Moving target classification and tracking from real-time video

Alan J. Lipton Hironobu Fujiyoshi Raju S. Patil

The Robotics Institute. Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA, 15213
email: {ajl|hironobu|raju}@cs.cmu.edu
URL: http://www.cs.cmu.edu/˜ vsam

Abstract

This paper describes an end-to-end method for extracting moving targets from a real-time video stream, classifying them into predefined categories according to image-based properties, and then robustly tracking them. Moving targets are detected using the pixelwise difference between consecutive image frames. A classificatoin metric is applied these targets with a temporal consistency constraint to classify them into three categories: human, vehicle or background clutter. Once classified, targets are tracked by a combination of temporal differencing and template matching.

The resulting system robustly identifies targets of interest, rejects background clutter, and continually tracks over large distances and periods of time despite occlusions, appearance changes and cessation of target motion.

1 Introduction

The increasing availability of video sensors and high performance video processing hardware opens up exciting possibilities for tackling many video understanding problems [1]. It is important to develop robust real-time video understanding techniques which can process the large amounts of data attainable. Central to many video understanding problems are the themes of target classification and tracking.

Historically, target classification has been performed on single images or static imagery [12, 13, 6]. More recently, however, video streams have been exploited for target detection [8, 14, 15]. Many methods like these, are computationally expensive and are innaplicable to real-time applications, or require specialised hardware to operate in the real-time domain. However, methods such as *Pfinder* [14], W^4 [8] and Beymer *et al* [10] are designed to extract targets in real-time.

The philosophy behind these techniques is the segmentation of an image, or video stream, into *object* vs. *non-object* regions. This is based on matching regions of interest to reasonably detailed target models. Another requirement of these systems is, in general, to have a reasonably large number of *pixels on target*. For both of these reasons, these methods would, by themselves, be inadequate in a general outdoor surveillance system, as there are many different types of target which could be important, and it is often not possible to obtain a large number of pixels on target. A better approach is one in which classification is based on simple rules which are largely independent of appearance or 3D models. Consequently, the classification metric which is explored in this paper, is based purely on a target's *shape*, and not on its image content.

Furthermore, the temporal component of video allows a temporal consistency constraint [7] to be used in the classification approach. Multiple hypotheses of a target's classification can be maintained over time until the system is confident that it can accurately classify the target. This allows the system to disambiguate targets in the case of occlusions or background clutter.

Many systems for target tracking are based on Kalman filters but as pointed out by [5], they are of limited use because they are based on unimodal Gaussian densities and hence cannot support simultaneous alternative motion hypotheses. Isard and Blake [5] present a new stochastic algorithm for robust tracking which is superior to previous Kalman filter based approaches. Bregler [11] presents a probabilistic decomposition of human dynamics to learn and *recognize* human beings (or their gaits) in video sequences.

This paper presents a much simpler method based on a combination of temporal differencing and image template matching which achieves highly satisfactory tracking performance in the presence of clutter and enables good classification. Hence the use of Kalman filtering or other probabilistic approaches is avoided. Future work involves using temporal filtering and building on some of the ideas presented in [5] and [11] to achieve target *recognition* and *multiple target tracking*.

Two of the basic methods for target tracking in real-time video applications are temporal differencing (DT) [2] and template correlation matching. In the former approach, video frames separated by a constant time δt are compared to find regions which have changed. In the latter approach each video image is scanned for the region which best correlates to an image template. Independently, these methods have significant shortcomings.

DT tracking is impossible if there is significant camera motion, unless an appropriate image stabilisation algorithm is employed [4]. It also fails if the target becomes occluded or ceases its motion. Template correlation matching generally requires that the target object's appearance remains constant. The method is generally not robust to changes in object size, orientation or even changing lighting conditions.

However, the tracking properties of these two methods are complementary. When the target is stationary, template matching is at its most robust while DT will fail. And when the target is in motion, DT will be successful where template matching will tend to "drift".

This is the motivation for combining the two methods. The idea is to use DT to detect moving targets and train the template matching algorithm. These targets are then tracked using template matching guided by the DT stage. This combination, obviates the need for any predictive filtering in the tracking process as the tracking is guided by motion detection. This simple paradigm produces remarkably robust results.

This paper describes a system for robustly tracking targets in a video stream and classifying the targets into "humans" and "vehicles" for an outdoor video surveillance application. Target tracking is based on two main principles; (a) *temporal consistency* which provides a robust way of classifying moving targets while rejecting background clutter, and (b) *the combination of motion detection with image-based template matching* which provides a highly robust target tracking scheme. Target classification is based on a simple application of maximum likelihood estimation after computing a simple shape based metric for each target.

1.1 System Overview

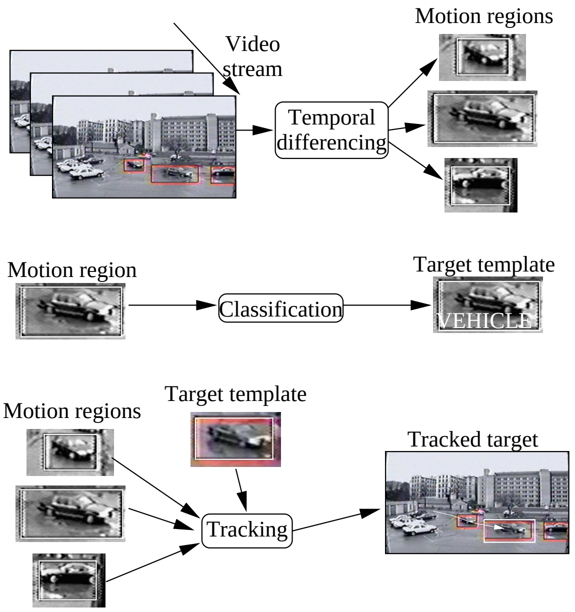

Figure 1: *Overview of the identification and tracking system. Moving objects are detected in a video stream using temporal differencing. Targets are then classified according to a classification metric. These targets can be tracked using a combination of motion information and image based correlation*

The system proposed in this paper consists of three stages as outlined in figure 1. In the first stage, all moving objects are detected using a temporal differencing algorithm. These are described as motion regions. Each one is classified at each time frame using an image-based classification metric. Classifications for each individual motion region are recorded over a period of time, and a simple Maximum Likelihood Estimation (MLE) criterion is used to correctly classify each target. Once a motion region has been classified, it can be used as a training template for the tracking process. The tracking process involves correlation matching between a template and the current video image which is used to adaptively update the template. However, this matching process is guided by the DT stage to increase the likelihood that a target will be correctly and robustly tracked.

2 Temporal differencing

There are many variants on the DT method, but the simplest is to take consecutive video frames and determine the absolute difference. A threshold function is then used to determine change. If I_n is the intensity of the n^{th} frame, then the pixelwise difference function Δ_n is

$$\Delta_n = |I_n - I_{n-1}|$$

and a motion image M_n can be extracted by thresholding

$$M_n(u,v) = \begin{cases} I_n(u,v) & , \quad \Delta_n(u,v) \geq T \\ 0 & , \quad \Delta_n(u,v) < T \end{cases}$$

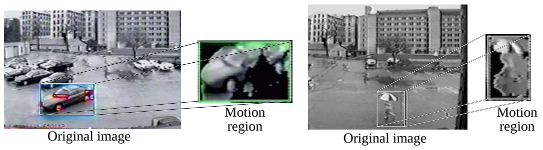

Figure 2: *Motion regions. Notice that much of the background information is not incorporated into the template.*

After the motion image is determined, moving sections are clustered into motion regions $R_n(i)$. This is done using a connected component criterion. Figure 2 shows the result of extracting motion regions.

3 Target Classification

There are two key elements to classifying targets; some identification metric operator $ID(x)$ which is used for distinguishing between types of targets (in this case, a very simple image-based metric is employed), and the notion of temporal consistency. If a target persists over time, it is a good candidate for classification. If not, it is considered to be background clutter. At each instant, it is classified according to $ID(x)$. These classifications are collected until a statistical decision can be made about the classification of the target. A version of MLE is used to make the classification decision.

3.1 Temporal consistency

The main difficulty with classification is that in any single frame, the instance of a particular motion region may not be representative of it's true character. For example, a partly occluded vehicle may look like a human, or some background clutter may briefly appear as a vehicle. To overcome this problem, a multiple hypothesis approach is used.

The first step in this process is to record all N_n potential targets $P_n(i) = R_n(i)$ from some initial frame. These regions are classified according to the classification metric operator $ID(x)$ (see section 3.2) and the result is recorded as a classification hypothesis $\chi(i)$ for each one.

$$\chi(i) = \{ID(P_n(i))\}$$

Each one of these potential targets must be observed in subsequent frames to determine whether they persist or not, and to continue classifying them. So for new frames, each previous motion region $P_{n-1}(i)$ is matched to the spatially closest current motion region $R_n(j)$ according to a mutual proximity rule. After this process, any previous potential targets P_{n-1} which have not been matched to current regions are considered transient and removed from the list, and any current motion regions R_n which have not been matched are considered new potential targets. At each frame, their new classifications (according to the metric operator) are used to update the classification hypothesis.

$$\chi(i) = \{\chi(i)\} \cup \{ID(P_n(i))\}$$

In this way, the statistics of a particular potential target can be built up over a period of time until a decision can be made about its correct classification. Furthermore, transient motion regions such as trees blowing in the wind will be thrown away.

3.2 Classification metric

Figure 3: *Typical dispersedness values for a human and a vehicle.*

To classify targets in real surveillance applications it is important to use a classification metric which is computationally inexpensive, reasonably effective for small numbers of *pixels on target*, and invariant to lighting conditions or viewpoint. It is clear that the most obvious types of targets which will be of interest are humans and vehicles [8, 9]. For this reason, a classifier to detect these two groups has been implemented. The metric is based on the knowledge that humans are, in general, smaller than vehicles, and that they have more complex shapes.

A bi-variate approach is employed, with the target's total area on one axis, and its dispersedness on the other. Dispersedness is based on simple target shape parameters and is given by

$$Dispersedness = \frac{Perimeter^2}{Area}$$

Clearly, a human, with its more complex shape, will have larger dispersedness than a vehicle - see figure 3. Figure 4 shows the distribution of a training sample

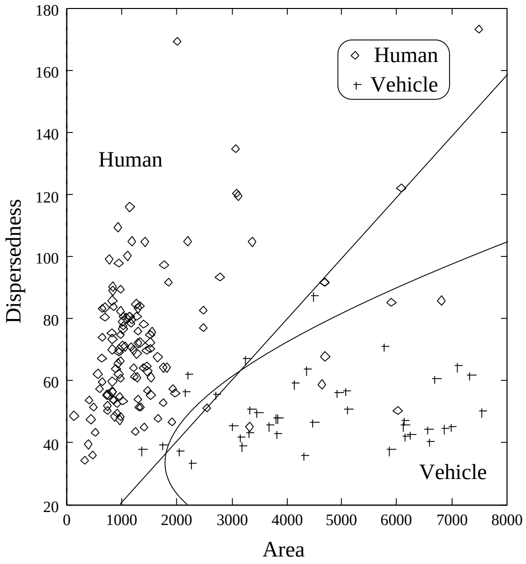

Figure 4: *Bi-variate classification data for training sample of over 400 images. Both linear and Mahalanobis clustering are shown.*

of over 400 targets. Also, shown is a linear segmentation and a Mahalanobis distance-based segmentation which provides a superior segmentation for classification purposes.

3.3 Target classification

In this implementation, a simple application of MLE is employed to classify targets. A classification histogram is computed for each motion region at each time and if the target persists for time t_{class}, the peak of the histogram is used to classify the target. Furthermore, at every time instant after t_{class}, the object can be reclassified.

One advantage of this method is that if an object is temporarily occluded, it will not adversely affect the ultimate classification. Figure 5 shows a situation in which an object is originally misclassified because of partial occlusion, but with the passage of time, the classification statistics correctly reclassify it.

A further advantage of this method is that it is robust to background clutter such as leaves blowing in the wind. These effects appear as very transient and unstable motion. It is unlikely that this motion will be present long enough to be classified at all. If it does persist, it is unlikely to be consistently misclassified for a long period of time.

4 Tracking

Classified motion regions are then used as training templates for the tracker. Tracking consists of a combination of appearance-based correlation matching and motion detection.

Motion regions can be used to guide correlation processing and template updating. This combination makes the tracker robust to changes of target appearance, occlusion, and cessation of target motion. The

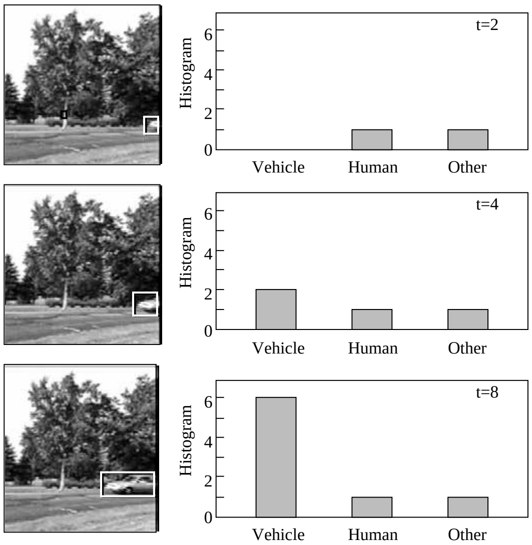

Figure 5: *Process of classification. Only after several frames can this object be correctly identified.*

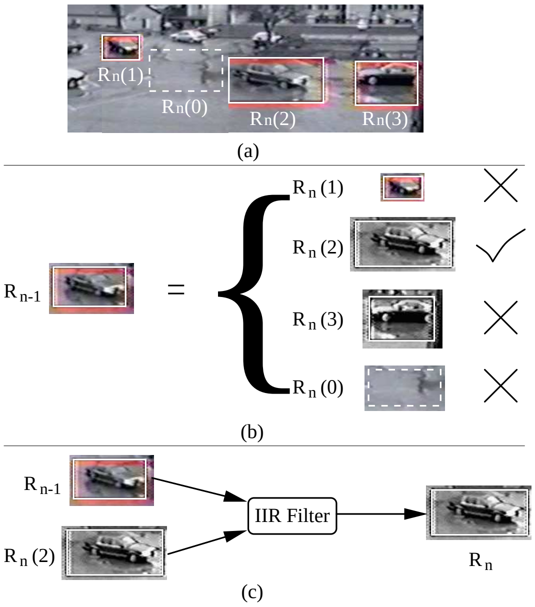

Figure 6: *The tracking process. (a) There are four target candidates – three moving targets and the previous template position. (b) The current template is compared to each of the candidates. (c) The current template is updated using an IIR filter.*

procedure is outlined in figure 6. Candidate motion regions $R_n(i)$ are selected and each of these are correlated with the current template R_{n-1} to find a best match. This will not be sufficient, however, if the target has ceased its motion, so an extra region, called $R_n(0)$, is also compared. $R_n(0)$ is made up of the pixels in I_n which correspond to the location of R_{n-1}. That is, it is the part of the image in which the target *used* to be located. Once the best correlation has been found from all of these candidates, it is merged with R_{n-1} through an infinite impulse response (IIR) filter (see section 4.0.1) to produce R_n. This is done so that the appearance of the template continues to match the appearance of the target.

Using the motion image to guide the template matching algorithm carries with it some distinct advantages over conventional techniques. Correlation matching is the most computationally expensive part of the tracking algorithm; if the correlation matching need only be performed where moving targets are detected, computation time can be reduced. Also, if correlation matching is biased towards areas where motion is detected, it is more likely to retain the target and not "drift" on to the background. Furthermore, if updating the content of the template is combined with the motion function then templates can be constructed which only contain "active" pixels and do not contain background information.

4.0.1 Updating templates

In this implementation, adaptive template updating is used to ensure that the current template accurately represents the new image of the object. So the new template R_n is generated by merging the previous instance R_{n-1} with current information from M_n and I_n using an infinite impulse response filter of the form

$$R_n = \alpha M_n + (1-\alpha) R_{n-1}$$

The effect of the IIR filter is shown in figure 7.

Figure 7: *The IIR filter. As the image changes from a face in profile to a frontal view, the template is updated using an IIR. If the image is stable for some time, the template also remains stable.*

One problem with DT motion detection is that it tends to include undesirable background regions on the periphery of the target where the object has "just been". Large amounts of this background information in the template is one of the causes of template "drift". One way to alleviate this problem is to use knowledge of the target's motion to crop these background regions from the template.

The 2D image velocity vector of the target $(\dot{u}, \dot{v})$(pixels/frame) can be approximately determined by calculating the difference between the centroid of the previous template R_{n-1} and the centroid of the new template R_n. It can be assumed that the region trailing the template is background material exposed by the passage of the target. This information can be cropped from R_n so that it contains mostly target pixels (see figure 8).

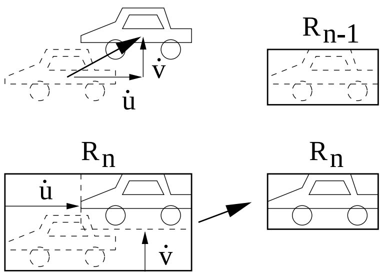

Figure 8: *Motion cropping. As the target moves, background information is included in the new template. Knowledge of the target's motion is used for cropping.*

5 Results

The system has been implemented on a Pentium 200Mhz system under Microsoft Windows 95 with a Matrox Meteor digitiser. The system can detect, classify and track targets at 14 frames/second over a 320×240 pixel image. The system has been applied to large amounts of live video in unstructured environments in which human and vehicular activity is present. Over six hundred instances of vehicles and humans have been identified and target tracking has been performed over the life span of over two hundred targets.

5.1 Classification

Figure 9 shows some examples of target classification. For single targets, this algorithm provides a robust classification. Note that trees blowing in the wind are correctly rejected as background clutter. Furthermore, accurate classification is largely independent of target size, speed or viewing aspect. However, when multiple human targets are close together, they can be misclassified as a vehicle. There are two reasons for this; the clustering of the motion regions is too liberal in this case, erroneously joining multiple regions, and the classification metric is not sophisticated enough to deal with this situation. Another limitation is that targets which are very small ($< 5 \times 5$ pixels) tend to be temporally inconsistent and hence rejected.

Table 1 shows the results of the classification algorithm applied to over four hours of live video in an unstructured environment. The main problem with vehicle recognition is that when vehicles are partially occluded for long times, they are sometimes rejected. Humans are much smaller than vehicles and are often not recognised as temporally stable objects. Also, humans tend to move in close groups that can be misclassified as vehicles according to the simple metric.

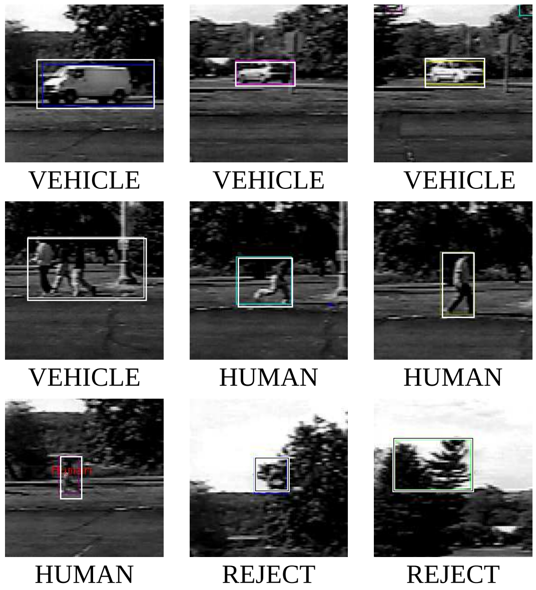

Figure 9: *Example of target classification. Notice that groups of people can be misclassified as a vehicle.*

Target	Tot.	Unclass.	Misclass.	Correct
Vehicle	319	10.7%	2.5%	**86.8%**
Human	291	11.0%	6.2%	**82.8%**
False	4			

Table 1: *Classification results from live video in unstructured environments.*

5.2 Tracking

In figure 10 a vehicle is tracked as it drives around a test site. In figures 10(b)-(c) other visibly similar targets are present, but the template tracking does not stray because it is guided by the motion regions. Even when the target is partially occluded by a similar target, the tracker remains stable. The target can be tracked over long distances and periods of time (≈ 2 mins. - the life span of the target), even as it becomes small. In figure 10(d) it is only 4×9 pixels.

In figure 11 two humans are detected and one of them is tracked. Over the life span of these targets (≈ 4 mins.), the tracker does not confuse them, even when one occludes the other, because the template matching algorithm "prefers" the correct target over a false one.

6 Conclusions

The two key elements which make this system robust are the classification system based on temporal consistency and the tracking system based on a combination of temporal differencing and correlation matching. The system effectively combines simple domain knowledge about object classes with time-domain statistical measures to classify target objects. Target models are simple and based purely on target shape so they are applicable to a large number of real-world

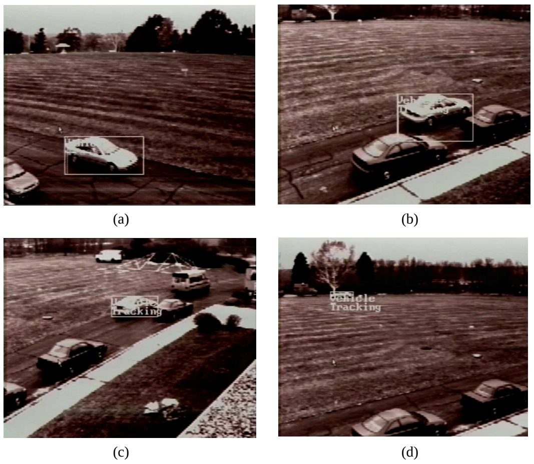

Figure 10: *Identification and tracking of a vehicle. A vehicle is classified and tracked as it drives for about 2 mins.*

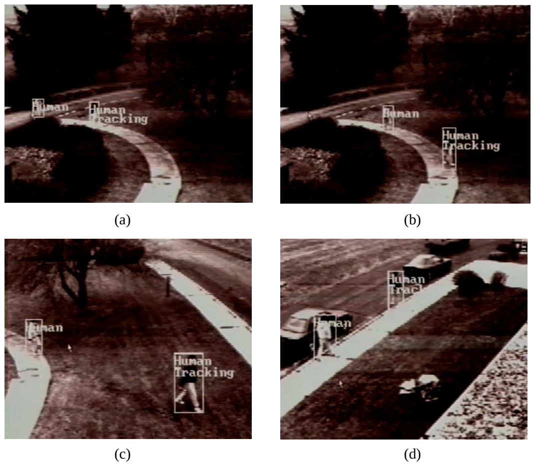

Figure 11: *Identification and tracking of human targets. Two humans are correctly classified and one of them is tracked for about 3 mins.*

video applications. Using a combination of domain knowledge and temporal consistency, targets are robustly identified in spite of partial occlusions and ambiguous poses, and background clutter is effectively rejected.

Using temporal differencing to guide vision-based correlation matching has three main advantages; it allows continuous tracking despite occlusions and cessation of target motion, it prevents templates "drifting" onto background texture, and it provides robust tracking without the requirement of having a predictive temporal filter such as a Kalman filter.

References

[1] T. Kanade, R. Collins, A. Lipton, P. Anandan, P. Burt "Cooperative Multisensor Video Surveillance" *Proceedings of DARPA Image Understanding Workshop 1997*, Vol. I, pp. 3-10, 1997.

[2] C. Anderson, P. Burt, G. van der Wal "Change detection and tracking using pyramid transformation techniques" *SPIE - Intelligent Robots and Computer Vision* Vol. 579, pp. 72-78, 1985

[3] A. Lipton, H. Fujiyoshi, R. Patil "Moving target detection and classification from real-time video" *submitted to IEEE WACV 98*, 1998.

[4] M. Hansen, P. Anandan, K. Dana, G. van der Wal, P. Burt "Real-time scene stabilization and mosaic construction" *Proceedings of DARPA Image Understanding Workshop 1994*, 1994

[5] M. Isard and A. Blake "Contour tracking by stochastic propagation of conditional density" *Proceedings of European Conf. on Computer Vision 96*, pp. 343-356, 1996

[6] K. Ikeuchi, T. Shakunaga, M. Wheeler, T. Yamazaki "Invariant histograms and deformable template matching for SAR target recognition" *Proceedings of IEEE CVPR 96*, pp. 100-105, 1996

[7] J. Davis, A. Bobick "The representation and recognition of human movement using temporal templates" *Proceedings of IEEE CVPR 97*, pp. 928 - 934, 1997

[8] I. Haritaoglu, D. Harwood, L. S. Davis "W^4 Who? When? Where? What? A Real Time System for Detecing and Tracking People" *FGR98 submitted*, 1998

[9] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio "Pedestrian detection using wavelet templates" *Proceedings of IEEE CVPR 97*, pp. 193-199, 1997

[10] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik "A real-time computer vision system for measuring traffic parameters" *Proceedings of IEEE CVPR 97*, pp. 495-501, 1997

[11] C. Bregler "Learning and recognizing human dynamics in video sequences" *Proceedings of IEEE CVPR 97*, pp. 568-574, 1997

[12] R. kasturi and R. C. Jain "Computer Vision; Principles" *IEEE Computer Society Press*, 1991

[13] M. Turk, A. Pentland "Eigenfaces for recognition." *Journal of Cognitive Neuroscience* , 3, 1, 71-86, 1991.

[14] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland "Pfinder: Real-Time Tracking of the Human Body" *IEEE Transactions on Pattern Analysis and Machine Intelligence* July 1997, vol 19, no 7, pp. 780-785

[15] D. Koller, K. Danilidis, H.-H. Nagel "Model-Based Object Tracking in Monocular Image Sequences of Road Traffic Scenes" *International Journal of Computer Vision*, 10-3, pp. 257-281, 1993