Noname manuscript No.
(will be inserted by the editor)

Detection and Classification of Vehicles from Omnidirectional Videos Using Multiple Silhouettes

Hakki Can Karaimer · Ipek Baris · Yalin Bastanlar

Received: date / Accepted: date

Abstract To detect and classify vehicles in omnidirectional videos, we propose an approach based on the shape (silhouette) of the moving object obtained by background subtraction. Different from other shape based classification techniques, we exploit the information available in multiple frames of the video. We investigated two different approaches for this purpose. One is combining silhouettes extracted from a sequence of frames to create an average silhouette, the other is making individual decisions for all frames and use consensus of these decisions. Using multiple frames eliminates most of the wrong decisions which are caused by a poorly extracted silhouette from a single video frame. The vehicle types we classify are motorcycle, car (sedan) and van (minibus). The features extracted from the silhouettes are convexity, elongation, rectangularity, and Hu moments. We applied two separate methods of classification. First one is a flowchart based method that we developed and the second is K nearest neighbour classification. 60% of the samples in the dataset are used for training. To ensure randomization in the experiments, 3-fold cross-validation is applied. The results indicate that using multiple silhouettes increases the classification performance.

Keywords traffic surveillance · omnidirectional camera · object detection · vehicle detection · vehicle classification

Hakki Can Karaimer
Department of Computer Engineering, Izmir Institute of Technology, 35430, Turkey
E-mail: karaimer@eecs.yorku.ca
Present address: Department of Electrical Engineering and Computer Science, Lassonde School of Engineering, York University, Canada

Ipek Baris
Department of Computer Engineering, Izmir Institute of Technology, 35430, Turkey
E-mail: ipekbaris@iyte.edu.tr

Yalin Bastanlar
Department of Computer Engineering, Izmir Institute of Technology, 35430, Turkey
E-mail: yalinbastanlar@iyte.edu.tr

1 Introduction

Omnidirectional cameras provide 360 degree horizontal field of view in a single image (vertical field of view varies). If a convex mirror is placed in front of a conventional camera for this purpose, then the imaging system is called a catadioptric omnidirectional camera. Example images from such a camera are given in Fig. 1. Despite its enlarged view advantage, so far omnidirectional cameras have not been widely used in object detection and also in traffic applications like vehicle classification.

(a) (b)

Fig. 1: Two sample omnidirectional images from our dataset. (a) Image with a van (b) Image with a car.

Object detection and classification is an important research area in surveillance applications. A diverse range of approaches have been proposed for object detection. A major group in these studies uses the sliding window approach in which the detection task is performed via a moving and gradually growing search window. Features based on gradients, gradient magnitudes, colours, etc. can be used for classification. A significant performance improvement was obtained with this approach by employing HOG (Histogram of Oriented Gradients) features [8]. Later on, this technique was enhanced with part based models [12].

Regarding HOG features, the sliding window approach was applied to omnidirectional cameras as well [7], where HOG computation was mathematically modified for catadioptric omnidirectional camera geometry. With a similar aim, [13] introduced distortion adaptive descriptors where SIFT and HOG descriptors were computed directly on the wide-angle image by compensating the effect of high amount of radial distortion. Haar-like features were also used with omnidirectional cameras either by converting the image to a panoramic one [17] or directly on the omnidirectional image [10].

Traffic applications require processing of videos where sliding windows in each frame is not feasible. In a recent study [1], HOG features are extracted from the image patches which were identified with a tracking module based on template matching. After the dimension of feature space is reduced, classes are modelled as Gaussian distributions. Classification is performed by assigning samples according to Maximum A Posteriori (MAP) criterion. In this study, vehicles are classified into two classes; tall vehicles (trucks, buses etc.) and short vehicles (cars, vans etc.).

Another major group for object detection uses shape based features after background subtraction step. For instance, [23] created a feature vector consisting of area, breadth, compactness, elongation, perimeter, convex hull perimeter, length, axes of fitted ellipse, centroid and five image moments of the foreground blobs. Linear Discriminant Analysis (LDA) is used to project the data to lower dimensions. Classification is performed by weighted K nearest neighbour (kNN).

When we compare the approaches that use image based features (HOG or Haar-like features) with the approaches that use shape features extracted from silhouettes, extracting shape features is computationally cheaper. Moreover, to decrease the computational load, one should extract image based features only for the region where the moving object exists. Even in that case, fitting a single window around the object is not an easy task especially for omnidirectional cameras. For instance, in [14], where HOG features are computed on virtual perspective views generated from omnidirectional images, object windows are located manually. This makes the approaches using image based features unsuitable for most real-time applications. Motivated by these facts, we decided to develop a shape based method for omnidirectional cameras. Before giving the details of our method, let us present more related work on shape based methods for vehicle classification.

In one of the earliest studies on vehicle classification with shape based features, authors first apply adaptive background subtraction on the image to obtain foreground objects [15]. Location, length, width and velocity of vehicle fragments are used to classify vehicles into two categories; cars and non-cars. In [20], position and velocity in 2D, the major and minor axis of the ellipse modelling the target and the aspect ratio of the ellipse are used as features in a Bayesian Network. In a ship classification study, researchers use MPEG-7 region-based shape descriptor which applies a complex angular radial transform to the shape and classify ships to 6 types with kNN [21]. A 3D vehicle detection and classification study which is based on shape based features, uses the overlap of the object silhouette with region of interest mask which corresponds to the region occupied by the projection of the 3D object model on the image plane [4]. In [6], a similar 3D model based classification is compared with using 2D shape based features and SVM classifier. Later on, they concatenated shape based features and HOG features to create a combined vector to represent each blob and used this method for semi-automatic annotation of vehicles from videos [5].

Instead of standard video frames, [22] employs time-spatial images which are formed by using a virtual detection line in a video sequence. Feature vector obtained from the foreground mask includes width, area, compactness, length-width ratio, major and minor axis ratio of fitted ellipse, and rectangularity. The samples are classified by K nearest neighbour algorithm.

Although not applied to vehicle classification, a radically different method using silhouettes was proposed by [9]. They define "silhouette distance signal" which is the sum of distances between centre of a silhouette and contour points. A silhouette is classified by comparing its distance signal with the ones in the template database. In [2], silhouettes are described with Shape Context descriptors and these are used to align the shapes, i.e. to recover the geometric transformation between the shape to be classified and the ones in the training set. Classification step employs Blurred Shape Model descriptions [11] and K nearest neighbours (kNN).

Regarding the shape based classification studies with omnidirectional cameras, the only work that we found in the literature [18] uses only the area of the blobs and classifies them into two classes; small and large vehicles. In our study, we detect each vehicle type separately using a higher number of features.

Previous work, that employ cameras fixed to buildings, use "area" as a feature to classify vehicles ([23], [18], [4], [22]). Since that feature becomes invalid when the distance between the camera and the scene objects change, the area of the silhouette (size of the vehicle) is not a feature in our method which makes it suitable for portable image acquisition platforms.

The main contribution in our study can be considered as exploiting the information available in multiple frames of the video for vehicle classification. The silhouettes extracted from a sequence of frames are combined to create an "average silhouette". This process is known as "temporal averaging of images" in image processing community and usually used to eliminate noise. We also investigated the use of decision-level fusion, where the classification is made for each video frame separately and the "consensus" of these decisions is determined. When a predefined percentage of samples make the same decision, that vehicle type is chosen. We experimentally show that both of these multi-frame approaches perform better than using a single frame. The classification performance of consensus approach is not as good as that of averaging silhouettes; however it's computation time is shorter. We also present the results of the real-time implementation of our method using consensus approach.

The vehicle types that we worked on are motorcycle, car (sedan) and van (minibus). We applied two different methods for vehicle classification. First one uses shape based features (such as convexity, elongation etc.) one after another in a flowchart (from now on will be referred as "flowchart method"). The second one is K nearest neighbour (kNN) classification. Vehicle classification with kNN was used many times before (e.g. [23], [21], [22]). Although they did not employ omnidirectional cameras, we can consider kNN with single silhouettes as the benchmark method and compare it with using multiple silhouettes for kNN classification.

Our omnidirectional video dataset, together with binary frames after background subtraction, can be downloaded from our website[1]. The organization of the paper is as follows. In Section 2, we introduce the details of silhouette averaging and consensus of silhouettes approaches. Vehicle detector and classifier methods are described in Section 3. Experiment results are presented in Section 4 and finally conclusions are given in Section 5.

2 Using Multiple Silhouettes

The silhouettes are obtained after a background subtraction step and a morphological operation step. For background subtraction, the algorithm proposed in [28] is used, which was one of the best performing algorithms in the review of Sobral and Vacavant [25].

We use the silhouettes as they are extracted from omnidirectional images. We also evaluated the approach where the silhouettes are unwarped from omnidirectional image sampling to perspective image sampling before classification. However, it did not improve the accuracy. We understand that the features we employ (elongation, convexity etc.) are not very sensitive to small amounts of bending in the silhouettes. Thus, we decided not to increase the computation time by unwarping.

In the literature, methods were proposed for using omnidirectional images but computing image features (HOG or SIFT) in the unwarped domain [7],[13]. In this way, if the technique works better on unwarped images, the cost of unwarping is avoided. In our study, since we did not see any improvement by unwarping, any technique to compute unwarped features does not bring any advantage.

2.1 Average Silhouettes

To obtain an "average silhouette" we need to define which frames are used and the silhouettes from these frames should coincide spatially. If a silhouette is in range of a previously specified angle (which we set as [-30°, 30°], and 0° is assigned to the direction that camera is closest to the road), then the silhouette is rotated with respect to the centre of omnidirectional image so that the centre of the silhouette is at the level of the image centre. This operation, also described in Figure 2, is repeated until the object leaves the angle range. Rotating the silhouettes as described is enough to align them since the vehicles are supposed to pass through the road, i.e. they can not have random rotations and sizes. Therefore, our method does not require a more complicated shape alignment process like the one proposed in [2].

Silhouettes obtained in the previous step are added to each other so that the centre of gravity of each blob coincides with others. The cumulative image is divided by the number of frames which results in "average silhouette" (Figure

[1] `http://cvrg.iyte.edu.tr`

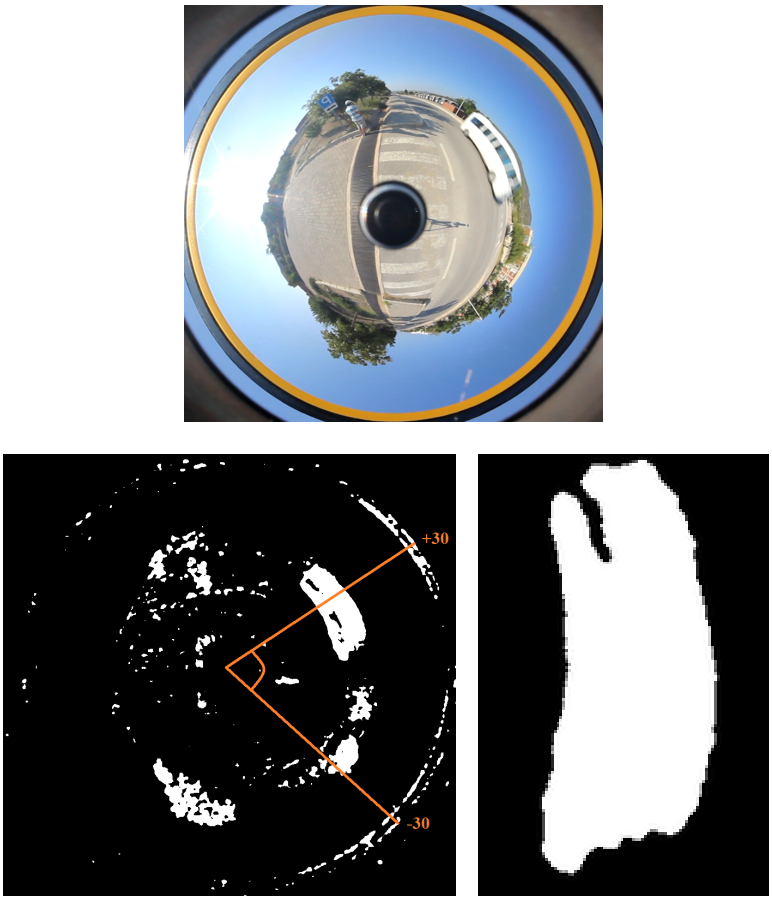


Fig. 2: Top: An example omnidirectional video frame containing a van. Bottom- left: The same frame after background subtraction. Also the angle range that we used, namely [30°, -30°], is superimposed on the image. Centroid of the largest blob is at 29°. Bottom-right: Rotated blob after morphological operations.

3). We then apply an intensity threshold to convert average silhouette to a binary image and also to eliminate less significant parts which were supported by a lower number of frames. Thus we can work with more common part rather than taking into account every detail around a silhouette (Figure 3g). The threshold we select here eliminates the lowest 25% of grayscale levels.

2.2 Consensus of Silhouettes

In addition to silhouette averaging, we present a second way to merge information in multiple frames. The largest blob for each frame is considered as an

input for the single frame classification method and a decision is made for each. When a predefined percentage, for instance 50%, of the samples make the same prediction, we consider that there is a "consensus" among the predictions of the frames and we call that prediction as the vehicle type.

In our analysis, we have seen that silhouette extraction for consensus of silhouettes is computationally cheaper than the average silhouette method. For consensus of silhouettes, morphological operations and rotation of silhouette with respect to omnidirectional image centre takes 15 ms per frame, although for average silhouette, extra two operations, coinciding centres and addition to previous silhouettes takes 169 ms per frame.

3 Detection and Classification

We compare three different approaches of using silhouettes, namely single silhouette that is closest to 0°, averaged silhouette and the consensus of multiple silhouettes. We apply two methods of classification details of which are given in the following.

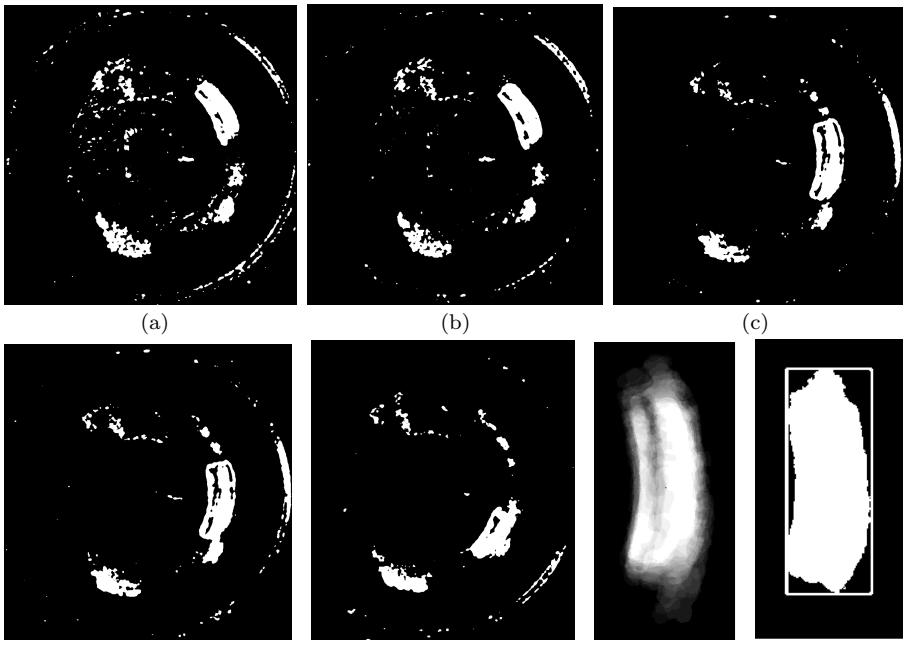


Fig. 3: Example binary images when the centroid of the object is at (a) 29° (b) 26° (c) 0° (d) -11° (e) -29°. (f) Resultant "average silhouette" obtained by the largest blobs in the binary images. (g) Thresholded silhouette and the minimum bounding rectangle.

3.1 Flowchart Method

The steps of this method are summarized in Figure 4. Firstly, a convexity threshold is applied to a silhouette obtained after morphological operations. If the silhouette averaging approach is used, then the silhouette here is the one obtained by the procedure described in Section 2.1. Otherwise it is a single-frame silhouette.

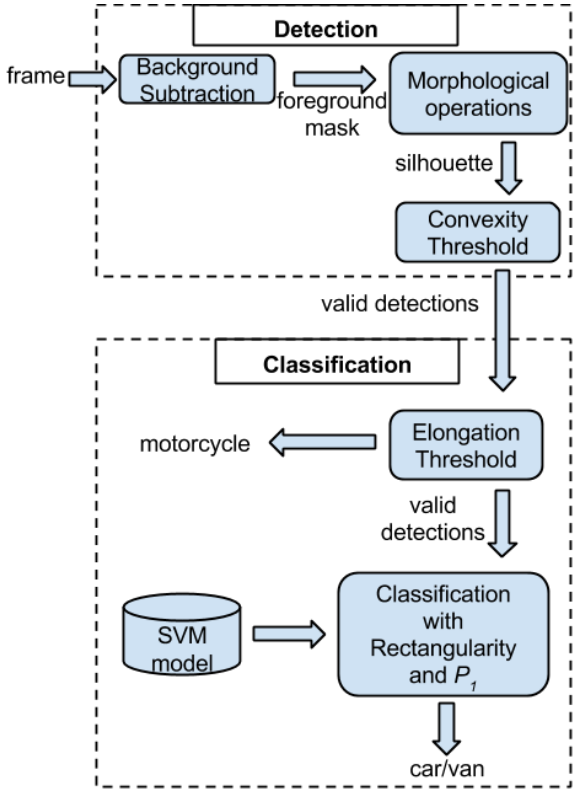


Fig. 4: Block diagram of the detection and classification system. With the proposed method, multiple frames are processed and the extracted average silhouette is used instead of a silhouette from a single frame.

The convexity (1) is used to eliminate detections that may not belong to a vehicle class or poorly extracted silhouettes from vehicles.

$$Convexity = \frac{O_{convexhull}}{O} \tag{1}$$

where $O_{convexhull}$ is the perimeter of the convex hull and O is the perimeter of the original contour [27]. Convexity is always ≤ 1. Since we do not look for

a jagged silhouette, the set of detected silhouettes $\{D_s\}$ is filtered to obtain a set of valid detections $\{D_v\}$ using the convexity threshold ρ.

$$\{D_v\} = \{D_s | Convexity_{D_s} > \rho\} \tag{2}$$

We set $\rho = 0.75$ for our experiments. Figure 5 shows an example silhouette which is eliminated by convexity threshold.

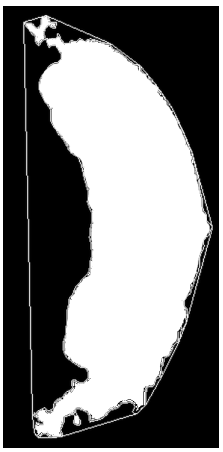


Fig. 5: An extracted silhouette and its convex hull. It is extracted from a van example using a single frame and its convexity is computed as 0.73 which is lower than the threshold. $\rho = 0.75$.

The set of valid detections $\{D_v\}$ is passed to the classification step. The features we employ for classification are; elongation, rectangularity, and Hu moments. Elongation (3) is computed as follows

$$Elongation = 1 - W/L \tag{3}$$

where W is the short and L is the long edge of the minimum bounding rectangle (Figure 3g) which is the smallest rectangle that contains every point in the shape [27].

We observed that the elongation is able to discriminate motorcycles from other vehicle types with a threshold. Then, the set of detected motorcycles $\{D_m\}$ is given by

$$\{D_m\} = \{D_v | Elongation_{D_v} < \tau\} \tag{4}$$

where τ is the elongation threshold. τ is determined using the samples in the training set.

Rectangularity (5) measures how much a shape fills its minimum bounding rectangle [27]:

$$Rectangularity = A_s/A_l \tag{5}$$

where A_s represents area of a shape and A_l represents area of the bounding rectangle. Rectangularity is a meaningful feature to distinguish between sedan cars and vans since the silhouette of a van has a tendency to fill its minimum bounding box. In our trials, however, we observed that setting a threshold for rectangularity alone is not effective enough to discriminate cars from vans. To discriminate the cars and vans better, we defined an extra feature, named P_1 (8), which is based on Hu moments and measures if an extracted silhouette resembles the car silhouettes in the training set more than it resembles the van silhouettes. P_1 (8) is an exemplar-based feature rather than a rule-based one and it is computed as follows:

$$C_1 = \frac{1}{\#cars} \sum_{i=0}^{\#cars} I_2(D_s, Car_i) \tag{6}$$

$$V_1 = \frac{1}{\#vans} \sum_{i=0}^{\#vans} I_2(D_s, Van_i) \tag{7}$$

$$P_1 = C_1 - V_1 \tag{8}$$

For a new sample, P_1 corresponds to the difference between the average I_2 (9) distance to the cars in the training set and the average I_2 distance to the vans in the training set. The mentioned I_2 distance is based on 7 Hu moments [16], used for computing the similarity of two silhouettes:

$$I_2(A, B) = \sum_{i=1\ldots7} \left| m_i^A - m_i^B \right| \tag{9}$$

$$m_i^A = sign(h_i^A) \cdot log(h_i^A) \tag{10}$$

$$m_i^B = sign(h_i^B) \cdot log(h_i^B) \tag{11}$$

where h_i^A and h_i^B are the Hu moments of shapes A and B respectively [3].

If a detection is not classified as a motorcycle, i.e. $Elongation > \tau$, then it can be either a car or a van. To determine the decision boundary between car and van classes we trained a SVM classifier (given in Section 4.1) with a linear kernel using the samples in the training set.

3.2 K Nearest Neighbours

Without using classification scheme in Figure 4, we applied kNN classification on our dataset. Since vehicle classification with kNN using features extracted from a single silhouette can be considered as a benchmark method (e.g. [23], [21], [22]), this way we can investigate the improvement gained by using multiple frames.

kNN method is applied on average silhouette, consensus of silhouettes, and single frame silhouette approaches. On our dataset we used the features of elongation, rectangularity, convexity. We also computed solidity and ellipse axes ratio features. However, increasing the number of features did not improve the results.

4 Experimental Results

4.1 Experiments With a Catadioptric Camera

Using a Canon 600D SLR camera and a mirror apparatus[2] we obtained a catadioptric omnidirectional camera. We constructed a dataset of 49 motorcycles, 124 cars and 104 vans totalling 277 vehicle instances. Dataset is divided into training and test sets. Training set contains approximately 60% percent of the total dataset corresponding to 29 motorcycles, 74 cars and 62 vans. The rest is used as test set. To ensure the randomization of data samples, the procedure is repeated three times with the dataset split randomly into training and test samples. We summarize our experiment results under two subsections belonging to the flowchart method and kNN classification.

4.1.1 Flowchart Method Experiments

We set $\rho = 0.75$ and SVM (using linear kernel)'s parameter $C = 0.2$ for our training set. The elongation threshold is determined by choosing the highest elongation values obtained from motorcycles in the training set since this value easily discriminates motorcycles from other vehicles (this fact can also be observed in Fig. 9b).

Regarding the training of car-van classifier, Figures 6a and 6c show the SVM's linear decision boundary, trained with the average silhouette and single frame silhouette respectively. Training the single frame method with the extracted single frame silhouettes would not be fair since they contain poorly extracted silhouettes. Therefore, samples are manually annotated to be used for the training of single frame method. The silhouette of the object to be annotated is superimposed onto the original video frame and manually corrected, i.e. all pixels that belong to the object are turned on, all others are turned off. Test results with and without averaging silhouettes are shown in Figures 6b and 6d respectively.

[2] `http://www.gopano.com`

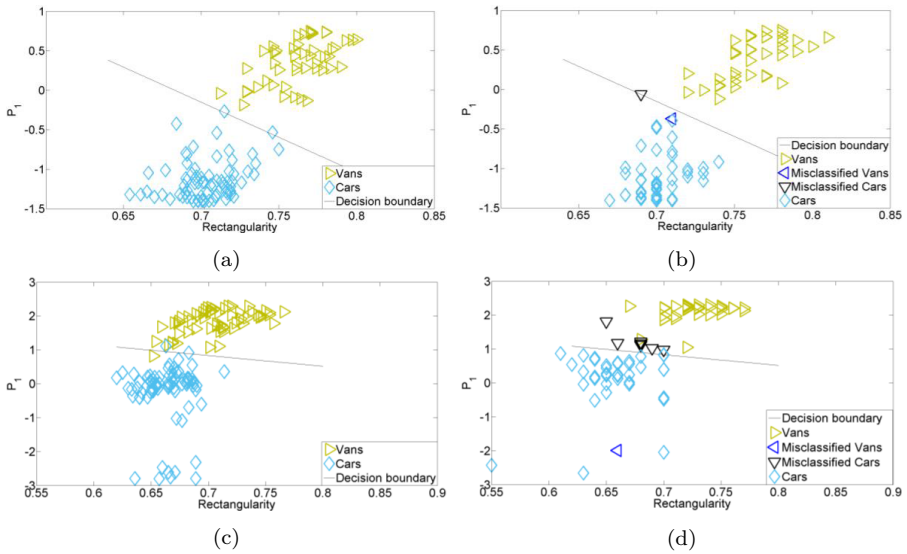


Fig. 6: a) Training result of SVM using the average silhouette method. (b) Test result with the average silhouette method. (c) Training result of SVM without averaging silhouettes (single frame method). (d) Test result without averaging silhouettes, i.e. using single frame silhouettes.

Table 1: Average classification accuracies for each class when $\rho = 0.75$ and $C = 0.2$ for the average silhouette method and for the single frame method.

	Motorcycle	Car	Van	Overall
Average silhouette method	95%	98%	83%	92%
Single frame method	80%	78%	81%	79%

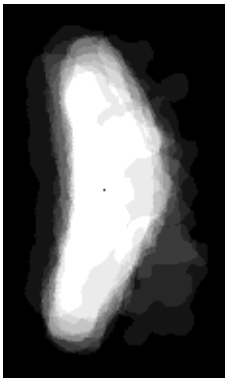
We report the average results in Table 1. Values in the table correspond to what percentage of the samples of a vehicle type is classified correctly. Not surprisingly, exploiting the information in multiple frames by averaging silhouettes has a better performance than using the silhouette in a single frame.

Tables 2 and 3 depict the number of correctly classified (labeled) and misclassified samples for each class with the average silhouette and single frame silhouette methods respectively. Missed samples are the ones eliminated by convexity threshold. Figure 7 shows an example where a car is correctly labeled using the average silhouette, whereas it is misclassified using a single silhouette. Such cases constitute the main performance difference between the two compared methods.

Regarding the convexity threshold ρ, we also tested values other than 0.75. For lower thresholds, less number of samples are eliminated but those samples are not classified correctly. For instance, with $\rho = 0.6$ out of 20 missed van samples (given in Table 2), 17 were passed but they all were classified as cars.

(a)

(b)



(c)

(d)

Fig. 7: Example car silhouettes (a) Original frame (b) Result of using a single silhouette which is misclassified with *rectangularity* = 0.56 and $P_1 = 3.381$, (c) Average silhouette, (d) Thresholded average silhouette classified as car *rectangularity* = 0.68 and $P_1 = -1.602$.

For $\rho > 0.75$, number of missed samples start to increase immediately some of which were correctly classified with $\rho = 0.75$. Therefore, accuracy decreases.

Thanks to using an effective background subtraction algorithm [28], our approach is robust to varying illumination and cases with shadows. Silhouettes are successfully extracted for samples with shining (mostly due to the windows of cars) and low contrast. Regarding shadows, in most frames only a minor amount of shadow is attached to the silhouette. For the frames that are severely affected by the shadow, the main advantage of our method shows its value. Effects of shadows are eliminated during silhouette averaging and thresholding. An visual example is given in Figure 8.

Table 2: Confusion matrix for the approach of using average silhouettes as sum of three folds (For each fold there are 20 motorcycles, 50 cars, and 42 vans in test set).

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	57	0	0
Car	2	146	2
Van	1	4	104
Missed	0	0	20

Table 3: Confusion matrix for single frame method as sum of three folds (For each fold there are 20 motorcycles, 50 cars, and 42 vans in test set).

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	48	8	13
Car	0	118	2
Van	2	20	101
Missed	10	4	10

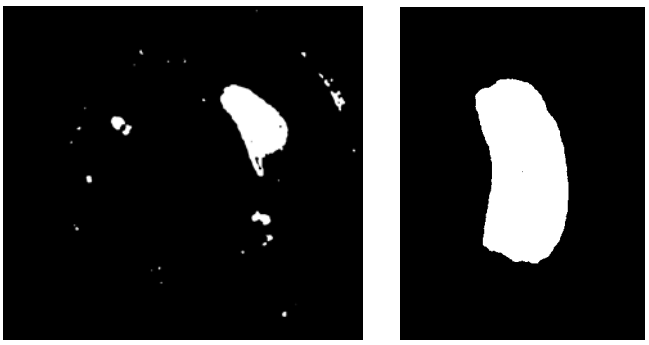


Fig. 8: Left: One of the silhouettes affected by shadow (sharp extrusion at the bottom of the silhouette). Right: Thresholded average silhouette of the same sample (van).

We also examined the performance of "consensus of silhouettes" with the flowchart method. Training set consists of annotated silhouettes. Thresholds, and SVM model used in scheme (cf. Figure 4) are obtained from the training set. As mentioned before, in consensus approach we require a predefined percentage of the samples make the same prediction. Table 4 shows classification accuracies when required consensus percentage changes from 70% to 34%. 34% is the lowest possible consensus percentage since after this value, the chosen class is no longer becomes the largest group. Samples having consensus value less than the defined percentage are assumed to be misclassified (i.e. false-negative). Table 5 shows the confusion matrix for the consensus approach

(34%). When we compare Table 1 and Table 4, we observe that the average silhouette approach has the highest performance. The overall performance of consensus approach (34%) is slightly below the single frame silhouette approach. An important point is the required time to compute the features in the flowchart method. In our analysis we saw that computing P_1 takes 5.46 seconds, while the rest of the features take only 7 milliseconds.

Table 4: Classification accuracies for each class for consensus approach. Required consensus percentage changes from 70% to 34%.

Threshold	Motorcycle	Car	Van	Overall
70%	83%	50%	40%	52%
60%	87%	63%	55%	64%
50%	90%	71%	67%	73%
40%	93%	73%	67%	74%
34%	95%	73%	67%	75%

Table 5: Confusion matrix for consensus approach as sum of three folds (For each fold there are 20 motorcycles, 50 cars, and 42 vans in test set).

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	57	5	18
Car	0	109	6
Van	0	11	84
Missed	3	25	18

4.1.2 K Nearest Neighbour Experiments

As mentioned before, we also examined the classification performance of kNN. Figure 9a shows the features of the annotated silhouettes of all samples (using Euclidean distance) in 3D where dimensions are rectangularity, elongation and convexity. Actual class labels are indicated with different shapes and colours. Top-view of Figure 9a is shown in Figure 9b, where x and y axes refer to rectangularity and elongation respectively. It can be observed that elongation plays a dominant role to discriminate motorcycle class from others. Figure 9c shows the 2D space with dimensions convexity and rectangularity. Rectangularity is not adequate to discriminate cars from vans. With the help of convexity and elongation, car/van classification becomes more accurate.

By dividing the dataset as train and test parts randomly and repeating the experiments three times, we computed average accuracies for different K values. In our experiments, K is selected 5, 10, and 15, and the results are quite similar to each other. Table 6 shows the results for averaged silhouettes, consensus of silhouettes, and single frame silhouettes when K is selected 5. We again observe that the average silhouette is the best performing approach.

Performance of consensus approach is not as good as average silhouette, but it is considerably better than using single frame silhouettes.

Tables 7, 8 and 9 show the confusion matrices of average silhouette, consensus and single frame approaches respectively, to enable readers examine the number of true-positives, false-positives and false-negatives rather than only seeing the average accuracy.

Table 6: Classification accuracies with kNN (K=5) for the average silhouette, consensus of silhouettes and single frame silhouette approaches.

	Motorcycle	Car	Van	Overall
Average silhouette	97%	98%	99%	98%
Consensus of silhouettes	95%	58%	100%	80%
Single frame silhouette	53%	53%	72%	60%

Table 7: Confusion matrix for the average silhouette approach classified with kNN ($K = 5$) as sum of three folds.

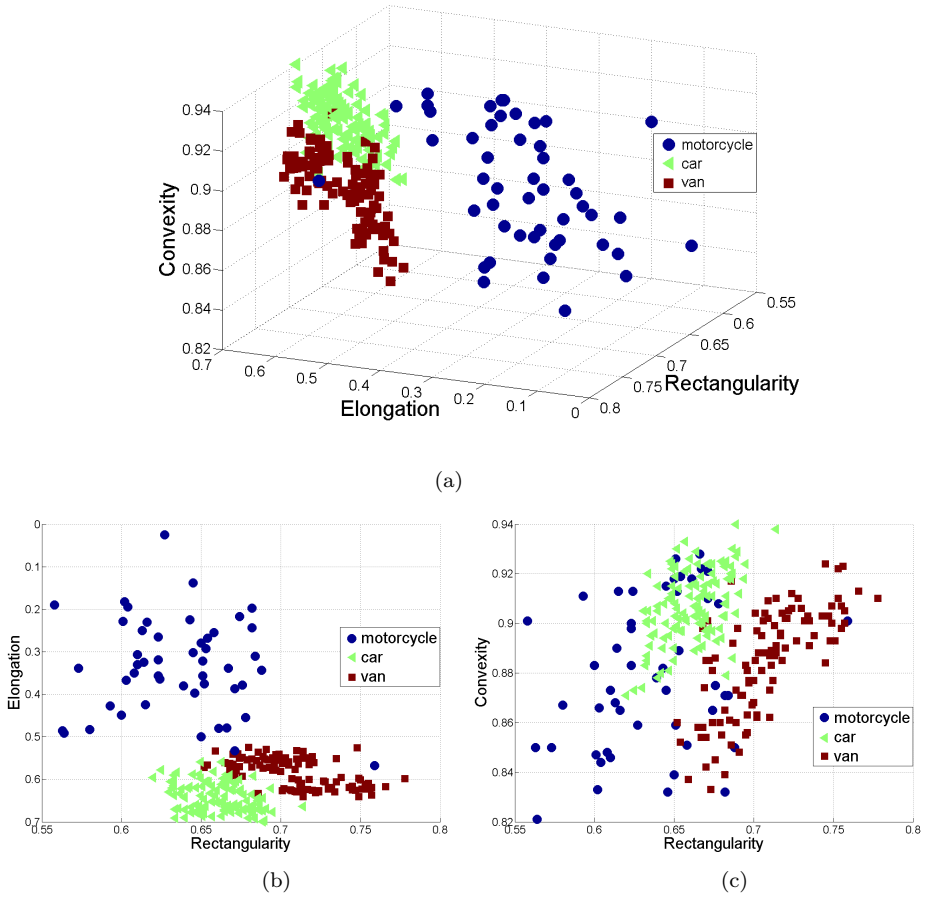
Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	58	0	0
Car	0	147	1
Van	2	3	125

Table 8: Confusion matrix for the consensus approach classified with kNN ($K = 5$) as sum of three folds.

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	57	28	0
Car	1	87	0
Van	2	35	126

Table 9: Confusion matrix for the single silhouette approach classified with kNN ($K = 5$) as sum of three folds.

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	32	63	5
Car	5	80	30
Van	23	7	91



(a)

(b)

(c)

Fig. 9: Extracted features of the annotated silhouettes. (a) All dimensions. (b) First two dimensions. (c) Last two dimensions.

Since P_1 feature is not used in kNN classification, calculation of features is much faster than the flowchart method. Regarding the two multi-frame approaches, although the performance of consensus approach is lower than average silhouette approach, it is more time efficient. Total time for consensus approach with kNN classification is 250 ms including silhouette and feature extraction (assuming 10 frames are used), whereas average silhouette with kNN takes 1850 ms. There is a trade-off between total computation time and classification accuracy for these two multiple frame methods. Computation time for the single frame method is 15 ms which is the shortest not surprisingly. However, consensus approach is also fast enough to be employed in a real-time implementation (an example is given in Section 4.2).

4.2 Real-time Experiments With a Fisheye Camera

Our experience in Section 4.1 reveals that if we want to use multiple silhouettes to increase the performance in a real-time system, consensus approach and the kNN classification is our only choice. Thus, we employed them in our real-time implementation. The overall classification accuracy was recorded as 80% for consensus+kNN approach in Section 4.1. To validate our results, we conduct another experiment. This time we used a fisheye camera. In this way, we can also investigate if the performance depends on the camera type or not. Our fisheye camera is Oncam Evolution 5MP 360-degree [3].

We again constructed a dataset with car, motorcycle and van samples. Test set consists of 76 motorcycles, 126 cars and 124 vans totalling 326 vehicle instances. Table 10 presents the classification results. Overall accuracy is computed as 81% which is very close to the one obtained with the consensus approach in the catadioptric omnidirectional camera (Table 6).

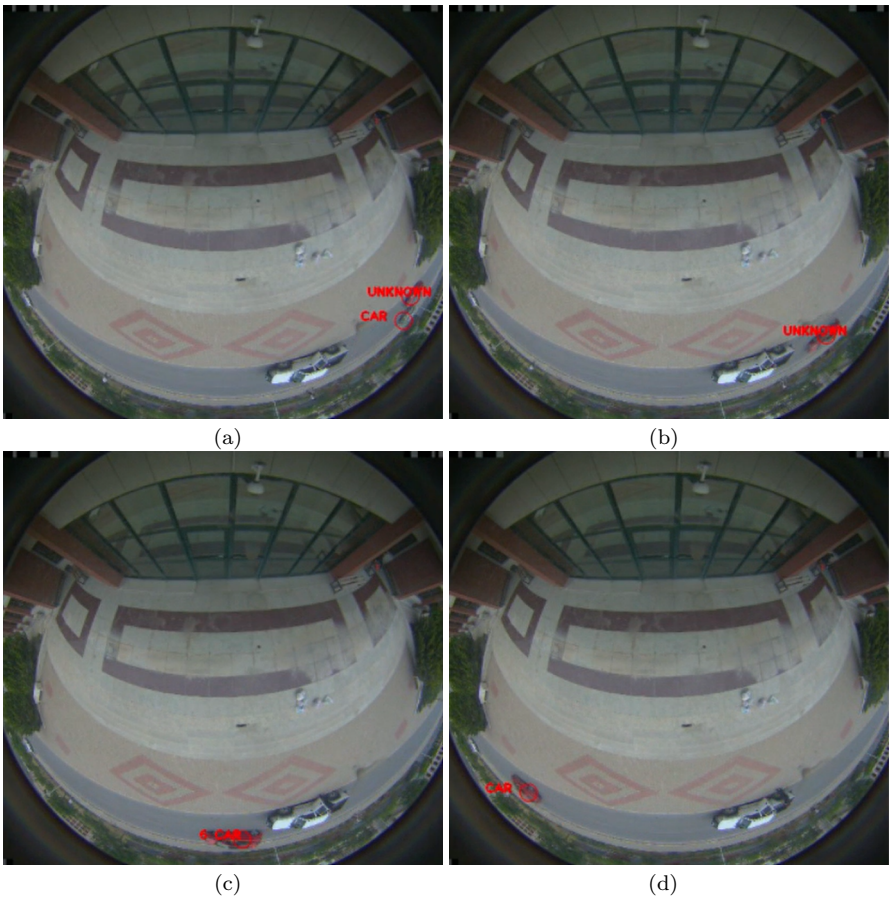
Table 10: Confusion matrix the fisheye camera experiment.

Labeled as:	Actual class: Motorcycle	Actual class: Car	Actual class: Van
Motorcycle	74	8	6
Car	2	95	23
Van	0	23	95
Class accuracy	97.4%	75.4%	76.6%

Another important property of the experiment in this subsection is that we added a tracking module to be able to handle the cases where there are multiple moving objects in the scene. The tracking module consists of tracking the blobs with Kalman Filter [26] and association between the blobs in current frame and previously detected blobs by using Hungarian Algorithm [19], [24]. 2D position (object centroid) and velocity are predicted with Kalman Filter. Hungarian Algorithm finds detection-track pairs with minimum cost which is calculated as the Euclidean distance between the centroid of the detection and the associated track.

Figure 10 shows an example of handling multiple objects. While an object labeled as car leaving the scene, another one is detected and labeled as unknown since its classification is not started yet. Later on, its silhouettes are classified frame by frame (Figure 10c) and final class is determined as car (Figure 10d). This sequence is also a good example of occlusion, since some of the silhouettes of the cars are partially occluded by the steady white pick-up on the road. We see that remaining silhouettes are enough to correctly classify the object as 'car'.

[3] `http://www.oncamgrandeye.com/security-systems/`



(a) (b)

(c) (d)

Fig. 10: Multiple object classification with consensus of silhouettes approach and kNN. (a) A car moving to the right was already classified and is about to leave the scene. At the same time, another car is entering the scene from the right side, detected as a moving object. Both cars are tracked with Kalman filter. (b) Recently entered car is being tracked, its label is still 'unknown' since classification is about to start. (c) Classification has been started, silhouettes are labeled frame by frame. (d) Object exits the classification range ([-30°,30°]) and the final class is determined as 'car'.

Acknowledgements This work was supported by The Scientific and Technical Research Council of Turkey (TUBITAK) Project No: 113E107.

5 Conclusions

We proposed to use multiple frames of a video for shape based classification of vehicles. We applied two different classification methods and compared the

performance of using a single silhouette with the performance of using multiple frames. The first classification method is using features one after another in a flowchart. The second one is kNN classification. We decided to include kNN in our experiments because using single frame silhouette with kNN classification can be considered as the benchmark method in shape based vehicle classification. Results of the experiments indicate a significant improvement in classification accuracy by using multiple frames.

When two alternative approaches of using multiple frames are compared, average silhouette has a higher performance than using consensus of decisions of multiple frames. However, consensus approach has the advantage of being computationally cheaper. In fact, we exploited this advantage and implemented a real-time vehicle classifier with consensus approach and kNN classification. We tested its performance by experiments.

In essence, the advantage of the proposed approach is utilizing the information available in a longer time interval rather than a single frame. Therefore the improvement can be expected for other objects types and domains other than traffic applications.

We use a portable image acquisition platform and our method is independent of the camera-object distance which is more practical than the previously proposed methods that fix the cameras to buildings and use the object's area as a feature since the distance to objects stays same.

References

1. Battiato, S., Farinella, G., Furnari, A., Puglisi, G., Snijders, A., Spiekstra, J.: An integrated system for vehicle tracking and classification. Expert Systems with Applications **42**, 7263–7275 (2015)
2. Battiato, S., Farinella, G., Giudice, O., Puglisi, G.: Aligning shapes for symbol classification and retrieval. Multimedia Tools and Applications **75**, 5513–5531 (2016)
3. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision with the OpenCV Library. O'Reilly Media (2008)
4. Buch, N., Orwell, J., Velastin, S.: Detection and classification of vehicles for urban traffic scenes. In: Visual Information Engineering, 2008. VIE 2008. 5th International Conference on, pp. 182–187 (2008)
5. Chen, Z., Ellis, T.: Semi-automatic annotation samples for vehicle type classification in urban environments. IET Intelligent Transport Systems **9**, 240–249 (2015)
6. Chen, Z., Ellis, T., Velastin, S.: Vehicle type categorization: A comparison of classification schemes. In: Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on, pp. 74–79 (2011). DOI 10.1109/ITSC.2011.6083075
7. Cinaroglu, I., Bastanlar, Y.: A direct approach for object detection with catadioptric omnidirectional cameras. Signal, Image and Video Processing **10**(2), 413–420 (2016). DOI 10.1007/s11760-015-0768-2
8. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 886–893 vol. 1 (2005). DOI 10.1109/CVPR.2005.177
9. Dedeoglu, Y., Toreyin, B., Gudukbay, U., Cetin, A.: Silhouette-based method for object classification and human action recognition in video. In: Computer Vision in Human-Computer Interaction, *Lecture Notes in Computer Science*, vol. 3979, pp. 64–77. Springer Berlin Heidelberg (2006). DOI 10.1007/11754336_7
10. Dupuis, Y., Savatier, X., Ertaud, J., Vasseur, P.: A direct approach for face detection on omnidirectional images. In: Robotic and Sensors Environments (ROSE), 2011 IEEE International Symposium on, pp. 243–248 (2011). DOI 10.1109/ROSE.2011.6058532

11. Escalera, S., Fornes, A., Pujol, O., Llados, J., Radeva, P.: Circular blurred shape model for multiclass symbol recognition. IEEE Trans Syst Man Cybern **41**, 497–506 (2011)
12. Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8 (2008). DOI 10.1109/CVPR.2008.4587597
13. Furnari, A., Farinella, G., Bruna, A., Battiato, S.: Distortion adaptive descriptors: Extending gradient-based descriptors to wide angle images. In: International Conference on Image Analysis and Processing (2015)
14. Gandhi, T., Trivedi, M.: Video based surround vehicle detection, classification and logging from moving platforms: Issues and approaches. In: Intelligent Vehicles Symposium, 2007 IEEE, pp. 1067–1071 (2007). DOI 10.1109/IVS.2007.4290258
15. Gupte, S., Masoud, O., Martin, R., Papanikolopoulos, N.: Detection and classification of vehicles. Intelligent Transportation Systems, IEEE Transactions on **3**(1), 37–47 (2002). DOI 10.1109/6979.994794
16. Hu, M.K.: Visual pattern recognition by moment invariants. Information Theory, IRE Transactions on **8**(2), 179–187 (1962). DOI 10.1109/TIT.1962.1057692
17. Karaimer, H.C., Bastanlar, Y.: Car detection with omnidirectional cameras using haar-like features and cascaded boosting. In: Signal Processing and Communications Applications Conference (SIU), 2014 22nd, pp. 301–304 (2014). DOI 10.1109/SIU.2014.6830225
18. Khoshabeh, R., Gandhi, T., Trivedi, M.: Multi-camera based traffic flow characterization and classification. In: Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE, pp. 259–264 (2007). DOI 10.1109/ITSC.2007.4357750
19. Kuhn, H.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**, 83–97 (1955)
20. Kumar, P., Ranganath, S., Weimin, H., Sengupta, K.: Framework for real-time behavior interpretation from traffic video. Intelligent Transportation Systems, IEEE Transactions on **6**(1), 43–53 (2005). DOI 10.1109/TITS.2004.838219
21. Luo, Q., Khoshgoftaar, T., Folleco, A.: Classification of ships in surveillance video. In: Information Reuse and Integration, 2006 IEEE International Conference on, pp. 432–437 (2006). DOI 10.1109/IRI.2006.252453
22. Mithun, N., Rashid, N., Rahman, S.: Detection and classification of vehicles from video using multiple time-spatial images. Intelligent Transportation Systems, IEEE Transactions on **13**(3), 1215–1225 (2012). DOI 10.1109/TITS.2012.2186128
23. Morris, B., Trivedi, M.: Improved vehicle classification in long traffic video by cooperating tracker and classifier modules. In: Video and Signal Based Surveillance, 2006. AVSS '06. IEEE International Conference on, pp. 9–9 (2006). DOI 10.1109/AVSS.2006.65
24. Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics **5**, 32–38 (1957)
25. Sobral, A., Vacavant, A.: A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. Computer Vision and Image Understanding **122**(0), 4 – 21 (2014). DOI 10.1016/j.cviu.2013.12.005
26. Welch, H., Bishop, G.: An introduction to the kalman filter. University of North Carolina, Department of Computer Science Technical Report TR 95-041 (1995)
27. Yang, M., Kpalma, K., Ronsin, J., et al.: A survey of shape feature extraction techniques. Pattern recognition pp. 43–90 (2008)
28. Yao, J., Odobez, J.: Multi-layer background subtraction based on color and texture. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1–8 (2007)