T.C.

KIRŞEHİR AHİ EVRAN ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

İLERİ TEKNOLOJİLER ANABİLİM DALI

DERİN ÖĞRENME TABANLI GELİŞTİRİLMİŞ YÜZ İFADESİ TANIMA SİSTEMİ

KARRAR ISMAEL MOHAMMED ALLAW

YÜKSEK LİSANS TIZI

KIRŞEHİR / 2022

T.C.
KIRSEHIR AHI EVRAN UNIVERSITY
INSTITUTE OF SCIENCES
DEPARTMENT OF ADVANCED
TECHNOLOGIES

DEEP LEARNING BASED ADVANCED FACIAL EXPRESSION RECOGNITION SYSTEM

KARRAR ISMAEL MOHAMMED ALLAW

Master's Thesis

Supervised by
Asst. Prof. Dr. MUSTAFA YAĞCI

KIRŞEHİR / 2022

Bu çalışma ……. tarihinde aşağıdaki jüri tarafından …… Anabilim Dalı, ……Programında Yüksek Lisans / Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Kırşehir Ahi Evran Üniversitesi
……Fakültesi

……
Kırşehir Ahi Evran Üniversitesi
…… Fakültesi

Prof. Dr. …….
Kırşehir Ahi Evran Üniversitesi
…… Fakültesi

Prof. Dr. …….
Kırşehir Ahi Evran Üniversitesi
…… Fakültesi

Prof. Dr. …….
…….. Üniversitesi
……. Fakültesi

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

KARRAR ISMAEL MOHAMEED

20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, Kırşehir Ahi Evran Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

ÖNSÖZ

Yüksek Lisansa / Doktoraya başlamamda ve yüksek lisans / doktora ders sürecinde kendisini tanıdığım günden bu yana gösterdiği sakin ve sabırlı hali ile her zaman bana örnek olmasının yanı sıra bir bilim adamının nasıl çalışması gerektiğini kendisinden öğrendiğim değerli danışmanım Doç. Dr. Mustafa YAĞCI'e büyük bir içtenlikle teşekkür ederim. Tezimin her aşamasında gerek sorularımla gerekse alt ayda bir yapılan tez izleme komitesi sunumlarında tezin şekillenmesinde ve nihai hale gelmesinde katkıları olan değerli jüri üyelerim Doç. Dr. Nezih ÖNAL ve Dr. Öğr Üyesi AKSU'e teşekkürlerimi içtenlikle sunarım.

Son olarak eğitim hayatım boyunca bana her türlü maddi ve manevi desteği sunan çok kıymetli aileme sonsuz şükranlarımı sunarım.

Haziran, 2022 KARRAR ISMAEL MOHAMEED

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	Explanation
FER	:facial expression recognition
KNN	:k-nearest neighbors
SVM	:Support Vector Machine
HCI	:Human-computer interaction
AI	:artificial intelligence
MDD	:Major Depressive Disorder
AGI	:Artificial General Intelligence
ML	:Machine learning
ESD	:Energy Spectral Density
ANN	:Artificial Neural Networks
HCA	:Hierarchical Cluster Analysis
CNN	:Convolutional Neural Network
FC	**:**fully-connected layers
MLP	:Multi-layer perceptron
SGD	:Stochastic gradient descent
ReLU	:Rectified linear unit
GUI	:Graphical User Interface
CK	:Cohn-Kanade Facial Expression Database
JAFFE	:Japanese Female Facial Expression

ÖZET

YÜKSEK LİSANS TIZI

DERİN ÖĞRENME TABANLI GELİŞTİRİLMİŞ YÜZ İFADESİ TANIMA SİSTEMİ

KARRAR ISMAEL MOHAMMED ALLAW

Kırşehir Ahi Evran Üniversitesi

Fen Bilimleri Enstitüsü

İleri Teknolojiler Anabilim Dalı

Danışman: Doç. Dr. Mustafa Yağcı

Yüz duygularının ifadesini tanıma, insanların yüz duygularının yüzlerindeki ifadelere göre sınıflandırılmasını içeren bir araştırma alanıdır. Akıllı insan-bilgisayar etkileşimi, biyometrik güvenlik, robotik ve depresyon, otizm için klinik tıp ve ruh sağlığı sorunları gibi birçok farklı uygulamada kullanılabilir. Bu tez, yüz ifadesi tanıma (FER) için ileri teknikleri araştırır ve analiz eder ve pratik uygulamalar için bir zeka sistemleri geliştirir. Bu çalışmada, FER doğruluğunu artırmak için birkaç derin öğrenme tabanlı çerçeve geliştirilmiştir. Belirli katmanlarda özellik çıkarma amacıyla üç ana tip önceden eğitilmiş ağ (AlexNet, GoogleNet ve SqueezeNet) kullanılır. Ayrıca, k-en yakın komşular (KNN), Destek Vektör Makinesi (SVM) ve Karar Ağacı sinir ağları algoritmaları, her tür önceden eğitilmiş ağ için sınıflandırıcı olarak kullanılır. Bu araştırmada, Yedi tür yüz ifadesini temsil eden çok sayıda görüntü içeren iki veri seti kullanılmıştır. SVM ile GoogleNet için elde edilen maksimum doğruluk 91.09, SVM 98.2766 ile SqueezeNet ve KNN ile AlexNet için yaklaşık %100'dür. Elde edilen sonuçlar, en iyi sınıflandırma sonuçları için yeniden eğitim zamanı ve kaynakları sağlayan öznitelik çıkarımı olarak önceden eğitilmiş bir ağ kullanabileceğimizi göstermektedir.

Haziran 2022, 76 Sayfa

Anahtar Kelimeler: Derin Öğrenme, Yüz İfadesi Tanıma, özellik çıkarma, önceden eğitilmiş CNN.

ABSTRACT

Master's Thesis

DEEP LEARNING BASED ADVANCED FACIAL EXPRESSION RECOGNITION SYSTEM

KARRAR ISMAEL MOHAMMED ALLAW

Kırsehir Ahi Evran University
Graduate School of Sciences and Engineering
Advanced Technologies Department

Supervisor: Doc. Dr. Mustafa Yağcı

Facial emotion expression recognition is a field of research that comprises the classification of face emotions of humans by expressions on their faces. It can be used in many different applications including intelligent human-computer interaction, biometric security, robotics and depression, and clinical medicine for autism, and mental health problems. This thesis explores and analysis advanced techniques for facial expression recognition (FER) and develops intelligence systems for practical applications. In this study, several deep learning-based frameworks have been developed to improve FER accuracy. Three main types of pre-trained networks (AlexNet, GoogleNet, and SqueezeNet) are utilized for feature extraction purposes at a certain layer. Moreover, k-nearest neighbors (KNN), Support Vector Machine (SVM), and Decision Tree neural networks algorithms are employed as a classifier for each type of pre-trained network. Two datasets are used in this research including a large number of images representing Seven types of facial expressions. The maximum accuracy obtained for GoogleNet with SVM is 91.09, SqueezeNet with SVM 98.2766, and AlexNet with KNN at about 100%. The results obtained indicate that we can use a pre-trained network as feature extraction which provides a pre-training time and resources for best classification results.

June 2022, 76 Pages

Keywords: Deep Learning, Facial Expression Recognition, feature extraction, pre-trained CNN.

1. INTRODUCTİON

Human-computer interaction (HCI) is playing an incrementally important part in people's daily lives. We have entered a time when paperwork does not need to be carried out by hand anymore and where many tasks may be completed from the comfort of one's own home using a computer. Biometric data and emotion detection are the two key topics of HCI study. Several applications can profit from them in a variety of industries, including commercial, medical, and consumer. Emotions have an impact on reasoning, perception, and human interaction. Emotions may be characterized in a variety of ways, which are referred to as distinct categories of emotions (Fear, Sad, Anger, Happy, Surprise, Disgust, and Neutral). A dimensional field, such as valence, arousal, and others, is another technique to explain emotions [1].

The practice of anticipating a single emotion or a point in the arousal-dimensional emotion space is known as emotion detection. This is accomplished by artificial intelligence (AI) systems learning various characteristics and patterns from recorded data of multiple modalities. Nevertheless, because their primary purpose is to search for certain emotions, some AI systems ignore some of the emotional states that underlie human behaviors and speech. As a result, systems are being designed to comprehend more emotions simple emotions, and documenting one's emotional state might help the system bridge the gap and enhance its reaction [2].

GSR, heart rate, EEG, and other factors such as facial expressions, speech, bodily movement, touch gestures, and GSR Biosensors such as cameras and fingerprints can be used to detect emotion. However, the most significant technique of expressing a person's feelings is through facial expressions. Because of the expressive activity of the face, they are intimately related to emotional representations, with the ability to capture rich material in multidimensional views. Facial expressions give communication indicators that can comprehend meaning, mood, and emotions concurrently. These expressions are a crucial topic of research for HCI since they are utilized as a manner of engaging with people to represent their moods and feelings [3].

Facial Expressions are crucial tools for expressing an individual's emotional response and/or condition throughout the daily routine. Many different types of expressions a person may

make, and each of them has a set of components that regulate the depth of the expressions. Intentions, action inclinations, assessments, other cognition, neuromuscular and physiological changes, expressive actions, and subjective sensations are all examples of these. These elements induce face muscles to move, resulting in a visual expression that others may perceive. Among the wide variety of human feelings, there are seven primary facial expressions: Sad, Happy, Surprise, Disgust, Angry, Fear, and neutral, as demonstrated in Figure 1.1[4]. several different emotions are related to facial expressions, although they are essentially minor variants of fundamental expressions. The goal of a new field of current research on emotional computation is to attempt and mathematically simulate Major Depressive Disorder (MDD) using personal facial expressions [5].

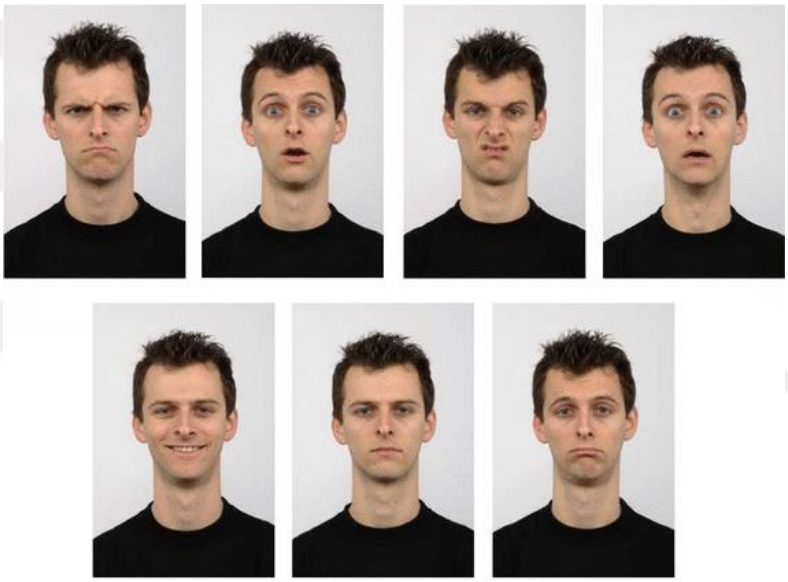


Figure1.1. Different seven facial expressions [4]

Before a system can learn to recognize facial expressions, an expert must first view and classify examples. However, a professional's accuracy is not always assured, as humans have difficulties understanding complex emotions like expectancy or remorse, which restricts robots' capabilities [6].

Any computer program that performs intelligent tasks is referred to as artificial intelligence. It might be a series of if-then statements or a complicated statistical model. AI can touch on anything from a chess-playing computer program to a voice-recognition and classification system such as Google talk. Narrow AI, Artificial General Intelligence (AGI), and superintelligent AI are the three major categories in which the technology may be classified. Machine learning is a portion of artificial intelligence. Machines take data and 'learn' by

themselves, according to the hypothesis. It is currently the most likely tool for enterprises in the AI pool. Machine learning (ML) systems excel in facial identification, speech recognition, and a variety of other tasks by quickly applying knowledge and training from vast datasets.

Machine learning includes deep learning as a portion. Deep learning artificial neural networks are a collection of algorithms that provide unprecedented levels of accuracy for a variety of issues, including image identification, sound recognition, recommender systems, and so forth [7].

In this study, a deep learning-based framework is presented to distinguish between different facial expressions which are Fear, Anger, Sad, Disgust, Surprise, Happy, and Neutral. The results obtained experimentally from different techniques are compared and discussed.

The AI and deep learning algorithms were used to identify facial expressions. Different AI algorithms were used for this purpose. In each algorithm, a data set of different digital images were utilized to train the network for the seven mentioned images. Then, a validation process is used to check whether the network can identify the trained images or not. After the validation succeeded, the process of testing starts, where different images are used now to be entered into the system and check the facial expression.

The AI algorithms that are used are support vector machine (SVM), decision tree, and k-nearest neighbors (KNN).

1.1. Problem Statement

Facial Expressions are important tools used to convey a person's emotional response and/or state during their daily activities. The process of detecting the facial expression is a complicated task for a machine as these expressions may have similarities that limit the computer system from differentiating between two or more of them. The classification system accuracy represents a big challenge in these types of systems. So it will be a promising research outcome if an AI system is created and could recognize different facial expressions.

1.2. Aim of the project

This work aims to use advanced AI techniques to improve the currently available systems that are related to human facial emotion detection using AI in deep learning. The research aims to explore and analyze the most popular techniques in deep learning.

1.3. Objectives

The use of facial expression recognition systems is now widespread in various areas, including security, analytical, predictive, educational, and political, so it is necessary to build an efficient computer model and system that has the ability to give accurate and quick results to identify a person's emotional features and these things are mainly reflected and clearly on a person's facial expressions. The overall objective of the thesis has been reached through several sub-objectives:

- Research and understand the techniques available for Facial Expression Recognition (FER) based on facial images in AI.
- Identifying facial areas that contribute significantly to each expression.
- Integrate cutting-edge deep learning techniques into a model that can predict a person's emotions.
- In this study, more than one technique will be applied and the result of each technique will be applied.
- Employing machine learning techniques and tools in all system stages to obtain the most efficient results and restructuring the neural networks based on the experiments performed to reach the optimum structure.

1.4. Contribution

There are multiple scientific contributions to this research:

- The presented method will make the (Alexnet, Googlenet, Squeezenet) the module is very useful in terms of speed because a simple signal function reduces the data to half or ½ with Energy Spectral Density (ESD), which is a very important value.
- Obtained features were classified by 3 classifiers Support Vector Machines, K-nearest neighbor, and Decision Trees. The results obtained with the K-nearest neighbor showed more efficient results compared to other classifiers and the studies presented in the literature.

1.5. Thesis Outlin

The thesis is divided into five chapters, as follows:

Chapter one includes a general explanation of the thesis its aims , objectives, and contribution. In chapter two theoretical information related to the subject as information about used algorithms such as decision trees, SVM, and KNN is presented, followed by previous literature studies. In chapter three, the methodology of the thesis is written followed by results in chapter four. The thesis conclusion is written in chapter five.

2. THEORETICAL BACKGROUND

2.1. Overview

Artificial Neural Networks (ANN) are structures that represent human neural networks that are designed to carry out computer learning. Artificial intelligence is the basic that some computer scientists are trying to reach using techniques like neural networks mimicking.
This chapter will provide a better understanding of artificial neural networks in general and convolutional neural networks in specific; in addition, literature studies of previous works are presented.

2.2. Artificial Neural Network

Artificial neural networks, which resemble human brain networks, are one of the most powerful computer learning techniques. Some computer scientists are trying to construct a subset of Artificial Intelligence via neural network imitation [8]. Analytical neural networks (ANNs) are distributed processors with a preference for collecting experimental data and making them available for further analysis. ANN is widely used as a replacement or response surface estimation model when dealing with multivariate and nonlinear modeling difficulties, such as function approximations and classification. An ANN architecture's description includes the number of inputs, outputs, neurons that are hidden, and hidden layers, as well as the hidden layers. According to the universal approximation theorem, a feed-forward network with a single hidden layer and a modest number of neurons may approximate continuous functions on compact subsets of Rn, where n is the number of inputs. An ANN with only one hidden layer is not always the most adaptive, fastest, or simplest to build. There are no universal criteria for identifying the appropriate ANN design given a set of input and output data (number of neurons and hidden layers). Figure 2.1[9]. shows the architecture of ANN.

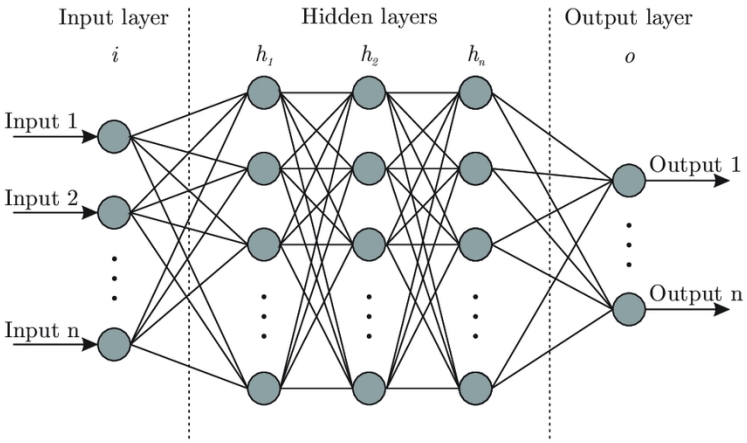


Figure 2.1. Artificial neural network architecture[9]

2.2.1. Machine Learning

Machine learning is a part of AI that can handle impossible or too time-consuming problems to address using "conventional" programming languages. An early example of machine learning is an email spam filter. Machine learning often outperforms earlier algorithms in terms of accuracy. Regardless of the various machine learning methods available, the data is perhaps more essential than the technique used. Data can have various problems, including inadequate data, poor data quality, erroneous data, missing data, irrelevantly sent info, duplicated values, etc [10].

2.2.1.1. Types of Machine Learning

Several primary forms of machine learning that you will come across (combinations of these are also possible) :

·Supervised learning

·Unsupervised learning

·Semi-supervised learning

Supervised learning implies that each data point in a dataset has a label that describes what it contains. The MNIST dataset, for example, has 28 × 28 PNG files, each containing one digit that is written by hand (i.e., 0 through 9 inclusive). All photographs with the digit zero have the label zero; all images with the digit one have one as a label. All other images are labeled according to the number displayed in those images.

Unsupervised learning, like with most clustering techniques, includes data without labels. Clustering is used in three standard unsupervised learning techniques: k-Means, Hierarchical Cluster Analysis (HCA), and Assumption Maximization.

Semi-supervised learning blends both previous types by classifying some data points while keeping others unlabeled. One approach is to utilize the labeled data to categorize (i.e.,

label) the unlabeled data before using a classification method.

2.2.1.2. Machine Learning Algorithms

Regression, Classification, and Clustering are three of the most often used machine learning methods. Regression uses supervised learning to predict numerical quantities. An example of regression work is predicting the value of a stock. Assuming that a stock will grow or decrease in value the following day is not the same as completing an assignment like this one (or any point in the future). Another example of regression work is estimating the value of a property based on data from a real estate database. Regression exercises are used in both of these exercises. In machine learning, regression techniques like as linear and extended linear regression are utilized. (In classical statistics, this is known as multivariate analysis.) Unsupervised learning techniques are used to predict categorical values via classification [11].

Detecting spamming attempts, fraud attempts, or finding the digit in an image file is all examples of classification tasks. Because the data has already been tagged in this scenario, comparing the label supplied to the given PNG's forecast is possible.

The following is a list of classification algorithms used in machine learning :

• Decision Tree, which is one tree
• Random Forests which are many trees
• k Nearest Neighbor)
• Logistic regression.
• Naïve Bayes
• SVM.

Several algorithms related to machine learning (such as KNN and random forests) can do both classification and regression. When it comes to SVMs, the scikit-learn implementation offers two APIs: SVC to classify and SVR to apply regression. The used model has been trained on a dataset and then utilized to generate a prediction for all mentioned approaches. On the other hand, a random forest comprises numerous independent trees (the user determines the value); each one initiates the prediction about the importance of a characteristic. To obtain the "final" forecast, take the mean or the mode (or apply some other computation) if the feature is numeric.

It is preferable to choose the mode (the most common class) as the outcome for categorical features, and if it is tied, pick one at random.

Decision Trees

Decision trees are a treelike form employed in another sort of classification approach. A data point's location in a generic tree is determined by logical reasoning. Consider the following situation: A dataset comprises a series of integers representing the ages of individuals, and it is always set to 50. This number is used as the tree's root, and any numbers less than the beginning number are added to the tree's left branch, while any numbers more significant than 50 are added to the right side of the tree. The numbers are 50, 25, 27, and 40. Then we may design a tree like this: In this example, the root node is 50, the left child of 50 is 25, the right child of 50 is 70, and the correct child of 20 is 40. Each new numeric value added to the dataset is examined at each node in the tree to determine which path to follow (left or right) [12].

Random Forests

Forests of Unknown Origin Multi-tree classification is an expansion of decision-tree classification (you specify the number). The average of the tree predictions is used if the data demands a numerical forecast. Data that includes a definite prediction establishes a tree's mode. Random forests aim to balance out losses with higher gains while trading like a financial portfolio's diversification strategy. If you want to know what's going to happen more often than a single tree can predict, you can use random forests to make predictions based on the consensus of the trees [13].

k-Nearest Neighbor (kNN)

The closest neighbor method assigns the class of most comparable labeled samples to unlabeled data based on their categorization features. Nearest neighbor approaches, despite their simplicity, are powerful. The ANN algorithm uses the closest neighbor technique for classification [14]. The user must provide the number of neighbors (k). For the ranking to be influential, it must be a single number with no more than equal votes. High k numbers, on the other hand, lower the method's efficiency. The error rate approaches the Bayesian error rate as K approaches infinity.

Nearest neighbor classifiers may also be used for numerical prediction, that is, to get an accurate value estimate. This approach is practical because it is simple and efficient, does not require any data assumptions, and the learning phase is quick. However, the requirement for a significant amount of memory, national characteristics, and further processing in the event of missing data can all be considered flaws, and the classification step is sluggish.

Finding the nearest neighbors of a sample, a distance function, or a formula that assesses similarity between two models is required. Distance may be calculated in a variety of ways. The spaces between Euclidean, Manhattan, and Minkowski are different [15].

An example of the nearest neighbor decision problem is shown in Figure 2.2. The new instance's class is searched in a two-class issue. It was decided that K = 8 would be the best option.

In this scenario, the eight closest examples to the new sample are chosen, and the class's maximum number of samples should be that class.

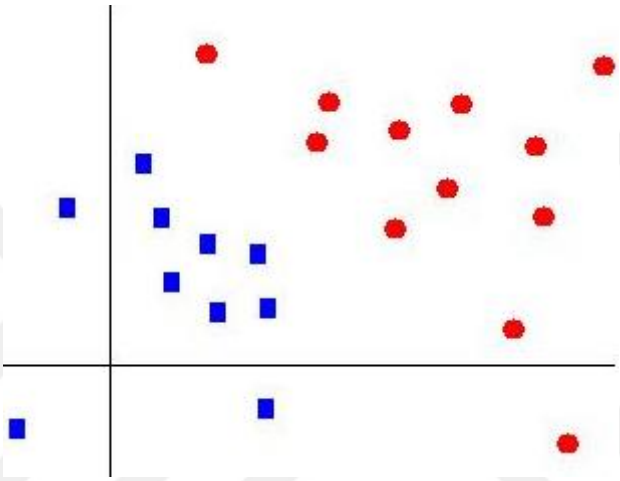


Figure 2.2. K-Nearest neighbor sample[15]

Logistic Regression

First developed as a classifier and linear model, logistic regression has evolved to include a binary output. The sigmoid function calculates probabilities in logistic regression, which works with numerous independent variables. The sigmoid activation function applies linear regression to accomplish binary classification, resulting in logistic regression. Logistic regression has applications in a wide range of disciplines. Machine learning, numerous medical professions, and social sciences are examples of such fields.

Logistic regression may assess a patient's risk of having a specific disease based on several characteristics seen in the patient. Engineering, marketing, and economics are some of the other disciplines that employ logistic regression. Binomial, multinomial, and ordinal are the three types of dependent variables (three or more outcomes for a dependent variable) and the three types of logistic regression (ordered dependent variables). Consider the following scenario: a dataset contains data from either class A, or it is from B class. If a new data point is given, logistic regression will tell whether it belongs in class A or B. Linear

regression, the same while, expects a numerical value, such as a stock's tomorrow’s value [10].

Naive Bayes

A Naive Bayes Classifier is a probabilistic classifier based on the Bayes theorem. It is expected that the NB classifier's properties are unaffected by the condition, and yet it still performs well even when this assumption is incorrect. If this assumption is valid, there are considerable savings in computation expenses, and the implementation is simple and linear. To top it all off, an NB classifier can be readily scaled up to handle larger datasets and deliver adequate results in most cases [16].

Main types of NB classifiers are Gaussian,MultinomialNB and Bernoulli naive bias.

SVM (Support Vector Machines)

Support Vector Machines employ a supervised machine learning technique to solve classification and regression issues. SVM may deal with data that is nonlinearly separable as well as data that is linearly separable. SVM transforms data using a method known as the kernel trick and then determines an appropriate boundary[17]. That transform entails increased dimensionality. The mentioned approach produces a separation of the altered data, after which a hyperplane may be found to divide the data into two groups. Using SVMs in classification jobs is more common than in regression challenges. Examples of SVM use cases are provided here :

- The classification of text tasks: assigning categories
- Spam detecting/sentiment analysis
- images: recognition based on aspect, classification based on color
- Recognizing digits that are written by hand.

Tradeoffs of SVMs

SVMs are highly recommended, but they come with compromises. SVMs provide several advantages.:

- The accuracy is high
- Works better on smaller, cleaner datasets
- Its efficiency can be increased because it just uses a fraction of training points.
- Can be used instead of cans when datasets are restricted
- Despite the capability of SVM, it captures more intricate interactions between data points.

There are some disadvantages of SVMs:

- Not good enough for large datasets: require high training time.
- SVMs, which have more factors than decision trees, are less effective on noisy datasets with overlapping classes [10].

2.2.2. Deep Learning

Machine learning refers to a computer's ability to do tasks without being explicitly programmed but acting and thinking like a machine. When it comes to activities like data extraction from images or videos, they lag significantly behind human abilities. Deep learning models have been rigorously built after the human brain to deal with these obstacles, making them ideal for this task. Data is shared between nodes (such as neurons) closely coupled using massive, multi-layered deep learning networks. A non-linear transformation of the data results, with the information becoming progressively esoteric. Figure 2.3 depicts the relationship between AI, machine learning, and deep learning. Convolutional Neural Network (CNN) is the most used Deep Learning approach for dealing with pictures. The theoretical backdrop of it will be explained in-depth in the next section [18].

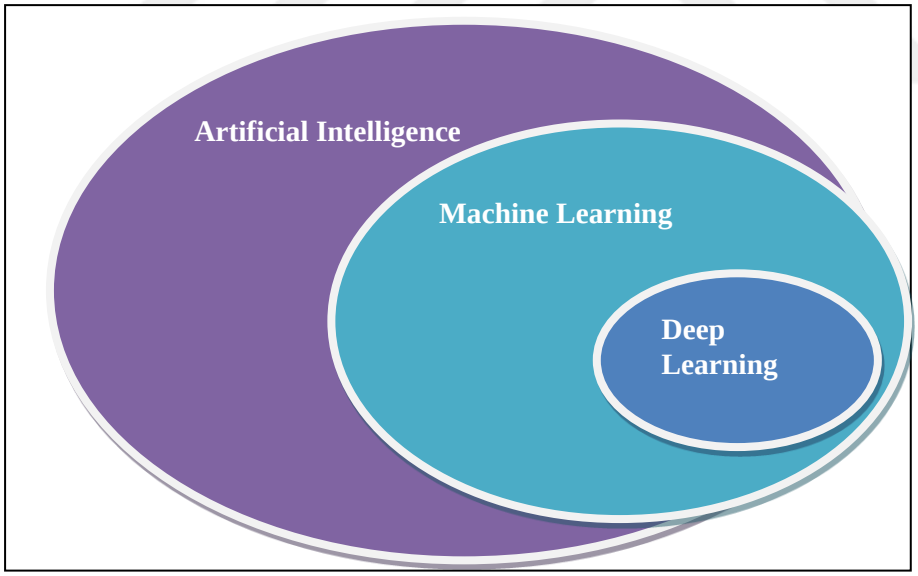


Figure 2.3. AI, Machine Learning, and Deep Learning

2.2.3. Convolutional Neural Network

Among deep learning designs for spatial data (or CNN or ConvNet), Convolutional Neural Networks (CNNs) is a multi-layer neural network that stands out. The visual perception of living beings informs the design of CNN. Even though AlexNet's world-record-breaking

performance in 2012 made it a household name, it was founded in 1980. With an increase in speed after 2012, CNN became the dominant player in several fields, such as natural language processing and computer vision. Compared to other ANNs with FC layers, the Convolutional Neural Network (CNN), also known as ConvNet, has remarkable generalizing potential because of its deep feed-forward architecture. More than previous FC-layer networks, it can learn and recognize abstract features of things more effectively, such as geographical data [14]. Input data may be used to teach a deep CNN model a variety of abstractions because of the network's various processing levels (such as an image). While the first layer learns and extracts information at a higher level, the second and third layers learn and extract data at a lower level. Figure 2.4 depicts the CNN conceptual model in its simplest form. Convolutional neural networks are more important in computer vision than other neural networks for various reasons. With the weight sharing feature, the network may be trained with fewer parameters, which minimizes the likelihood of overfitting and increases the possibility of generalization. At each stage of CNN's learning process, the classification and feature extraction layers work together to improve the model's accuracy and predictability. Using other neural networks instead of Convolutional Neural Networks will be more challenging to build an extensive network. Classification tasks, object identification, face recognition, speech recognition, automotive recognition and expression recognition, text recognition, and a slew of other computer vision-based applications have all seen encouraging outcomes thanks to CNN. An overview of CNN's major components and fundamental building parts is provided below.

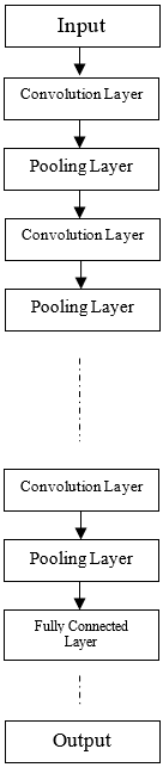


Figure 2.4. CNN Layers

2.2.3.1. Network Layers

As we previously stated, a CNN consists of several different components. In the next section, some of these architectural components and their roles in CNN's architecture will be discussed in further detail.

Convolutional Layer

The convolutional layer1 is the essential part of any CNN architecture. Convolutional kernels (also known as filters) are used with the input image (N-dimensional metrics) to create an output feature map. It is possible to visualize the weight of a kernel as a grid of discrete or integer values. At the beginning of the CNN model's training phase, all of a kernel's values are allocated randomly (various ways are available for initializing the weights). After a few training sessions, the consequences are fine-tuned, and the kernel learns to identify meaningful information. 2D filter is seen in Figure 2.5 [19].

0	1
-1	2

Figure 2.5. Example of 2D Filter [19]

Pooling Layer

Subsampling feature maps (produced following convolution operations) is done using the pooling layers, resulting in a smaller feature map. The most significant features (or information) in each pool stage are kept when the feature map is reduced. P pooling, like convolution, may be achieved by defining a specific size and stage of the process. Various pooling algorithms, such as max pooling, min pooling, average pooling, gated pooling, tree pooling, and so on, are employed in multiple pooling layers. A common pooling strategy is max pooling, the most common and widely utilized. The pooling layer's biggest drawback is that it has the potential to affect CNN's overall performance on occasion. As a result, the pooling layer of the CNN helps determine whether or not a specific feature exists in the given input picture [20].

Activation Functions (Non-Linearity)

Any activation function in a neural network-based model is to translate inputs to output values, with the input value formed by adding bias to the weighted sum of the neuron's input (if there is a bias). Alternately, to determine if a neuron will respond to some form of input, the activation function creates an output that does so. Non-linear activation layers (weighted layers such as convolutional and FC layers) have been employed in CNN architecture since the learnable layer. Because of the non-linearity activity in those layers, the CNN model may be able to learn more complex things and transfer inputs to outputs in a non-linear manner. The model needs a differentiable activation function to be trained by error back-propagation [20].

Fully-Connected Layer

Each layer's neuron is connected to every other layer's neuron through fully-connected layers (FC Layers), the last component of a CNN architecture (used for classification). The CNN architecture's last layer, the output layer (classifier), comprises just Fully-Connected layers. In some ways, FCLs resemble Multi-layer perceptron(MLP) neural networks in that

they are feed-forward artificial neural networks (ANNs). Final CNN output is produced by the final convolutional or pooling layer receiving a set of measurements (feature maps) from the FC layers shown in Fig 2.6 [21].

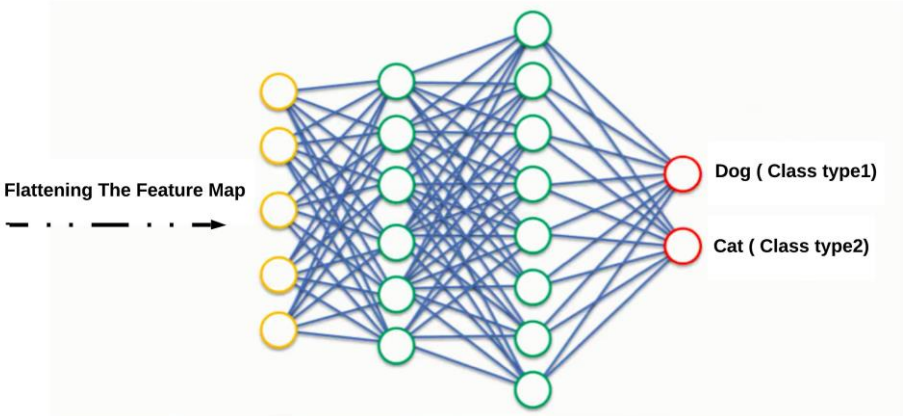


Figure 2.6. Final CNN[19]

Training of CNN

The fundamental concepts of convolutional neural networks (CNN) and the many critical components of CNN design have been previously discussed in detail. This section will outline the training or learning process of a CNN model employing specific concepts to reduce training time and improve model accuracy. The following are the stages of the training process: Enhancement and pre-processing of data. The parameters have been set up. CNN's uniformity. Select an optimizer from a list provided to you. There are extensive instructions for each of those stages in the next section.

Data Preprocessing and Augmenting Data

Data pre-processing is applying artificial modifications to a raw dataset to improve its cleanliness, feature density, learnability, and consistency. If it is not first entered into the CNN model, the data is pre-processed. An accurate convolutional neural network (CNN) is directly proportional to the amount of input used to train it, demonstrating that pre-processing is always critical to the model's performance. On the other hand, poor pre-processing might damage the model's performance. Enhancing training datasets is the primary purpose of data augmentation. Data samples from the training dataset were subjected to various processes to create new data samples (new versions) that could be utilized in the training process. Since only a small quantity of training data is available at

any given moment in most real-world demanding circumstances (e.g., medical datasets), data augmentation plays a significant role since a more proficient CNN model may be created with more training data samples. If you need a little help with your data manipulations, there are many solutions available. These approaches can be used alone or in combination to generate a wide range of possible outcomes from a single data sample.

Parameter Initialization

In some prominent CNN, there are billions or millions of parameters. Consequently, it must be appropriately started at the start of the training phase, as initializing of weight affects how fast and accurately the CNN model converges. This section will go through some of the most frequent parameter initialization procedures used in CNN: Setting all weights to 0 is the most straightforward way. This was an oversight because if we set the importance of all layers to 0, the output and gradients computed by each neuron in the network would be equal (during backpropagation). As a consequence, all weights would be updated at the same time. Consequently, neurons are similar, and no valuable characteristics are required [19].

Regularization to CNN

Generalization refers to a deep learning algorithm's ability to learn from fresh or previously unknown input drawn from the same distribution as the training data. Overfitting is the major obstacle to a CNN model's ability to generalize. Overfitting occurs when a model outperforms its training data but underperforms its test data (unseen data). An under-the-fitted model, on the other hand, works well on both trained and tested data because it has not had enough learning from the training data. Just fitted models are the name given to these models.

Figure 2.7 [22].attempts to demonstrate over, under, and just fitted models.

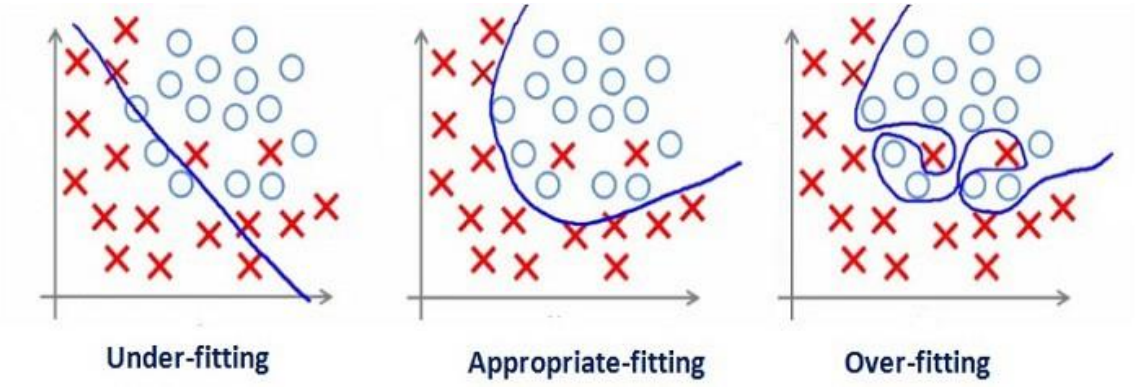


Figure 2.7. Fitting States[22]

2.3. Classification of images

It has been assumed that the input picture comprises a single item in image classification. Then we use CNN models to classify the image into one of the pre-selected target ones. In the coming sections, there are several popular CNN architectures (models) for image classification:

2.3.1. LeNet-5

LeNet-5 [23]. architecture was one of the earliest attempts at classifying handwritten numbers by a convolutional neural network. LeCun and colleagues first put out this idea in 1998. Two FC and three convolutional layers comprise the LeNet-5's five weighted (trainable) layer sets. Each of the first two convolution layers is followed by a max-pooling layer, while two connected layers follow the final convolution layer. The classifier, which can categorize up to 10 digits, is the last layer of those ultimately linked levels. The LeNet-5 design is shown in Figure 2.8.

• The MNIST digit dataset was used to train the LeNet-5.
• The activation function was sigmoid non-linearity.
• It employed a 20-epoch Stochastic gradient descent (SGD) learning technique.
• The momentum factor was set to 0.02.
• On the MNIST data set, it decreased the test error rate by 95%

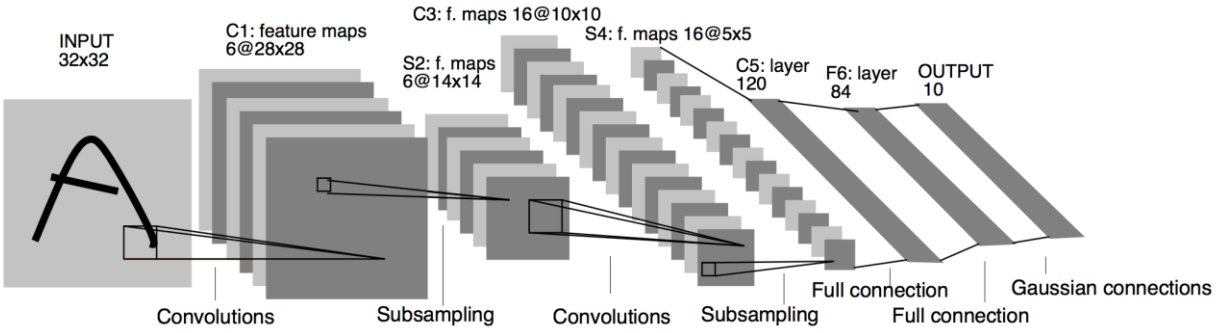


Figure 2.8. Architecture of LeNet[19]

2.3.2. Alexnet

An image classification model based on LeNet, AlexNet, was created by Krizhevky and colleagues in 2012. Rather than using convolutional or fully connected layers, it uses eight weighted (learnable) layers. For ImageNet data, the final output layer uses 1,000 units to classify incoming photos into one of a thousand classes. Figure 2.9 [19]. depicts the architecture of AlexNet. After each convolution and fully connected layer, AlexNet utilizes a non-linear activation function called a Rectified linear unit (ReLU). After each LRN and the final convolutional layers, it employed a max-pooling layer.

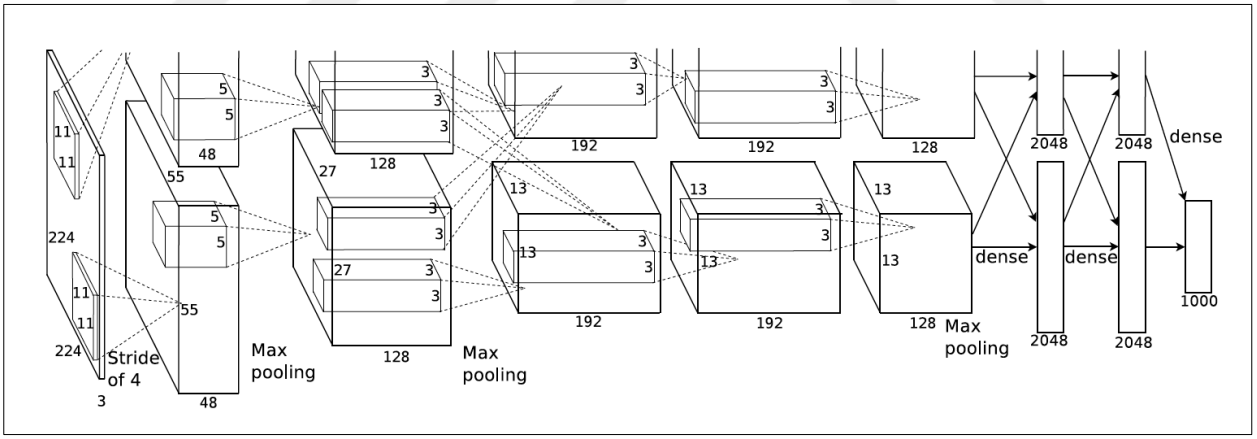


Figure 2.9. Architecture of LAlexnet[19]

- Many regularization methods, such as dropout and augmentation, are used to avoid overfitting since the number of weights is so large (learnable).
- It was trained using the Stochastic gradient descent (SGD) technique with the minimum batch size being 128, the weight decay being 0.0005, and the momentum factor being 0.90.
- over six days, two NVIDIA GTX 580 (each with 3 GB of RAM) were used to train AlexNet on the ImageNet dataset.

2.3.3. Googlenet

GoogLeNet, with its 22 layers, is a well-known deep convolutional neural network. The ImageNet [24]. and Places365 [25].[26]. datasets are used to train GoogleNet, a pre-trained network. Over a million images from ImageNet were used to prepare the network, which now can classify images into over a thousand distinct object categories, such as a mouse, keyboard, pencil, autos, and numerous animal species. An ImageNet-like dataset called Places 365 categorizes photos into one of the 365 categories. The network was also trained on this dataset. This network has developed several feature representations for a wide range of pictures. The technique uses a 224-by-224 input size for a three-channel view of the trained network as its input layer. The diagram below shows the typical GoogLeNet structure.

The table below depicts the conventional GoogLeNet architecture.

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Figure 2.10. GoogLeNet configuration details[27]

The architectural structure of classifiers is as follows:

- A 5×5 and stride of an average pooling layer.
- A convolutional layer of 1×1 with 128 filters ReLU activation for dimension reduction.
- A 1025 fully connected layer with ReLU activation.
- Dropout layer, Regularization with dropout ratio = 0.7
- A 1000 classes output layer with softmax classifier.

This network architecture accepts images of RGB color channels size 224 x 224. All the

convolutions layers inside this architecture have Rectified Linear Units (ReLU) as their activation functions. Challenges in face recognition systems.

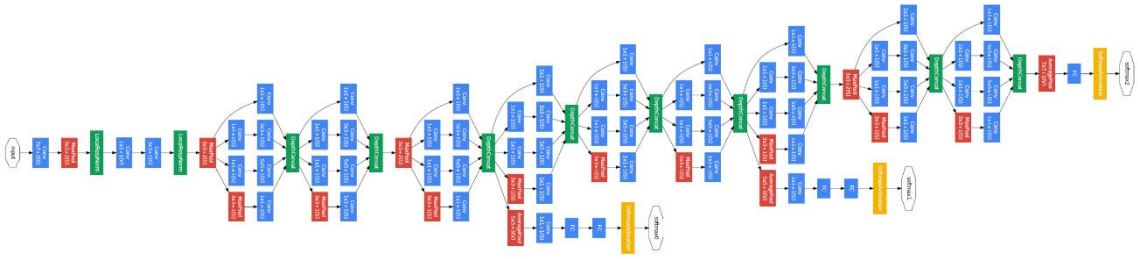


Figure 2.11. Architecture of GoogLeNet [27]

2.3.4. Squeezenet

There are 18 layers of deep convolutional networks in SqueezeNet. The ImageNet collection, which contains over a million photos, was also used to train these models. For example, a mouse, keyboard, pencil, and various other objects were all taught to SqueezeNet throughout its extensive training. As a result, the network has learned a wide range of visual feature extraction representations. If you want a network with equal accuracy to the SqueezeNet v1.0 network while using fewer floating-point operations per prediction, there are two varieties of SqueezeNet v1.1. The network has approved your work. To connect to the web, you must have at least two twos and two twos, as well as twos and twos [28].

The table below depicts the conventional SqueezeNet architecture.

layer name/type	output size	filter size / stride (if not a fire layer)	depth	s_{1x1} (#1x1 squeeze)	e_{1x1} (#1x1 expand)	e_{3x3} (#3x3 expand)	s_{1x1} sparsity	e_{1x1} sparsity	e_{3x3} sparsity	# bits	#parameter before pruning	#parameter after pruning
input image	224x224x3										-	-
conv1	111x111x96	7x7/2 (x96)	1				100% (7x7)			6bit	14,208	14,208
maxpool1	55x55x96	3x3/2	0									
fire2	55x55x128		2	16	64	64	100%	100%	**33%**	6bit	11,920	5,746
fire3	55x55x128		2	16	64	64	100%	100%	**33%**	6bit	12,432	6,258
fire4	55x55x256		2	32	128	128	100%	100%	**33%**	6bit	45,344	20,646
maxpool4	27x27x256	3x3/2	0									
fire5	27x27x256		2	32	128	128	100%	100%	**33%**	6bit	49,440	24,742
fire6	27x27x384		2	48	192	192	100%	**50%**	**33%**	6bit	104,880	44,700
fire7	27x27x384		2	48	192	192	**50%**	100%	**33%**	6bit	111,024	46,236
fire8	27x27x512		2	64	256	256	100%	**50%**	**33%**	6bit	188,992	77,581
maxpool8	13x12x512	3x3/2	0									
fire9	13x13x512		2	64	256	256	**50%**	100%	**30%**	6bit	197,184	77,581
conv10	13x13x1000	1x1/1 (x1000)	1					**20%** (3x3)		6bit	513,000	103,400
avgpool10	1x1x1000	13x13/1	0									
	activations			parameters			compression info				1,248,424 (total)	**421,098** (total)

Figure 2.12. Squeezenet configuration details [29]

SqueezeNet's CNN feature extraction in SqueezeNet 1.1 [29]. is incredibly effective despite

its small size. According to Figure 2.10, SqueezeNet 1.1 consists of a single convolution layer (conv1), three top pooling layers, eight fire modules (Fire2 9), and a final convolution layer (conv10). By employing the fire module instead of the traditional convolution layer, SqueezeNet reduces network parameters and improves overall accuracy.

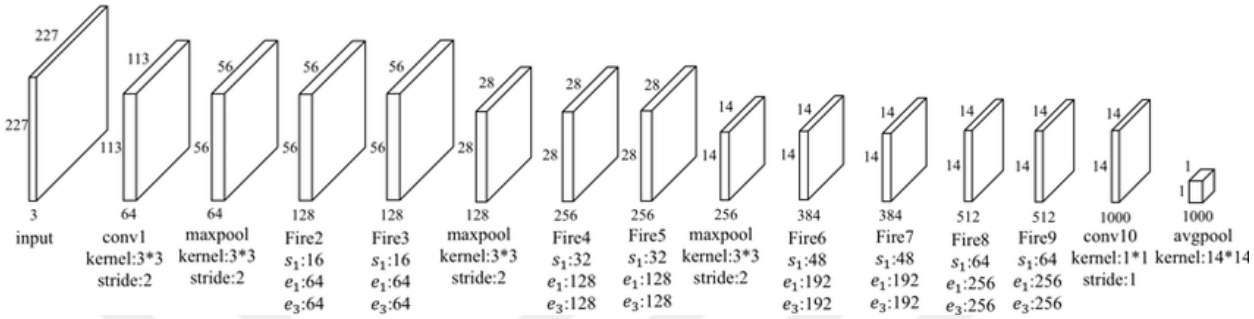


Figure 2.13. The architecture of SqueezeNet 1.1

2.4. Challenges in face recognition systems

Major technical challenges in the area of self-recognition are the following [23].Security issues.

Deep facial recognition systems are becoming increasingly vulnerable to presentation threats, adversarial assaults, template attacks, and digital manipulation attacks.

1) A presenting attack employing a 3D silicone mask with a skin-like look that moves with the face, offering a challenge to anti-spoofing procedures.

2) Despite the recent proposal of adversarial perturbation detection and mitigation approaches, the core cause of adversarial susceptibility remains unknown, and new forms of adversarial assaults are continuously being developed regularly.

3) Another key difficulty is how to develop entirely preventable templates without compromising accuracy, and the stolen deep feature template might be used to recover its face look.

4) A digital manipulation attack enabled by GANs may synthesize wholly or partially modified faces via expression swap, identity swap, attribute manipulation, and full-face synthesis, which remains a key hurdle to deep FR security.

2.4.1. Privacy-preserving face recognition.

Because of the current dissemination of biological information, there has been an increase in

privacy issues. Gender, age, ethnicity, and other demographics may be inferred from facial photographs, but they can also be used to infer genetic information. Semi-Adversarial Networks and further pioneering research have looked at ways to generate recognized biometric data that can hide some of the sensitive information given in face photos. To solve public privacy issues, more study into the principles of visual cryptography, signal mixing, and picture disruption is required—a more advanced kind of facial recognition. The bulk of the time, deep facial recognition algorithms are deemed superior to humans.

2.5. Literature Survey

AlexNet [30].was reported to have attained SOTA recognition accuracy in the ImageNet large-scale visual recognition competition 2012, exceeding the previous outstanding result by a wide margin. The AlexNet system incorporates techniques like dropout, augmented data, and the corrected linear unit (ReLU) with five convolutional layers and three fully connected layers. For deep learning, ReLU was often regarded as the most crucial component. An established network architecture using 3 3 convolutional filters with 22 pooling was introduced in 2014 by VGGNet [31]. It increased the network's depth to 16-19 weight layers, allowing deep architectures to learn progressively nonlinear mappings with substantially higher flexibility.

"Inception Module" was introduced to GoogleNet[32]. a 22-layer network, in 2015, with the addition of two more intermediate softmax supervised signals. Integrating multi-resolution data concatenates all feature maps and performs several convolutions in parallel with different receptive fields (1, 33, and 55).

Instead of learning a desired underlying mapping directly, ResNet [33]. recommended that layers learn a residual mapping concerning the layer inputs $F(x):= H(x)x$ to make training of intense networks easier (up to 152 layers). Shortcut connections are used to implement the original mapping, which is now written as $F(x) + x$.

An SE block that dynamically adjusts channel-wise feature responses by explicitly modeling channel interdependencies was the ILSVRC 2017 winner, SENet [34].Existing architectures, such as ResNet, can benefit from adding these blocks to their representational capabilities.

VGGface [35].devised a method utilizing the internet to obtain a huge dataset. After training with this dataset, it used a triplet loss method similar to FaceNet to fine-tune the networks. VGGface has a 98.95 percent success rate.

Angle softmax (A-Softmax) loss was created in 2017 by SphereFace [36].utilizing a 64-layer ResNet architecture to learn discriminative face traits. Using this method on LFW, the success percentage increases to 99.42%.

Using three continuous emotions,[37].... proposed a CNN model to understand facial emotions. This model uses Xception-inspired residual blocks and depth-separable convolutions to decrease the total number of parameters to 33k. A convolutional neural FER network is used to identify emotional stability. Convolutional neural networks (CNNs) may learn to extract attributes from images via convolutional methods, eliminating the requirement for human feature extraction from images. The proposed technique has an overall accuracy of 81% for unseen outcomes. Positive and negative emotions are detected with 87% and 86% accuracy, respectively. On the other hand, the accuracy of neutral emotion detection is only 51%.

The OpenCV computer vision class library, which is based on the AdaBoost algorithm, was used by [38].to accomplish face recognition using a framework focusing on deep facial expression learning. When building neural networks, the CNN design relies on the open-source artificial neural network software Keras, written in the Python programming language. The Descent Gradient Stochastic technique is used in this model (SGD). He utilized the FER2013 database for training CNN neural networks.

This study employed FER-2013 data and bespoke datasets from Kaviya and colleagues [39]. The RGB image is converted to a grayscale image for emotional recognition before processing. Faces may be recognized in images of Haar taken in real-time or at a fixed location. The facial characteristics can be resized and processed if the face is located. According to the five moods, the facial characteristics are trained using CNN to identify them. The community's emotional state is measured using a weighted average feeling. Finally, a speech synthesizer is employed to provide an audio output based on the expected emotion of the group. The model's testing accuracy was 65 percent for the FER-2013 dataset and 60 percent for bespoke datasets.

The authors of[40].... proposed a new deep CNN-based emotional intelligence system for detecting and diagnosing mental health issues. The proposed method employs a unique approach to evaluate facial photos and measure emotional and temporal production. As a result, the final classification results are based on a linear LDA rather than the fully linked AlexNet 6 sheet. It has three sections: input of facial expression videos, pre-processing of pictures, and the expected interpretation of facial expressions. Tests show that this technology exceeds primary precise and efficient ways, indicating that it may be utilized as a smart, low-cost cognitive aid for the recognition, tracking, and diagnosis of a patient's mental health via automated facial expression analysis.

[41]....examined and contrasted two of the most often utilized FER techniques to provide insight into their accuracy. LBP and CNN are the methods used in this case. LBP's major goal in feature extraction is to prepare the data so that SVM classifiers can sort it. CNN beats LBP

in terms of its integrated classifier, according to the implementation results (softmax).

[42].... suggested a novel technique to analyze facial expressions by using an attention mechanism. The use of LBP features is also included in network attention layers. Using LBP characteristics, which capture the minor variances in skin texture, we can better recognize motions with little nuances. Nanchang University Facial Emotions, a new dataset for recognizing facial emotions, was also developed and registered by them (NCUFE). There are 490 photographs of 35 people with seven various facial emotions (disgust, anger, fear, joy, sad, astonished, and neutral) in this collection. Both RGB and depth photos were captured for each participant. Five datasets were studied extensively by the researchers. Some data sets are obtained in the field while others are collected in the laboratory, such as the CK+ and JAFFE datasets. The model's output is also compared to the most recent speech recognition technologies. There is evidence that the model outperforms conventional dataset methods. As a result, this technique can only be used with two-dimensional pictures.

A study by [43]....showed that CNN might be used to identify facial emotions (RBF). Corrective measures can be implemented more promptly if the psychological difficulties of the patients are discovered sooner rather than later. Observing their facial expressions may explain how people feel, cope, and get sick. In the (FER) process, feature extraction and classification are crucial. A designation is a critical tool for distinguishing emotions such as joy, sorrow, fury, hatred, and shock. Static, sluggish, and heavy are the three sorts of signals that may be seen on the surface. The experimental results suggest that the (FER) 2013 data set is more accurate when the recommended method mix is used.

[44].... presented a new loss feature called the advanced softmax loss to eliminate inconsistent training expressions. The recommended losses guarantee each class has a level playing field and potential by using fixed (unlearnable) weight parameters of equal magnitude and equally allocated in angular space. According to the research, proposed (FER) techniques outperform several current (FER) methods. The proposed loss function can be used as a single signal or in conjunction with other loss functions. Research on FER2013 and the real-world practical face (RAF) databases has shown that ASL is significantly more accurate and effective than many existing approaches.

[45].... offered two novel CNN architectures based on the FER-2013 dataset. These are both primary and unique in terms of hyperparameter collecting at various levels of the network. Network topologies in Models 1 and 2 attain human precision with the FER-2013 dataset. Model 2 is a more refined version of the first. Model 1 architecture is unusual since it uses a set kernel size and determines the number of filters across the network depth. The number of

filters decreases as the network depth grows in this setup. Model 2 is smaller than Model 1, yet it has the same features. When comparing the proposed models to the most up-to-date signs of the best architecture for the FER-2013 data set, both use a kernel size of eight.

[46].... suggested a two-stage social signal analysis approach based on DCNN to evaluate face expression. The suggested system is separated into two stages: the first stage is derived from a series of facial expressions using the SoftMax score for a binary CNN. The second stage is based on a neural network. A neutral expression framework and a fully expressed frame are provided to the DCNN that matches them in the second round of training. The suggested method efficiently removes individual differences with the neutral language system's variations and the entire expression frame. Students' emotional states were studied with 96.28 percent accuracy in an e-learning scenario.

Using the DCNN technique created by [47].... the researchers were able to identify the emotional state of a person's face. A biological feature extractor (BFE) is used to obtain low-level information. The GF feature extractor will create two local intermediate features (M and D). The probability values of seven phrases are calculated using a Softmax classifier after the proposed DCNN model. This process uses measured and discrete data to combine the results based on the specified model. According to empirical research, the FER mission's success will be aided by local and holistic components. However, FER's efficacy in a controlled laboratory context is frequently lower. In addition, it is necessary to test the model in real-world scenarios.

A DCNN-enhanced VGG expression reconnaissance model was introduced by [48]....(CNN). The VGG-19 is used to improve network structure and parameters in the model. The lack of visual examples necessitated the employment of migrating learning approaches. Facial expression data from the CK+ Database is used to train and assess the shallow CNN, Alex-Net, and the revised VGG-19 deep CNN models. The models' outcomes are then compared. An image database should determine how many layers a DCNN should have to extract picture properties more effectively than a shallow CNN. As a result, the condition known as "over-fitness" might emerge. More useful input data features and fewer parameters can be achieved by using convolution layers with smaller filters with the same susceptibility. The enhanced VGG-19 network model's performance will be on par with current network models.

[49]...reviewed the literature on facial expression detection and included several approaches and strategies for analyzing facial expressions. They concluded that computers could identify human emotional responses due to the fast development of these approaches and human-machine interaction. They concluded that for many software systems in many application domains, quick analysis and precise detection of facial emotions are critical.

Some research has been done to assess the psychological states of human faces in moving and static photographs based on emotional expressions.[50].... employed the haar cascade functions to evaluate dynamic changes in the eye and mouth areas, which are critical in recognizing emotional expressions. For image analysis, unique algorithms were employed in the C++ programming language. As a consequence of the research, it has been proved that facial expressions in an image may be identified.

[51].... built software for image indexing and retrieval in their work. A categorization strategy based on dominating colors of photographs is suggested in this work. In HLS space, the image is given a colorimetric profile (Hue, Lightness, Saturation - Hue, Light, Saturation). First, the hue was defined using a fuzzy representation that took into account the irregularity of the color distribution. Then the profile was generated using fuzzy functions that represented the degree of picture membership owing to distinct classes. It is advised that pixel areas rather than individual pixels be highlighted and analyzed to improve performance and establish more accurate profiles. An edge detection method was used to produce these areas. A sample of pixels inside the region was chosen to determine the region's color. The study indicated that such software might be used to distinguish compatible and incompatible images, among other things, based on the prevailing colors discovered.

Although individuals can quickly see and understand faces and facial expressions in an image [52].... suggest a novel target representation and localization technique, a critical component in nonrigid object visual tracking. According to the research, spatial masking with an isotropic core was used to organize feature target representations based on a hectogram. Because the gravitational field of the local maxima may be used to frame the target localization problem, a metric derived from the Bhattacharyya coefficient is used as a similarity measure, and the mean shift technique is used to carry out the optimization. It is also discussed how to employ motion filters and data association methods in conjunction with integration. Background data, Kalman tracking based on motion patterns, and face tracking are only a few of the applications that have been discovered. The novel technique successfully handles difficulties connected to camera motions, partial occlusions, clutter, and target scale changes, according to the monitoring samples supplied from the investigations.

[53].... created a near real-time face recognition (verification) system that monitors the person's head motions by detecting and recognizing the human face and comparing it to known people's facial traits and then decides his identification. Face recognition is accomplished by reducing it to a two-dimensional object recognition issue, which uses the fact that faces can be characterized with a minimal collection of 2D distinctive appearances owing to their upright

orientation under typical settings. They reflected the difference in a feature region ("face space") comprised of the codes with the most significant resolution capabilities among the known facial images. They called this part of the face "Eigenface." They also said that they are eigenvectors of face sets and cannot be used to represent single characteristics like eyes, ears, and nose. They concluded that the ability to learn in an unsupervised manner is offered due to the experiments.

[54].... created a face recognition system unaffected by substantial variations in lighting and facial expression. Each pixel in the image is assessed as a coordinate value in a high-dimensional space using the pattern classification method. If the face is the surface of a shadowless Lambert, they used the finding that some facial pictures are situated in a 3D linear subspace of the high-dimensional image space in fixed distortion under a different lighting scheme. The images, however, diverged from this linear subfield because the faces did not wholly match the Lambert surfaces and so caused shading on themselves. Rather than modeling the aberration directly, they projected the image linearly into a subspace, resulting in significant abnormalities in some regions of the face. They constructed well-grouped classes that live in a low-dimensional subspace based on Fisher's Linear Discriminant as a projection technique, even with extreme variations in illumination and face expressions. They concluded that the Eigenface methodology, based on linearly projecting the image space into a low-dimensional subspace, has similar processing demands. However, in testing the Harvard and Yale Face Database, their detailed experimental results revealed that the suggested "Fisherface" approach had a lower error rate than the Eigenface methodology.

2.6. Summary

This chapter introduced the theoretical background related to Artificial Neural networks in general and Convolutional Neural Networks (CNN). In addition to a survey of recent AI-based face detection methods.

3. PROPOSED SYSTEMS

3.1. Introduction

In this section, the proposed system implementation is described. A method is proposed to detect and monitor facial expressions to categorize them into seven face states (Angry, Disgust, Fair, Happy, Neutral, Sad, and Surprise), this study proposes a method for recognition of facial expressions in real-time application.

Design of the proposed FER system

The block diagram shown in Figure 3.1 illustrates the system process flow. The face image is captured from a real-time video frame using a webcam. The image frames are preprocessed by normalization, resizing, and Grayscale to RGB conversion. Feature extraction and classification process take place in the processed image. Finally, the classification results will be displayed on a specially designed Graphical User Interface (GUI) for real-time facial expression recognition.

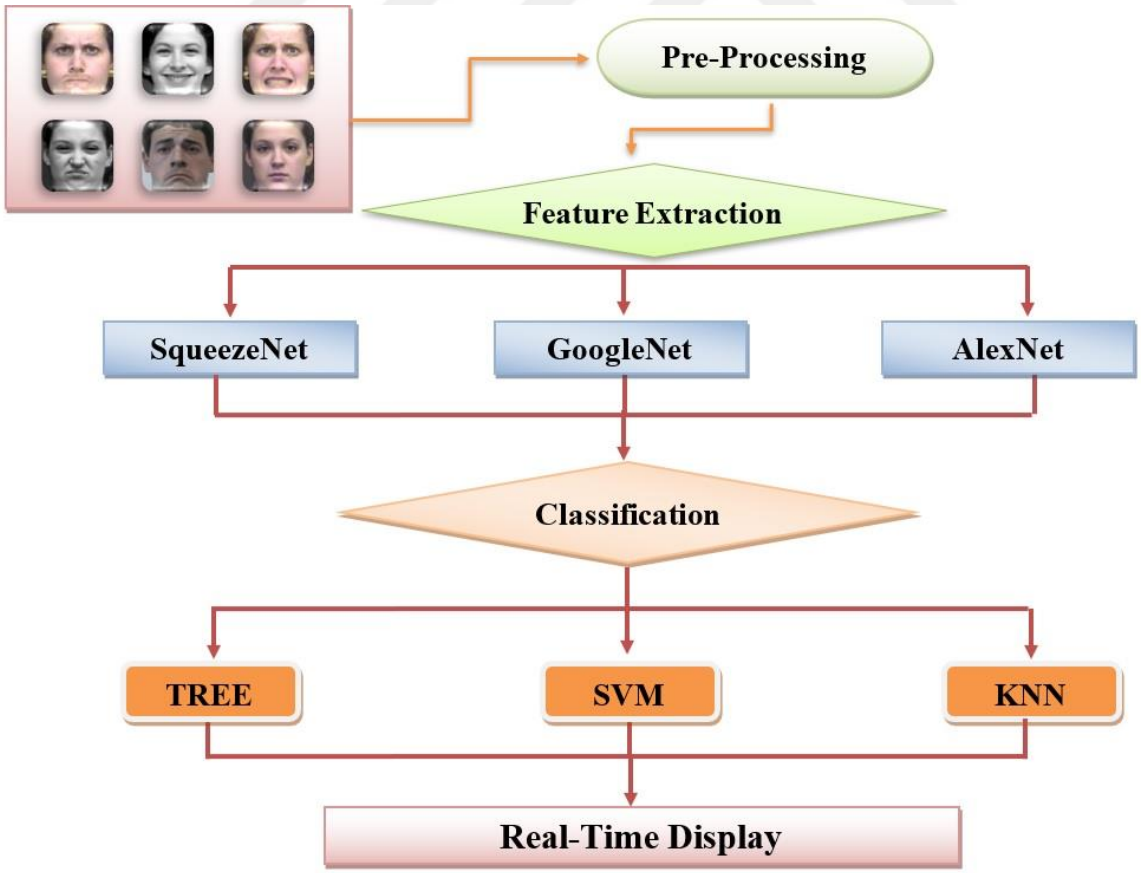


Figure 3.1. Illustrate The System Process

where the proposed methods consist of five stages :

1. Dataset.
2. Preprocessing.
3. Feature extraction.
4. Classification.
5. Real-Time.

3.2. Dataset

In this research, two datasets of facial expression are utilized for training, and the validation process is discussed as follows:

Dataset: Evaluating facial expression recognition techniques requires using one or more databases, in our thesis; we use two datasets to evaluate our proposed methods:

3.2.1. Cohn-Kanade Facial Expression Database (CK)

Cohn-Kanade Facial Expression Database (CK) In 2000, This dataset was composed of seven classes last for different facial expressions. The (angry) class contain 31 of different size and different color mode (RGB and Grayscale). Disgust class contains 51 different sizes and grayscale images. The fear class contains 24 images of different sizes and different color modes. The happy class contains 57 different sizes and grayscale images. The neutral class contains 261 images of different sizes and different color modes. Sad class 24 of different sizes and different color modes. Surprise class 57 different sizes and grayscale images. .[55].

Table 3.1. Total sample of CK data set (505 samples)

Natural	Happy	Surprise	Anger	Fear	Disgust	Sadness
261	57	57	31	24	51	24

Figure 3.2. Sample images of CK dataset[55]

3.2.2. Japanese Female Facial Expression (JAFFE) Database

This dataset contains set of images for each facial expression with a total of 212 images of 256x256 pixel grayscale images.

Japanese Female Facial Expression (JAFFE) Database A group of researchers including Miyuki Kamachi, Michael Lyons, and Jiro Gyoba the University of Kyushu planned this database using photos taken in the Psychology department at the University of Kyushu. The JAFFE database contained 212 images expressing different facial expressions, including Natural, Happy, Sad, Surprise, Anger, Disgust, and Fear, using Japanese female models with tiff image sized 256×256 pixels, Figure 3.3 is showing Sample images of the JAFFA dataset [56].

Table 3.2. Total sample of JAFFA data set

Natural	Happy	Surprise	Anger	Fear	Disgust	Sadness
31	30	30	30	32	31	28

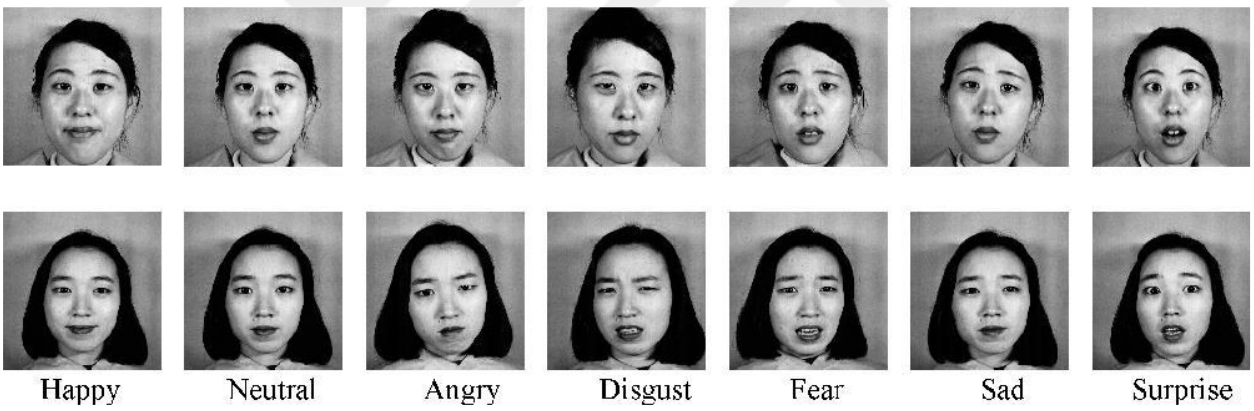


Figure 3.3. Sample images of the JAFFE dataset [56]

3.3. Preprocessing

Image preprocessing is a very important step for FER since it ensures that image data are well prepared for certain types of training and testing. The input size and the number of color channels for each network are calculated the input size of the network is applied to training and testing images to make the image size suitable for it. So, the images will be resized into (224x224) pixels for AlexNet and GoogleNet, and (227x227) pixels for SqueezeNet. The (augmented Image Datastore) function in Matlab provides a very good option to convert a grayscale image into RGB which is called “Color Preprocessing” as shown in Figure 3.

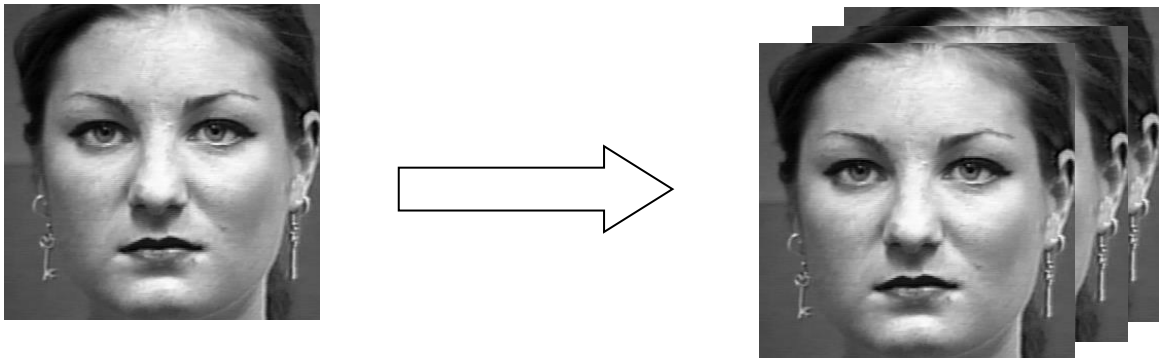
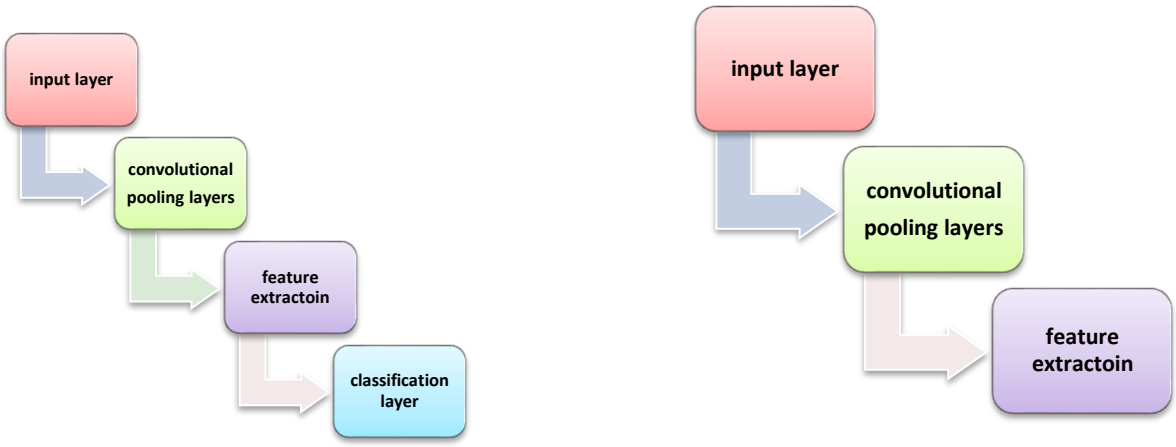


Figure 3.4. Convert Gray image to RGB image

3.4. Feature Extraction

In this research, feature extraction depends on pre-trained networks that are specially designed for feature extraction and classification purposes. In general, the pre-trained network architecture is shown in figure 3.5.a composed of four stages: input layer, multiple-convolutional layers, pooling layers, feature extraction layer, and classification layer. So, we will modify these networks as shown in figure 3.5.b by removing the classification layers.



A b

Figure 3.5.a. Orignal architecture of the pre-trained network**. Figure 3.5.b.** modified pre-trained network

3.5. Three types of CNN pre trind network are used in this work as follows

3.5.1. AlexNet CNN

In this method, the pre-trained AlexNet which is mentioned in chapter two, this network is used for feature extraction. The first convolutional layer contains a 96 set of weights as shown in Figure 3.6-a which can be visualized as activation of the first layer as shown in figure 3.6.b feature layer of this network is 'fc7' which provides 4096 values for each image. A (KNN, Tree, and SVM) model (also described in chapter two) is established as a classifier for the network. The classifier is trained for the feature extracted from the feature extraction layer of AlexNet. The overall structure of the feature and classifier network is shown in figure 3.6

a b

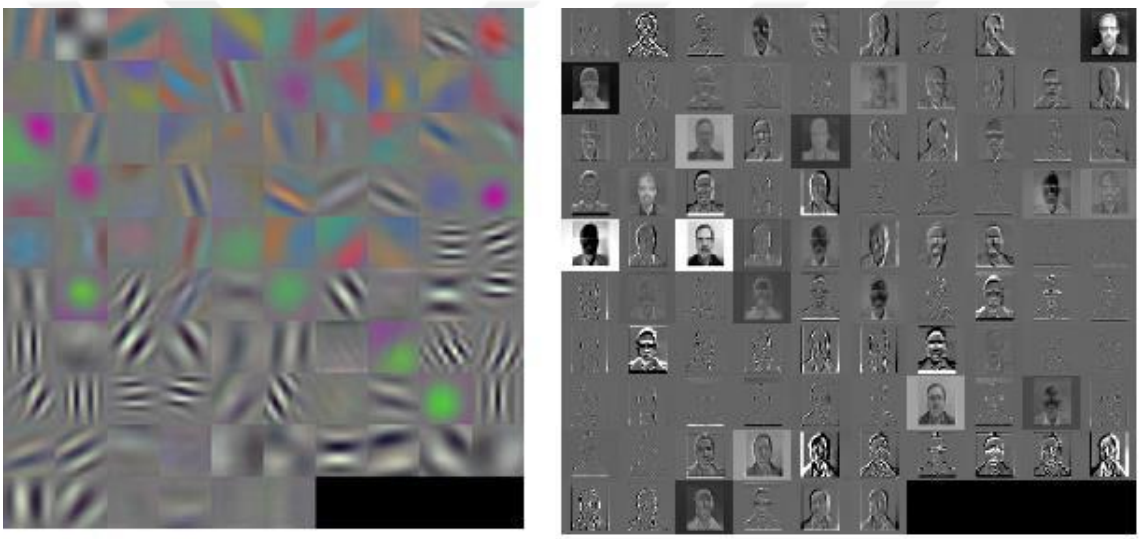


Figure 3.6.a. First convolutional layer weights in AlexNet (96 filers). **Figure 3.6.b.** show Activations from the conv1 layer in AlexNet

The training process can be summarized as follow

Step 1: load the CK or JAFFE dataset and split it to train sets and test sets with different sizes for each set (80% for training and 20% for testing).

Step 2: Preprocessing images including resizing and color processing.

Step 3: import AlexNet and use the 'fc7' layer as the target feature extraction layer to find a vector of features (4096 features) from each image and save it as a features matrix.

Step 4: the feature matrix obtained from the feature layer is applied to classifier (SVM, KNN, or Decision TREE) algorithms.

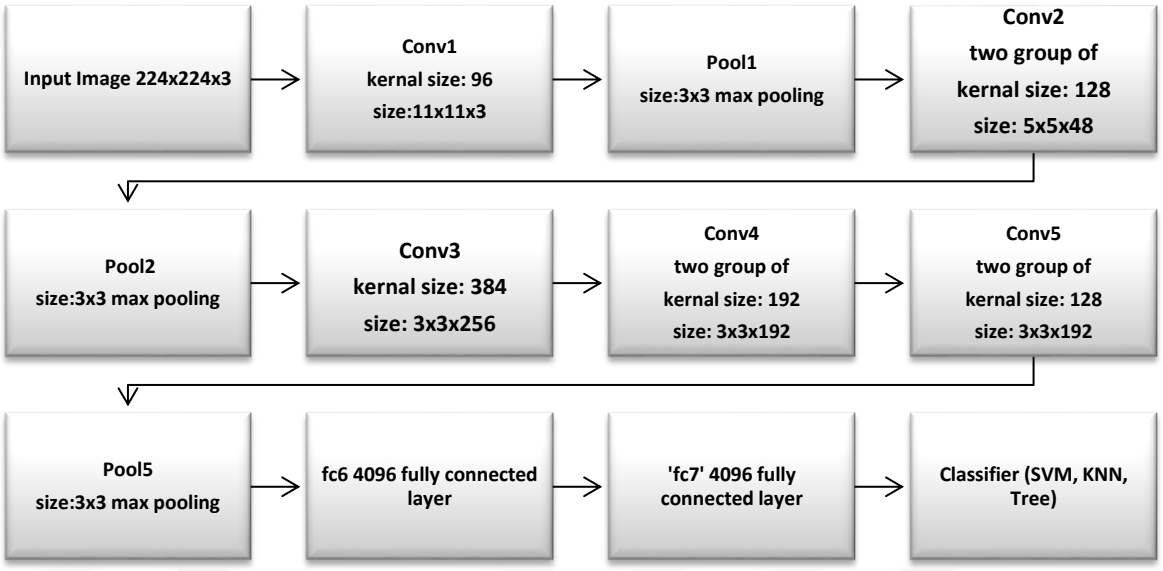


Figure 3.7. Block Diagram of modified AlexNet and KNN classifier

Three classifier layer are used for output as follow

3.5.1.1. AlexNet SVM

The output of feature training which represents the activation of feature layer 'fc7' by augmented training dataset, the results are fed into SVM as a classifier. SVM is trained with these options.

Learner: 'linear' Leanear classification model.

Coding: 'onevsall', For each binary learner, one class is positive and the rest are negative. This design exhausts all combinations of positive class assignments.

ObservationsIn : 'columns', For faster training time.

3.5.1.2. AlexNet Decision Tree

The output of the pre-trained network is also fitted by a Tree classifier which returns a fitted binary classification tree based on the input (feature extracted) contained in matrix X and output labels. The returned binary tree splits branching nodes based on the values of a column of X.

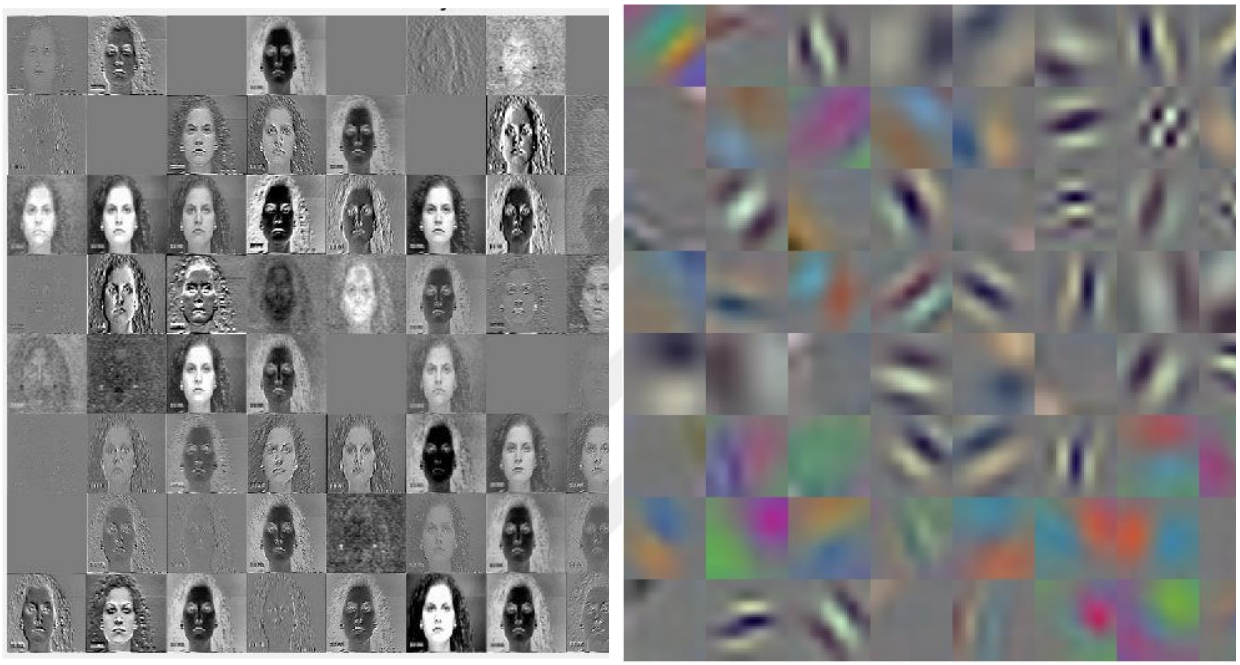
3.5.1.3. AlexNet KNN

In this method, a KNN was also tried as a classifier a for the modified pre-trained network to ensure maximum evaluation matrices. The KNN algorithm returns a k-nearest neighbor

classification model based on the predictor feature data and response label.

3.5.2. GoogleNet CNN

The second method includes using the pre-trained GoogleNet as a feature extractor which is also described in chapter two. The first convolutional layer contains 64 sets of weights as shown in Figure 3. 8-a, these weights can be visualized as activation of the first layer using a Matlab function "`mat2gray(w1)`" as shown in figure 3.8.b feature layer of this network is 'inception_5b-output' which provides 50176 values for each image.



(a) Activation from conv1 layer (b) Weights of first conv layer

Figure 3.8.(a). First convolutional layer weights in GoogleNet. **Figure 3.8.(b).** show Activations from the conv1 layer in GoogleNet

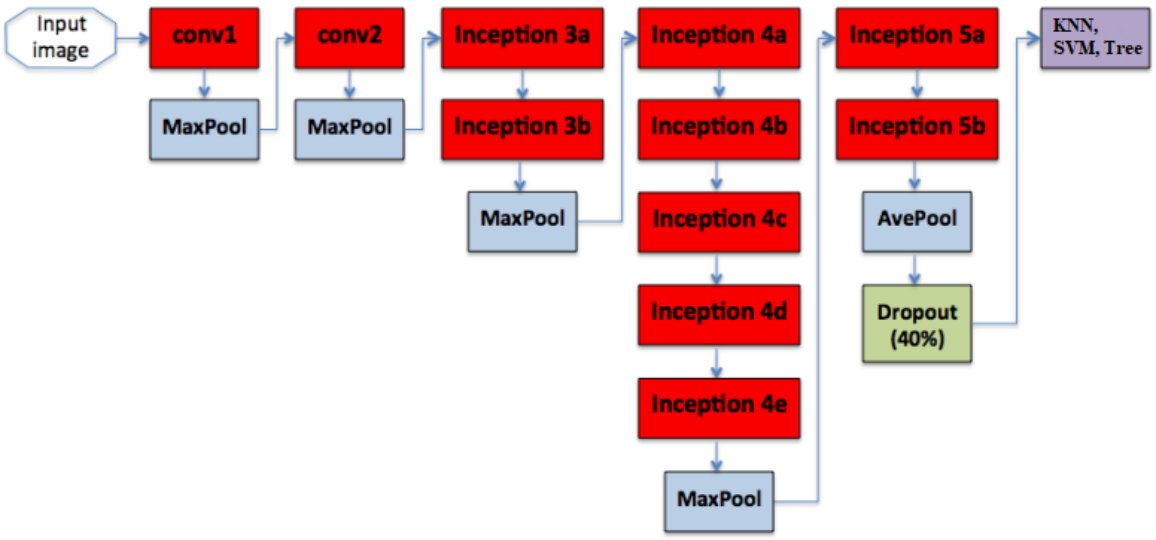


Figure 3.9. The modified GoogleNet architecture

The **GoogleNet method** algorithm can be summarized in four steps.

Step 1: load the CK or JAFFE dataset and split it to train sets and test sets with different sizes for each set (80% for training and 20% for testing).

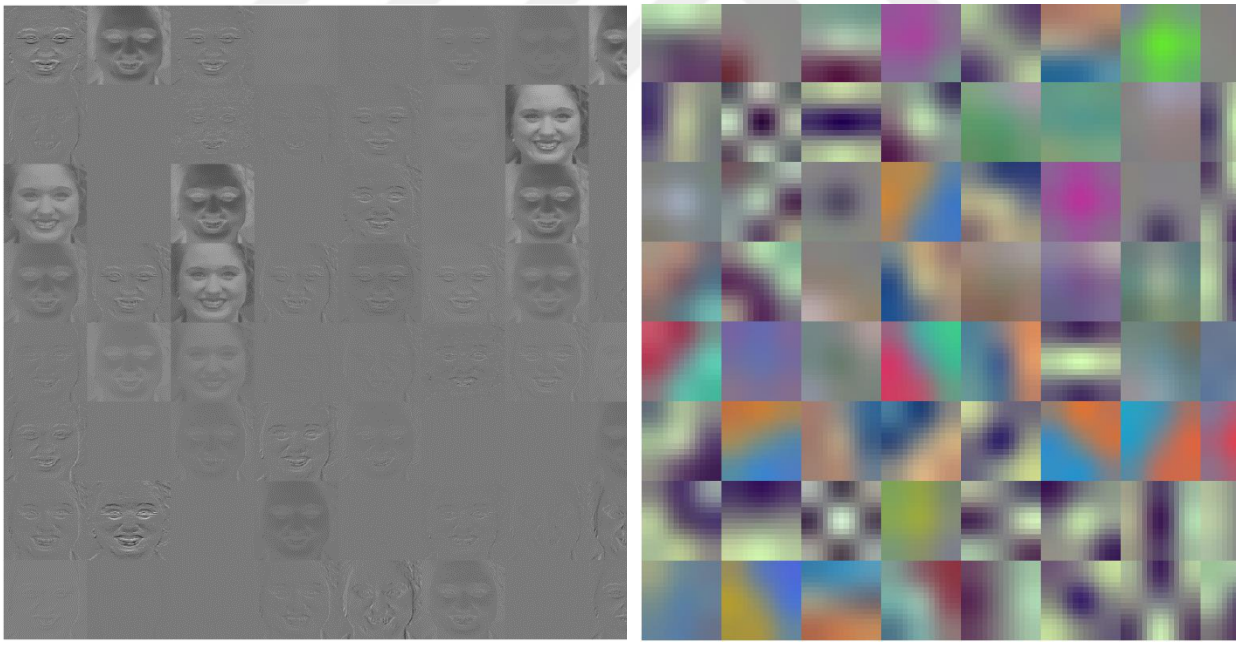
Step 2: Preprocessing images including resizing and color processing.

Step 3: import GoogleNet and use the 'inception_5b-output' layer as the target feature extraction layer to find a vector of features (50176 features) from each image and save it as a features matrix.

Step 4: the feature matrix obtained from the feature layer is applied to classifier (SVM, KNN, or Decision TREE) algorithms.

3.5.3. SqueezeNet CNN

The final method involves using the SqueezeNet deep learning network for feature extraction purposes. The first convolutional layer contains also has a 96 set of weights. Figure 3.9-a which can be visualizes the activations and weights of the network. A feature layer of this network is 'fire9-concat' for deep feature extraction. This layer provides a 100352 activation value for each image. Figure 3.10 Shows the modifier Squeezenet architecture.



(a) Activation from conv1 layer (b) Weights of first conv layer

Figure 3.10.a. First convolutional layer weights in SqueezeNet. **Figure 3.10.b.** show Activations from the conv1 layer in SqueezeNet

Step 1: load the CK or JAFFE dataset and split it to train sets and test sets with different sizes for each set (80% for training and 20% for testing).

Step 2: Preprocessing images including resizing and color processing.

Step 3: import SqueezeNet and use the 'fire9-concat' layer as the target feature extraction layer to

find a vector of features (100352 features) from each image and save it as a features matrix.

Step 4: the feature matrix obtained from the feature layer is applied to classifier (SVM, KNN, or Decision TREE) algorithms.

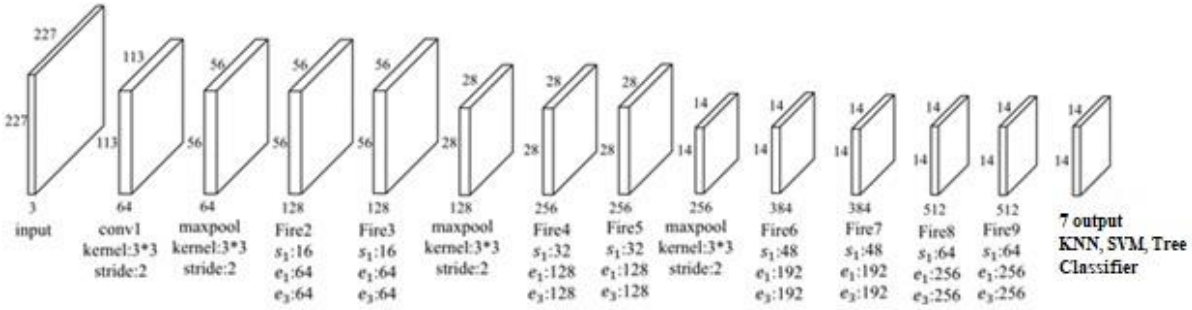


Figure 3.11. Structure of modified Squeezenet

3.6. Real-Time Display

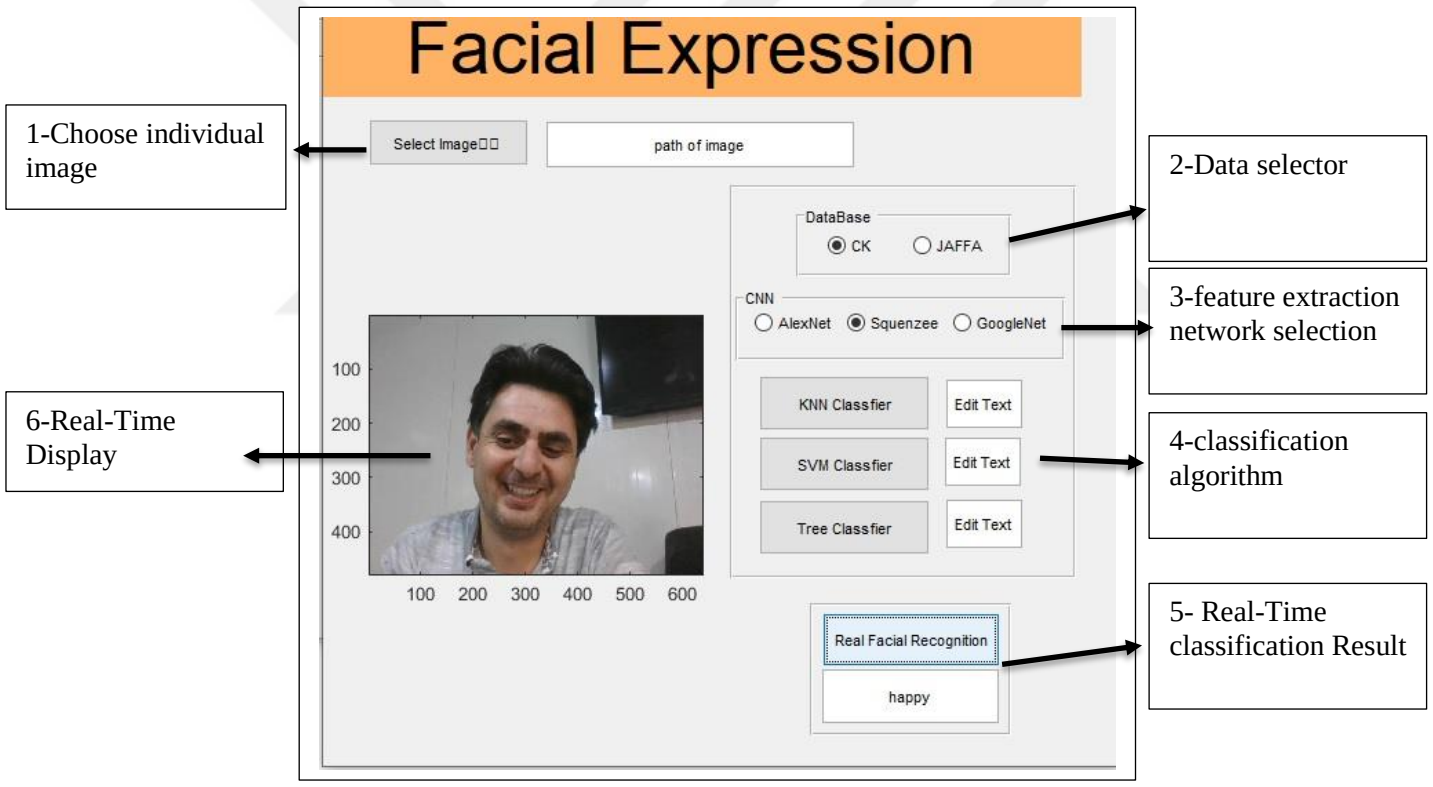


Figure 3.12. Real-Time Structure

The final stage in this system is to design a real-time user interface that provides an active user experience. The user interface is build-up using a Matlab GUI which makes it easy to link the feature extraction models, classification model, dataset, and classification result display. The designed GUI contains several objects as follows:

1. Choose an individual image: to classify a single image.

2. Data selector: to select a used dataset
3. Feature extraction network selection: to select a pre-trained network as a feature extractor.
4. Classification algorithm: enables us to choose a classification method.
5. Real-Time classification Result: display the results.
6. Real-Time Display: display real-time video from a webcam.

4. Results and Discussions

4.1. Introduction

In this chapter, methodological findings are discussed and analyzed. The response of three proposed neural networks is measured that are primarily trained for FER. Also, the real-time simulation is obtained based on the prediction result of the neural network.

4.2. Performance evaluation

The performance evaluation of the FER system was investigated based on statistical data. In this section , results of the training in all deep learning networks are presented. The results are discussed using a complex matrix, the confusion matrix.

A confusion matrix is the most powerful measurement tool in classification problems and can be applied for both binary and multiclass classification. The confusion matrix enables visualizing and summarizing the network performance.

The confusion matrix has four basic elements as shown in Fig 4.1 that are utilized to determine the classifier's measurement parameters. These four matrices are: [57].

1. True Positive (TP) indicates the number of accurately classified positive samples.
2. True negative (TN) represents the number of accurately classified negative examples.
3. False Positive (FP) shows the number of actual negative samples misclassified as positive.
4. False Negative (FN) total number of actual positive examples misclassified as negative.

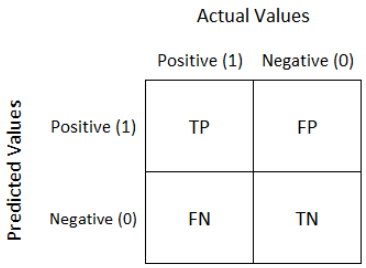


Figure 4.1. Confusion matrix elements

The performance of a model (using a confusion matrix) is calculated using the given formula below [57].

Recall=$\frac{TP}{TP+FN}$ ……..equation (1)

Precision =$\frac{TP}{TP+FP}$ ……..equation (2)

And F1-Score is calculated by the following equation also known as (Accuracy):

$$\text{F1-Score}=2*\frac{\text{precision}*\text{recall}}{\text{precision}+\text{recall}} \quad \text{.....equation (3)}$$

4.2.1. Alexnet Response

Alexnet is trained with both JAFFA and CK datasets using KNN, SVM, and Tree, the confusion matrix as well as precision, Recall, and F1-Score for each case explained below:

- JAFFA dataset and KNN

The total accuracy of the f1-score is 100%.

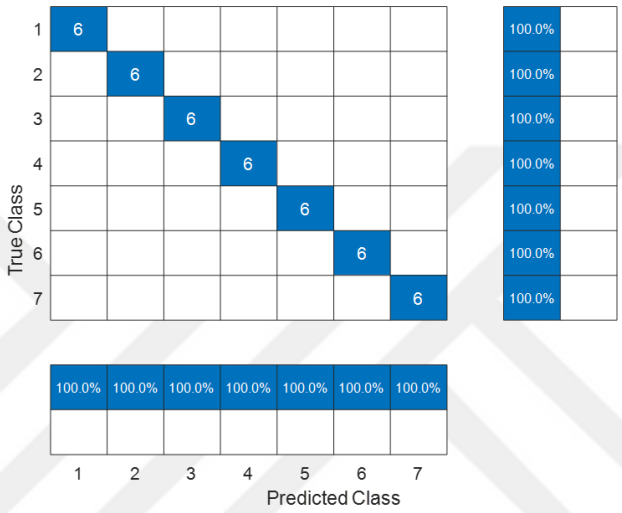


Figure 4.2. JAFFA dataset and KNN confusion matrix

Table 4.1. Alexnet response with KNN and jaffa dataset

	Precision	Recall	F1_Score
1	100	100	100
2	100	100	100
3	100	100	100
4	100	100	100
5	100	100	100
6	100	100	100
7	100	100	100

- **JAFFA dataset and SVM:**

  The total accuracy obtained is 67.1429%.

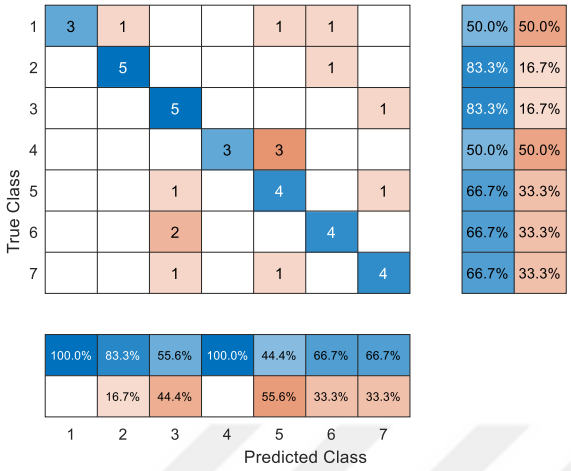


Figure 4.3. Alexnet with JAFFA dataset and SVM confusion matrix

Table 4.2. Alexnet response with KNN and JAFFA dataset

	Precision	Recall	F1_Score
1	100	50	66.667
2	83.333	83.333	83.333
3	55.556	83.333	66.667
4	100	50	66.667
5	44.444	66.667	53.333
6	66.667	66.667	66.667
7	66.667	66.667	66.667

- **JAFFA dataset and Tree:**

  Total accuracy is 83.8070%.

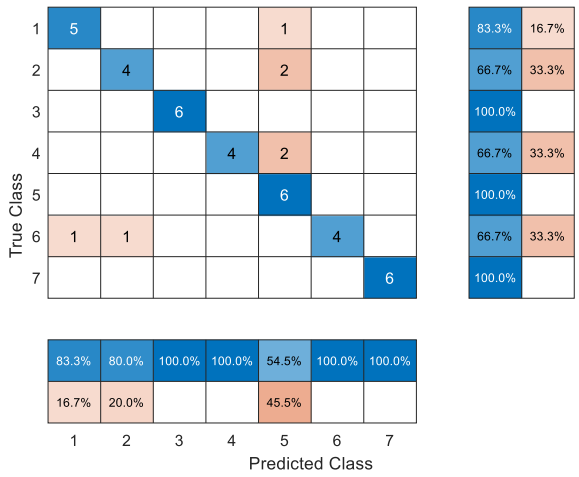


Figure 4.4. Alexnet with JAFFA dataset and Tree Confusion matrix

Table 4.3. Alexnet response with JAFFA dataset and Tree

	Precision	Recall	F1_Score
1	83.333	83.333	83.333
2	80	66.667	72.727
3	100	100	100
4	100	66.667	80
5	54.545	100	70.588
6	100	66.667	80
7	100	100	100

CK dataset and KNN:

Total accuracy is 87.3988

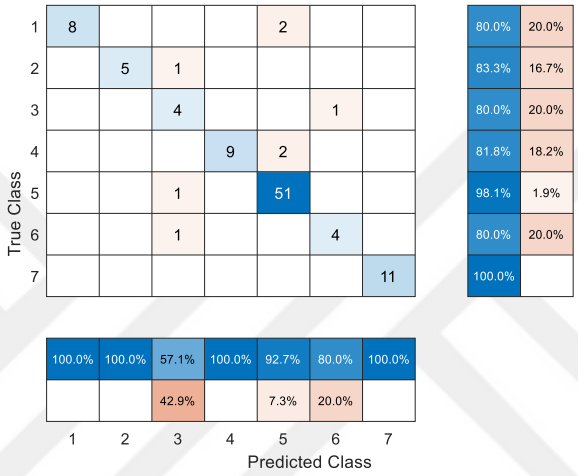


Figure 4.5. Alexnet with CK dataset and KNN Confusion matrix

Table 4.4. Alexnet response with CK dataset and KNN

	Precision	Recall	F1_Score
1	100	80	88.889
2	100	83.333	90.909
3	57.143	80	66.667
4	100	81.818	90
5	92.727	98.077	95.327
6	80	80	80
7	100	100	100

- **CK dataset and SVM:**

The total accuracy is 80.5568.

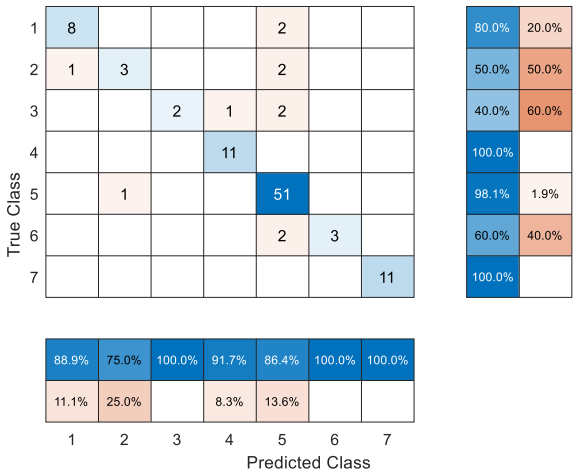


Figure 4.6. Alexnet with CK dataset and SVM Confusion matrix

Table 4.5. Alexnet response with CK dataset and SVM

	Precision	Recall	F1_Score
1	88.889	80	84.211
2	75	50	60
3	100	40	57.143
4	91.667	100	95.652
5	86.441	98.077	91.892
6	100	60	75
7	100	100	100

- **CK dataset and TREE:**

Total accuracy is 76.6945

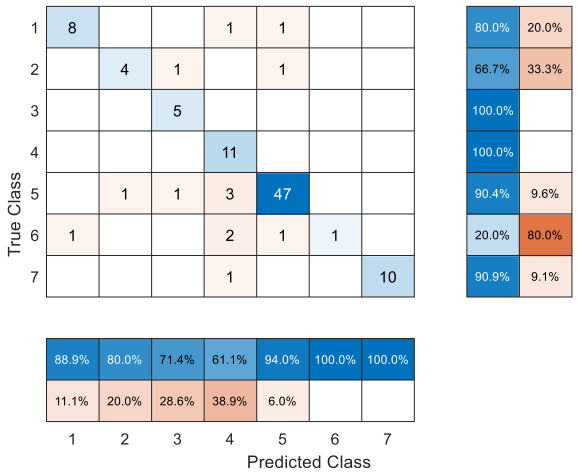


Figure 4.7. Alexnet with CK dataset and TREE Confusion matrix

Table 4.6. Alexnet response with CK dataset and TREE

	Precision	Recall	F1_Score
1	88.889	80	84211
2	80	66.667	72.727
3	71.429	100	83.333
4	61.111	100	75.862
5	94	90.385	92.157
6	100	20	33.333
7	100	90.909	95.23

4.2.2. GoogleNet

The results were obtained by using GoogleNet architecture also with both JAFFA and Ck datasets with KNN, SVM, and TREE.

- **JAFFA dataset and KNN:**

  Total accuracy is 87.9692

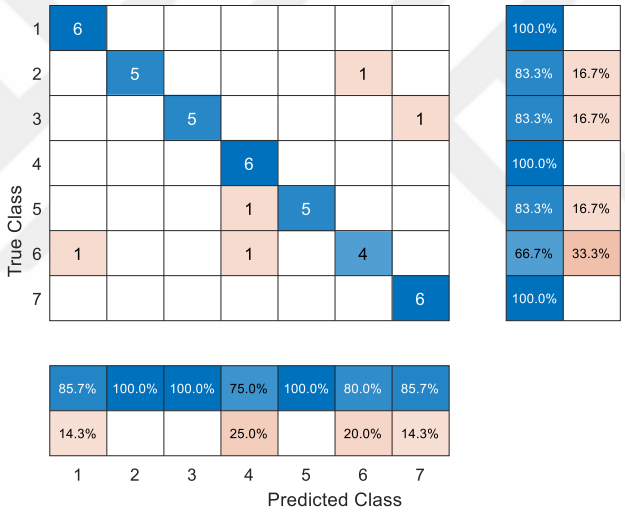


Figure 4.8. Googlenet with JAFFA dataset and KNN Confusion matrix

Table 4.7. Googlenet response with JAFFA dataset and KNN

	Precision	Recall	F1_Score
1	85.714	100	92.308
2	100	83.333	90.909
3	100	83.333	90.909
4	75	100	85.714
5	100	83.333	90.909
6	80	66.667	72.727
7	85.714	100	92.308

- **JAFFA dataset and SVM:**

Total accuracy is 84.7924

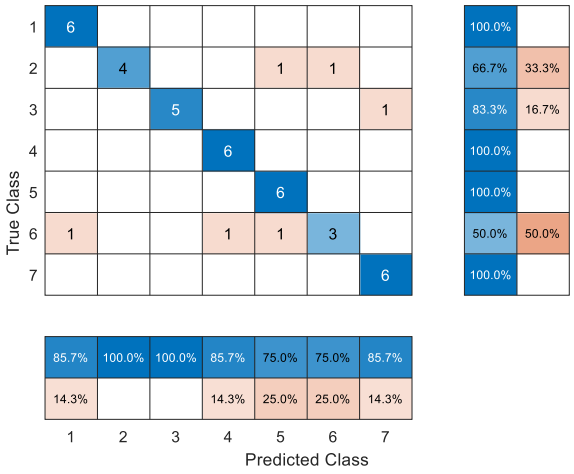


Figure 4.9. Googlenet with JAFFA dataset and SVM Confusion matrix

Table 4.8. Googlenet response with JAFFA dataset and SVM

	Precision	Recall	F1_Score
1	85.714	100	92.308
2	100	66.667	80
3	100	83.333	90.909
4	85.714	100	92.308
5	75	100	85.714
6	75	50	60
7	85.714	100	92.308

- **JAFFA dataset and TREE:**

Total accuracy is 78.0691

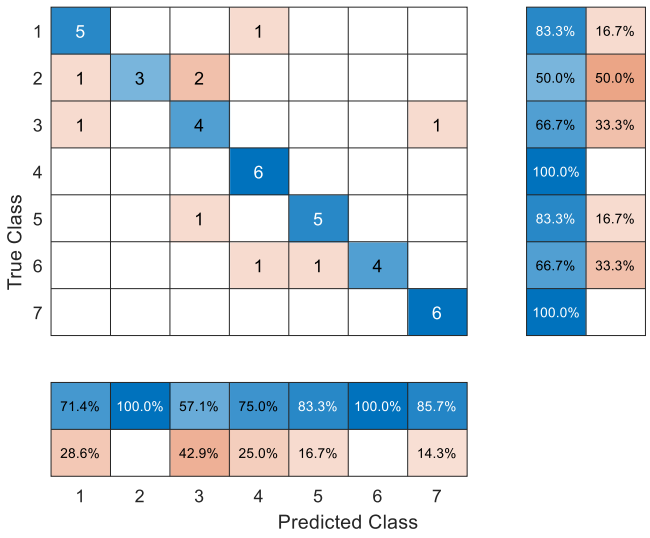


Figure 4.10. Googlenet with JAFFA dataset and TREE Confusion matrix

Table 4.9 Googlenet response with JAFFA dataset and TREE

	Precision	Recall	F1_Score
1	71.429	83.333	76.923
2	100	50	66.53
3	57.143	66.667	61.538
4	75	100	85.714
5	83.333	83.333	83.333
6	100	66.66	80
7	85.714	100	92.308

- **CK dataset and KNN:**

  Total accuracy is 84.3364

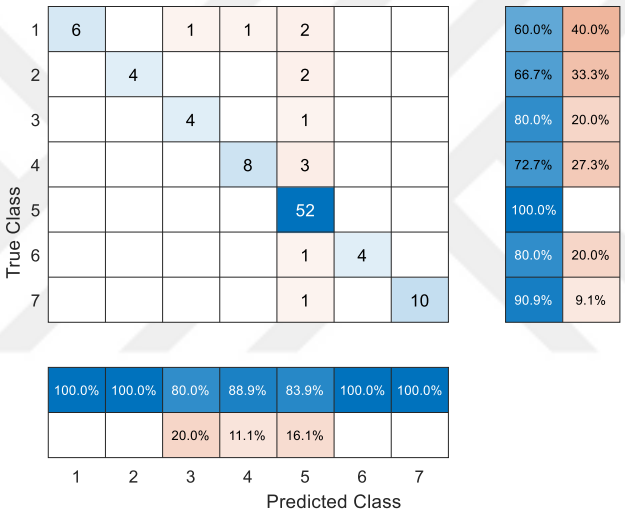


Figure 4.11. Googlenet with CK dataset and KNN Confusion matrix

Table 4.10. Googlenet response with CK dataset and KNN

	Precision	Recall	F1_Score
1	100	60	75
2	100	66.667	80
3	80	80	80
4	88.889	72.727	80
5	83.871	100	91.22
6	100	80	88.88
7	100	90.909	95.238

- **CK dataset and SVM:**

Total accuracy is 91.0961

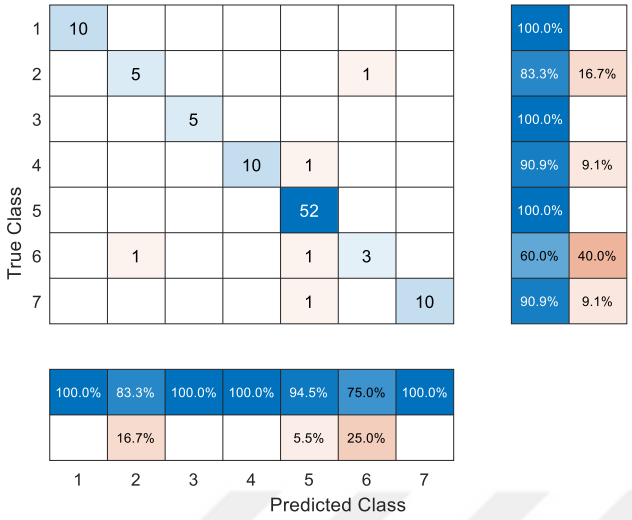


Figure 4.12. Googlenet with CK dataset and SVM Confusion matrix

Table 4.11. Googlenet response with CK dataset and SVM

	Precision	Recall	F1_Score
1	100	100	100
2	83.333	83.333	83.333
3	100	100	100
4	100	90.909	95.238
5	94.545	100	97.196
6	75	60	66.667
7	100	90.909	95.238

- **CK dataset and TREE:**

The total accuracy is 75.2897

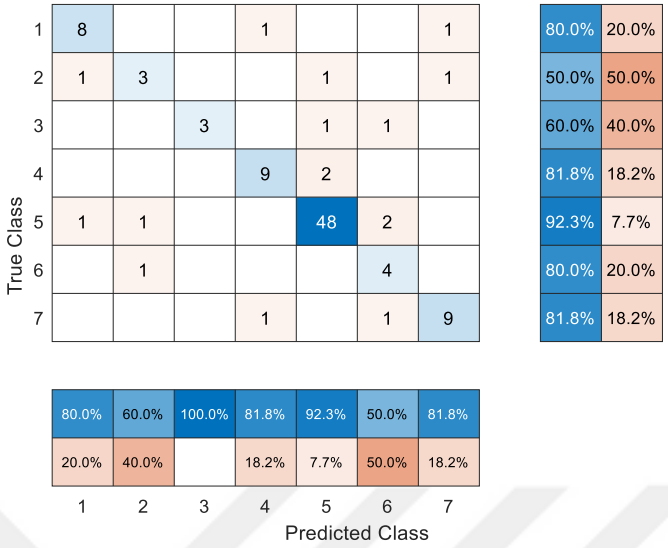


Figure 4.13. Googlenet with CK dataset and TREE Confusion matrix

Table 4.12. Googlenet response with CK dataset and TREE

	Precision	Recall	F1_Score
1	80	80	80
2	60	50	54.545
3	100	60	75
4	81.818	81.818	81.818
5	92.308	92.308	92.308
6	50	80	91.538
7	81.818	81.818	81.818

4.2.3. SqueezeNet

The last part of the results obtained SqueezeNet deep learning network also for JAFFA and CK datasets with KNN, SVM, and TREE algorithms as explained below.

- JAFFA dataset and KNN:

Total accuracy is 95.2048

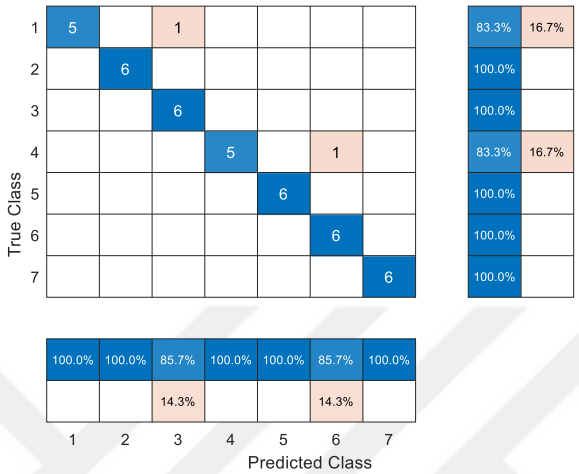


Figure 4.14. SqueezeNet with JAFFA dataset and KNN Confusion matrix

Table 4.13. SqueezeNet response with JAFFA dataset and KNN

	Precision	Recall	F1_Score
1	100	83.333	90.909
2	100	100	100
3	85.714	100	92.308
4	100	83.333	90.909
5	100	100	100
6	85.714	100	92.308
7	100	100	100

- **JAFFA dataset and SVM:**

  Accuracy is 95.2214

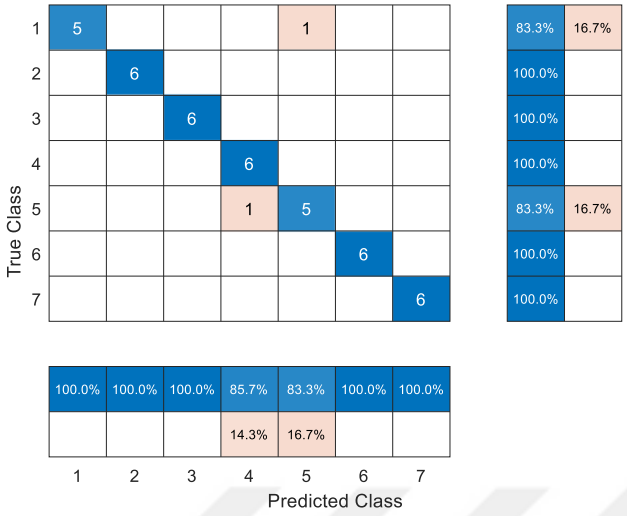


figure 4.15. SqueezeNet with JAFFA dataset and SVM Confusion matrix

Table 4.14. SqueezeNet response with JAFFA dataset and SVM

	Precision	Recall	F1_Score
1	100	83.333	90.909
2	100	100	100
3	100	100	100
4	85.714	100	92.308
5	83.333	83.333	83.333
6	100	100	100
7	100	100	100

- **JAFFA dataset and TREE:**

  Accuracy is 88.0786

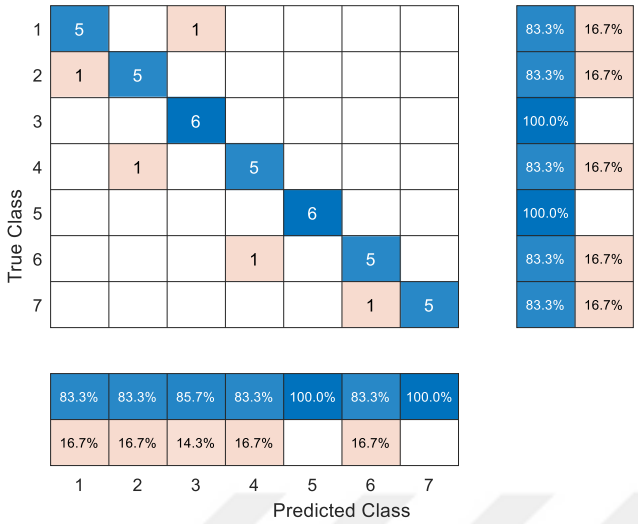


Figure 4.16. SqueezeNet with JAFFA dataset and TREE Confusion matrix

Table 4.15. SqueezeNet response with JAFFA dataset and TREE

	Precision	Recall	F1_Score
1	83.333	83.333	83.333
2	83.333	83.333	83.333
3	85.714	100	92.308
4	83.333	83.333	83.333
5	100	100	100
6	83.333	83.333	83.333
7	100	83.333	90.909

- **CK dataset and KNN:**

  Total accuracy is 96.6102

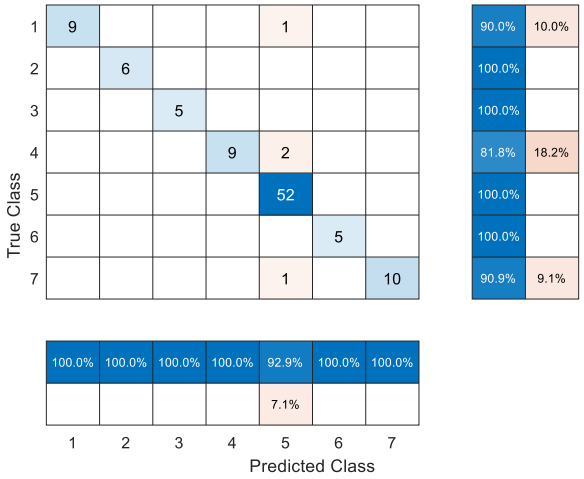


Figure 4.17. SqueezeNet with CK dataset and KNN Confusion matrix

Table 4.16. SqueezeNet response with CK dataset and KNN

	Precision	Recall	F1_Score
1	100	90	94.737
2	100	100	100
3	100	100	100
4	100	81.818	90
5	92.857	100	96.296
6	100	100	100
7	100	90.909	95.238

- **CK dataset and SVM:**

  Total accuracy is 98.2766

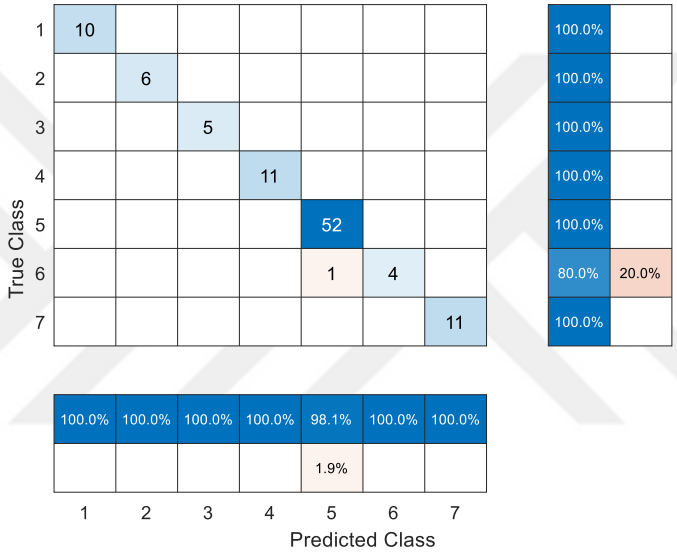


Figure 4.18. SqueezeNet with CK dataset and SVM Confusion matrix

Table 4.17. SqueezeNet response with CK dataset and SVM

	Precision	Recall	F1_Score
1	100	100	100
2	100	100	100
3	100	100	100
4	100	100	100
5	98.11	100	99.048
6	100	80	88.88
7	100	100	100

- **CK dataset and TREE:**

  Total accuracy is 85.4921

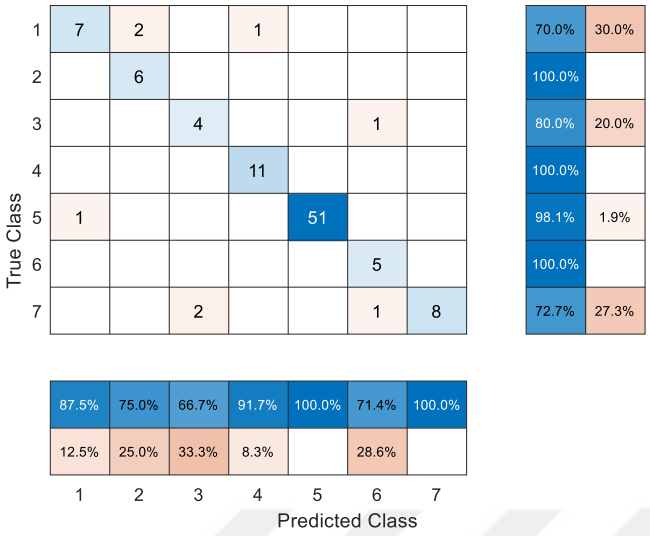


Figure 4.19. SqueezeNet with CK dataset and TREE Confusion matrix

Table 4.18. SqueezeNet response with CK dataset and TREE

	Precision	Recall	F1_Score
1	87.5	70	77.778
2	75	100	85.714
3	66.667	80	72.727
4	91.667	100	95.652
5	100	98.077	99.029
6	71.429	100	83.333
7	100	72.727	84.211

4.2.4. Real-time GUI response

The following Figure (4.20, 4.21,4.22,4.23,4.24,4.25,4.26) shows the real-time recognition of different facial expressions which gives a realistic and smooth recognition.

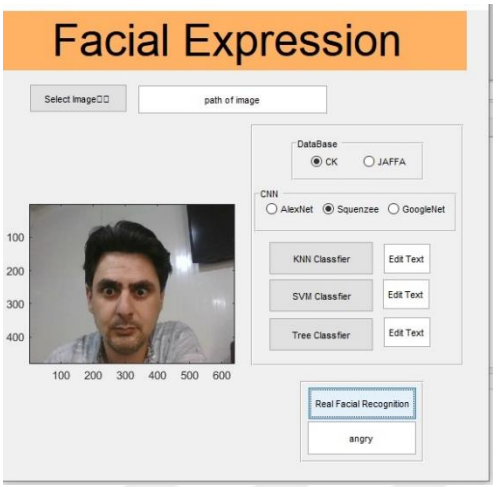


Figure 4.20. Angry Face

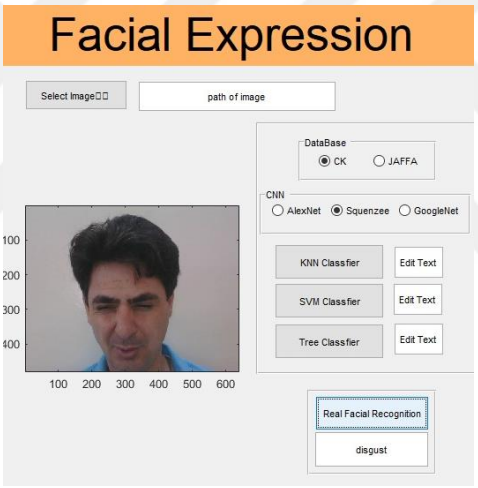


Figure 4.21. Disgust Face

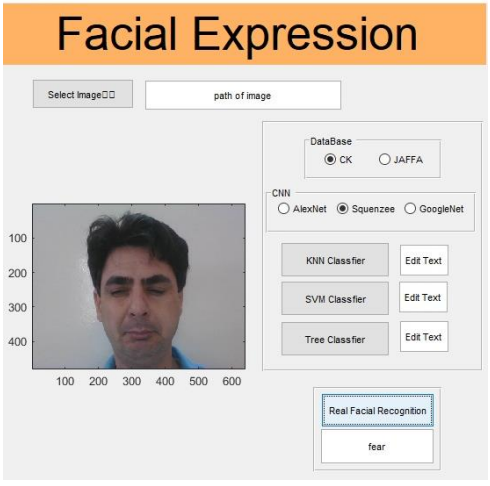


Figure 4.22. Fear Face

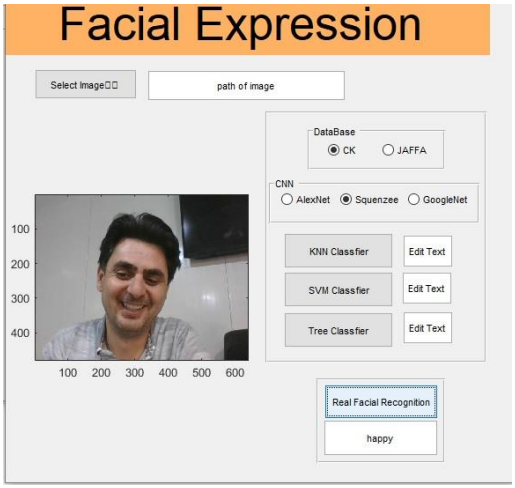


Figure 4.23. Happy Face

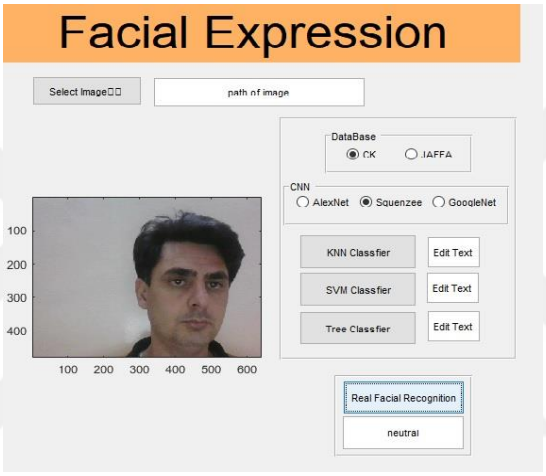


Figure 4.24. Neutral Face

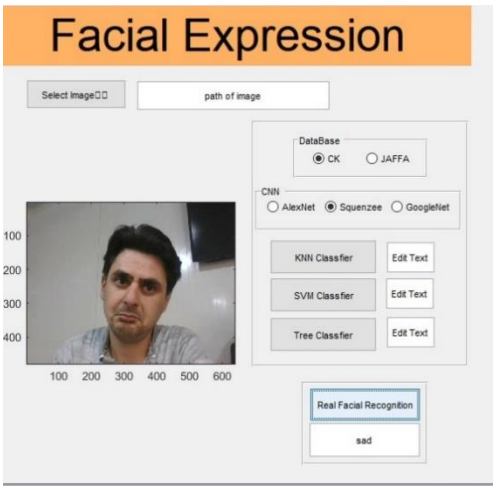


Figure 4.25 Sad Face

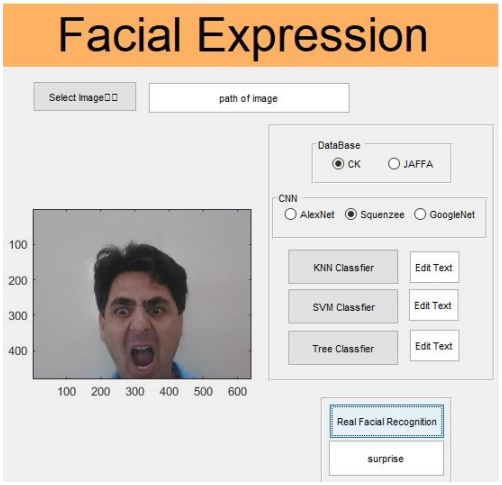


Figure 4.26. Surprise Face

CONCLUSİON

This research proposed a new FER system based on deep learning to analyze facial expressions. This work aimed to use advanced AI techniques to improve the currently available systems that are related to human facial emotion detection using AI in deep learning. The research aims to explore and analyze the most popular techniques in deep learning.

The system is designed based on a CNN architecture specific to the facial expression problem. Where facial images were analyzed using pre-trained high-level features extraction deep learning algorithms (AlexNet, googlenet squeezenet) combined with another classification neural network. The most useful information is extracted from the 'fc7' layer for AlexNet, 'fc8 ' for GoogleNet, and 'fire9-concat' for SqueezeNet. The output of these layers is forwarded into classification using (KNN, SVM, or Decision TREE) neural network whereas these networks are trained to classify (Anger, Disgust, Fear, Happy, Sad, Surprise, and Neutral) classes as final output. The system is trained nine times for each dataset using these features extraction and classification algorithm. The purpose of these studies is to improve the quality of extracted features and speed up the process.

The system was tested on two standard databases, which are Japanese Female Facial Expression (JAFFE) and Cohn-Kanade (CK), and where the data was used in The training phase in different sizes and percentages, amounting to 80% percent of the total data volume

The best classification process was obtained when using (alexnet) network trained in feature extraction process and using KNN In the process of discrimination where a result of 100% was obtained. This Facial emotions expression recognition research comprises the classification of face emotions of humans by expressions on their faces.

In future studies, we recommend

- involving multiple face detection and FER classification for each person.
- improving classification techniques using networks such as Recurrent Neural Networks and deep webs of belief. With algorithms such as (VGG, and RESNET) for the classification of more facial and body movements.

REFERENCES

[1]. Ueda, J., & Okajima, K. (2019). Face morphing using average face for subtle expression recognition. 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), 187–192.

[2]. Theekapun, C., Tokai, S., & Hase, H. (2007). Facial expression recognition from a side view face by using face plane. 2007 International Conference on Wavelet Analysis and Pattern Recognition, 3, 1096–1101.

[3]. Khorsheed, J. A., & Yurtkan, K. (2016). Analysis of Local Binary Patterns for face recognition under varying facial expressions. 2016 24th Signal Processing and Communication Application Conference (SIU), 2085–2088.

[4]. Carcagnì, P., Del Coco, M., Leo, M., & Distante, C. (2015). Facial expression recognition and histograms of oriented gradients: a comprehensive study. SpringerPlus, 4(1), 1–25.

[5]. Petpairote, C., Madarasmi, S., & Chamnongthai, K. (2017). A pose and expression face recognition method using transformation based on single face neutral reference. 2017 Global Wireless Summit (GWS), 123–126.

[6]. Smeets, D., Fabry, T., Hermans, J., Vandermeulen, D., & Suetens, P. (2010). Fusion of an isometric deformation modeling approach using spectral decomposition and a region-based approach using ICP for expression-invariant 3D face recognition. 2010 20th International Conference on Pattern Recognition, 1172–1175.

[7]. Majaj, N. J., & Pelli, D. G. (2018). Deep learning—Using machine learning to study biological vision. Journal of Vision, 18(13), 2.

[8]. Kok, J. N., Boers, E. J., Kosters, W. A., Van der Putten, P., & Poel, M. (2009). Artificial intelligence: definition, trends, techniques, and cases. Artificial Intelligence, 1, 270–299.

[9]. Bre, F., Gimenez, J. M., & Fachinotti, V. D. (2018). Prediction of wind pressure coefficients on building surfaces using artificial neural networks. Energy and Buildings, 158, 1429–1441.

[10]. Talayero, A. P., Yürüşen, N. Y., Ramos, F. J. S., & Gastón, R. L. (2022). Machine learning based met data anomaly labelling. Journal of Physics: Conference Series, 2257(1), 12015.

[11]. Mahesh, B. (2020). Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet], 9, 381–386.

[12]. Cha, S.-H., & Tappert, C. C. (2009). A genetic algorithm for constructing compact binary decision trees. Journal of Pattern Recognition Research, 4(1), 1–13.

[13]. Biau, G. (2012). Analysis of a random forests model. The Journal of Machine Learning Research, 13(1), 1063–1095.

[14]. Tan, Y., & Zhang, G.-J. (2005). The application of machine learning algorithm in underwriting process. 2005 International Conference on Machine Learning and Cybernetics, 6, 3523–3527.

[15]. Goularas, D., & Kamis, S. (2019). Evaluation of deep learning techniques in sentiment analysis from twitter data. 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), 12–17.

[16]. Vembandasamy, K., Sasipriya, R., & Deepa, E. (2015). Heart diseases detection using Naive Bayes algorithm. International Journal of Innovative Science, Engineering & Technology, 2(9), 441–444.

[17]. Wang, L. (2005). Support vector machines: theory and applications (Vol. 177). Springer Science & Business Media.

[18]. Fairbanks, M. (n.d.). Application of Deep Learning with Brain Computer Interfaces.

[19]. Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2020). Fundamental concepts of convolutional neural network. In Recent trends and advances in artificial intelligence and Internet of Things (pp. 519–567). Springer.

[20]. Sun, M., Song, Z., Jiang, X., Pan, J., & Pang, Y. (2017). Learning pooling for convolutional neural network. Neurocomputing, 224, 96–104.

[21]. Sun, D., Wulff, J., Sudderth, E. B., Pfister, H., & Black, M. J. (2013). A fully-connected layered model of foreground and background flow. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2451–2458.

[22]. Silva Segundo, L. B. da. (2018). Classificação de Personagens Animados usando Redes Neurais Convolucionais Profundas Pré-treinadas e Fine-tuning.

[23]. Wang, M., & Deng, W. (2021). Deep face recognition: A survey. Neurocomputing, 429, 215–244.

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

[25]. Zhou, B., Khosla, A., Lapedriza, A., Torralba, A., & Oliva, A. (2016). Places: An image database for deep scene understanding. ArXiv Preprint ArXiv:1610.02055.

[26]. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., & Torralba, A. (2017). Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(6), 1452–1464.

[27]. Taheri, S., & Toygar, Ö. (2019). On the use of DAG-CNN architecture for age estimation with multi-stage features fusion. Neurocomputing, 329, 300–310.

[28]. SOUZA, C. O. de. (2020). Usando convolução separável em profundidade na otimização da arquitetura SqueezeNet. Universidade Federal de Pernambuco.

[29]. Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. ArXiv Preprint ArXiv:1602.07360.

[30]. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25.

[31]. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. ArXiv Preprint ArXiv:1409.1556.

[32]. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–9.

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

[34]. Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 7132–7141.

[35]. Islam, S. M. S., Mahmood, H., Al-Jumaily, A. A., & Claxton, S. (2018). Deep learning of facial depth maps for obstructive sleep apnea prediction. 2018 International Conference on Machine Learning and Data Engineering (ICMLDE), 154–157.

[36]. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., & Song, L. (2017). Sphereface: Deep hypersphere embedding for face recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 212–220.

[37]. Hussein, E. S., Qidwai, U., & Al-Meer, M. (2020). Emotional stability detection using convolutional neural networks. 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), 136–140.

[38]. Nie, Z. (2020). Research on facial expression recognition of robot based on CNN convolution neural network. 2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), 1067–1070.

[39]. Kaviya, P., & Arumugaprakash, T. (2020). Group facial emotion analysis system using convolutional neural network. 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 643–647.

[40]. Fei, Z., Yang, E., Li, D. D.-U., Butler, S., Ijomah, W., Li, X., & Zhou, H. (2020). Deep convolution network based emotion analysis towards mental health care. Neurocomputing, 388, 212–227.

[41]. Abdullah, S. M. S., & Abdulazeez, A. M. (2021). Facial expression recognition based on deep learning convolution neural network: A review. Journal of Soft Computing and Data Mining, 2(1), 53–65.

[42]. Li, J., Jin, K., Zhou, D., Kubota, N., & Ju, Z. (2020). Attention mechanism-based CNN for facial expression recognition. Neurocomputing, 411, 340–350.

[43]. Meryl, C. J., Dharshini, K., Juliet, D. S., Rosy, J. A., & Jacob, S. S. (2020). Deep learning based facial expression recognition for psychological health analysis. 2020 International Conference on Communication and Signal Processing (ICCSP), 1155–1158.

[44]. Jiang, P., Liu, G., Wang, Q., & Wu, J. (2020). Accurate and reliable facial expression recognition using advanced softmax loss with fixed weights. IEEE Signal Processing Letters, 27, 725–729.

45. Agrawal, A., & Mittal, N. (2020). Using CNN for facial expression recognition: a study of the effects of kernel size and number of filters on accuracy. The Visual Computer, 36(2), 405–412.

[46]. Chen, J., Lv, Y., Xu, R., & Xu, C. (2019). Automatic social signal analysis: Facial expression recognition using difference convolution neural network. Journal of Parallel and Distributed Computing, 131, 97–102.

[47]. Mohan, K., Seal, A., Krejcar, O., & Yazidi, A. (2020). Facial expression recognition using local gravitational force descriptor-based deep convolution neural networks. IEEE Transactions on Instrumentation and Measurement, 70, 1–12.

[48]. Cheng, S., & Zhou, G. (2020). Facial expression recognition method based on improved VGG convolutional neural network. International Journal of Pattern Recognition and Artificial Intelligence, 34(07), 2056003.

[49]. Razazzadeh, N., & Khalili, M. (2015). A high performance algorithm to diagnosis of skin lesions deterioration in dermatoscopic images using new feature extraction. 2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE), 1207–1212.

[50]. Li, G., Lin, Y., & Qu, X. (2021). An infrared and visible image fusion method based on multi-scale transformation and norm optimization. Information Fusion, 71, 109–129.

[51]. Li, N., Zhao, L., Chen, A.-X., Meng, Q.-W., & Zhang, G.-F. (2009). A new heuristic of the decision tree induction. 2009 International Conference on Machine Learning and Cybernetics, 3, 1659–1664.

[52]. Kirelli, Y., & Arslankaya, S. (2020). Sentiment analysis of shared tweets on global warming on twitter with data mining methods: a case study on Turkish language. Computational Intelligence and Neuroscience, 2020.

[53]. Roy, S., Menapace, W., Oei, S., Luijten, B., Fini, E., Saltori, C., Huijben, I., Chennakeshava, N., Mento, F., & Sentelli, A. (2020). Deep learning for classification and localization of COVID-19 markers in point-of-care lung ultrasound. IEEE Transactions on Medical Imaging, 39(8), 2676–2687.

[54]. Kaskavalci, H. C., & Gören, S. (2019). A deep learning based distributed smart surveillance architecture using edge and cloud computing. 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), 1–6.

[55]. Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. 2010 Ieee Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, 94–101.

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

[57]. Singh, P., Singh, N., Singh, K. K., & Singh, A. (2021). Chapter 5 - Diagnosing of disease using machine learning (K. K. Singh, M. Elhoseny, A. Singh, & A. A. B. T.-M. L. and the I. of M. T. in H. Elngar (eds.); pp. 89–111). Academic Press. https://doi.org/https://doi.org/10.1016/B978-0-12-821229-5.00003-3

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Karrar Ismael Mohammed ALLAW
Doğum Yeri	
Doğum Tarihi	
Uyruğu	❒ T.C. ☑ Diğer:

Eğitim Bilgileri	
Lisans	
Üniversite	Kerbela Üniversitesi
Fakülte	Bilim Fakültesi
Bölümü	Bilgisayar Bölümü
Mezuniyet Yılı	2009

Yüksek Lisans	
Üniversite	Kırşehir Ahi Evran Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	İleri Teknolojiler Anabilim Dalı
Programı	İleri Teknolojiler Tezli Yüksek Lisans
Mezuniyet Tarihi	2022