Classification of Moving Targets Based on Motion and Appearance

Y.Bogomolov[(1)], G.Dror[(2)], S.Lapchev[(1)], E.Rivlin[(1)], M.Rudzsky[(1)]
[(1)]Computer Science Department, Technion -
Israel Institute of Technology 32 000 Haifa, Israel
{rudzsky}@cs.technion.ac.il
[(2)]The Academic College of Tel-Aviv-Yaffo, 4 Antokolski st,
Tel-Aviv, Israel

Abstract

We describe a system for detection and classification of moving targets. The system includes change detection and tracking modules, which are based on adaptive background, updated with information from target level processing. The classification module performs hybrid classification which combines motion and appearance features. The system is able to perform real time detection, tracking and classification of different types of targets in natural, real life setting. Experiments demonstrate that the proposed hybrid architecture of classifiers improves classification significantly.

1 Introduction

In this paper we present a surveillance system that is able to detect, track and classify moving targets. Our system extracts static and dynamic characteristic features of moving targets and uses them to assign the targets to one of several predefined categories. The system requires minimal user input, and is able to work under diverse illumination conditions, including noisy background, while using various types of video sensors. The system provides real-time video performance due to short detection, lock-on and classification time for each target. We can logically divide the system into three major parts: the target detection module, the target tracking module and the classification module. These modules were designed to support classification of the following categories: *vehicle, animal, human, group of people, crawling man* and others. The system supports classification of activities and complex motions (human with a carriage, human on a bicycle). To provide reliable, view independent, classification we combine features based on motion characteristics with features based on target's shape. By so doing we create a hybrid system that uses shape and motion for classification. Reliable classification strongly depends on the quality of objects segmentation, so significant effort was put into improving targets segmentation and tracking. In subsequent sections we describe individual modules of the system. In Section 2 we briefly describe the target detection and target tracking modules. Detailed description of these modules can be found in [14]. In Section 3 we give a de-

scription of the target classification process. In Section 4 we present our experimental results. Section 5 contains conclusions and discussion of perspectives for future work.

2 Target Detection and Tracking

Background subtraction and temporal differencing of consecutive frames are popular methods for target detection in object tracking [8], intruder detection [6], traffic monitoring [11] and others. While temporal differencing is adaptive to changes in the environment, it does not detect the entire object. On the other hand, background subtraction can provide more reliable information about moving objects, but it requires more complex processing for online adaptation of the background to changes in environmental conditions, mainly for illumination. It may also lead to "holes" when stationary objects attributed to the background start to move. Therefore, in some works [5] a combination of both approaches is applied.

Global thresholding is the simplest method for change detection; it can be improved by local thresholding, particularly when the scene illumination varies locally over time. A review on background subtraction in video surveillance systems can be found in [13].

Creation of an initial background model is an important not entirely solved problem. The general assumption that the background can be extracted by using a scene without moving objects is not always valid for outdoor sequences. For background initialization we use the following assumptions: each pixel in the image will reveal the background for at least a short interval in the sequence; the background is approximately stationary; only small background motion may occur; and a short processing delay is allowed subsequent to acquiring the training sequence. In our algorithm the background initialization is done in the first $1-3$ seconds of the processing, when the background model is learned by the system. We initialize the background image by the first frame and create a binary mask by thresholding the difference between the two consecutive frames. Then the background image and the binary mask are updated using information obtained by thresholding the difference between every two consecutive frames. An adaptive update of the background is a desirable feature to have due to changes in the background that are caused by moving objects, or by illumination fluctuations (stochastic motion of plants, clouds and so on). In our system, the assignment of specific pixel to the foreground or to the background is made using information about the assignment of its neighbors to the background or to the foreground. This method shows sufficient improvement in moving objects segmentation in comparison with standard methods (see also [9]). Consequently our implementation of background adaptation algorithm is deeply embedded in our motion and target detection algorithms.

For target detection we perform a number of standard operations: down sampling of the image before processing; background subtraction; connected component analysis, and morphological filtering.

The process of threshold adaptation sets the thresholds' value in order to maximize the detection's SNR. At this processing stage we need only blob-like silhouette of the target. Only blobs whose area exceeds a threshold are taken for further processing. The information about a blob, whose size falls below threshold, does not enter into the target detector data base (DB) and their pixels are considered as part of the background.

Two of the most important variables in the part of the algorithm described above, are

the background subtraction threshold for each pixel and the minimal blob's area (received from the connected component analysis of an image resulting from background subtraction) that permits further processing. We calculate local SNR as a ratio of the total area

Figure 1: Target detection: top row - the original frames, targets are within their bounding boxes; bottom row - extracted and scaled targets from the first row

in pixels of all detected blobs (those that pass area filtering) to the sum of total number of pixels "turned on" by background subtraction algorithm. Background subtraction has several serious problems, such as sensitivity to changes in illumination, or handling moving background objects (MBO), which are objects that were initially associated with the background and started to move while leaving a constant difference silhouette on a subtraction image and others. To overcome these problems we use additional algorithms. To deal with sensitivity to smooth background changes due to changes in illumination we use dynamic background modelling and updating which is a simple IIR filter. To deal with MBO, we developed an algorithm based on localized temporal subtraction. We find the motion of a blob, produced by MBO, as soon as it appears by the background subtraction algorithm and store the blob's initial position in a DB. While processing each new frame we calculate "speed", measured in object size units. If the distance between current position of the object and the place, where it was observed at the first time, is large enough to let it leave its initial location, all pixels of the background that belong to object's initial location are replaced with corresponding pixels from current frame (i.e. hopefully real background) and object's initial location area is removed from the MBO's list. Fig. 1 presents several examples of detected targets and their segmentation.

The tracking module collects information about the target to enable tracking in subsequent time steps and in order to produce features for reliable classification. The first goal demands knowledge of target position and velocity in the frame. The second goal demands good segmentation of the tracked target. Tracking and segmentation of targets are performed in the following steps: selection of optimal search window, motion detection in the search window and segmentation.

Search for optimal window size and location is performed based on previous target's size, variations in time, previous location, and velocity. The rest of the processing is performed locally within this window.

For detection of moving objects we apply background subtraction with individual thresholds for each image pixel for every target. For each pixel that is marked by the motion detector as *possibly foreground* we calculate the likelihood that this pixel belongs to the target. This procedure combines information available from local (pixel) and global (target) levels of analysis to provide more accurate and robust tracking and segmentation. It permits reliable segmentation and tracking of targets in noisy image sequences with

(a) (b) (c)

Figure 2: Detection and tracking of targets in noisy gray scale sequence: (a) A scene with noisy background; (b) and (c) Detected and tracked targets

changing illumination conditions without special parameter tuning. To deal with very small targets we add temporal subtraction that appeared to be quite effective in case of small moving objects. In Fig. 2 we show detection of targets in noisy gray scale sequence with significant motion in the background.

While tracking, the system accumulates the data required for classification based on the current appearance of the target and its dynamical behavior.

3 Classification

Our system performs classification of moving objects into several predefined classes which can be divided into three basic classes: *Vehicle*, *Animal* and *Human*. Examples of sequences with objects of these types can be seen in Fig. 5. The class *Human* is subdivided further into several sub-classes: *Human*, *Group*, *Carriage*, *Crawling*, *Cyclist*. These classes describe rigid objects (*Vehicle*), non-rigid objects (*Human*, *Group*, *Crawling*) and objects which combine rigid and non-rigid characteristics (*Carriage*, *Cyclist*). The approach we took uses both shape and motion characteristics. We found that the hybrid approach improves the classification significantly. As our basic classifier we use the SVM classifier with sigmoid kernel [15].

3.1 Appearance and Similarity Models

The approach which we present here is based on exploiting the similarity between static silhouettes of objects belonging to the same class. As a training set we use binary pictures of extracted silhouettes within their bounding boxes, rescaled to standard size of $N \times N$ pixels ($N = 32$). We implemented and tested three machine learning methods for classifying an object based on its static silhouettes, as extracted from the segmentation process: the *template based classifier*, *Mahalanobis distance based classifier* and *SVM classifier*. The *template based classifier* calculated the mean silhouette of a class, and computed the similarity of a test silhouette to each of the N_c templates by counting the number of erosion operations needed to identify them. The *Mahalanobis-distance based classifier* calculated for each test silhouette the *Mahalanobis distance* to each of the N_c sets of silhouettes, comprising the training set. The *SVM classifier* used raw silhouettes to learn a non-linear SVM model which optimally separates between classes. Considerable

experimentation with these methods showed significant superiority of the *SVM classifier*. All subsequent references to the similarity-based classifier refer, therefore, to the SVM version thereof.

3.2 Motion based model

The system extracts motion features from target contours (see Fig. 3). We tried two methods for obtaining contours from grey images. The first method is the geodesic active contours [2],[7], in which a contour of a target is sought as a curve $\mathscr{C}$, which should minimize a functional

$$S[\mathscr{C}] = \int_0^{L(\mathscr{C})} g(\mathscr{C})ds + \alpha \int_{\mathscr{C}} da.$$

Here the first term is a geometric functional and the second term is an area minimization term, known as the balloon force [4]. The function $g()$ is a positive edge indicator function that depends on the image, it gets small values along the edges and higher values elsewhere.

In the second method we apply Canny edge detector [1]. Excessive edges obtained from Canny edge detector are then excluded by morphological filters. Our experimentations show that contours obtained with the first algorithm are cleaner than those obtained by the second method, but their extraction demands larger processing time. Accordingly we decided to use the second method in our system. Our experiments demonstrated that target classification was not effected heavily from this decision.

Time dependent features carry considerable amount of information concerning the

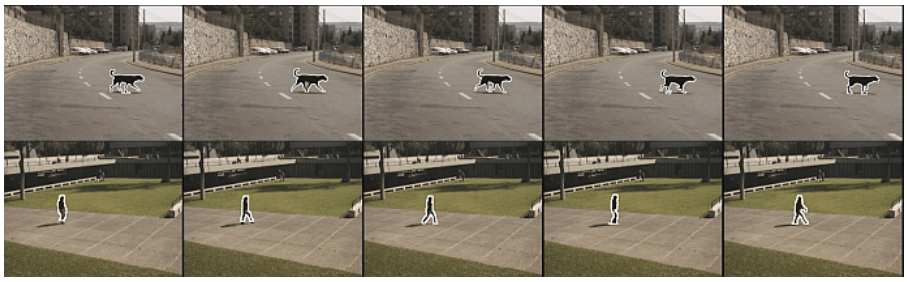
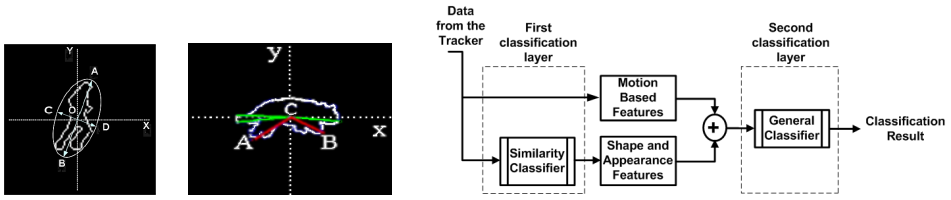


Figure 3: Target contours are used for motion features extraction

identity of an object. For example, the periodicity of human gait is very effective for separating a walking human from a moving car. However, real time constraints enable us to work with rather few, simple, time dependent features. We started with fifteen features. Using an exhaustive search we found an optimal subset of eight features which are based on geometric properties of the fitted ellipse (Fig. 4 (a)) and the star skeleton (Fig. 4 (b)), that is created by connecting the center of mass of the moving object with contour points corresponding to the local maxima of the function measuring the distance between the contour and the center of mass (see [12]). Features we used for description of the temporal characteristics of motion include, for example, the tilt of the "horizontal" axis of the ellipse, (i.e. $\angle DOX$ in Fig. 4(a)) and the angle between the "legs" of the star



(a) (b) (c)

Figure 4: (a) Fitted ellipse (b) Star skeleton (c) Classification flow chart

skeleton ($\angle ACB$ in the Fig. 4(b)). The system accumulates those measurements during 24 consecutive frames and uses this information for construction of the classification vector.

3.3 Combining appearance and motion based features

It is well known that merging several classification methods results in improvement of accuracy and robustness of classification [10]. The performance of both the classifier based on appearance features, described in Subsection 3.1, and the classifier based on time dependent features, described in Subsection 3.2, suggested that a combination thereof is required. Fig. 4 (c) describes the architecture of our classifier. Appearance based data is processed by SVM classifier in the first layer of the classification system. Features vectors used by the hybrid classifier contain processed results of the first-layer classifier, which we call shape and similarity features, and the motion-based features, described in the Subsection 3.2. Each cycle of the second-layer classification requires 24 frames, the number of frames needed for accumulation of motion based information. First-layer classification could in principle be done every frame, but due to real-time performance considerations it is activated once every 8 frames. This means that for every classification done in the second-layer one has 3 first-layer classification results. For every class we calculate the relative fraction p of votes V obtained for this class by first level classifications, namely $p(k) = V(k)/T, \quad k = 1 \ldots N$, where T denotes the total number of first level classifications of this target. It follows that the $k-th$ component of the feature vector is the likelihood that the target belongs to the $k-th$ class. The components from N_c+1 to N_c+8 of the feature vector are assigned to the motion-based features. The feature vector constructed in this way is used as an input to the second-layer SVM classifier. As can be seen from the experiments provided in the Section 4, the classification results achieved by the hybrid classifier were considerably better than the results achieved by each of the basic classifiers separately.

The combined classifier also proved to be more robust in noisy target segmentation conditions. For making the classification robust for occlusions and other sources of misclassification we use a multiple hypothesis approach which is similar to the approach of [12]. All potential targets are classified according to the scheme depicted in Fig. 4 (c) and the result is recorded as a classification hypothesis $\chi(k)$ for each target. Every 24 frames this hypothesis is updated. We accumulate the statistics for targets over a period of time (3 seconds) by building a classification histogram for each moving object. A decision is made at the end of the period based on this histogram.

4 Experiments

We performed experiments with various types of objects in different outdoor scenes. More than 100 sequences were analyzed. Examples of sequences are shown in Fig. 5. To

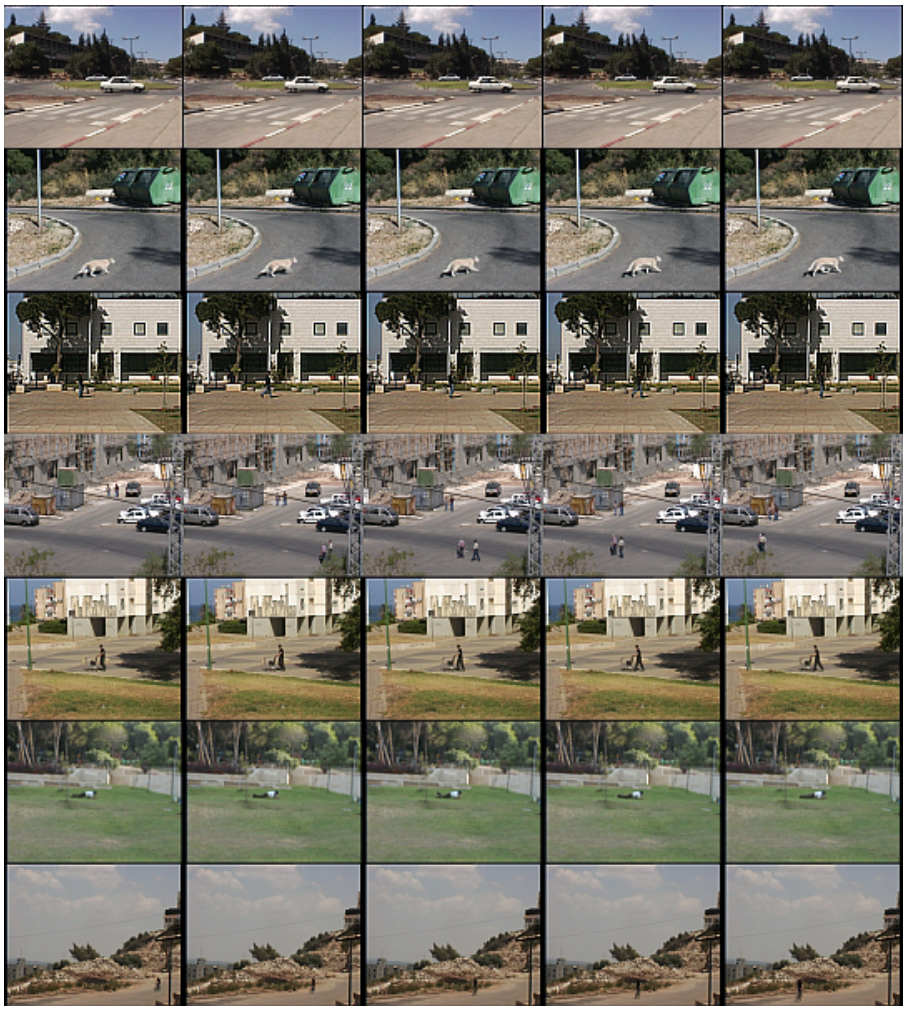


Figure 5: Sequences with targets of all classes: *Vehicle, Animal, Human, Group, Carriage, Crawling and Cyclist*

demonstrate the performance of our classification system we compare confusion matrices which were calculated for three types of classifiers: SVM classifier which used motion features only; SVM classifier which was given shape information only, and hybrid classifier as shown in Fig. 4 (c). For all SVM classifiers we used the LIBSVM library [3]. The results quoted below are obtained by m-fold cross validation with $m = 3$. Caution was taken to perform the splitting of the set of examples object-wise, namely that all frames

of each object belong to either the training set or the validation set.

We tested the classifiers on three different combinations of target classes. The first combination consists of three classes *Vehicle, Animal, Human*. Examples of sequences are shown in Fig. 5. The classification confusion matrices for these classes are given in Table 1.

Table 1: Motion Based Classifier Shape Based Classifier Combined Classifier

	Veh	Ani	Hum	Veh	Ani	Hum	Veh	Ani	Hum
Veh	80.00	13.13	6.67	81.86	10.55	7.59	95.77	2.82	1.41
Ani	26.92	69.23	3.85	16.00	65.00	19.00	3.57	96.43	0.00
Hum	3.09	2.06	94.85	1.13	0.85	98.02	1.00	0.00	99.00

As can be seen in Table 1, both shape based and motion based classification have some difficulties in classifying *Animal*, while *Vehilce* and especially *Human* are classified relatively well by both classifiers. A possible explanation to this phenomenon might be related to the small variability in the *Animal* class (mainly dogs and cats) in the database and to the large variability in the scene range. However the combined classifier gave good results for all the different generic classes.

The second set of targets includes examples of rigid objects - *Vehicle*, non-rigid objects - *Human*, and compound objects - *Carriage* and *Cyclist*. Compound object is a moving target which consists of a non-rigid object (*Human*), in some combination with a rigid object (*Carriage, Cyclist*). The results for our classifiers are given in Tables 2, 3. Here we see that motion based and shape based classifiers have problems in distinguishing between *Cyclist* and *Human*. A possible cause for this result might be related to the fact that direction of motion for all classes was not constrained. As a result, a cyclist riding straight towards the observer (as can be seen in seventh row of Fig. 5) looks very much like walking human. Note again that the combined classifier has good results for all the different classes.

Table 2: Motion Based Classifier Shape Based Classifier

	Veh	Hum	Carr	Cycl	Veh	Hum	Carr	Cycl
Veh	88.52	1.00	1.64	3.28	87.00	10.00	3.00	0.00
Hum	3.09	71.13	15.46	10.31	3.00	91.60	5.00	0.40
Carr	2.56	5.00	74.36	12.82	3.00	12.00	85.00	0.00
Cycl	0.00	25.70	18.60	55.70	11.00	12.00	22.00	65.00

Results for the full set of classes we investigated are shown in Table 4. In Fig. 6 we show examples of scenes, where multiple targets from various classes were detected, tracked, and classified correctly.

Comparing all these tables one can see that performance of the hybrid classifier is significantly better than the performance of classifiers which use only motion based or shape based information. This conclusion is true for all types of considered targets.

Table 3: Combined Classifier

	Veh	Hum	Carr	Cycl
Veh	99.00	1.00	0.00	0.00
Hum	0.00	96.60	0.00	3.40
Carr	4.97	5.00	87.00	3.03
Cycl	0.00	4.00	4.00	92.00

Table 4: Combined Classifier

	Veh	Ani	Hum	Gr	Carr	Crawl	Cycl
Veh	91.89	5.41	1.35	0.00	1.35	0.00	0.00
Ani	3.57	93.86	0.00	0.00	0.00	3.57	0.00
Hum	0.88	0.00	92.11	6.14	0.00	0.00	0.88
Gr	2.70	0.00	0.00	89.19	8.11	0.00	0.00
Carr	4.88	0.00	4.88	2.44	87.90	0.00	0.00
Crawl	2.70	2.70	0.00	0.00	2.70	91.89	0.00
Cycl	0.00	0.00	0.00	4.00	4.00	0.00	92.00

5 Discussion

We present a system which is capable of accurate detection, segmentation, tracking and classification of moving objects in real time. The system is planned for outdoor scenes, and adapts itself to various illumination conditions, to various viewing angles, distances and to previously unknown and noisy backgrounds. We use sophisticated background

Figure 6: Detection and classification of multiple targets

adaptation, allowing for high quality segmentation and tracking of several objects within frame. Targets are classified into one of 7 predefined classes by a hybrid classifier, which combines motion based features and shape based features. Our results show significant superiority of the hybrid classifier over each of the motion based or appearance based classifiers separately. We demonstrate that it is possible to discriminate between considerable number of classes, some of which are quite similar.

A number of themes discussed in this work deserves further development. We plan to improve the appearance and shape based classifier, to include additional shape features. We also plan to continue the investigation on the optimal configuration of classifiers ade-

quate for the problem considered.

References

[1] J. Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, 8(6):679–698, 1986.

[2] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proceedings ICCV'95*, pages 694–699, Boston, Massachusetts, June 1995.

[3] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for svm.

[4] L. D. Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, 1991.

[5] R. T. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto O., and Hasegawa. Vsam final report. technical report CMU-RI-TR-00-12, CMU, Robotics Institute, May 2000.

[6] T. J. Ellis, P. Rosin, and P. Golton. Model-based vision for automatic alarm interpretation. *IEEE Aerospace and Electronic Systems Magazine*, 6(3):14–20, 1991.

[7] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Fast geodesic active contours. *IEEE Transactions of Image Processing*, 10(10):1467–1475, October 2001.

[8] I Haritaoglu, D Harwood, and L Davis. W4: Who, when, where, what: A real-time system for detecting and tracking people. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 222–227, 1998.

[9] M. Harville. A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models. In *ECCV'2002*, 2002.

[10] A. Jaimes and Shih-Fu Chang. Integrating multiple classifiers in visual object detectors learned from user input. In *ACCV 2000*, Taiwan, January 8-12 2000.

[11] D. Koller, J. Weber, and J. Malik. Robust multiple car tracking with occlusion reasoning. In *Proceedings of ECCV*, pages 189–196, 1994.

[12] A. J. Lipton, H. Fujiyoshi, and R. S. Patil. Moving target classification and tracking from real-time video. In *IEEE Workshop on Applications of Computer Vision (WACV)*, pages 8–14, Princeton NJ, October 1998.

[13] Alan McIvor, Vicky Zang, and Reinhard Klette. The background subtraction problem for video surveillance systems. In *International Workshop Robot Vision 2001, Auckland, New Zealand, February 2001.*, pages 176–183, 2001.

[14] E. Rivlin, M. Rudzsky, R. Goldenberg, U. Bogomolov, and S. Lepchev. A real-time system for classification of moving objects. In *ICPR'02*, volume 3, pages 688–691.

[15] V. Vapnik. *Statistical Learning Theory*. John Willey and Sons,inc., 1998.