5-2010

A Study of Three Artificial Neural Networks Models' Ability to Identify Emotions from Facial Images

Timothy Scott Hyde
Columbus State University

A STUDY OF THREE ARTIFICIAL NEURAL NETWORKS MODELS' ABILITY TO IDENTIFY EMOTIONS FROM FACIAL IMAGES

Timothy Scott Hyde

Columbus State University

D. Abbott Turner College of Business and Computer Science

School of Computer Science

The Graduate Program in Applied Computer Science

A Study of Three Artificial Neural Networks Models' Ability to Identify Emotions from Facial Images

A Thesis in

Applied Computer Science

by

Timothy Scott Hyde

Submitted in Partial Fulfillment
of the Requirements
for the Degree of
Master of Science

May 2010

I have submitted this thesis in partial fulfillment of the requirements for the degree of Master of Science

6/14/2010
Date

Timothy Scott Hyde

We approve the thesis of Timothy Scott Hyde as presented here.

6/14/2010
Date

Dr. Shamim Khan Associate Professor of Computer Science, Thesis Advisor

6/14/2010
Date

Dr. John Theis, Associate Professor, Accounting & Finance

6/14/2010
Date

Dr. Edward Bosworth, Associate Professor of Computer Science

Abstract

Facial expressions conveying emotions are vital for human communication. They are also important in the studies of human interaction and behavioral studies. Recognition of emotions, using facial images, may provide a fast and practical approach that is noninvasive. Most previous studies of emotion recognition through facial images were based on the Facial Action Coding System (FACS). The FACS, which was developed by Ekman and Freisen in 1978, was created to identify different facial muscular actions. Previous artificial neural network-based approaches for classification of facial expressions focused on improving one particular neural network model for better accuracy. The purpose of this present study was to compare different artificial neural network models, and determine which model was best at recognizing emotions through facial images. The three neural network models were:

1. The Hopfield network
2. The Learning Vector Quantization network
3. Multilayer (Feedforward) back-propagation network

Several facial parameters were extracted from facial images and used in training the different neural network models. The best performing neural network was the Hopfield network at 72.50% accuracy. Next, the facial parameters were tested for their significance in identifying facial expressions and a subset of the original facial parameters was used to retrain the networks. The best performing network using the subset of facial parameters was the LVQ network at 67.50% accuracy. This study has helped to understand which neural network model was best at identifying facial

expression and to understand the importance of having a good set of parameters representing the facial expression. This study has shown that more research is needed to find a good set of parameters that will improve the accuracy of emotion identification using artificial neural networks.

Table of Contents:

List of Tables

List of Figures

Acknowledgements

I would like to thank my supervisor, Dr. Shamim Khan, whose encouragement, guidance and support from the beginning to the end enabled me to develop an understanding of the subject and complete this difficult project. He has been a constant source of encouragement and support throughout our work together. I would like to thank Dr. Edward Bosworth and Dr. John Theis for kindly agreeing to serve on my committee and offer their guidance.

I would like to thank Joni Christiansen for her time and editing abilities. I would like to thank Craig Tuin for his support, and for traveling with me to my defense. I would also like to thank my friends Chris, Dave, Jeff, Randy, Ray, and Satyakant for their support.

Lastly, I would like to thank my family for their constant support during the completion of this thesis.

Chapter 1: Background and Objectives

Visual communication is an important part of human communication as it helps convey meaning to other forms of communication. Charles Darwin was the first to recognize human faces displaying emotions (Darwin, 1872/1965) and Ekman connected emotions to facial expressions using a facial action coding system (FACS)(Ekman, Friesen, & Hager, 2002). Automation of reading, and the interpretation of facial expressions into emotions, have both been researched, as of late. The recent aim has been for computers to read a facial image and be able to identify correctly the emotion being displayed to it. Computers recognizing emotions from facial expressions, has been a desired implementation to improve Human-Computer interaction. Recently, there has been a growing interest in automating facial expression recognition. Research, present and historically, has been conducted in this area by scientists in the fields of computer science, engineering, neuroscience, and psychology. The automation of recognizing facial expressions can be applied to human-computer interaction, stress-monitoring systems, low-bandwidth videoconferencing, human behavior analysis, and to help humans who are unable to identify emotions in others, i.e. humans with blindness or autism.

1.1 Research Objectives

The objectives researched to further knowledge in the use of artificial neural networks to determine emotions from facial images:

- Comparative analysis of the different neural networks

The aim was to understand which one particular neural network model is better at identifying four most common emotions visible in facial expressions – anger, fear, happiness and surprise. Furthermore, to determine which neural network model performed the best and why. The three types of artificial neural networks selected for the study are *recurrent, feedforward,* and *competitive* networks. These neural network models differ in their architecture, training and operation.

- Comparative analysis of the influence of each parameter

The aim was to understand which parameters from facial images best help determine the emotion being displayed.

Chapter 2: Literature Review

In his book, *The Expression of the Emotions in Man and Animals* (Darwin, 1872/1965), Darwin suggests three principles to account for most of the involuntary expressions and gestures used by humans and lower animals:

- The principle of serviceable associated habits
- The principle of antithesis
- The principle of actions due to the constitution of the nervous system, independently from the first of the will, and independently to a certain extent of habit

Darwin's book also included eight chapters on human emotions, discussing facial expressions and the muscles involved in making the expressions. Ekman then outlined six basic emotions defined by distinct facial expressions (Ekman et al., 2002). The six basic emotions are happiness, sadness, surprise, fear, anger, and disgust. Ekman and Friesen first proposed a facial action coding system (FACS) to measure facial behavior. FACS determines facial behavior by measuring differences in visible muscular movements by using "action units" (AUs) (Ekman et al., 2002).

FACS was developed to determine changes in the appearance of the face by the contraction of each facial muscle, both singly, and in combination with, other muscles. FACS uses Action Units (AUs) instead of muscles, for two reasons:

1. A single AU may be a combination of more than one muscle because the changes in appearance produced by one muscle could not be distinguished.

2. A single muscle would produce appearance changes that were separated into multiple AUs representing relatively independent actions of different parts of the muscle.

(Ekman, Friesen, & Hager, 2002)

Forty-four AUs were defined, each corresponding to the independent movement of a single or group of muscles. A trained human FACS coder decomposes an observed facial expression into AUs, which then describes the expression. This requires many hours of training and is not very practical.

A possible solution to the time constraint is an automatic facial expression system. Such a system will examine an image, look for and identify the facial image, extract the facial image, and finally interpret the facial expression. An automatic facial expression system has three stages (see Figure 2-1 below):

1. Face detection – The face detection stage locates and captures the facial region from an image.
2. Facial feature data extraction – This phase takes the facial image and converts it into a format suitable for processing.
3. Facial expression classification – This stage identifies the expression/emotion from the extracted data.

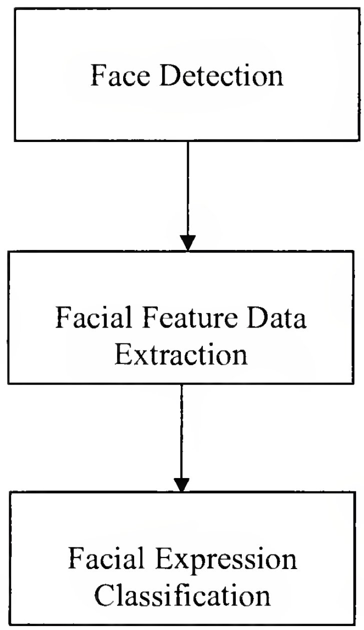

Figure 2-1 The three stages to an automatic facial expression system

2.1 Approaches to Facial Expression Detection

There have been different approaches to automate facial expression detection. Two basic approaches are algorithmic and non-algorithmic (based on the use of artificial neural networks). These have been described below.

- The Algorithmic approach uses algorithms to examine the extracted facial image data. It compares the data with a known facial expression and determines the emotion displayed from the extracted facial image. The steps in the algorithm are known and can be modified. This allows others to easily make improvements to the algorithm. Methods have been tested for faster automation, such as, measurement of facial motion through optic flow (Mase, 1991), and analysis of surface textures based on principal component analysis (PCA) (Rosenblum, Yacoob, & Davis, 1996). Some of the newer techniques include using Gabor wavelets (Lyons, Akamatsu, Kamachi, & Gyoba, 1998), linear discriminant analysis (Mase, 1991), local feature analysis (Lanitis, Taylor, & Cootes, 1997), independent component analysis (Bartlett & Sejnowski, 1997), use of facial expression with speech patterns (Busso et al., 2004), and transforming the feature vector data into tree structure representation (Wong & Cho, 2006).

- In the artificial neural network approach, the system learns to identify facial expression from a training set of data presented during the learning phase. During the learning phase, the weights interconnecting neurons adjust to the patterns in the training set, until the network has learned the patterns. With successful training, the network is able to classify a different set of data. Research done so far with artificial neural network-based facial expression detection is described below.

A three multi-layer perceptrons (MLPs) used to recognize action units in the eyebrows, the eyes, and the mouth regions. Su et al (Su, Hsieh, & Huang, 2007) trains a network reading the output from the three MLPs and identifies one of five different emotions based on the three inputs. Another approach (Kulkarni, 2006) uses multiple neural networks, choosing the best ones to identify the emotion. Facial parameters from a facial image were extracted and used to train multiple, generalized, and specialized neural networks. The generalized networks try to identify the emotion based on the facial parameters. If the networks were unable to identify the emotion, then the data was fed to the specialized networks. The emotions the generalized networks could not classify sometimes were anger, disgust, fear and sadness. The best performing generalized and specialized networks were joined to form an integrated committee neural network system that would determine final selection of emotion shown in the facial image.

Chapter 3: Artificial Neural Networks

Artificial neural networks (ANN) are made up of very simple (and highly interconnected) processors, called neurons, similar to the neurons in the human brain. The connections between the neurons are numerically weighted links that pass signals from one neuron to another. The numerical weight of a link expresses the strength of each input. Each neuron takes the signals from its input links, computes the weighted sum, and then produces an output based on this weighted sum and a non-linear transfer function. A commonly used transfer function is the step function; in which, when the net input is greater than, or equal to the threshold, the neuron is activated, and an output of +1 is produced. A neuron receives any number of input signals from its connections, but only produces one output signal that, in turn, is transmitted through the neuron's outgoing connection. The continuous adjustment of the weights in a neural network constitute learning. The four most common transfer functions are:

- Step and Sign – also known as hard limit functions. These compare input values to a threshold value.
- Sigmoid- This function transforms the input value into a value between 0 and 1.
- Liner activation – The output of this function is equal to the neuron's weighted input. (Negnevistky, 2005)

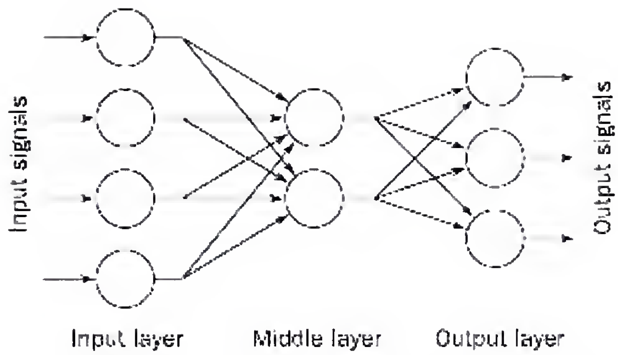

Figure 3-1 Architecture of a typical artificial neural network (Negnevistky, 2005, p. 167)

In this study, three popular ANN models were tested for a relative comparison of their performance. These are:

- Hopfield Network - The Hopfield network is a recurrent network that has feedback loops from it's outputs to it's inputs. When applying a new input, the network output is calculated and fed back to adjust the input. This process is repeated until the output becomes constant (Negnevistky, 2005, p. 188).

  The Hopfield network has three main steps:

  1. Storage – Development of a Hopfield network stores a set of patterns, known as fundamental memories or training vectors.
  2. Testing – Confirmation is needed of the Hopfield networks' ability to recall the patterns. The patterns are inputted into the network. If the network correctly identifies all of the patterns then one may continue to the last step.
  3. Retrieval – The Hopfield network is introduced to unknown vectors. Typically, the unknown vectors represent an incomplete,

or corrupted, version of the patterns. The network retrieves one of the patterns from its memory that is closest to the inputted vector. This is determined by measuring the number of elements of the vectors that differ. The measurement of distance used for such vectors is called the *Hamming distance*. Hence, the inputted vector will map to the pattern whose Hamming distance is the least from the input vector (Coppin, 2004, p. 307).

Two potential problems with the Hopfield network are false memories and storage capacity. False memories happen when the stable state does not represent one of the fundamental memories. Storage capacity is the largest number of fundamental memories that can be stored and retrieved correctly. This can be calculated with the following formula: Maximum number of stored patterns = .15 x number of neurons.

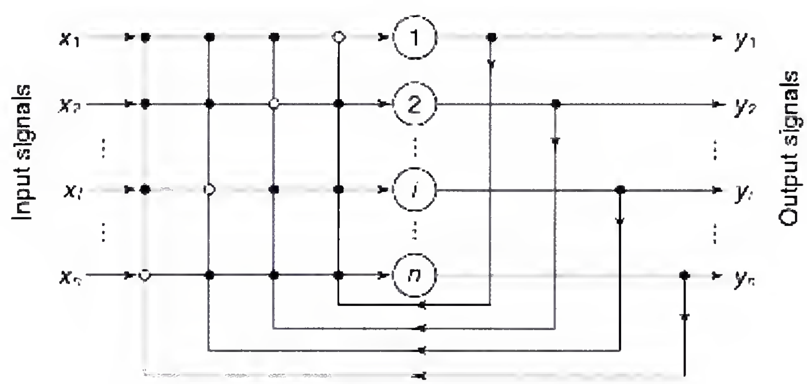

Figure 3-2 Single-layer n-neuron Hopfield network (Negnevistky, 2005, Figure 6.17)

- Multilayer (Feedforward) back-propagation network – The network consists of three or more layers.
    - Input layer – Accepts the input signals and redistributes them to the hidden layer. The input layer rarely includes any computing neurons.
    - One or more hidden layers - Neurons in the hidden layer detect the features; the weights of the neurons represent the features hidden in the input patterns. These features are then used by the output layer in determining the output pattern.
    - Output layer – The output layer accepts output signals from the hidden layer and establishes the output pattern from the entire network. During training, if the output pattern does not match the pattern expected for a given input, the difference between the two, known as the error, is propagated back through the network, adjusting the weights. (Negnevistky, 2005, p. 175)

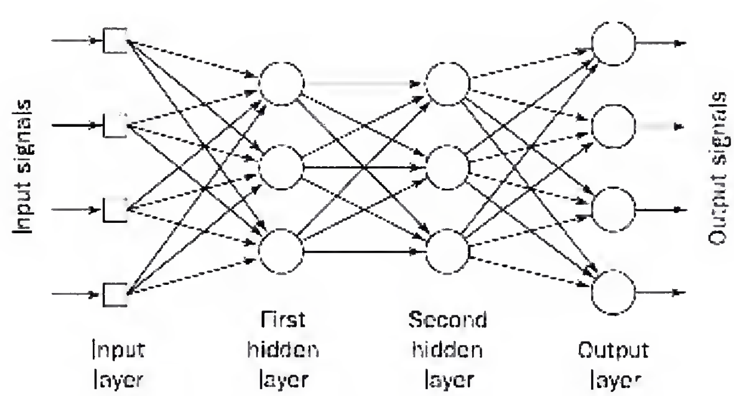

Figure 3-3 Multilayer perceptron with two hidden layers (Negnevistky, 2005, Figure 6.8)

- Learning Vector Quantization (LVQ) – This is a supervised competitive learning network where the neurons compete to be activated by the input vector. Each neuron in the input layer is connected to each neuron in the output layer, also known as the learning vector quantization layer or competitive layer. The neuron in the competitive layer, whose weight vector is closest to the input vector, is the winning neuron. The weight vector of this neuron, as well as those of its neighbors, is adjusted so that it will be even closer to the input vector. This increases the chance of the neuron winning the competition next time the same input vector is introduced. Finally, a linear layer transforms the competitive layer's classes into target classifications defined by the user. In Figure 3.3 below, the peach colored neuron has won the competition for the input vector and will have its weight adjusted to match the input vector. The red colored neurons, neighbors to the peach colored neuron, will also have their weights adjusted. (Hertz, Krogh, & Palmer, 1991)

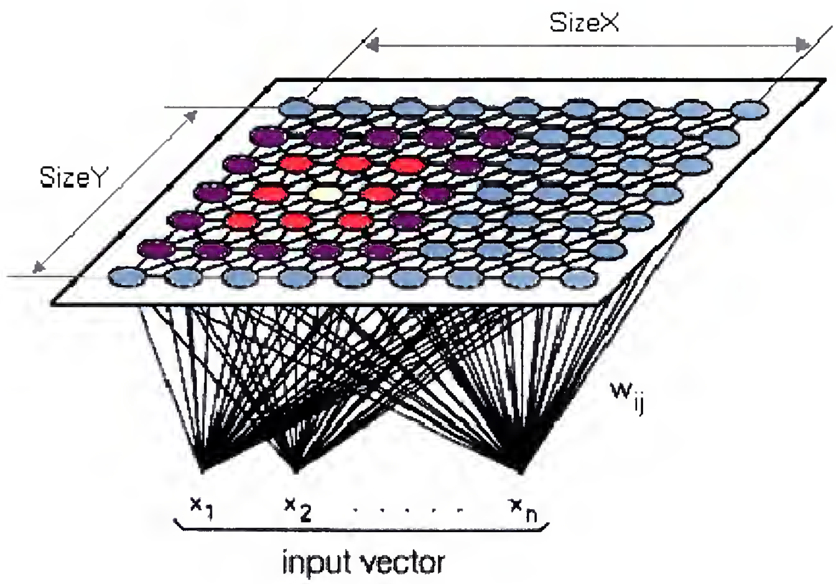

Figure 3-4 A Learning Vector Quantization (LVQ) neural network (SDL Component Suite, 2008)

- **Peach colored neuron – winner of the competition.**
- **Red colored neurons – neighbors of the winning neuron.**
- **Purple colored neurons - neighborhood of the winning neuron.**

Chapter 4: The Experiment

The principal objective of this study was to develop specialist artificial neural networks (ANNs) to identify different emotions from facial images. To meet this goal, fifteen facial parameters were used to measure facial movements that produced different emotional expressions, and identify the emotion. These fifteen facial parameters were the same ones used by Kulkarni in his study (Kulkarni, 2006). I chose to use the same parameters as Kulkarni, as Kulkarni had success with them. Also, the parameters did correspond well with the action units. Facial expression images were obtained from the Cohn-Kanade database (Kanade, Cohn, & Tian, 2000). The database contained around 2000 images from 97 subjects expressing six basic emotions (anger, disgust, fear, sad, happiness and surprise).

Two types of parameters were used—real-valued and binary. Real-valued parameters measure the distance, in pixels, between two objects and give a definite value. The real-valued parameters were normalized to a value between 0 and 1 before being applied as input to the networks. The three different network models selected for this study – the Hopfield network, the Multilayer Backpropagation network and the LVQ network - were trained with all fifteen parameters, described below, as inputs and the corresponding facial expression as targets. Next, analysis was done on the fifteen parameters to determine a better set of parameters to improve performance of the neural networks. The Principal Component Analysis, explained in section 4.8, was used to determine the best set of parameters. Eight of these were real-valued and two were binary. The three neural networks were retrained and tested using the new set of parameters. A comparison was made of the network models performances using all fifteen parameters as well as using an optimal set of parameters. Descriptions of the parameters used in this study are given below.

4.1 Real -valued Parameters

The eight real-valued parameters used are:

1. ***Eyebrow Raise distance*** – The distance between common point of upper and lower eyelid and the lower central tip of the eyebrow. A decrease in this distance would indicate a presence of action code 4, used to detect one of the following expressions: anger, fear, or sadness (see Figure 4-1).
2. ***Upper eyelid to eyebrow distance*** – The distance between the upper eyelid and eyebrow surface. A decrease in this distance would indicate a presence of action code 5, used to detect one of the following expressions: anger, fear, or surprise (see Figure 4-1).
3. ***Inter-eyebrow distance*** – The distance between the lower central tips of both the eyebrows. A decrease in this distance would indicate a presence of any of the following action codes; 1, 2, and/or 4, used to detect one of the following expressions: anger, fear, sadness, or surprise (see Figure 4-1).
4. ***Upper eyelid – lower eyelid distance*** – The distance between the upper eyelid and lower eyelid. A decrease in this distance would indicate a presence of action code 6, used to detect one of the following expressions: happiness, or sadness (see Figure 4-1).

5. ***Top lip thickness*** – The measure of the thickness of the top lip. Increase in distance would indicate a presence of action code 10, used to detect one of the following expressions: anger or disgust (see Figure 4-1).

6. ***Lower lip thickness*** – The measure of the thickness of the lower lip. Increase in distance would indicate a presence of any of the following action codes; 25, 26, or 27 used to detect one of the following expressions: anger, fear, or surprise. A decrease in this distance would indicate a presence of any of the following action codes; 23, or 24 used to detect anger (see Figure 4-1).

7. ***Mouth width*** – The distance between the tips of the lip corner. Increase in distance would indicate a presence of action code 12, used to detect happiness (see Figure 4-1).

8. ***Mouth opening*** – The distance between the lower surface of top lip and upper surface of lower lip. Increase in distance would indicate a presence of any of the following action codes; 25, 26, or 27 used to detect one of the following expressions: anger, fear, or surprise (see Figure 4-1). A decrease in this distance would indicate a presence of any of the following action codes; 15, 16, or 17 used to detect one of the following expressions: anger, disgust, or sadness.

Figure 4-1 Real-valued measures from a sample neutral expression image. 1-eyebrow raise distance, 2-upper eyelid to eyebrow distance, 3-inter-eyebrow distance, 4-upper eyelid to lower eyelid distance, 5-top lip thickness, 6-lower lip thickness, 7-mouth width, 8-mouth opening. (Facial expression image from the Cohn-Kanade database. (Kanade et al., 2000) Used with permission)

4.2 Binary Parameters

The binary parameters show the presence or non-presence of a facial feature—where zero represents non-presence and one represents the presence of the facial feature. The seven binary parameters used are:

1. ***Upper teeth visible*** – Presence or absence of visibility of the upper teeth. An absence of visibility would indicate action code 23, or 24, used to detect anger (see Fig. 4-2a). Presence of visibility would indicate any emotion.

2. ***Lower teeth*** - Presence or absence of visibility of the lower teeth. An absence of visibility would indicate action code 23 and 24, used to detect anger (see Fig. 4-2a). Presence of visibility would indicate any emotion.

3. ***Forehead lines*** - Presence or absence of wrinkles in the upper part of the forehead. A presence of wrinkles would indicate action code 2, used to detect fear or surprise (see Fig. 4-2b).

4. ***Eyebrow Lines*** – Presence or absence of wrinkles in the region above the eyebrows. A presence of wrinkles would indicate action code 4, used to detect anger, fear, or sadness (see Fig. 4-2b).

5. ***Nose Lines*** – Presence or absence of wrinkles in the region between the eyebrows extending over the nose. A presence of wrinkles would indicate action code 9, used to detect disgust (see Fig. 4-2c).

6. ***Chin Lines*** – Presence or absence of wrinkles or lines on the chin region just below the lower lip. A presence of wrinkles would indicate action code 17, used to detect one of the following; anger or sadness (see Fig. 4-2d).

7. ***Nasolabial lines*** – Presence or absence of thick lines on both sides of the nose extending until the upper lip. A presence of wrinkles would indicate action code 10, used to detect anger or disgust (see Fig. 4-2c).

(a) (b)

(c) (d)

Figure 4-2 Binary measures from sample expression images. 1-upper teeth visible, 2- lower teeth visible, 3-forehead lines, 4-eyebrow lines, 5-nose lines, 6-chin lines, 7- nasolabial lines. (Facial expression image from the Cohn-Kanade database. (Kanade et al., 2000) Used with permission)

4.3 Parameter Extraction

Real-valued and binary parameters were extracted from the facial images from sixty-five subjects constituting of 243 images. The training dataset was created from 40 different subjects with a total of 162 facial images.

All of the training dataset were used to extract the eight real-valued parameters. Table 4-1 shows the average value for all of the eight real-valued parameters, from the training set data, by facial expression and all facial expressions combined.

Table 4-1 Average value from the training set data of the eight real-valued parameters for different expressions.

Real-valued Parameter	Average value by facial expression - Training set				
	Anger	Fear	Happiness	Surprise	All
Eyebrow raised distance	30.23	38.08	35.86	49.48	38.52
Upper eyelid-eyebrow distance	17.18	24.58	21.26	33.05	24.23
Inter eyebrow distance	39.4	41.26	44.83	48.71	43.66
Upper eyelid-lower eyelid distance	19.88	16.24	21.12	19.33	19.2
Top lip thickness	7.93	7.87	9.12	9.07	8.52
Lower lip thickness	10.3	15.16	13	18.64	14.3
Mouth width	80.73	88.66	104.79	69.67	85.69
Mouth opening	3.38	3.97	10.86	11.17	7.48

All of the training dataset were used to extract the seven binary parameters. The features represented by these parameters were either present or absent in each expression. Table 4-2 shows the average value for all the eight binary parameters, from the training set data, by facial expression and all facial expressions combined.

Table 4-2 Average value from the training set data of the seven binary parameters for different expressions.

Binary Parameter	Average value by facial expression - Training set				
	Anger	Fear	Happiness	Surprise	All
Upper teeth visible	0.15	0.89	0.79	0.55	0.59
Lower teeth visible	0.1	0.71	0.33	0.43	0.39
Forehead lines	0.15	0.18	0	0.55	0.22
Eyebrow lines	0.55	0.21	0	0	0.19
Nose lines	0.65	0.34	0.24	0.1	0.33
Chin lines	0.63	0.21	0.33	0.12	0.32
Nasolabial lines	0.83	0.84	0.95	0.52	0.78

The scatter plot of the parameter *eyebrow raised distance* in different expressions is shown in Figure 4-3. It was observed that this parameter's values were, on average, the lowest for the expression of anger. Also observed, the parameter values were overall the highest in the expression of surprise. Finally, it was observed, that the parameter's values for happiness and fear were similar.

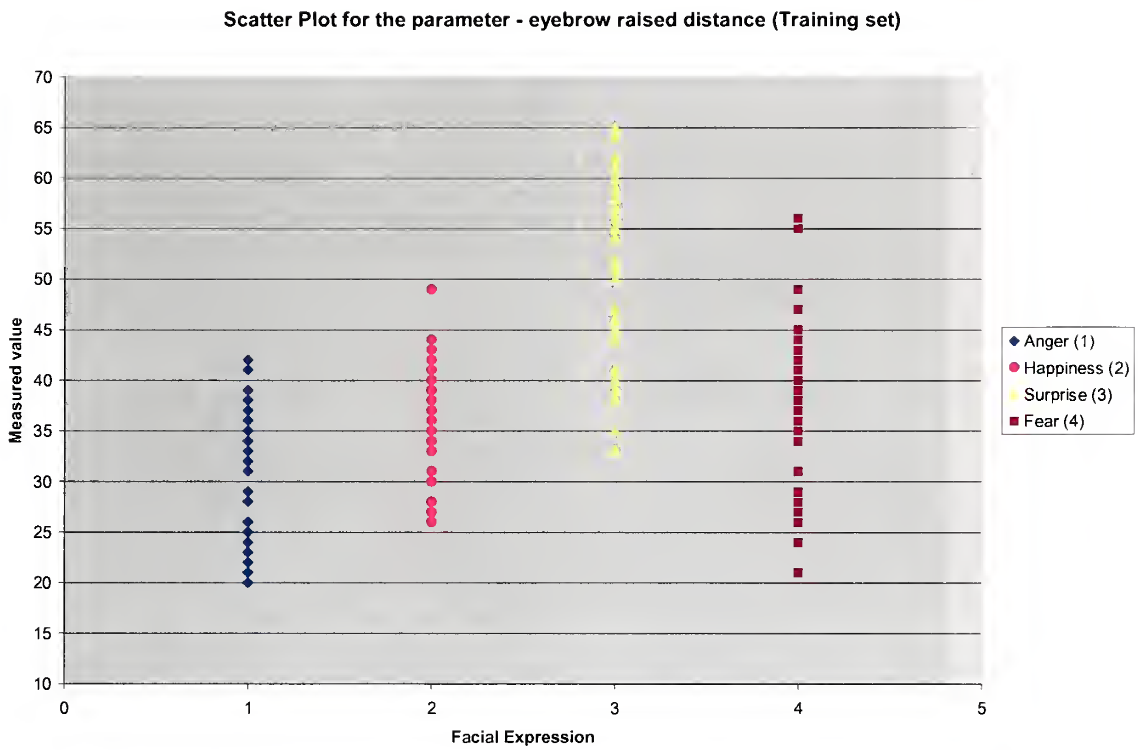

Figure 4-3 Scatter plot, by facial expression, for the real-valued parameter, eyebrow raised distance calculated from the training dataset.

The scatter plot of the parameter *upper eyelid - eyebrow distance* in different expressions is shown in Figure 4-4. It was observed that this parameter's values were, on average, the lowest for anger. Also observed, the parameter's values were, on average, the highest in the expression of surprise. Finally, it was observed, that the parameter's values for the expressions of anger, happiness and fear, were similar.

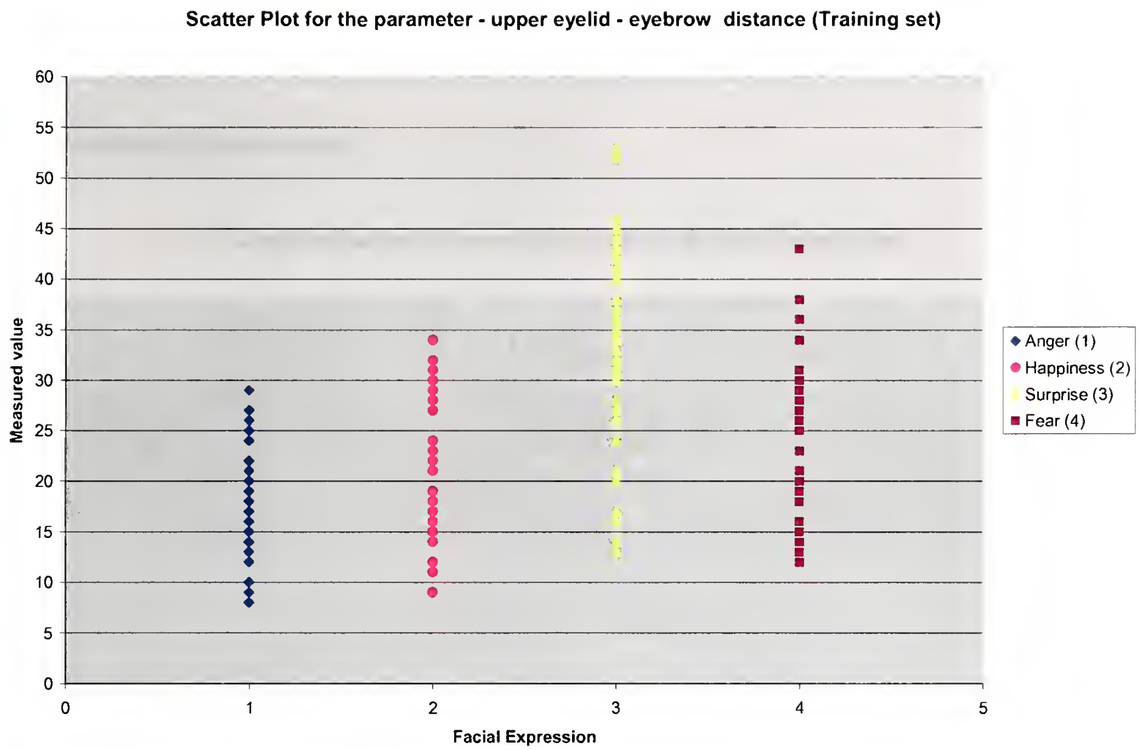

Figure 4-4 Scatter plot, by facial expression, for the real-valued parameter, upper eyelid - eyebrow distance calculated from the training dataset.

The scatter plot of the parameter *inner eyebrow distance* in different expressions, is shown in Figure 4-5. It was observed that this parameter's values were widely scattered for all the facial expressions.

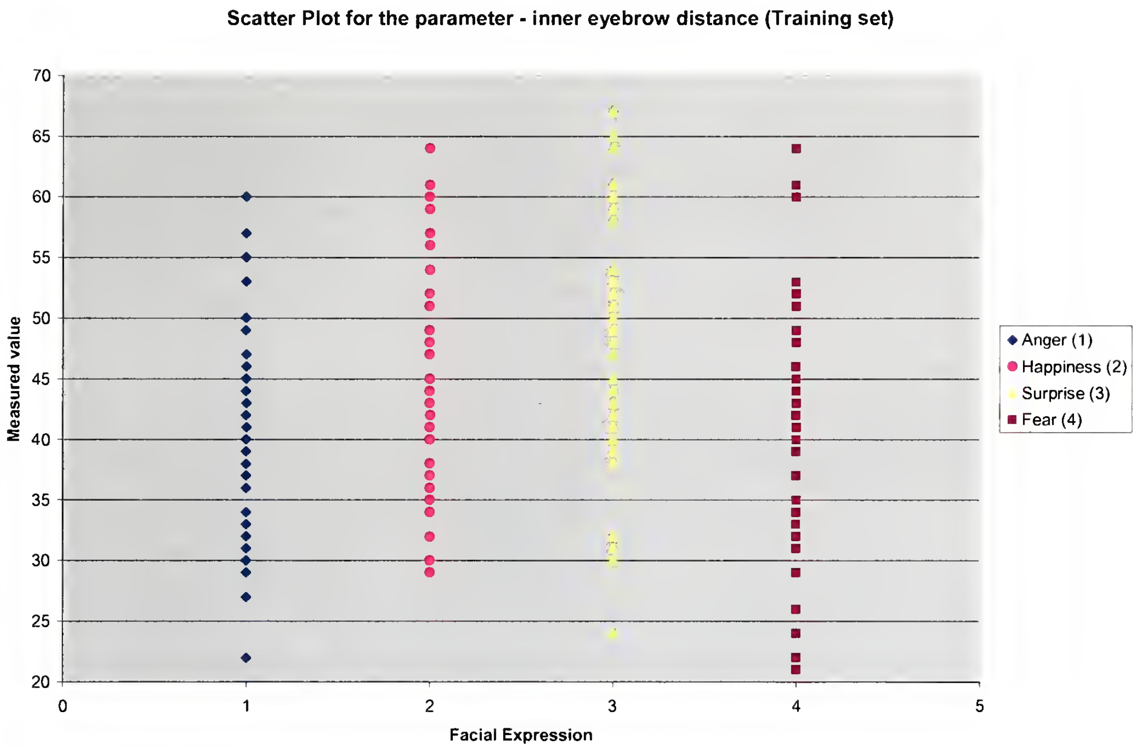

Figure 4-5 Scatter plot, by facial expression, for the real-valued parameter, inner eyebrow distance calculated from the training dataset.

The scatter plot of the parameter *upper eyelid – lower eyelid distance* in different expressions, is shown in Figure 4-6. It was observed that this parameter's values were widely scattered for all the facial expressions.

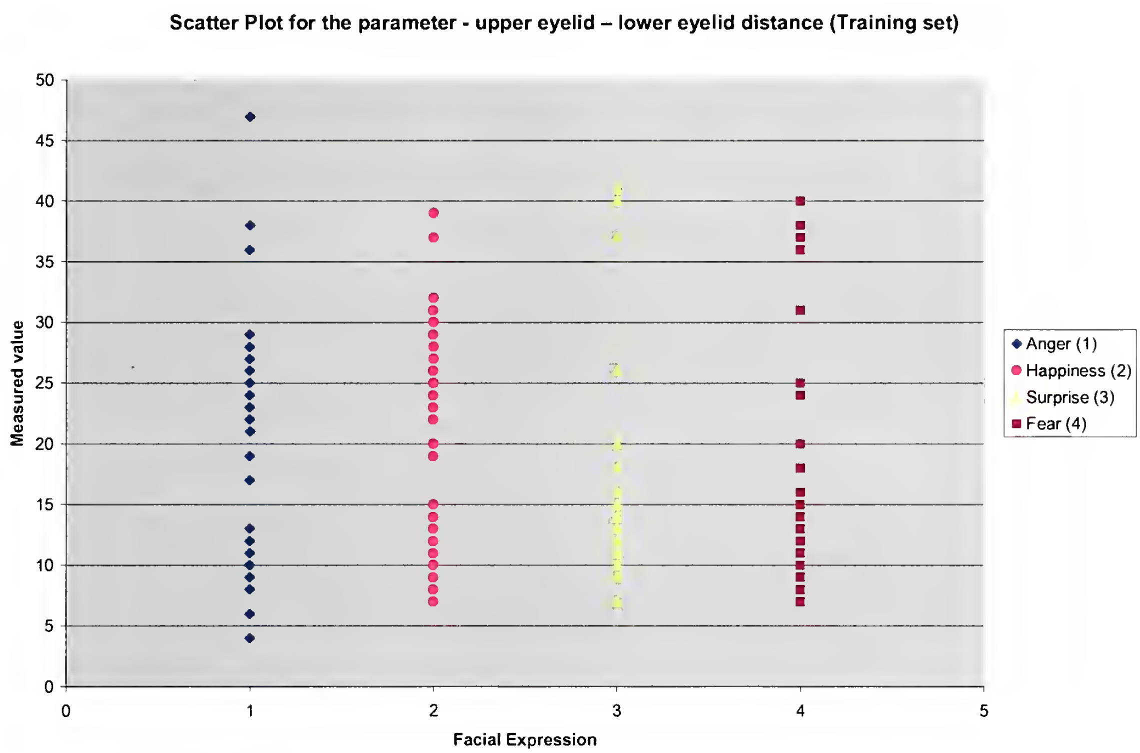

Figure 4-6 Scatter plot, by facial expression, for the real-valued parameter, upper eyelid – lower eyelid distance calculated from the training dataset.

The scatter plot of the parameter *top lip thickness* in different expressions, is shown in Figure 4-7. It was observed that this parameter's values were closely scattered for all the facial expressions, and, at about the same range.

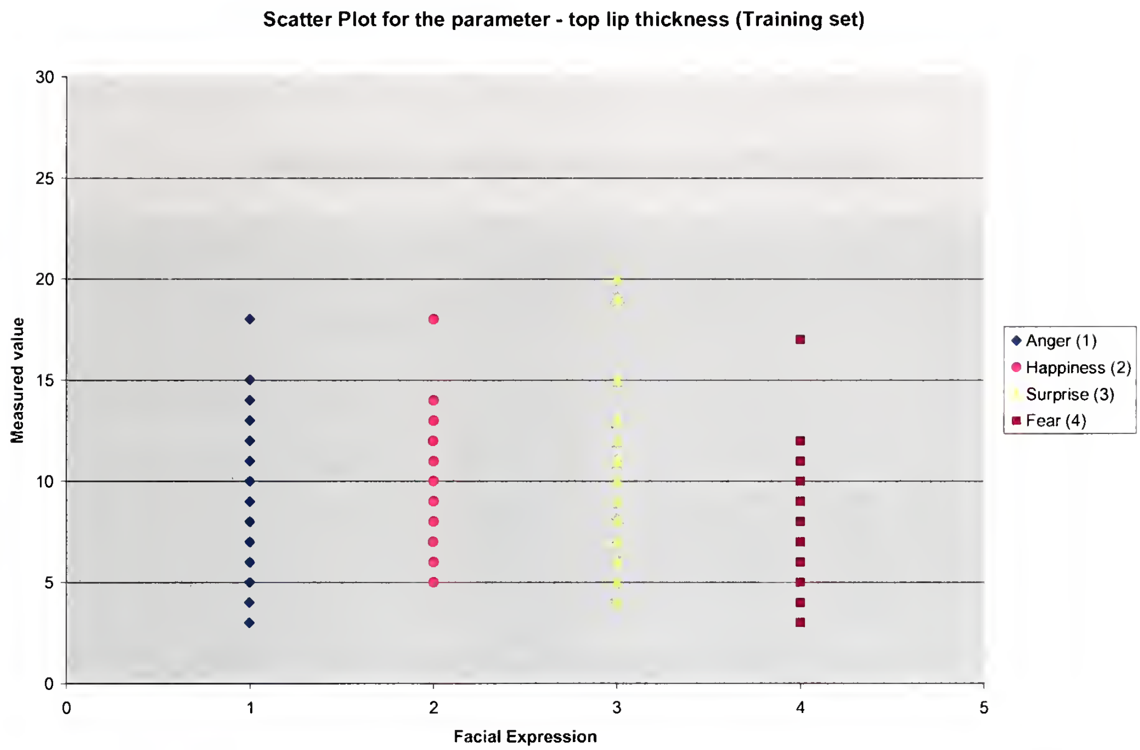

Figure 4-7 Scatter plot, by facial expression, for the real-valued parameter, top lip thickness calculated from the training dataset.

The scatter plot of the parameter *lower lip thickness* in different expressions is shown in Figure 4-8. It was observed, that this parameter's values were, on average, the lowest for the expression of anger. Also observed, this parameter's values were, on average, the highest, in the expression of surprise. Finally, it was observed that the parameter's values for the expression of fear were widely scattered.

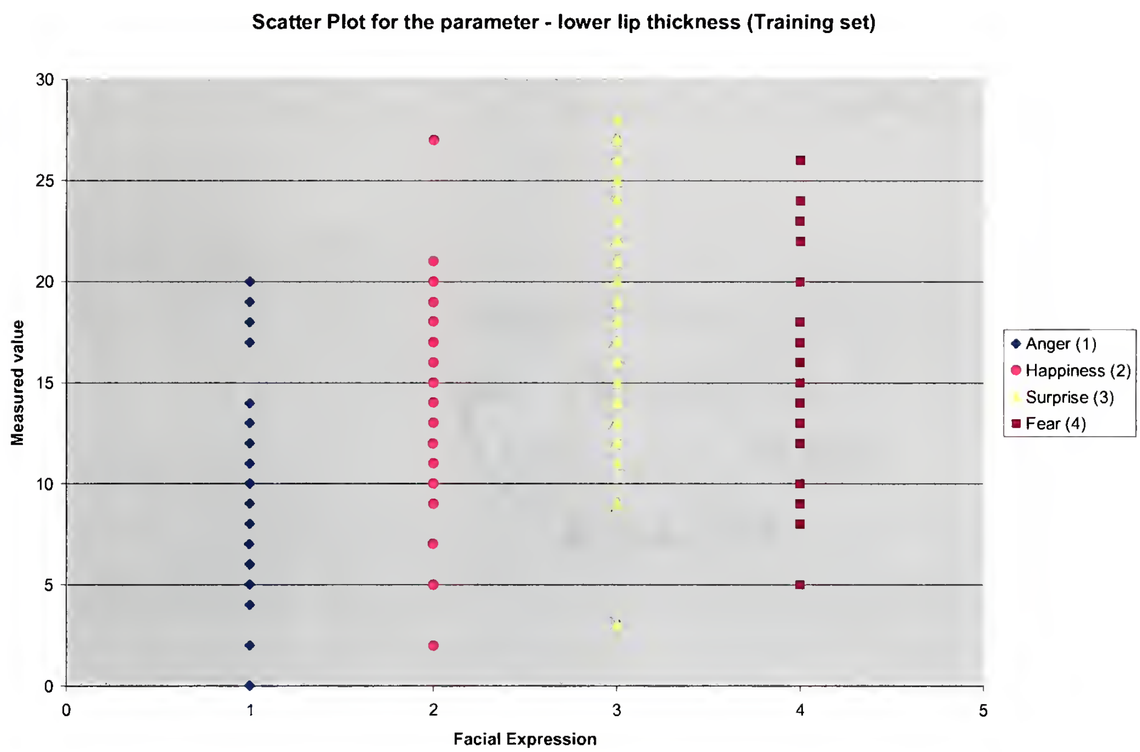

Figure 4-8 Scatter plot, by facial expression, for the real-valued parameter, lower lip thickness calculated from the training dataset.

The scatter plot of the parameter *mouth width* in different expressions, is shown in Figure 4-9. It was observed that this parameter's values were, on average, the lowest for the expressions of anger and surprise. Also observed, this parameter's values were, on average, the highest in the expression of happiness. Finally, it was observed that the parameter's values for the expression of fear and surprise were widely scattered.

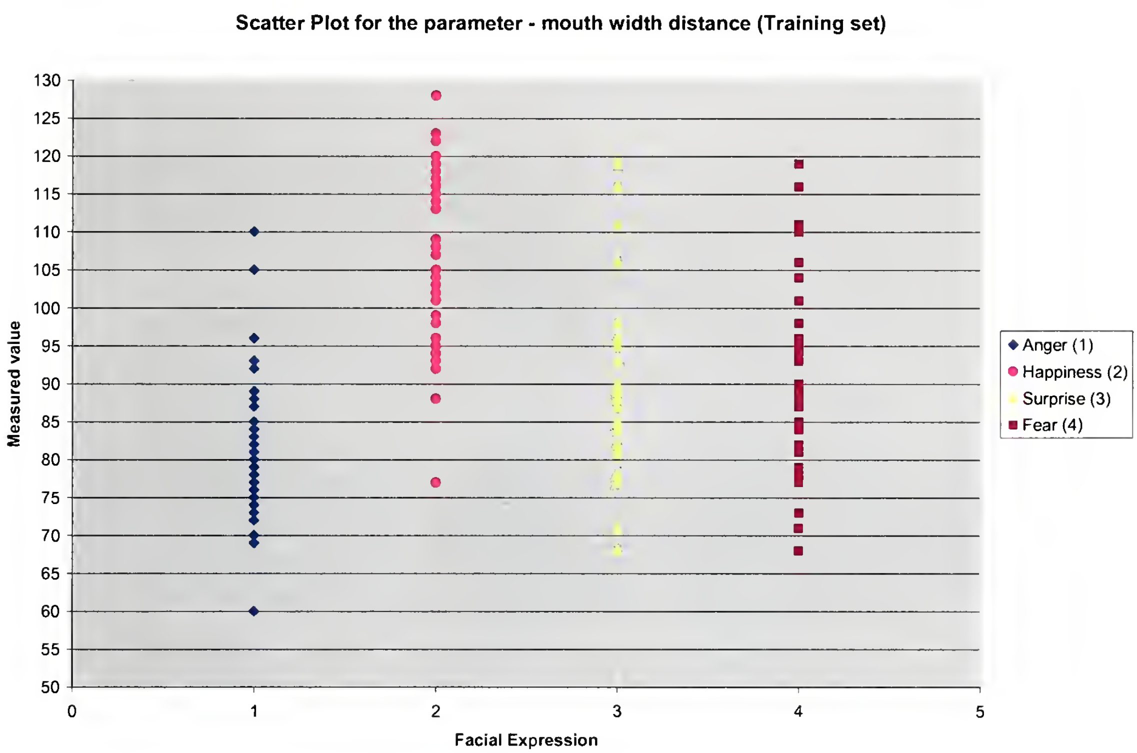

Figure 4-9 Scatter plot, by facial expression, for the real-valued parameter, mouth width calculated from the training dataset.

The scatter plot of the parameter *mouth opening distance* in different expressions, is shown in Figure 4-10. It was observed that this parameter's values were, on average, the lowest for the expression of fear. Also observed, this parameter's values were, on average, the highest in the expression of surprise. Finally, it was observed that the parameter's values for the expression of anger and surprise were widely scattered.

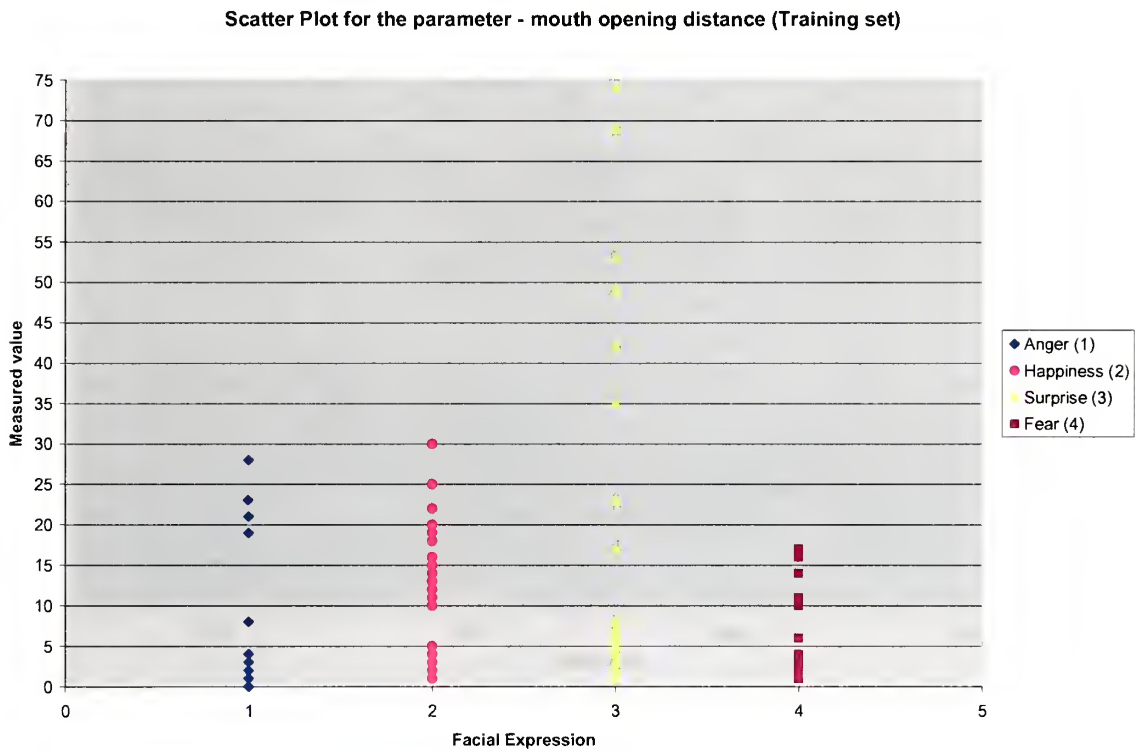

Figure 4-10 Scatter plot, by facial expression, for the real-valued parameter, mouth opening distance calculated from the training dataset.

The average trend line of the parameter *upper teeth visible* in different expressions is shown in Figure 4-11. It was observed that the parameter *upper teeth visible* were, on average, the lowest for the expression of anger, and were, on average, the highest for the expressions of fear and happiness.

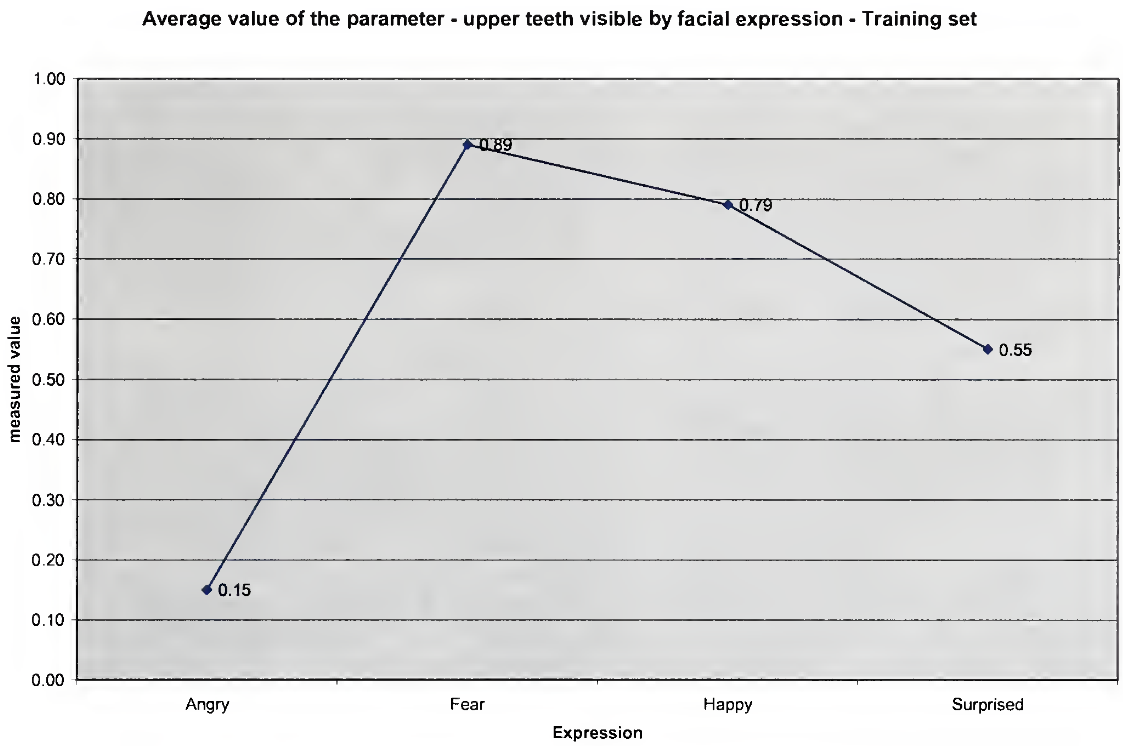

Figure 4-11 Plot of the average value, by facial expression, for the binary parameter, upper teeth visible calculated from the training dataset.

The average trend line of the parameter *lower teeth visible* in different expressions is shown in Figure 4-12. It was observed that the parameter *lower teeth visible* were, on average, the lowest for the expressions of anger and happiness, and were, on average, the highest for the expression of fear.

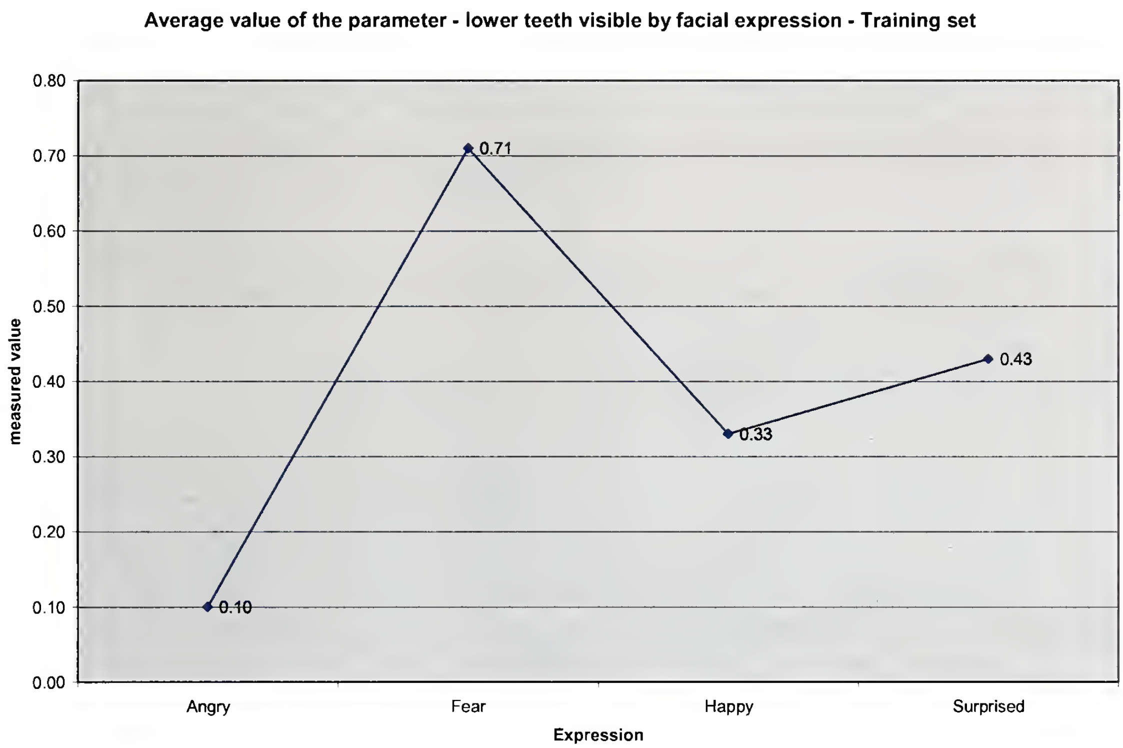

Figure 4-12 Plot of the average value, by facial expression, for the binary parameter, lower teeth visible calculated from the training dataset.

The average trend line of the parameter *forehead lines* in different expressions, is shown in Figure 4-13. It was observed that the parameter *forehead lines* were, on average, the lowest for the expressions of anger, fear, and happiness, and were, on average, the highest for the expression of surprise.

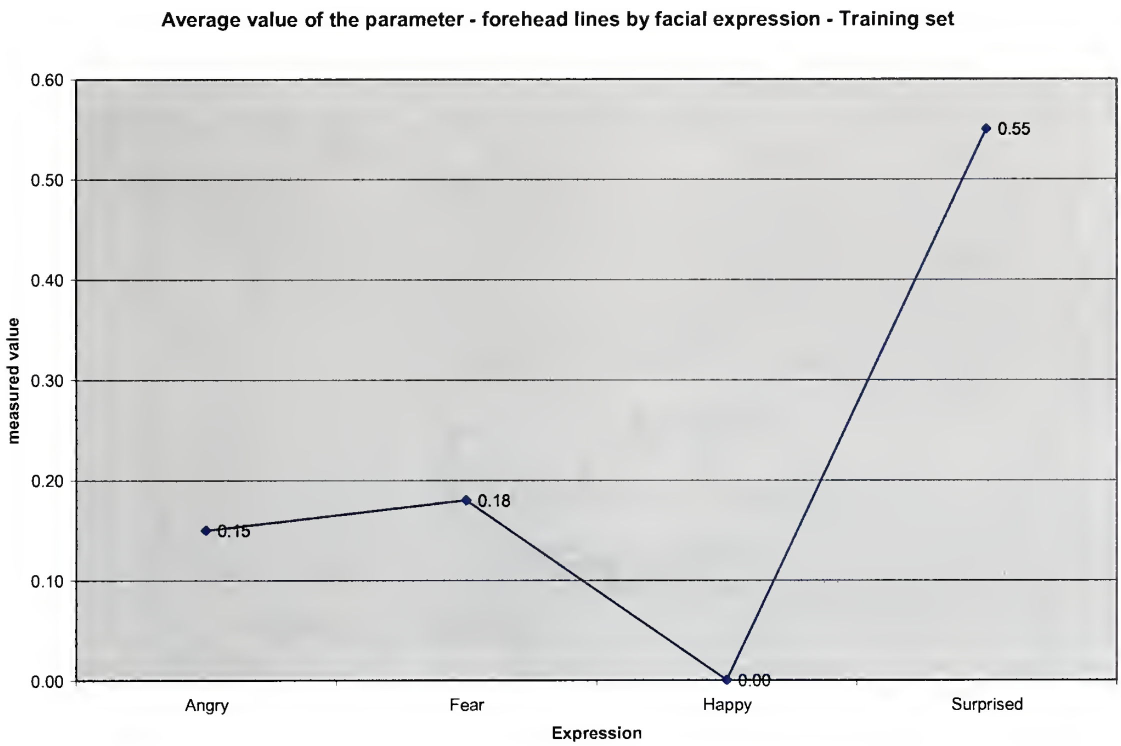

Figure 4-13 Plot of the average value, by facial expression, for the binary parameter, forehead lines calculated from the training dataset.

The average trend line of the parameter *inner eyebrow distance* in different expressions is shown in Figure 4-14. It was observed that the parameter *inner eyebrow distance* was, on average, the lowest for in the expressions of fear, happiness, and surprise and were on average the highest for the expression of anger.

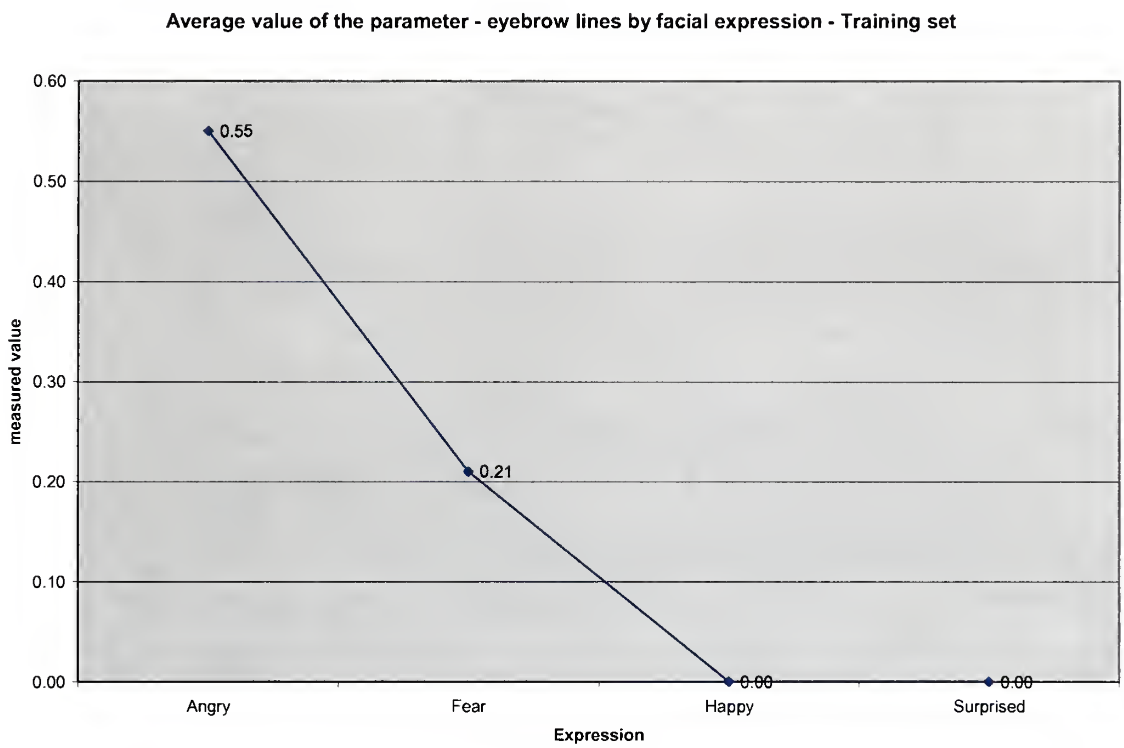

Figure 4-14 Plot of the average value, by facial expression, for the binary parameter, eyebrow lines calculated from the training dataset.

The average trend line of the parameter *nose lines* in different expressions is shown in Figure 4-15. It was observed that the parameter *nose lines* were, on average, the lowest for the expressions of happiness and surprise, and were, on average, the highest for the expression of anger.

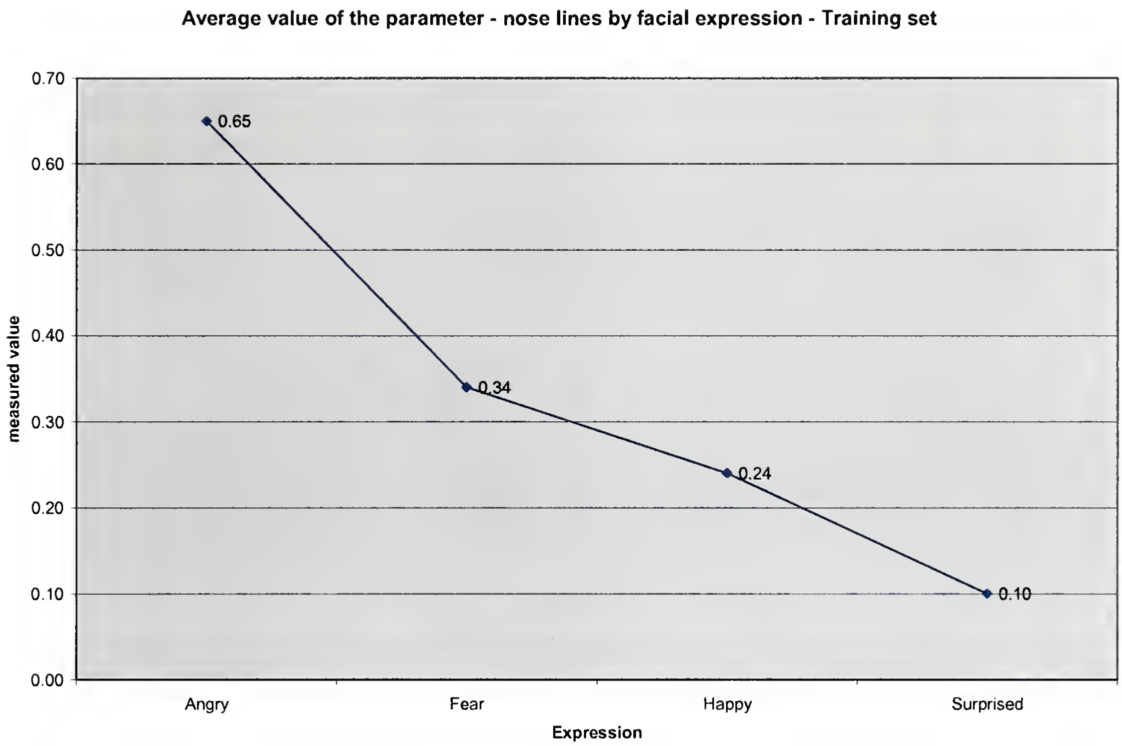

Figure 4-15 Plot of the average value, by facial expression, for the binary parameter, nose lines calculated from the training dataset.

The average trend line of the parameter *chin lines* in different expressions is shown in Figure 4-16. It was observed that the parameter *chin lines* were, on average, the lowest for the expressions of fear and surprise, and were, on average, the highest for the expression of anger.

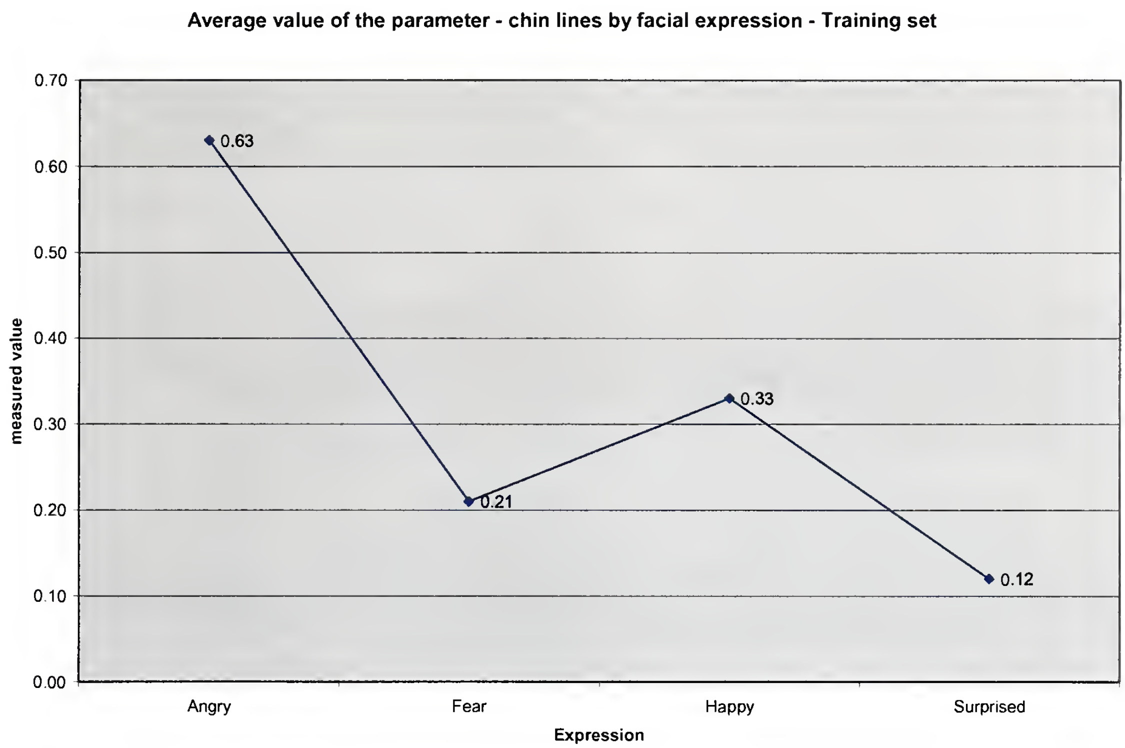

Figure 4-16 Plot of the average value, by facial expression, for the binary parameter, chin lines calculated from the training dataset.

The average trend line of the parameter *nasolabial lines* in different expressions is shown in Figure 4-17. It was observed that the parameter *nasolabial lines* were, on average, the lowest for the expression of surprise, and were, on average, the highest for the expressions of anger, fear and happiness.

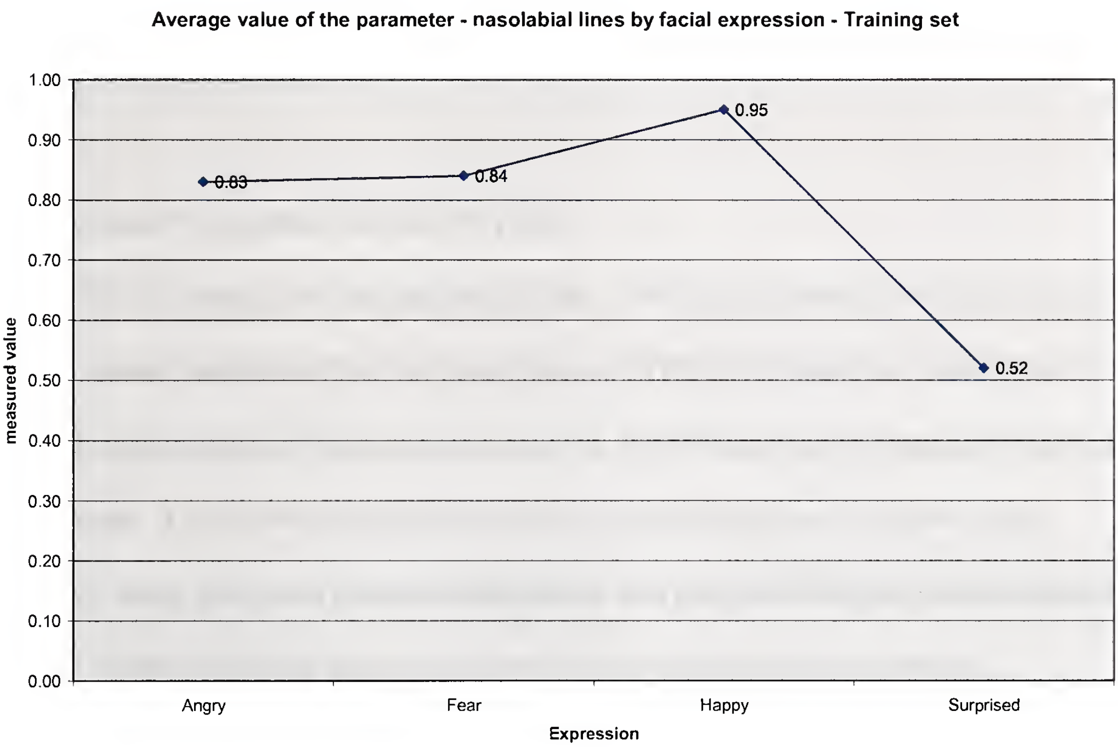

Figure 4-17 Plot of the average value, by facial expression, for the binary parameter, nasolabial lines calculated from the training dataset.

4.4 Facial Image Database

The Cohn-Kanade database (Kanade, Cohn, & Tian, 2000) contained facial images taken from 97 subjects aged from 18 to 30 years. The database had sixty-five percent female subjects. Fifteen percent of the subjects were African-American and three percent were Asian or Latino. The database images were taken using a Panasonic WV3230 camera. The camera was located directly in front of the subject. The subjects performed different facial displays (single action units and combinations of action units) starting and ending with a neutral face. The displays were based on descriptions of prototypic emotions (i.e., joy, surprise, anger, fear, disgust, and sadness). The final frame of each sequence was coded based on FACS by a certified FACS coder. The image sequences were digitized into 640 by 480 pixel arrays, with 8-bit precision for grayscale values. The image format used in this study was png. The tool described below in section 4.5 was used to visualize the images in png. format.

4.5 Parameter Extraction Tool and Method

The UTHSCSA *ImageTool* (ImageTool website, 1996-2002) software was used to extract the real-valued parameters from the facial image. UTHSCSA *ImageTool*, developed in C++, can acquire, display, edit, analyze, process, compress, save, and print gray scale and color images. It includes image analysis functions like dimensional (distance, angle, perimeter, area), gray scale measurements (point, line and area histogram with statistics), standard image processing functions such as contrast manipulation, sharpening, smoothing, edge detection, median filtering and spatial convolutions with user-defined convolution masks. The UTHSCSA *ImageTool* based contrast adjustment and edge detection were used to identify the presence or absence of the binary parameters on the facial image in different expressions. The real-valued parameters were the distances measured between specified facial features by the number of pixels. The binary parameters were characterized by the presence or absence of the facial muscle contractions or facial patterns formed due to these contractions.

4.6 Artificial Neural Network Tool

MATLAB software (The MathWorks website, 1994-2010) was used to create, train, and simulate the three artificial neural networks. MATLAB software provides a high-level programming language that performs various numerical operations. It also provides an interactive technical computing environment and the ability to develop algorithm, analysis and visualization of data, and numeric computation. MATLAB offers add-on toolboxes that extend the MATLAB environment to solve particular classes of problems in these application areas. The Neural Network Toolbox was used for this experiment. The Neural Network Toolbox provides tools for designing, implementing, visualizing, and simulating artificial neural networks. The Neural Network Toolbox provides support for most artificial neural network models, as well as GUIs that enable the designing and managing the networks.

4.7 Setup and Training of the Neural Networks

4.7.1 The Training and Testing Datasets

The training dataset consisted of 162 facial images of 40 subjects displaying the four emotions – anger, happiness, surprise and fear. The testing dataset consisted of 80 facial images of 25 subjects displaying the four emotions – anger, happiness, surprise and fear. The distributions of different types of these emotions in the training and testing dataset are shown in tables 4-3 and 4-4, displayed below.

Table 4-3 Distribution of the training dataset

Facial Expression Demonstrated	Number of facial images
Anger	40
Fear	42
Happiness	38
Surprise	42
Total	162

Table 4-4 Distribution of the testing dataset

Facial Expression Demonstrated	Number of facial images
Anger	19
Fear	21
Happiness	18
Surprise	22
Total	80

4.7.2 Setup and Training of the Hopfield Artificial Neural Network

A Hopfield neural network was designed to classify the different expressions using the fifteen parameters. The network was created using the MATLAB Neural Network Toolbox. Shown below is the Hopfield network architecture (Demuth, Beale, & Hagan, 1992-2009).

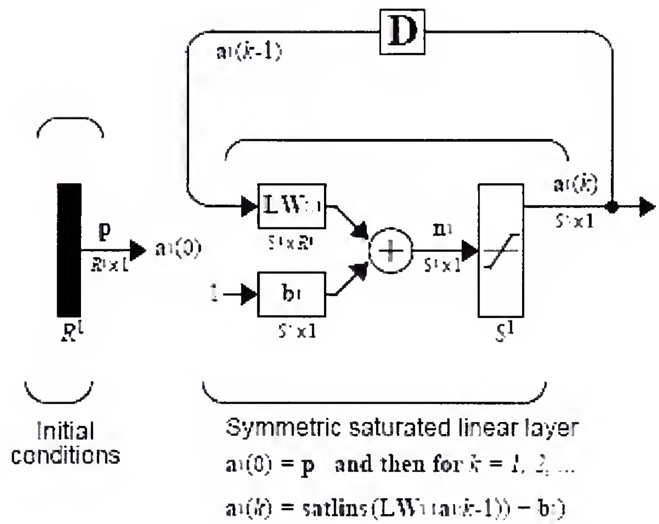

Figure 4-18 Hopfield network architecture used by MATLAB

Input vectors consisted of fifteen elements, and each element represented one of the fifteen dataset parameters. The vectors were fed to the network representing the targets. There were four target vectors, each one representing one of the different facial expressions anger, fear, happiness, or surprise. The parameters values were converted to bipolar values (+1 or -1) with a +1 representing a value above the parameters' average, taken from the training set, and a -1 representing a value below the parameters' average.

Steps followed to convert the real-valued parameter values in the training and testing data sets into bipolar values, are outlined below:

1. Compute the average for each parameter in the training dataset (Avg1).
2. Compute the average for each parameter, for each facial expression target (Avg2).
3. Calculate bipolar value for each parameter, for each facial expression: For each facial expression, if Avg2 > Avg1 then assign +1 to bipolar value, else assign -1 to bipolar value.

4. For binary parameters, 0s and 1s were converted into -1s and +1s, respectively, for a bipolar representation.

Table 4-5 Training dataset for the Hopfield network, both real-valued and binary parameters represented as bipolar values.

Facial Expression Demonstrated	Real-Valued Parameters									Binary Parameters					
Anger	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	-1	1	1	1	1
Happiness	-1	-1	1	1	1	-1	1	1	1	-1	-1	-1	-1	1	1
Fear	1	1	1	1	1	1	-1	1	-1	1	1	-1	-1	-1	-1
Surprise	-1	1	-1	-1	-1	1	1	-1	1	1	-1	1	1	-1	1

The Hopfield neural network was then created using MATLAB. The number of neurons was equal to the number of elements in the input vector, fifteen. The network was trained with the training set, and finally simulated using the testing set. The transfer function used in MATLAB was *satlins*. *Satlins* is the saturated linear transfer function. For inputs less than -1 *satlins* produces -1. For inputs in the range -1 to +1, it simply returns the input value. For inputs greater than 1, it produces +1. When the network was simulated, the network would attempt to converge to one of the target vectors for each input vector from the test set. When attempts to converge to one of the target vectors failed, the result would be ambiguous.

4.7.3 Setup and Training of the LVQ Artificial Neural Network

An LVQ network was created to classify the different expressions using the fifteen parameters. The LVQ neural network was created using the Neural Network Toolbox of MATLAB. Shown below is the LVQ network architecture (Demuth, Beale, & Hagan, 1992-2009).

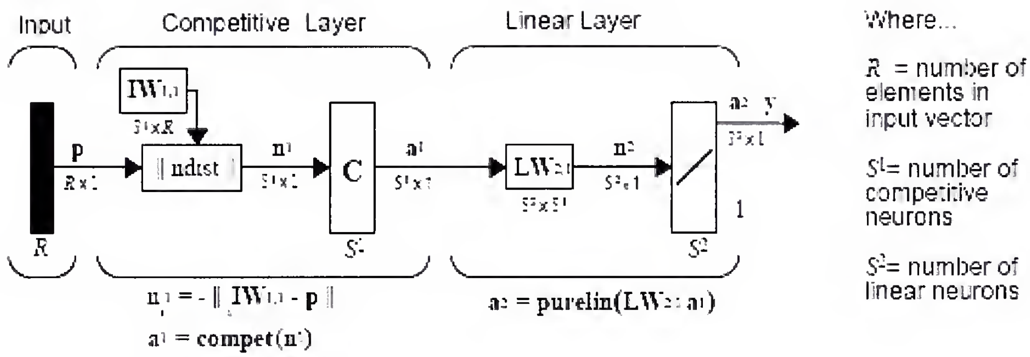

Figure 4-19 LVQ network architecture used by MATLAB

The linear layer was set up with four neurons representing the four target facial expressions – anger, fear, happiness and surprise. The competitive layer was set up with eight neurons representing subclasses connected to the four neurons in the linear layer. Each neuron in the competitive layer was connected to one of the four neurons in the linear layer. The training datasets were normalized using the minimum and maximum values and fed into the network. The network was trained using different numbers of epochs from 50-500 and with different weight distributions assigned to the four targets in the linear layer. The LVQ neural network was trained with the training data set, and simulated using the testing data set. When the network was simulated, the network would converge to one of the target neurons for each input vector from the test set. The winning neuron, in the linear layer, was indicated with a value of one.

4.7.4 Setup and Training of the Feedforward Artificial Neural Network

A feedforward network was created to classify the different expressions using the fifteen parameters. The feedforward neural network was created using the Neural

Network Toolbox of MATLAB. Shown below is the feedforward network architecture (Demuth, Beale, & Hagan, 1992-2009).

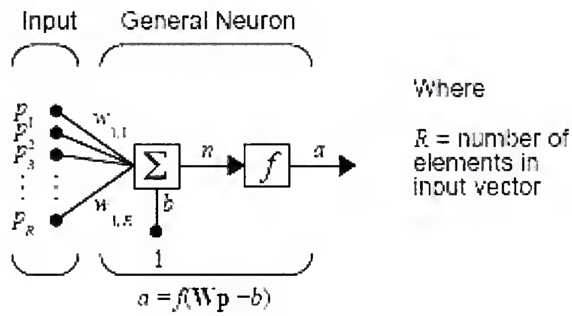

Figure 4-20 Feedforward network architecture used by MATLAB

Several different configurations of the networks (described below) were trained to classify different expressions to find the best performing feedforward network. Each dataset corresponded to a target representing one of the four target facial expressions – anger, fear, happiness or surprise. Networks were trained using different numbers of hidden layers (1, 2, and 3), different initial weights, different numbers of neurons in the hidden layers (6, 8, 10, 12, 14, 15, 20, 21, 24, 28, 30), different learning rates, and different momentum constants. Networks were trained with different transfer functions (*tansig, purelin*). As explained previously in Chapter 3, transfer functions calculate a layer's output from its net input. *Tansig* is a hyperbolic tangent sigmoid transfer function. It takes one input and returns each element of the input squashed between -1 and 1. *Purelin* is a linear transfer function. It takes one input and returns each element of the input squashed between -1 and 1. Each network had fifteen input nodes, each corresponding to one of the fifteen input parameters. Each of these networks had four output nodes, each corresponding to one of the four expressions (*anger, fear,* *happiness, or surprise*). Each output node was represented by a target value. Associated with each target value was an acceptable range. The acceptable range was values above and below a target value and closest to the target value. Since the normalized input data was in the range of -1 to 1, *tansig* function was used for the hidden layer neurons. Both the *tansig* and *purelin* functions were tried as the transfer function for the output layer neurons. The output of each node was compared to each target values' accepted range to determine the facial expression:

Table 4-7 Feedforward facial expression targets and accepted range

Facial Expression	Output Target	Accepted Range
Anger	-1	-1.33 : -.67
Happiness	-0.33	-.66 : 0
Surprise	0.33	0:.66
Fear	1	.67 : 1.33

The networks were trained using the Gradient Descent (*traingdm*) technique using MATLAB and the maximum number of epochs used for training was varied from 500-5000.

4.7.4.1 Data Used for Training and Testing

The training data set was divided into three subsets. The first subset was the training set, which was used for computing the gradient and updating the network weights. The second subset was the validation set. The error on the validation set was monitored during the training process. The validation error normally decreases during the initial phase of training, as does the training set error. However, when the network begins to over train, leading to a loss of its generalization capability, the error on the validation set *typically* begins to rise. When the validation error was increased for a specified number of iterations, the training was stopped, and the weights at the minimum of the validation error were returned. The third subset was the test data set, which was not used during training, but it was used to compare different models. It was also useful to plot the test set error during the training process. If the error in the test set reaches a minimum at a significantly different iteration number than the validation set error, this might indicate a poor division of the data set. (Demuth et al., 1992-2009, p. 208) The partitioning of the available data set into training, validation and testing subsets is given below:

- Training subset = 60%
- Validation subset = 20%
- Test subset = 20%

4.8 Principal Component Analysis

Principal Components Analysis (PCA) is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analyzing data. The other main advantage of PCA is once the patterns in the data are found, and the data is compressed, i.e. by reducing the number of dimensions, then much of information is not lost. (Smith, 2002, Chapter 3)

4.8.1 Principal Component Analysis Method

Listed below is the method used to perform a principal component analysis (Smith, 2002, Chapter 3).

1. **Get data -** The training dataset was used, which consisted of 162 facial images of 40 subjects displaying the four emotions – anger, happiness, surprise and fear.
2. **Subtract the mean -** For PCA to work properly, the mean must be subtracted from each of the data dimensions. The mean subtracted is the average across each dimension. This produces a data set whose mean is zero.
3. **Calculate the covariance matrix-** Covariance is the measurement to find out how much the dimensions vary from the mean with respect to each other. A covariance matrix is all the possible covariance values between all the different dimensions.
4. **Determine the number of components needed -** The relative significance of each component is indicated by its *eigenvalue*. An eigenvalue represents the amount of variance that is accounted for by a given component. The first principal component will have the largest eigenvalue, and succeeding components will have smaller eigenvalues, as their significance in the data decreases. The eigenvalue-one function was used to determine the number of components to keep. This approach accepts any component with an eigenvalue greater than 1.00.

Chapter 5 Results and Analysis

Two hundred and forty-three facial images were used for training and testing the artificial neural networks. The images were from sixty-five subjects. Eight real-valued and seven binary parameters were extracted from the images.

Data used to train the artificial neural networks came from 40 subjects consisting of 162 images. The artificial neural networks were trained to classify images into one of the following four expressions: anger, fear, surprise, or happiness.

The artificial neural networks were tested with data from 80 images of 25 subjects. The testing data was different from the data used in the training. Comparison was made of the performances of the three artificial neural network models. A confusion matrix (see section 5.1) was used to evaluate the performance of each network. The best performing network at identifying facial expressions was the Hopfield network.

5.1 Confusion Matrix

A confusion matrix (Hamilton, 2007) is a matrix of actual and predicted classifications done by a classification system, in this case, the neural networks. The entries in the confusion matrix have the following meaning in the context of two types of possible predictions – positive and negative.

- *a* is the number of **correct** predictions that an instance is **negative**,
- *b* is the number of **incorrect** predictions that an instance is **positive**,
- *c* is the number of **incorrect** of predictions that an instance **negative**, and
- *d* is the number of **correct** predictions that an instance is **positive**.

		Predicted	
		Negative	Positive
Actual	Negative	**A**	**b**
	Positive	**C**	**d**

Figure 5-1 Confusion matrix (Hamilton, 2007)

Here are the terms that can be calculated from the confusion matrix:

- The *accuracy* (*AC*) is the proportion of the total number of predictions that were correct. It is determined using the equation:

$$AC = \frac{a+d}{a+b+c+d} \qquad \text{Eqn. 1}$$

- The *recall* or *true positive rate* (*TP*) is the proportion of positive cases that were correctly identified, as calculated using the equation:

$$TP = \frac{d}{c+d} \qquad \text{Eqn. 2}$$

- The *false positive rate* (*FP*) is the proportion of negative cases that were incorrectly classified as positive, as calculated using the equation:

$$FP = \frac{b}{a+b} \qquad \text{Eqn. 3}$$

- The *true negative rate* (*TN*) is defined as the proportion of negative cases that were classified correctly, as calculated using the equation:

$$TN = \frac{a}{a+b} \qquad \text{Eqn. 4}$$

- The *false negative rate* (*FN*) is the proportion of positive cases that were incorrectly classified as negative, as calculated using the equation:

$$FN = \frac{c}{c+d} \qquad \text{Eqn. 5}$$

- Finally, *precision* (*P*) is the proportion of the predicted positive cases that were correct, as calculated using the equation:

$$P = \frac{d}{b+d} \qquad \text{Eqn. 6}$$

The accuracy equation may not be an adequate performance measure when the number of negative cases is much greater than the number of positive cases. Another performance measure that accounts for this, by including the TP in the product, is the *geometric mean* (*g-mean*), as defined below,

$$g - mean_1 = \sqrt{TP * P} \qquad \text{Eqn. 7}$$

$$g - mean_2 = \sqrt{TP * TN} \qquad \text{Eqn. 8}$$

This study used equation 7 to calculate the geometric mean.

In this particular study, there were four possible correct and incorrect predictions because of the four types of facial expressions.

5.2 Performance of the Hopfield Network

The Hopfield network performance was best for the expression of surprise, followed by happiness, anger and fear, with accuracy ranging from 50% - 81.82%. The Hopfield network had one ambiguous result. An ambiguous result happens when the Hamming distance between the input vector and two or more target vectors are equal. The confusion matrix in Table 5-1 summarizes the performance of the Hopfield network and Table 5-2 shows the geometric mean of that performance.

Table 5-1 Confusion matrix showing the performance of the Hopfield network

Emotion Presented	Anger	Happiness	Surprise	Fear	Angry/ Surprise	Total	Accuracy
Anger	14	4			1	19	73.68%
Happiness		17	1	3		21	80.95%
Surprise		2	18	2		22	81.82%
Fear	2	1	6	9		18	50.00%
						Overall	72.50%

Table 5-2 The calculated geometric mean of the Hopfield network performance

Emotion Presented	Geometric Mean
Anger	80.30%
Happiness	75.72%
Surprise	76.75%
Fear	56.69%
Overall	72.37%

5.3 Performance of the LVQ Network

The LVQ network performance was best for the expression of surprise, followed by happiness, anger and fear, with accuracy ranging from 38.89 - 100%. Training was at 350 epochs. The weight distribution was .2 for anger, happiness, surprise, and .4 for fear. The confusion matrix in Table 5-3 summarizes the performance of the LVQ network and Table 5-4 shows the geometric mean of that performance.

Table 5-3 Confusion matrix showing the performance of the LVQ network

Weight Distribution	**0.2**	**0.2**	**0.2**	**0.4**		
Emotion Presented	**Angry**	**Happy**	**Surprise**	**Fear**	**Total**	**Accuracy**
Anger	10	3	1	5	19	52.63%
Happiness	2	15	0	4	21	71.43%
Surprise	0	0	22		22	100.00%
Fear	8	1	2	7	18	38.89%
					Overall	67.50%

Table 5-4 The calculated geometric mean of the LVQ network performance

Emotion Presented	Geometric Mean
Anger	51.30%
Happiness	75.09%
Surprise	93.81%
Fear	41.25%
Overall	65.36%

5.4 Performance of the Feedforward Network

The Feedforward network was best for the expression of anger, followed by surprise, happiness and fear, with accuracy ranging from 38.89 – 84.21%. The best performing feedforward network was setup with one hidden layer with fifteen neurons. The Transfer function for the hidden and output layers was *tansig*. The learning rate was set to .1 with a momentum of .6. The number of epochs was set to 3000 and the network early stopped at 1337 epochs. The confusion matrix in Table 5-5 summarizes the performance of the Feedforward network and Table 5-6 shows the geometric mean of that performance.

Table 5-5 Confusion matrix showing the performance of the Feedforward network

Emotion Presented	Angry	Happy	Surprise	Fear	Total	Accuracy
Anger	16	2	0	1	19	84.21%
Happiness	4	13	4	0	21	61.90%
Surprise	0	2	14	6	22	63.64%
Fear	2	3	6	7	18	38.89%
					Overall	62.50%

Table 5-6 The calculated geometric mean of the Feedforward network performance

Emotion Presented	Geometric Mean
Anger	80.10%
Happiness	65.08%
Surprise	49.07%
Fear	55.00%
Overall	62.31%

5.5 Comparison of the Network Models' Performance

A comparison plot of the three network models performance based on accuracy is shown in Figure 5-2. Figure 5-3 shows the three network models performance based on the calculated geometric mean. The Hopfield network showed the highest rate of correct identification based on accuracy and based on the geometric mean.

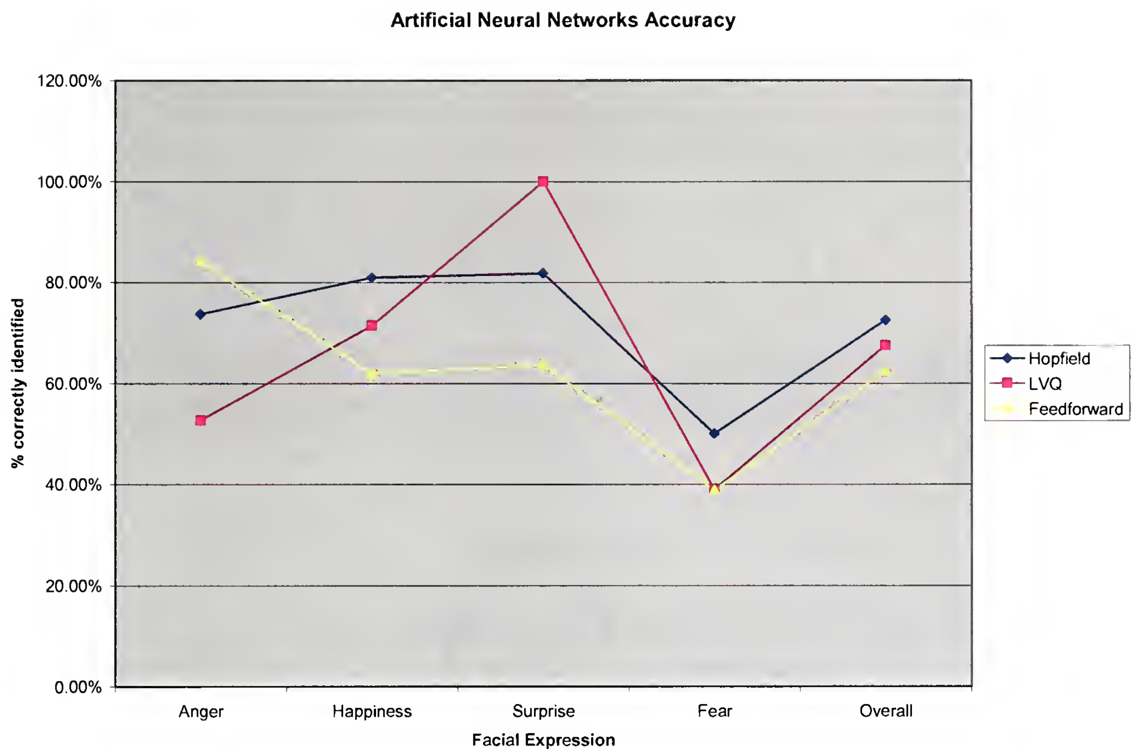

Figure 5-2 artificial neural networks accuracy by facial expression

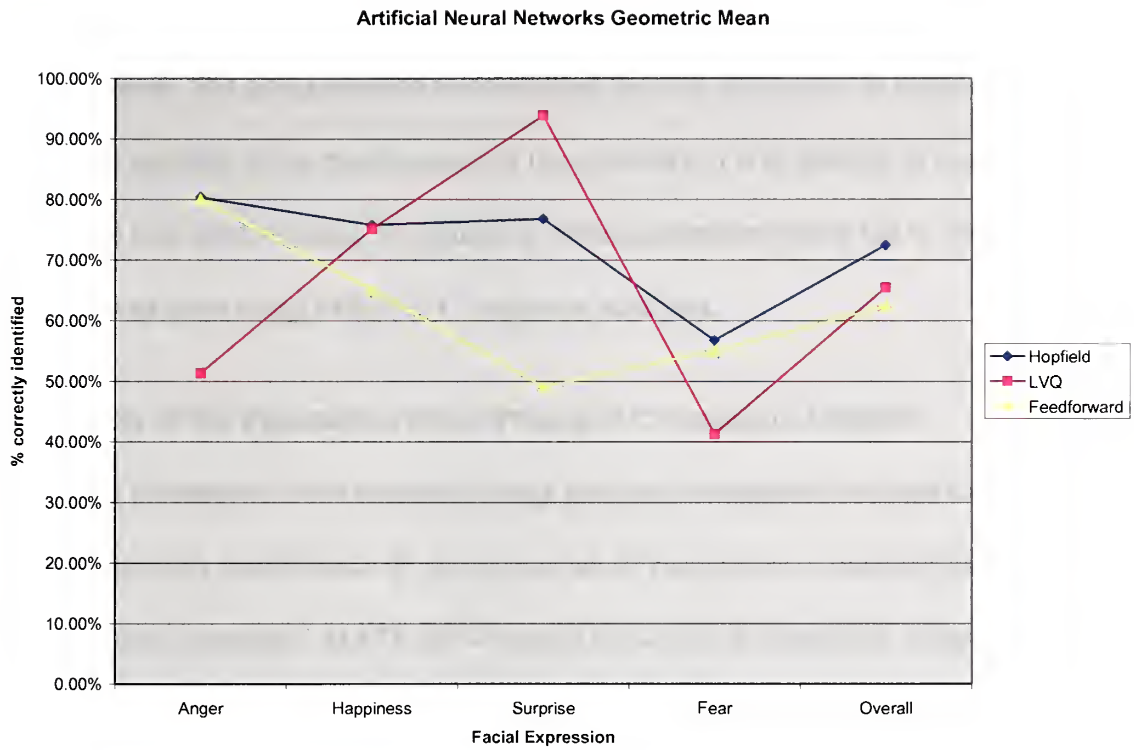

Figure 5-3 artificial neural networks geometric mean by facial expression

The three neural network models produced similar results in identifying facial expressions. Based on accuracy, the best performing network was the Hopfield with a 72.5 % success rate. When considering the geometric mean, the best performing was still the Hopfield network at 72.37 % success rate. The feedforward network was the best at identifying the anger expression with an 84.21% success rate. The Hopfield network was the best at identifying the happiness expression, with an 80.95% success rate. The LVQ was most successful at identifying the surprise expression at 100%. The three network models had a difficult time identifying the fear expression. The Hopfield was the best, at 50 percent, and the LVQ and Feedforward followed with a 38.89 percent success rate.

Interesting, the Hopfield and feedforward networks misclassified the fear expression as anger the most. The LVQ network misclassified the fear expression as surprise the most. During the analysis of the performance of the networks, it was decided to look for ways to improve their performance. Examination of the parameters being fed to the artificial networks was done using Principal Component Analysis.

5.6 Analysis of the Parameters Using Principal Component Analysis

The fifteen parameters were examined using principal component analysis to determine which parameters would make up an optimal set of parameters to train and test the three artificial neural networks. MATLAB was used to execute the principal component analysis. First, the training dataset was standardized by dividing each column (parameter) by its standard deviation. Principal component analysis was done on the standardized data and a vector containing the eigenvalue explained by the corresponding principal component. The eigenvalues were converted to a percent to show the total variability explained by each principal component.

Table 5-7 The eigenvalue for each principal component

Parameter	Eigenvalue
Real-Valued #1	23.9
Real-Valued #2	15.4
Real-Valued #3	11.57
Real-Valued #4	8.56
Real-Valued #5	7.48
Real-Valued #6	6.44
Real-Valued #7	5.96
Real-Valued #8	4.55
Binary #1	3.94
Binary #2	3.13
Binary #3	2.68
Binary #4	2.27
Binary #5	1.87
Binary #6	1.59
Binary #7	0.66

The binary #7 parameter, nasolabial lines, was the only parameter with an eigenvalue of less then one. The other fourteen parameters had an eigenvalue greater then one, and were chosen to be the optimal parameters. The optimal parameters were used to look for possible improvements in performance.

5.7 Performance of the Hopfield Network Using Optimal Parameters

The Hopfield network was best for the expression of surprise, followed by anger, fear and happiness, with accuracy ranging from 47.62% - 86.36%. The Hopfield network had eight ambiguous results. The ambiguous results happened when the Hamming distance between the input vector and two or more target vectors were equal. The confusion matrix in Table 5-8 summarizes the performance of the Hopfield network and Table 5-9 shows the geometric mean of that performance.

Table 5-8 Confusion matrix showing the performance of the Hopfield network using optimal parameters

Emotion Presented	Anger	Happiness	Surprise	Fear	Happiness /Surprise/ Fear	Surprise/ Fear	Happy/ Fear	Total	Accuracy
Anger	14	2		2			1	19	73.68%
Happiness		10	2	5	3	1		21	47.62%
Surprise		1	19	1		1		22	86.36%
Fear	2	1	4	9		**2**		18	50.00%
								Overall	**65.00%**

Table 5-9 The calculated geometric mean of the Hopfield network performance using optimal parameters

Emotion Presented	Geometric Mean
Anger	80.30%
Happiness	58.32%
Surprise	81.02%
Fear	51.45%
Overall	67.77%

5.8 Performance of the LVQ Network Using Optimal Parameters

The LVQ network performance was best for the expression of surprise, followed by happiness, anger and fear, with accuracy ranging from 50 - 100%. Training was at 350 epochs. The weight distribution was .2 for anger, happiness, surprise and .4 for fear. The confusion matrix in Table 5-3 summarizes the performance of the LVQ network and Table 5-4 shows the geometric mean of that performance.

Table 5-10 Confusion matrix showing the performance of the LVQ network

Weight Distribution	0.2	0.2	0.2	0.4		
Emotion Presented	**Angry**	**Happy**	**Surprise**	**Fear**	**Total**	**Accuracy**
Anger	10	2	1	6	19	52.63%
Happiness	2	13	0	6	21	61.90%
Surprise	0	0	22	0	22	100.00%
Fear	6	1	2	9	18	50.00%
					Overall	67.50%

Table 5-11 The calculated geometric mean of the LVQ network performance

Emotion Presented	Geometric Mean
Anger	54.07%
Happiness	70.92%
Surprise	93.81%
Fear	46.29%
Overall	66.27%

5.9 Performance of the Feedforward Network Using Optimal Parameters

The Feedforward network was best for the expression of surprise, followed by happiness, anger and fear, with accuracy ranging from 27.78 – 86.36%. The best performing feedforward network was setup with one hidden layer consisting of fifteen neurons. The transfer function for the hidden and output layers was *tansig*. The learning rate was set to .1 with a momentum of .6. The number of epochs was set to 2500 and the network early stopped at 168 epochs. The confusion matrix in Table 5-12 summarizes the performance of the Feedforward network and Table 5-13 shows the geometric mean of that performance.

Table 5-12 Confusion matrix showing the performance of the Feedforward network

Emotion Presented	Angry	Happy	Surprise	Fear	Total	Accuracy
Anger	11	6	2	0	19	57.89%
Happiness	0	16	5	0	21	76.19%
Surprise	0	1	19	2	22	86.36%
Fear	0	2	11	5	18	27.78%
					Overall	63.75%

Table 5-13 The calculated geometric mean of the Feedforward network performance

Emotion Presented	Geometric Mean
Anger	76.09%
Happiness	69.83%
Surprise	66.59%
Fear	44.54%
Overall	64.26%

5.10 Comparison of the Network Models Performance Using Optimal Parameters

A comparison plot of the three network models performance based on accuracy is shown in Figure 5-4. Figure 5-5 shows the three network models performance based on the calculated geometric mean. The LVQ showed the highest rate of correct identification based on accuracy and based on the geometric mean.

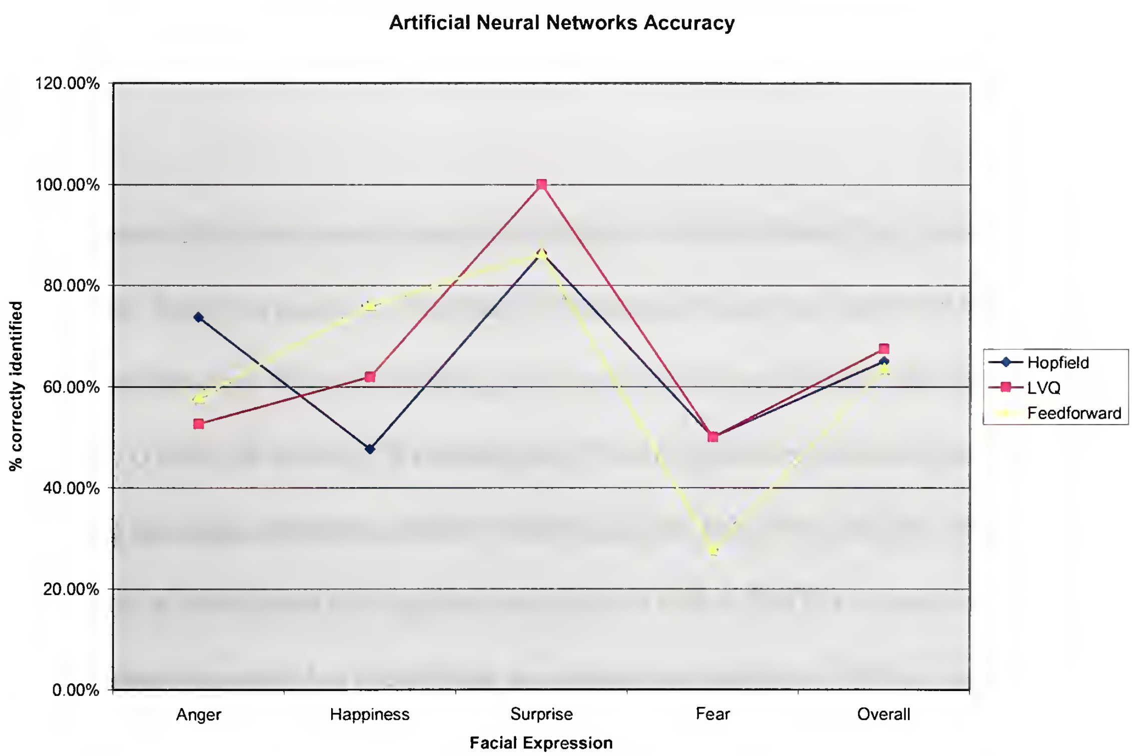

Figure 5-4 artificial neural networks accuracy by facial expression using optimal parameters

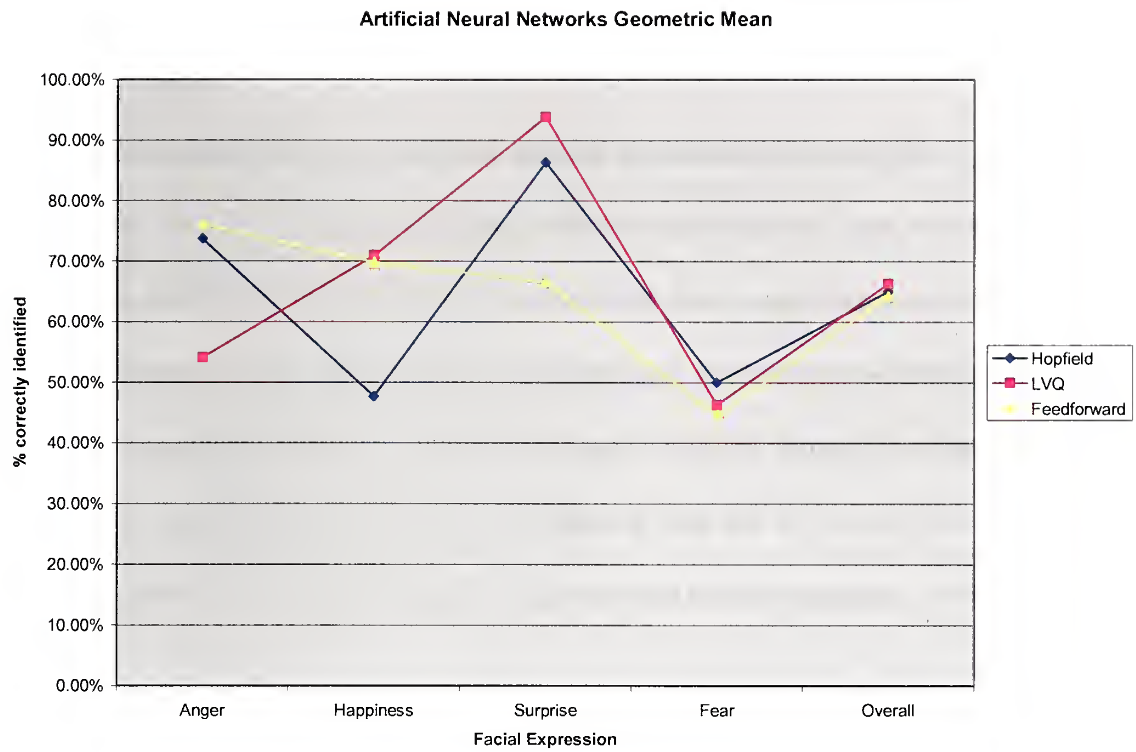

Figure 5-5 artificial neural networks geometric mean by facial expression using optimal parameters

The three neural network models produced similar results in identifying facial expressions. Based on accuracy, the best performing network was the LVQ with a 67.50% success rate. When considering the geometric mean, the best performing was still the LVQ network at 66.27% success rate. The Hopfield network was the best at identifying the anger expression with a 73.68% success rate. The feedforward network was the best at identifying the happiness expression with a 76.19% success rate. The LVQ was most successful at identifying the surprise expression at 100%. The three network models had a difficult time identifying the fear expression. The Hopfield and the LVQ networks were the best at 50 % and the feedforward followed with a 27.78 % success rate. Interestingly, the Hopfield and feedforward networks misclassified the fear expression as surprise the most. The LVQ network misclassified the anger expression as surprise the most.

The Hopfield performed worse with the optimal parameters than with the original parameters. The LVQ network performed with the same accuracy rate, as with fifteen parameters, of 67.50% and the feedforward network saw a slight improvement going from 62.50% to 63.75%. The Hopfield saw a decrease in its ability to identify the happiness expression going from 80.95% correct to 47.62%. Among the four types of expressions, surprise produced the best recognition rates for all the artificial neural network models. The worst recognition accuracy was for the expression of fear.

5.11 Limitations of the Study

A bigger set of facial images for training and testing would improve the accuracy of the three artificial neural networks. The parameters need to be studied more for parameters that will better distinguish the facial expressions. The parameters in the present study were manually extracted from the image. Techniques need to be developed for automated extraction of these parameters. The accuracy could be further improved by first classifying the image into a positive (happiness and surprise) and negative (anger and fear) groupings. The image grouping could then be sub-classified by developing a specialist network for each grouping.

5.12 Significance of the Study

The automatic recognition of human emotions by computers has become a desirable goal. Automation of this process will allow for the development of real-time applications that will effectively identify emotions. The research presented here, has helped better understand the artificial neural approach to automatic recognition of human emotions. This study examined three common artificial neural networks models and their ability to identify human emotions from facial images. The networks did not show much difference in their ability to recognize facial expressions in facial images, though they did demonstrate they could identify emotions with a success rate between 62.50% - 72.50%. This study also examined the importance of the parameters used by the networks. While the fifteen parameters used did perform well with the artificial neural networks, the parameters were studied for improvement. Using principal component analysis, an optimal set of the parameters was found and used with the three networks. While the results were mixed using the optimal parameters, this showed the importance of selecting parameters that clearly distinguish between facial expressions. For instance, better parameters are needed to distinguish the fear emotion. None of the three networks did better then 50% identifying the fear emotion.

Chapter 6 Conclusions

The purpose of this present study was to compare different artificial neural network models, determine which model was best at recognizing emotions through facial images, and examine the importance of having a good set of parameters.

The conclusions of the study were:

1) The Hopfield demonstrated a better performance with the original fifteen parameters with a 72.50% accuracy and a 72.37% geometric mean. The difference between the best and worst performing network was 10%.

2) The three networks performed relatively the same with the fourteen optimal parameters. The best performing artificial neural network was the LVQ with a 67.50% accuracy and a 66.27% geometric mean. The difference between the best and worst performing network was 3.75%.

3) The fourteen optimal parameters derived using principal component analysis gave mixed results as the feedforward network improved, the LVQ stayed the same, and the Hopfield performance was decreased.

Chapter 7 Suggestions for Future Work

While this present study has demonstrated artificial neural network's abilities to identify emotions through facial images, further research is still needed. Listed below are some suggestions:

1) The artificial neural network models used in this study require more research to improve their performance in classifying facial expressions from facial images.
2) In the present study, the parameters were extracted manually from the facial image. Automated extraction methods have to be developed to extract parameters for use in real time application. This is needed for a fully automated emotion identification system.
3) Specialist neural networks for each expression (anger, fear, happiness and surprise) could be developed for sub-classification of expressions for improved accuracy.
4) Investigate other methods to determine the optimal set of parameters. Needed is a minimum set of parameters for effective facial expression classification. One example is the Factor Analysis.
5) Other artificial neural network models could be trained and tested on identifying facial expressions from facial images.
6) More artificial neural networks could be trained based on positive and negative expressions and then sub-classified into corresponding positive expressions (happiness and surprise) or negative expressions (anger, disgust, sad and fear).
7) The networks could be improved by testing them on a bigger facial expression databases.

References

Bartlett, M. S., & Sejnowski, T. (1997). Viewpoint invariant face recognition using independent component analysis and attractor networks, in Advances in Neural Information Processing Systems. *Neural Information Processing Systems - Natural and Synthetic*, 9, 817-823.

Busso, C., Deng, Z., Yildrim, S., Bulut, M., Lee, C. M., & Kazemzadeh, A. (2004). Analysis of emotion recognition using facial expressions, speech and multimodal information. *Sixthi ACM Int'l Conf. Multimodal Interfaces (ICMI '04)*, 205-211.

Coppin, B. (2004). Neural Networks. In (Ed.), *Artificial Intelligence Illuminated* (pp. 291-326). Sadbury, MA: Jones & Bartlett Publishers.

Darwin, C., The Expression of Emotions in Man and Animals, 1872/1965

Demuth, H., Beale, M., & Hagan, M. (1992-2009). Neural network toolbox 6 user's guide. Retrieved from http://www.mathworks.co.kr/access/helpdesk/help/toolbox/nnet/nnets_ug.html

Ekman, P., Friesen, W. V., & Hager, J. C. (2002). *Description of Facial Action Coding System*. Retrieved from http://face-and-emotion.com/dataface/facs/description.jsp

Ekman, P., Friesen, W. V., & Hager, J. C. (2002). *The Facial Action Coding System* (2nd ed.). Retrieved from http://face-and-emotion.com/dataface/facs/description.jsp

Hamilton, H. (2007). *Confusion Matrix*. Retrieved from http://www2.cs.uregina.ca/~hamilton/courses/831/notes/confusion_matrix/confusion_matrix.html

Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to The Theory of Neural Computation* . Redwood City CA : Addison-Wesley Publishing Company.

ImageTool website. (1996-2002). http://ddsdx.uthscsa.edu/dig/itdesc.html

Kanade, T., Cohn, J., & Tian, Y. (2000, March). Comprehensive Database for Facial Expression Analysis. *4th IEEE International Conference on Automatic Face and Gesture Recognition*, 46-53.

Kulkarni, S. S. (2006). *Facial Image Based Mood Recognition Using Committee Neural Networks* (Masters thesis, University of Akron). Retrieved from http://drc.ohiolink.edu/handle/2374.OX/3711

Lanitis, A., Taylor, C., & Cootes, T. (1997). Automatic interpretation and coding of facial images using flexible models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19(7)*, 743-756.

Lyons, M. J., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998). Coding Facial Expressions with Gabor Wavelets. *Int'l Cobf. Automatic Face and Gesture Recognition*, 200-205.

Mase, K. (1991). Recognition of facial expressions from optical flow. *IEICE Transactions*, *74(10)*, 3474-3483.

Negnevistky, M. (2005). *Artificial intelligence: A guide to intelligent systems* (2nd ed.). Harlow, England: Pearson Education Unlimited.

Rosenblum, M., Yacoob, Y., & Davis, L. (1996). Human expression recognition from motion using a radial basis function network architecture. *IEEE Trans. Neural Networks*, *7(5)*, 1121-1138.

SDL Component Suite. (2008). http://www.lohninger.com/helpcsuite/kohonen_network_-_background_information.htm

Smith, L. I. (2002). *A tutorial on Principal Components Analysis*. Retrieved from http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf

Su, M. C., Hsieh, Y. J., & Huang, D. Y. (2007). A simple approach to facial expression recognition. CEA'07 International Conference on Computer Engineering and Applications

The MathWorks website. (1994-2010). http://www.mathworks.com/

Wong, J. J., & Cho, S. Y. (2006). Facial emotion recognition by adaptive processing of tree structures. *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 23-30). Dijon, France: ACM. doi: http://doi.acm.org/10.1145/1141277.1141282