

Automatic Fatigue Detection System

T. Tinoco De Rubira, Stanford University

December 11, 2009

1 Introduction

Fatigue is the cause of a large number of car accidents in the United States. Studies done by the National Highway Traffic Safety Administration, estimate that about 1,550 deaths, 71,000 injuries, and 12.5 billion dollars in monetary losses are the cost of fatigue related car accidents each year. To prevent these accidents, there are a number of devices available in the market. Many of these products are small devices worn by drivers on their ear that generate an alarm when the driver's head falls forward. However, these devices produce the alarm after the driver is no longer in conditions to be driving. The alarm essentially makes the driver wake up and can itself be the cause of an abrupt reaction that can lead to an accident.

An alternative and better approach is proposed by [1]. Here, a computer vision-based system is described that keeps track of the eyes and detects the sleep onset of fatigued drivers. The proposed system uses template matching for detecting the state of the eyes. This approach however, is computationally intensive, it is subject-dependent and requires calibration routines for adjusting for light conditions. In this paper, we present a more robust alternative based on machine learning for detecting and tracking the fatigue level of a driver.

2 Approach

The National Sleep Foundation suggests a list of signs that can be used for determining when a driver is no longer in conditions for being driving. These signs are the following:

- Difficulty focusing and daydreaming.
- Frequent blinking and heavy eyelids.
- Trouble remembering the last few miles driven.

- Frequent yawning or rubbing eyes.
- Drifting from lane or tailgating.
- Trouble keeping head up.

From this list, we decided to focus on the detection of frequent blinking, heavy eyelids and frequent yawning for determining the fatigue level of a driver. We propose the following approach for performing this task: First, a video camera records data of a person while driving and sends the data in real time to a computer vision system that detects the driver's face within the video frames. Once a face image is extracted, a Support Vector Machine (SVM) classifies the image as being fatigued or not fatigued. The representation of the classification output is either a +1 or -1 and this number is used by a system that monitors the driver status. Specifically, the classification output is the input of a weighted running sum that increases rapidly when the classification output is +1 and decreases slowly with the presence of -1 outputs. The frequency at which the sum goes above a specified threshold can be used to track the fatigue level of a driver and detect the sleep onset with a safe margin. The details of the subsystems and the experiments carried in this research are presented in the next sections.

3 System Overview

The proposed system is divided into four subsystems. These are the video capture unit, face detection unit, fatigue detection unit and alert unit. The video capture unit is a module that records video data in real time of the driver's face. The video is sampled with a constant period and the sampled frames are sent to the face detection unit. For the purpose of this research, we used a camera to manually collect videos and converted these videos to

sequences of images that consisted of the frames sampled every second.



Figure 1: Images extracted from the videos.

The face detection unit is a module that receives a video frame from the video capture unit and uses a cascade of classifiers that work with haar-like features to detect the face within the video frame. Once the face is detected by the classifier, it is scaled to a size of 100x100 pixels and then sent to the fatigue detection unit. This system was implemented using Intel’s Open Source Computer Vision Library (OpenCV) with decision tree classifiers that were trained with human faces.

The fatigue detection unit is a module that consists of an SVM that classifies the face images in the categories of fatigued and not fatigued. The decision for implementing this module using an SVM was due to the binary nature of the posed classification problem, the efficiency of SVMs in working with high dimensional feature vectors and their flexibility in handling both linearly and nonlinearly separable data sets. This system was implemented using the LIBSVM library [2].

The alert unit is a module that consists of a weighted running sum that adds every output of the SVM. The idea is that this sum can be configured so that it rapidly increases when the SVM outputs +1, and slowly decreases towards zero when the SVM outputs -1. One can then compute the number of times in a fixed time window that the value of the sum goes above a specified threshold and use this to estimate the fatigue level of a driver.

4 Training the SVM

The formulation of the SVM used in this research is the one that is based on the following primal

optimization problem:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}w^T w + C \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i, i = 1, \dots, m \\ & \xi_i \geq 0, i = 1, \dots, m \end{aligned}$$

In this expression, the set $\{(x^{(i)}, y^{(i)}) \mid i = 1, 2, \dots, m\}$ represents the training examples with the corresponding labels.

To construct the training set, we first collected videos of 10 persons. In each of these videos, the persons being recorded were asked to perform three actions: First look at the camera with the eyes open, then pose for a few seconds with the eyes closed and then yawn. From each of these videos we extracted a sequence of 350x350 grayscale images sampled once every second. The total number of images obtained were 284. A subset of these is shown in Figure 1. We then used the face detection unit to extract the faces from the images and used these to form the training set. A sample of the extracted faces can be seen in Figure 2. An important result from using the face detection unit to extract the faces was that the images obtained had the eyes at the same level. This facilitated the subsequent steps of image filtering and feature selection since it allowed us to focus on fixed subregions of the images.

For testing the prediction performance of the SVM, we constructed a test set by separating a set of images that corresponded to a particular person that was completely removed from the training set. We decided to use this scheme, as opposed to selecting a random subset of the training set, because we wanted to make sure that the SVM was tested on faces of persons that it had not seen before. We believe that this scheme gives a better estimate of the generalization performance of the SVM.



Figure 2: Output images from the face detection unit.

The features that we used for training the SVM were the pixel values of the image. However, since

we were able to rely on the consistency of the face detection unit, resulting in the eyes being at the same level in all images, we used the pixels from only two fixed subregions of the images. These were an 80x30 subregion and a 50x40 subregion located to contain the eyes and the mouth of the person. By using only the pixel values of these regions, we decreased the size of the feature vectors from 10,000 to 4,400 without any penalty on performance.

To try to simplify the classification task, we processed the eye and mouth images in the following way: For the eyes, we first enhanced the edges and then filtered the images using a median filter twice. The first time using an environment of size 7 and the second using an environment of size 3. For the mouth images, we only processed them using a blur filter. We chose this procedure experimentally by executing the training algorithm with the processed images and looking at the number of support vectors. We found, among the processing schemes tried, that the one mentioned above resulted in the smallest number of support vectors. We concluded from this that the image processing scheme mentioned above performed better in separating the positive (fatigued) and negative (not fatigued) examples. Figure 3 shows the subregions used for creating the feature vectors and the effects of the image processing algorithm.

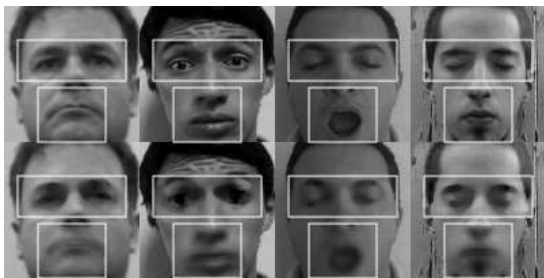


Figure 3: Subregions used for creating feature vectors and effects of image processing algorithm.

To visualize how the distribution of fatigued and not fatigued face images looked like, we used Principal Component Analysis (PCA) to project the data onto a three dimensional subspace. That is, from our training examples $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$, we constructed the matrix $\frac{1}{m} \sum_{i=1}^m (x^{(i)} - \mu)(x^{(i)} - \mu)^T$, where μ is the mean of the $x^{(i)}$, and computed the first six principal components of the data. We

found that the projection of the data onto the three dimensional space spanned by the fourth, fifth and sixth principal eigenvectors showed a separation between fatigued and not fatigued faces. This can be seen in Figure 4. We concluded from this analysis that the features and image processing scheme chosen provide enough information for performing the classification.

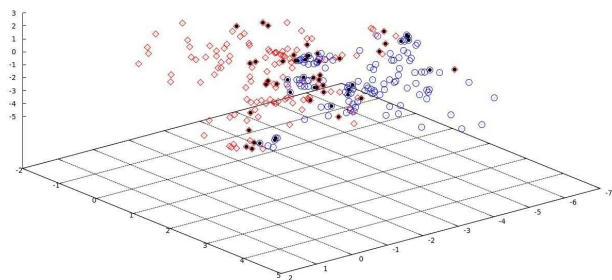


Figure 4: Projection of the data onto the three dimensional subspace spanned by the fourth, fifth and sixth principal components of the data. The red points correspond to fatigued faces, the blue points correspond to not fatigued faces and the black dots are the support vectors found after running the training algorithm.

Figure 5 shows the results of an experiment we carried to evaluate the effects of feature selection and image processing on the performance of the SVM. For this experiment, feature sets A and B correspond to all the pixel values of the eye and mouth images before and after applying the image processing algorithm. Feature sets C and D correspond to the histograms of the eye and mouth images before and after applying the image processing algorithm. To generate this data, we gradually incremented the training set size by adding a new person at a time, and then trained the SVM using each of the different feature sets. We see from the plot that the number of support vectors obtained when using feature set B is slightly lower than when using feature set A. As mentioned before, we concluded from this that the image processing algorithm implemented was in fact contributing to the separation of the data. We also see from the plot that the number of support vectors obtained when using the histogram values as features is much higher than when using all the pixel values. The histogram values were among some of the alternatives we ex-

perimented with in trying to find lower dimensional feature sets that could separate the data. Other features tried were the sums of the columns and rows of the images, but we also found that these did not provide any useful information that could help classification.

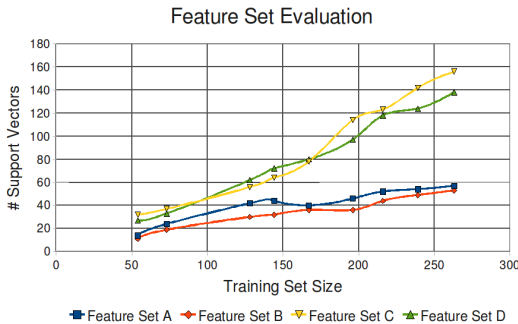


Figure 5: Number of support vectors as a function of training set size.

Figure 7 shows a similar experiment carried to evaluate the performance of the SVM using the feature sets described above. This data was generated by executing the learned hypothesis on the test set for various training set sizes and for each of the different feature sets. We see from the plot that, for the particular test set used, the image processing algorithm improved the performance of the SVM. A sample of the predicted labels of the test set is shown in Figure 6. We also notice that there is a significant increase in prediction accuracy when the training set reaches a size of 130 approximately. An explanation for this could be that at that point, the new person added to the training set looked similar to the person present in the test set. After this particular result, we decided to investigate the performance of the SVM on other test sets, namely, on the test sets formed with each of the other nine persons.



Figure 6: Example of labels predicted by SVM.

As explained in the previous paragraph, we found

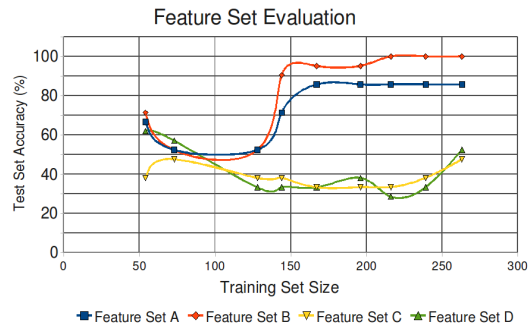


Figure 7: Prediction accuracy on test set as a function of training set size.

that the performance of the SVM on the test set improved significantly with the inclusion of a particular person to the training set. To understand this better, we evaluated the prediction performance of the SVM by using a customized cross validation scheme. Specifically, we tested the prediction performance of the SVM on each of the ten persons. For each trial, we picked one of the ten persons to form the test set and used the remaining nine to train the SVM. Once we did this for all ten persons, we averaged the percent accuracy obtained from each trial. Table 1 shows the results obtained.

Test Set	Prediction Accuracy (%)	
	Features A	Features B
Person 1	85.7	100.0
Person 2	94.5	94.5
Person 3	92.6	79.6
Person 4	75.9	100.0
Person 5	93.8	93.8
Person 6	100.0	100.0
Person 7	100.0	70.0
Person 8	89.5	94.7
Person 9	100.0	100.0
Person 10	95.8	100.0
Average	92.8	93.3
Std Dev	7.6	10.3

Table 1: Results of cross validation.

The data shows that for two test sets, the image processing algorithm actually lowered the prediction accuracy of the SVM. We see however that for all test sets, at least one of the feature sets

resulted in a performance of above 90%. We concluded from this that the system performed reasonable well in predicting the labels, and we discarded the possibility that the results obtained in the previous experiment, in which the prediction accuracy was very high, were a special case. The fact that at least one feature set performed very well suggests that perhaps an image processing algorithm that enhanced the edges less and did less smoothing could still achieve high prediction accuracy on average but with a smaller variance.

5 Conclusions

In this paper, we have proposed a system that uses machine learning for detecting the fatigue level of a driver. We saw from the experiments that with simple features and a simple image processing algorithm, we were able to obtain an average prediction accuracy of 93.3% by training the SVM only with nine persons. The data obtained also showed that there is room for improvement for obtaining high prediction accuracy with a smaller variance, that is, for obtaining a more robust system that performs well for a wide variety of faces. The next step in this research would be to try different image processing algorithms, increase the training set size, and execute the overall system in real time to investigate the performance of the proposed alert unit.

References

- [1] A. B. Albu, B. Widsten, T. Wang, J. Lan, and J. Mah "A Computer Vision-Based System for Real-Time Detection of Sleep Onset in Fatigued Drivers." 2008 IEEE Intelligent Vehicles Symposium.
- [2] Chih-Chung Chang and Chih-Jen Lin "LIBSVM: A Library For Support Vector Machines." 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [3] C. M. Bishop "Pattern Recognition and Machine Learning." 2006. Springer.