The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Fast Shadow Detection for Urban Autonomous Driving Applications

Sooho Park and Sejoon Lim, *Student Member, IEEE*

Abstract— **This paper presents shadow detection methods for vision-based autonomous driving in an urban environment. Shadows misclassified as objects create problems in autonomous driving applications. Real-time efficient algorithms in dynamic background settings are proposed. Without the static background assumption, which was often used in previous work to develop fast algorithms, our scheme estimates the varying background efficiently. A combination of various features classifies each pixel into one of the following categories: road, shadow, dark object, or other objects. In addition to pixel level classification, spatial context is also used to identify the shadows. Our results show that our methods perform well for autonomous driving applications and are fast enough to work in real time.**

I. Introduction

The DARPA Urban Challenge posed a challenging problem of autonomous driving in an urban environment [1]. Detecting and tracking other cars and obstacles is essential to make autonomous driving feasible and safe. In conjunction with many other sensors, color cameras have been used frequently for this task [17].

However, shadows prevalent in outdoor scenes interfere with tasks such as the segmentation, tracking, localization and classification of moving objects [9], [19], [10]. For instance, the image segmentation algorithms, which are fundamental for many high-level tasks, often fail to divide two objects into two separate segments due to the shadow cast between them; two cars can be misrecognized as one car. Also, the moving shadow of a car can be misinterpreted as a separate moving object; an autonomously driven car may make an unnecessary attempt to avoid a collision with a shadow. Another challenge is to make a fast shadow detection system which can make decisions fast enough to keep up with the other parts of an autonomous driving system.

Almost all fast shadow detection studies are done in a static background setting with background subtraction methods or using a reference image that does not contain foreground objects [19]. However, in the autonomous driving setting, the background can quickly change. Some methods do not require a background estimate, but these methods use the time-consuming algorithms such as image segmentation [11] and belief propagation [16], and assume an ideal color generation model [16], [7], which may not be robust for outdoor scenes.

In this paper, we propose methods to detect the shadows on an urban road with a fast computation time. First, we start with a pixel-based method that is well suited for an urban driving setting. We choose a sampling region near the vehicle where we can reliably sample the road pixels to estimate the characteristics of the road. Based on this estimate and a machine learning method, each pixel of the image is classified into one of the following categories: road, shadow, dark object, and brighter object. The features used for learning incorporate various characteristics of the pixels including various color space aspects and spatial relations. In particular, a Boosting algorithm was used because it has good generalization performance and fast prediction time. Second, we introduce an efficient region-based method that uses spatial context. This method runs on the result of the pixel-based method and improves the classification performance. The important spatial context information is that shadow pixels should neighboring road pixels around boundaries. Our method utilizes this information without computationally intensive image segmentation. Finally, we show experimental results of these methods using real movie data provided by the MIT DARPA Urban Challenge team. We observed that our methods perform well even for a non-static background setting.

The authors are with the Computer Science and Artificial Intelligence Laboratory, MIT, 32 Vassar St., Cambridge, MA 02139, USA. {dreamneo, sjlim}@mit.edu

II. Background

There have been many efforts to detect shadows for various applications. These efforts include intelligent highway systems, automatic surveillance, picture restoration, and various tracking applications. A good survey on the shadow detection problem is given in [19], [14].

Most of the shadow detection studies attempt to solve a static background problem [9], [19], [14]. In this case, the camera is fixed at a certain location, thus having a static background. Often, the goal is to detect moving foreground objects, such as cars and people. Then, shadow detection algorithms solve a classification problem to separate the shadowed background pixels, which move along with the foreground objects, from the actual foreground objects pixels. In these studies, background subtraction techniques are employed to estimate the color information of the background by averaging or learning a color distribution over multiple frames [8]. Good results of shadow detection have been achieved by using invariant color features that can compare the material nature of a pixel in the current frame to the estimated background and by using other techniques such as moving edge detection [9], [19], [14]. Unfortunately, for the vision-guided driving problem, there is no static background since the car is constantly moving and the scene changes. This work addresses this case.

The shadow classification methods are divided into statistical and deterministic methods [14]; the former uses a training

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

set for the parameter selection and the latter uses manual selection of the parameters. Both methods can be divided into two categories: pixel-based methods and region-based methods. Pixel-based methods use the information of a pixel or a small neighbor around a pixel, while the region-based methods utilize the higher-level spatial information such as geometric constraints between regions [9].

The methods described in [14] are all pixel-based methods using color and illumination information. Many of them rely on the Lambertian surface assumption that the perceived color is the product of the illumination and the spectral reflectance [19], [14]; shadow pixels have similar reflectance to that of the background pixels but lower illumination. However, pixel-based methods often fail to distinguish a dark pixel from a shadow pixel. This problem can be alleviated to a certain degree by enforcing spatial smoothness [10], [13].

Many region-based methods attempt to distinguish shadow boundaries from material boundaries. The shadow boundary is where the illumination change occurs on the same underlying material due to different illumination. The material boundary is the boundary between two different materials. A shadow detection algorithm using the color ratio of neighboring pixels on boundaries is proposed in [3], while the Support Vector Machines (SVM) are used to classify the shadow boundaries among the boundaries obtained by a segmentation algorithm [11]. The pixels inside the darker region touching a shadow boundary were classified as the shadow pixels. In many region-based methods, the image segmentation is required and the performance of the detection can be greatly affected by the segmentation [12].

III. Pixel-based Classification

We develop three kinds of online pixel-based classification methods. First, we address the shadow detection problem in a dynamic background setting whereas most previous online pixel-based classification methods assume a static background and use the background subtraction techniques [9], [19], [14]. Second, we automatically tune the relevant feature parameters through a Boosting method. In many classification tasks, discriminative methods such as the Boosting method [18] have worked better than the generative methods used in some previous works [9], [13]. The choice of the Boosting method over other discriminative methods such as the SVM is explained in Section III-C. Third, we distinguish between dark objects and other objects. Previous works acknowledged that most of the classification errors of their previous methods came from dark pixels of the foreground objects [13], [9]. We suggest a simple way to separate the foreground objects into dark or bright objects using color information and show that this separation improves the classification and gives a better performance metric. Additionally, this separation helps the region-based classification, which is explained in Section IV.

A. Online Background Estimation using Road Sampling

Our goal is to find the shadows on the road plane, cast by other cars and objects. In other words, roads are the background and objects on roads are the foreground. The challenge is to estimate the characteristics of the road, such as the color, from the constantly changing background. With the information, we can utilize various background subtraction techniques to classify shadow pixels.

We describe a method of finding the sampling region of pixels that represents unshadowed road color with high probability. First, we ensure that the sampling region is the road by selecting the regions where it is very unlikely that an object lies; the region close to the car should be the road pixels most of the time. This is true, assuming that the other parts of the autonomous car work properly, since the car will keep a distance from other cars or obstacles. Fig. 1 shows the example where the two boxes are not occupied by other objects. We can configure the camera angle so that a minimal distance between cars is enough for an image to contain these void regions. Second, we ensure that the sampling region is not shadowed by the car itself. The direction of shadow can be easily found by using the sunlight direction determined by the local time and the car orientation through using the on-car compass. We select a region that is not in the direction of shadow.

On this selected sampling region, we use random sampling of the pixels to get the sample average, and we use the running average method [13] to remove the effect of moving shadows created by other objects like other cars and trees. If the shadow is not removed by the running average since the shadow stays for a long time, e.g., in a tunnel, we can just regard it as the road color; it is equivalent to the situation where shadow does not exist at all.

Fig. 1. A scene taken from the camera mounted on the car. The two white boxes represent the sampling regions close to the car where other objects should not lie. The pixels in these regions are randomly sampled.

B. Feature Set

In the literature, various features have been suggested for the classification task. A few of them, such as the color ratio between a pixel and the background estimate [15] and texture information [10], have shown to be quite robust to the different lighting conditions and scene environments, given a good set of parameters which are usually upper and lower bound thresholds [14]. However, the parameter selection process was often done through an extensive manual analysis [15]. In addition, the validity of the features depends on some assumptions such as those in the color generation model [15].

We choose to utilize all of them by learning a data-driven model of these features. Though they may be individually noisy, it would provide a good feature set when they all are combined. There are two categories of the features in general: color-based and gradient-based.

The first category is the color-based features. The color ratio between the color of a pixel $P(x, y)$ and the estimated background color of the pixel $\hat{P}(x, y)$ is used in most previous work [11], [14], [7], [16], [15]. Most studies assume the Lambertian surface color model

$$P_C(x, y) = I_C(x, y)R_C(x, y), C = \{R, G, B\}, \quad (1)$$

where C is one of the three color channels, $P_C(x, y)$ is the value of the C channel of the pixel at (x, y), $I_C(x, y)$ is the illumination from the light sources and $R_C(x, y)$ is the reflectance, which is the characteristic of the material at (x, y). This model implies that the color ratios of the two pixels P_1 and P_2 of one material depend only on the illumination, while they are also affected by the reflectance if the pixels are from different materials; the shadow pixel may have reflectance similar to that of the background pixel. Different works suggested the use of other color spaces rather than the standard RGB space for improving the classification. The chromaticity, the normalized RGB color, was used in [6]. [5] suggested the use of the HSV color space.

The second category is the gradient features. The gradient can be used to achieve texture and edge information [9], [13], [10]. For instance, the metal surface of a car is smooth while the road surface is rough, and the reflectivity of the surfaces is different. We capture this textural information of a small patch around the pixel by the variations of the gradient directions and magnitudes in a patch. The gradient angle variation of two different surfaces is shown in Fig. 2. We used the horizontal, vertical, block and diagonal windows of different sizes to capture this information in various directions and lengths.

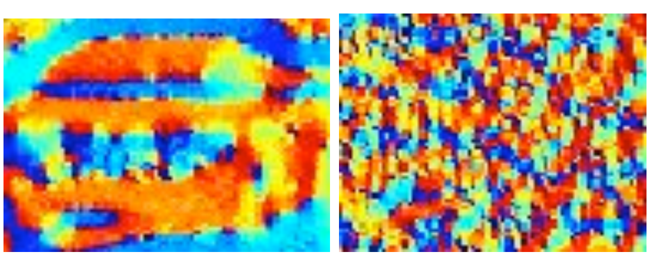

Fig. 2. Gradient angles of a car patch (left) and a road patch (right). The same angles are represented with the same color.

C. The Choice of the Learning Algorithm

Among popular discriminative methods such as the Boosting and the SVM algorithms [2], we choose the Boosting method. It has advantages over the SVM for online detection systems. Though SVM gives a sparse solution, the number of support vectors still grows almost linearly with the number of samples. When the number of the support vectors is n, the prediction takes $O(n^2)$. In our case, one image has $X \times Y$ pixels, where $X = 720$ and $Y = 480$. Using the training samples from many images will result in a huge number of support vectors. However, the Boosting algorithm learns a parameterized model and the prediction only takes $O(k)$ where k is the number of the weak classifiers used and k may be similar to the number of features.

Additionally, in the literature, many pixel-based classification methods using the background subtraction techniques have used one or two thresholds (upper and lower) in some feature space, which is equivalent to using one or two decision stumps. Thus, the decision stumps are used as the weak learners for the Boosting with expectation of good discrimination.

D. The Separation of the Foreground Pixels

The previous works have separated the pixels into three classes; foreground (F), background (B), and shadow (S) pixels. It was noted in many previous works that dark objects such as windshields and dark cars are most often misclassified as shadow [13], [9]. However, dark foreground objects and other foreground objects are put into the same class F, possibly due to the manual work required to separate them.

The common metric for the shadow detection performance is the shadow detection rate η and the shadow discrimination rate ξ [14]. Let the number of true positive pixels of a class X be TP_X and the number of the false negative pixels be FN_X. Also, let the number of pixels of class Y which are not misclassified as class X be $\overline{TP}_Y(X)$. Then

$$\eta = \frac{TP_S}{TP_S + FN_S}, \ \xi = \frac{\overline{TP}_F(S)}{TP_F + FN_F} \quad (2)$$

where S represents the shadow, and F represents the foreground.

We further separate F into the bright objects O and the dark objects D. Intuitively, the classification between S and D will be harder than between S and O, as the hard-to-classify pixels will be more present in D. In this case, the discrimination rate ξ becomes

$$\xi = \frac{\overline{TP}_O(S) + \overline{TP}_D(S)}{TP_O + FN_O + TP_D + FN_D} = \frac{\overline{TP}_O(S) + \overline{TP}_D(S)}{N_O + N_D} \quad (3)$$

where N_X is the total number of pixels from class X. Note that ξ depends on N_O and N_D. Even when $\frac{\overline{TP}_D(S)}{N_D}$ is really low, meaning that an algorithm fails to distinguish dark objects from shadow, ξ can be still high if $\frac{\overline{TP}_O(S)}{N_O}$ is high and $N_D \ll N_O$.

The separation would allow better evaluation of the algorithms. Another advantage is that cascaded classification [18] is possible with the separation. We may use a small number of features to classify the pixels into $\{B, O\}$ and $\{S, D\}$. Then, we use more features to classify the pixels in $\{S, D\}$ into S and D. This will reduce the computation time and improve the classification rate as well.

In this work, we simply choose to use the road color estimate of the current frame to divide the foreground pixels of the frame into two separate classes O and D. If a pixel in F is darker than the estimated road color, it is classified as a D pixel, but otherwise as an O pixel. Note that manual work to separate the pixels into O and D is not required.

IV. Region-based Method

The pixel-based classification has the limitation that it does not fully utilize the spatial context. The labels of neighbor pixels should be similar and we have only partly addressed this problem by using the features generated from a spatial neighborhood. Additionally, the shadowed road regions should be neighboring the not-shadowed road regions; the existence of the shadow is geometrically constrained [9]. To address these issues, many post-processing approaches to the pixel-based classification have been suggested, including morphological operations, probabilistic relaxation, and image segmentation [19].

In this section, we describe two region-based methods that can improve upon the pixel-based classification. First, we apply probabilistic smoothing to impose the consistency of the labels in a spatial neighborhood. Our second method further improves the result by using the geometric information that a shadowed road region should be neighboring a not-shadowed road region as in [9], [15], but with special attention to the dark pixels. We show that the good classification of dark and shadow pixels are critical in this process. Finally, we describe a lazy classification approach to improve the processing speed.

A. Imposing Spatial Consistency

We impose spatial consistency on the pixel-based classification results using a probabilistic smoothing technique. Given a pixel x, we use a prior that the labels of the neighboring pixels $\{x_1, x_2, ...x_n\}$ should be equal to the label of the pixel x. Typically, the pixels in a square window of small size are chosen as neighbors. However, instead of using all pixels in a window as in [9], we only consider the pixels inside a region defined by the surrounding edges. This will help to choose only the pixels of the same class as the neighbors, because usually the edges well separate the two different regions, which contain the pixels of two different classes, as in Fig. 3.

Given the predicted labels of the neighboring pixels $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_n)$, the maximum likelihood estimator of the label of the center pixel x is given by

$$\begin{aligned} x &= \underset{x=S,D,R,O}{\operatorname{argmax}} \; P(\hat{\mathbf{x}}|x_1 = x_2 = \cdots = x_n = x) \\ &= \underset{x=S,D,R,O}{\operatorname{argmax}} \; P(\hat{x_1}|x)P(\hat{x_2}|x)\cdots P(\hat{x_n}|x) \end{aligned}$$

using the prior information that the labels should be consistent in the neighborhood. We use the cross-validation error of the learned model as $P(\hat{x}|x), x = \{S, D, R, O\}$.

B. Detection of the Shadow Boundary Pixels

The detection process of the shadow boundary pixels is first initiated with the edge detection that finds both material boundaries and shadow boundaries. We use the Canny edge detector, which is considered to be an ideal edge detection algorithm due to its robustness [4].

Fig. 3. The edge detection result of a frame. Three types of boundaries are shown (blue: shadow and road, red: road and dark, yellow: shadow and bright).

A shadow boundary pixel should have the road pixels on the brighter side and the shadow pixels on the darker side. Let a small set of the neighboring pixels at the brighter side be K_B and the darker side be K_D. Then, the probability of a edge pixel p to be a shadow boundary pixel is given by

$$P(p \text{ is on a shadow boundary}) \propto \prod_{i \in K_B} P(\hat{x}_i|x_i = R) \prod_{j \in K_D} P(\hat{x}_j|x_j = S)$$

If $P(p \text{ is on a shadow boundary})$ is higher than a certain threshold, p will be classified as a shadow boundary pixel. After the detection of the shadow boundaries, only the pixels that are connected to a shadow boundary pixel are finally classified as the shadow pixels. We claim that the boundary detection process should be very selective, with a high threshold, as only a few pixels of a boundary have to be detected. The boundary detection algorithms such as the one in [15] classify dark pixels as shadow candidate pixels using the color information alone. This may result in spurious shadow boundary detection in the case such as the one in Fig. 3, where the dark car share a boundary with the road.

The above step eliminates the false positives of the shadow class. Simple morphological processes can be applied to flip the misclassified pixels inside the shadow regions to eliminate the false negatives [9].

C. Lazy Classification of the Labels

For the purpose of detecting shadow pixels, we can do the pixel-based classification for only a subset of all the pixels in the image once we know that it is required; so we call this lazy classification. Once an edge is detected by an edge detection method such as Canny method, it is sufficient to know the labels of the pixels nearby the edges. Thus, we need to classify only the pixels nearby the edges. After detecting the shadow boundary by the method described in this section, we can do the process of classifying the pixels that are connected to a pixel on the shadow boundary.

V. RESULTS

The driving video sequence of a vehicle in a city environment was provided by the MIT DARPA Urban Challenge team. We used one sequence for the training and another sequence, which was taken at a different time on a different road, for the validation. There are 500 frames per each video sequence. For training, the pixels in the first sequence are manually labeled[1]. The online background estimation algorithm was run on the video sequence to get the road pixels information of each frame and the features are generated per each labeled pixel using this information. For the validation, the same background estimation algorithm gives the road pixels information per each frame and the pixels in the frame are classified using this information. For the results shown in TABLE I and TABLE II, we randomly sampled the same number of pixels of each class per each frame, to avoid the temporal and spatial bias in the samples.

The one-versus-one approach was used for the multi-class classification, where binary classification is done per each pair of the class and the final class is decided by the votes from these binary classifications [2]. The number of weak classifiers used for the Boosting models was selected through a cross validation in the training set. The use of many features were helpful for the better classification and the Boosting algorithm was successful in avoiding overfitting despite the much increased dimension of the feature space as shown in Fig. 4. The number of weak classifiers was different for each classification task as the difficulty of the separation was expected to be different; D vs. O classifier used only one weak classifier while R vs. D classifier used 45 weak classifiers.

The detail of the pixel-based classification using the Boosting method is shown in TABLE I and TABLE II, while the latter had only three classes without the separation of the dark and other (not-dark) objects. The simple separation resulted in better classification. We already explained that putting O and D in one class can distort the prediction result because O and S are very easy to separate while D and S are not. For example, if F contains many more samples of O , a

[1]We randomly selected objects (shadow, cars, etc.) in each frame and used a polygon to label an object

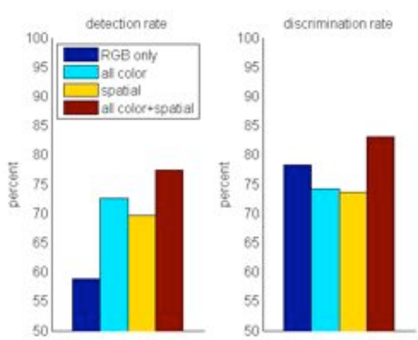

Fig. 4. The detection rate and the discrimination rate using the different subset of the features. Using the color features from all color spaces combined with the spatial features performed the best.

TABLE I

THE CLASSIFICATION RATE OF THE DIFFERENT CLASSES WITH SEPARATING THE FOREGROUND OBJECTS INTO THE TWO CLASSES

True label	Predicted label $\hat{R}$	$\hat{S}$	$\hat{D}$	$\hat{O}$	# samples
R	77.62%	2.25%	5.04%	15.09%	50000
S	1.41%	75.39%	22.85%	0.35%	50000
D	5.01%	29.84%	65.15%	0.00%	50000
O	3.12%	0.90%	2.18%	93.80%	50000

$\eta = 75.39\%$, $\xi = 83.54\%$

TABLE II

THE CLASSIFICATION RATE OF THE DIFFERENT CLASSES WITHOUT SEPARATING THE FOREGROUND OBJECTS INTO THE TWO CLASSES

True label	Predicted label $\hat{R}$	$\hat{S}$	$\hat{F}$	# samples
R	60.17%	34.75%	5.07%	50000
S	0.01%	75.54%	24.45%	50000
F	3.51%	49.16%	47.33%	100000

$\eta = 75.54\%$, $\xi = 50.84\%$

classifier that misclassifies all D and correctly classifies all O will have a good overall prediction error. Both the detection rate and the discrimination rate in TABLE I are comparable to the best performance reported for the pixel-based methods in a static background setting [14].

We compared the performance of our sampling-region-based method as described in Section III-A with that of an adaptive static background estimation method using the runtime moving average of the color information per each pixel. The comparison result is shown in TABLE III. Our sampling strategy to sample the road colors from the near-region of the car have a significantly better detection rate. Especially, if there is busy traffic and the road is occupied by many cars, the difference should be even greater as the runtime average of a pixel is the average color of different cars and not the road.

The ability of the region-based method to remove the false positives of the shadow is shown in TABLE IV quantitatively and in Fig. 5 qualitatively. Note that with a reasonable detection rate it may be possible to detect shadows better with a post-processing step such as the ones suggested in [9]. However, the discrimination rate can be more important in the driving application as the car should avoid the risk of identifying other vehicles as shadow.

We were able to process about 20 images per second with C implementation on a Pentium 4 2.4 Ghz machine, even

TABLE III

THE SHADOW DETECTION RESULTS WITH DIFFERENT BACKGROUND ESTIMATION METHODS

Background estimation method	η	ξ
Our sampling strategy	76%	84%
Adaptive static background estimation	48%	86%

TABLE IV

THE COMPARISON OF THE PIXEL-BASED CLASSIFICATION AND THE REGION-BASED CLASSIFICATION

Classification method	η	ξ
Pixel-based	76%	84%
Region-based	75%	91%

(a) Original (b) Pixel-based (c) Boundary-based

Fig. 5. The step-by-step improvement of the shadow detection.

without fully applying the speed-up techniques introduced in this paper. This is much faster than the other methods, not using a background estimation, such as the belief propagation algorithm in [16] which reported 5 minutes processing time per image.

VI. Conclusion

We presented a fast shadow detection algorithm for an urban autonomous driving application. Most, if not all, of the existing fast shadow detection algorithms rely on the fact that the background is static. With the background information, the foreground objects and the shadow pixels were separated in various feature spaces including the color ratio space, which is shown to be invariant. We noted the fact that the background in our problem, the urban road, is rather uniformly colored and flat though its color gradually shifts and its shape changes. Then, we introduced the sampling strategy through which the color information of the road can be estimated. The color ratio between the color of a pixel and the estimated road color, combined with the spatial features that can take into account the characteristics of the different surfaces, has been proved effective to classify the shadow pixels on the road accurately.

Instead of manually choosing a few features out of many useful features for the pixel-based classification, a Boosting algorithm was chosen to learn the parameters of the classification model automatically. The learning algorithm succeeded in avoiding overfitting despite the much increased dimension of the feature space, where a manual parameter selection would not be feasible. We also have shown that the separation between the darker and the brighter foreground pixels has enabled better classification as well as fair evaluation of the classification performance, as the brighter foreground pixels are easy to classify and the result may look better with many of them in the validation set. The mistakes made in the pixel-based classification could be fixed with the region-based method that enforces spatial consistency and geometric constraints.

The computation time of the method was shown to be fast, without careful optimization, as its overall complexity is $O(n)$ where n is the number of pixels in an image and the learned model is an efficient parametric model. Further speed-up techniques for the classification were described, including cascaded classification and lazy classification. We also note that the pixel-based classification can be parallel processed.

References

[1] http://www.darpa.mil/grandchallenge/index.asp.

[2] E. L. Allwein, R. E. Schapire, and Y. Singer. Reducing multiclass to binary: a unifying approach for margin classifiers. *The Journal of Machine Learning Research*, 1:113–141, 2001.

[3] K. Barnard and G. Finlayson. Shadow identification using colour ratios. *IS&T/SID 8th Colour Imaging Conference: Colour Science, Systems and Appl*, pages 97–101, 2000.

[4] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[5] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti. Improving shadow suppression in moving object detection with hsvcolor information. *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pages 334–339, 2001.

[6] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction. *Lecture Notes in Computer Science*, pages 751–767, 2000.

[7] G. D. Finlayson, S. D. Hordley, and M. S. Drew. Removing shadows from images. *Lecture Notes in Computer Science*, pages 823–836, 2002.

[8] T. Horprasert, D. Harwood, and L. S. Davis. A statistical approach for real-time robust background subtraction and shadow detection. *Proc. IEEE ICCV*, 99:1–19.

[9] A. J. Joshi, S. Atev, O. Masoud, and N. Papanikolopoulos. Moving shadow detection with low-and mid-level reasoning. *Robotics and Automation, 2007 IEEE International Conference on*, pages 4827–4832, 2007.

[10] A. Leone, C. Distante, and F. Buccolieri. A texture-based approach for shadow detection. *Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.*, pages 371–376, 2005.

[11] M. D. Levine and J. Bhattacharyya. Removing shadows. *Pattern Recognition Letters*, 26(3):251–265, 2005.

[12] L. Lucchese and S.K. Mitra. Colour Image Segmentation: A State-of-the-Art Survey. *Proceedings. Indian National Science Academy Part A*, 67(2):207–222, 2001.

[13] I. Mikic, P. C. Cosman, G. T. Kogut, and M. M. Trivedi. Moving shadow and object detection in traffic scenes. *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 1, 2000.

[14] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara. Detecting moving shadows: Algorithms and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 918–923, 2003.

[15] E. Salvador, A. Cavallaro, and T. Ebrahimi. Shadow identification and classification using invariant colormodels. *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, 3, 2001.

[16] M. F. Tappen, W. T. Freeman, and E. H. Adelson. Recovering intrinsic images from a single image. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, pages 1459–1472, 2005.

[17] A. Tsalatsanis, K. Valavanis, and A. Yalcin. Vision based target tracking and collision avoidance for mobile robots. *J. Intell. Robotics Syst.*, 48(2):285–304, 2007.

[18] P. Viola and M. Jones. Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade. *Advances In Neural Information Processing Systems*, 2:1311–1318, 2002.

[19] Li Xu, Feihu Qi, Renjie Jiang, Yunfeng Hao, Guorong Wu, Li Xu, Feihu Qi, Renjie Jiang, Yunfeng Hao, and Guorong Wu. Shadow detection and removal in real images: A survey, 2006.