Shadow removal in indoor scenes

A. T. Nghiem, F. Bremond, M. Thonnat
Project Pulsar INRIA Sophia Antipolis France
2004 Route des Lucioles BP 93 06902 Sophia Antipolis France
atnghiem@sophia.inria.fr, Francois.Bremond@sophia.inria.fr, Monique.Thonnat@sophia.inria.fr

Abstract

In this paper, we propose a shadow removal algorithm for indoor scenes. This algorithm uses three types of constraints: chromaticity consistency, texture consistency and range of shadow intensity. The chromaticity consistency is verified in both HSV and RGB color spaces. The texture verification is based on the local coherency (over a pixel neighbourhood) of intensity reduction ratio between shadows and background. Finally, for the range of shadow intensity, we define a localized lower bound of the intensity reduction ratio so that dark mobile objects are not classified as shadows. Because the chromaticity constraint is only correct if the chromaticity of ambient light is the same as that of diffuse light, our algorithms only works in the indoor scenes.

1. Introduction

Object detection is the first stage in many video processing application such as traffic monitoring and video surveillance. During this stage, differentiating moving objects from shadows is a crucial task because shadows provoke various problems: object shape distortion, ghost objects etc.

Most of shadow removal algorithms (see review [11]) use one of two assumptions: shadows do not change object texture and chromaticities (hue and saturation value in HSV color space). For the chromaticity based algorithms [6], [7], [4], many of them do not construct an explicit model of shadows. Hence, they are unable to indicate in which case their hypotheses that shadows do not change the chromaticity are valid. For example, Cavallaro et al., [4] assumes that in RGB color space, the rank of color components (which channel is the biggest) does not change when a shadow occurs. In general this assumption is correct but it would be violated if for instance the object is gray and several light sources have different chromaticities in the scene as shown in figure 1. Finally, to characterize colors, many color spaces such as HSV, normalized RGB, La^*b^*, Lu^*v^* have been used without a complete analysis of light effect in complex real scenes. Consequently, some of them as the ones in [6] cannot be generalized to complex illumination conditions. A detail example is given in section 3.1 of this paper. The texture based algorithms [10] are less dependent on global illumination change but they are often dependent on background texture and do not often comply with real time requirements. Moreover, it is not uncommon that both background and moving objects have the same texture or no texture at all. Therefore, these algorithms should be combined with other approach such as chromaticity based to get a better performance.

In this paper, we propose a shadow removal algorithm that uses both texture and chromaticity features. Moreover, we try to find the localized lower bound of the intensity reduction ratio between shadows and background so that dark moving objects are not classified as shadows. The proposed algorithm works at the speed of 20 frames/s which could satisfy the real time requirement.

The paper is organized as follows. In section 2, we present the shadow model based on the Phong reflection model. From the shadow model, we set up the working conditions for our algorithm. Then section 3 and 4 discuss about chromaticity and texture, the two main features in our algorithm. Section 5 describes how to find the lower bound of intensity reduction ratios. Section 6 present our experiment and finally section 7 is our conclusion.

2. Shadow model

In this section, we first present the Phong reflection model [12], a simple illumination model widely used in 3D computer graphic. This model helps us to define the working conditions for our algorithm and to prove the local coherency (over a pixel neighbourhood) of intensity reduction ratio used in texture verification. This model is also used in [8] to detect shadows but only for gray scale images. Then, based on this model, the shadow effect is analyzed in case of complex real scenes. Finally, we present the lighting conditions under which our shadow removal algorithm works.

For the sake of simplicity, it is assumed that there is only one light source. The approach can be extended to the case of multiple light sources. According to the Phong illumination model, a surface point is lit by three types of light: ambient light i_a, diffuse light i_d, and specular light i_s. Then the point luminance in the image is described by the following formula:

$$I = k_a i_a + k_d (L \cdot N) i_d + k_s (R \cdot V)^{\alpha} i_s \tag{1}$$

where k_a is the ambient reflection constant, k_d is the diffuse reflection constant, k_s is the specular reflection constant, L is the direction vector from the point on the surface towards the light source, N is the normal at this point on the surface, R is the direction that a perfectly reflected ray of light (represented as a vector) would take from this point of the surface, V is the direction towards the viewer, α is a constant, $(\cdot)$ is the dot product operation.

The specular highlight is usually small and it is negligeable specially when the surface is not very shiny. Therefore, to simplify the model, we omit this term. Beside that, if we consider $k_d = k_a = k$ then (1) becomes:

$$I = k(i_a + (L \cdot N) i_d) \tag{2}$$

In RGB color space, (2) becomes:

$$I^j = k^j (i_a^j + (L \cdot N) i_d^j) \tag{3}$$

where index j corresponds to red, green and blue.

A shadow occurs when light power from the light source to a surface is partially or completely blocked by an object. Then, the equation of the point luminance become:

$$I^j_{shadow} = k^j (i_a^j + \beta (L \cdot N) i_d^j) \tag{4}$$

where $\beta \in [0, 1]$ indicating how much diffuse light has been blocked.

If the nonlinearity of the camera response function and the camera gain control are not considered, the point luminance becomes the pixel intensity in the image.

Equations (3), (4) show that, on image, the luminance of a particular object depends on object reflection constant and the light power that this object receives. In case of shadow, the power of the diffuse light coming onto this object reduces and it is lit mainly by the ambient light. If the chromaticity of ambient light is very different from the chromaticity of diffuse light, then the chromaticity invariance assumption would not be valid. For example, in the outdoor scene, the chromaticity of sunlight (the main light source) is yellow. Whereas, the ambient light is quite blue because it is the reflection of the blue sky. Hence, object becomes yellower when it is under the sun and it becomes bluer when it is inside the shadow. For a detail description, see [9], [5]. Or, in the extreme case as in [1], if the background is gray and if there are three light sources of three different chromaticities red, green, blue, then the shadow of the same object could have various chromaticities (Figure 1).

Figure 1. Color shadows occur when background is grey and there are multiple light sources with different chromaticities

Then the constraint on chromaticity consistency used by many algorithms such as [6], [7], [4] is valid only if light in the scene satisfies the following condition: *"The ambient light chromaticity is nearly the same as the chromaticity of the diffuse light"*.

Outdoor scenes, as discussed earlier, do not satisfy this condition. On the other hand, the above condition could be satisfied in the indoor scenes if the reflection constant of most of the objects in the scene is not too biased to a particular wavelength. Then the ambient light chromaticity is similar to the main light source chromaticity because it is the reflection of the main light source on the objects in the scene. In case of multiple light sources, the chromaticities of these light sources should not be very different.

3. Detecting shadow

Shadows can be detected using the features extracted from three domains: spectral [6], [7], [4], spatial [10] and temporal [8]. Nevertheless, temporal features are not very reliable because they depend heavily on the object speed and the frame rate of the camera. Hence, this paper mainly focuses on spectral and spatial features. Particularly, the following characteristics are exploited to detect shadow:

- *Chromaticity*: when there is shadow, object chromaticity remains the same.
- *Texture*: similarly, shadows could only slightly change object texture.
- *Intensity reduction*: for a specific scene and a specific lighting configuration, shadows could not reduce too much object luminance.

Beside that, feedbacks from higher level modules such as tracking and classification could help to improve the detection quality [4]. Yet, this post processing step is out of scope of our article.

The proposed shadow removal works as follows: (for each pixel P_a of the current image):

- **Step 1:** Verify if P_a's chromaticity is similar to the chromaticity of the background at the corresponding position. If yes, go to step 2, else P_a belongs to a moving object.
- **Step 2:** Verify if P_a's intensity is in range of shadows. If yes, go to step 3, else P_a belongs to a moving object.
- **Step 3:** Verify if the texture at P_a is similar to the texture of the background at the corresponding position. If yes, P_a is a shadow pixel, else P_a belongs to a moving object.

3.1. Chromaticities

Chromaticities can be represented using the standard RGB color space or some of its transformations such as $Lu * v*$, La^*b^* or HSV. In this section we discuss which color space is suitable for detecting shadow.

RGB advantages: In RGB color space, chromaticities are represented by two values r and g:

$$r = \frac{R}{R+G+B}; g = \frac{G}{R+G+B} \tag{5}$$

where R,G,B is the intensity level of red, green, blue in RGB color space. Because these values are independent, a small chromaticity change provokes a small change of r or g or both of them. Similarly, in the color spaces such as La^*b^* or Lu^*v^*, chromaticies are also represented by two independent dimensions. In [5], Benedek and Sziranyi claim that Lu^*v^* is better than RGB in detecting shadows in their experiment. However, the conversion from RGB (the original representation from the camera) to this color space is time consuming.

In HSV color space, chromaticities are represented by two values hue (H) and saturation (S). Hue represents the dominant wavelength of light and saturation expresses the ratio of the dominant wavelength power to the power of the other wavelengths. Unlike RGB color space, there is a strong relation between H and S. For instance, when S is nearly 0, the big difference in H does not mean a big difference in chromaticities because there is no dominant wavelength. Thus it would be ineffective if the thresholds for these two values are set independently to model the small chromaticity change as in [6].

Figure 2 shows the case when the algorithm using HSV color space in [6] is less effective than normalized RGB. This scene contains two light sources with two different chromaticities: the bulb is quite red and the fluorescent lamp is a bit blue. The person in the scene has blocked the light from the bulb and causes a shadow on the floor which is bluer than the normal background. For example, in RGB color space, the luminance of pixel A in figure 2 when there is shadow is (71, 75, 74) and when there is no shadow is (90, 91,86). In HSV color space, this two values correspond to (H = 165, S = 5%, V = 29%) and (H = 72, S = 5%, V = 36%), a big difference in H value. From this figure, we could notice that that the algorithm using HSV color space misses some shadow region on the floor even with a high threshold on the hue value ($d_H = 0.2$ with $H \in [0, 1]$, 2/5 of the hue difference value, $d_S = 0.25$ with $S \in [0, 1]$). The reason is that [6] does not take into account the relationship between H and S. In this case the difference in H does not mean much because the value of S is too small. On the other hand, the normalized RGB performance is quite good because the ratio between the intensity values of basic colors R,G,B does not change much when there is shadow on the floor.

HSV advantages: When the chromaticity difference between the ambient and diffuse lights is considerable, HSV color space will be useful if the saturation is considerable. For example, if the main light source is strongly dominated by a particular dominant wavelength (S is high in HSV color space), for example yellow, and if the reflectance coefficient k of the objects in the scene for yellow is not too small, the dominant wavelength of the ambient light could also be yellow because it is the reflection of the diffuse light on the objects in the scene. In HSV color space, the dominant wavelength is represented by H. Hence, despite of the big chromaticity difference between ambient and diffuse light, the value of H remains nearly the same. In contrast, if RGB is used, a high threshold is necessary to accommodate to this difference. Nevertheless, due to this weak chromaticity constraint and a lack of representation of dominant wavelength in RGB color space, two chromaticities with two different dominant wavelengths could be considered as similar. Consequently, algorithms using RGB color space commit more false positive errors. The same situation happens when the object reflectance coefficient to a particular wavelength is significantly higher than the reflection coefficients to other wavelengths. Then, whether there is shadow or not, light reflecting from these objects always have the same dominant wavelength. In both cases, the chromaticities of the pixels in the image have a high value of saturation in HSV color space.

Conclusion: when the saturation is high, chromaticity constraint should be verified in RGB color space with additional constraint on Hue. In contrast, if the saturation is close to 0, RGB color space should be used to detect shadow.

3.2. Texture

In many cases light coming from mobile objects and light coming from the background have the same chromaticity. If there is only the chromaticity constraint, these objects

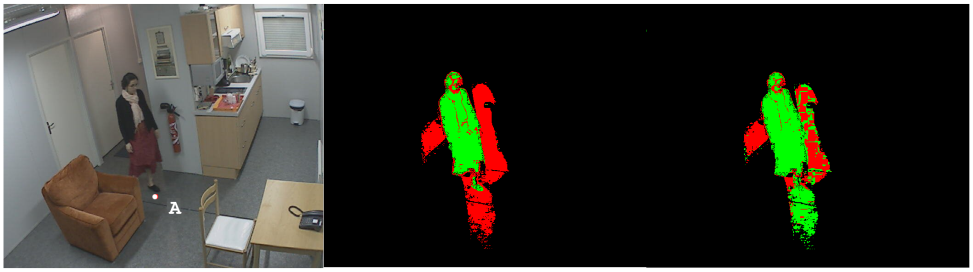

Figure 2. Shadow detection results using normalized RGB color space (center image) and HSV color space (right image) as in [6]. Red regions correspond to detected shadow regions. In RGB color space, the luminance of pixel A when there is shadow is (71,75,74) and when there is no shadow is (90,91,86). In HSV color space, this two values correspond to (H = 165, S=5%,V = 29%) and (H = 72, S = 5%,V = 36%), a big difference in hue.

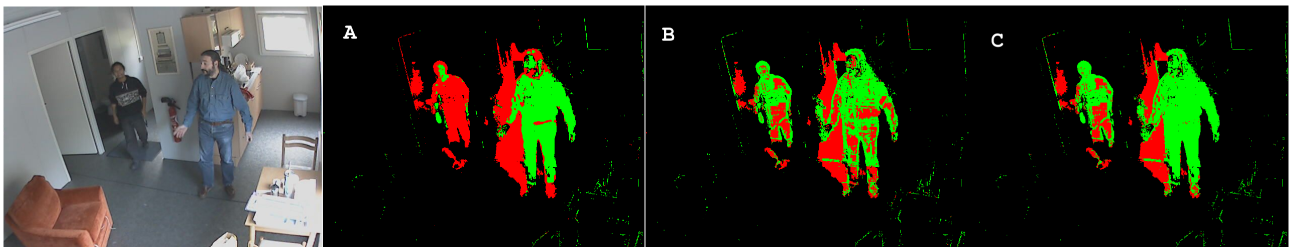

Figure 3. Shadow detection results of the algorithm using only chromaticity constraint (image A), of the algorithm using only texture constraint (image B) and of the combination of these two algorithms (image C). Chromaticity and texture complement each other to have a better result.

will be classified as shadows. For instance, image A in figure 3 shows the shadow detection results of an algorithm having only the chromaticity verification. Clearly, many parts of the person on the left are considered as shadow regions because the light coming from these regions and the light coming from the background when there is no shadow have the same chromaticity. In this case, texture could help improve the shadow detection performance because it imposes a constraint on the intensity level of adjacent pixels. However, if we employ a complicated texture based algorithm as in [10], the real time requirement could not be satisfied. In this paper, a simpler algorithm that exploits the texture information has been used.

We define the intensity reduction ratio as the ratio of pixel intensity when there is shadow to pixel intensity when there is no shadow. This ratio is computed using (3) and (4) as:

$$\begin{aligned} R^j &= \frac{I^j_{shadow}}{I^j} \\ &= \frac{k^j(i^j_a + \beta(L\cdot N)i^j_d)}{k^j(i^j_a + (L\cdot N)i^j_d)} \\ &= \frac{i^j_a + \beta(L\cdot N)i^j_d}{i^j_a + (L\cdot N)i^j_d} \end{aligned} \tag{6}$$

where index j corresponds to red, green or blue.

The proposed algorithm is based on the hypothesis that the change of light inside a shadow is quite smooth. In other words, inside a shadow, two adjacent pixels would have the same intensity reduction ratio.

If P_A and P_B are two adjacent pixels on the same flat surface, the term $(L \cdot N)$ in equation (6) of these two pixels would be the same. In addition, because of their close distance, if they are inside the same shadow, they would receive the same amount of ambient and diffuse light. Consequently, their intensity reduction ratios are nearly equal.

If P_A and P_B are on a curve surface, the term $(L \cdot N)$ of these pixels would be a bit different. Then a higher threshold of the difference between their intensity reduction ratios is necessary to model the texture. However in our paper, a unique threshold is used for the whole scene.

If there are multiple shadows of the same object, at the border of the intersection of these shadows, two adjacent pixels may receive two different amounts of light that make the assumption about the consistency of intensity reduction incorrect. Similar problem happens at the border of the two different surfaces such as the wall and the floor. However, in these cases the relative lighting on the two surface may change and other texture based algorithms could make the same mistake. Moreover because these errors only reside on thin lines, they could be eliminated using morphology operations.

Then to verify the texture at P_A, which is a potential shadow pixel (after the step of detecting shadows using chromaticity constraint), firstly we find the set$N(P_A)$ containing neighboring pixels of P_A which are also potential shadow pixels. If the difference between the intensity reduction ratio of P_A and that ratio of all pixel in $N(P_A)$ is smaller than a threshold d then PA is classified as a shadow pixel. If not, P_A belongs to a moving object.

In our experiment, the neighbouring pixels of P_A are among the 8-neighbour pixels of P_A and the relative value of d is 10% of the intensity reduction ratio of P_A.

Figure 3 illustrates the performance of our texture based algorithm without the chromaticity constraint. Comparing with the results of the chromaticity based algorithm, the texture base algorithm has recovered some textured regions of the person on the left but it also misses some regions of the person on the right. This figure also illustrates the results of combining these two constraints. The combination outperforms algorithms based on only one criteria.

3.3. Intensity reduction

From figure 3, it is clear that constraints on chromaticity and texture are not enough to detect shadows. Another feature could be exploited is the intensity reduction ratio. This ratio depends on the amount of light coming to the background that has been blocked by mobile objects. This in turn depends on the geometry of the background, light source, mobile object size and position. If we can model all these factors, we can compute exactly the intensity reduction ratio for every pixel inside the shadow. However this is impossible because we do not know yet which pixels belong to the mobile objects and which object causes the current shadow. Therefore a looser constraint on this ratio would be more feasible.

Normally, there is sufficient ambient light for the shadow to be not so dark. Therefore, there exists a lower bound on the intensity reduction ratio. The higher this lower bound, the less chance that dark mobile objects are classified as shadows. Because this lower bound depends on the ratio between ambient/diffuse light, it is often difficult to model. Moreover a unique lower bound for the whole scene is not effective. Hence, we propose to learn this value for each point in the scene using the ground truth. On the other hand, the upper bound of intensity reduction ratios remain 1 for most of the case for indoor scenes.

Learning lower bound on intensity reduction ratio

For a particular region in the scene, the lower bound on the intensity reduction ratio is the lowest ratio due to a shadow on this region. The lowest intensity reduction ratio of a point inside a shadow is located at the center of the shadow and close to the mobile object. Hence, for each shadow, the region that satisfies the above condition could be chosen to compute the lower bound. However this ratio

Figure 5. Pseudo color plot of lower bound on intensity reduction ratio of red channel of the scene in figure 4. Blue means the region does not have learning Data. Dark red means the region have a high lower bound, yellow means the region have a low lower bound

is often very small and is not a strong constraint on intensity reduction. Furthermore, for video processing applications, a small amount of undetected shadows very close to mobile objects is not a serious problem because it does not really change much object shape. Therefore it would be better if inside this shadow, the chosen region should have a reasonable distance from the mobile object. Due to this difficulty, the learning shadow regions have to be chosen manually. Then, the lower bound on the intensity reduction ratio of every pixel inside this region is the min value of the ratio caused by current shadow minus a certain constant. In case of multiple light sources, the same object would have multiple shadows on the same surface. Then the region of the darkest shadow will be chosen for learning. Figure 4 shows an example of selecting learning regions.

Generalizing learning results

Up to now, due to the notion of "reasonable distance from the mobile object", the selection of learning region still has been done manually. In addition, each shadow only provides learning data for a small region and it is impossible to have enough video data containing shadows over all pixels in the scene. For instance, from figure 4 we can see that only few regions have shadows to be used as learning data. Therefore learning results should be generalized.

To generalize the learning results for region R_1 to a new region R_2, the following heuristic is used: "If two pixels P_A and P_B have the same intensity value (the intensity of red, green, blue in RGB color space, for example), then the two corresponding points in the scene have the same lighting conditions (they receive the same amount of ambient and diffuse light, they have the same k and $(L \cdot N)$ in equation (2)), and therefore according to (6) they could use the same lower bound on intensity reduction ratio". Although this heuristic is incorrect with some specific combinations of contributing factors (reflection constant, ambient/diffuse light etc), generally the number of exceptions is small.

Because each pixel has its own lower bound, the learning results are not seriously affected by local illumina-

Figure 4. Only few regions (white regions) are selected for learning the lower bound of intensity reduction ratio

tion change. The pixels in regions not having illumination change could still use the old lower bound. Therefore this method could be used even when there is not much learning data with different lighting conditions.

Figure 5 illustrates the results of generalization from only few learning regions in figure 4. In this scene, for the red channel for example, the highest value of lower bound on intensity reduction ratio is 0.87 and the lowest value of this bound is 0.47. The default lower bound will be given to the regions not having learning data or not being generalized. Figure 6 shows the shadow detection results of the algorithm using the constraints on chromaticity and intensity reduction ratio. This figure also shows the advantage of the proposed method to the one using a unique lower bound for the whole scene as in [6]. With a unique lower bound (0.6, the mean value of 0.87 and 0.47), the region at the head of the person on the right is classified as shadow due to a too low lower bound whereas the region above the head of the person on the left is classified as moving object region due to a too high lower bound. With the learning approach, these regions are classified correctly thank to an adaptive lower bound of intensity reduction ratio.

4. Experimental results

To evaluate the performance of shadow removal algorithms, the results should be analyzed in various scenes with various moving objects. However, it is difficult to have precise shadow ground truth for many video data due to the complex shadow shape and fuzzy shadow boundaries. Furthermore, for a given scene and a given mobile object, the characteristics about their chromaticity and texture do not change much. Hence, the length of the video does not guaranty the algorithm performance. With a limited number of images containing several moving objects (with different chromaticity and texture) at various positions in the scene, the evaluation of the shadow detection performance could be acceptable. Thus, a quantitative evaluation has been performed on several images of three video sequences taken from two project CASSIOPEE [2] and ADVISOR [3]. Figure 7 illustrates sample images of these sequences. In term of shadow removal, the selected video sequences are difficult because the chromaticity of light from moving objects is similar to the chromaticity of light from background. Because the lighting conditions in these two scenes are quite homogeneous, a unique lower bound on intensity reduction ratio is sufficient for every pixel in the scene.

The evaluation results are described in table 1. In this table, HSV is the chromaticity based algorithm in [6] using HSV color space. CT is our algorithm using both chromaticity and texture constraints. Because the lighting conditions of these scenes are quite simple, we use the same intensity reduction ratio for the whole video sequences.

The algorithm performance are described using three metrics: precision, sensitivity, and F-Score (the balance between precision and sensitivity, $F-Score = 2*P*S/(P+S)$) of the object detection process. From this table we can see that the precision, sensitivity, and consequently, F-score of CT is always higher than that of HSV. The difference is small due to the small size of shadows.

Table 2. Evaluation results of algorithms using fixed vs localized lower bounds on intensity reduction ratio

	Precision	Sensitivity	F-Score
Fixed	0.910	0.735	0.813
Localized	0.916	0.750	0.825

Table 2 illustrates the quantitative evaluation of two algorithms using fixed and localized lower bounds on intensity reduction ratio on the video sequence in section 3. Because lighting condition of this sequence is complex, each region has its own lower bound. As a result, the algorithm using localized lower bound achieve a higher evaluation results.

5. Conclusions

In this paper, we present the shadow model based on Phong shading model. The shadow model has shown that the chromaticity consistency constraint, widely used to detect shadows, is valid only if the ambient light chromaticity is not very different from the chromaticity of diffuse light. Based on this analysis and the shadow model, we propose a shadow removal algorithm exploiting three types of constraints: chromaticity consistency, texture consistency and lower bound on the intensity reduction ratio.

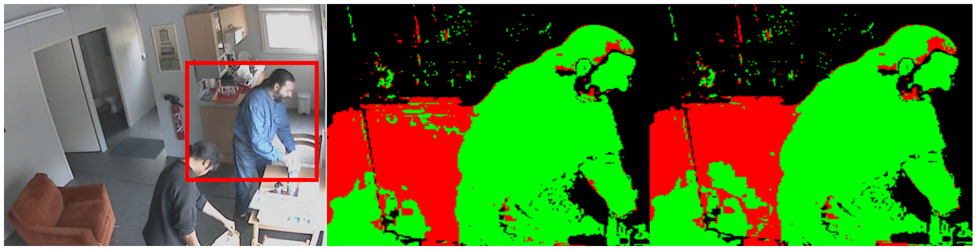

Figure 6. Shadow detection results of learned intensity reduction ratio algorithm (center image) and fixed ratio algorithm (right image). These images correspond to the extracted red region of the left image. Learning method detects more object region and has less false positive errors

Table 1. Evaluation results of chromaticity based algorithm (HSV) and chromaticity-texture based algorithm (CT)

	Cassiopee			Metro Station 1			Metro Station 2		
	Precision(P)	Sensitivity(S)	F-Score(F)	P	S	F	P	S	F
HSV	0.858	0.601	0.728	0.567	0.707	0.629	0.823	0.710	0.762
CT	0.875	0.626	0.730	0.602	0.711	0.652	0.873	0.755	0.809

Chromaticities can be represented using various color spaces such as RGB, HSV etc. Nevertheless, each of them has its own advantages and disadvantages. Therefore, the analysis has proven that a combination of RGB and HSV could have a better shadow detection performance than either RGB or HSV alone.

The texture constraint is an useful complement for the chromaticity constraint when the chromaticity of light coming from moving objects is similar to the chromaticity of light coming from background. Dedicated to shadow characteristics, this texture verification is quite simple comparing to other texture based algorithms because it does not directly detect the scene texture. Despite of its simpleness, the experiment has proven its efficiency in distinguishing moving objects from shadows.

Finally, a high value of lower bound on intensity reduction ratio is necessary because it avoids classifying dark objects as shadows. Hence to be highest possible, this lower bound should be localized when the intensity reduction ratio is very different from regions to regions. Due to the difficulties of computing this lower bound based on scene geometry, lighting conditions, mobile object position, a learning approach has been adopted. Due to the lack of learning data, the learning results have been generalized based on the chromaticity and intensity similarity.

In the future, we will extend our algorithm to work with outdoor scenes. To do this we will study the difference between the sunlight and the ambient light in the outdoor scenes and propose an appropriate chromaticity constraints. Moreover, the intensity reduction in outdoor scenes changes rapidly according to the intensity of the sun light. Therefore an efficient adaptation method is necessary.

References

[1] Colored shadows. http://www.ics.uci.edu/ eppstein/pix/sv/ex/ColorShadow.html. 2

[2] Colored shadows. http://www-sop.inria.fr/orion/ personnel/ Francois.Bremond/ topicsText/ cassiopeeProject.html. 6

[3] A. Avanzi, F. Brémond, C. Tornieri, and M. Thonnat. Design and assessment of an intelligent activity monitoring platform. *EURASIP Journal on Applied Signal Processing, Special Issue on "Advances in Intelligent Vision Systems: Methods and Applications"*, 14(8):2359–2374, 2005. 6

[4] A. Cavallaro, E. Salvador, and T. Ebrahimi. Detecting shadows in image sequences. pages 165–174, 2004. 1st European Conference on Visual Media Production. 1, 2

[5] C.Benedek and T. Sziranyi. Study on color space selection for detecting cast shadows in video surveillance. *International Journal of Imaging Systems and Technology*, 17:190–201, 2007. 2, 3

[6] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25:1337–1342, 2003. 1, 2, 3, 4, 6

[7] T. Horprasert, D. Harwood, and L. Davis. A robust background subtraction and shadow detection. 2001. Asian Conference on Computer Vision. 1, 2

[8] J. O. J. Stauder, R. Mech. Detection of moving cast shadows for object segmentation. *IEEE Transaction on Multimedia*, 1:65–76, 1999. 1, 2

[9] E. Khan and E. Reinhard. Evaluation of color spaces for edge classification in outdoor scenes. In *IEEE International Conference on Image Processing*, volume 3, pages 952–955, 2005. 2

[10] A. Leone, C. Distante, and F. Buccolieri. A texture-based approach for shadow detection. pages 371–376, 2005. IEEE Conference on Advanced Video and Signal Based Surveillance,. 1, 2, 4

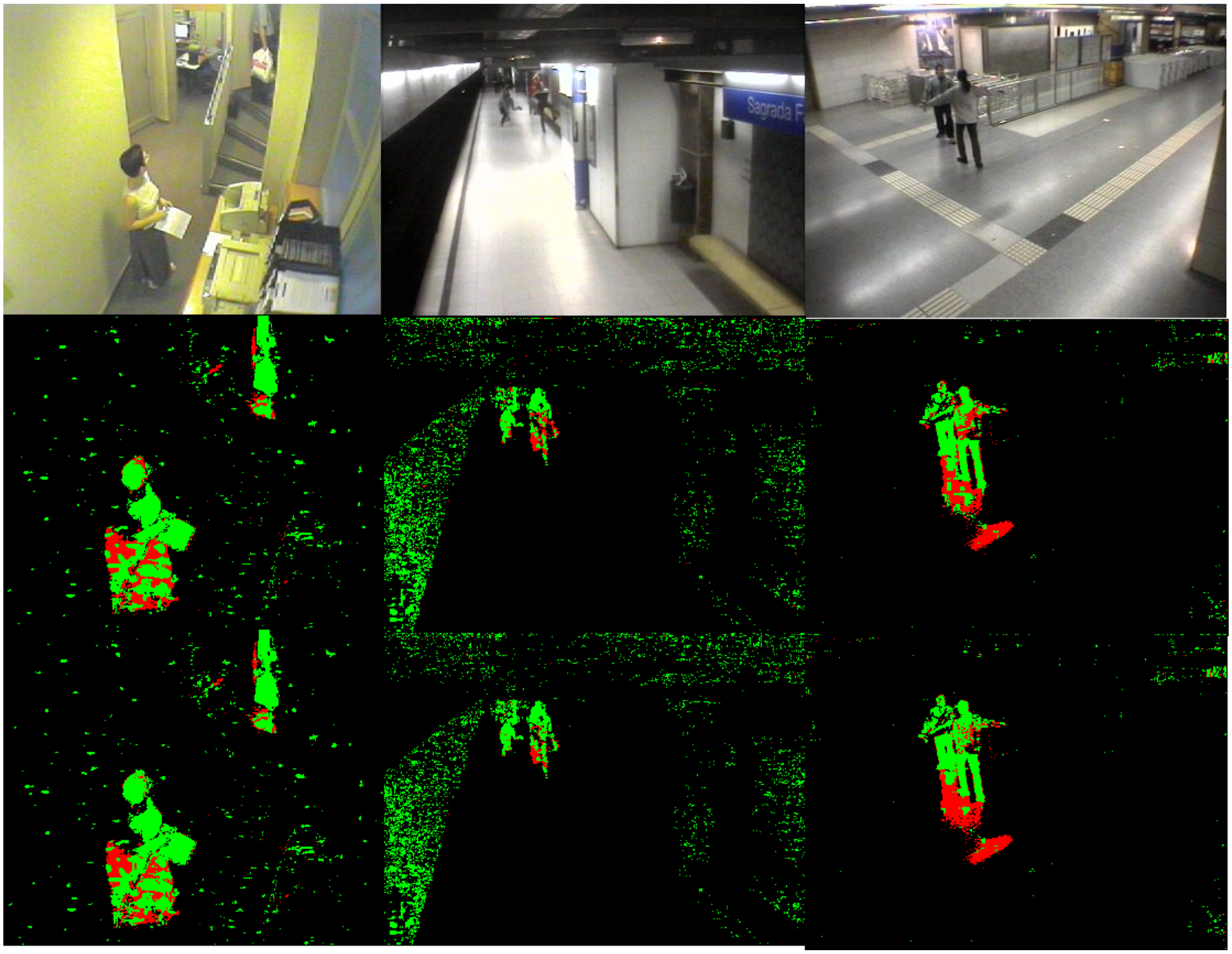

Figure 7. First row: sample images of video sequences from CASSIOPEE (image on the left) and ADVISOR (2 images on the right) projects. Second row: object detection results of hsv based method. Third row: detection results of proposed method. The red regions correspond to the detected shadow. The green regions correspond to the detected moving objects. The proposed method is better at differentiating moving objects and shadows.

[11] A. Prati, I. Mikic, M. Trivedi, and R. Cucchiara. Detecting moving shadows: Algorithms and evaluation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25:918–923, 2003. 1

[12] T.P.Bui. Illumination for computer generated pictures. *Communications of the ACM*, 18:311–317, 1975. 1