Illumination Normalization with Time-dependent Intrinsic Images for Video Surveillance

Yasuyuki Matsushita[†] Ko Nishino[‡] Katsushi Ikeuchi[†] Masao Sakauchi[†] † Institute of Industrial Science, The University of Tokyo, Tokyo, Japan {yasuyuki@sak, ki@cvl, sakauchi@sak}.iis.u-tokyo.ac.jp ‡Department of Computer Science, Columbia University, New York, U.S.A. kon@cs.columbia.edu

Abstract

Cast shadows produce troublesome effects for video surveillance systems, typically for object tracking from a fixed viewpoint, since it yields appearance variations of objects depending on whether they are inside or outside the shadow. To robustly eliminate these shadows from image sequences as a preprocessing stage for robust video surveillance, we propose a framework based on the idea of intrinsic images. Unlike previous methods for deriving intrinsic images, we derive time-varying reflectance images and corresponding illumination images from a sequence of images. Using obtained illumination images, we normalize the input image sequence in terms of incident lighting distribution to eliminate shadow effects. We also propose an illumination normalization scheme which can potentially run in real time, utilizing the illumination eigenspace, which captures the illumination variation due to weather, time of day etc., and a shadow interpolation method based on shadow hulls. This paper describes the theory of the framework with simulation results, and shows its effectiveness with object tracking results on real scene data sets for traffic monitoring.

1 Introduction

Video surveillance systems involving object detection and tracking require robustness against illumination changes caused by variation of, for instance, weather conditions. Annoying obstacles include not only the change of illumination conditions, but also the large shadows cast by surrounding structures, i.e. large buildings and tall trees. Since most visual tracking algorithms rely on the appearance of the target object, typically using color, texture and feature points as cues, these shadows degrade the quality of tracking. In urban scenes, where building robust traffic monitoring systems is of special interest, it is usual to have large shadows cast by tall buildings surrounding the road, e.g. Figure 2 (a) below. Building a robust video surveillance system under such an environment is a challenging task. To make the system insensitive to dramatic change of illumination conditions and robust against large static cast shadows, it would be valuable to cancel out those illumination effects from the image sequence. Our goal



Figure 1: System diagram for illumination-normalization.

is to "normalize" the input image sequence in terms of the distribution of incident lighting to remove illumination effects including shadow effects. We should note that our method does not consider shadows cast by moving objects but those cast by static objects such as buildings and trees. To achieve this goal, we propose an approach based on *intrinsic images*. Our method is composed of two parts as shown in Figure 1.

The first part is the estimation of intrinsic images, which is an off-line process, depicted in Figure 1 A. In this part, first, the scene background image sequence is estimated to remove moving objects from the input image sequence. Using this background image sequence, we then derive intrinsic images using our estimation method which is extended from Weiss's method [1]. Using estimated illumination images, which is a part of intrinsic images, we are able to robustly cancel out the illumination effects from input images of the same scene, enabling many vision algorithms such as tracking to run robustly. After the derivation, we construct a database using PCA, which we refer to as *illumination eigenspace*, which captures the variation of lighting conditions in the illumination images. The database is used for the following direct estimation method. The second part is direct estimation of illumination images, shown in Figure 1 B. Using the pre-constructed illumination eigenspace, we estimate an illumination image directly from an input image. To obtain accurate illumination images, shadow interpolation using shadow hulls is accomplished.

In the remainder of this paper, we first overview related work in Section 1.1, and in Section 2, we propose a method to derive time-varying reflectance images R(x, y, t) and corresponding illumination images L(x, y, t). Derivation of the illumination-invariant images using the illumination images L(x, y, t) is described in Section 3. In Section 4, we propose a method to estimate R and L directly from an input image using PCA. In addition, to obtain more accurate illumination images, we use a shadow hull based interpolation method, which is described in Section 4.1. Experimental results are described in Section 5. Finally, we conclude the paper in Section 6.

1.1 Related Work

Barrow and Tenenbaum proposed to consider every retinal image as a composition of a set of latent images, which they refer to as intrinsic images [2]. One type of the intrinsic images, R, contains the reflectance values of the scene, while the other type, L, contains the illumination intensities, and their relationship can be described by $I = R \cdot L$. Since illumination images, L, represent the distribution of incident lighting onto the scene while reflectance images, R, depict the surface reflectance properties of the scene, this representation becomes useful to analyze and manipulate the reflectance/lighting properties of the captured scene.

While decomposing a single image into intrinsic images, namely a reflectance image and an illumination image, remains a difficult problem [2, 4, 5], deriving intrinsic images from image sequences has seen great success. Recently, Weiss developed an ML estimation framework [1] to estimate a single reflectance image and multiple illumination images from a series of images captured from a fixed view point but under significant lighting condition variation. Finlayson et al. [14] proposed a similar approach to ours independently. They derive the scene texture edges from the lighting-invariant image, and by subtracting those edges from the raw input images, they successfully derive shadow-free images of the scene. We also take advantage of the fact that the reflectance image essentially models the scene texture in a manner invariant to lighting conditions. We accomplish edge substitution between the reflectance image and illumination images, enabling robust derivation of scene-texture-free illumination images.

Several other work on shadow detection have been proposed. Deterministic model-based approaches to detect shadow regions are proposed by Kilger [6] and Koller et al. [7] that exploit gray level, local and static features. In statistical approaches, Stauder et al. [8] and Jiang et al. [9] proposed a non-parametric approaches independently that use color, global and dynamic features for enhancing object detection.

2 Intrinsic image estimation

Weiss's method to derive intrinsic images is useful for diffuse scenes, however, it has a problem when applied to scenes containing non-Lambertian surfaces. Weiss's method implicitly assumes the scene is composed of Lambertian surfaces, and this assumption is inevitable from the definition of the reflectance image which has to be independent from illumination changes. For real world scene, we cannot expect the assumption to hold. A typical example is white lines on the road surface, which show variable reflection with respect to illumination changes. Therefore, while the time invariant reflectance image R(x, y), derived by Weiss's framework, reasonably describes the scene texture without lighting effects, the estimated illumination images L(x, y, t) tend to contain considerable amount of scene texture. Those scene textures should not really be a component of the "illumination" image, since illumination images should represent the distribution of incident lighting. These annoying scene textures in illumination images arise at scene regions where surfaces of different reflectance properties meet. Therefore it is necessary to assume a set of time-varying reflectance images R(x, y, t) instead of a single one.

Our estimation method is based on Weiss's method. We first estimate Weiss's reflectance image to use it as a scene texture image. We denote Weiss's reflectance image and illumination image with subscript w, i.e. R_w and L_w , and our reflectance image and illumination image, R and L respectively. First, we apply Weiss's ML estimation method to the image sequence to derive a single reflectance image, $R_w(x, y)$, and a set of illumination images, $L_w(x, y, t)$. Our goal is to derive time-varying, i.e. lighting condition dependent, reflectance images R(x, y, t) and corresponding illumination images L(x, y, t) that do not contain scene texture.

$$I(x, y, t) = R(x, y, t) \cdot L(x, y, t)$$
(1)

We use lower-case letters to denote variables in log domain, e.g. r represents the logarithm of R. With n-th derivative filters f_n , a filtered reflectance image r_{wn} is computed by taking median along the time axis of $f_n \star i(x, y, t)$ where \star represents convolution. We used two derivative filters, i.e. $f_0 = [0 \ 1 \ -1]$ and $f_1 = [0 \ 1 \ -1]^T$. With those filters, input images are decomposed into intrinsic images by Weiss's method as described in Equation (2). The method is based on the statistics of natural images [3].

$$\hat{r}_{wn}(x,y) = \text{median}_t \{ f_n \star i(x,y,t) \}$$
(2)

The filtered illumination images $l_{wn}(x, y, t)$ are then computed by using estimated filtered reflectance image r_{wn} .

$$\hat{l}_{wn}(x, y, t) = f_n \star i(x, y, t) - \hat{r}_{wn}(x, y)$$
 (3)

To be precise, l is computed by l = i - r in the unfiltered domain in Weiss's original work while we estimate l in the derivative domain for the following edge-based manipulation.



Figure 2: (a) an input image i(x, y, t), (b)Weiss's reflectance image $r_w(x, y)$, (c) Weiss's illumination image $l_w(x, y, t)$, (d) our time-varying reflectance image r(x, y, t), (e) our illumination image l(x, y, t).

We use the output of Weiss's method as initial values of our intrinsic image estimation. As mentioned above, the goal of our method is to derive time-dependent reflectance images R(x, y, t) and their corresponding illumination images L(x, y, t). The basic idea of the method is to estimate timevarying reflectance components by canceling the scene texture from Weiss's illumination images. To factor the scene textures out from the illumination images and associate them with reflectance images, we use the texture edges of r_w . We take a straightforward way to remove texture edges from l_w and derive illumination images l(x, y, t) with the following Equation (4) (5).

$$l_n(x, y, t) = \begin{cases} 0 & \text{if} |r_{wn}(x, y)| > T \\ l_{wn}(x, y, t) & \text{otherwise} \end{cases}$$
(4)

$$r_n(x, y, t) = \begin{cases} r_{wn}(x, y) + l_{wn}(x, y, t) & \text{if } |r_{wn}| > T \\ r_{wn}(x, y) & \text{otherwise} \end{cases}$$
(5)

where T represents a threshold value. While we currently manually set the threshold value T used to detect texture edges in r_{wn} , we found the procedure is not so sensitive to the threshold as long as it covers texture edges well. Since the operation is linear, the following equation is immediately confirmed.

$$\begin{aligned} f_n \star i(x, y, t) &= r_{wn}(x, y) + l_{wn}(x, y, t) \\ &= r_n(x, y, t) + l_n(x, y, t) \end{aligned}$$
(6)

Finally, time-varying reflectance images r(x, y, t) and scene texture-free illumination images l(x, y, t) are recovered from filtered reflectance r_n and illumination images l_n through the following deconvolution process, which is the same to Weiss's method.

$$(\hat{r},\hat{l}) = g \star \left(\sum_{n} f_{n}^{r} \star (\hat{r}_{n},\hat{l}_{n})\right)$$
(7)

where f_n^r is the reversed filter of f_n , and g is the filter which satisfies the following equation.

$$g \star \left(\sum_{n} f_{n}^{r} \star f_{n}\right) = \delta \tag{8}$$

To demonstrate the effectiveness of our method for deriving time-dependent intrinsic images, we rendered a CG scene which contains cast shadows and surface patches with different reflectance properties, which is analogous to real road surfaces, e.g. white lines on a pedestrian crossing. Figure 2 shows a side-by-side comparison of the results applying Weiss's method and our method. The first row is the CG scene, where the scene has the property that the histogram of derivative-filtered output is sparse, which is the required property of the ML estimation based decomposition method and also is the statistics usually found in natural images. As can be seen clearly, texture edges are successfully removed from our illumination image while they obviously remain in Weiss's illumination image. Considering an illumination image to be an image which represents the distribution of incident lighting, our illumination image is much better since incident lighting has nothing to do with the scene reflectance properties.

3 Shadow Removal

Using the obtained scene illumination images by our method, the input image sequence can be normalized in terms of illumination.

To estimate the intrinsic images of the scene where video surveillance systems are to be applied, it is necessary to remove moving objects from the input image sequence because our method requires the scene to be static. Therefore we first create background images in each short time range (ΔT) in the input image sequence, assuming that the illumination does not vary in that short time period. We simply use the average image of the short input sequence as the background image, but of course the more complicated methods would give the better



Figure 3: An input image I (left) and the illuminance-invariant image N (right).

background images [10]. These background images B(x, y, t) are used for the estimation of intrinsic images. Using the estimation method described in the former section, each image in the background image sequence is decomposed into corresponding reflectance images R(x, y, t) and illumination images L(x, y, t).

$$B(x, y, t) = R(x, y, t) \cdot L(x, y, t)$$
(9)

Once decomposed into intrinsic images, any image whose illumination condition is captured in the series of B(x, y, t)can be normalized with regards to its illumination condition by simply dividing the input image I(x, y, t) by its corresponding estimated illumination image L(x, y, t). Through the normalization, cast shadows are also removed from the input images.

Since the incident lighting effects are fully captured in illumination images L(x, y, t), the normalization by dividing with L corresponds to removing the incident lighting distribution from the input image sequence. Let us denote the resulting illuminance-invariant image with N(x, y, t), that can be derived by the following equation.

$$N(x, y, t) = I(x, y, t) / L(x, y, t)$$
(10)

Figure 3 shows the result of our normalization method. The left-hand side figure shows the input image I and the right figure represents the illuminance-invariant image N. Note shadows of the buildings are removed in N.

Since we consider the time-dependent reflectance values, accurate normalization of incident lighting can be done compared to using Weiss's illumination images. Figure 4 depicts the difference between using our illumination image l (the left image of each pair) and using Weiss's illumination image l_w (the right image of each pair). As can be seen in the images, white lines appear over the trucks in every right-hand side image. On the other hand, when using our illumination images, the ghost white lines are almost vanished as seen in every left-hand side image. This is because our method handles reflectance variation properly while the reflectance values are fixed in Weiss's method.

4 Illumination eigenspace for direct estimation of illumination images

The intrinsic image estimation method described in the former section is fully off-line because of the computational cost.



Figure 4: The difference of the normalization results between using our illumination images L and Weiss's illumination image L_w . The left images of each pair, (a) and (b), show the results using our illumination image L, and the right ones, (a') and (b'), are the results using Weiss's illumination image L_w .

However, realtime processing is required for practical use. In this section, we describe our approach for realtime derivation of illumination images for shadow removal. Our method first stores a lot of illumination images captured under different illumination conditions. Using stored illumination images, realtime estimation of illumination image from an input image is accomplished.

We propose *illumination eigenspace* to model variations of illumination images of the scene. The illumination eigenspace is an eigenspace into which only illumination effects are transformed. We use PCA to construct the illumination eigenspace of a target scene, in our case, the crossroad shown in Figure 5. PCA is widely used in signal processing, statistics, and neural computing. The basic idea in PCA is to find the basic components $[s_1, s_2, ..., s_n]$ that explain the maximum amount of variance possible by n linearly transformed components. Figure 6 shows the hyper-plane constructed by mapping illumination images onto the eigenspace using all eigenvectors.

In our case, we mapped $L_w(x, y, t)$ into the illumination eigenspace, instead of mapping L(x, y, t). This is because, when given an input image, the reflectance image $R_w(x, y)$ is useful to eliminate the scene texture by computing $I(x, y, t)/R_w(x, y)$, and the resulting image becomes $L_w(x, y, t)$. We keep the mapping between $L_w(x, y, t)$ and L(x, y, t) to derive final L(x, y, t) estimates. First, an illumination space matrix is constructed by subtracting \bar{L}_w , which is the average of all L_w , i.e. $\bar{L}_w = \frac{1}{n} \sum_n L_w$, from each L_w and stacked column-wise.

$$\mathbf{P} = \{ L_{w_1} - \bar{L}_w, L_{w_2} - \bar{L}_w, \dots, L_{w_n} - \bar{L}_w \}$$
(11)

P is a $N \times M$ matrix, where N is the number of pixels in the illumination image and M is the number of illumination images L_w . We made the covariance matrix **Q** of **P** as follows.

$$\mathbf{Q} = \mathbf{P}\mathbf{P}^{\mathrm{T}} \tag{12}$$

Finally, the eigenvectors \mathbf{e}_i and the corresponding eigenvalues λ_i of \mathbf{Q} are determined by solving,

$$\lambda_i \mathbf{e}_i = \mathbf{Q} \mathbf{e}_i. \tag{13}$$



Figure 5: Direct estimation of intrinsic images. Each row shows the different weather condition. (a)An input image I, (b)the pseudo illumination image L_w^* , (c)the estimated illumination image \hat{L}_w by the NN search in the illumination eigenspace, (d)the corresponding background image B to (c).



Figure 6: Illumination eigenspace constructed using 2048 images from 120 days data of a crossroad.

To solve Equation (13), we utilized Turk and Pentland's method [13], which is useful to compute the eigenvectors when the dimension of \mathbf{Q} is high. Figure 6 shows the illumination eigenspace on which all the illumination image, $L_w(x, y, t)$, of 2048 images from 120 days(7:00-15:00) are mapped.

Using the illumination eigenspace, direct estimation of an illumination image can be done given an input image which contains moving objects. We consider that the global similarity of the illumination image is measured by the distance weighted by the contribution ratio of eigenvectors in the illumination eigenspace. Thus, we divide the input image by a reflectance image to get a pseudo illumination image L^* which includes dynamic objects: $L^* = I(x, y, t)/R_w(x, y)$. Using it as a query, the best approximation of the corresponding illumi-

nation image \hat{L} is estimated from the illumination eigenspace.

$$\hat{L}_w = \arg\min_{L_{w_i}} \sum_j w_j \sqrt{\left(\mathcal{F}(\mathbf{L}^*, j) - \mathcal{F}(\mathbf{L}_{\mathbf{w}_i}, j)\right)^2}$$
(14)

where \mathcal{F} is a function which maps an illumination image into the *j*-th eigenvector, and $w_j = \lambda_j / \sum_{\Omega} \lambda_i$ where λ_i s are the eigenvalues. Finally, the illumination image L(x, y, t) is derived using the mapping table from \hat{L}_w to *L*. For the highdimensional nearest neighbor search (NN search), we employed the SR-tree method [11] which is known to be fast especially for high-dimensional and non-uniform data structures such as natural images. The number of stored images for this experiment was 2048 and the contribution ratio was 84.5% at 13 dimensions, 90.0% at 23 dimensions, and 99.0% at 120 dimensions. We chose to use 99.0% of eigenratio for this experiment. The compression ratio was approximately 17:1, and the disk space needed to store the subspace was about 32 MBytes when the image size is 320×243 .

Results of illumination image search is shown in Figure 5. In this figure, starting with the left hand side column, the first column shows input images I, the second column shows pseudo illumination images L^* , the third column corresponds to estimated illumination images \hat{L}_w . The right end column shows the background images which correspond to the estimated illumination images. The NN search in PCA is reasonably robust to estimate the most similar illumination image L_w from the pseudo illumination image L^* . However, since the sampling of the illumination images is sparse, there are slight differences in the shadow shapes. It is possible to acquire the exact illumination image L when the database is dense enough, but it is not easy to prepare such a database. To solve this problem, we propose an approach to compute intermediate illumination images by interpolating shadow regions using geometry estimated from sampled cast shadow regions and sunlight angles. The details are described in Section 4.1.

As for the computational cost, the average time of the NN search is shown in Table 1 with MIPS R12000 300MHz, when the number of stored illumination images is 2048, the image size is 360×243 and the number of output search results is 5. Since the input image is obtained at the interval of 33ms (at 30 frames/sec), the estimation time is fast enough for realtime processing.

Dimension	13	23	48	120
Contribution ratio(%)	84.5	90.0	95.0	99.0
NN Search time(μs)	6.7	6.8	7.9	12.0

Table 1: Dimension of the illumination eigenspace, contribu-tion ratio and NN search cost.

4.1 Shadow interpolation using shadow hulls

Unfortunately, it is difficult to store all illumination images under every possible illumination condition. NN search in the illumination eigenspace gives good results, however, it is often the case that they are slightly different from the true illumination image. Therefore we take an approach to interpolate NN searched results to generate the final estimate of the illumination image. We assume global intensity changes are linear as long as they are densely sampled, but the motion of cast shadows cannot be represented by linear image interpolation. Thus, we create shadow hulls from given shadow regions, that are derived from illumination images, and sunlight angles, computed from time stamps of the input image sequence¹. The resulting hull is not necessarily precise, but it gives *enough* information to compute cast shadow region between sampled illumination conditions.

In our approach, input images are decomposed into intrinsic images first. By thresholding, shadow regions are derived from the illumination images. We assume the intrinsic parameters of the camera is estimated beforehand. Shadows are cast on a plane in the real world and a projection matrix from the image plane to this scene plane can be computed by providing several correspondences between the two planes. We do this manually. Shadow regions are then mapped onto the world coordinate, and shadow volumes are computed using shadow regions associated with sunlight angles. By taking the intersection of shadow volumes in the 3D space, we get the rough geometry of objects casting shadows, which has enough information for computing intermediate cast shadows (Figure 7 (a)). Figure 7 (b) shows the results of shadow interpolation in a CG scene using an estimated shadow hull. The dark regions show the interpolated shadow regions, while the lighter regions represent the sampled shadow regions.

Shadow interpolation using shadow hulls is useful to estimate the intermediate shadow shapes between sampled lighting conditions. Figure 8 shows the interpolated result of the real world scene. The left-hand column represents the estimated shadow regions, while the right-hand column shows



(a) Computing shadow hulls using shadow regions associated with sunlight angles.



(b) Result of shadow interpolation Figure 7: Interpolation of cast shadow using a shadow hull.



Figure 8: Shadow hull based shadow interpolation. Figures in top and bottom row are shadow regions and sampled illumination images. The middle row shows the interpolated results. The grid is overlaid for better visualization.

¹Sunlight angles can be computed precisely provided the latitude and longitude of the scene and the date and time.



Figure 9: Comparison with the ground truth. (a)interpolated result using our method, (b) the ground truth, (c) result of image differencing between (a) and (b).

the corresponding illumination images. The top and bottom row represent the images under sampled illumination conditions, and the middle row depicts the interpolated result. To obtain the intermediate illumination image, we first estimate shadow boundaries using the estimated shadow hull. Once we obtain intermediate shadow boundaries, we then compute the intermediate illumination image $L_{int}(x, y)$ with the following equation.

$$L_{int}(x,y) = \begin{cases} \delta_{int}(x,y) \frac{\sum_{k} w_k \delta_k(x,y) L_k(x,y)}{\sum_{k} w_k \delta_k(x,y)} \\ \bar{\delta}_{int}(x,y) \frac{\sum_{k} w_k \bar{\delta}_k(x,y) L_k(x,y)}{\sum_{k} w_k \bar{\delta}_k(x,y)} \end{cases}$$
(15)

where L_k is the k-th illumination image obtained by NN search in the illumination eigenspace, and w_k is the weighting factor which is the distance from L_w^* to L_{w_k} (See Section 4) in the illumination eigenspace. $\delta_k(x, y)$ is the function which returns 1 if $L_k(x, y)$ is inside the shadow region, otherwise returns 0. $\overline{\delta}_k(x, y)$ is the inverse of $\delta_k(x, y)$. The resulting intermediate image is shown in mid-right in Figure 8. It is more evident by comparing the resulting image with the ground truth. Figure 9 shows the comparison between the result of our method (a), the ground truth (b) and the difference between the estimate and ground truth (c). We can notice the slight difference between them from (c), however, it shows a globally correct shadow shape which is useful to remove shadow effects from the input image.

5 Experimental Results

Evaluation of our shadow removal method is based on a basic object tracking algorithm. Our goal is to obtain more accurate tracking results from preprocessed image sequences, i.e. normalized image sequences using our method, than from original image sequences. We chose the block matching algorithm for actual tracking to show even the simplest and widely used tracking method can achieve good results after utilizing our illumination normalization preprocess. The block matching based tracking is accomplished by pursuing the most similar window in the neighboring frame evaluated by Equation (16).

$$e_B(x,y) = \min_{i,j} \left\{ \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |f_t(x+m,y+n) - f_{t-1}(x+m+i,y+n+j)| \right\}$$
(16)

For the experiments, we chose image sequences containing frames of vehicles crossing boundaries of cast shadows since we focus especially on the advantage of our shadow elimination. The parameter set of vehicle tracking for each sequence is totally even, i.e. the same initial window position, 10×10 pixels of window, 10 pixels for maximal search distance, and 10 fps of frame-rate.

The result is shown in Figure 10. In Figure 10, the time axis is represented from top to bottom. The first column of each pair and the second column of that represent results of the block matching based tracking applied to the original image sequence and image sequence with our preprocessing of illuminance normalization, respectively. In the original image sequence (a),(b),(c), we get typical results where the block matching fails at shadow boundaries, because of the large intensity variation between inside and outside the shadow. On the other hand, after proper illumination normalization using our method, we get successful results.

We accomplished the tracking experiments over 502 vehicles in 11 sequences under different lighting conditions. Since we cannot have the ground truth, we carefully assessed the tracking results. The success rate using the original input image sequences was 55.6%, while with normalized input it improved to 69.3%. The effectiveness of our method is clearly confirmed by the result that 45.3 % of originally failed results were rescued by our method. On the other hand, 11.5% got worse after applying our method. This poor effect happens typically when the shadow-edge cast on the vehicle surface largely differs from the shadow-edge in the illumination image. It happens because our method currently can handle only



Figure 10: Results of tracking based on block matching. Along row from top to bottom it shows the frame sequence. The first column of each pair, (a),(b),(c), shows the tracking result over the original image sequence, and the second column of each pair, (a'),(b'),(c'), shows the corresponding result after our preprocessing.

two-dimensional shadows on the image plane, but the actual shadow is cast three-dimensionally on the scene. When the gap of the shadow-edge position is large, the error of the normalization gets large, as a result, the block matching fails. Our system currently does not handle this problem since the error rate is small compared to the improved correct rate, but we are investigating on handling cast shadows 3-dimensionally.

6 Conclusions

We have described a framework for normalizing illumination effects of real world scenes, which can be effectively used as a preprocess for robust video surveillance. We believe it provides a firm basis to improve the existing monitoring systems. We started from Weiss's method to derive intrinsic images from image sequences, and extended the method to properly handle surfaces with non rigid reflectance properties. This is accomplished by modifying pseudo derivative illumination images with regards to the scene texture edges that can be derived from the pseudo reflectance image estimated through Weiss's algorithm. The proposed method is remarkably robust and does not require the information of scene geometry, camera parameter and lighting condition at all, but requires the camera to be fixed and several lighting conditions to be observed. As a key component of our framework, we proposed to utilize *illumination eigenspace*, a pre-constructed database which captures the illumination variation of the target scene, to directly estimate illumination images for elimination of lighting effects of the scene, including elimination of cast shadows. As for the intermediate illumination images, that cannot be represented by linear combinations of sampled illumination images, we propose an approach to use shadow hulls to interpolate cast shadow regions using sunlight angles and camera parameters. The effectiveness of the proposed method is confirmed by comparing the tracking results between the original image sequence and the image sequence with our method used as a preprocess. Since our method is used as a preprocessing stage, we believe this method can be applied to many video surveillance systems to increase the robustness against lighting variations. Also, we have investigated direct estimation of

illumination images corresponding to real scene images using the illumination eigenspace and shadow interpolation based on shadow hulls. Though our current implementation of the direct estimation in research code is not fast enough for realtime processing, we believe the framework has the potential to be processed in realtime.

References

- Y. Weiss, "Deriving intrinsic images from image sequences", In Proc. of 9th IEEE Int'l Conf. on Computer Vision, pp. 68–75, Jul., 2001.
- [2] H.G. Barrow and J.M. Tenenbaum, "Recovering intrinsic scene characteristics from images.", In A. Hanson and E. Riseman, editors, Computer Vision Systems. Academic Press, pp. 3–26, 1978
- [3] J. Huang and D. Mumford, "Statistics of natural images and models", In Proc. Conf. Computer Vision and Pattern Recognition, pp. 541–547, 1999.
- [4] E.H. Adelson, and A.P. Pentland, "The Perception of Shading and Reflectance," In D. Knill and W. Richards (eds.), Perception as Bayesian Inference, pp. 409–423, 1996.
- [5] E.H. Land, "The Retinex theory of color vision". Scientific American, 237(G), No. 6, pp. 108–128, Dec. 1977.
- [6] M. Kilger, "A shadow handler in a video-based real-time traffic monitoring system," In *Proc. of IEEE Workshop on Applications of Computer Vision*, pp. 11–18, 1992.
- [7] D. Koller, K. Daniilidis, and H.H. Nagel, "Model-based object tracking in monocular image sequences of road traffic scenes," In *Int'l Journal* of Computer Vision, vol. 10, pp. 257–281, 1993.
- [8] J. Stauder, R. Mech, and J. Ostermann, "Detection of moving cast shadows for object segmentation," IEEE Transactions on Multimedia, vol. 1, no. 1, pp. 65–76, Mar. 1999.
- [9] C. Jiang and M.O. Ward, "Shadow identification,", In Proc. of IEEE Int'l Conf.on Computer Vision and Pattern Recog., pp. 606–612, 1992.
- [10] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, "Wallflower: Principles and Practice of Background Maintenance", In *Proc. of Int'l Conf. on Computer Vision*, pp. 255–261, 1999.
- [11] N. Katayama and S. Satoh, "The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries,", In Proc. of the 1997 ACM SIGMOD International Conf. on Management of Data, pp. 369–380, 1997.
- [12] S.K. Nayar, H. Murase, and S.A. Nene, "Parametric Appearance Representation", In S.K. Nayar and T. Poggio, editors, Early Visual Learning, pp. 131–60, 1996.
- [13] M. Turk and A. Pentland, "Eigenfaces for recognition", The Journal of Cognitive Neuroscience, 3(1): pp. 71–86, 1991.
- [14] G.D. Finlayson, S.D. Hordley and M.S. Drew, "Removing Shadows from Images", In Proc. of European Conf. on Computer Vision Vol.4, pp. 823–836, 2002.