied to the feature vector, PCA was applied. 200-dimensional feature vector was found by latency is around %99. Table 6.6 shows that results by classifying the combination of the LBP and deep features in LOSO strategy without using pair matching formulation.

Table 6.6 : The confusion matrix of the LOSO experiments on the JAFFE dataset.

	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**	**Ne**
An	**66.67**	10.00	0	0	10.00	0	13.33
Di	13.79	**44.83**	6.90	0	31.03	0	3.45
Fe	3.12	6.25	**62.50**	0	15.62	0	12.50
Ha	0	0	0	**77.42**	0	0	22.58
Sa	9.68	6.45	12.90	3.23	**64.52**	0	3.23
Su	0	0	20.00	6.67	0	**63.33**	10.00
Ne	6.67	0	10.00	0	6.67	0	**76.67**

Table 6.7 shows that multi-class classification results based on subjects for seven expression. According to results, some subjects such as YM, KA, KL, KM can be classified with high accuracy, on the other hand, some subjects can not be classified easily.

Table 6.7 : Subject based facial expression accuracy on the JAFFE dataset.

Subject	Accuracy (%)
KA	82.61
KL	86.36
KM	86.36
KR	45.00
MK	80.95
NA	52.38
NM	40.00
TM	38.10
UY	42.86
YM	90.91
Average	**65.26**

6.2.2 Pair matching for facial expression recognition

According to test pair results for each image, each image has been classified with one of the seven expressions like the CK+ dataset. Our results can be shown in Table 6.8. After applying pair matching formulation, we achieved 96.24% accuracy on the JAFFE dataset. We observed an important improvement for facial expression recognition on JAFFE dataset. Uniform distribution of the number images based on facial expressions in the JAFFE database yields high improvement on facial expression accuracy by using pair matching when compared with the CK+ dataset.

Table 6.8 : Our results on the JAFFE dataset.

Methods	Accuracy (%)
LBP & SVM (LOSO)	64.32
LBP+VGG & SVM (LOSO)	65.26
LBP & SVM (LOSO - Pair Matching)	**96.24**

Comparison with state-of-the-art methods

Comparison of the facial expression accuracy on the JAFFE dataset has been reviewed in [64]. There are different approaches that dataset split into train and test set. We discussed the previous works with higher accuracy in LOSO strategy as ours experiment set, Performance comparison in terms of seven expressions are given in the Table 6.9.

Table 6.9 : Performance comparison on the JAFFE in terms of seven expressions.

Methods	**Accuracy** (%)
Dahmane et al. [30]	85.65, 86.69
Buciu et al. [31]	90.34
Liu et al. [24]	91.80
Ours	**96.24**

Liu et al. [24] have been conducted experiments in LOSO strategy for seven experiments similar to our experimental setup. Buciu et al. [31] conducted several experiments with the combination of the different hybrid systems. Gabor wavelets are combined with SVM has been pointed as the best recognition system and achieved better results than other proposed methods in [31]. Dahmane et al. [30] used SVM with Linear and RBF kernel, and achieved 85.65% and 86.69%, respectively. As shown in Table 6.9, we achieved the higher accuracy when compared our results with state-of-the-art results.

6.3 Facial Expression Recognition in the Wild

In this section, several experiments were conducted on the wild datasets which are the FER-2013 and the SFEW datasets. One of the state-of-the-art deep CNN model was used for fine-tuning. The pre-trained VGG-Face [12] CNN model on the face images was fine-tuned on the FER-2013 dataset to learn facial expression recognition task. The FER-2013 dataset contains sufficient training images for fine-tuning, although the number of training examples in this dataset is not enough for training from scratch. The previous studies show that transferring a pre-trained CNN model has better performance compared to training a task-specific CNN model from scratch when only limited data are available for the specific task. This way, domain adaptation was applied to the pre-trained VGG-Face model on facial expression images. Caffe deep learning framework [65] was used for our experiments.

During performing fine-tuning, parameters were initialized with parameters of the pre-trained VGG-Face CNN model. The learning rate of last fully connected layers was changed with ten times. This is the common approach for fine-tuning in order to learn the task-specific part for classification instead of learning common features from early layers.

In addition, fine-tuned CNN model on the FER-2013 dataset was used as feature extractor for the JAFFE dataset.

As mentioned dataset explanation in Section 5.1, the FER-2013 dataset has three parts: private, public and training which corresponds test, validation and train set. These parts were used as given in this study.

6.3.1 Results on the FER-2013 dataset

The confusion matrix on the FER-2013 dataset for seven expressions is given in Table 6.10. According to results, "sad and neutral", "fear and angry" were confused with each other. As shown in 6.11, we achieved higher accuracy than human accuracy [15] on this dataset.

Table 6.10 : The confusion matrix on the FER-2013 dataset.

	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**	**Ne**
An	**65.58**	1.22	10.18	3.26	11.41	1.43	6.92
Di	12.73	**76.36**	5.45	0	3.64	1.82	0
Fe	10.80	0.57	**53.03**	2.84	17.99	7.58	7.20
Ha	1.59	0	1.71	**89.53**	2.05	1.93	3.19
Sa	7.24	0.17	11.62	4.71	**59.76**	0.84	15.66
Su	1.44	0	7.69	4.09	0.96	**84.62**	1.20
Ne	4.47	0.32	5.43	3.51	14.54	1.60	**70.13**

Results are reported on the private set for the FER-2013 dataset in Table 6.11. [32] is the winner of ICML 2013 Challenges on the FER-2013 dataset [33] that propose a CNN is similar to AlexNet [40], but SVM loss function has been used instead of Softmax function. We achieved a higher result than the winner of the challenge [32].

Table 6.11 : Performance comparison on the FER-2013 dataset in terms of seven expressions.

Methods	Accuracy (%)
Human Accuracy [15]	65 ± 5
Mollahosseini et al. [22]	66.4
Tang, 2013 [32]	71.2
Ours	**71.8**

6.3.2 Results on the SFEW dataset

As stated the previous section, the SFEW dataset is wild facial expression dataset which has static images from AFEW dataset. Fine-tuned VGG Face model on the FER-2013 dataset was used to fine-tune on this dataset. Dataset is split into train, validation, and test set in [14]. Table 6.12 indicates the confusion matrix on the SFEW validation set for seven expressions.

Table 6.12 : The confusion matrix on the SFEW dataset.

	An	**Di**	**Fe**	**Ha**	**Sa**	**Su**	**Ne**
An	**62.34**	2.60	2.60	12.99	2.60	3.90	12.99
Di	4.35	**8.70**	0	21.74	21.74	17.39	26.09
Fe	30.43	4.35	**10.87**	4.35	10.87	17.39	21.74
Ha	22.22	0	0	**72.22**	1.39	0	4.17
Sa	12.33	2.74	5.48	6.85	**46.58**	8.22	17.81
Su	19.64	1.79	3.57	7.14	17.86	**37.50**	12.50
Ne	9.52	7.14	4.76	8.33	8.33	4.76	**57.14**

In Table 6.13, results on the SFEW dataset are compared to previous studies. The SFEW dataset includes high and low-resolution images from close to real-world images, recognition becomes a more complex when compared with lab-controlled datasets [14]. In [14], the baseline classification accuracies are found as given in Table 6.13. In [29], four different methods based on deep metric learning and combining loss functions for identity-aware facial expression recognition have been proposed. Our results are better than baseline results, while Liu et al. [29] achieved better results than ours.

Table 6.13 : Performance comparison on the SFEW dataset in terms of seven expressions.

Methods	Accuracy (%)
Liu et al. [29]	49.77, 50.75, 53.36, 54.19
Mao et al. [34]	44.7
LPQ & SVM [14]	43.71
PHOG & SVM [14]	46.28
Ours	**48.72**

7. CONCLUSIONS

In this thesis, we proposed a pair matching formulation to annotate the large datasets and increase the facial expression accuracy on the small dataset. The first aim of this problem definition is to be able to define the weather two unlabeled facial expressions are the same or different each other. Another aim of this thesis was to increase facial expression recognition by using pair matching. According to pair matching results, facial expressions were predicted and facial expression accuracy was higher than usual classification results.

The CK+ and the JAFFE datasets which are controlled dataset and commonly used for facial expression recognition were chosen for conducting experiments. Our baseline approach to providing this pair matching formulation is uniform LBP for feature extraction and SVM for classification. Our results were not performed better than other state-of-the-art methods. However, we showed that our proposed pair matching approach is improved the facial expression recognition. Since the JAFFE dataset has the uniform distribution for the frequency of images for each emotion, recognition accuracy on the JAFFE dataset has more improvement than the CK+ dataset. In addition to pair matching experiments, we conducted extensive experiments for facial expression recognition in the wild. The pre-trained CNN model on face images was chosen to fine-tune on the FER2013 and the SFEW datasets. Our results showed that facial expression recognition in the wild is a challenging task when compared with the facial expression recognition in the controlled environment. Facial expression accuracy is increased in the wild datasets by using deep learning techniques.

For the future work, we are planning to use more deep learning methods such as embedding learning and different kind of loss function to be able to define and learn relations between for all possible combination for every class/expression. In addition, metric learning algorithms can be used for pair matching classification.

REFERENCES

[1] **Ekman, P.**, **Friesen, W.V. and Hager, J.C.** (1978). Facial action coding system (FACS): a technique for the measurement of facial movement, *Consulting Psychologists, San Francisco*, *22*.

[2] **Ekman, P. and Friesen, W.V.** (1971). Constants across cultures in the face and emotion., *Journal of Personality and Social Psychology*, *17*(2), 124.

[3] **Huang, G.B.**, **Ramesh, M.**, **Berg, T. and Learned-Miller, E.** (2007). Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments, **Technical Report 07-49**, University of Massachusetts, Amherst.

[4] **Fabian Benitez-Quiroz, C.**, **Srinivasan, R. and Martinez, A.M.** (2016). EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.5562–5570.

[5] **Lucey, P.**, **Cohn, J.F.**, **Kanade, T.**, **Saragih, J.**, **Ambadar, Z. and Matthews, I.** (2010). The Extended Cohn-Kanade Dataset (ck+): A complete dataset for action unit and emotion-specified expression, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, pp.94–101.

[6] **Liu, C. and Wechsler, H.** (2002). Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition, *IEEE Transactions on Image processing*, *11*(4), 467–476.

[7] **Dalal, N. and Triggs, B.** (2005). Histograms of Oriented Gradients for Human Detection, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, IEEE, pp.886–893.

[8] **Lowe, D.G.** (2004). Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision*, *60*(2), 91–110.

[9] **Ojala, T.**, **Pietikäinen, M. and Harwood, D.** (1996). A comparative study of texture measures with classification based on featured distributions, *Pattern Recognition*, *29*(1), 51–59.

[10] **Bay, H.**, **Tuytelaars, T. and Van Gool, L.** (2006). Surf: Speeded up robust features, *In European Conference on Computer Vision*, 404–417.

[11] **Mase, K.** (1991). Recognition of facial expression from optical flow, *IEICE transactions (E)*, *74*, 3474–3483.

[12] **Parkhi, O.M.**, **Vedaldi, A. and Zisserman, A.** (2015). Deep Face Recognition., *In BMVC*, volume 1, p. 6.

[13] **Lyons, M.**, **Akamatsu, S.**, **Kamachi, M. and Gyoba, J.** (1998). Coding facial expressions with gabor wavelets, *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, IEEE, pp.200–205.

[14] **Dhall, A.**, **Goecke, R.**, **Lucey, S. and Gedeon, T.** (2011). Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark, *In Proceedings of the IEEE International Conference on Computer Vision Workshops*, IEEE, pp.2106–2112.

[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] **Liu, M.**, **Shan, S.**, **Wang, R. and Chen, X.** (2014). Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1749–1756.

[17] **Scovanner, P.**, **Ali, S. and Shah, M.** (2007). A 3-dimensional sift descriptor and its application to action recognition, *In Proceedings of the 15th ACM international conference on Multimedia*, ACM, pp.357–360.

[18] **Wang, L.**, **Qiao, Y. and Tang, X.** (2013). Motionlets: Mid-level 3d parts for human motion recognition, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.2674–2681.

[19] **Zhao, G. and Pietikainen, M.** (2007). Dynamic texture recognition using local binary patterns with an application to facial expressions, *Pattern Analysis and Machine Intelligence,IEEE transactions on,*, *29*(6), 915–928.

[20] **Klaser, A.**, **Marszałek, M. and Schmid, C.** (2008). A spatio-temporal descriptor based on 3d-gradients, *In BMVC 2008, 19th British Machine Vision Conference*, British Machine Vision Association, pp.275–1.

[21] **Liu, M.**, **Li, S.**, **Shan, S. and Chen, X.** (2013). Au-aware deep networks for facial expression recognition, *In Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, IEEE, pp.1–6.

[22] **Mollahosseini, A.**, **Chan, D. and Mahoor, M.H.** (2016). Going deeper in facial expression recognition using deep neural networks, *In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, pp.1–10.

[23] **Szegedy, C.**, **Liu, W.**, **Jia, Y.**, **Sermanet, P.**, **Reed, S.**, **Anguelov, D.**, **Erhan, D.**, **Vanhoucke, V. and Rabinovich, A.** (2015). Going deeper with

convolutions, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1–9.

[24] **Liu, P.**, **Han, S.**, **Meng, Z. and Tong, Y.** (2014). Facial expression recognition via a boosted deep belief network, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1805–1812.

[25] **Hasani, B. and Mahoor, M.H.** (2017). Facial Expression Recognition Using Enhanced Deep 3D Convolutional Neural Networks, *arXiv preprint arXiv:1705.07871*.

[26] **Hasani, B. and Mahoor, M.H.** (2017). Spatio-Temporal Facial Expression Recognition Using Convolutional Neural Networks and Conditional Random Fields, *IEEE International Conference on Automatic Face and Gesture Recognition Workshop*.

[27] **Szegedy, C.**, **Ioffe, S.**, **Vanhoucke, V. and Alemi, A.A.** (2017). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, *AAAI*, pp.4278–4284.

[28] **Meng, Z.**, **Liu, P.**, **Cai, J.**, **Han, S. and Tong, Y.** (2017). Identity-aware convolutional neural network for facial expression recognition, *In Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, IEEE, pp.558–565.

[29] **Liu, X.**, **Kumar, B.V.**, **You, J. and Jia, P.** (2017). Adaptive Deep Metric Learning for Identity-Aware Facial Expression Recognition, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, pp.522–531.

[30] **Dahmane, M. and Meunier, J.** (2014). Prototype-based modeling for facial expression analysis, *IEEE Transactions on Multimedia*, *16*(6), 1574–1584.

[31] **Buciu, I.**, **Pitas, I.** *et al.* (2003). ICA and Gabor representation for facial expression recognition, *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 2, IEEE, pp.II–855.

[32] **Tang, Y.** (2013). Deep learning using linear support vector machines, *arXiv preprint arXiv:1306.0239*.

[33] **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.

[34] **Mao, Q.**, **Rao, Q.**, **Yu, Y. and Dong, M.** (2017). Hierarchical Bayesian Theme Models for Multipose Facial Expression Recognition, *IEEE Transactions on Multimedia*, *19*(4), 861–873.

[35] **LeCun, Y.** (1989). Generalization and network design strategies, **Technical Report CRG-TR-89-4**, University of Toronto.

[36] **LeCun, Y.**, **Bottou, L.**, **Bengio, Y. and Haffner, P.** (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, *86*(11), 2278–2324.

[37] **Goodfellow, I.**, **Bengio, Y. and Courville, A.** (2016). *Deep Learning*, MIT Press, http://www.deeplearningbook.org.

[38] **Zhou, Y. and Chellappa, R.** (1988). Computation of optical flow using a neural network, *IEEE International Conference on Neural Networks*, volume1998, pp.71–78.

[39] **Srivastava, N.**, **Hinton, G.E.**, **Krizhevsky, A.**, **Sutskever, I. and Salakhutdinov, R.** (2014). Dropout: a simple way to prevent neural networks from overfitting., *Journal of Machine Learning Research*, *15*(1), 1929–1958.

[40] **Krizhevsky, A.**, **Sutskever, I. and Hinton, G.E.** (2012). Imagenet classification with deep convolutional neural networks, *Advances in Neural Information Processing Systems*, pp.1097–1105.

[41] **Russakovsky, O.**, **Deng, J.**, **Su, H.**, **Krause, J.**, **Satheesh, S.**, **Ma, S.**, **Huang, Z.**, **Karpathy, A.**, **Khosla, A.**, **Bernstein, M.** *et al.* (2015). Imagenet large scale visual recognition challenge, *International Journal of Computer Vision*, *115*(3), 211–252.

[42] **Simonyan, K. and Zisserman, A.** (2015). Very deep convolutional networks for large-scale image recognition, *In International Conference on Learning Representations (ICLR)*.

[43] **He, K.**, **Zhang, X.**, **Ren, S. and Sun, J.** (2016). Deep residual learning for image recognition, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778.

[44] **Deng, J.**, **Dong, W.**, **Socher, R.**, **Li, L.J.**, **Li, K. and Fei-Fei, L.** (2009). Imagenet: A large-scale hierarchical image database, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp.248–255.

[45] **Wolf, L.**, **Hassner, T. and Maoz, I.** (2011). Face recognition in unconstrained videos with matched background similarity, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp.529–534.

[46] **LeCun, Y.**, **Bengio, Y. and Hinton, G.** (2015). Deep learning, *Nature*, *521*(7553), 436–444.

[47] **Yosinski, J.**, **Clune, J.**, **Bengio, Y. and Lipson, H.** (2014). How transferable are features in deep neural networks?, *Advances in Neural Information Processing Systems*, pp.3320–3328.

[48] **Ojala, T.**, **Pietikainen, M. and Maenpaa, T.** (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(7), 971–987.

[49] **Ahonen, T.**, **Hadid, A. and Pietikäinen, M.** (2004). Face recognition with local binary patterns, *In European Conference on Computer Vision (ECCV)*, Springer, pp.469–481.

[50] **Jolliffe, I.T.** (2002). *Principal Component Analysis (Second ed.)*, Springer.

[51] **Bishop, C.M.** (2006). *Pattern Recognition and Machine Learning*, Springer.

[52] **Alpaydın, E.** (2014). *Introduction to Machine Learning, Third edition*, MIT Press.

[53] **Raschka, S.** (2015). *Python machine learning*, Packt Publishing Ltd.

[54] **Schölkopf, B. and Smola, A.J.** (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT Press.

[55] **Chang, C.C. and Lin, C.J.** (2011). LIBSVM: A library for support vector machines, *ACM Transactions on Intelligent Systems and Technology*, *2*, 27:1–27:27, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[56] **Gross, R.**, **Matthews, I.**, **Cohn, J.**, **Kanade, T. and Baker, S.** (2010). Multi-PIE, *Image and Vision Computing*, *28*(5), 807–813.

[57] **Pantic, M.**, **Valstar, M.**, **Rademaker, R. and Maat, L.** (2005). Web-based database for facial expression analysis, *IEEE Conference on Multimedia and Expo*, IEEE, pp.5–pp.

[58] **Mavadati, S.M.**, **Mahoor, M.H.**, **Bartlett, K.**, **Trinh, P. and Cohn, J.F.** (2013). DISFA: A Spontaneous Facial Action Intensity Database, *IEEE Transactions on Affective Computing*, *4*(2), 151–160.

[59] **Lucey, P.**, **Cohn, J.F.**, **Prkachin, K.M.**, **Solomon, P.E. and Matthews, I.** (2011). Painful data: The UNBC-McMaster shoulder pain expression archive database, *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, IEEE, pp.57–64.

[60] **Dhall, A.** *et al.* (2012). Collecting large, richly annotated facial-expression databases from movies, *IEEE Multimedia*.

[61] **Ekman, P.**, **Friesen, W. and Hager, J.** (2002). Facial Action Coding System: Research nexus, *Network Research Information, Salt Lake City, UT, USA*.

[62] **Jung, H.**, **Lee, S.**, **Yim, J.**, **Park, S. and Kim, J.** (2015). Joint fine-tuning in deep neural networks for facial expression recognition, *In Proceedings of the IEEE International Conference on Computer Vision*, pp.2983–2991.

[63] **Jung, H.**, **Lee, S.**, **Yim, J.**, **Park, S. and Kim, J.** (2015). Joint fine-tuning in deep neural networks for facial expression recognition, *In Proceedings of the IEEE International Conference on Computer Vision*, pp.2983–2991.

[64] **Mery, D.**, **Zhao, Y. and Bowyer, K.** (2016). On accuracy estimation and comparison of results in biometric research, *In Biometrics Theory, Applications and Systems (BTAS), 2016 IEEE 8th International Conference on*, IEEE, pp.1–8.

[65] **Jia, Y.**, **Shelhamer, E.**, **Donahue, J.**, **Karayev, S.**, **Long, J.**, **Girshick, R.**, **Guadarrama, S. and Darrell, T.** (2014). Caffe: Convolutional Architecture for Fast Feature Embedding, *arXiv preprint arXiv:1408.5093*.

CURRICULUM VITAE

Name Surname: Deniz Engin

Place and Date of Birth: Çanakkale, Turkey - 03/12/1991

E-Mail: enginde@itu.edu.tr

EDUCATION:

- **B.Sc.:** Istanbul Technical University
- **M.Sc.:** Istanbul Technical University

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2016 -2017: ITU scholarship as a project assistance under the scope of "TUBITAK project no. 113E067: Artificial Vision For Assisting Visually Impaired in Social Interactions"

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS

- **D. Engin**, H. K. Ekenel,"Facial Expression Pair Matching" *IEEE Signal Processing and Communications Applications Conference (SIU)*, May 15-18, 2017 Antalya, Turkey.

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS

- **D. Engin**, B. Ors, “Implementation of Enigma Machine Using Verilog on an FPGA”, *9th International Conference on Electrical and Electronics Engineering (ELECO)*, 2015.