A survey of cast shadow detection algorithms

Nijad Al-Najdawi a, Helmut E. Bez b,*, Jyoti Singhai c, Eran.A. Edirisinghe b

a Department of Information Technology, Prince Abdullah Bin Ghazi Faculty for Science and Information Technology, Al-Balqa’ Applied University, Amman, Jordan
b Department of Computer Science, Loughborough University, UK
c Department of Electronics and Communication Engineering, Maulana Azad National Institute of Technology, Bhopal, India

ABSTRACT

Cast shadows need careful consideration in the development of robust dynamic scene analysis systems. Cast shadow detection is critical for accurate object detection in video streams, and their misclassification can cause errors in segmentation and tracking. Many algorithms for shadow detection have been proposed in the literature; however, a complete, comparative evaluation of existing approaches is lacking. This paper presents a comprehensive survey of shadow detection methods, organised in a novel taxonomy based on object/environment dependency and implementation domain. In addition, a comparative evaluation of representative algorithms, based on quantitative and qualitative metrics, is presented to evaluate the algorithms on a benchmark suite of indoor and outdoor video sequences.

1. Introduction

Computer vision systems are often required to differentiate between objects and their shadows. Consequently, shadow detection is useful in many applications including: scene interpretation, image segmentation and object recognition and tracking. Shadows are a major issue for object recognition in video sequences – as a shadow has similar dynamics to the object it is cast by. Further, shadow points are easily misclassified as foreground since they typically differ significantly from the background. For these reasons, object recognition, shadow identification is critical for both image sequences (video) and still images. Automated video surveillance systems require mechanisms for tracking objects in the field of view. In object tracking, cast shadows can be classified as objects – due to their visual characteristics. Hence the misclassification of shadows may result in object merging and shape alteration, which may cause significant confusion to the tracking system (Prati et al., 2003). A number of shadow detection algorithms have been proposed in literature, based on physical, geometrical, and heuristic techniques. A survey on moving cast shadow detection, conducted by Prati et al. (2003), classified algorithms using two layer taxonomy – i.e., deterministic and statistical approaches, based on whether the decision process introduces and exploits uncertainty. Alternative ways of classifying cast shadow detection algorithms exist, and this paper proposes the novel four-layer taxonomy, which is complementary to that of Prati’s, shown in Fig. 1. The proposed taxonomy is based on object/environment dependency and the implementation domain of the algorithm. The first two layers of the taxonomy are concerned with object and environment dependencies, whilst layers three and four relate to the implementation domain – i.e., pixel vs transform domain and monochrome vs colour space. The proposed new taxonomy will assist in categorising and comparing the capabilities of existing algorithms for particular applications and in distinct implementation domains – i.e., algorithms designed for detecting cast shadows of a particular object (e.g., human/vehicle) in a given environment (e.g., indoor/outdoor/aerial/morphological images), or algorithms independent of object type or environment type.

There is a large volume of literature on shadow detection and it is not practical to detail the precise contribution of each publication; the survey conducted in this paper is therefore based on a selection of key papers representative of the distinct approaches to shadow detection that have been proposed. For clarity of presentation, the paper is organised as follows: Section 2 introduces a widely used cast shadow modelling hypothesis, Section 3 discusses the approaches to cast shadow detection found in the literature and Section 4 presents a comprehensive performance evaluation of the algorithms. Based on the proposed four-layer taxonomy discussed above, the material of Section 3 is further divided into sub-Sections 3.1, 3.2, and 3.3.
2. Shadow modelling hypothesis

2.1. Shadow classification

A shadow is a region of relative darkness that occurs when an object totally, or partially, occludes direct light from a light source. The shadow generated by an object may be classified as either ‘self’ or ‘cast’: self shadow occurs on the object occluding the light, whilst cast shadow is that generated, by the object, on other objects in the scene. A cast shadow may be further sub-divided into umbra and penumbra regions – see Fig. 2. The umbra (Latin: “shadow”) is the darkest part of a shadow. Within the umbra, the light source is completely blocked by the object casting the shadow. The penumbra (Latin: “almost-shadow”) is that part of the shadow where the light source is partially blocked. Penumbras occur only when the light source is not a point-source.

In many computer vision applications, e.g., surveillance, object recognition and object tracking the detection of self shadow is of less importance – the primary objective being the segmentation of an image into object, object cast-shadow and background. In this situation it is clearly appropriate to classify self-shadow pixels as object pixels. Fig. 3 illustrates such segmentation in a video sequence; image (a) shows a background frame, in image (b) a human figure (object) is present and casts a shadow onto the background. Image (c) shows the cast-shadow region detected by the authors (Al-Najdawi et al., 2006) cast-shadow detection process. Frames (a) and (c) enable each pixel of (b) to be classified as either background, object or cast-shadow. The self-shadow pixels of the object which are present, for example on the garments, are not identified as cast-shadow.

An aspect of digital photography is the enhancement of still images. In this topic the determination and processing of self-shadow may also be important in improving the visual quality of images, as illustrated in Fig. 4.

2.2. Reflection, illumination and shadow model

This section provides an introduction to the reflection, illumination and shadow model that forms the basis of many shadow detection algorithms. It also presents a summary of the basic assumptions (Stauder et al., 1999) of many methods.

When digital cameras capture shadows in a scene, it is the light reflected from the surface that records that part of the scene. Therefore, the luminance (brightness) of a point \(q\) at the 2D image position \((x, y)\) and time instant \(t\), can be described by the following reflection model: \(\psi_t(x, y) = \zeta_t(x, y) \rho_q(x, y)\)

where \(\rho_q(x, y)\) is the reflectance of the object surface at \(q\), i.e., the reflection coefficient, and \(\zeta_t(x, y)\) is the irradiance (illumination), i.e., the amount of luminance energy received at \(q\); \(\zeta_t(x, y)\) is a function of the direction \(L_{x, y}\) of the light source with respect to the object surface normal \(N_{x, y}\), the intensity of the direct light \(I_d\) and the ambient light \(I_a\) received at point \(q\). The illumination, \(\zeta_t\), when in or out of a shadow has been modelled as:

\[\zeta_t = k \left( \frac{1}{1 + \left( \frac{d}{D} \right)^2} \right)^n\]

where \(k\) is a constant, \(d\) the distance from the light source to the shadow edge, \(D\) the maximum distance from the light source to any point on the surface, and \(n\) the illumination exponent.

---

1. When considering the following model, \((x, y)\) corresponds to the 2D projection of the 3D environment. Thus \(q\), which is \((x, y, z)\) projects to \((x, y)\).

2. Assuming the illumination spectrum is constant for each wavelength (white illumination) and matte surfaces.
The above model, which is based on Lambert’s cosine law, describes illumination both before and after a shadow is cast and is called the illumination and shadow model. The term \( \lambda_x \leq \lambda_y \leq 1 \) describes the transition inside the penumbra, and depends on the light source and scene geometry. In addition to the above model, the following general assumptions are made by many of the existing shadow detection methods:

**Assumption 1.** The light source intensity \( c_P \) is high; i.e., the frame difference, at a pixel, due to a moving cast shadow will be large.

**Assumption 2.** Camera and background are static.\(^3\)

**Assumption 3.** The background is plane, and light source position is at a distant from the background.

\[
\zeta_t(x, y) = \begin{cases} 
  c_P n_{xy} \cdot L_{xy} + c_A & \text{no object (noshadow)} \\
  \lambda_x c_p n_{xy} \cdot L_{xy} + c_A & \text{penumbra} \\
  c_A & \text{umbra}
\end{cases}
\]

3 Static camera requires a frame taken before the object enters the scene, and another frame taken after the object enters the same scene. Static background requires a scene illumination that does not change overtime, otherwise dynamic background generation algorithms would be required.

**Assumption 4.** The distance between the moving object and the background is not negligible compared to the distance between the light source and the object.

Consider a background pixel at \((x,y)\) and assume that the pixel is outside a cast shadow at time instant \( t_1 \) and inside a cast shadow at time instant \( t_2 \). It follows that, if assumptions 1 hold i.e., \( c_P \) is high at time \( t_1 \), then the difference \( \zeta_t(x, y) = \zeta_{t_2}(x, y) - \zeta_{t_1}(x, y) \) will be high. Note that, the reflectance of a static background does not change with time, thus \( \rho_s(x, y) = \rho_s(x, y) \) holds. If both assumptions 1 and 2 hold, the result of the difference equation will be high in the presence of cast shadows covering a static background. This implies (as assumed in many other approaches) that shadow points can be obtained by thresholding the frame difference image.

### 3. Cast shadow detection

In this section, algorithms in each category of the proposed four-layer taxonomy introduced in Section 1 are described in some detail. Section 3.1 presents analysis of object dependent algorithms and Sections 3.2 and 3.3 present analysis of object independent algorithms in environment dependent and independent scenario respectively. These sections are further subdivided based on implementation domain, i.e., pixel domain or transform domain.

#### 3.1. Object dependent algorithms

This section discusses algorithms, which are designed and implemented to detect shadows of a particular type of object (vehicle/human cast shadows) in a given environment (indoor/outdoor/aerial/morphological images).

##### 3.1.1. Pixel domain methods

Due to the fundamental differences in the algorithm used, these approaches are categorised into monochrome and colour domains.

3.1.1.1. Monochrome domain methods. Onoguchi (1998) proposed a deterministic model based approach. In his work, the author presented a method for eliminating pedestrian shadows. The proposed method removes the shadow areas using height information, since...
most of the shadow areas accompanying moving objects are assumed to be on the plane of the road. Two cameras are set at locations so that their shared visual fields include the surveillance area. The image obtained from one of the cameras is inverted and projected to the road plane and the projected image on the road plane is transformed to the view from the other camera. Shadows existing on the road plane occupy the same areas in the transformed image and in the image acquired from the other camera, whereas objects areas with different heights from the road plane occupy different areas in these images. Therefore, shadow areas can be removed by subtracting these images. The algorithm requires shadows to be on a flat road plane. Further, objects and shadows must be visible to both cameras, and the method requires manual registration and objects’ height.

Hsieh et al. (2003) proposed a deterministic model based approach, similar to Onoguchi’s (1998), that represents an algorithm for eliminating shadows of multiple pedestrians using Gaussian shadow modelling. First, a set of moving regions is segmented from the static background using a background subtraction technique. For moving cast shadow detection, a histogram-based method is proposed for isolating each pedestrian from the extracted moving region. Based on the results, a coarse-to-fine shadow modelling process is then applied for eliminating the shadow from the detected pedestrian. At the coarse stage, a moment-based method is first used for obtaining the rough shadow boundaries. Then, the rough approximation of the shadow region can be further refined through Gaussian shadow modelling. The chosen shadow model is parameterised with several features including: the orientation, mean intensity, and centre position of a shadow region. The novelty of the method, comes from the fact that it uses vertical and horizontal image projection of binary silhouettes, and finds the points where feet and shadow intersect. However, the algorithm requires knowledge of the light source, it works only for human objects and requires the shadow to be on the ground.

Yoneyama et al. (2003) proposed another deterministic model based approach, to eliminate moving cast shadows based on a simplified 2D vehicle/shadow model of six types projected to a 2D image plane. The parameters of vehicle and shadow models are estimated from the input video without the light source and camera calibration information. Distinguishing the cast shadow region from the vehicle itself is done via the determination of parameters of the joint model.

Bevilacqua (2003) proposed an algorithm to detect moving shadows in the context of an outdoor traffic scene, for visual surveillance purposes. The algorithm exploit some foreground photometric properties concerning shadows. The proposed method is based on multi-gradient operations applied on the division image (the division image between the current frame and the background of the scene) which aim to find the most likely shadow regions. Further, a binary edge matching is performed on the background of the scene) which aim to find the most likely shadow regions. The important characteristics shadows possess in both intensity and geometry, the important gradient values of shadow regions are obtained, in RGB colour space, by convolution of the image with different filters. Then a shadow detection algorithm (based on partial differential equations) is applied, which takes the gradient values as the parameter for edge detection. The experimental results indicate that the boundaries of segmented shadow regions are preserved well, and the information in non-shadow region remains unaffected. The algorithm is iterative, and performance is dependent on the sharpness of extracted images – i.e., the 2D filters used. The algorithm generates noise in the regions from which shadows have been removed.

Lalonde et al. (2010) presents a practical algorithm for automatically detecting shadows cast onto the ground in a single image of ‘consumer’ quality. Their key hypothesis is that the types of materials constituting the ground in outdoor scenes is relatively limited – most commonly asphalt, brick, stone, mud, grass, concrete etc. – and that a system can therefore be trained on a limited set of suitable images. The shadow detector proposed is a three-tier process, comprising: (i) training a decision tree classifier, (ii) a Conditional Random Field (CRF)-based optimisation and (iii) the incorporation of an existing classifier specifically trained to detect ground in images. In the first stage, the classifier is trained on a set of shadow sensitive features like ratios of brightness and colour filter responses at different scales and orientations, and texture and intensity distribution on both sides of the edge. These features are used with a trained decision classifier to detect a shadow edge. In the second stage, detected shadow edges are grouped to generate coherent shadow contours using the CRF, connecting likely shadow edges and removing isolated/spurious edges and T-junctions. The CRF is expressed as log-likelihood of a particular labelling y (i.e., the assignment of shadow/non-shadow to each boundary) given observed data x as a sum of unary $\phi_i(y_i)$ and pair-wise potentials $\psi_{ij}(y_i, y_j)$:

$$-\log P(y|x; \lambda, \beta) = \lambda \sum_{i \in B} \phi_i(y_i) + \sum_{(ij) \in E} \psi_{ij}(y_i, y_j) - \log Z_{\lambda, \beta}$$

where $B$ is the set of boundaries, $E$ the set of edges between them, and $\lambda$ and $\beta$ are model parameters. $Z_{\lambda, \beta}$ is the partition function that depends on the parameters $\lambda$ and $\beta$, assigned values of 0.5 and 16 respectively, but not on the labelling $y$ itself. In the third stage, a global scene layout descriptor is incorporated within CRF to estimate the location of the ground pixels. This helps in reducing false positive (non-shadow) detections outside the ground. A key claim for the algorithm is its ability to detect shadow in low definition images. Although shadow detection is automatic for a single image, it detects shadows on the ground only. Self-shadows and shadows on vertical surfaces are not detected. In addition, its efficiency...
depends on training – which increases the computational requirement.

3.1.2. Transform domain methods

Shen et al. (2007) proposed an algorithm, based on heuristics, to detect and eliminate the human cast shadow. Moving region geometrical analysis of the human body (based on human body geometrical properties) with the help of Hough Transform is used to determine the shadow existence and its approximate location. Image-orientation information measure is used to detect the shadow. The frame ratio image is segmented into shadow-like and non-shadow-like regions. Finally the approximate shadow location and the shadow-like region are fused to form the fine shadow region, and the shadow is then eliminated. The limitation of this algorithm is that it eliminates shadow of only objects with known geometrical properties and image orientation. The complexity of the algorithm increases with multiple objects of varying dimension, and with multiple sources of illumination.

3.2. Object independent and environment dependent algorithms

This section discusses algorithms which are designed and implemented for detecting shadows casted in a particular environment (indoor/outdoor/aerial/morphological images), but are independent of object type.

3.2.1. Pixel domain methods

3.2.1.1. Monochrome domain methods. The work of Stauder et al. (1999)\(^4\) is based on the reflection model introduced in Section 2. Given a video sequence the authors exploit the local changes due to shadow by computing the ratio \(\zeta(x,y)\) of the pixel in the actual frame (the frame with shadow) at \((x,y)\), with the pixel value at \((x,y)\) in a reference frame (the frame with no shadow) as:

\[
\frac{\zeta(x,y)}{\zeta_{\tau_1}(x,y) \leq 1}.
\]

Using the reflection model \(\delta(x,y) = \zeta(x,y)\rho(x,y)\) and assuming constant reflectance through time \(\tau_1\) and \(\tau_2\), i.e., \(\rho_{\tau_1}(x,y) = \rho_{\tau_2}(x,y)\), it follows that:

\[
\frac{\zeta_{\tau_2}(x,y)}{\zeta_{\tau_1}(x,y) \leq 1}.
\]

Therefore, by using the illumination and shadow model \(\zeta(x,y)\) (see Section 2), the ratio \(\zeta(x,y)\) can be written as:

\[
\zeta_{\tau_1}(x,y) = \frac{\zeta_{\tau_2}(x,y)}{\zeta_{\tau_1}(x,y) \leq 1} \text{ i.e., } \frac{\zeta_{\tau_2}(x,y)}{\zeta_{\tau_1}(x,y) \leq 1} \text{ i.e., } \frac{\zeta_{\tau_2}(x,y)}{\zeta_{\tau_1}(x,y) \leq 1}.
\]

Moreover, following Assumption 3 of Section 2, Stauder et al. assumed \(N_{k,y}\) is spatially constant in a neighbourhood of the point. Thus, the pixel is marked as ‘possible shadow’. The authors use various heuristic techniques in order to exploit all four assumptions (such as edge detection and gradient calculation). Results show an excellent detection and removal of indoor shadows. However, the limitations come from the fact that the approach is not applicable for outdoor shadows (outdoor shadows are harder to detect), and the fact that it requires the background to be of a uniform colour.

By ignoring the shadow penumbra, Toth et al. (2004) used the following simplified version of the illumination and shadow model described in Section 2:

\[
\zeta_{\tau_1}(x,y) = \begin{cases} 
\frac{c_p N_{k,y} - L_{k,y} + c_A}{c_A} & \text{no object} \\
\text{umbra} & \text{umbra}
\end{cases}
\]

\(^4\) This work forms the basis of many other algorithms proposed in literature.

and defined the ratio between the shadow region in frame \(I_s\) and its corresponding region in the background frame \(I_b\) (where it is illuminated), as follows:

\[
k(x,y) = \frac{I_s(x,y)}{I_b(x,y)} = \frac{c_A}{c_p N_{k,y} - L_{k,y} + c_A} \
\leq 1 \text{ i.e., } \frac{\text{shadow}}{\text{illumination/background}}.
\]

Since background reflectance does not change with time, the authors assumed that \(k(x,y)\) represents the ideal case without any noise. They also assumed that images are corrupted with Gaussian white noise. Accordingly, their background images were modelled as follows:

\[
\hat{I}_b = I_b + \epsilon(x,y) \quad \text{where } \epsilon(x,y) \sim N(0, \sigma^2)
\]

with \(\sigma^2\) being the camera noise, which is either known or can be estimated. Hence, for a shadow point in the foreground image, it can be assumed that:

\[
\hat{I}_s(x,y) = k(x,y) \cdot I_b(x,y) + \epsilon(x,y).
\]

The authors then apply thresholds to classify a point as a non-shadow pixel or as a shadow point.

Chien et al. (2002) proposed a moving object segmentation algorithm for real-time applications. A background registration technique is used to construct a background image from the accumulated frame difference information. The moving object region is separated from the background region by comparing the current frame with the constructed background image. Finally, a post-processing step is applied on the resulting object mask to remove noise regions and to smooth the object boundary. Morphological gradient operations are used to filter out the shadow area while preserving the object shape. However, the use of morphological gradient operations can reduce the effects of insignificant indoor shadows only, and not the effects of outdoor shadows, or significant indoor shadows.

Xu et al. (2005, 2004) assumed that Stauder’s approach generates many false negative edges, as he considered the moving edges as static edges. They proposed an alternative method of moving cast shadow detection and removal, in normal indoor scenes where the hypothesis and the general assumptions in Section 2 hold. Their shadow detection and removal algorithm includes: the generation of initial Change Detection Masks (CDM), shadow region detection by multi-frame integration, edge matching and region growing, and finally shadow region removal and post-processing for eliminating noise and tuning object boundaries. The results were compared with the related gradient filter approach proposed by Chien et al. (2002). Results have proved the high efficiency of the algorithm for shadow detection. However, the method is intended for insignificant indoor shadows, and is only applicable for indoor environments.

Jacques et al. (2005) proposed that in shadow regions it is expected that a certain fraction of incoming light is blocked. The authors assume that the observed intensity of shadow pixels is directly proportional to incident light; consequently, shadow pixels are scaled versions (darker) of corresponding pixels in the background model. The Normalised Cross-Correlation (NCC) is used to detect shadow pixel candidates. The NCC is used as an initial step for shadow detection, followed by a refinement process using local statistics of pixel ratios. Let \(B\) be the background frame, and \(I\) be the current frame of the video sequence. For each foreground pixel \((x,y)\) a \((2N + 1) \times (2N + 1)\) neighbourhood is defined by \(T_{ios}(n,m) = I(x + n,y + m)\), for \(-N \leq n \leq N, -N \leq m \leq N\). The NCC between \(T_{ios}\) and image \(B(x,y)\) is given by:
\[
\text{NCC}(x, y) = \frac{E_b(x, y)}{E_b(x, y)E_f(x, y)}
\]

where \(E_b\), \(E_R\) and \(E_f\) are defined as:
\[
E_b(x, y) = \sum_{n=-N}^{N} \sum_{m=-N}^{N} B(x + n, y + m) T_{xy}(n, m)
\]
\[
E_R(x, y) = \sum_{n=-N}^{N} \sum_{m=-N}^{N} R(x + n, y + m) T_{xy}(n, m)^2
\]
\[
E_f(x, y) = \sum_{n=-N}^{N} \sum_{m=-N}^{N} T_{xy}(n, m)^2
\]

A pixel at position \((x, y)\) is pre-classified as being in a shadow if:
\[
\text{NCC}(x, y) = L_{\text{arc}} \quad \text{and} \quad E_f < E_b(x, y)
\]

where \(L_{\text{arc}}\) is a fixed threshold. The proposed refinement stage holds if the ratio \(\text{I}_{\text{arc}}(x, y) / \text{R}(x, y)\) in a neighbourhood around each shadow pixel candidate is approximately constant, by computing the standard deviation of \(\text{I}_{\text{arc}}(x, y) / \text{R}(x, y)\) within the neighbourhood. Although the results show a good classification of shadow regions in indoor environments, it is shown in the results that there is a large number of misclassified pixels in weak shadow areas, where the authors apply a morphological operator to remove the misclassifications. The authors acknowledge that in outdoor environments, i.e., with strong shadows, the algorithm fails and shadows will be misclassified as foreground objects.

Nicolas (2005) presented a scalable block-based video compression scheme for video-surveillance applications. Each block is classified, according to its content, as background, foreground, or cast shadow. Using Assumption 3 in Section 2, the author assumed that the shadow ratio between a shaded point in the image \(I_t\) and the same illuminated point in the reference frame \(I_{\text{ref}}\) can be expressed as:
\[
R_t(q) = \frac{I_t(q)}{I_{\text{ref}}(q)} + \eta_t
\]

where \(\eta_t\) is a constant and \(I_t(q)\) is the intensity of the ambient light at point \(q\) and time \(t\). \(I_{\text{ref}}(q)\) is the intensity of the ambient light at a point \(q\) in the reference frame. The method is not considered as a complete physical model for shadows and illumination, since the direct light received at a point \(q\) is totally ignored. Even with a distant light source, direct light is still received at a point \(q\) unless blocked by an object in the scene. Moreover, no results are reported.

Jung (2009) presents a background subtraction and shadow removal algorithm for processing grey scale video sequences, using a statistical approach combined with geometrical constraints. In this algorithm, a background model using a metrically trimmed mean is extracted in a training stage, and foreground pixels are obtained in the operation (or test) stage. The statistical model is combined with expected geometrical properties for shadow identification and removal. Finally, morphological operators are applied to remove isolated foreground pixels. A small neighbourhood \(\Omega(x)\) around each pixel \(x\), belonging to the foreground are probably far from the estimated metrically trimmed mean \(\lambda(u)\) of the distribution. Hence a pixel \(x\) is assigned to the foreground if
\[
\sum_{u \in \Omega(x)} w(u) |I_t(u) - \lambda(u)| > k \sum_{u \in \Omega(x)} w(u) \sigma(u)
\]

where \(w(u)\) is the weighing mask of for each pixel \(u\) of \(\Omega(x)\), \(\lambda(u)\) is \(\lambda\)-metrically trimmed mean and \(k\) controls the maximum allowed deviation from the mean with respect to the standard deviation. The \(\lambda\)-metrically trimmed mean value, \(\lambda(u)\), for pixel \(x\) using \(\lambda = 0.3\) for the \(T\) image frames, \(\{I_t(x), \ldots, I_{T}(x)\}\), used in the training period, is given by
\[
\lambda_\lambda(x) = \frac{1}{T - \lfloor xT \rfloor} \sum_{t = \lfloor xT \rfloor}^{T} I_t(x)
\]

where \(S(x) = \{t: f(x, t) \in T - [xT]\}\).

In practical applications the detection of foreground pixels, using relation (1), may produce some isolated pixels, or holes in the interior of valid objects. Thus sequentially, an opening and a closing morphological operator is applied. In such background subtraction algorithms, shadows are undesirably detected as foreground objects. To detect shadows they assumed that the intensity \(I_t(x)\) of a pixel \(x\) in a shadowed region is a scaled version of the background model plus additive Gaussian noise, i.e.,
\[
I_t(x) = \alpha(x) \hat{I}(x) + \eta(x), \quad \text{where} \quad \eta(x) \sim \mathcal{N}(0, \sigma^2(x))
\]

Here \(0 < \alpha(x) < 1\) relates to the intensity of the shadow and \(\hat{I}(x)\) is used from the background model. It is assumed that \(\alpha(x) \approx \alpha\) is constant within a small neighbourhood \(\Omega(x)\) of the penumbra centred at pixel \(x\), then
\[
R(x) = \frac{I_t(x)}{\hat{I}(x)} = v(x)
\]

where \(v(x) = \alpha + \eta(x) / \hat{I}(x) \sim \mathcal{N}(0, \sigma^2(x) / \hat{I}(x))^2\). Also \(\mu^2(x)\) and \(\sigma^2(x)\) are the mean and standard deviation of \(R(x)\) within neighbourhood \(\Omega(x)\). The pixel \(x\) is identified as shadow if
\[
S(x) = 0 \quad \text{if} \quad |D(x)| > k_s^\mu \sigma^2(x), \quad \text{and} \quad S(x) = 1 \quad \text{otherwise}, \quad k_s^\mu \text{is constant depending on the confidence level, and} \quad \Omega_s \text{is a suitably defined subset of} \quad \Omega.
\]

This algorithm is too complex, iterative and requires training and accurate assumptions of many parameters like \(w(u), \alpha, k_s^\mu\) etc. These limitations make this algorithm image dependent, environment dependent and training dependent. It fails in the presence of multiple illumination sources and in outdoor environment with strong shadow.

3.2.1.2. Colour domain methods. Elgammal and Harwood (1999) define a local assumption on the ratio between shadow and non-shadow point luminance. The approach uses colour information to suppress non-shadow points from being detected, by separating colour information from lightness information. Given the colour variables, \(R, G, B\), the chromaticity coordinates \(r, g, b\) can be calculated as \(r = \frac{G}{G + B}, g = \frac{B}{R + B}, b = \frac{R}{R + G}\), and the lightness as \(s = \frac{R + G + B}{2}\). The work assumes that the use of chromaticity coordinates in shadow detection has the advantage of being more insensitive to small changes in illumination due to shadows. The method starts from Assumption 3 (see Section 2), and its basic idea is as follows: let the expected value for a pixel be \((r, g, s)\). Assume that this pixel is covered by a shadow in frame \(t\) and let \((r_s, g_s, s_s)\) be the observed value for this pixel at this frame. Then, using: \(\alpha \leq s_s \leq b \leq 1\), it is expected that the observed value \(s_s\) will be darker than the normal value \(s\) up to a certain limit. The work also assumes that a similar effect is expected for a highlighted background, where the observed value is brighter than the expected value, up to a certain limit.

Horprasert et al. (1999) proposed a similar approach to Elgammal and Harwood (1999). Their work also assumes that shadows have similar chromaticity and lower brightness in comparison to the same pixel brightness in the background image. The algorithm is also based on the proposed computational colour model, which separates the brightness from the chromaticity component. Reported results show a good detection of shadows in indoor environments, and shadows in outdoor environments in overcast situations, where cast shadows are weak. It is observed
that as the shadow gets stronger, pixels tend to be increasingly identified as foreground pixels.

Javed and Shah (2002) assume that pixels in the shadow regions are darker than those in the reference background, and that shadows retain some texture and colour information of the underlying surface under general viewing conditions. In their work, all foreground regions in the image that are darker than the corresponding regions in the reference image are extracted. The algorithm then performs colour segmentation on the extracted regions. The algorithm uses the K-means approximation to perform colour segmentation. Each pixel value in a potential shadow region is checked against existing K Gaussian distributions until a match is found. Reported results show a good classification of indoor shadows.

Siala et al. (2004) applied the illumination and shadow model, described in Section 2. The authors proposed that the distortion between a background image $I_{bg}$ and a current image $I_t$, where $t$ denotes time, of a video-surveillance sequence expressed in the RGB colour space, can be approximated for shadow regions by:

$$R_{sh} = d_{Rsh} G_{sh}, B_{sh} = d_{Bsh} B_{sh},$$

where $R_{sh}, G_{sh}, B_{sh}$ and $R_{bg}, G_{bg}, B_{bg}$ are respectively the RGB colour values of shadow pixel in $I_t$ and non-shadow pixel in $I_{bg}$ and the colour ratios $d_{R} = \frac{R_{bg}}{R_{sh}} \leq 1, d_{G} = \frac{G_{bg}}{G_{sh}} \leq 1, d_{B} = \frac{B_{bg}}{B_{sh}} \leq 1$. To detect shadows, the authors apply a learning stage, where a representative image containing the three classes: foreground, moving shadow, and background, is arbitrarily selected. The moving shadow regions are manually segmented. Colour ratios $d_{R}, d_{G}, d_{B}$ are computed for pixels issued from a bootstrap sample. Although the use of learning methods is computationally exhaustive, it is supposed to give more accurate results. However, the results show a large number of misclassifications.

Tsai (2006) used a transformed invariant colour model to identify shadows, transforming the input RGB image $I_t$ into HSI colour space. The ratio of the hue image ($H_t$) over the intensity image ($I_t$) for each pixel was then calculated to construct the ratio map, $R$, defined by:

$$R(x, y) = \frac{H_t(x, y) + 1}{I_t(x, y) + 1}$$

which is used to identify shadows. Otsu’s thresholding method is then applied to determine global threshold $T$ of the constructed ratio map, which is used for separating shadow and non-shadow pixels.

$$S(x, y) = \begin{cases} 1, & R(x, y) > T, \\ 0, & \text{otherwise} \end{cases}$$

where $S(x, y) = 1$ denotes the shadow pixel at position $(x, y)$. Tsai also introduces a shape preservation process to preserve shape information of objects casting shadows in the shape map $S_b$. After performing the logical AND operation on the shape map $S_b$ and the shadow map $S$, the shape information of objects can be preserved. But the results cited show that algorithm has limited performance when difference in object (non-shadow) region and shadow region is small, and misclassification occurs.

Chung et al. (2009) improved the performance of Tsai’s method using a Successive Thresholding Scheme (STS) to detect shadows more accurately. This algorithm proposes a modified ratio map to stretch the gap between the ratio values of shadow and non-shadow pixels. The modified ratio map $R'$ is defined by:

$$R'(x, y) = \begin{cases} 255 e^{-\frac{(x+y)^2}{2\sigma^2}}, & \text{if } r(x, y) < T, \\ 255, & \text{otherwise} \end{cases}$$

where $T$, and $\sigma$ are determined empirically, and $r$ is given by:

$$r(x, y) = \frac{H_t(x, y)}{I_t(x, y) + 1}$$

The proposed STS applies a global thresholding process to create a coarse-shadow map for classifying the input colour aerial image into the candidate shadow pixels and the non-shadow pixels. In order to detect the true shadow pixels from the candidate shadow pixels a coarse to fine strategy is used. A connected component process is first performed on the candidate shadow pixels for grouping the candidate shadow regions. For each candidate shadow region, the local thresholding process is performed iteratively to extract the true shadow pixels from the candidate shadow region. Finally, for the remaining candidate regions, a fine-shadow determination process is presented to determine whether each remaining candidate shadow pixel is a true shadow pixel or not. This algorithm has performed well for removal of shadows from colour aerial images but it does not consider the shadows of moving objects.

### 3.2.2. Transform domain methods

Amamoto and Fujii (1999)5 described a method for tracking moving vehicles on a road. In the proposed method, the varying region in the monitoring image is derived from the background difference, and is further classified into moving objects, stationary objects, shadows and highlights. Their work is one of the few attempts that have been made to detect shadows in the frequency domain. The work uses the Discrete Cosine Transform (DCT) to detect shadows of vehicles on roads. The authors suggest that the shadow of an object varies the pixel values uniformly in comparison with the background. In addition, within the DCT domain, the authors assume that shadows of moving objects may be identified by their dc values, while moving objects maybe identified by their ac values. Although it is a very simple method, highly applicable for real-time applications, and can be used in the compressed domain, its limitations come from the domain constrained assumptions, that if generalised will fail.

Within the transform domain, citegroupEtemadnia and Alsharif (2003) proposed an approach, which assumes that the illumination component of an image is generally characterised by slow spatial variation, while the reflection component tends to vary abruptly, particularly at the junctions of dissimilar objects. These characteristics lead to associate the low frequency components of the Fourier transform of an image with illumination, and the high frequencies with reflection. A Low-Pass-Filter and a High-Pass-Filter are defined to detect the shadows. Although the method is applicable in the compressed domains, the associated results are obtained in very simple indoor environments, with an insignificant shadow.

### 3.3. Object independent and environment independent algorithms

Whilst most of these methods are intended to be independent, it is clear that they are not totally independent – many have minor assumptions about object/scene geometry, or the spectral distribution of light sources. This section discusses some algorithms which are independent of object types and environment types including illumination conditions and scene geometry.

#### 3.3.1. Pixel domain methods

##### 3.3.1.1. Monochrome domain methods

Nadimi and Bhanu (2004, 2002) presented a multi-stage method, based on a spatio-temporal test that accounts for illumination from both the sun and sky, to detect moving cast shadows in outdoor environments. Each stage of the algorithm removes moving object pixels, which cannot be shadow pixels. The method is independent of object types, models, background-colour and scene geometry. It is also capable of detecting umbra in outdoor scenes. Various experimental results are shown, however the main drawback is that the approach assumes...
the spectral power distribution of each illumination source to be equal.

Lin et al. (2010) propose an algorithm for removing shadows by combining texture and statistical models. They use a Gaussian Mixture Model for background removal and the detection of moving shadows in test images. They also define two indices for characterising non-shadowed regions; one of which indicates the characteristics of edges and the other is characterised by the information in grey scales of images used for building modified darkening factors based on Gaussian models. The algorithm was able to detect and locate the foreground pixels of non-shadow regions in different environments, but was unable to evaluate the performance caused by non-uniform distributions of light reflections in the daytime and failed to show results for indoor scenes and scenes with multiple illuminants.

3.3.1.2. Colour domain methods. Mikic et al. (2000), Trivedi et al. (2000) introduced an algorithm\(^7\) for moving cast shadow detection in traffic scenes. The method uses three sources of information to distinguish between moving cast shadows and their corresponding objects:

- Local information: based on the appearance of the individual pixels, a point covered by a shadow gets darker, its blue component increases and the red component decreases compared to the appearance when illuminated.
- Spatial information: objects and shadows inhabit compact regions in the image.
- Temporal information: object and shadow positions can be predicted from previous frames.

The mean and variance of all three-colour components for each background pixel is calculated (Gaussian distributions are assumed for background and shadow pixels and a uniform distribution is assumed for foreground). The decisions are made for each pixel. The segmentation starts by comparing the feature vector for each pixel (a three-dimensional vector of R, G and B colour components) with the mean at that location in the background model. If not significantly different, the pixel is classified into the background class.

Otherwise, prior probabilities are assigned to that location. Colour features and the neighbourhood information are used to produce smoother classifications. Temporal information is used by modifying class prior probabilities, based on predictions from the previous frame. These two methods show excellent results and performance even in complex environments. However, they require information about the position of the sun with respect to the camera.

Cucchiara et al. (2003, 2004)\(^7\) exploit a similar concept to the work of Elgammal and Harwood (1999) and Horprasert et al. (1999). They present a technique for shadow detection and suppression in moving visual object detection and tracking. The novelty of the shadow detection technique is that the analysis is carried out in Hue-Saturation-Value (HSV) colour space, to improve the accuracy in detecting shadows. Estimate of how occlusion due to shadow changes the values of H, S and V is presented. Each pixel from objects resulting from the segmentation step is classified as shadow if the hue and saturation components have changed within certain strict, experimentally determined and environmentally dependent, limits. The difference in saturation must be an absolute difference, while the difference in hue is an angular difference. Thus a shadow mask \( S^p \) is defined for each point \( p \), resulting from motion segmentation, based on the following shadow model:

\[
S^p(p) = \begin{cases} 
1 & \text{if } x \leq \frac{p^V}{y^V}, y^V \leq \beta \land |f^p(p) - B^p(p)| < \tau_S \land \Delta H < \tau_H; \ x, \beta \in [0, 1], \\
0 & \text{otherwise} 
\end{cases}
\]

where (i) \( D_H = \min(|f^p(p) - H - B^p(p)H|, 360 - |f^p(p) - B^p(p)H|) \), (ii) \( f^p(p)V \) is the intensity value for the component V of the HSV pixel in the current frame at time \( t \) and (iii) \( B^p(p)V \) is the V component of \( p \) in the reference frame at time \( t \) (reference frame contains the scene with no shadows, i.e., no object is present). The three conditions pertain to the V, S and H components respectively. The lower bound, \( x \), is used to define a maximum value for the darkening effect of shadows on the background and is approximately proportional to the light source intensity. Thus, for strong and high sun, a lower value of \( x \) must be chosen. The upper bound, \( \beta \), prevents the system from identifying the points where the background was darkened too little as shadow points. Approximate values for these parameters, and \( \tau_S \) and \( \tau_H \), are determined empirically and depend, for example, on scene illumination parameters that can be measured directly. The fundamental assumption, beyond decreased luminance, is that a shadow modifies the \( H \) and \( S \) components within determinable, environmentally-dependent limits defined by \( \tau_S \) and \( \tau_H \).

Shastri and Ramakrishnan (2004) applied the above model, while Baiheng and Yunqi (2004) used the slightly modified version defined by:

\[
S^p(p) = \begin{cases} 
1 & \text{if } |f^p(p) - B^p(p)| < \tau_S \land \Delta H < \tau_H & \text{and } \Delta V < \tau_H, \\
0 & \text{otherwise} 
\end{cases}
\]

here \( 1 \leq R \leq 3 \) is found experimentally.

Duque et al. (2005) used a modified version of the shadow model to detect shadows and highlights; their shadow and highlight masks \( L^p \) are defined, respectively, at each point \( p \) by:

\[
L^p(p) = \begin{cases} 
1 & \text{if } x \leq \frac{p^V}{y^V}, y^V \leq \beta \land |f^p(p) - B^p(p)| < \tau_S \land |f^p(p) - H - B^p(p)| < \tau_H \\
0 & \text{otherwise} 
\end{cases}
\]

These methods (Cucchiara et al., 2001, 2003, 2004; Baiheng and Yunqi, 2004; Duque et al., 2005; Shastri and Ramakrishnan, 2004) are capable of detecting weak penumbra shadows on flat surfaces. However, they require all illumination sources to be white, and assume that both shadow and non-shadow points have similar chrominance. The results obtained show that dark shadow tends to be misclassified.

Joshi and Papanikolopoulos (2008) proposed a semisupervised learning technique to detect moving shadows in dynamic environments. Their method exploits characteristic differences in illumination, colour and edges to extract features at each pixel from the video frame. These features are used for classifying each pixel as either belonging to shadow region or foreground region in the frame. Their algorithm uses a learning technique that employs a Support Vector Machine (SVM) capable of finding more complex separating boundaries and is co-trained by a small set of human trained data. The advantage of their semisupervised learning technique is that a system can be trained using a small set of labelled data and it improves with the help of large number of unlabelled examples – this also reduces the computational complexity of the SVM. But the statistical results show that performance is highly dependent on training; when appropriately trained shadow detection accuracy is high, compared to parametric methods, but reduces by 7 – 10% otherwise. For the same reason, performance decreases with changing illumination conditions.

\(^7\) The algorithm proposed by Mikic et al. (2000), Trivedi et al. (2000) is same. However, the applications are different.
3.3.2. Transform domain methods

The shadow detection method proposed by Leone and Distante (2007) is based on an automatic segmentation procedure, and the characterisation of textured patches of the shadow region is by Gabor filters. Texture analysis is performed by projecting the neighbourhood of pixels onto a set of Gabor functions, extracted by applying a generalised scheme of the Matching Pursuit strategy. The Euclidean distance between the extracted features is evaluated, and a thresholding strategy allows the identification of shadow pixels to good approximation. The algorithm is designed for moving objects and is environment independent. However, the performance of algorithm is dependent on patch or block size and no of features used to detect texture in theses patches. In some cases boundary of shadows are not well defined – these situations are due to the textural information that is substantially different in the patches extracted in the current image and the corresponding background.

3.3.3. Pixel and transform domain methods

A novel method for detecting moving cast shadows using a physically-derived shadow test condition has been introduced by Al-Najdawi et al. in (Al-Najdawi et al., 2006). The method is easily implemented in the pixel, Fourier or wavelet domains and is known for its simplicity, accuracy, and high efficiency. If \( p \) be a point on the surface of an object in an illuminated 3-dimensional scene and \( n_p \) be a neighbourhood of \( p \) in the surface; then, using a simple geometric representation of light rays and a simple reflection model, it is possible to show that the light energy received at points of \( n_p \) in the absence of an object casting a shadow over \( n_p \) is, to a high degree of approximation, affinely related to the energy received when a shadow is cast over \( n_p \) by an object. It is of course clear that when a shadow is cast over a neighbourhood, less light is received there – compared to the illuminated state – and this condition is included in the shadow model. Given that the reflected energies are similarly related, the luminance function \( L : n_p \rightarrow \mathbb{R} \) when no shadow is cast over \( n_p \) is affinely related to the luminance function \( L' : n_q \rightarrow \mathbb{R} \) when a shadow is cast; i.e., for \( n_q \) to be in shadow we have:

1. \( L'(r) < L(r) \), for all \( r \in n_q \), and
2. \( L'(r) = \lambda L(r) + \mu \), for all \( r \in n_q \),

for some constants \( \lambda \) and \( \mu \). The potential affine parameters for a neighbourhood, should an affine relation exist, are not required to be determined, or estimated, a priori – they can be computed directly from the (neighbourhood) pixel values to which they relate. Unlike other models, the threshold ‘c’ for the affine condition to be accepted is the only parameter that needs to be estimated.

An extensive review of the performance of techniques discussed above is provided in Section 4.

4. Performance evaluation

This section summarises the properties and performance of the algorithms discussed in this paper for which data is available in the literature. Information covering the following three aspects of performance is provided.

1. Table 1 provides a classification of the algorithms based on the four layer taxonomy introduced in Section 1 of this paper. The table has been compiled from a careful study of the approaches taken and assumptions made in the design, implementation and testing of the algorithms discussed in Section 3. The information presented may be used to assess the suitability of algorithms to particular applications – for example, algorithms suited to an application for which only monochrome images are to be used and for which object dependency is required are readily identified.

2. Sections 4.1 and 4.2 consider the key issue of accuracy; i.e., the ability of the algorithms to correctly detect shadow. The test data is that for which the most extensive comparative results are available in the literature; i.e., the widely used suite of benchmark videos comprising the ‘Campus’, ‘Laboratory’ and ‘Intelligent Room’ sequences. Section 4.1 discusses the environmental constraints of these videos, and comparative performance data, based on two standard shadow detection metrics, is given in Table 3.

3. Tables 1 and 3 provide information on the scope and accuracy of the algorithms, however other metrics are required for a more complete assessment of both scope and performance. The list given below includes the most significant of these found in the literature.

(a) Robustness to noise and/or compression artefacts, these can be assessed by analysis, e.g., region based classification is likely to be less sensitive to noise than pointwise approaches, or by comparative evaluation on benchmark images.

(b) Execution time estimates to assess, for example, the suitability of an algorithm for real-time application. There is relatively little discussion of this issue in the literature. Complexity analysis and comparative execution times on benchmark images would provide useful information for this purpose.

(c) The degree to which an algorithm is illumination independent; an algorithm may not deliver uniform performance in all circumstances – e.g., under particular illumination conditions that give rise to shadows of different strengths; e.g., light and heavy – both of which may occur in the same image.

(d) The performance, or flexibility, of the algorithm with regard to the detection of shadows of different types, or textures, within an image – e.g., shadows with umbra and penumbra regions, or shadows with sharp or less well-defined edges etc.

The information available in (or can be deduced from) the literature on the additional metrics discussed above is presented in Table 4 of Section 4.3.

4.1. Experimental environment and benchmark videos

The environments in which the benchmark videos were taken, can be characterised by:

- nature of the background surface: carpets, wooden floors, concrete textured walls, concrete neutral walls, asphalt roads,
- distance to objects: 2–250 feet,
- lighting conditions in the indoor environments: single point source and neutral walls, single point source and textured walls, multiple point sources spectrally equal but of different intensities, multiple point sources spectrally equal and of equal intensities, multiple point sources with arbitrary combination of lights,
- lighting conditions in outdoor environment: sunlight, overcast, dusk and dawn with artificial lights, dusk and dawn without artificial lights, night with single light source, and night with multiple light sources,
- surface orientation: vertical, horizontal, and sloping.
Fig. 5 illustrates frames from the benchmark videos, which are all in the public domain, and Table 2 summarises specific characteristics of each of the three videos.

4.2. Quantitative evaluation on the benchmarks

In order to systematically evaluate a shadow detection algorithm, it is useful to identify the following two important quality metrics: good detection (low probability of misclassifying a shadow point) and good discrimination (the probability of classifying non-shadow points as shadow should be low, i.e., low false alarm rate). Good detection corresponds to minimising the number of false negatives (FN), i.e., the shadow points classified as background/foreground. Good discrimination corresponds to minimising the number of false positives (FP), i.e., the foreground/background points detected as shadows (Prati et al., 2003).

Onoguchi (1998) proposed two metrics for moving object detection evaluation: the Detection Rate (DR) and the False Alarm Rate (FAR). Assuming TP as the number of true positives (i.e., the shadow points correctly identified), these two metrics are defined as follows:

\[
DR = \frac{TP}{TP + FN} \quad \text{and} \quad FAR = \frac{FP}{TP + FP}
\]

Prati et al. (2003) in their work showed that Onguchi metrics are not selective enough for the evaluation of shadow detection methods, since the metrics do not take into account whether a point detected as shadow belongs to a foreground object or to the

<table>
<thead>
<tr>
<th>Four layer taxonomy</th>
<th>Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object Dependent</td>
<td>Environment Dependent</td>
</tr>
<tr>
<td></td>
<td>Lalone et al. (2010)</td>
</tr>
<tr>
<td></td>
<td>Xu et al. (2005)</td>
</tr>
</tbody>
</table>

Fig. 5. The benchmark videos used to evaluate the shadow detection algorithms, (a) laboratory (b) campus (c) intelligent room.
background. Therefore, if shadow detection is used to improve moving object detection, only the first case is problematic, since false positives belonging to the background affect neither the object detection nor the object shape. To account for this, they have modified the above metrics, defining the shadow detection rate $\eta$ and the shadow discrimination rate $\nu$ as follows:

$$\eta = \frac{TP_S}{TP_S + FN_S}$$  and  $$\nu = \frac{TP_F}{TP_F + FN_F}$$

where $S$ denotes shadow and $F$ denotes foreground. $TP_F$ is the number of ground-truth points of the foreground objects minus the number of points detected as shadows, but belonging to foreground objects.

To compute the metrics described above, the ‘ground truth’ for each frame is required – the ground truth being obtained by segmenting the images with an accurate manual classification of points in the foreground, background, and shadow regions.

Based on the metrics of Prati et al., Table 3 provides a quantitative comparison of the algorithms, based on the benchmark video sequences.

The environmental constraints of the benchmark videos, as discussed in Section 4.1, are clearly limited and exclude, for example, shadows cast onto surfaces that reflect specularly (i.e., shiny surfaces) and shadows cast onto surfaces that have regions of self-shadow under most illumination conditions. Self-shadowing surfaces of this type include very rough surfaces, such as some types of brickwork and ‘natural’ surfaces such as tree bark or lawned-garden. The benchmark tests do not evaluate the performance of algorithms in these situations – which are challenging for all shadow detectors.

4.3. Qualitative evaluation

A qualitative evaluation, based on the additional metrics: robustness to noise, computational complexity, flexibility to the type of shadow, sharpness of the detected shadow and illumination independence, of the algorithms considered in this paper is presented in Table 4.

5. Summary

The paper provides a comprehensive overview of selected key algorithms, and hypotheses, for cast shadow detection. The algorithms are classified under a four-layer taxonomy based on object and environment dependency and domain of implementation. A further classification into monochrome/colour, for pixel domain algorithms is given. Though many shadow detection and removal algorithms have been discussed in the literature, the most general are those that are both object and environment independent.

Colour shadow models distinguish shadow pixels from non-shadow pixels by difference in brightness, with respect to the corresponding background, while chromaticity remains same. Colour information may improve shadow detection, but for a dynamic background the determination of the several required threshold values increases the complexity of the system – as they need be re-computed for each change of context. Moreover their performance depends on illumination conditions, since hue computation can be inaccurate under poorly illuminated conditions.

Texture models assume that the texture of the foreground object is distinct to that of the background, and that the texture is distributed uniformly inside the shadow region. Texture based models perform better under unstable conditions of illumination.

Geometric models attempt to remove shadowed regions, or the shadowing effect, using the geometric properties of objects. Geometric modes are more adaptive to specific scenes, due to dependency on the geometric relations between objects and scenes. To date, geometric models have been prevailingly applied to simulated environments containing specific objects. Computational complexity implies that their suitability to real-time applications is limited.

Transform domain algorithms can be more robust to noise and are less complex – as the number of features extracted are fewer and more precise. However, their performance is limited by the flexibility and the sharpness of shadow detected. Raw video content captured in cameras is presently used in video analysis algorithms incorporated within the camera hardware (e.g., FPGA implementations), due to the fact that they are free from

---

Table 2: Benchmark video specifics – conclusions drawn from sources.

<table>
<thead>
<tr>
<th>Benchmark video</th>
<th>Sequence type</th>
<th>Image size</th>
<th>Shadow strength</th>
<th>Shadow size</th>
<th>Object size</th>
<th>Noise levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Campus</td>
<td>outdoor</td>
<td>352x288</td>
<td>low</td>
<td>large</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Laboratory</td>
<td>indoor</td>
<td>320x240</td>
<td>low</td>
<td>medium</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Intelligent room</td>
<td>indoor</td>
<td>320x240</td>
<td>very low</td>
<td>small</td>
<td>small</td>
<td>medium</td>
</tr>
</tbody>
</table>

Table 3: Quantitative evaluation of the algorithms, based on the benchmark video sequences.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Campus</th>
<th>Laboratory</th>
<th>Intelligent room</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horprasert et al. (1999)$^a$</td>
<td>80.58</td>
<td>69.37</td>
<td>72.82</td>
</tr>
<tr>
<td>Stauder et al. (1999)$^b$</td>
<td>69.10</td>
<td>62.96</td>
<td>62.00</td>
</tr>
<tr>
<td>Mikic et al. (2000)$^c$</td>
<td>72.43</td>
<td>74.08</td>
<td>76.27</td>
</tr>
<tr>
<td>Cucchiara et al. (2001)$^d$</td>
<td>82.87</td>
<td>86.65</td>
<td>78.61</td>
</tr>
<tr>
<td>Siala et al. (2004)$^e$</td>
<td>77.21</td>
<td>94.85</td>
<td>91.42</td>
</tr>
<tr>
<td>Al-Najdawi (2006) Pixel domain</td>
<td>89.13</td>
<td>85.52</td>
<td>88.98</td>
</tr>
<tr>
<td>Al-Najdawi (2006) DCT domain</td>
<td>90.67</td>
<td>93.34</td>
<td>92.24</td>
</tr>
<tr>
<td>Al-Najdawi (2006) DWT domain</td>
<td>84.31</td>
<td>91.50</td>
<td>82.63</td>
</tr>
<tr>
<td>Joshi and Papanikopoulos (2008)$^f$ with co-training</td>
<td>N/A</td>
<td>N/A</td>
<td>91.02</td>
</tr>
<tr>
<td>Joshi and Papanikopoulos (2008)$^g$ with training for other videos</td>
<td>N/A</td>
<td>N/A</td>
<td>97.66</td>
</tr>
<tr>
<td>Jung (2009)</td>
<td>87.69</td>
<td>92.18</td>
<td>97.67</td>
</tr>
</tbody>
</table>

---

$^a$ The results of the first four algorithms are obtained from Prati et al. work Prati et al., 2003.
$^b$ Siala et al. (2004) provided their own results as shown in the fifth algorithm.
$^c$ Joshi and Papanikopoulos (2008) provide their results in last two algorithms.
A qualitative evaluation of the algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Robustness to noise</th>
<th>Computational complexity</th>
<th>Flexibility to shadow</th>
<th>Shadow detection</th>
<th>Illumination independence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horprasert et al. (1999)*</td>
<td>medium</td>
<td>low</td>
<td>medium</td>
<td>high</td>
<td>medium</td>
</tr>
<tr>
<td>Stauder et al. (1999)*</td>
<td>low</td>
<td>high</td>
<td>high</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Mikic et al. (2000)*</td>
<td>medium</td>
<td>high</td>
<td>medium</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Cucchiara et al. (2003)*</td>
<td>high</td>
<td>low</td>
<td>low</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>Tsai (2006)</td>
<td>medium</td>
<td>medium</td>
<td>medium</td>
<td>medium</td>
<td>edium</td>
</tr>
<tr>
<td>Al-Najdawi (2006)</td>
<td>low</td>
<td>low</td>
<td>high</td>
<td>medium</td>
<td>medium</td>
</tr>
<tr>
<td>Shen et al. (2007)</td>
<td>high</td>
<td>medium</td>
<td>low</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>Leone and Distant (2007)</td>
<td>high</td>
<td>low</td>
<td>medium</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>Yue Wang (2008)</td>
<td>low</td>
<td>low</td>
<td>medium</td>
<td>high</td>
<td>low</td>
</tr>
<tr>
<td>Chung et al. (2009)</td>
<td>medium</td>
<td>low</td>
<td>medium</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>Shoaib et al. (2009)</td>
<td>low</td>
<td>low</td>
<td>low</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Lin et al. (2010)</td>
<td>medium</td>
<td>medium</td>
<td>high</td>
<td>medium</td>
<td>low</td>
</tr>
</tbody>
</table>

* The information has been obtained from Prati et al. (2003).

compressed artifacts. Thus pixel domain approaches are particularly useful in processing raw, real-time video footage or images. In contrast many practical applications attempt video forensics where the vision algorithms are applied on previously stored video footage. Such videos/images are normally encoded in a particular format (e.g. MJPEG, MPEG-2, H.264). Transform domain approaches are particularly useful in such applications as the stored compressed video files or compressed images need not be completely decoded to perform shadow removal, thus applying the algorithm in a more compacted signal, reducing the computational complexity.

Performance of algorithms in different domain of implementations are compared quantitatively and qualitatively. Benchmark test video sequences are used to compare quantitative performance of the algorithms. A qualitative evaluation of the algorithms is given to compare their properties, such as robustness to noise, computational complexity, flexibility to shadow type and illumination conditions, in different environments.

6. Conclusions and future directions

The review has highlighted a number of aspects of shadow detection research; the most fundamental of which is the diversity, and dynamic nature, of environments for which shadow detection is required in vision applications. The algorithms developed therefore often have specific strengths and limitations (e.g. indoor/outdoor only), and are designed for particular data domains (e.g. colour/monochrome, pixel/transformation). A particular algorithm may be optimal for a specific application and may perform effectively without modification. However, due to the complex nature of many environments, adaptive and/or hybrid forms of existing approaches may best be able to meet the needs of dynamically changing conditions. This review has identified the need for further research in this direction which will require a comprehensive analysis of the specific environment, and its dynamic nature, prior to the determination of optimal combinations. The absence of a comprehensive de facto standard database of video content to analyse shadow detection and removal algorithms makes the objective comparison of algorithms difficult. As successfully done within face recognition, the preparation and publication of such databases of video content would be a significant contribution to future research in this field.

References


