# Shadow detection using a physical basis

**P.J. Withagen**[1,], **F.C.A. Groen**[2], **and K. Schutte**[3]
[1]X-ray, BL Components
Philips Medical Systems
Paul.Withagen@philips.com
[2]Intelligent Autonomous Systems group
University of Amsterdam
F.C.A.Groen@uva.nl
[3]Electro Optics group
TNO Defence, Security and Safety
Klamer.Schutte@tno.nl

Shadow detection is an important aspect of foreground/background classification. Many techniques exist, most of them assuming that only the intensity changes under shadow. In this paper we show that in most practical indoor and outdoor situations there will also be a color shift. We propose an algorithm for estimating this color shift from the images, and using it to remove shadow pixels. The proposed algorithm is compared experimentally to an existing algorithm using real image sequences. Results show a significant improvement of performance.
**Keywords:** Shadow; Foreground/background classification; Object segmentation; Object detection

# IAS

intelligent autonomous systems

# Contents

**Intelligent Autonomous Systems**
Informatics Institute, Faculty of Science
University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam
The Netherlands

Tel (fax): +31 20 525 7461 (7490)
http://www.science.uva.nl/research/ias/

**Corresponding author:**

F.C.A. Groen
tel: +31 20 525 7461
Groen@science.uva.nl
http://www.science.uva.nl/~Groen/

# 1    Introduction and previous work

Shadow classification is an important aspect of most object detection and tracking algorithms. Practically any scene, both indoor and outdoor, contains shadows. Object detection algorithms that do not rely solely on intensity-invariant features detect these shadows as foreground. This alone is a problem, as object segmentation will be inaccurate and objects can become connected by their shadows.

A more severe problem occurs for object tracking algorithms that use a model of the moving object. Shadow is labelled part of a moving object, leading to updating the model with a shaded version of the background. The color model of the moving objects is often less sensitive than that of the background, as it should describe several object colors using less training data. Therefore, after updating the model with shaded background, the unshaded background can also give a high object probability. This may lead to situations where large parts of the background are misclassified as objects. Therefore, specifically for this kind of algorithms it is important to remove shadows from foreground objects, i.e. to reassign shadow pixels that were initially classified as foreground to the background.

An overview of shadow modelling techniques is given in [10]. The most simple solution is to ignore intensity information altogether by using an intensity invariant color space such as [2]. This assumes that shadow causes the intensity to change, while leaving the color information unchanged. An important drawback of this approach is that almost black and almost white pixels are treated equal to other pixels, while for these pixels the color information is highly unreliable.

A simple shadow detection approach uses the assumption that the intensity changes equally over small image regions [9]. This allows for intensity correction for this region, eliminating shadows. More advanced is the use of additional features such as combining color and intensity [3, 7] or adding gradient information [6, 5]. In [4] a course-to-fine approach is used. First the rough shadow boundaries are obtained using a moment-based method. The rough approximation of the shadow region is then further refined using features such as the orientation, mean intensity, and center position of a shadow region.

In [3, 7] the authors allow distortions on the pixel value: the intensity and the color. This is equivalent to assuming a cylinder in RGB–space through $(0, 0, 0)$ and the old background RGB value. A pixel located inside this cylinder is classified as (shaded or highlighted) background, while pixels outside the cylinder are classified as foreground. This approach allows to assign pixels with high or low intensity to the foreground in the case additional noise is assumed. When the image noise is multiplicative, a cone should be used instead of a cylinder.

Both the normalized intensity approach and the cylinder approach assume that under shadow the color change is small and equally probable in each direction. This is only true when all light sources have the same color. However, this is generally not the case. For example: an outdoor scene usually has two light sources, direct sunlight and blue ambient sky illumination. Shadows will cause a color shift of which the direction can be predicted.

Generally, the colors of the illumination sources will be relatively constant over time. They can therefore be learned. When the colors of the illumination sources are known, the color shift caused by occluding one or more of the light sources can be predicted. A known color shift allows a smaller color area to be classified as shadow, leading to less missed objects.

In this paper we analyze the physical aspects of the color change caused by shadow. We propose the $q$–space and an algorithm to predict the color change caused by shadow. This leads to an algorithm for accurate shadow classification by taking into account the predicted color shift. The proposed algorithm is evaluated using real image sequences.

Our approach is most similar to [8]. However, instead of using a constant, predefined color-shift towards blue, we learn the color shift from the images. In addition, our approach is not

limited to pre-defined background materials with large enough homogenous areas and we allow highlights, which may be caused by shadow of the background image.

This paper is structured as followed. In section 2 the physical nature of shadow is analyzed. Then in section 3 the computational efficient $q$-space for shadow detection is introduced, together with an algorithm to determine the color shift caused by shadow. In section 4 the proposed algorithm is evaluated experimentally. Finally, conclusions are presented in section 5.

## 2   Physics of shadow

The amount of light reaching the CCD sensor $C_c(p)$ of sensor band $c \in \{R, G, B\}$ at position $p$ is given by [1]

$$C_c(p) = \int_\lambda e(\lambda, p) \cdot \rho(\lambda, p) \cdot f_c(\lambda) \ \mathrm{d}\lambda \tag{1}$$

with $e(\lambda, p)$ the illumination spectrum, $\rho(\lambda, p)$ the surface reflection (surface albedo), $f_c(\lambda)$ the sensor response function and $\lambda$ the wavelength. $\rho(\lambda, p)$ will generally depend on the direction of the incoming light and the direction of the camera. We will assume only diffuse reflections.

The illumination spectrum can be modelled as a sum of all contributing, independent, light sources $e_1 \cdots e_{N_l}$:

$$e(\lambda, p) = \sum_{l=1}^{N_l} e_l(\lambda, p) \ , \tag{2}$$

where $e_l(\lambda, p)$ depends on the direction of the light and the camera.

Due to changing (partial) occlusion, the illumination of each of the light sources can change over time. Under the assumption that this only affects the intensity and not the spectral composition of the light sources, this can be modelled by

$$e(\lambda, p, t) = \sum_{l=1}^{N_l} \alpha_l(p, t) \cdot e_l(\lambda, p). \tag{3}$$

with $\alpha_l(p, t)$ a time and position dependent factor accounting for a changing illumination intensity over time.

In a local area, where the illumination spectrum of each light source can be considered spatially constant, the illumination spectrum no longer depends on the pixel location $e_l(\lambda, p) = e_l(\lambda)$. This results in a sensor illumination $C_c(p, t)$ at time $t$ of

$$C_c(p, t) = \sum_{l=1}^{N_l} \alpha_l(p, t) \int_\lambda e_l(\lambda) \cdot \rho(\lambda, p) \cdot f_c(\lambda) \ \mathrm{d}\lambda, \tag{4}$$

under the assumption that the scene is static and the location and spectral features of the illumination and camera are constant over time.

When either the illumination spectrum or the reflectance spectrum can be considered constant over the spectral range of each of the camera bands[1] this equation can be further simplified. In this case, the term that may be considered constant can be put in front of the integral. We can then split the sensor illumination in a part that depends on time and light source $\alpha_l(p, t)$,

---

[1]So there are no peaks in both illumination and reflection spectra This assumption holds for smooth light spectra such as that of the sun and normal light bulbs. Fluorescent light in combination with peaked reflection spectra may cause problems.

a part that depends on the light source and camera band $L_{c,l}$ and a part that depends on the location and camera band $R_{c,p}$:

$$C_c(p,t) = R_{c,p} \cdot \sum_{l=1}^{N_l} \alpha_l(p,t) \cdot L_{c,l} \ , \tag{5}$$

with

$$L_{c,l} = \int_\lambda e_l(\lambda) \cdot f_c(\lambda) \ \mathrm{d}\lambda \tag{6}$$

$$R_{c,p} = \frac{\int_\lambda \rho(\lambda,p) \cdot f_c(\lambda) \ \mathrm{d}\lambda}{\int_\lambda f_c(\lambda) \ \mathrm{d}\lambda}. \tag{7}$$

## 2.1   Adding the CCD model

We show in [13] that for sufficiently large pixel values the CCD sensor can be modelled by

$$i_c(p,t) = g(t)^\gamma \big(h(t)C_c(p,t) + N_{\mathrm{S}}\big)^\gamma \ . \tag{8}$$

with $C_c(p,t)$ the pixel intensity for color channel $c \in \{R,G,B\}$. $g(t)$ denotes the camera gain, $h(t)$ is a factor related to the shutter time and iris setting. $\gamma$ is the gamma-correction of the camera. The noise $N_{\mathrm{S}}$ is zero-mean multiplicative shot noise.

## 2.2   Two light sources

Let us consider one pixel at location $p_0$ at times $t_0$ and $t_1$. There are two light sources with colors $\vec{L}_1$ and $\vec{L}_2$. The reflectivity of the scene depicted in this pixel is given by $\vec{R}(p_0)$. This gives for the (noise free) pixel intensity in the two frames:

$$i_c(p_0,t_0) = g^\gamma(t_0) \left(R_c(p_0)h(t_0)(\alpha_1(p_0,t_0)L_{c,1} + \alpha_2(p_0,t_0)L_{c,2})\right)^\gamma \tag{9}$$

$$i_c(p_0,t_1) = g^\gamma(t_1) \left(R_c(p_0)h(t_1)(\alpha_1(p_0,t_1)L_{c,1} + \alpha_2(p_0,t_1)L_{c,2})\right)^\gamma \ , \tag{10}$$

With $\alpha$ a factor depicting the brightness of the light sources. We assume that the reflectivity is equal in both images, e.g. constant scene.

## 3   The $q$–space for shadow

We will assume $\gamma = 1$ and $g$ and $h$ constant for shadow detection. Alternatively, images can be corrected for gamma before shadow detection. The quotient of one pixel at two instances in time is given by:

$$q_c(p_0,t_{1,0}) = \frac{i_c(p_0,t_1)}{i_c(p_0,t_0)} = \frac{\alpha_1(p_0,t_1)L_{c,1} + \alpha_2(p_0,t_1)L_{c,2}}{\alpha_1(p_0,t_0)L_{c,1} + \alpha_2(p_0,t_0)L_{c,2}} \ . \tag{11}$$

This quotient is independent of the scene reflectivity. When the two images have equal illumination, so there is no shadow, $q_c = 1$ for all color bands. We now assume that non-unity values of $q_c$ are due to shadows or highlights.

We consider a region $S$ in the image with equal reference illumination, for which the $L$'s and $\alpha$'s are assumed equal for each pixel at $t_0$. We define this illumination $\vec{L}_0$:

$$\vec{L}_0(p_i \in S) = \alpha_1(p_i,t_0)\vec{L}_1 + \alpha_2(p_i,t_0)\vec{L}_2 \ , \tag{12}$$
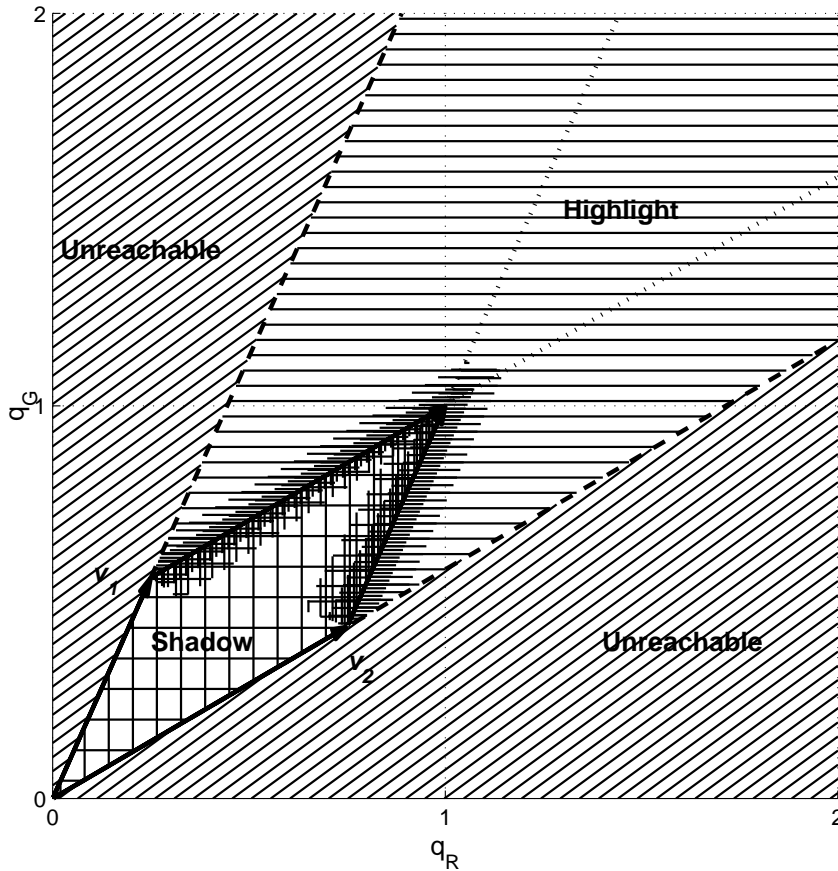
for all $p_i \in S$.

**Figure 1:** Analysis of a 2D projection of the $q$–space. A surface is spawn by two vectors $\vec{v}_1$ and $\vec{v}_2$. A shaded pixel falls inside the double-hatched parallelogram, a highlighted pixel in the horizontally hatched area. When only one illumination source is (partly) occluded a pixels moves along the corresponding dotted line. The two diagonally hatched areas depict areas that cannot be reached. Some areas are hatched more dense, indicating the higher density of the $q$–space for a typical image.

For different $\alpha$ values at time $t_1$, the $\vec{q}$–values for each pixel $p_i \in S$ lie on a flat surface in the 3D $q$–space. This surface is defined by the two vectors $\vec{v}_1$ and $\vec{v}_2$ with elements $v_{1,c}(p_i, t_1) = \frac{\alpha_1(p_i,t_1)L_{c,1}}{L_{c,0}(p_i)}$ and $v_{2,c}(p_i, t_1) = \frac{\alpha_2(p_i,t_1)L_{c,2}}{L_{c,0}(p_i)}$ respectively. This surface has the shape of a triangle with one corner at $(0,0,0)$ and two sides given by the two vectors. Point $(1,1,1)$ always lies on this surface. See Figure 1 for a 2D plot of the $q$–space. The $q$–space is divided in three volumes:

- Values outside the triangle with its top at $(0,0,0)$ cannot occur when the assumptions of constant scene and constant illumination are valid.

- The triangle defines all possible $q$'s. It is divided in two 2D surfaces:

  - Inside a parallelogram with corners at $(0,0,0)$ and $(1,1,1)$ and sides in the direction of $\vec{v}_1$ and $\vec{v}_2$ are all pixels that are under shadow.

  - Outside this parallelogram, but inside the triangle are all pixels that are highlighted, i.e. their illumination intensity is higher than that of the reference image for at least one of the light sources. This occurs when the reference image contains shadows.

Due to noise, the flat surface will in reality have a certain thickness.

### 3.1   Online estimation of the $q$–space parameters

For known colors of illumination sources, the vectors $\vec{v}_1$ and $\vec{v}_2$ can be calculated and used for shadow correction. However, in general this information is not available. The relative color of the illumination sources should then be estimated from the image sequence.

We consider the case that only one light source is (partially) occluded. The shadow area in $q$–space is now formed by a straight line through $(1, 1, 1)$. Assuming the second light source is occluded, $\vec{q}$ is given by

$$\vec{q}(p_0, t_{1,0}) = 1 + (\frac{\alpha_2(p_0, t_1)}{\alpha_2(p_0, t_0)} - 1)\vec{v}_2 \ . \tag{13}$$

This is a straight line through $(1, 1, 1)$ with the direction given by $\vec{v}_2$. In Figure 1 these values of $\vec{q}$ are densely hashed for both light sources. Occluding only the second light source generates the line form the end of vector $\vec{v}_1$ to $(1, 1, 1)$.

We project the $q$–space such that different points in the $q$–space that lie on a straight line through $(1, 1, 1)$ are projected on the same point. The following projection enables this:

$$\left( \begin{array}{c} b_1 \\ b_2 \end{array} \right) = \frac{1}{F(\vec{q})} \left( \begin{array}{ccc} 0 & \cos(30°) & -\cos(30°) \\ 1 & -\sin(30°) & -\sin(30°) \end{array} \right) \left( \begin{array}{c} q_R \\ q_G \\ q_B \end{array} \right) , \tag{14}$$

with the normalization factor $F(\vec{q})$

$$F(\vec{q}) = \frac{q_R + q_G + q_B}{3} \ . \tag{15}$$

From the values $b_1$ and $b_2$ for all pixels in $S$ (defined in Equation 12) we estimate the parameters $\overline{b_1}$ and $\overline{b_2}$ using the mean or median. These estimates define the color shift caused by partial occlusion of one light source for image region $S$.

Generally, different pixels will have a different ratio of $\alpha_2(p_0, t_1)$ and $\alpha_2(p_0, t_0)$ in Equation 13. For example, close to an object a larger part of the sky will be occluded than at some distance, while direct sunlight can be fully occluded for both pixels. So we need to estimate a line-segment in the $b$–space. This line segment will have certain thickness. It spans the V-shaped plane in the $q$–space, also with certain thickness. The line segment is estimated using an ellipse, where the two foci are assumed to be the endpoints of the line segment, so the coordinates of the vectors $\vec{v}_1$ and $\vec{v}_2$.

The length of the line segment with respect to the width of the line is determined by:

- The amount of color saturation in the images.
- The difference in color of the two illumination sources.
- The amount of variation in the ratio of $\alpha$'s between pixels.
- The amount of shadow pixels.
- The amount of image noise.

## 4   Experimental evaluation

In order to quantitatively evaluate the proposed algorithm it will be used to remove shadow in a foreground/background classification task on real image sequences. The classification results given in [14] will be used as initial foreground/background segmentation. Pixels classified as background are used to estimate an ellipse in the $b$–space, assuming that the background region also contains shadow pixels. This ellipse is them used to classify the foreground pixels. Evaluation is performed by comparing the classification result to that of a reference algorithm.

|        |        |        |        |
|--------|--------|--------|--------|
| (a) 500 | (b) 750 | (c) 1000 | (d) 1250 |

**Figure 2:** Some images from the Intratuin sequence.



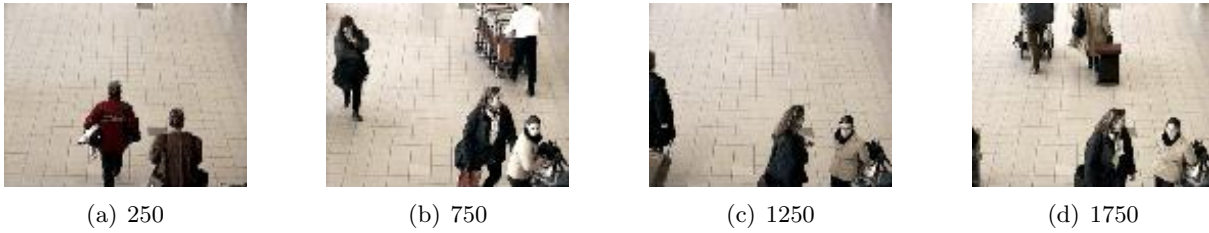| | | | |
|---|---|---|---|
| (a) 250 | (b) 750 | (c) 1250 | (d) 1750 |

**Figure 3:** Some images from the Schiphol sequence.

## 4.1    Implementation details and image sequences

The image sequences used and the value of fixed parameters will be given below. The other parameters were varied for parameter optimization.

### 4.1.1    Image sequences

Three image sequences were used for the evaluation, see also figures 2, 3 and 4:

- *Intratuin*: Parking lot with waving tree branches. In this sequence there is no significant variation in intensity.The sequence contains cars and pedestrians, moving both slowly and fast. This sequence has 150x350 pixels and 1250 frames.

- *Schiphol*: Recorded in the main hall of Schiphol airport. There are a lot of global intensity variations due to automatic gain control. The sequence contains relatively large objects, some of which become stationary. This sequence has 90x120 pixels and 1750 frames.

- *PETS 2001*: We used a cut-out from the images of dataset 3, training, camera 1 from the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance 2001 (PETS). There are large local changes in illumination intensity due to clouds. The images contain relatively few object pixels. The part of the image that is used is between rows 300 and 520, skipping the odd rows and between columns 350 and 750, skipping the odd columns. This sequence has 120x200 pixels and 5500 frames.

All sequences are RGB color video data with eight bit per color. For each sequence, five to eleven images were manually labeled. Each pixel was labeled: foreground, background or any. The label any is used for the edges of objects, where it is difficult (for a human) to decide whether this pixel should be labeled as either foreground or background. It is also used for some artifacts in the images like moving objects that stop moving.

The Intratuin and Schiphol sequences are recorded by us and they are available through our website (www...(left blank for review)), together with the manually labeled ground truth for all sequences. The PETS 2001 data is available through pets2001.visualsurveillance.org.

(a) 2000          (b) 3500          (c) 4500          (d) 5500

**Figure 4:** Some images from the PETS sequence.

### 4.1.2   Reference algorithm

The reference method classifies pixels based on their change in color. This is equivalent to using a cylinder in the $q$–space with its axis through points $(0, 0, 0)$ and $(1, 1, 1)$. The cylinder has a width $T_{\mathrm{cylinder}}$. Pixels with a value within the cylinder are classified as shadow, pixels outside the cylinder as foreground.

### 4.1.3   Upper and lower bounds

Pixels with a component of $\vec{q}$ below $T_{\mathrm{qmin}}$ or above $T_{\mathrm{qmax}}$ are not considered to be shadow. The same yields for pixels for which one of the components of $\vec{q}$ is closer to one than $T_{\mathrm{qone}}$. These three parameters are used for both the proposed and the reference algorithm.

### 4.1.4   Threshold

The proposed algorithm iteratively estimates the smallest ellipse containing the fraction $T_{\mathrm{ellipse}}$ pixels on the background pixels in the $b$–space.

### 4.1.5   Post-processing

In order to prevent missing objects, prior knowledge is used in a post-processing stage. We know that a shadow is cast by an object. It can be either connected to that object or not. We assume that a foreground region is an unconnected shadow region when more than a fraction $T_{\mathrm{total}}$ of the pixels in the region are classified as shadow based on their color.

Shadows are in most cases cast on the ground. When a shadow region is connected (at least one pixel is 4–connected) to a moving object, we assume that the shadow starts near the lowest point of the object. Therefore, shadow regions that do not have shadow pixels in the lowest $T_{\mathrm{connected}}$ fraction of the blob are re-labelled object.

We used for both the reference and the proposed algorithm $T_{\mathrm{total}} = 0.8$ and $T_{\mathrm{connected}} = 0.2$.

## 4.2   Demonstration of the $b$–space

Figures 5 and 6 show four examples of color shifts due to shadow and object. For one pixel over a number of frames the $q$–values are calculated by dividing the value of one pixel for a number of frames by the pixel value without shadow/object. From these $q$–vales, $b$–values are calculated and using the median over all frames. The color shift is calculated. The estimated value of $b$ defines the dashed line in the $q$–space.

In Figure 5 two examples are shown from the Intratuin sequence. Figure 5(a) is an example of shadow. The plot of the $b$–space shows that the $b$–values for the different frames are clustered away from the center, in the direction of red. This means that the shadow causes a color shift towards red. It gives in the $q$–space a line through $(1, 1, 1)$ away from the line through $(0, 0, 0)$ and $(1, 1, 1)$, which is the orientation of the cylinder in the reference algorithm. It is worth

mentioning that most shadows in our experiments did not cause values of $q_c$ below 0.5, lower values were caused by objects.

The distance to the origin in the $b$–plot gives the amount of color shift. The larger this distance, the better the proposed approach will perform compared to the approach using a cylinder or cone.

In Figure 5(b) an example of a blue object is given. It can be seen that the values in the $b$–plot are much less clustered. The direction of the color shift is now opposite to that in the case of shadow.

Figure 6 shows two examples for the PETS01-3TR1 image sequence. Again, one example of shadow and one example of an object. The color shift due to shadow is here towards blue. The example of an object in Figure 6(b) shows a white object passing the pixel. The $b$–values are again much less clustered.

## 4.3   Evaluation criterion

The shadow detection is intended to be used in conjunction with foreground/background classification. The performance of shadow detection will be evaluated with this application in mind.

As shadow pixels will be considered background, there is no sense in performing shadow classification on pixels classified as background. So only pixels classified as foreground by the foreground/background classification algorithm will be considered.

The set of pixels classified as foreground contains two sub-sets: true foreground $F_T$, those pixels correctly classified as foreground, and false foreground $F_F$, those pixels that are actually background. Pixels in set $S$ of pixels classified as shadow belong to either of these subsets. We propose two fractions

$$r_{\mathrm{bad}} = \frac{N(S \in F_T)}{N(F_T)} \tag{16}$$

$$r_{\mathrm{good}} = \frac{N(S \in F_F)}{N(F_F)} \ , \tag{17}$$

with $N(S)$ the number of pixels in (sub)set $S$. Perfect shadow classification gives $r_{\mathrm{good}} = 1$ and $r_{\mathrm{bad}} = 0$. To evaluate different settings of the algorithm, a plot will be made of $r_{\mathrm{bad}}$ against $r_{\mathrm{good}}$ for different parameter settings. This plot will be called the good/bad–plot. It looks like an Receiver Operator Characteristic (ROC) curve [11, 12]. As with the ROC curve, only points on its convex hull will be drawn.

## 4.4   Experimental results

In Figure 7 the convex hull of the good/bad–plot is given for the proposed and the reference method. Except for the PETS01-3TR1 image sequence, the proposed algorithm performs better than the reference method based on a cylinder. Numerical results in Table 1 show an average error reduction of a factor two. The results for the PETS01-3TR1 sequence are almost identical for both the proposed and the reference algorithm. This is probably caused by the lack of color saturation or the equal color of illumination sources in this image sequence. This causes very small color shifts, reducing the results of the proposed algorithm to be similar to that of the reference algorithm.

## 5   Conclusions and future work

In this paper a new shadow detection algorithm has been introduced. It was argued that for scenes with two or more illumination sources with different colors, shadow causes not only the intensity but also the color to change.

**Table 1:** Surface above the good/bad–plots. All numbers are percentages where lower is better.

| Data: | Intratuin | Schiphol | PETS01-3TR1 | Average |
|---|---|---|---|---|
| Good/bad–plot Cylinder | 20.9 | 36.6 | 10.63 | 22.7 |
| Good/bad–plot Proposed | 10.1 | 12.6 | 11.07 | 11.3 |

We have demonstrated that the color shift depends linearly on the change in intensity. The direction of the shift in color depends on the difference in color of the light sources. It has been shown that the shift direction can be estimated and used for shadow detection.

Experiments on real images with our proposed error measure show for the proposed shadow removal method an error reduction of 50 percent compared to a reference algorithm.

## 5.1   Future work

The proposed algorithm has been evaluated on a per-frame basis. It would be better to learn the color of the light sources over time, leading to a less computationally complex and a more stable algorithm. The illumination color should be estimated using a different threshold than that used for pixel classification. Additionally, also the location of light sources could be modelled, leading to a 3D model of where shadows can occur, and what color-shift can be expected at that location.

# References

[1] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, Englewood Cliffs, NJ, USA, 2002.

[2] Michael Greiffenhagen, Visvanathan Ramesh, and Heinrich Niemann. The systematic design and analysis cycle of a vision system: A case study in video surveillance. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages II:704–711, 2001.

[3] Thanarat Horprasert, David Harwood, and Larry S. Davis. A statistical approach for real–time robust background subtraction and shadow detection. In *Proc. IEEE ICCV'99 FRAME-RATE Workshop*, pages II:751–767, 1999.

[4] Jun-Wei Hsieh, Chia-Jung Chang, Wen-Fong Hu, and Yung-Sheng Chen. Shadow elimination for effective moving object detection by Gaussian shadow modeling. *Image and Vision Computing*, 21(6):505–516, 2003.

[5] Omar Javed, Khurram Shafique, and Mubarak Shah. A hierarchical approach to robust background subtraction using color and gradient information. In *Proc. IEEE Workshop on Motion and Video Computing (WMVC)*, pages 22–27, 2002.

[6] Omar Javed and Mubarak Shah. Tracking and object classification for automated surveillance. In *Proc. European Conf. Computer Vision (ECCV)*, pages IV:343–357, 2002.

[7] Pakorn KaewTraKulPong and Richard Bowden. An improved adaptive background mixture model for real–time tracking with shadow detection. In *Proc. European Workshop Advanced Video Based Surveillance Systems (AVBS)*, pages 149–158, 2001.

[8] Sohail Nadimi and Bir Bhanu. Physical models for moving shadow and object detection in video. *IEEE Trans. Pattern Anal. Machine Intell.*, 26(8):1079–1087, 2004.

[9] Naoya Ohta. A statistical approach to background subtraction for surveillance systems. In *Proc. IEEE Int'l Conf. Computer Vision (ICCV)*, pages II:751–767, 2001.

[10] Andrea Prati, Ivana Mikic, Mohan M. Trivedi, and Rita Cucchiara. Detecting moving shadows: Formulation, algorithms and evaluation. *IEEE Trans. Pattern Anal. Machine Intell.*, 25(7):918–923, 2003.

[11] Foster J. Provost and Tom Fawcett. Robust classification for imprecise environments. *Machine Learning*, 42(3):203–231, 2001.

[12] M.J.J. Scott, M. Niranjan, and Richard W. Prager. Realisable classifiers: Improving operating performance on variable cost problems. In *Proc. Brittisch Machine Vision Conf. (BMVC)*, pages 306–315, 1998.

[13] Paul J. Withagen, Frans C.A. Groen, and Klamer Schutte. CCD characterization for a range of color cameras. In *Proc. IEEE Instrumentation and Measurement Technical Conference (IMTC)*, pages III:2232–2235, 2005.

[14] Paul J. Withagen, Klamer Schutte, and Frans C.A. Groen. A probabilistic framework for object detection and tracking. *Int'l J. Computer Vision*, Under review, 2005.
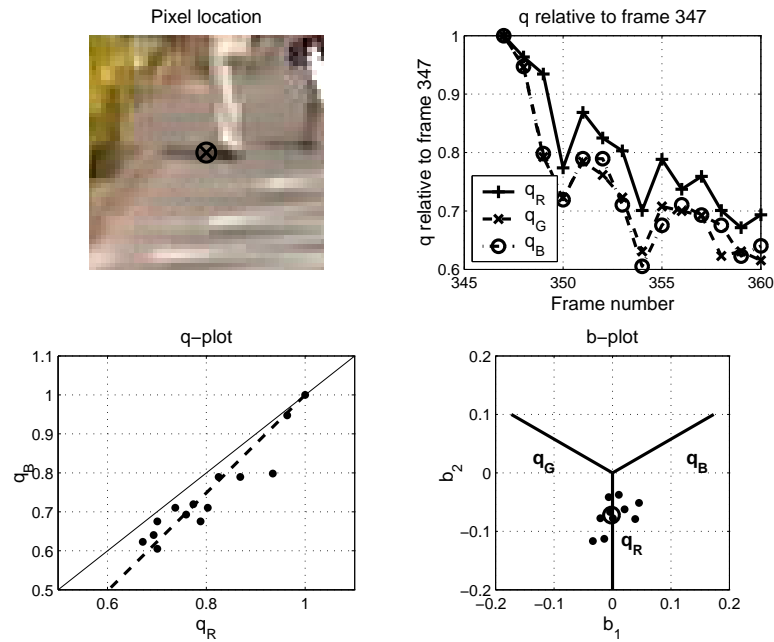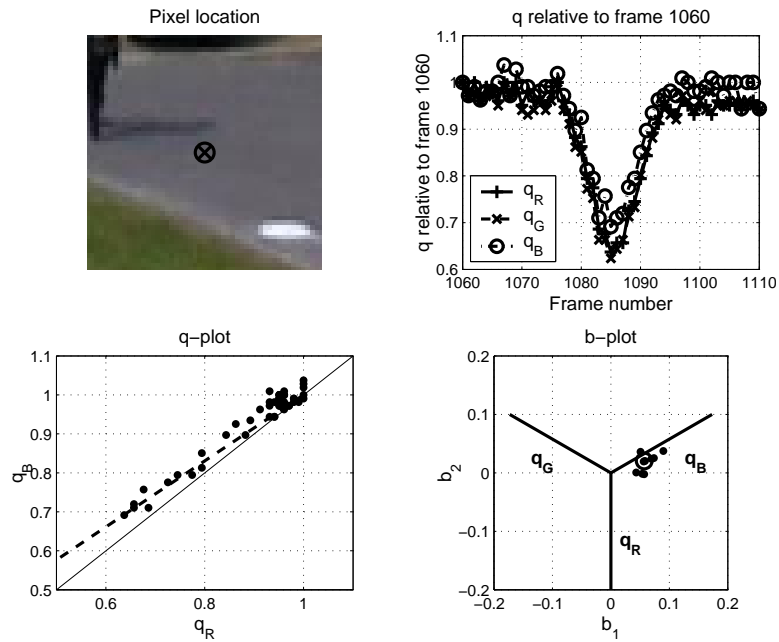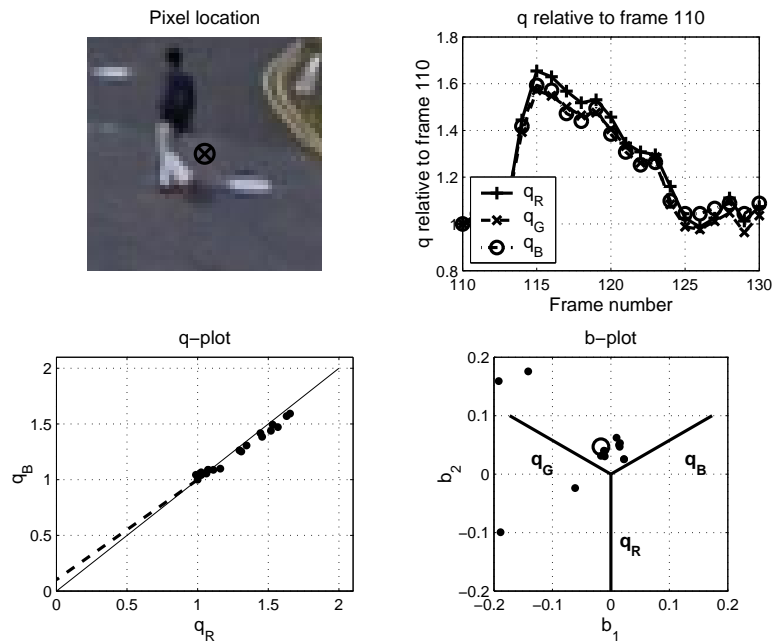
(a) Intratuin, shadow
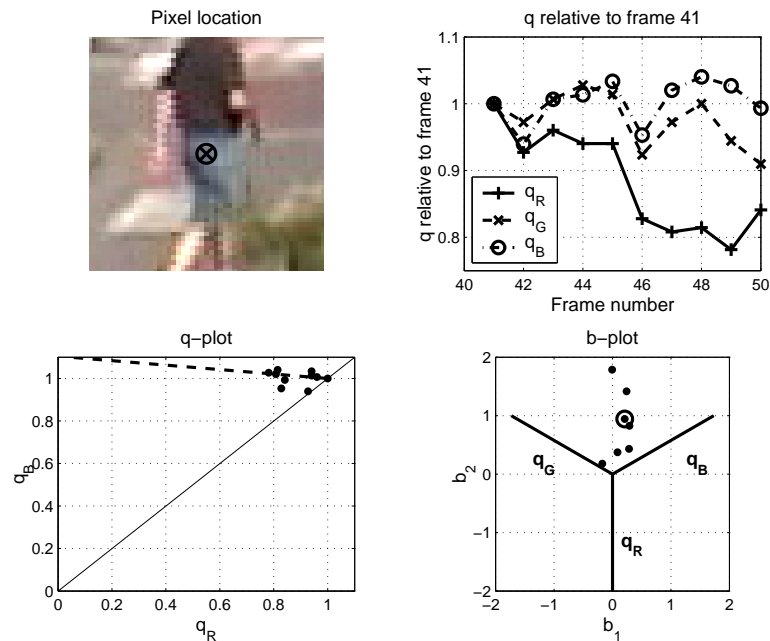


(b) Intratuin, object

**Figure 5:** Examples of the $b$–space. Top left shows the location of the pixel. The $q$–values relative to a frame without shadow are plotted in the figure on the top-right. The bottom left figure shows a plot of $q$ for two dimensions. The $b$–space is given in the bottom right figure. A circle in this plot depicts the median of all data points. This value of $b$ defines the dashed line in the bottom-left plot.

(a) PETS01-3TR1, shadow



(b) PETS01-3TR1, object

**Figure 6:** Examples of the $b$–space. Top left shows the location of the pixel. The $q$–values relative to a frame without shadow are plotted in the figure on the top-right. The bottom left figure shows a plot of $q$ for two dimensions. The $b$–space is given in the bottom right figure. A circle in this plot depicts the median of all data points. This value of $b$ defines the dashed line in the bottom-left plot.
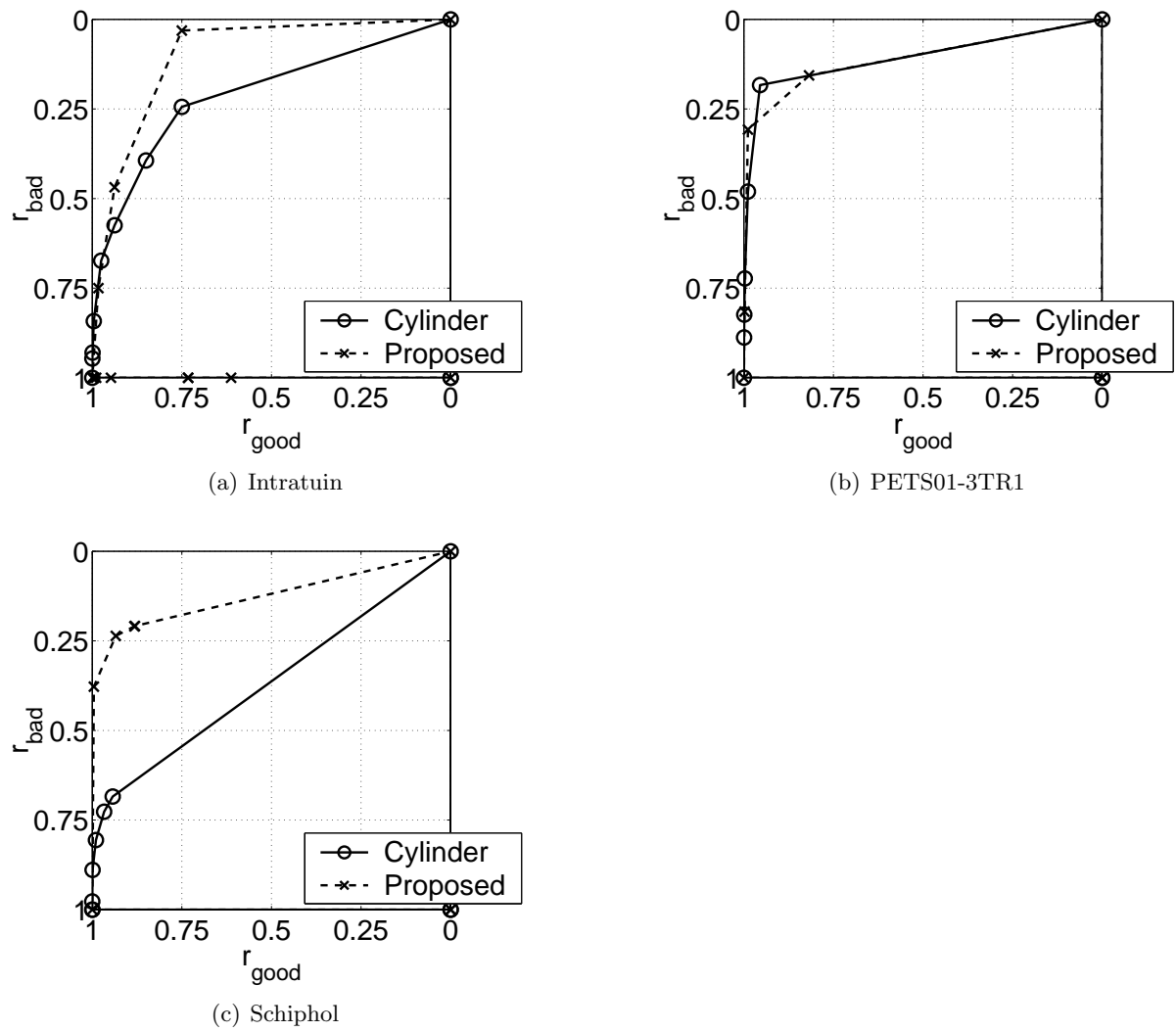
(a) Intratuin

(b) PETS01-3TR1

(c) Schiphol

**Figure 7:** Good/bad–plots. Shown is the convex hull of results with different parameter settings for both foreground/background classification and shadow removal.

# Acknowledgements

While this research was conducted, PW was affiliated with [2] and [3]

## IAS reports

This report is in the series of IAS technical reports. The series editor is Bas Terwijn (`bterwijn@science.uva.nl`). Within this series the following titles appeared:

P.J. Withagen and K. Schutte and F.C.A. Groen *Global intensity correction in dynamic scenes.* Technical Report IAS-UVA-07-01, Informatics Institute, University of Amsterdam, The Netherlands, January 2007.

A. Noulas and B. Kröse *Probabilistic audio visual sensor fusion for speaker detection.* Technical Report IAS-UVA-06-04, Informatics Institute, University of Amsterdam, The Netherlands, March 2006.

J. Nunnink and G. Pavlin *Fault localization in bayesian networks.* Technical Report IAS-UVA-06-03, Informatics Institute, University of Amsterdam, The Netherlands, May 2006.

All IAS technical reports are available for download at the IAS website, `http://www.science.uva.nl/research/ias/publications/reports/`.