SHADOW REMOVAL WITH BLOB-BASED MORPHOLOGICAL RECONSTRUCTION FOR ERROR CORRECTION

Li-Qun Xu[†], *José Luis Landabaso*[‡], *Montse Pardàs*[‡]

† BT Research & Venturing, UK ‡ Technical University of Catalunya (UPC)

ABSTRACT

Dealing with shadows and highlights is essential in object detection and tracking applications such as automated video surveillance systems. This is especially true for outdoor scenarios subject to variable lighting and weather conditions. In this paper, we present a novel scheme for effective shadows (highlights) detection using both color and texture cues. Since in any such algorithm, misclassifications often occur, resulting in distorted object shapes, the core of this scheme is the introduction of a technique capable of correcting these errors. The technique is based on morphological reconstruction of the shadow-removed blobs conditioned on the blobs prior to a shadow-removal process, assuming that the object shapes are properly defined along most part of their contours after the initial detection. Experiments on variety real-world video data demonstrate the favorable performance and robustness of the proposed scheme.

1. INTRODUCTION

One of the fundamental challenges in computer vision for accurate object detection and tracking is to achieve invariance to illumination changes, and more prominently, to shadows and highlights. The two types of shadows that should be treated differently are:

- Cast-shadows refer to areas in the background projected by objects in the direction of light rays, producing distorted objects silhouettes. (see e.g., Figure 3a)
- Self-shadows are parts of an object not illuminated. A good shadow removal scheme must not remove them, as they are part of the silhouette.

As for highlights they are areas of exceptional lightness in an image. Cluttered scenes in the background, e.g., trees, should not be detected as new objects when being directly shone by sun lights in cloudy days, for instance.

Usually, shadows and highlights detection algorithms form part of more general object tracking systems. These object tracking systems often first segment incoming images into foreground and background representations by means of different background learning techniques. In these techniques, probabilistic adaptive models are created for each pixel to classify incoming image pixels into foreground or background. Afterwards, a connected component analysis (CCA) [5] is usually employed to isolate meaningful blobs from individual foreground pixels. For each blob some representative features can be extracted to describe its spatial-temporal properties. Finally, there is a blob-based feature matching process in order to find persistent blob correspondences between consecutive frames. An example of object tracking systems can be found in [6].

Shadow removal algorithms are usually incorporated in the background subtraction/modeling step. Several studies have been carried out to extract cues from the background reference images/models and use them to identify if a pixel is a cast shadow/highlight pixel or not. Prati *et al.* have presented an in-depth survey of these algorithms [4].

There are two main sets of works that incorporate these extracted cues, including the use of color (texture) information to find chrominance (texture) similarities between the background representation and the incoming frame. And a combination of the two is still an open issue. But even combining these two approaches, shadow removal algorithms tend to be somewhat noisy and often misclassify foreground pixels. In order to correct these errors we propose to use images prior to the shadow-removal process where shapes are still well defined to assist blob reconstruction.

The paper is organized as follows. Section 2 describes the techniques for pixel-domain analysis, leading to the segmented foreground object blobs. Section 3 discusses issues concerning color and texture-based shadows detection, whereas a combination of the two is explained in Section 4, along with the proposal of a novel morphological foreground reconstruction technique. Section 5 gives some experimental results. The paper concludes in Section 6.

2. LEARNING THE BACKGROUND

Background learning techniques are very useful to achieve accurate and robust foreground objects segmentation in a dynamic scene. There are techniques in which an explicit reference image is first generated to be used in the 'background subtraction' process. Whereas, new approaches perform a classification process based on a pixel-wise probabilistic model, thus avoiding any explicit subtraction step.

The Stauffer and Grimson (S&G) [1] algorithm has become a reference in the area of probabilistic classification of background and foreground. In the following, we first outline this technique, and then explain the necessary steps to take to suppress falsely detected foreground pixels and extract a reference image prior to handling cast shadows and highlights removal.

2.1 The Stauffer and Grimson algorithm

The main idea of S&G algorithm is to model the photometric variations of each pixel along the time course by a mixture of K Gaussian distributions. Different Gaussians are assumed to characterize different color appearances in each pixel, and each Gaussian is weighted (w) depending on how often the Gaussian has explained the same appearance. Using multiple Gaussians ensures that repetitive moving background as in tree leaves can be represented by different probabilistic functions.

An incoming pixel is considered to be explained by a Gaussian distribution if its color value is within say 2.5 standard deviations of the distribution mean. Basically, this is the same as in any clustering process.

Then, every time a Gaussian explains an incoming pixel, its variance (σ^2) and mean (μ) are updated as in (1).

$$\begin{aligned} \mu_t &= (1-\rho)\mu_{t-1} + \rho X_t \\ \sigma_t^2 &= (1-\rho)\sigma_{t-1}^2 + \rho (X_t - \mu_{t-1})^T (X_t - \mu_{t-1}) \end{aligned} \tag{1}$$

where ρ is the Gaussian adaptation learning rate.

By updating the mean and variance, the system is allowed to adapt to slow illumination changes. The weight w_t associated to each Gaussian is also updated depending on if the Gaussian explains the incoming pixel or not as in (2).

$$\begin{aligned} w_t &= w_{t-1} + \alpha(1 - w_{t-1}) && \textit{matched} \\ w_t &= (1-\alpha) w_{t-1} && \textit{non-matched} \end{aligned} \tag{2}$$

α being the weight learning rate.

Thus, the more often a Gaussian explains an incoming pixel, the higher is its associated weight.

In order to classify an incoming pixel as being part of the foreground or background, the Gaussians of each pixel are reordered according to w/σ in descending order. The first few in the list most likely represent the background as the background is often very static (low variance) and appears most of the time (high weight w). Analogously, the incoming foreground pixels correspond to the last Gaussians in the list.

This can be stated as follows: When a pixel matches any of the first B Gaussians decided by (3), it is classified as a background pixel, otherwise, a foreground pixel.

$$B = \arg\min \left(\sum_{k=1}^{b} w_k > T \right) \tag{3}$$

2.2 Suppression of falsely detected foreground pixels

The S&G background learning is very robust, though there remain classification errors due to the noise manifested in the images. On certain occasions, some background points fail to match their Gaussian and are classified as foreground. Research has been carried out to overcome this well-known problem [2]. Although typical post-processing techniques often depend on the background learning technique employed, a more general approach using local neighborhood information is introduced here. The proposal is that, when a pixel is classified as foreground, it is again examined by its 3x3 spatial neighboring pixel models. If 5 or more models agree on that it's a background pixel, then it's considered as a false detection. By means of this simple rule many small errors are automatically corrected and system operation is more robust.

2.3 Extracting a background reference image

Since the classification of foreground pixels in the scene is directly performed on incoming images, so far an explicit background reference image is not required. However, the needs arise in shadow removal techniques where the properties of the shadowed regions and the corresponding background are to be examined in conjunction.

For such purpose, a simple procedure is used to extract an adaptive background image as follows: The pixel colors in the background image assume those of the incoming image if they are classified as background. In the case that the incoming pixels have been classified as foreground, then the mean of the Gaussian distribution with the largest weight and lowest variance (the most probable background color in the pixel) is chosen as the background pixel color.

3. COLOR- & TEXTURE-BASED SHADOW DETECTION

A shadow is normally an area that is not or only partially irradiated or illuminated because of the interception of radiation by an opaque object between the area and the source of radiation. Assuming that the irradiation consists only of white light, the chromaticity in a shadowed region should be the same as when it is directly illuminated. The same also applies to lightened areas in the image. Based on the same assumption, a normalized chromatic color space, e.g., $r = R/(R+G+B)$, $g = G/(R+G+B)$, is immune to shadows, but the lightness information is unfortunately lost. Keeping it is important in order to avoid some simple errors such as confusing a white car with a grey road.

Another important issue is that we are only interested in detecting shadows that form part of the foreground objects. Shadows that form part of the background are not a problem as they don't have to be tracked. Specifically, a shadow removal algorithm needs to analyze foreground pixels and detect those that have similar chromaticity but lower brightness to the corresponding region when it is directly illuminated. The adaptive background reference image provides the desired information.

3.1 Color-based detection

In view of the fact that both brightness and chromaticity are very important, a good distortion measure between foreground and background pixels should account for the discrepancies in both their brightness and chromaticity components as in [3]. Brightness distortion (BD) can be defined as a scalar value that brings expected background close to the observed chromaticity line. Similarly, color distortion (CD) can be defined as the orthogonal distance between the expected color and the observed chromaticity line. Both measures are shown in Figure 1 and formulated in (4).

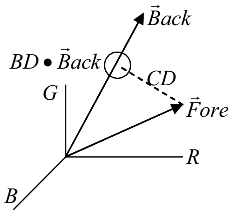

Figure 1. Distortion measurements in the RGB color space: $\vec{Fore}$ denotes the RGB value of a pixel in the incoming frame that has been classified as foreground. $\vec{Back}$ is that of its counterpart in the background.

Brightness distortion values over 1.0 correspond to lighter foreground. On the other hand, the foreground is darker when BD is below 1.0.

$$BD = \arg\min_{\alpha}\left(\vec{Fore} - \alpha\vec{Back}\right)^2 \qquad (4)$$
$$CD = \left\|\vec{Fore} - \alpha\vec{Back}\right\|$$

BD can be easily computed as $BD = \vec{Fore} \bullet \vec{Back} / \vec{Back}^2$.

Finally, a set of thresholds need to be defined to assist the classification into foreground, highlight or shadow pixel, as shown in Table 1.

Table 1. Thresholds for shadow and highlight detection.

If CD *< 10.0 then:*
If 0.5 < BD *< 1.0 then SHADOW*
Else if 1.0 < BD *< 1.25 then HIGHLIGHT*
Else FOREGROUND

Note that this technique fulfils its objective not to remove self-shadows as they do not share similar brightness or chromaticity with their background reference image.

Note also that it is still possible to achieve more precise results by normalizing variations in color bands at the expense of increased computational cost. Also, many other approaches, e.g., [2], are based on the same underlying idea of decomposing color and brightness. Our reconstruction process to be described in Section 4 does not rely on any particular implementation, so any approach can be used.

3.2 Texture-based detection

The same regions with or without cast shadows should have the same texture properties. Similar to the color-based shadow removal procedure, a texture distortion measure can be defined to detect possible foreground shadow pixels.

A simple way of computing the texture is to use the first-order spatial derivatives, though other more sophisticated measures can also be employed. We apply Sobel filters to both the background and incoming frame and then compute the Euclidean distance between them. If this distance is lower than a certain threshold, *i.e.* very similar texture, then the pixels are probably part of a shadow region.

4. HYBRID SHADOW REMOVAL

The color- and texture-based shadow removal techniques suffer from weaknesses of their own. The color-based algorithm generates errors when the underlying assumptions are violated, meaning that foreground objects having similar colors to that of the shadowed background regions may be wrongly diagnosed and removed. Similarly with the texture based approach, the foreground regions having similar textures to that of their corresponding background may also be deleted by mistake.

In our approach, both the aforementioned color and texture-based procedures are used in parallel, followed by an assertion process that combines the results of the two, *i.e.*, a pixel is confirmed as shadow if and only if the results of the two approaches agree. This process paves the way for the proposed object shape reconstruction process.

4.1 Foreground reconstruction

The cast shadows/highlights removal algorithm is a destructive process in the sense that, despite the assertion process described above, original object shapes are likely distorted and some pixels will remain misclassified. Mathematical morphology theory can be employed in order to reconstruct the original image without cast shadows or highlights.

Mathematical morphology reconstruction filter uses an image called “marker” image as a mark to rebuild an object inside an original image called “mask” image. In our case the “marker” image (Figure 3c) is a binary image where a “1” pixel corresponds to a true foreground. On the other hand, the “mask” image (Figure 3b) is also a binary image where a “1” pixel can correspond to a foreground, or cast shadow/highlight pixel, or speckle noise.

It is highly desirable that the “marker” image, $\widetilde{M}$, contains only true object pixels, not any shadows/highlights so that those regions will not be reconstructed. Therefore, the use of very aggressive thresholds is necessary in the foregoing color-based removal process to ensure that all the shadow/highlight pixels are removed. A speckle noise removal filter, shown in (5), is also applied to suppress remaining isolated noisy foreground pixels and obtain a good quality “marker” image, $\widetilde{M}$.

$$\widetilde{\mathrm{M}} = \mathrm{M} \cap (\mathrm{M} \oplus \mathrm{N}) \qquad (5)$$

where M is the binary image generated after shadow removal and assertion process; N denotes the structuring element in Figure 2 with the origin at the centre:

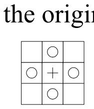

Figure 2. The 3x3 morphological structuring element used for speckles filtering.

The dilation operation $M \oplus N$ in (5) identifies all the pixels that are four-connected to (*i.e.* next to) a pixel of M. Hence, $\widetilde{M}$ identifies all the pixels that are in M and also have a four-connected neighbor, thus eliminating the isolated points in M.

(a)
(b)
(c)
(d)

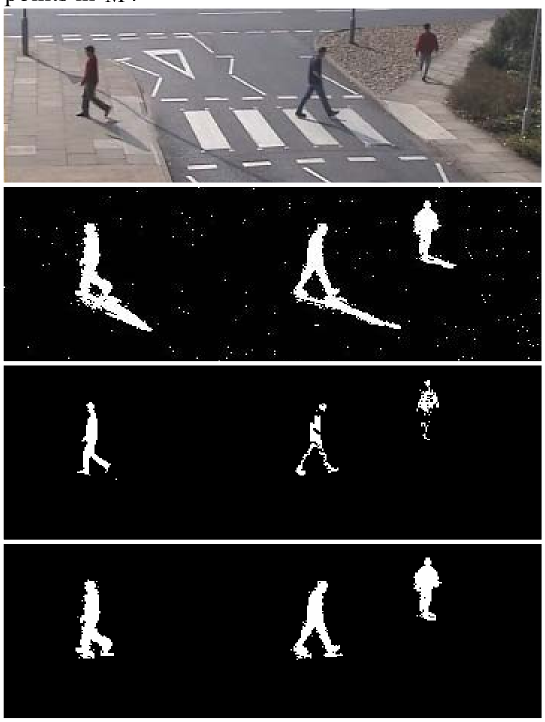

Figure 3. Illustration of the shape reconstruction process for the foreground regions. (a) the incoming image; (b) the "mask" image from initial segmentation; (c) the "marker" image after shadows /highlights removal, and (d) the final reconstructed objects shapes.

As a result, only the regions not affected by noise that are clearly free from shadows/highlights (Figure 3c) are subject to the shape reconstruction process shown in (6):

$$R = M_s \cap (\widetilde{M} \oplus SE) \tag{6}$$

where M_s is the mask, $\widetilde{M}$ the marker and SE the structuring element whose size usually depends on the size of the objects of interest, though a 9×9 square element proved to work well in all our tests. Basically this process consists of a dilation of the "marker" image, followed by the intersection with the "mask" image. The underlying idea is that the shadow removed blobs keep at least a number of points that have been robust to erroneous shadow removal. These robust points are appropriate for leading the reconstruction of neighboring points as long as they form part of the silhouette in the original blob (Figure 3b). The fully reconstructed blobs are shown in Figure 3d.

5. EXPERIMENTAL RESULTS

The algorithm performs well in our experiments on various outdoors scenarios and recordings except for very large cast shadows where sometimes they are not completely removed. This is mainly due to the fact that brightness decreases below the BD threshold. The problem can be corrected using lower thresholds in the BD with the drawback of introducing false shadow pixel detection. An example results is shown in Figure 3 on a real world scenario. A small defect is noted that the reconstructed image contains a segment of shadows in the objects exteriors where the cast shadow starts (see the feet of the persons in Figure 3d). This segment has 1/2 size of the structuring element used, and is produced during the dilation. Intersection with the mask image cannot suppress the segment as all the shadow regions form part of the mask.

Finally, this novel scheme has been incorporated in our object tracking system, which has been evaluated broadly using the publicly available benchmarking video sequences PETS 2001 and our own recordings. The sequences contain persons, groups of people and vehicles. Some results can be found at: http://gps-tsc.upc.es/imatge/_jl/Tracking.html.

6. CONCLUSION

We have presented in this paper a novel scheme for effective shadows and highlights detection, which has been successfully incorporated in an object tracking system. The scheme exploits information from both color and texture cues between an incoming image and an adaptive background reference, and performs an error correction procedure to recover original object shapes using conditional morphological reconstruction process. Experiments have demonstrated favorable results on various real-world scenes on both raw and compressed image sequences. Some of the future works include using region-based instead of pixel-based domain processing in both the texture and color-based shadow detection as well explore heuristics to inhibit the reconstructions of minor shadows in objects exteriors.

ACKNOWLEDGMENTS

This work was partially supported by the EU Framework V SCHEMA project (IST-2000-32795). JLL also acknowledges the support of a BT internship and the bursary from Spanish national project grant number TIC2001-0996.

REFERENCES

[1] C. Stauffer, W.E.L. Grimson, "Learning Patterns of Activity Using Real-Time Tracking", *IEEE Trans. on PAMI*, **22**(8), 2000.

[2] A. Elgamal, R. Duraiswami, D. Harwood, L. Davis, "Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance," *Proc. of the IEEE*, **90**(7), 2002.

[3] T. Horpraset, D. Harwood and L. Davis, "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection," *Proc. of IEEE ICCV'99 FRAME-RATE Workshop*.

[4] A. Prati, I. Mikic, M.M. Trivedi and R. Cucchiara, "Detecting Moving Shadows: Algorithms and Evaluation", *IEEE Trans. on Pattern Recognition and Machine Intelligence*, **25**(7), 2004.

[5] K.P. Horn, **Robot Vision**, pp. 299-333, MIT Press, 1986.

[6] O. Javed and M. Shah, "Tracking and object classification for automated surveillance," *Proc. of ECCV'2002*, 2002.