Blink Rate Monitoring for a Driver Awareness System

David Tock and Ian Craw

Department of Mathematical Sciences *

University of Aberdeen, Scotland

1 Introduction

We describe a working prototype computer vision system for measuring the blink rate of car drivers. It combines an intelligent face recognition system with feature tracking, and statistical image analysis methods to locate and track the head position of a driver in a car, and extract specific measurements from the eyes. This is done using standard hardware (a Sun IPC workstation and frame grabber) without the use of image processing or other specialised hardware. The information obtained is used to give an indication of driver *alertness* which forms one of the inputs to a complete driver monitoring system.

2 Background

Recent studies [10, 8] (and others in Germany, Israel, the UK and USA) have shown that the majority of road accidents take place in good driving conditions, and can not be attributed to bad weather, alcohol/drugs or mechanical failure. The obvious conclusion is that such accidents are caused by driver error. The clustering of accidents around the hours 1am to 6am further indicates a relationship with driver fatigue. As part of the European PROMETHEUS programme, a system is being developed to monitor the driver's status. Although the aim is to obtain this information directly from sensors on the vehicle and its controls, there is no obvious correlation between these sensor inputs and driver status. Currently researchers at Stirling University are developing a neural network system to integrate these sensory inputs, but that still leaves the problem of correlation to driver alertness

Past research has shown a good correlation between a persons alertness and their blink rate, so this has been chosen to provide the necessary reference input with which to train the neural network.

As the system must work in a car, and will be used for extensive testing both on the road and on simulators, a non intrusive, non contact method of measurement was essential. This ruled out the use of special glasses or helmets, or methods which involved the driver keeping their head in a fixed position. The method chosen was to mount a video camera on the dashboard pointing towards the driver and to extract the measurements visually.

*This research was supported by a grant from the Ford Motor Company Limited

BMVC 1992 doi:10.5244/C.6.54

3 Design Considerations

The system we describe deals with two aspects of the problem independently. Firstly the position of the drivers head, or more specifically, the drivers eyes must be determined; and secondly, the eyes must be measured in some way to identify blinks.

A typical blink lasts approximately 200ms, so a frame rate of at least 5 frames per second must be achieved to avoid blinks occurring entirely between frames[1]. Obviously, a higher frame rate is desirable, and would make detection both easier and more reliable.

One possible approach was simply to develop FindFace, our existing face recognition system [17, 4], to perform the task. This may have been possible with considerable refinement and tuning, but would still fail to address some of the problems associated with working within a vehicle. Alternatively we could have developed systems designed to locate eyes in images, such as Nixon [12], Yuille *et al* [19], Hallinan [6] or Bennett and Craw [2]. All of these systems are designed to work with single images, and would not exploit the benefits of working with an image sequence.

Image sequence analysis is a well documented area of machine vision ([9, 13, 16] give general overviews) which attempts to extract particular information from an image sequence. Some of the more common objectives include;

- to determine the pose (position and attitude) of a known object moving in an otherwise stationary scene (e.g.[7]);
- to determine the shape and structure of an unknown object (or objects) moving in an image sequence (e.g. [14]); and
- to determine the position and movement of the camera within the environment it is viewing (e.g. [3]).

Although our objective differs from these, it shares some common ground with each.

We know approximately what we are looking for in an image, but not well enough to predict its appearance – everyone's face is different. We do not need to use movement to determine the 3D shape of the face as we are only interested in 2D appearance, and we must contend with the the problems of the image changing due to camera movement. Furthermore, the drivers head may remain stationary for long periods of time, or move only very slowly. This makes the *optical flow* approach inappropriate, and the *point correspondence* approach requires as input the very data we are hoping to obtain via the sequence analysis, namely the location of fixed points on the moving object. Despite these problems, there are advantages to be gained from the redundancy of data inherent in such a sequence.

Under most driving conditions camera vibration can be eliminated by mounting it securely to the car's bodywork. We then treat the interior of the vehicle as a stationary scene in which the driver moves. No suitable camera position could be found that gave adequate coverage of the driver without

[1] Failing to detect occasional blinks will not upset the results – the important measurement is the change in frequency of blinks over a (relatively) long period of time.

including window regions. The camera can therefore *see* out of the side and back windows. A number of idea were considered, such as polarising the windows and camera lens so as to eliminate most of the outside light, but these ideas were rejected due to the possibility of interference with the driver's visibility. The system must therefore cope with a considerable amount of changing background.

A further complication that the system must contend with is poor contrast and rapidly changing lighting on the drivers face. We have not tackled the problems of operating in night conditions[2]; but even ignoring these problems, because of the environment, the drivers face is usually in the shadow caused by the vehicle. The images obtained have either very low contrast when in the shade, or very high contrast when directly illuminated. These conditions change rapidly when a vehicle is in motion. Obviously, adding visible light sources to the vehicle is inappropriate on the safety grounds. Adding IR or UV illumination may be acceptable.

The camera's field of vision and the lighting characteristics thus limit the effectiveness of using image sequence information for tracking the head. Furthermore, they also presented problems for performing the eye measurements on single images using the FindFace system. Initial experiments with our usual technique for outline location [2] failed to perform satisfactorily, locating onto regions of higher contrast in the background. Similarly the eye detectors, given the poor contrast of the eye region, produced results of unacceptably low accuracy and reliability. We investigated some inter-frame relationships, such as simple differencing between images, but as expected this produced unacceptable levels of noise and spurious response. More significantly, these early efforts revealed a deficiency in our equipment which resulted in corresponding pixels in successive eight bit images varying by up to ±15. This was believed to be due to an interaction between camera and frame grabber, and although no cure could be found, the work had to proceed (at least initially) with this equipment, despite the atrocious signal/noise level. The variation was not random noise, rather a slow undulation making single images look perfectly normal, with the problem only apparent on sequences.

4 System Overview

The system that developed has two independent subsystems; a fast and simple algorithm which performs the tracking and eye measurement functions based on certain assumptions, and a slower, more reliable *watchdog* system that confirms these assumptions and hence that the fast system is performing correctly.

4.1 Feature Tracking System - Stage I

Due to the problems mentioned above, simple inter-frame differencing produces poor results. We do however know what the image without a driver should look like. The system can be initialised at regular intervals when no driver is present, either automatically by the watchdog system, or manually. Alternatively, a

[2] There are plans to investigate both NIR (near infra-red) and UV (ultra violet) illumination, and high sensitivity cameras.

number of images obtained under various conditions can be stored, and the most appropriate selected. These constitute *static* images, and give the system a base on which to build. As the system is only being used under controlled conditions at present, the driver can be ask to re-initialise it manually from time to time; for simplicity this approach has been adopted at present. In contrast we refer to images obtained at run time as *dynamic* images, and these are obtained as required by the system.

Rather than performing a simple difference between two dynamic images, or a dynamic image and a static image, a three way interest operator is applied.

We identify an image m with an array $\{m_{ij}\}$ and as usual refer to m_{ij} as a pixel value. Our initial images have $0 \leq m_{ij} \leq 255$ and $0 \leq i, j \leq 127$, but recall the bottom three or four bits are subject to high levels of noise. We write $\bar{m}$ for the mean of the pixel values of the image m, and m^{σ} for the corresponding standard deviation. Our system is concerned with images s, p and c, respectively the static, previous and current images. Typical static and dynamic images are shown in figures 1 and 2; due to the noise, these must be preprocessed before the location/tracking stage.

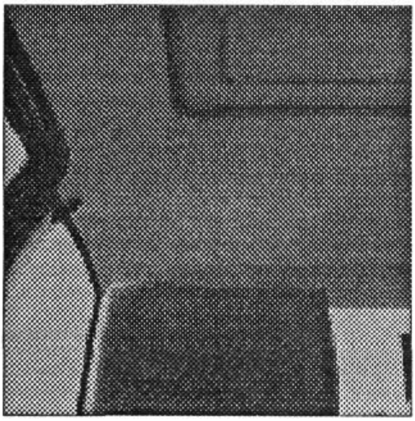

Figure 1: *Typical static image.*

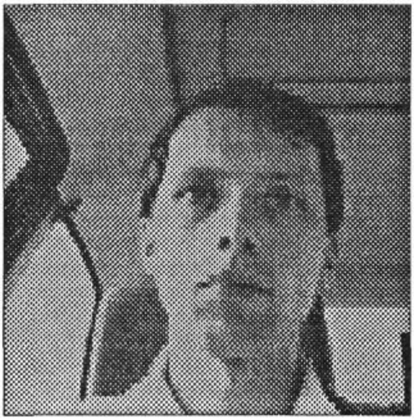

Figure 2: *Typical dynamic image.*

One way to reduce the noise would be to use one of the many region operators, such as Sobel or Prewitt, or simply to blur the image over a small region. Instead we choose to reduce the resolution by pixel averaging. We refer to the reduced resolution images as *meta-images*. Given an image m, we write M_{ij} for the 4×4 image $\{m_{4i+k,4j+l} : 0 \leq k, l \leq 3\}$ and then call $M = \{\bar{M}_{ij}\}$ the meta-image given by M. Thus M_{ij} is the 4×4 image and $\bar{M}_{ij}$ the corresponding pixel of M; by abuse of language, we refer to both M_{ij} and $\bar{M}_{ij}$ as a meta-pixel.

This provides a number of additional benefits over a simple region operator;

- it reduces the size of the image we are working with which helps maintain performance. At this stage we are simply aiming to locate the position of the eyes, and provided the head occupies a reasonable proportion of the image, 32 pixels square is sufficient for this [1];

- we calculate in addition to the mean, the variance for each M_{ij}. This is largely independent of lighting levels, reflecting more the nature of the meta-pixel region; this reflects whether the meta-pixel is a homogeneous region (such as roof lining or seat) or quite variable, such as a window border.

Additionally, we perform an operation similar to the Moravec operator [11] for each meta-pixel to obtain both a direction (one of eight) and *strength* rating, M^θ and M^ι. This gives us a primitive *structure* description of the inside of the car which varies little with lighting.

In order to determine the location of the head, frame differencing is performed on the static meta-image S and current meta-image C using the variance and direction information. Due to lighting changes and the variability of the windows, this is biased with the location information obtained from processing the previous frame. Each meta-pixel is given a score according to the following conditions, each scoring one point if true;

- $|S^\sigma_{ij} - C^\sigma_{ij}| > \mathcal{T}_{dev}$ where $\mathcal{T}_{dev}$ is the deviation threshold,

- $|\bar{S}_{ij} - \bar{C}_{ij}| > \mathcal{T}_{mean}$ where $\mathcal{T}_{mean}$ is the mean threshold,

- $S^\theta_{ij} \neq C^\theta_{ij}$ and either $S^\iota > \mathcal{T}_{resp}$ or $C^\iota > \mathcal{T}_{resp}$ where $\mathcal{T}_{resp}$ is the edge response threshold,

- a bonus point if P_{ij} was a potential face pixel.

The thresholds $\mathcal{T}_{dev}$ and $\mathcal{T}_{mean}$ depend on whether the meta-pixel P_{ij} was determined to be part of the face or not, and were determined empirically. Meta-pixels scoring two or more points are considered potential head pixels, a typical meta-image with meta-pixels scoring two or more marked is shown in figure 3. *Salt and pepper* noise is reduced by a dilation/erosion stage yielding an image such as figure 4.

The next stage attempts to match a head outline to the region marked as potential head. There have been a number of model matching systems described that would be capable of locating an head outline in out image (e.g. Waite and Welsh [18], Taylor and Cooper [15] or Davis and Taylor [5]). In each case, the desired shape can be deformed within statistically determined limits. We adopt the methods used by FindFace, and specifically, the method described by Bennett and Craw [2]. The end result of this system is the approximate location of the head at low resolution in a representation compatible with the quality assurance system discussed below. Bennett and Craw use edge strength to guide the outline, as their system must contend with a lot of extraneous edges. We have a very clear, albeit irregular, outline to match, which allows for a number of simplifications to their method.

Specifically, the outline is defined by 10 points, rather than the 20 originally used. A continuous outline is generated by connecting the points with straight lines. To overcome the lack of gradient information used to guide the original towards the correct edge (the original uses a large region operator to allow approximate matches to be drawn towards the correct position) each of the 10 model points records its orientation. The intervening points that are created

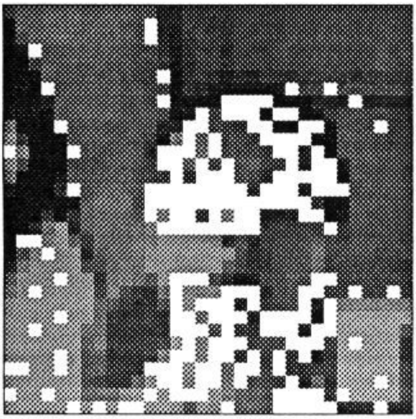

Figure 3: *Meta-image before noise reduction. Meta-pixels scoring more two or more are indicated.*

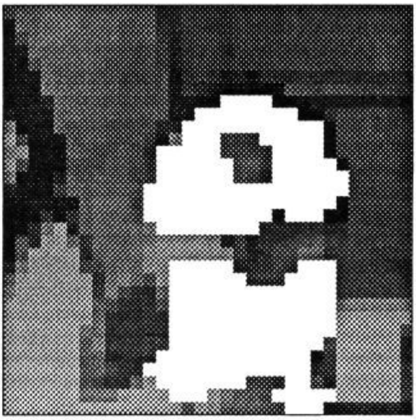

Figure 4: *Figure 3 after dilation/erosion to remove salt and pepper noise.*

derive their orientation from the two end points of the connecting line. Each edge point then knows, given that it is either on the face or on the background area, in which direction it must move to locate the edge. This approach does not allow for the subtlety of movement of the original, but performs satisfactorily at the resolution being used. The model is only allowed to deform in a symmetrical way by independent scaling; the refinment obtained from the second stage of the original algorithm would be wasted at these low resolution.

Having positioned the head outline model on the meta-image, the meta-pixels within the model area are counted; at least 80% must be correctly classified for the system to believe the presence of a head.

4.2 Feature Tracking System - Stage II

Having identified the head location, part of the statistical model from FindFace can be used to estimate the position of the eyes. The face model used, and the methods used to determine face location, do not allow for lateral rotation of the face, and allow for only a small vertical rotation. This is considered acceptable, as frequent large movements, as observed during driving in heavy traffic for example, would be detected by the QA stage. Furthermore, this degree of movement can generally be taken as an indication that the driver is awake, so the degradation of the system under these circumstances is not critical.

With a predicted location of the eyes, the full resolution (128x128) image can now be used, in particular the eye regions can be examined. The most appealing way of obtaining the required eye measurements is to use one of the eye recognition systems already mentioned [12, 19, 6, 4], thus obtaining a *direct* eye lid separation reading from which to deduce blinks. Two reasons preclude the use of these systems at present; the poor contrast of the image which leads to unreliable results, and the algorithms speed.

A more naive approach is used which simply calculates the grey level histogram for the eye region. When the eye is open, a characteristic bimodal histogram is obtained. When the eyelid is shut, this reduces to a single peak. The profiles differ sufficiently between open and closed eyes (see figures 5 and 6) to detect blinks. The overall profile change is largely independent of lighting levels, the image noise we experience, and slight inaccuracies in location of the eye region. Although the information obtained by no means as detailed as that from the deformable template model, the desired results are obtained. Figure 7 shows figure 2 with the eye regions and outline marked.

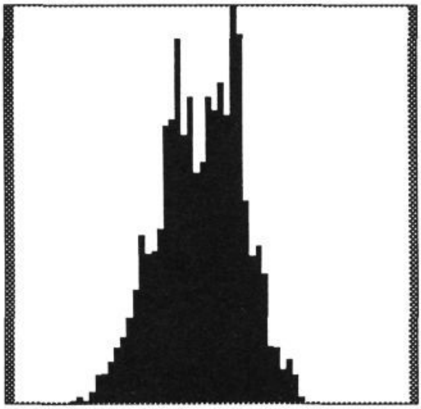

Figure 5: *Histogram of region containing open eye.*

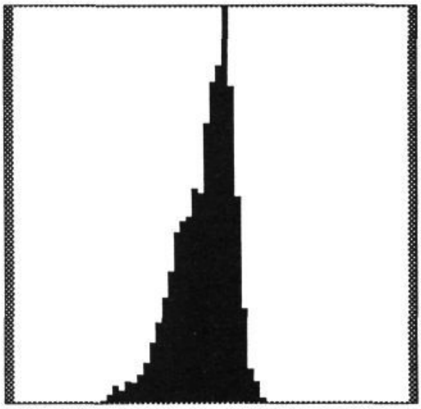

Figure 6: *Histogram of region containing closed eye.*

An alternate method currently being investigated involves template matching with the a template extracted from one of the images with a specific driver in it. Using the QA system described below, the eye regions can be reliably extracted, and simple correlation performed. Although this does not prove satisfactory if a generic template is used, correlation with the drivers own eye region should be much more reliable. Whether this proves more reliable than the statistical method will be determined following testing.

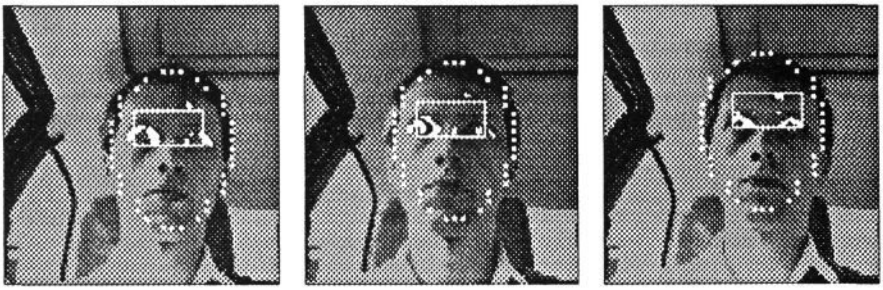

Figure 7: *Series of three processed images showing located head outline and estimated eye region from which the eye histograms are calculated.*

4.3 Quality Assurance System

The quality assurance system is essentially a slimmed down version of FindFace, described in [17], working at a somewhat faster rate (how fast?). The input image sequence is subsampled and processed by the modified FindFace system. As this takes quite a few seconds, it is not possible to use this system directly for obtaining eye measurements. The results of the processed image can be used for a number of purposes.

The location of the eyes as determined by FindFace can be checked with the location obtained by the tracking system for that particular image. This ensures that the tracking system has not been distracted and is still measuring the right region. As the results from FindFace can not be used to initialise the tracking due to the delay between image capture and the availability of results, this is more useful as a measure of reliability rather than as part of a feedback loop.

Feedback can be obtained by using the FindFace results to customise the head model. The model used by the tracking system is a modified version of that used by FindFace; specifically, it contains only the outline and eye locations, and the outline points are used to produce a complete, low resolution, outline, rather than specific points. As mentioned above, sideways rotations of the head are not allowed for, but the tracking model can be adjusted to provide better eye locations by generating a dynamic model from the outline and eye positions obtained from FindFace.

Also as mentioned above, the alternate eye measuring technique relies on the position of the eyes as produced by FindFace to extract the eye template.

5 In Operation

The complete system is written in C, and runs on a Sun IPC workstation. The original system used our elderly Imaging Technology FG100-V frame grabber which required the use of a bus protocol converter (we used a BIT-3 system). This was fed a video signal from a Pulnix TMC-516 remote head camera mounted on the dashboard of a Ford Granada, which housed the equipment in the boot.

In this configuration, and without the QA system running in parallel, approximately six frames per second could be processed, with the head and eye locations displayed on a monitor for confirmation. Ultimately, the monitor output will be sacrificed to gain additional speed.

As new equipment has become available, we have moved to an SBus based system using a DataCell S2200 frame grabber with TMC-6 camera. This is a colour system and offers a number of new ideas to be tried. No attempt has been made yet to optimise the code originally written for FG100-V, the result of which is only a marginal improvement in performance at present. A significant performance increase is expected when the S2200 is treated as a memory mapped device rather than a serial data stream. Early results with the new hardware do however suggest that we no longer have the problems with high levels of noise in the images.

6 Future Developments

Although the QA system has been proven, this still needs further work before being fully integrated into the system; in particular, the conversion of the relevant parts from POP11 into C will be needed to obtain the required performance. The FindFace system is designed to process single images of different individuals. Although some scope is included to adapt dynamically to changing classes of faces, this characteristic must be more fully explored if many images of the same person are to be processed. There are a number of ways the efficiency of the QA system could be improved by using such adaptability.

Although the use of colour must be explored carefully, as moving to NIR or UV illumination would almost certainly render such systems useless, a number of ideas will be considered. The eyes are often of a colour significantly different to the rest of the face. Looking for the change in colour in the eye region may provide blink indication. Cars tend not to have *flesh* coloured interiors, so this could be used to aid location/tracking of the driver.

With better quality data, we will investigate further the different methods of sequence analysis and feature tracking mentioned earlier. Hopefully this would allow a more efficient initial location leading to both a more accurate location of, and more frequent measure of the eye. The ultimate development of this would be to measure the eye lid separation rapidly enough to generate an analogue output. This may allow trends in driver alertness to be determined at an earlier point in time.

References

[1] Talis Bachmann. Identification of spatially quantised tachistoscopic images of faces: How many pixels does it take to carry identity? *European Journal of Cognitive Psychology*, 3(1):87–103, 1991.

[2] Alan Bennett and Ian Craw. Finding image features using deformable templates and detailed prior statistical knowledge. In Peter Mowforth, editor, *British Machine Vision Conference 1991*, pages 233–239, 1991.

[3] D A Castelow and A J Rérolle. A monocular ground plane estimation system. In *Proceedings of the British Machine Vision Conference*, pages 392–395, 1991.

[4] Ian Craw, David Tock, and Alan Bennett. Finding face features. In *Proceedings of the Second Eurpoean Conference on Computer Vision*, 1992. To be published.

[5] D N Davis and C J Taylor. An intelligent segmentation system for lateral skull x-ray images. In *Proceedings of the British Machine Vision Conference*, pages 251–255, 1989.

[6] Peter W Hallinan. Recognizing human eyes. *SPIE - Geometric Methods in Computer Vision*, 1991.

[7] Chris Harris and Carl Stennett. Rapid - a video rate object tracker. In *Proceedings of the British Machine Vision Conference*, pages 73–77, 1990.

[8] Jim Horne. Stay awake, stay alive. *New Scientist*, pages 20–24, 1992. 4th January.

[9] T S Huang, editor. *Image Sequence Analysis*. Springer Series in Information Sciences. Springer-Verlag, 1981.

[10] Merrill Mitler. Catastrophes, sleep and public policy. *Sleep*, 11, 1988.

[11] Hans P Moravec. Towards automatic visual obstacle avoidance. In *5th International Joint Conference on Artificial Intelligence*, 1977.

[12] Mark Nixon. Eye spacing measurement for facial recognition. *Proceedings of SPIE*, August 1985.

[13] A Rosenfeld. *Motion: Analysis of Time-varying Imagery*, pages 173–183. Cambridge University Press, 1983.

[14] T N Tan, G D Sullivan, and K D Baker. Structure from constrained motion using point correspondences. In *Proceedings of the British Machine Vision Conference*, pages 301–309, 1991.

[15] C J Taylor and D H Cooper. Shape verification using belief updating. In *Proceedings of the British Machine Vision Conference*, pages 61–66, 1990.

[16] Graham Thomas. *Image Processing*, chapter 3, pages 40–57. McGraw-Hill, 1991.

[17] David Tock, Ian Craw, and Roly Lishman. A knowledge based system for measuring faces. In *Proceedings of the British Machine Vision Conference*, pages 401–406, 1990.

[18] J B Waite and W J Welsh. An application of active contour models to head boundary location. In *Proceedings of the British Machine Vision Conference*, pages 407–412, 1990.

[19] Alan Yuille, David Cohen, and Peter Hallinan. Feature extraction from faces using deformable templates. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1989.