Cast Shadow Identification and Image Restoration by Clustering Technique

Ambrose E. Ononye[1] *and Philip W. Smith*[2]
[1]*Center for Imaging Science, Rochester Institute of Technology*
Rochester, NY 14623, USA
email: ononye@cis.rit.edu
[2]*Department of Electrical and Computer Engineering, University of Tennessee*
Knoxville, TN 37996, USA

Abstract

In this paper, we present a novel approach for identifying and removing cast shadow in a color image. The technique employs clustering and color normalization procedures without the usual assumption that the darkest region constitutes a shadow or requiring the camera to be linear. The input image is transformed to a feature space spanned by the R-,G-,B- colors and the Mean Shift Algorithm used for clustering. The number of such clusters denotes the number of significant distinct color regions including the shadow in the image. Color normalization has the tendency to remove shadow if a linear color space exists but does not if the space is non-linear.

Using normalized color and Euclidean distance measure constraint, pairs of closest clusters called *shadow candidate pairs* are formed. Any shadow candidate pair whose distance apart is greater than the constraint is discarded as an invalid pair and vice-versa. The darker of the valid pair is regarded as shadow and can then be extracted. Our technique is also able to recover the background that is partly in shadow and partly illuminated.

Image restoration is done by a mapping process whereby pixels that are resident in the shadow cluster are mapped to the mean of the ones they are closest to.

We present some results using real color images with shadows on single- and multi-color backgrounds captured with a Kodak Zoom 120DC camera.

1. Introduction

From theory of classical physics, shadows are formed when there is an obstruction along the direction of light propagation. For an extended light source, a shadow consists of two regions, the darker called the umbra and a slowly darkening region, the penumbra which surrounds the former. Shadows may be classified as *self* or *cast*. A self shadow is formed on the object while the cast shadow is formed outside the object (on the background).

It is often difficult to recognize shadow in a scene by an artificial vision system. Due to this difficulty, some previous research efforts have employed a number of cues that suggest the existence of shadow for its identification. The four most prominent cues are summarized below.

1. **Darkest Region Gambit** - the image region with the lowest average intensity is likely to represent pixels in shadow.
2. **Hue/Saturation Invariance** - the apparent hue and saturation of surfaces in and out of shadows remains constant.
3. **Illuminant Direction Dependence** - the shape, size, and position of shadows are directly dependent on scene illuminant directions.
4. **Surface Texture Invariance** - cues for recovering texture that do not depend on absolute intensity are preserved across shadow boundaries [1].

Due to the nature of imaging sensors, it is often safe to assume that the darkest regions of images are cast shadows. Thus, a number of approaches, primarily developed for gray-scale imagery, assume shadows are located in the areas of the image with the lowest intensities [2, 3], an assumption that does not always hold.

The hue/saturation invariance property has been employed in various research efforts [2, 3, 4, 5] to eliminate shadows in scenes. Given an image pixel on a constant color surface outside the shadow, $\vec{E}_o(x, y)$, the invariance property states that any pixel on the same surface inside the shadow, $\vec{E}_s(x, y)$, is such that

$$\vec{E}_s(x, y) = \alpha\vec{E}_o(x, y)$$

where α is a constant such that $\alpha < 1$. Thus, shadows can be eliminated from the image via normalization. Unfortunately, image normalization removes apparent color changes that arise from surface shape variation as well. Furthermore, techniques based on normalization provide only for removal, not identification, of pixels in shadow.

In his work on visual recognition of shadows, Funka-Lee [6, 7] used an active observer equipped with an extendible probe for casting its own shadows on the scene. This allowed the observer to experimentally determine the number, location, and spectral content of the light sources present. The hue/saturation invariance property was then used to identify shadows in a unique way. Constrained by the information obtained regarding the illumination environment and assuming a sensor with linear gain, the color space of the image is searched for radial line features that result from the slowly darkening nature of shadow pixels in the penumbra. Pixels which fall along these lines are assumed to be in shadows. While this approach works well in both controlled and uncontrolled environments, many imaging sensors, including most off-the-shelf color cameras, have a non-linear gain component that distorts the expected radial lines, creating radial curves which are often undetected.

A new type of invariance property called the c_1, c_2, c_3 model was utilized by Salvador et al [8]. Although their model works fine, it was not extended to shadows on multi-color backgrounds. Barnard and Finlayson [9] used color ratios technique to identify shadows. These existing techniques did not address the issue of image restoration.

We now present a novel algorithm for cast shadow identification and image restoration. Employing a simple variation on the hue/saturation invariance based method described above, this technique is able to identify and remove shadows from color images acquired using non-linear gain sensors without employing the often unrealistic darkest region assumption. This paper focuses on cast shadows and does not address self shadows in color imagery. We assume as in [8, 10] that the background is non-textured.

2. Shadow Detection Algorithm Assuming a Linear Gain Camera

In this section, we present a new technique which uses the hue/saturation properties of images with shadows, along with color-based image clustering/segmentation, to not only remove, but also identify image pixels in shadow regions.

The input image is transformed to a feature space spanned by the R-,G-,B- colors using the Mean Shift Algorithm (MSA) [11] for clustering. Pixels with similar RGB distributions cluster together. The number of such clusters denotes the number of significant distinct color regions including the shadow(s) in the image domain. Since under typical illumination conditions, the normalized color of two points on any given surface of the same material will be the same even if one of these points is directly illuminated and the other is in a shadow. For our case, the normalized color features, rgb used are defined by:

$$r = \frac{R}{\|R+G+B\|}$$
$$g = \frac{G}{\|R+G+B\|}$$
$$b = \frac{B}{\|R+G+B\|} \tag{1}$$

The input image is normalized (in the sense of eq(1)) and the same clustering algorithm applied to both normalized and unnormalized images. Let m_u and m_n be the number of clusters corresponding to the unnormalized and normalized images respectively. If $m_u = m_n$, it suggests there is no shadow in the scene. If m_n is less than m_u, then the difference, m_u - m_n shadow clusters have vanished as a result of the process. The shadow clusters have consequently merged with other color clusters with identical normalized color. To detect the shadow cluster from any merged pair, we examine the RGB distribution of their means. The one with a lower RGB values (for unnormalized clusters) belongs to the shadow (see figure (1)). In figure (1), (a) is a synthetic input image with shadow and (b1, b2) are the normalized images. The cluster centers corresponding to (a) and (b) are respectively shown in (e) and (f). The cluster center or mean cluster labeled (3) corresponds to the shadow and is completely eliminated after normalization. The enhanced image of (b1) is shown in (b2) while the restored image is shown in (d).

In section 3, we will discuss the case for non-linear color space.

3. Shadow Detection in a Non-Linear Color Space

If the color space is not linear, the shadow detection technique described above will not work well. In this case, the color space and of course the camera will have to be linearized by doing some gamma correction. This correction may not completely eliminate shadow after normalization (see figures (3 and 6)b). The procedure in the previous section will therefore have to be modified. Instead of searching for vanished clusters, we find the normalized clusters that are closest to each other and form pairs of such clusters, called the *shadow candidate pairs*. The formation of these pairs may result to some invalid ones which can be eliminated by a user defined Euclidean distance measure constraint, ϵ. Clusters are said to have merged if ϵ is close to being zero. Hence, any shadow candidate pair whose Euclidean distance measure is less than ϵ is regarded as a valid pair.

For shadow on a multi-color background such as the one illustrated in figure (2) below, the same clustering tech-

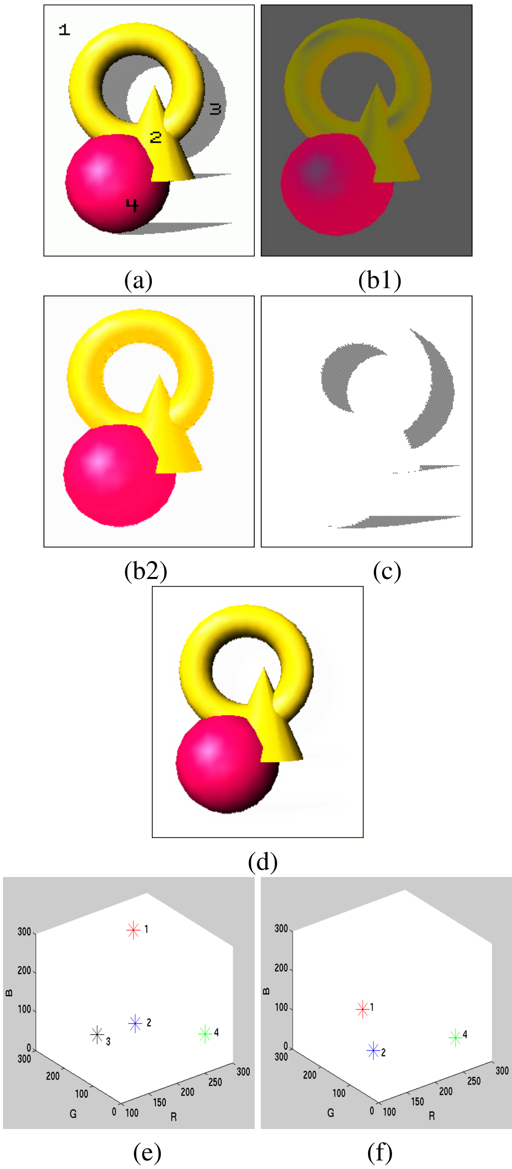


Figure 1: *Shadow extraction and removal procedure: (a) is synthetic image with shadow, (b1) is the result of the normalization and (b2) is the post-processed image of (b1). (c) is the extracted shadow and (d) is the restored image. (e and f) show the cluster centers corresponding to images in (a and b1) above. The mean cluster labeled 3 corresponds to the shadow. (f) shows the clusters corresponding to (b2). The shadow is gone after normalization*

nique is applicable. After normalization, the shadow cluster corresponding to **S1** will be paired with the background labeled **B1**. Similarly, **S2** will be paired with background **B2**. Like in the case of a single-color background, false shadow candidate pairs will be eliminated by a carefully selected Euclidean distance measure constraint. For any valid shadow candidate pair with clusters, $\mathbf{C}_1$ and $\mathbf{C}_2$, the darker of the two is the shadow and by keeping track of the pixels that reside in such cluster, the shadow image could be extracted.

Results of the shadow detection and removal are shown in figures 3 and 6 through 8 below. Figure 8 illustrates the case of extraction of shadow on multi-color backgrounds.

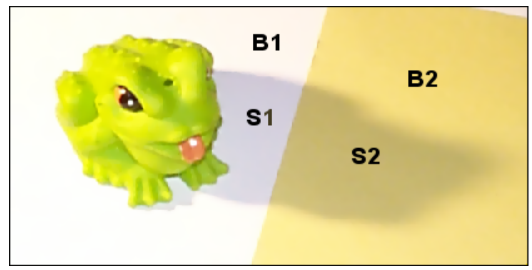


Figure 2: *Illustration of shadow cast on multi-color background.*

4. Image Restoration

We have so far discussed how to recognize and extract shadows. The main issue to be addressed is how does one restore the given image? This issue has not been well addressed in literature. To address the above issue, we consider a pair of merging clusters $\mathbf{C}_1$ and $\mathbf{C}_2$ say. If $\mathbf{C}_2$ is the darker of the two, it is therefore regarded as shadow. The corresponding pixels that are in $\mathbf{C}_2$ are then mapped to the mean of $\mathbf{C}_1$. This type of mapping results to a noticeable penumbra edge effect as shown in figures (3e and 6e). The reason for this phenomenon is that real illuminants do not cast sharp shadows because they are not point sources and may only be partially obstructed. The umbra (which is the darker region of a shadow is due to complete obstruction of the light source) and the penumbra (which results from partial obstruction of illuminant) regions are shown in figure (4) as **DE** and **EG** respectively. Consequently, the transition from shadow to non-shadow is not a step function but a slowly varying function as shown in the figure. In this figure, $\mathbf{M}_1$ and $\mathbf{M}_2$ are the means of the shadow and non-shadow clusters respectively, that merged together. The decision boundary **PQ** tends to equalize the distance between the two means. After the preliminary mapping of shadow pixels to $\mathbf{M}_2$, the region to the left of the decision boundary, **PQ** gets mapped to $\mathbf{M}_2$ leaving the partial penumbra region, **FG** unmapped as illustrated in figure (5). At the final phase of the shadow removal, the pixels in the partial penumbra region **FG** are then mapped to the mean, $\mathbf{M}_2$. This eliminates the edge effect as shown in figure (3f and 6f).

5. Algorithm for Shadow Detection and Removal

The following algorithm is used to detect and remove cast shadow(s) in color images.

1. Pre-process the given image by smoothening.
2. Cluster the image and determine the number of clusters. Call this m_u.
3. Normalize the pre-processed input image and cluster. Let the number of clusters in this case be m_n.
4. If $m_u > m_n$, determine the clusters that have merged and go to step 8.
5. If $m_u = m_n$ (case of non-linear color space), form pairs of normalized clusters that are closest to one another (*the shadow candidate pairs*). Call them $\mathbf{C}_i$ and $\mathbf{C}_j$ with means $\mathbf{M}_i$ and $\mathbf{M}_j$ respectively.
6. If the distance between $\mathbf{M}_i$ and $\mathbf{M}_j > \epsilon$, stop - there is no shadow else go to 7.
7. Extract the darker of $\mathbf{C}_i$ and $\mathbf{C}_j$.
8. Locate the shadow pixels in the image domain and map them to the mean of the pixels they have merged with or are closest to.
9. Post-process to eliminate the penumbra edge effect.
10. End

6. Results of Shadow Detection and Removal

Results of shadow detection and removal in real images are shown in figures (3, 6 and 7) for single-color background. In figure (3), the input is a real image of a synthetic frog captured with Kodak Zoom 120DC camera with a cast shadow on a gray background. In the said figure, the normalized image is shown in (b) while the extracted shadow image is shown in (c). The background which is set to white while the main object in the image is set to black is shown in (d). The restored image which exhibits some penumbra edge effect is shown in (e). This image is further processed to get rid of the edge effect due to the penumbra in the shadow. The final result is shown in (f). Similar results are illustrated in figure (6), where a cast shadow of a human head is formed on a wall. Further results for identification and removal of cast shadows on multi-color backgrounds are shown in figure (8).

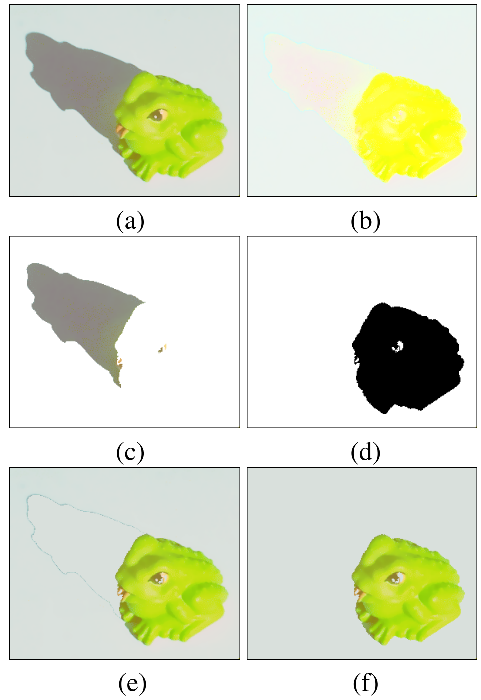


Figure 3: Illustration of shadow extraction: (a) is real image of an artificial frog. (b) is the normalized image (c) shows the extracted shadow and the background is shown in (d). (e) is the restored image with edge effect due to penumbra and (f) is the final image after post-processing.

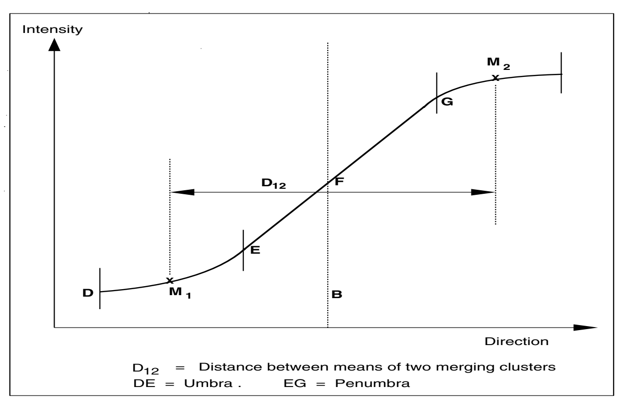


Figure 4: Transition from shadow to non-shadow. $\mathbf{M}_1$ *is the mean of the shadow cluster.* $\mathbf{M}_2$ *is the mean of the non-shadow cluster which the shadow cluster merged with* **PQ** *is the boundary between* $\mathbf{M}_1$ *and* $\mathbf{M}_2$

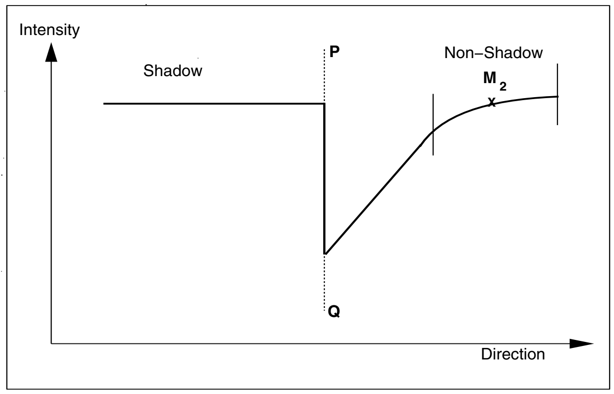
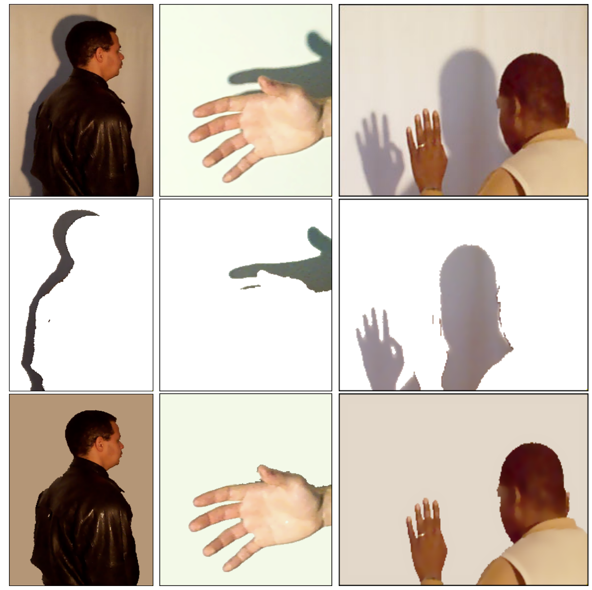
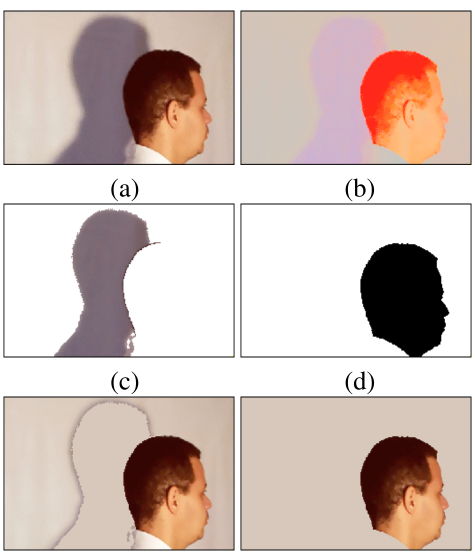


Figure 5: Mapping of shadow to the corresponding non-shadow mean: The result of mapping the shadow pixels in regions **DE** *and* **EF** *shown in figure 4 to the mean,* $\mathbf{M}_2$*. The edge effect is due the partial penumbra region* **FG** *not mapped to* $\mathbf{M}_2$*.*



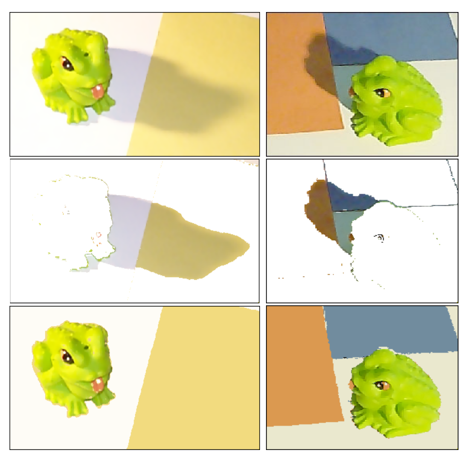
(a) (b) (c)

Figure 7: Further examples of cast shadow extraction and Image Restoration.



(a) (b) (c) (d) (e) (f)

Figure 6: Extraction of shadow cast by a human head: (a) is real image of a human head with a cast shadow on a wall. (b) is the normalized image, (c) shows the extracted shadow, (d) is the background, (e) is the restored image with edge effect due to penumbra and (f) is the final image after post-processing.



(a) (b)

Figure 8: Examples of removal of cast shadow on multi-color backgrounds.

7. Conclusion

The shadow detection and removal algorithm detects cast shadows that drop on the image backgrounds. It also recovers the background that is partly in shadow and partly lit and works equally well on multi-color backgrounds. The algorithm which is based on clustering and normalization techniques uses a user defined Euclidean distance measure constraint tool to validate the shadow candidate pairs.

Since shadow removal is based on mapping of the shadow pixels to the mean of the corresponding background pixels they reside on, the technique will not work well on a textured and random pixel backgrounds. The algorithm is also dependent on the clustering algorithm used. The robustness of our technique is therefore limited by the efficiency of the clustering scheme used.

A single light source was used in this study. Future work will focus on the extension of this to scenes illuminated with multiple light sources.

References

[1] A. Witkin, "Intensity-based edge classification.," *In Proceedings of the National Conference on AI* , pp. 36–41, August 1982.

[2] M. Ali, W. Martin, and J. Aggarwal, "Color-based computer analysis of aerial photographs," *Computer Graphics and Image Processing* **9**(4), pp. 282–293, 1979.

[3] B. Schachter, L. Davis, and A. Rosenfeld, "Scene segmentation by cluster detection in color space," *University of Maryland Computer Science Department* (424), 1975.

[4] E. Hall, *Computer Image Processing and Recognition*, Academic Press, 1979.

[5] A. Elgammal, D. Harwood, and L. Davis, *Non-parametric Model for Background Subtraction*, www.eecs.lehigh.edu/FRAME/Elgammal/bgmodel.html.

[6] G. Funka-Lea, "The visual recognition of shadows by an active observer," *A Dissertation in Computer and Information Science, University of Pennsylvania* , 1994.

[7] G. Funka-Lea and R. Bajcsy, "Combining color and geometry for the active, visual recognition of shadows," *Proc. International Conference on Computer Vision* , pp. 203 – 209, 1995.

[8] E. Salvador and a. E. A. Cavallaro, "Shadow identification and classification using invariant color models," *Proc. of IEEE Signal Processing Society International Conference on Acoustics, Speech, and Signal Processing (ICASSP-2001) Salt Lake City (Utah-USA)* , pp. 7–11, May 2001.

[9] K. Barnard and G. Finlayson, "Shadow identification using colour ratios," *Proc. of the IS&T/SID Eight Color Imaging Conference: Color Science, Systems and Applications* , pp. 97–101, 2000.

[10] C. Jiang and M. Ward, "Shadow segmentation and classification in a constrained environment," *CVGIP : Image Understanding* **59**(2), pp. 213–225, 1994.

[11] Y. Cheng, "Mean shift, mode seeking and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17**, pp. 790–799, 1995.