Emotional Stress and Drowsiness Detection in Drivers Using Support Vector Machine

JESSIE EMMANUEL ADANTE, Ateneo de Davao University
JAKE RANDOLPH MUNCADA, Ateneo de Davao University

Car accidents occur every day, and a significant quantity of accidents occur because of driver error. Some accidents happen because the driver is stressed or sleepy. Because of this, a real-time stress and drowsiness detection system is proposed, which recognizes emotions through the driver's facial expressions. The proponents propose using a Feature Extraction System using Computer Vision which extracts facial feature points from the driver's face. A Support Vector Machine is then used to recognize emotion. The video frame is sampled and each frame is processed to detect emotion. The percentage of frames in which stressful emotion is shown will be the basis whether the driver is stressed.

General Terms: Stress, Drowsiness, Emotion, Computer Vision, Support Vector Machine

Additional Key Words and Phrases: Facial Feature Point, Feature Extraction

1. INTRODUCTION

1.1 Background of the Study

Cars are one of the most used means of transportation in our world today. Driving from place to place has become the prevailing method of land travel. And as car use has gone up, accidents like car crashes and collisions have also been commonplace.

As humans we are emotional beings our emotions greatly affect how we make decisions we become irrational if we feel sad, angry, or helpless; while we are calm we are more aware of our surroundings. When a person is stressed their train of thought will suffer greatly. This will influence the way a person drives, and driving while under severe stress or anger has been known to cause accidents. Emotions play a great part in a person's ability to make decisions, and as such, this should also be taken into consideration when driving.

High level stress may damage self-confidence, narrow attention and eventually disrupt concentration. This may lead to accidents like car crashes. To lessen the risk, it is necessary to detect stress and take certain actions to relax the driver.

There have been previous works about stress detection. Some works are based on physiological features and use electromyograms and electrocardiograms, but such methods are intrusive. Other works use respiration, heart rate, and skin conductance, but these are bulky, limits movement, and generally impractical for drivers. Therefore, stress detection using computer vision is proposed, because the only apparatus is a dashboard camera and it doesn't restrict the driver.

1.2 Problem Statement

The study seeks to detect emotional stress using computer vision on a driver's facial expressions. Moreover, the study seeks to answer the following questions:

1. What are the different emotions that can be detected using computer vision?
2. How can computer vision detect a person's facial expression?
3. How can Support Vector Machines classify emotions based on a person's facial expression?
4. How can Support Vector Machines detect drowsiness based on the person's eyes?

1.3 Objectives

The study intends to detect emotional stress in a driver's facial expression using computer vision. Specifically, the study intends to accomplish the following objectives:

1. Find out the different emotions that can be detected using computer vision.
2. Explore how computer vision can detect a person's facial expression.

3. Find out how Support Vector Machines can classify emotions based on a person's facial expression.
4. Determine how Support Vector Machines can detect drowsiness.

1.4 Significance of the Study

Detecting stress in a driver is important because it can be used to warn a driver of a potential danger. High level stress may damage self-confidence, narrow attention and eventually disrupt concentration. This may lead to accidents like car crashes. To lessen the risk, it is necessary to detect stress and take certain actions to relax the driver.

1.5 Scope and Limitations

The study covers the detection of emotional stress through emotion recognition using computer vision.

The emotions detected will only base on the facial expressions and the six basic emotions (anger, disgust, fear, happiness, sadness, and surprise). The emotion will be identified without any degree or severity. The stress will be based on the amount of time the emotion was displayed.

This study uses Haar-like features as a cascade classifier for facial and facial feature detection. This requires the camera to be front-facing and upright position. The face will not be detected if and when it deviates from this position.

The system may not be as accurate if the driver has features that are different from the training data, for example if the driver is wearing eyewear or if the driver has significant facial hair. Characteristics like these may prevent the Haar cascades to improperly identify features.

Additionally, lighting plays a major factor in the system. Insufficient lighting will cause the program to not work. There must be adequate lighting for the system to function properly.

Moreover, the data set must acquire enough training data. The training images are obtained through the Take Picture button provided in the UI of the program. Insufficient data may result to lowered accuracy.

In the case of the system determining the driver is stressed, the system is not responsible for lowering stress. The system is only capable of detecting and possible informing the driver of possible stress.

2. REVIEW OF RELATED LITERATURE

2.1. Emotion Detection

Recognizing emotions is one of the problems under the field of affective computing. The crucial issue is selecting a suitable way/s to represent emotional states. Various features associated with stress, including hormone responses, physical appearance, speech, and physiological responses, have been utilized for stress detection. Some works combine different methods into a single system, like the works of Chen et al. [1] and De Silva et al [15]. They studied a combination of facial and vocal expressions as emotional representations. However, most studies use single methods separately. Some use human voices as emotional representation [35] [36] [37], others by physiological pattern recognition [38], some use hand gestures [39], body movements [40] [41] [42] or facial expression analysis. [43] However, traditional physiological-based detection methods are contact methods, i.e., sensors must be attached to individuals during feature measurement, which is not convenient for operation.

2.2. Face Detection

The goal of face detection is to determine whether or not there are faces in the image, and if yes, return the image location and extent of each face. [15] Face detection can be performed with a variety of methods. Caridakis et al. [28] used nonparametric discriminant analysis with a Support

Vector Machine (SVM) which classifies face and non-face areas, thereby reducing the training problem dimension to a fraction of the original with negligible loss of classification performance. [16] [17]

Huang and Huang [2] apply a point distribution model (PDM). In order to achieve a correct placement of an initial PDM in an input image, Huang and Huang utilize a Canny edge detector to obtain a rough estimate of the face location in the image. The valley in pixel intensity that lies between the lips and the two symmetrical vertical edges representing the outer vertical boundaries of the face generate a rough estimate of the face location. The face should be without facial hair and glasses, no rigid head motion may be encountered and illumination variations must be linear for the system to work correctly.

2.3. Pose Estimation and Correction

The accuracy of the feature extraction depends on the head pose. Due to pose variation or camera setup, the face pose in the test frames may not match the training data very well. Therefore, pose correction must first be performed.

To estimate the head pose, the left and right eyes are located in the corresponding eye candidate areas. After locating the eyes, pose is estimated by calculating the angle between the horizontal/vertical plane and the line defined by the eye centers. [28]

A 3D Cylindrical Head Model (CHM) [18] may also be used. It is applied with a simplified 3D face surface. Using the estimated head pose parameters, the CHM is rotated and 2D texture pixels are projected onto the CHM with bilinear interpolation.

After head pose is computed, the head is rotated to an upright position and new feature-candidate segmentation is performed on the head using the same rules so as to ensure facial features reside inside their respective candidate regions. These regions containing the facial features are used as input for the facial feature extraction stage.

2.4. Facial Feature Data Extraction

The facial features are the prominent features of the face - eyebrows, eyes, nose, mouth, and chin. In general, three types of face representation are mainly used in facial expression analysis: holistic [2], analytic [3] and hybrid [4]. Depending on the face model template-based or a feature-base method is applied for facial feature data extraction.

2.4.1. Template-Based Methods

Template-based methods fit a holistic face model to the input image or the track it in the input image sequence. In this method a training image is mapped using triangulation algorithms in order for the control points to match the mean of the shape. After the training data are generated a multivariate multiple regression analysis is applied to model the relationship between the model displacement and the image difference [5].

In image sequences, a holistic approach or a hybrid approach can also be used. To cope with large motions a coarse-to-fine gradient-descent strategy is used. By applying gradient-based optical flow algorithm you can estimate the motion of the facial areas. By taking advantage of the face's symmetry you can make an estimate in the distance of the facial features [7].

2.4.2. Feature-Based Methods

Feature-base methods localize the features of an analytic face model in the input image or track them in the input sequence. The face is mapped by outlining the features of the face; each feature acts as a base in order to calculate the distance of each part from one another. However this method has its shortcomings; faces with facial hair and/or glasses can obstruct the detection of the features [6].

In image sequences, facial points are placed by hand around the facial features in the first frame of the image sequence. For the rest of the frames a hierarchical optical flow method is used to track the flow of the landmark points. The displacement of each landmark point is

calculated by subtracting its normalized position in the first frame from the current frame. The displacement represents the facial information used for recognition of the displayed actions [8].

2.4.2.1. Artificial Neural Networks

Artificial neural network is a mathematical model inspired by the neural networks of the brain. It is a system that is composed of multiple processing elements called neurons that operate in parallel. It is an adaptive system that is used to find relationships in inputs and outputs or find patterns in the data.

ANN is similar to the brain in two aspects:

1. It obtains knowledge through learning.
2. Interneuron connection strength or synaptic weights are used to store the knowledge learned in the network.

Caridakis et al.[28] proposed using neural networks as a way to detect the boundaries of each of the facial features. Using neural networks, the boundaries were detected and feature masks were created and then combined to form the final facial feature mask.

2.5. Facial Expression Classification

By surveying the facial expression you can classify the encountered expression as a facial expression, an emotion or both. The mechanism used to classify the expressions are either template-based, neural-network-based or a rule-based classification.

2.5.1. Template-Based Methods

A template is used for comparison as defined by each expression category; the best match decides the expression's classification. The face and the parameters are extracted and then identify the pose and facial expression. The classifier assumes that the pose and expression is very similar for each individual [5].

In a sequence of images, the system will predict where facial points would displace between the frames of the image sequence. Then the portions are compared to templates; however there are multiple combinations of expressions while there are only a finite set of templates [12].

2.5.2. Neural-Network-Based-Methods

This uses a black box approach and can be considered a template approach, however this method uses multiple classes which is something that the template method can't perform. For the classification of the expression into one of the six basic emotions a neural-network is created by corresponding the input layers to the brightness distribution data extracted from an input facial image. The network is trained with images of the six basic facial expressions from various sets of subjects [10].

2.5.3. Rule-Based Methods

This method classifies the expressions based on previously encoded facial expressions; this method uses a prototypic approach by comparing current expressions from previous ones to check which expression fits the category. From the current expression the contours are subtracted from the expressionless face of the subject and by comparing the result with a previously obtain outcome the expression is classified [11].

In image sequences, motion parameters are used to derive the midlevel predicates that describe the motion of the facial features. Each midlevel predicate is represented as a rule;

each rule is then used to detect the beginning and ending expression. The rules are applied to the predicates of the midlevel [13].

2.5.4. Support Vector Machines

Support Vector Machine, or SVM, is a supervised learning approach to data mining. SVMs are part of the general category of kernel methods. [32] SVMs are used to classify and analyze patterns, and are widely applied to bioinformatics such as DNA or protein sequences or structures. [32]

Dumas [33] and Michel [34] proposed using Support Vector Machines for Emotion Detection/Classification. SVM was used to classify emotion through facial expression in images. They performed data extraction on user-defined training set of face images grouped by emotion. The trained SVM classifier is then able to classify the test data into its corresponding emotion.

2.5. Eye Movements Data

By analyzing the movement of the eyes and categorizing those into either alert driving data or sleepy driving data. With this data one can determine if the driver is drowsy and/or sleepy. To categorize those data blink frequency, PERCLOS, and gaze direction and fixation time are used.

2.5.1. Blink Frequency

The frequency is calculated by obtaining the difference of a start and end blink times of a specified time window and dividing it with the time window size [45].

2.5.2. PERCLOS

PERCLOS was proposed by Wierwille et al. first [44]. The percentages of the time the eyelids are closed are calculated using the first 100 sets of eye movement data. Each eye is calculated independently from one another due to it being important factor to eyelid sizes [45].

2.5.3. Gaze Direction and Fixation Time

In order to monitor driver state on gaze direction and fixation time, the fixation region is divided into two categories. The first category is normal fixation region and the other fixation region. Gaze direction is then calculated with the use of the two categories obtain and fixation time can be obtained from the gaze direction [45].

2.6. Theoretical Framework

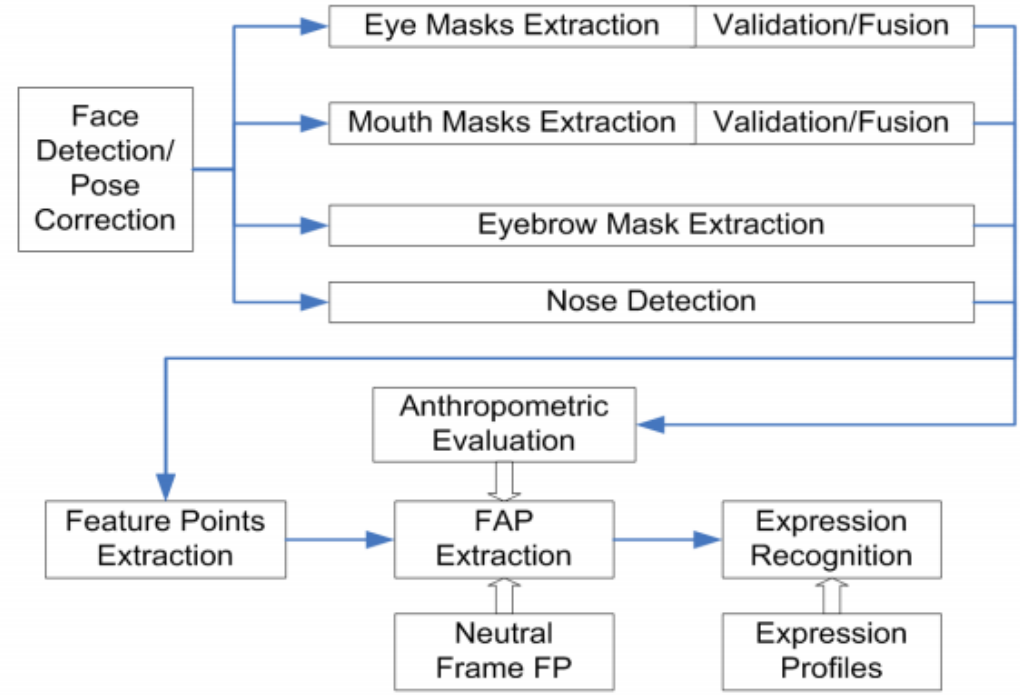


Figure 1: Theoretical Framework for Facial Feature Extraction [28]

Figure 1 illustrates the flow of how Caridakis et al.[28] extracted facial features and recognized emotion. First the face is detected and the pose is corrected. The face is divided into feature-candidate areas and then the facial feature masks are extracted. Then the final feature mask created by combining all the facial feature masks into one final mask. Next, the Facial Animation Parameters is extracted from the Feature Points and compared to the neutral expression. Based on the magnitude of the change in expression, the facial expression is recognized as one of the six basic emotions.

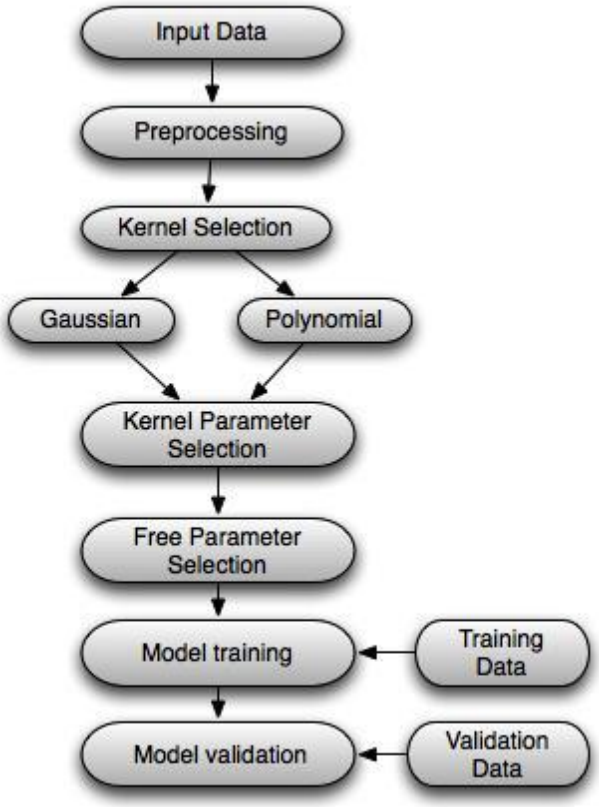


Figure 2: Support Vector Machine Diagram [29]

Figure 2 shows the flow for Support Vector Machines. Based from Vapnik, Osowski, [30]., the methodology used for the design, training and testing of SVM is [30]:

a. Preprocess the input data and select the most relevant features, scale the data in the range [−1, 1], and check for possible outliers.

b. Select an appropriate kernel function that determines the hypothesis space of the decision and regression function.

c. Select the parameters of the kernel function the variances of the Gaussian kernels.

d. Choose the penalty factor C and the desired accuracy by defining the ε-insensitive loss function.

e. Validate the model obtained on some previously, during the training, unseen test data, and if not pleased iterate between steps (c) (or, eventually b) and (e).

2.7. Theoretical Background

2.7.1 Feature Extraction

This section will discuss about feature extraction and algorithms that are used in feature extraction

2.7.1.1 What is Feature Extraction?

[W.Philpot, 2011],Feature Extraction starts from an initial set of measured data and builds derived values which are called features which are non-redundant and informative. Feature extraction involves reducing the amount of resources required to describe a large set of data. It defines the set of features that will efficiently represent the information that is important for analysis and classification.

2.7.1.2 Haar-like Features

Haar-like features are digital image features used in object recognition and was used in the first real-time face detection. Historically feature calculation was used and was computationally expensive. In the past the RGB pixels were compared to every pixel in the image in order to get the features. However Viola and Jones adapted the idea of using haar wavelets and developed haar-like features hence the name. Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, it sums up the pixel intensities and then gets the difference of the sums which is then used to categorize subsections of the image. The advantage of haar-like feature over other features is its calculation speed, since it uses integral images [46].

2.7.1.3 Canny Edge Detector

Canny edge detector, which was developed by John F. Canny, is an edge detection operator that uses multi-stage algorithm to detect a wide range of images. It can be broken down to five steps [47]:

a. Apply Gaussian filter to remove noise
b. Find intensity of gradients of image
c. Apply non-maximum suppression to get rid of spurious response to edge
d. Apply double threshold to determine edge
e. Track edge by hysteresis

2.7.1.3.1 Gaussian Filter

All edge detection results are easily affected by image noise so it is essential to filter them out. A Gaussian filter is applied to coil or entwine with the image, this will smooth the image to reduce the obvious noise on the edge detector. The equation for a Gaussian filter kernel is given by (8). The size of the Gaussian kernel will affect the performance of the detector [47].

$$g(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Figure 8: Formula for calculating Gaussian Filter

2.7.1.3.2 Finding the Intensity Gradient of the Image

An edge may point in a variety of direction, which is why filters must be used to detect horizontal, vertical and diagonal edges in the image. The edge detection operators returns a value for the first derivative in the horizontal and vertical direction (Gx,Gy). The edge gradient can be determined using (9) and edge direction can be determined using (10). The edge direction angle is rounded to one of four angles representing vertical, horizontal and the two diagonals [47].

$$\mathbf{G} = \sqrt{\mathbf{G}_x{}^2 + \mathbf{G}_y{}^2}$$

Figure 9: Hypot function for G

$$\mathbf{\Theta} = \operatorname{atan2}(\mathbf{G}_y, \mathbf{G}_x)$$

Figure 10: Arctangent function with two arguments

2.7.1.3.3 Non-Maximum Suppression

After applying the gradient calculation the edge extracted value is still blurred. Non-maximum suppression is applied to thin the edge. Non-maximum suppression can help to suppress all the gradient value to 0 except the local maximal, which indicates location with the sharpest change of intensity value. To obtain the pixels in the gradient image first you compare the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient directions. If the edge strength of the current pixel is the largest compared to the other pixels in the mask with the same direction, the pixel value will be persevered. Otherwise the value will be suppressed. In more accurate implementations, linear interpolation is used between the two neighboring pixels that straddle the gradient direction. It is also worth noting that the direction is irrelevant [47].

2.7.1.3.4 Double Threshold

After non-maximum suppression is applied there are still some edge pixels that are caused by noise and color variation. In order to get rid of these it is essential to filter out the edge pixel with weak gradient value and preserve the edge with the high gradient value. As such two threshold values are set to clarify the different types of edges pixels one is called high threshold value while the other is the low threshold value. If the edge pixel's gradient value is higher than the high threshold it is mark as a strong edge. If the pixel value is smaller than the high threshold but lager than the low threshold then it is marked as a weak edge. If the pixel is smaller than the low threshold then it is suppressed. The thresholds are defined when applying to the image [47].

2.7.1.3.5 Edge Tracking by Hysteresis

To achieve an accurate result the weak edges cause by noise and color variations should be removed. The criteria to determine which case the weak edge belongs to is that the weak edge pixel caused from true edges will be connected to the strong edge pixel. To track the edge connection, Binary Large Object-analysis is applied by looking at a weak edge pixel and its 8 connected neighborhood pixels. As long as there is one strong edge pixel involved the weak edge should be preserved [47].

2.7.2 MACHINE LEARNING

This section will discuss about machine learning and support vector machine (SVM).

2.7.2.1 What is Machine Learning?

Machine learning is a field of study that trains computers to learn without being programmed [A. Samuel, 1959]. It is a process where sample data sets are used to build a scientific model. Machine learning has to be evaluated quantitatively as it is hugely dependent upon the training data it is fed or other factors such as type of training, performance evaluation and the strength of the problem definition [48].

2.7.2.2 Support Vector Machine

A support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite-dimensional space, which can be used for classification, regression, or other tasks. To keep the computational load reasonable, the mappings used by the SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, using the kernel function to suit the problem. The hyperplanes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant. The vectors defining the hyperplanes can be chosen to be linear combinations with parameters of the images of feature vectors that occur in the data base [49].

2.7.2.2.1 Fisher Kernel

The Fisher kernel is used for a generative probabilistic model. It is used for term frequency-inverse document frequency, Naïve Bayes, and probabilistic

latent semantic analysis model. The Fisher Kernel can also be applied to image representation for classification or retrieval problems. This kernel can result in a compact representation, which is more desirable for image classification [49].

2.7.2.2.2 Graph Kernel

Graph kernels are functions that compute an inner product on graphs. These kernels can be intuitively understood as functions measuring the similarity of pairs of graphs. This allows learning algorithms to work directly on graphs without using feature extraction to transform them to a fixed-length [49].

2.7..2.3 Polynomial Kernel

The polynomial kernel looks at given features of input samples of the given feature as well as combination of these features. The feature space of a polynomial kernel is equivalent to that of polynomial regression, but without the combinatorial blowup in the number of parameters to be learned. The polynomial kernel is defined as (11) for degree d polynomials, where x and y are the vectors in the input space. Various ways of computing the polynomial kernel have been used as alternatives to the non-linear SVM algorithms, including full expansion of the Kernel prior to training or testing with linear SVM, basket mining and inverted indexing of support vectors [49].

$$K(x, y) = (x^{\mathsf{T}} y + c)^d$$

Figure 11: kernel polynomial

2.7..2.2.4 Radial Basis Function Kernel

The radial basis function is a real-valued function whose value depends only on the distance from the origin or alternatively on the distance from some other point c called a center. Any function that satisfies the property is a radial function. Sums of radial basis functions are typically used to approximate given functions. This approximation process can also be interpreted as a simple kind of neural network. RBF are typically used to build up function approximations in the form of (12), where the approximating function y(x) is represented as a sum of N radial basis function, each associated with a different center x_i, weighted by an appropriate coefficient w_i. Approximation schemes of this kind have been particularly used in the time series prediction and control of nonlinear systems [49].

$$y(\mathbf{x}) = \sum_{i=1}^{N} w_i \, \phi(\|\mathbf{x} - \mathbf{x}_i\|),$$

Figure 12: Function approximations

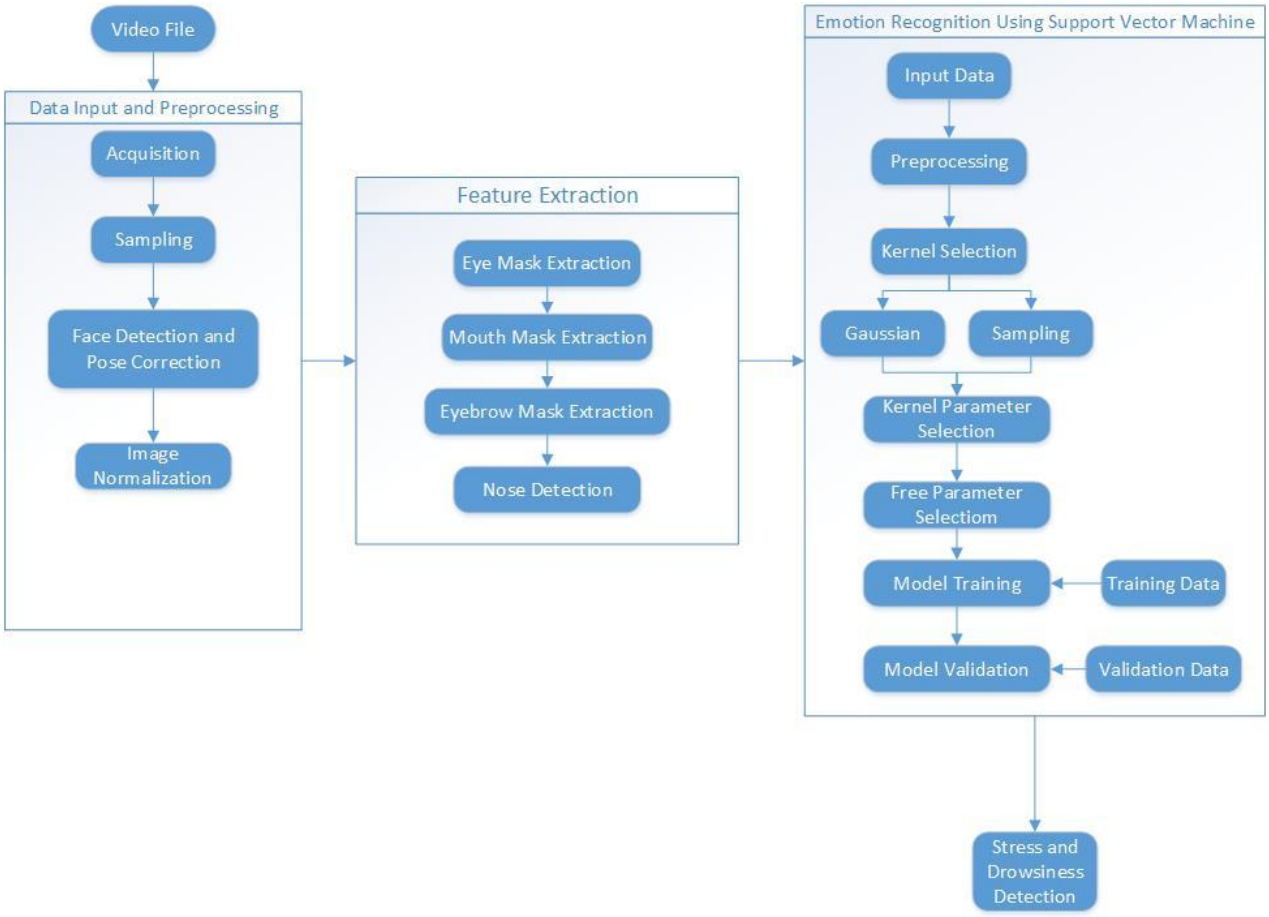
2.7.2.2.5 String Kernel

String kernel, as its namesake, is a kernel function that operates on strings such as finite sequences of symbols that does not need to be the same length. String kernel can intuitively understood as functions measuring the similarity of pairs of strings. Using string kernels with kernelized learning algorithms allows the algorithms to work with strings without having to translate these to fixed-length, real valued feature vectors. For several algorithms data enters into the algorithm only in expressions involving an inner product of feature vectors. The string kernel method is to be contrasted with earlier approaches for text classification where feature vectors only indicated the presence or absence of a word. This not only improves on the approach but it is an example for a whole class of kernels that can be adapted

to data structures. A desirable consequence of this is that one doesn't need to calculate the transformation only the inner product via the kernel which may be a lot quicker [49].

3. METHODOLOGY

3.1. Conceptual Framework



3.2. Methodology

In this study of stress detection through facial expressions using computer vision, the phases are as follows:

1. Data Input and Preprocessing
2. Feature Extraction Using Neural Network
3. Emotion Recognition Using Support Vector Machine
4. Stress Detection

3.2.1. Data Input and Preprocessing

The system's initial input is a video.

3.2.1.1. Sampling

The video acquired in the acquisition phase is then passed on to the sampler. Here, the video is sampled into frames. Each frame will be treated as an individual image.

3.2.1.2. Face Detection

In each image, it is determined whether or not there is a face. If so, the facial image's coordinates will be obtained. Haar-like features as a cascade classifier will be used to obtain the coordinates.

3.2.1.3. Image Normalization

The image will then be processed. Given the coordinates of the face, the non-significant areas of the image will be cropped and the image will be normalized into a standard size.

3.2.1.4. Feature Extraction

The normalized image frames are then passed on to the feature extraction phase where the Regions of Interest (ROI), are extracted. The Regions of Interest are the eyes, eyebrows, nose, and mouth of the subject. Each ROI is obtained also through Haar cascades.

3.2.1.4. Canny Edge Detection

Each ROI will then be passed to a Canny Edge detector. This process starts with a conversion to grayscale of the image. Then a Gaussian blur will be applied. Lastly, the Canny Edge detection is done. This returns a mask where the edges are shown in white pixels.

3.2.1.4. ROI Normalization

The edge masks will then be resized to a normalized resolution according to the facial feature. This is done so that the total number of pixels will be uniform throughout the dataset.

3.2.2. Emotion Recognition Using Support Vector Machine

The proponents intend to use SVM as an approach to classifying emotions represented by the facial feature points. SVM is a classifier algorithm that executes classification by creating hyperplanes in a multidimensional space which separates objects of different class labels. It uses an iterative training algorithm that is useful in minimizing error functions. The SVM classification of phonemes involves a training and a testing process. The training data will be obtained through the program, wherein a Take Picture button will be provided. The user will be required to provide images of his/her face showing the different emotions, as shown in Figure 5.

Figure 5: Sample images used for training emotion classifiers

3.2.2.1. Training

The SVM will be trained using the facial images provided by the user. The kernel used will be polynomial kernel, and the parameters will be tested in a range of values. The training data will be put in folders specifying whether it is stressed or not stressed.

3.2.2.1. Testing

Each SVM model given a set of parameters will be tested on a percentage of the dataset. These testing images should not be part of the training data. Each SVM model will be evaluated based on accuracy and the Kappa statistic.

3.2.3. Stress Detection

After each image frame will be classified as one of the six basic emotions or neutral, the number of frames that qualified as a stressful emotion will be counted against the number of total frames. If within a time window the percentage exceeds a certain threshold, the driver is considered as stressed.

3.2.4 Drowsiness Detection

After the eyes are located the gaze direction (GD) is obtained using the normal fixation region(NFR) and the other fixation region(OFR) by using (6) and the fixation time(FT) is obtained using (7).

$$GD = \begin{cases} 1, & \text{if } NFRX_{min} \le GDX \le NFRI_{max} \\ & \text{and } NFRY_{min} \le GDX \le NFRY_{max} \\ & \text{and } NFRZ_{min} \le GDX \le NFRZ_{max} \\ 0, & \text{other,} \end{cases}$$

Figure 6: Pseudocode for obtaining gaze direction

$$FT1 = t_0 \times f_0 \times \sum_{j=1}^{n_1} m_j, \quad \text{if } GD = 1,$$

$$FT0 = t_0 \times f_0 \times \sum_{j=1}^{n_2} m_j, \quad \text{if } GD = 0,$$

Figure 7: Formula for obtaining the fixation time

4. RESULTS AND DISCUSSION

4.1 Feature Extraction

4.1.1 Face Detection

The proponents of the study used Haar-like features to detect the faces in the image. The most significant advantage of using this is its speed. While other methods are more computationally expensive, Haar offers faster computation based on Haar wavelets instead of the usual image intensities. [46] However, using Python, it is not possible to multi-thread the process. It still causes a noticeable drop in the frame rate, from 30 fps to around 9 fps.

4.1.2. Facial Feature Extraction

After getting the coordinates of the rectangle bounding the face, the image is cropped to include only the face. The surrounding background is dropped to maximize efficiency. Having only the face as the image, the eyes, eyebrows, nose, and mouth are detected by again using Haar. One the coordinates of these facial features are obtained, the image is again cropped.

4.1.3. Canny Edge Detection

Each of the facial features will undergo Canny Edge Detection to find their outlines. First Gaussian blur is applied before Canny Edge. [47] The proponents had the choice between Canny Edge and OpenCV's Contour algorithm. Canny seemed to have the more suitable edge output when examined by eye.

4.1.4. Normalization

After getting the output edge, the images are normalized to provide uniform pixel count. The edge mask will be resized to a standard size depending on the feature. Each pixel will have a grayscale value of 0 to 255. These pixels will then be fed to the SVM for training or testing.

4.2. Support Vector Machine

4.2.1. SVM Parameters

When using SVC with Polynomial kernel, there are two parameters: C and degree.

C is the measure of complexity of the decision surface. Low C values will make decision planes smooth, but may be more prone to misclassification, while high C values will attempt to classify all input correctly, at the cost of having a more complex decision surface [50]. On the other hand degree is the degree of the curve of the polynomial.

4.2.1.1. Parameter Optimization

To use the SVM optimally, the proponents had to choose the correct C and degree for a Polynomial kernel SVM. The range used for C was from 1.0 to 10.0 and degree ranged from 2 to 4.

4.2.2. Evaluation Metrics

4.2.2.1. Accuracy

The accuracy is simply the number of instances that were classified correctly divided by the total number of samples. The higher the accuracy, the better.

4.2.2.2. Kappa Statistic

The Kappa Statistic is a measurement of the agreement between different, independent observers that are evaluating the same thing. The Kappa Statistic is calculated based on [51][52]:

		Observer 1 Result		
		Yes	No	Total
Observer 2 Result	Yes	a	b	m_1
	No	c	d	m_0
	Total	n_1	n_0	N

$$Kappa = \frac{(Observed\ Agreement - Expected\ Agreement)}{1 - Expected\ Agreement}$$

4.2.3. SVM Evaluation

The proponents conducted several tests to determine the optimum parameters for the SVM, i.e. the C and degree parameters. The process for this is as follows:

1. Divide the data set into two parts: the training and test sets. The proponents decided to test 30 stressed facial images and 30 not stressed facial images.
2. Cycle through the parameters. For C, go from 1.0 to 10.0 in 1.0 increments. For degree, cycle from 2 to 4.
3. Train the SVM with the train function using the parameters given.
4. Test the data set.
5. Record the parameters and the findings: accuracy and Kappa statistic.

The results are as follows:

C	Degree	Matrix	Accuracy	Kappa
1.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
1.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
1.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
2.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
2.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
2.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
3.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
3.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
3.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
4.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
4.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
4.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
5.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
5.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
5.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
6.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
6.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
6.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
7.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
7.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
7.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
8.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
8.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
8.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333

9.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
9.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
9.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333
10.00	2	[[25, 5], [5, 25]]	0.833333333	0.66667
10.00	3	[[22, 5], [8, 25]]	0.783333333	0.56667
10.00	4	[[18, 5], [12, 25]]	0.716666667	0.43333

The proponents noticed that it seems the most important parameter for Polynomial kernel is its degree. The accuracy and Kappa changes with the degree, and C doesn't seem to have any effect. The accuracy ranges from 71.67% to 83.33% while the kappa from 0.43333 to 0.66667.

Actual testing on webcam video shows that the result does indeed fluctuate between stressed and not stressed. This discuss by a variety of reasons. First, the Haar cascade classifier does not always properly detect the face or facial features. Second, the lighting affects the detection in both the Haar cascade classification and the Canny edge detection. Lastly, the SVM will have some errors as shown in the testing phase.

5. CONCLUSION AND RECOMMENDATIONS

After all the studies have been conducted, the proponents discovered that it is indeed possible to use Support Vector Machines in detecting the driver's stress and sleepiness. Using training data based on the user's facial expressions, SVM can correctly classify the different emotions as well as detect if the person is showing signs of drowsiness. In the study, the highest accuracy obtained was 83.33% while the highest Kappa Statistic was 0.6667. By using Haar cascades the system was able to created regions of interest and extracts the important facial features used for detecting emotions and sleepiness. Canny Edge Detection was then used to obtain the contours of specific facial features and these contours are used for both as training data as well as obtaining inputs via a live video feed.

There were a number of obstacles in accurately determining the classification of each facial expression. Because of these obstacles, it is uncertain whether this system will work under driving conditions and outside the lab. The results were obtained only through testing on a webcam in a well-lit room. Conditions like these are subject to change when applied to driving. In a car, the lighting will not be as optimal as in a room. This will probably cause the system to not work properly.

One recommendation is to add a lot more facial images to the training data. This should make the system more robust and accurate. As of the moment, the system will work only if the user's own face is in the database. The proponents recommend adding images of facial expressions of various people.

The system showed itself to be relatively slow. The frame rate dropped from 30 fps to around 8 fps when the facial feature detection is switched on. This observed frame rate may vary on different computers. Thus, the proponents recommend using a more powerful computer. Additionally, the computer's graphics processing unit (GPU) may be used to further improve the performance. Its processing power may be harnessed through the use of CUDA cores so that feature detection and stress analysis may be improved.

Another recommendation is to use 3D cylindrical head model for the facial extraction. This will help improve the accuracy of the feature extraction and pose correction. The current system only relies on Haar cascade classifiers, so the pose cannot be corrected. The 3D head model may also be

used to get the feature points instead of Canny Edge detection, further eliminating the chance of errors in the Canny Edge portion of the system.

The results of the study showed that the combination of Support Vector Machines with Computer Vision and Image Processing, particularly Haar-like cascade classifiers and Canny Edge Detection is viable and possible to use in a laboratory setting. The overall performance of the stress detection and emotion recognition is relatively accurate. However, further attempts at increasing the accuracy should be taken.

The results of the study may be used in future work in attempting to reduce the danger of driving while under stress. Another field of research that may benefit from this study is Affective Computing in Computer Vision, because this study shows the ability of SVMs to classify emotions.

REFERENCES

1. L.S. Chen, T.S. Huang, T. Miyasato, and R. Nakatsu, "Multimodal Human Emotion/Expression Recognition," Proc. Int'l Conf. Automatic Face and Gesture Recognition, pp. 366-371, 1998.
2. C.L. Huang and Y.M. Huang, "Facial Expression Recognition Using Model-Based Feature Extraction and Action Parameters Classification," J. Visual Comm. and Image Representation, vol. 8, no. 3, pp. 278-290, 1997.
3. A.L. Yuille, D.S. Cohen, and P.W. Hallinan, "Feature Extraction from Faces Using Deformable Templates," Proc. Computer Vision and Pattern Recognition, pp. 104-109, 1989.
4. K.M. Lam and H. Yan, "An Analytic-to-Holistic Approach for Face Recognition Based on a Single Frontal View," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 7, pp. 673-686, July 1998.
5. G.J. Edwards, T.F. Cootes, and C.J. Taylor, ªFace Recognition Using Active Appearance Models,º Proc. European Conf. Computer Vision, vol. 2, pp. 581-695, 1998.
6. H. Kobayashi and F. Hara, "Recognition of Mixed Facial Expressions by Neural Network," Proc. Int'l Workshop Robot and Human Comm., pp. 387-391, 1992.
7. T. Otsuka and J. Ohya, "Recognition of Facial Expressions Using HMM with Continuous Output Probabilities," Proc. Int'l Workshop Robot and Human Comm., pp. 323-328, 1996.
8. B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," Proc. Joint Conf. Artificial Intelligence, pp. 674-680, 1981.
9. G.J. Edwards, T.F. Cootes, and C.J. Taylor, "Face Recognition Using Active Appearance Models," Proc. European Conf. Computer Vision, vol. 2, pp. 581-695, 1998.
10. Z. Zhang, M. Lyons, M. Schuster, and S. Akamatsu, "Comparison between Geometry-Based and Gabor Wavelets-Based Facial Expression Recognition Using Multi-Layer Perceptron," Proc. Int'l Conf. Automatic Face and Gesture Recognition, pp. 454-459, 1998.
11. M. Pantic and L.J.M. Rothkrantz, "Expert System for Automatic Analysis of Facial Expression," Image and Vision Computing J., vol. 18, no. 11, pp. 881-905, 2000.
12. J.F. Cohn, A.J. Zlochower, J.J. Lien, and T. Kanade, "Feature-Point Tracking by Optical Flow Discriminates Subtle Differences in Facial Expression,' Proc. Int'l Conf. Automatic Face and Gesture Recognition, pp. 396-401, 1998.
13. M.J. Black and Y. Yacoob, "Recognizing Facial Expressions in Image Sequences Using Local Parameterized Models of Image Motion," lnt'l J. Computer Vision, vol. 25, no. 1, pp. 23-48, 1997.
14. L.C. De Silva, T. Miyasato, and R. Nakatsu, "Facial Emotion Recognition Using Multimodal Information" Proc. Information, Comm., and Signal Processing Conf., pp. 397-401, 1997.
15. M. H. Yang, D. Kriegman, N. Ahuja, Detecting Faces in Images: A Survey, IEEE Trans. on Pattern Analysis and Ma-chine Intelligence, Vol.24(1), 2002, pp. 34-58.
16. R. Fransens, Jan De Prins, SVM-based Nonparametric Discriminant Analysis, An Application to Face Detection, Ninth IEEE International Conference on Computer Vision, Volume 2, October 13 - 16, 2003.
17. ERMIS, Emotionally Rich Man-machine Intelligent System IST-2000-29319, http://www.image.ntua.gr/ermis
18. J. Xiao, T. Moriyama, T. Kanade, and J. Cohn, "Robust Full-Motion Recovery of Head by Dynamic Templates and Re-registration Techniques," International Journal of Imaging Systems and Technology, vol. 13, pp. 85–94,September 2003.
19. P. Ekman and W. V. Friesen, Unmasking the Face: A Guide to Recognizing Emotions from Facial Clues. Cambridge, MA:ISHK, 2003.
20. C. Izard, "Innate and universal facial expressions: Evidence from developmental and cross-cultural research," Psychol. Bull., vol. 115, pp. 288–299, 1994.
21. P. Ekman, Universals and Cultural Differences in Facial Expressions of Emotion. Lincoln, NE, USA: Univ. of Nebraska Press, 1971.
22. D. F. Dinges, R. L. Rider, J. Dorrian, E. L. McGlinchey, N. L. Rogers, Z. Cizman, S.K. Goldenstein, C. Vogler, S. Venkataraman, and D. N. Metaxas, "Optical computer recognition of facial expressions associated with stress induced by performance demands," Aviation, Space, Environ. Med., vol. 76, no. 6, pp. B172– B182, Jun. 2005.
23. J. Healey and R. Picard, "Detecting stress during real-world driving tasks using physiological sensors," IEEE Trans. Intell. Transportation Syst., vol. 6, no. 2, pp. 156–166, Jun. 2005.
24. W. Liao, W. Zhang, Z. Zhu, and Q. Ji, "A real-time human stress monitoring system using dynamic Bayesian network," in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., 2005, p. 70.
25. M. Haak, S. Bos, S. Panic, and L. J. M. Rothkrantz, "Detecting stress using eye blinks and brain activity from EEG signals," presented at the Proc. 1st Driver Car Interact. Interface, Prague, Czech Republic, 2008.
26. P. Ekman, "Universals and Cultural Differences in Facial Expression of Emotion," J. Cole ed. Nebraska Symposium on Motivation, vol. 19, pp. 207–282, 1972.
27. P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotionspecified expression," in IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2010, pp. 94–101.
28. Caridakis, G., Malatesta, L., Kessous, L., Amir, N., Raouzaiou, A., and Karpouzis, K. Modeling Naturalistic affective States via Facial and Vocal Expression Recognition. School of Electrical and Computer Engineering, National Technical University

of Athens, Politechniupoli, Zographou, Greece. , 2006.
29. Antonio, A.M., Artemio, S., Efren, G., Carlos, P.J., Manuel, R.J., Sandra, C., and Emilio, V. Advances in Airborne Pollution Forecasting Using Soft Computing Techniques. Facultad de Informatica, Universidad Autonoma de Queretaro, Mexico.
30. Vapnik, V., Golowich, S., Smola A.: Support method for function approximation regression estimation, and signal processing. Advance in Neural Information Processing System 9. MIT Press, Cambridge, MA. 1997.
31. C.L. Huang and Y.M. Huang, ªFacial Expression Recognition Using Model-Based Feature Extraction and Action Parameters Classification,º J. Visual Comm. and Image Representation, vol. 8, no. 3, pp. 278-290, 1997.
32. Ben-Hur, A., and Wetson, J. A User"s Guide to Support Vector Machines. Retrieved January 13, 2014 from http://matt.colorado.edu/teaching/categories/jsw7.pdf
33. Dumas, M., Emotional Expression Recognition using Support Vector Machines. Department of Computer Science, University of California, San Diego
34. Michel, P, Support Vector Machines in Automated Emotion Classification, Colleges of the University of Cambridge, Churchill College. Cambridge, England
35. T. Johanstone, R. Banse, and K.S. Scherer, ªAcoustic Profiles inPrototypical Vocal Expressions of Emotions, º Proc. Int'l Conf. Phonetic Science, vol. 4, pp. 2-5, 1995.
36. V.A. Petrushin, ªEmotion in Speech: Recognition and Application to Call Centers,º Proc. Conf. Artificial Neural Networks in Eng., 1999.
37. T.S. Polzin and A.H. Waibel, ªDetecting Emotions in Speech,º Proc. Conf. Cooperative Multimedia Comm., 1998.
38. R.W. Picard and E. Vyzas, ªOffline and Online Recognition of Emotion Expression from Physiological Data,º Emotion-Based Agent Architectures Workshop Notes, Int'l Conf. Autonomous Agents, pp. 135-142, 1999.
39. V.I. Pavlovic, R. Sharma, and T.S. Huang, ªVisual Interpretation of Hand Gestures for Human-Computer Interaction Review,º IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 677-695, 1997.
40. R.J. Holt, T.S. Huang, A.N. Netravali, and R.J. Qian, ªDetermining Articulated Motion from Perspective Views,º Pattern Recognition, vol. 30, no. 9, pp. 1435-1,449, 1997.
41. A.D. Wilson and A.F. Bobick, ªParametric Hidden Markov Models for Gesture Recognitionº IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 21, no. 9, pp. 884-900, Sept. 1999.
42. H.K. Lee and J.H. Kim, ªAn HMM-Based Threshold Model Approach for Gesture Recognition,º IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 21, no. 10, pp. 961-973, Oct. 1999.
43. M. Pantic, L. J.M. Rothkranzt, "Automatic Analysis of facial Expressions: The State of the Art ", December 2000.
44. W. W. Wierwille, L. A. Ellsworth, S. S. Wreggit, R. J. Fairbanks, and C. L. Kirn, "Research on vehicle-based driver status/performance monitoring: development, validation, and refinement of algorithms for detection of driver drowsiness," Final Report DOT HS 808 247, National Highway Traffic Safety Administration, 1994.
45. J. Lisheng, N. Qingning, J. Yuying, X. Huacai, Q. Yanguang, X. Meijiao, "Driver Sleepiness Detection System Based on Eye Movements Variables", Transportation College, Jilin University, Changchum, Jilin 130022,China, China-Japan Union Hospital of Jilin University, Changchun 130033, China , September 20 , 2013.
46. Viola and Jones, "Rapid object detection using a boosted cascade of simple features", Computer Vision and Pattern Recognition, 2001
47. Canny, J., *A Computational Approach To Edge Detection*, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
48. *C. M. Bishop (2006). Pattern Recognition and Machine Learning. Springer.*
49. *Ben-Hur, A., and Wetson, J. A User"s Guide to Support Vector Machines. Retrieved January 13, 2014 from* http://matt.colorado.edu/teaching/categories/jsw7.pdf
50. RBF SVM parameters. Retrieved September 29, 2014, from scikit-learn: http://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html
51. Thai, L.H., Hai, T.S., and Thuy, N.T.Image Classificaiton using Support Vector Machines and Artificial Neural Network. Retrieved January 8, 2014, from http://www.mecs-press.org/ijitcs/ijitcs-v4-n5/v4n5-5.html
52. Tzostos, A. A Support Vector Machine Approach for Object Based Image Analysis. Retrieved January 8, 2014 from ResearchGate: http://www.researchgate.net/publication/225929583_Support_Vector_Machine_Classification_for_Object-Based_Image_Analysis/file/d912f511bb0d3389b4.pdf