UNIVERSITY OF MINNESOTA

CENTER FOR
TRANSPORTATION STUDIES

INTELLIGENT
TRANSPORTATION
SYSTEMS
INSTITUTE

Finding What the Driver Does

Final Report

Prepared by:
Harini Veeraraghavan
Stefan Atev
Nathaniel Bird
Paul Schrater
Nilolaos Papanikolopoulos

Artificial Intelligence, Robotics, and Vision Laboratory
Department of Computer Science and Engineering
University of Minnesota

CTS 05-03

HUMAN-CENTERED TECHNOLOGY TO ENHANCE SAFETY AND MOBILITY

Technical Report Documentation Page

1. Report No.	2.	3. Recipients Accession No.
CTS 05-03		

4. Title and Subtitle	5. Report Date
Finding What the Driver Does	May 2005
	6.

7. Author(s)	8. Performing Organization Report No.
Harini Veeraraghavan, Stefan Atev, Nathaniel Bird, Paul Schrater, Nikolaos Papanikolopoulos	

9. Performing Organization Name and Address	10. Project/Task/Work Unit No.
University of Minnesota Department of Computer Science and Engineering 4-192 EE/CSci Building 200 Union Street SE Minneapolis, MN 55455	CTS project number 2004059
	11. Contract (C) or Grant (G) No.

12. Sponsoring Organization Name and Address	13. Type of Report and Period Covered
Intelligent Transportation Systems Institute Center for Transportation Studies University of Minnesota 511 Washington Avenue SE, Suite 200 Minneapolis, MN 55455	Final Report
	14. Sponsoring Agency Code

15. Supplementary Notes

http://www.cts.umn.edu/pdf/CTS-05-03.pdf

16. Abstract (Limit: 200 words)

Most research depends on detection of driver alertness through monitoring the eyes, face, head or facial expression. This research presents methods for recognizing and summarizing the activities of drivers using the appearance of the driver's position—and changes in position—as fundamental cues, based on the assumption that periods of safe driving are periods of limited motion in the driver's body.

The system uses a side–mounted camera and utilizes silhouettes obtained from skin–color segmentation for detecting activities. The unsupervised method uses agglomerative clustering to represent driver activity throughout a sequence, while the supervised learning method uses a Bayesian eigen–image classifier to distinguish between activities. The results validate the advantages of using driver appearance obtained from skin–color segmentation for classification and clustering purposes. Advantages include increased robustness to illumination variations and elimination of the need for tracking and pose determination.

17. Document Analysis/Descriptors		18.Availability Statement
Driver Activity Driver Distraction Learning Methods	Unsupervised Learning Behavior Skin Color Detection	No restrictions. Document available from: National Technical Information Services, Springfield, Virginia 22161

19. Security Class (this report)	20. Security Class (this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	24	

Finding What the Driver Does

Final Report

Prepared by:
Harini Veeraraghavan
Stefan Atev
Nathaniel Bird
Paul Schrater
Nikolaos Papanikolopoulos

Artificial Intelligence, Robotics, and Vision Laboratory
Department of Computer Science and Engineering
University of Minnesota

May 2005

Intelligent Transportation Systems Institute
University of Minnesota

CTS 05-03

Acknowledgements

The author(s) wish to recognize those who made this research possible. This study was funding by the Intelligent Transportation Systems Institute (ITS), University of Minnesota. The ITS Institute is a federally funded program administered through the Research & Innovative Technology Administration (RITA).

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

Executive Summary

This paper presents two different learning methods applied to the task of driver activity monitoring. The goal of the methods is to detect periods of driver activity that are not safe, such as talking on a cellular telephone, eating, or adjusting the dashboard radio system. The system presented here uses a side-mounted camera looking at a driver's profile and utilizes the silhouette appearance obtained from skin-color segmentation for detecting the activities. The unsupervised method uses agglomerative clustering to succinctly represent driver activities throughout a sequence, while the supervised learning method uses a Bayesian eigen-image classifier to distinguish between activities.
The results of the two learning methods applied to driving sequences on three different subjects are presented and extensively discussed.

CHAPTER 1 INTRODUCTION

Overview

The goal of this project is to develop a camera-based system for monitoring the activities of automobile drivers. As in any system deployed for monitoring driver activities, the primary goal is to distinguish between safe and unsafe driving actions. There is no fixed list of actions that qualify as unsafe driving behaviors. In general, an activity or an action that reduces a driver's alertness or awareness of their surroundings should be classified as unsafe driving behavior. Some examples of unsafe driving behavior include driver fatigue, talking on a cellular telephone, eating, and adjusting the controls of the dashboard stereo while driving.

In this work, we present methods for summarizing and recognizing the activities of a driver, using the appearance of the driver's pose as fundamental cues. The position of the hands, arms and the head vary across different activities, and vary among individual drivers. While there is a lot of work in driver activity monitoring through head and eye tracking [1], [9], [10], [13], [17], [20], there is very little work that makes use of the changes in the appearance resulting from the motion of the driver inside the automobile.

The skin-tone regions of the input video are used as the features in the classifiers. In the unsupervised method, binary skin-tone masks are agglomerated across an entire action sequence to assign a probability of observing skin tones for each pixel in the image during the action. Action sequences are separated from one another by detecting substantial movements in the image, signified by large differences between the skin-tone masks of sequential frames. In the supervised method, key frames corresponding to safe driving actions and unsafe driving actions are specified by the user. These key frames are used for obtaining the subspace densities corresponding to an individual action. In this work, talking on a cellular telephone is classified as an unsafe action. A Bayesian eigen-image method is used for classifying the activities.

Related Work

Most of the work on driver activity monitoring is focused on the detection of driver alertness through monitoring eyes [9], [10], [17], face, head, or facial expressions

[1], [13], [20]. In order to deal with the varying illumination, methods such as [21] use infrared imaging in addition to normal cameras. Learning-based methods such as [2], [19] exist for detecting driver alertness and gaze directions. In our work, both learning methods make use of the silhouette of the subjects for detection of activity. Several silhouette-based activity recognition methods exist in the literature such as the motion history image method by [6], the W4 system by [8], and the Pfinder system by [18]. The supervised learning or the Bayesian eigen-image method is based on the face recognition work of [11]. This method basically seeks a low-dimensional representation of the data for classification. Several dimensionality reduction techniques exist, such as [3], [4], [15], and the manifold learning methods in [5], [12]. An example of an unsupervised method for learning human behaviors is presented in [14], where a maximum likelihood method is used to learn the structure of a triangulated graph of feature point-based human motions. In [7], the general segment of the body region where significant motion takes place is detected, and this information is used as a cue for matching activities.

CHAPTER 2 UNSUPERVISED CLUSTERING OF DRIVING BEHAVIORS

The most basic cue about a driver's actions is his pose. However, tracking a driver's articulated motion in an environment with rapidly varying illumination and many potential self-occlusions is prohibitive both in terms of computational resources (for model-based tracking) and since the initialization of an articulated model is non-trivial in an automatic fashion. Our approach does not depend on an estimation of a driver's pose, but on the observation that periods of safe driving are periods of little motion of the driver's body. Of course, a driver does not move much while talking on a cellular telephone (an unsafe driving behavior), so the need arises to classify periods of minimal motion into safe-driving periods and unsafe-driving periods.

Detecting motion in a moving car's interior is complicated since the illumination of the interior can change very rapidly. Furthermore, the outdoor environment is visible through the car's windows, so motion will be always detected in the image regions corresponding to the car's windows. To address this problem, we only detect motion of skin-like regions, for example a driver's face and hands. This approach is advantageous since skin color detection can be fairly robust to various illumination conditions. Skin tones are also unlikely to appear in the window regions, so motion in the outside environment is unlikely to be detected. Portions of the car's interior that are misclassified as skin are static and will contribute nothing to the detected motion, so such regions are not problematic as well.

Skin Color Detection

We perform the classification of color pixels into skin tones and non-skin tones by working in the normalized RGB space. The normalization is effective against varying illumination conditions, and can also be motivated by the fact that human skin tones have very similar chromatic properties regardless of race [16].

An RGB triplet (r, g, b) with values for each primary color between 0 and 255 is normalized into the triplet (r', g', b') using the relationships:

Figure 2.1 Skin color detection (bottom row) on various images (top). Skin color is indicated in black. The results are post-processed by a sequence of morphological erosions and dilations.

$$r' = \frac{255 \cdot r}{r+g+b}, \quad g' = \frac{255 \cdot g}{r+g+b}, \quad b' = \frac{255 \cdot b}{r+g+b} \tag{1}$$

We classify a normalized color (r', g', b') as a skin color if it lies within the region of normalized RGB space described by the following rules (found in [16]):

$$\begin{gathered} r' > 95, \quad g' > 45, \quad b' > 20 \\ \max\{r', g', b'\} - \min\{r', g', b'\} > 15 \\ r' - g' > 15, \quad r' > b' \end{gathered} \tag{2}$$

Figure 2.1 shows the results of the skin color detection for various subjects and lighting conditions. It should be noted that other skin-tone detection methods can be used without affecting the rest of the algorithm. We tried using a non-parametric Bayesian skin probability map as an alternative approach, but its results were of unsatisfactory quality as the number of training images used to create the map was small and the images themselves were obtained under radically different lighting conditions than those during our driver monitoring experiments. However, if a better skin-color detection method is available, it can be substituted in favor for the rule-based one.

Detecting Changes in Behavior

Since our goal is to detect and classify relatively motion-free periods, we use inter-frame differencing to decide when a period starts and ends. If the change between two consecutive skin-color masks obtained by the color classification step is significant, the current low-motion period terminates. When the interframe difference drops, we start accumulating data about a new low-motion period.

Given the image region R, the change between two consecutive binary skin-color masks $I_{t\text{-}1} : R \rightarrow \{0, 1\}$ and $I_t : R \rightarrow \{0, 1\}$ is described by the total number of pixels whose classification changed:

$$c(t) = \sum_{p \in R} \left| I_t(p) - I_{t-1}(p) \right| \tag{3}$$

Whenever $c(t)$ is large, a transition in driver behavior is detected. A global threshold cannot be used to determine whether the change $c(t)$ is significant or not, since different low-motion actions differ in the typical amount of "natural" motion that occurs throughout the action. Additionally, the amount of noise in the skin classification masks may differ from one run of the algorithm to another. Finally, the significance of a change $c(t)$ depends on how much of a driver's skin is exposed. For these reasons, we chose to have a relative threshold for $c(t)$'s significance that depends on the observed variation in $c(t)$ over a period of time.

Assuming that a low-motion period started at time t_1, we consider the change at time t_n significant if $c(t_n)$ is more than 2 standard deviations away from the mean of the changes $c(t_{n\text{-}w})$, $c(t_{n\text{-}w+1})$, …, $c(t_{n\text{-}1})$, where w is the history window size (set to 900 frames, which corresponds to 30 seconds of past activity). Both the mean and standard deviations are computed incrementally. Since we start recording data for a new action immediately during the onset of the significant change, the deviation in the first few samples (i.e. $c(t_1)$, $c(t_2)$, …) is larger, which limits the number of spurious short periods identified by the algorithm. This is advantageous since the sequence of images leading to a low-motion action will contribute to the action model and thus will allow us to distinguish between otherwise similar low-motion periods based on information about the high-motion events that preceded them.

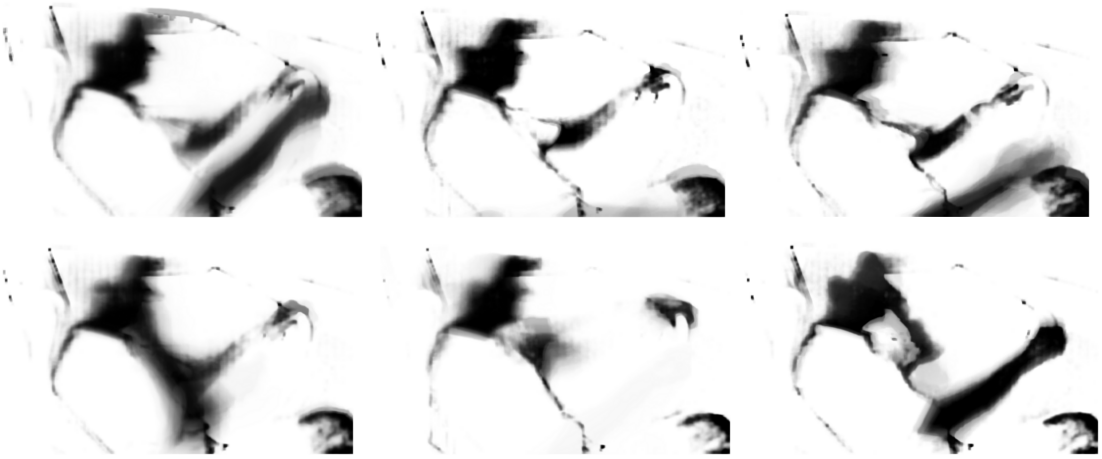

Figure 2.2 Skin probability maps for several action clusters (representing more than 80% of the driver's activity). Darker regions indicate higher probability of observing skin tones.

Action Models

The change in the binary skin tone masks indicates the need to start recording an action model. Each action model is simply a probability map that describes the expectation of observing a skin-color at every location in the input images. Given the binary skin masks $I_{t_1}, I_{t_2}, ..., I_{t_n}$ for a low-motion action with duration from time t_1 until time t_n, the probability map P is defined by:

$$P = \frac{1}{n}\sum_{t=t_1}^{t_n} I_t \tag{4}$$

Sample probability maps for several actions are shown in Figure 2.2. Individual actions that are determined to be similar are merged together into clusters. The goal of the clustering is to produce clusters that correspond to a single type of behavior (safe or unsafe). Such clustering facilitates further analysis of a driver's activities as it reduces tremendously the amount of data that needs to be analyzed (thousands of video frames versus tens of activity models). The similarity between an action model P and an action model Q is defined as:

$$d(P,Q) = \frac{\sum_{i \in R} \sqrt{P(i)Q(i)}}{\sqrt{\left(\sum_{i \in R} P(i)\right)\left(\sum_{i \in R} Q(i)\right)}} \tag{5}$$

The measure is the Bhattacharya coefficient for two normalized histograms, and ranges from 0 to 1. A high similarity measure corresponds to similar action models,

while a low measure corresponds to dissimilar models. A model is compared to the means of all clusters and merged with the most similar one if the similarity measure exceeds a certain threshold. Since we cluster according to the distance to the mean rather than the mean distance, each cluster can be represented by a single action model. The model P is merged into a cluster represented by the model Q according to:

$$Q(i) \leftarrow \frac{n}{n+m} P(i) + \frac{m}{n+m} Q(i), \tag{6}$$

where n and m are the number of video frames represented in P and Q, respectively.

CHAPTER 3 BAYESIAN EIGEN-IMAGE ACTIVITY CLASSIFICATION

For a side-mounted camera, the significant observable motion is the motion of the driver's hands in the image. However, one important issue with using hand motions is the problem of self-occlusion for extended periods of time and the resulting pose ambiguity. This problem precludes the use of region-based hand-trackers to detect hand motion and position. Instead, a snapshot representative of a particular action is used as the classification feature.

Figure 3.1. Input image for training.

Training Method

The goal of training is to find a representation for the images of a given class. Essentially, we want to find a low-dimensional representation for the data. Several methods such as the Karhunen-Loéve Transform, Principal Component Analysis, and eigen-images exist for computing the low-dimensional representation. We use the eigen-image method originally proposed by [15] for face recognition and extended later on by [11]. The method's robustness to illumination variations and self-occlusions can be achieved by using multiple suitable training images.

(a) Largest eigenvector for drive activity (b) Largest eigenvector for talk activity

Figure 3.2. Largest eigenvectors for drive and talk class.

For a given set of images corresponding to a given class, the largest eigenvalues and eigenvectors represent the distribution of the data along the most significant component direction. This is the basis of this method. For the given set of images, $I_i^1, ..., I_i^K$ belonging to a class C_i, an eigenvalue decomposition is performed to obtain Σ_i, the eigenvectors for the class C_i. This operation is performed off-line for each class of images.

Figure 3.2 shows the second largest principal eigen-image for the talk and drive actions. A typical image used for training is shown in Figure 3.1. As shown in Figure 3.1, only the skin portions of the image are chosen for training. The skin regions are detected automatically using the method described earlier in Chapter 2. This removes irrelevant portions of the image from consideration during training. Further, since only the skin portions corresponding to the hands and face are significant for the two classes, any skin segments around the leg regions are masked. This step helps reduce the dimensionality of the data, thereby improving the accuracy of the training with fewer training samples.

Figure 3.4 shows the effect of the number of training samples on the accuracy of classification for all samples belonging to the

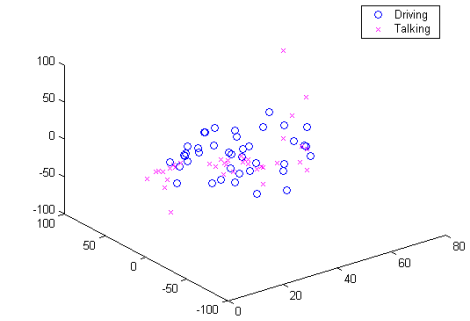

Figure 3.3. Distribution of the two classes under three largest principal components (starting from the second highest). The safe driving class is represented by circles and the unsafe driving class is represented by crosses.

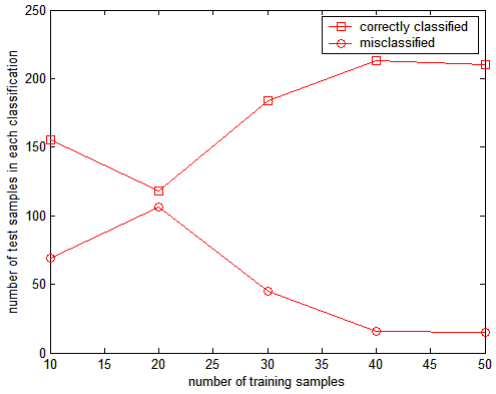

Figure 3.4. The results of classification for the training images of the unsafe driving class.

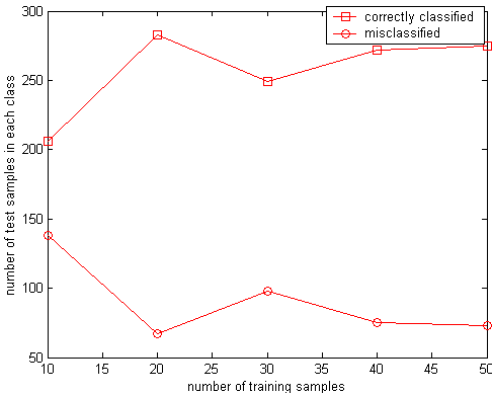

Figure 3.5. Classification results for the safe driving class on the training set.

talk action. The number of correctly classified and misclassified images for different training sample size is shown in Figure 3.4. Based on Figure 3.4, we chose a sample size of 40 where the number of correctly classified samples is maximized and the number of incorrectly classified samples is minimized.

Figure 3.5 shows the results of classification for samples containing the driving action for different sample sizes. Finally, Figure 3.3 shows the distribution of the two classes along the three principal components. The largest eigenvalues and eigenvectors capture the largest variation within a class. However, given the small training data size compared to the dimensionality of the data, the errors in skin segmentation are also modeled. Hence, we take the eigenvalues and eigenvectors starting with the second highest principal component for classification.

Activity Classification

The activity in each frame is evaluated by computing its similarity with the set of training images for each activity class. A probabilistic measure of similarity is used instead of the usual Euclidean metric. Given a candidate image I_x, its similarity to an image I_i^j from class C_i is computed by projecting the difference of the two images, $\mu = I_x - I_i^j$ onto the principal eigenvectors of class C_i. This can be represented as,

$$P(\mu \mid C_i) = \frac{e^{-\frac{1}{2}\mu^T \Sigma_i \mu}}{(2\pi)^{d/2} |\Sigma_i|^{1/2}}, \tag{7}$$

where Σ_i contains the largest eigenvectors for class C_i and *d* is the dimensionality of the data. This operation is repeated over all member images of a class until a maximum score is found. For recognizing the activity, this operation needs to be performed over all the training images in all the classes. This computation can be very expensive as the number of classes and the number of images increases. To reduce the computational burden, an off-line whitening transformation is performed as described in [11]. Each of the $I_i^1, ..., I_i^K$ images in class C_i are transformed using the eigenvalues and eigenvectors:

$$im_i^j = D_i^{-\frac{1}{2}} S_i I_i^j, \tag{8}$$

where D_i and S_i are the eigenvalues and eigenvectors computed for the class C_i. Given these pre-computed transformations, the match for a new image I_x is computed as:

$$P(\mu \mid C_i) = \frac{e^{-\frac{1}{2}\left\| im_x - im_i^j \right\|^2}}{(2\pi)^{d/2} \left|\Sigma_i\right|^{1/2}}, \tag{9}$$

where im_x is the transformed image of I_x computed from the eigenvectors and eigenvalues of C_i as in Equation (8). The activity in a frame is classified as safe driving or talking based on the relative values of $P(\mu \mid Driving)$ and $P(\mu \mid Talking)$. Activities having almost equal probabilities for both classes are rejected and not classified as belonging to either class. That occurs when the probability of association is in the range from 0.45 to 0.55.

CHAPTER 4 RESULTS AND DISCUSSION

Experimental Setup

Table 4.1. Unsupervised method results.

Subject	Frames	Clusters (Singletons)	Confusion
1	10,231	33 (24)	4.54%
2	10,110	38 (23)	16.8%
3	10,380	16 (8)	11.1%

Test data for the methods is comprised of three example videos of individuals pretending to drive a stationary automobile. The video camera used to record the videos was placed on a tripod directly outside the passenger-side window viewing the driver in profile. Each video features a different individual sitting in the car pretending to drive; different ethnicities and genders are represented. The lighting conditions vary throughout the videos as the car was in an outdoor parking lot. Each of the three videos is about six minutes long (between 10,500 and 11,000 frames), full-color, and at full 720 × 480 resolution.

During the course of each video, the driver goes through periods of driving normally and performing distracting actions. Distracting actions include talking on a cellular telephone, adjusting the controls of the dashboard radio, and drinking from a soda can. These actions were chosen as the unsafe behaviors to test for because they are very common.

Activity Clustering

The goal of the clustering method is to produce as few activity clusters as possible, while not merging together safe and unsafe activities. If safe and unsafe activities are merged together, the subsequent classification of clusters into safe and unsafe activities will introduce errors. If too many clusters are created, the method would have failed its goal to summarize a driver's activities. We tested the method using different settings for the similarity threshold. Table 4.1 shows the performance of the clustering using a threshold value of 0.85. The total number of clusters corresponds to the number of distinct activities recognized by the method. Singleton clusters are clusters that contain only one action model—such clusters usually reflect short periods of high motion indicative of transitions between different actions.

Each sequence was manually segmented into safe driving periods and unsafe driving periods. Since the goal of the clustering method is to group activities for further analysis, it must not group together activities from the two different classes. The proportion of incorrectly merged frames is indicated in the last column of Table 4.1.

The majority of the incorrectly clustered action models represents failures of the skin-color segmentation. For subject 1, the forearm was not segmented properly on several occasions. Subject 2 has no experience driving and was constantly in motion during the whole sequence. The head pose for subject 2 varied significantly during both safe driving and unsafe driving periods, which contributes to the higher confusion. Finally, subject 3's results suffer from under-segmentation, but improve significantly if the similarity threshold is increased. We suspect that this is due to the fact that subject 3's skin-color masks had fewer skin pixels as compared to the other subjects, primarily due to skin segmentation failures. Our future work on the unsupervised method will concentrate on making the similarity threshold relative rather than absolute and on improving the skin-color segmentation further.

Activity Classification

The supervised Bayesian eigen-image method was tested on the same subjects and sequences as the unsupervised method. Training images were free of noise in segmentation, and irrelevant parts of the scene were masked out. The test sets were used as is. Some of the training images had the leg portions masked out for improving the accuracy of the training. Training images were excluded from the test sequences. For training, 20 examples of safe driving and 40 examples of unsafe driving were used. The test set was comprised of 963 test samples.

The system correctly classified 95.84% of the safe driving activity, and 73.91% of the unsafe driving activity frames. 1.6% of samples in the safe driving activity class were misclassified as unsafe activity and 14.35% of samples in the unsafe activity were misclassified as a safe driving activity. 11.74% of samples in unsafe activity were detected in both classes, as were 3.16% of the samples in the drive activity. The main causes of misclassification were:

(a) Bad skin-tone segmentation (b) Pose ambiguity due to self-occlusion

Figure 4.1. Bad segmentation and self-occlusions can affect the accuracy of classification.

- Noise in the segmentation of the test frames.
- Ambiguous posture of the subjects in either class.

Noise in segmentation of the skin portions of drivers resulted mostly from extreme saturation of the color image due to very bright illumination and in some cases, the coloration of the driver's clothing. An example of a poorly segmented image is shown in Figure 4.1 where the subject's hands were under-segmented resulting in poor classification. Another source of misclassification was the result of ambiguous posture of the driver. For example, a driver leaning too close to the window was misclassified. Another case was when only one of the hands was visible due to self occlusion. In this case, the safe driving activity was confused with the unsafe activity where only one hand is in contact with the steering wheel. An example is shown in Figure 4.1 where the system detected the driver to be in either safe or unsafe driving states.

While the supervised learning method obtains high classification accuracy, its main drawback is that it is unsuitable for real-time driver activity classification. The supervised learning method includes using two distinct classes and across different subjects to account for the variability in the appearance of the hands and arms with subjects. However, training for only two classes limits the performance of the system when applied to detect activities such as adjusting the dashboard radio controls.

In this work, our main focus was in distinguishing safe versus unsafe driving activities in general. One extension would be to detect different subsets under each class of activities, in particular the unsafe driving class. Instead of using only one camera and the appearance cue, we would like to extend this work to using multiple cues obtained from multiple cameras, charting such observations as eye gaze and head motion.

Although the two learning methods are employed separately, one extension would be to use the two methods together. In other words, the supervised learning method can be used to classify the clusters generated by the unsupervised method. This would allow the system to collect information about a driver's activities in an online fashion, since individual clusters are produced or updated only when a change in behavior is detected.

Conclusions

We have presented two different methods for monitoring driving activities under challenging imaging conditions. The results obtained validate the advantages of using driver appearance obtained from skin-color segmentation for classification and clustering purposes. Specific advantages of this approach are the increased robustness to illumination variations and elimination of the need for tracking and pose determination.

REFERENCES

1. S. Baker, I. Matthews, J. Xiao, R. Gross, T. Kanade, and T. Ishikawa, "Real-time non-rigid driver head tracking for driver mental state estimation," In *11th World Congress on Intelligent Transportation Systems*, October 2004.

2. S. Baluja and D. Pomerleau, "Non-intrusive gaze tracking using artificial neural networks," Technical Report CMU-CS-94-102, Carnegie Mellon University, 1994.

3. M. S. Bartlett, J. R. Moverllan, and T. J. Sejnowski, "Face recognition by independent component analysis," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450-1464, November 2002.

4. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. "Eigenfaces vs fischerfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.

5. M. Belkin and P. Niyogi. "Laplacian eigenmaps for dimensionality reduction and data representation." *Neural Computation*, vol. 15, no. 6, pp. 1373-1396, 2003.

6. A Bobick and J. Davis, "The representation and recognition of action using temporal activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257-267, 2001.

7. J. Gao, R. T. Collins, A. G. Hauptman, and H. D. Wactlar, "Articulated motion modeling for activity analysis," In *IEEE Workshop on Articulated and Nonrigid Motion, held in conjunction with CVPR 2004*, 2004.

8. I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809-830, August 2000.

9. Q. Jia and X. Yang, "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance," *Real Time Imaging*, vol. 8, no. 5, pp. 357-377, October 2002.

10. C. Jiangwei, J. Linsheng, G. Lie, G. Keyou, and W. Rongben, "Driver's eye state detecting method design based on eye geometry feature," In *Intelligent Vehicles Symposium*, pp. 357-362, June 2004.

11. B. Moghaddam, T. Jebara, and A. Pentland, "Bayesian face recognition," *Pattern Recognition*, vol. 22, no. 11, pp. 1771-1782, November 2000.

12. S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, no. 290, pp. 2323-2326, December 2000.

13. P. Smith, M. Shah, and N. da Vitoria Lobo, "Eye and head tracking based methods—determining driver visual attention with one camera," *IEEE Transactions on Intelligent Tranportation Systems*, vol. 4, no. 4, pp. 205-218, December 2003.

14. Y. Song, L. Goncalves, and P. Perona, "Learning probabilistic structure for human motion detection," In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 771-777, December 2001.

15. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, 1991.

16. V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques," In *Proceedings Graphicon '03*, pp. 85-92, September 2003.

17. E. Wahlstrom, O. Masoud, and N. Papanikolopoulos, "Vision-based methods for driver monitoring," In *IEEE Intelligent Transportation Systems Conference*, vol. 2, pp. 903-908, October 2003.

18. C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780-785, 1997.

19. X. Liu, F. Xu, and K. Fujimura, "Real-time eye detection and tracking for driver observation under various light conditions," In *IEEE Intelligent Vehicle Symposium*, June 2002.

20. Y. Zhu and K. Fujimura, "Head-pose estimation for driver monitoring," In *IEEE Intelligent Vehicles Symposium*, pp. 501-506, June 2004.

21. Z. Zhu, K. Fujimura, and Q. Ji, "Real-time eye detection and tracking under various light conditions," In *Proceedings Symposium on Eye Tracking Research and Applications*, pp. 134-144, ACM Press, 2002.