



Stochastic shadow detection using a hypergraph partitioning approach[☆]



Vitor Gomes, Pablo Barcellos, Jacob Scharcanski^{*}

Institute of Informatics, Federal University of Rio Grande do Sul, 91501-970, Porto Alegre, RS, Brazil

ARTICLE INFO

Article history:

Received 9 July 2015

Received in revised form

14 September 2016

Accepted 15 September 2016

Available online 20 September 2016

Keywords:

Shadow detection

Chromaticity

Gradient correlation

Hypergraph segmentation

ABSTRACT

Discriminating shadows from the objects casting them often is challenging in practice, since the moving targets and their shadows tend to present similar motion patterns, and foreground detection methods often confuse cast shadows with foreground objects. To overcome these shadow detection difficulties, we propose a new stochastic shadow detection approach. In the proposed method, chromatic and gradient information are integrated with image hypergraph segmentation using a cascade of shadow/non-shadow classifiers, and a stochastic majority voting scheme is used to detect the shadow regions. The proposed method receives as input the segmented foreground objects and their cast shadows (mask), and outputs the shadows detected in the foreground mask. The experimental results were obtained with seven well known datasets, and suggest that the proposed shadow detection scheme can be more robust to different video acquisition conditions than other shadow detection methods, that are representative of the state-of-the-art.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Detection of moving objects (i.e. foreground detection) is an important step in many image- and video-based monitoring applications [1,2], like traffic monitoring [3–5] and people detection and/or tracking. Unfortunately, cast shadows have the same motion patterns as the objects casting them, and most foreground detection methods tend to confuse cast shadows with foreground objects [6], downgrading the performance of these methods. Besides, cast shadows usually are adjacent to moving objects, and the segmentation process tends to merge foreground objects separated by cast shadows, leading to erroneous object detection and recognition [7]. Therefore, the reliable detection of shadows is an important step in foreground and background detection.

Shadowing occurs when a light source is occluded by an object in the scene. The part of the object that is not illuminated is called self-shadow, and the area projected on the scene corresponding to the illumination occlusion is called cast shadow, or moving cast shadow if the object is in motion [8]. A substantial amount of work has been published on this topic and some representative recent contributions are discussed next.

Prati et al. [9] proposed two metrics to evaluate the performance of shadow detection methods, namely shadow detection rate (η) and shadow discrimination rate (ξ), that are calculated as follows:

$$\eta = \frac{TP_S}{TP_S + FN_S} \quad \text{and} \quad \xi = \frac{TP_F}{TP_F + FN_F}, \quad (1)$$

where TP and FN are the true positive and false negative pixels detected in shadow (S) and (non-shadow) foreground (F) regions. If the shadow detection rate η increases, the number of correctly detected cast shadow pixels increases. The shadow discrimination rate ξ tries to estimate the proportion of foreground pixels that are mistakenly labeled as background pixels, and increases when the number of pixels mistakenly assigned to the background decreases. There is a compromise between the two measures η and ξ .

A recent survey on shadow detection organizes the available methods in four categories [10]: (a) chromaticity-based methods; (b) physically inspired methods; (c) geometry-based methods; and (d) texture-based methods.

Cucchiara et al. proposed a chromaticity-based method [1] that tries to detect shadows by measuring the rate of change between the HSV components of a video frame and a background reference. This method assumes that shadow pixels in the video frame and in the background reference do not differ substantially in their hue component, and have low saturation and intensity values. However, just color information may not be able to discriminate

[☆]The authors would like to thank Digicon LTDA. and CAPES-Coordenação de Aperfeiçoamento de pessoal de Nível Superior-for financial support.

^{*} Corresponding author.

E-mail addresses: vogomes@inf.ufrgs.br (V. Gomes), prmbarcellos@inf.ufrgs.br (P. Barcellos), jacobs@inf.ufrgs.br (J. Scharcanski).

correctly shadows and foreground objects that have darker colors similar to shadows (e.g. dark cars and their cast shadows).

Physically based methods try to learn the appearance of shadow pixels, and may achieve higher accuracies than chromaticity-based methods [11]. Unfortunately, these methods tend to fail when dealing with objects having chromaticities similar to the background [10].

Geometry-based methods evaluate geometric features of shadows, such as the shadow orientation, size and shape [12]. The disadvantage of these methods is that previous knowledge of the scene is required, which often is unavailable [13]. These methods may fail when the shadow region and the foreground object have similar orientations [10].

Texture-based methods assume that shadows preserve most of the scene textures. For example, the method proposed in [7] uses chromaticity information to identify shadow regions, and then uses gradient information to refine the initial estimates of shadow or non-shadow pixels. This method provides good shadow detection results, and improves on the results of the chromaticity method in [1] by combining chromaticity and gradient information. However, significant color differences between foreground objects and cast shadows are required to obtain high quality shadow detection with this method [7].

In a recent survey presented by Al-Najdawi et al. [14], a different taxonomy was proposed to organize shadow detection methods. The shadow methods are classified based on object/environment dependency, or based on the implementation domain, which can be pixel domain or transform domain. According to [14], object/environment dependent methods are designed to detect a particular type of shadow (e.g. vehicle or human cast shadows) in a specific environment (e.g. indoor or outdoor scenes). Pixel domain methods are further divided into monochrome domain and color domain methods, and use pixel information (e.g. color) for shadow detection. Transform domain methods use different types of information for shadow detection, such as frequency domain, texture, or yet geometric information. Some transform domain methods are illustrated next.

The method proposed in Al-Najdawi et al. [15] removes insignificant cast shadows in video sequences based on edge and region information in multiple frames. A shadow is called insignificant when edges of the shadow region are not as sharp as the edges of the corresponding object. First, a mask containing moving objects and cast shadows is obtained. Then, the Canny edge detector is used to generate an edge map. The shadow regions are then removed using edge matching and region growing in multiple frames. Finally, the boundaries of the objects are improved and noise is removed by using a post-processing procedure. The motivation for this approach is that an insignificant shadow region often appears in an area where the gray levels change gradually from the background to the shadowed area. A disadvantage of this method is that if the object moves slowly, there is little change at the boundary between the object and the background, affecting negatively the results [15].

Guo et al. [16] proposed a single-image shadow detection and removal method based on a paired-regions approach. First, the image is segmented using the mean-shift algorithm [17], then a trained classifier based on color and texture information is used to estimate the confidence that each region is a shadow region. This classifier is trained using manually labeled regions, and a Support Vector Machine (SVM) is used to find similar regions under different illumination conditions. Next, regions with the same and different illuminations are represented by a relational graph, which is partitioned using graph-cuts in shadow and non-shadow regions. Finally, the results are improved by using image matting to smooth the transitions between shadow and non-shadow regions, and shadow-free images are obtained by relighting. A

disadvantage of this method is that it requires training a shadow classifier, which may involve manual labeling of shadow regions for generic scenes.

Another typical transform domain method [13] handles shadow detection using the robust wavelet watershed segmentation algorithm [18–20]. The segmented image regions are classified as shadow and non-shadow regions using a HSV-based approach, similar to the method proposed by Cucchiara et al. [1]. A problem with this method is that using only HSV color information to classify shadow and non-shadow regions may lead to erroneous results when the foreground objects colors are similar to shadows.

The method proposed in [21] uses the MTM (Matching by Tone Mapping) transformation as the distance between image patches of a video frame and a background reference. Shadowing effects are assumed to be non-linear tone mappings of the background gray levels. Since the MTM distance is invariant to non-linear mappings between corresponding image patches in shadow and non-shadow-regions, this MTM distance results in small values for shadowed areas and in large values for foreground patches that differ from the background [21]. To detect shadows, the MTM transformation is applied to the spatial neighborhood of each pixel in the foreground, generating a MTM distance map. The Otsu thresholding method [22] is then applied to the MTM distance map to discriminate between shadow and non-shadow regions. Unfortunately, thresholding the MTM distance map to guarantee an accurate discrimination between shadow and non-shadow regions is not trivial.

Khare et al. [23] proposed a shadow detection method based on the dual-tree complex wavelet transform to measure the difference between a video frame and the background reference in the HSV color space. The dual-tree complex wavelet transform of the difference images in the saturation and value channels is calculated. The standard deviation of the wavelet coefficients is computed, each coefficient is adaptively thresholded, and the image is reconstructed by discarding the wavelet coefficients that are associated to shadows. Finally, morphological operations are used as post-processing. This method provides interesting results, but since the wavelet coefficients are calculated based on color information only, the method performance tends to decrease if the foreground objects have chromaticities similar to shadows.

The method in Lalonde et al. [24] detects automatically shadows of objects on the ground, from a single image. They assume that the types of materials constituting the ground in outdoor scenes are limited (e.g., asphalt, brick, stone, mud, grass, concrete). Thus, the appearances of the shadows on the ground are not as widely varying and can be learned from a set of training images. The first stage of the detector consists in training a decision tree classifier on a set of sensitive shadow features based on each edge of the image. Then, a Conditional Random Field (CRF) based optimization is used to enforce a grouping of the shadow edges, creating longer contours. Finally, a global scene layout descriptor, specifically trained to detect grounds in images, is incorporated within the CRF. A disadvantage of this method is its focus only on outdoor scenes. The ground material in indoor scenes may vary greatly, resulting in classification errors. Also, using only a single image (i.e. no background model) sometimes do not provide enough information to accurately classify cast shadows.

Our proposed method was designed to overcome the common drawbacks found in most shadow detection methods available. In the proposed method, chromatic and gradient information are integrated with the image (or video frame) hypergraph segmentation, and finally a stochastic majority voting scheme is used to detect shadow regions. It is assumed that the foreground objects (mask) have already been detected by other methods, and that the foreground mask contains both the object and its shadow. Initially, a weighted image hypergraph is partitioned into K sub-

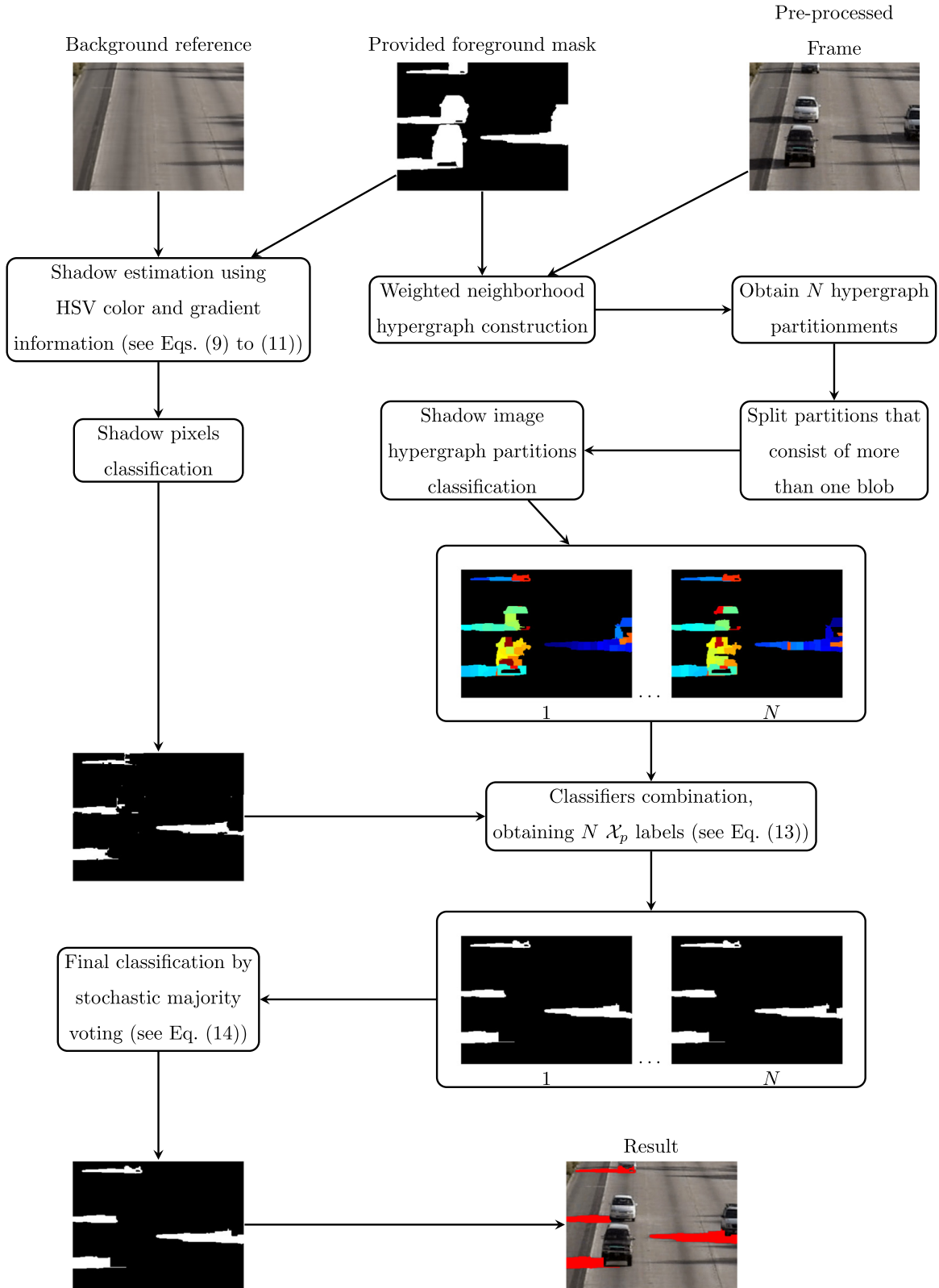


Fig. 1. Summary of the proposed shadow detection scheme.

hypergraphs using chromaticity and locality information. Then, pixels in these sub-hypergraphs are classified as shadow or non-shadow pixels based on chromaticity and gradient local information. Next, the regions associated with the sub-hypergraphs are classified as shadow or non-shadow regions using the mean and standard deviation of the V channel in HSV color space, and the

shadow pixels within the sub-hypergraphs regions are validated (or relabeled as non-shadow pixels). Finally, a stochastic majority voting scheme is applied to obtain the final shadow or non-shadow label of the image pixels. The datasets used in our experiments provide the foreground masks. Fig. 1 shows an overview of our proposed shadow detection scheme.

The proposed method combines hypergraph image segmentation with chromaticity and gradient information to detect shadows in video sequences, leading to improved shadow detection and discrimination rates in comparison with other methods that are representative of the state-of-the-art. In fact, the proposed approach provides a standalone framework for shadow detection that can be effective in a variety of scene configurations (e.g. indoor or outdoor, with light or dark shadows). Additionally, the proposed scheme can be customized to specific user needs by adjusting its parameters accordingly (e.g. for shadow detection in traffic monitoring). Guidelines for adjusting the parameters, and suggestions of optimized fixed parameters values are provided (see more details in Section 3.1). Nevertheless, the stochastic hypergraph partitioning component of our approach compensates for possible negative effects of small parameters misadjustments.

In the sequence, Section 2 explains in detail the proposed method. Section 3 presents our experimental results. We present our conclusions in Section 4.

2. Proposed method

Initially, we discuss how the images are pre-processed to improve the contrast between shadow and non-shadow regions. Then, we briefly introduce hypergraph image segmentation concepts. Next, we provide details of each step of the proposed shadow detection method.

2.1. Image pre-processing

The first step in our proposed method is to pre-process the image I by normalizing its V component in the HSV-color space. More formally, let $I = \{H^I, S^I, V^I\}$ be the HSV components of the image I , the normalized V component can be obtained by:

$$V_m^I = \frac{V^I - \min(V^I)}{\max(V^I) - \min(V^I)}, \quad (2)$$

where V_m^I is the normalized V component and V^I is the original value of the V component of the image I . Therefore, the pre-processed image I_m is obtained by substituting the original V component by its normalized version (i.e. $I_m = \{H^I, S^I, V_m^I\}$).

2.2. Hypergraphs for image segmentation

Hypergraphs are an extension of graphs in which edges are allowed to connect arbitrary, non-empty, sets of vertices [25]. Therefore, hypergraphs can model more general types of relations than graphs, and are used in image analysis applications involving data inter-dependencies [25–28]. Hypergraphs may be partitioned in a way to minimize a graph-cut cost function. The normalized cut method proposed by Shi and Malik [29] is an alternative criterion which attempts to correct the tendency of the graph cut algorithms to favor isolated graph vertices. Rital et al. [25] proposed a weighted adaptive approach for image segmentation using hypergraphs, which has the appeal of a strong theoretical basis. Most graph partitioning approaches try to find two subgraphs A and B of a graph G that minimize a cost function $cut\{A, B\} = \sum_{i \in A, j \in B} w(u, v)$, where $w(u, v)$ is the weight of the edge connecting the vertices u and v , with the following constraints: $A \cup B = G$, $A \cap B = \emptyset$ and $A \neq \emptyset, B \neq \emptyset$.

Our method uses a modified version of the weighted adaptive image neighborhood approach proposed in [25]. More formally, a hypergraph is denoted by a pair $H = (V, E)$, where $V = v_1, v_2, \dots, v_n$ is the set of vertices (or nodes) and $E = E_1, E_2, \dots, E_m$, where $E_i \subseteq V$ and $i = 1, \dots, m$, is the set of hyperedges.

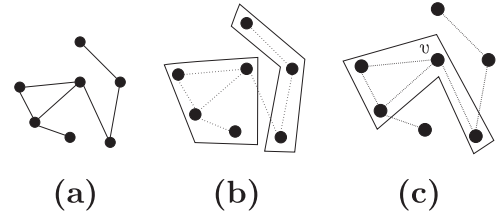


Fig. 2. Graph, hypergraph and neighborhood hypergraph example: (a) graph illustration; (b) hypergraph, and two sets of hypergraph vertices called hyperedges; (c) neighborhood hyperedge of the vertex v considering a distance of 1 (see Eqs. (3) and (4)).

Consider a graph $G(V; e)$, where e is the set of edges, and $H(V, E)$ is the hypergraph with the set of vertices of G and the hyperedges E representing the neighborhoods of these vertices, namely the *neighborhood hypergraph* of the graph G (see Fig. 2). In fact, we can associate a neighborhood hypergraph to each graph G as follows:

$$H_G = (V, (E_v = \{v\} \cup \Gamma(v))), \quad (3)$$

where

$$\Gamma(v) = \{u \in V, (v, u) \in e\}. \quad (4)$$

A color image can be represented by the following mapping:

$$I: V \subseteq \mathbb{Z}^2 \rightarrow C \subseteq \mathbb{Z}^n. \quad (5)$$

where the vertices of V are pixels and the elements of C are colors. A distance d on V defines a grid (a connected, regular graph, without loops or multi-edges). A distance d' can be defined on C in terms of the color similarity between two pixels. We then define the neighborhood relation in the image I by:

$$\Gamma_{\lambda, \epsilon}(v) = \{u \in V, |d'(I(v), I(u))| \leq \lambda \text{ and } d(v, u) \leq \epsilon\}, \quad (6)$$

where the neighborhood of v on the grid is $\Gamma_{\lambda, \epsilon}(v)$. Consequently, we can associate an *Image Neighborhood Hypergraph* (INH) to an image I [25] as follows:

$$H_{I, \lambda, \epsilon}(I) = (V, (V \cup \Gamma_{\lambda, \epsilon}(v))_{v \in V}). \quad (7)$$

The parameter ϵ controls the size of the hyperedges in the hypergraph, and larger values tend to increase the size of the segmented regions. In fact, ϵ can be understood as a spatial distance $\in [1, \infty)$, but $\epsilon \in [1, 5]$ is more realistic (see more details in Section 3.1).

The parameter λ indicates the maximum color distance for pixels being considered similar in the hypergraph partitioning process (i.e. it determines whether pixels belong to the same hyperedge of the hypergraph). Therefore, neighboring pixels need to have a color difference larger than λ to be considered different and be assigned to distinct hyperedges (regions). The parameter λ can be adjusted for each video sequence based on the average shadow intensity, being assigned two distinct values to λ based on a fixed optimized threshold (see Section 3.1).

The hypergraph segmentation approach in [25] was designed for grayscale images, so we modified the method in [25] to work with color images. We first convert the image to the Lab color space and then use the ΔE_{ab}^* distance [30], to estimate the similarity between two colors according to the human perception:

$$\Delta E_{ab}^* = \sqrt{(L_1^* - L_2^*)^2 + (a_1^* - a_2^*)^2 + (b_1^* - b_2^*)^2}, \quad (8)$$

where L^*, a^*, b^* are the color components in the Lab color space.

Given the neighborhood hypergraph of the input color image representing the foreground objects, we convert the image to grayscale and associate two weights to each vertex v_i : w_{v_i} and w_{h_i} .

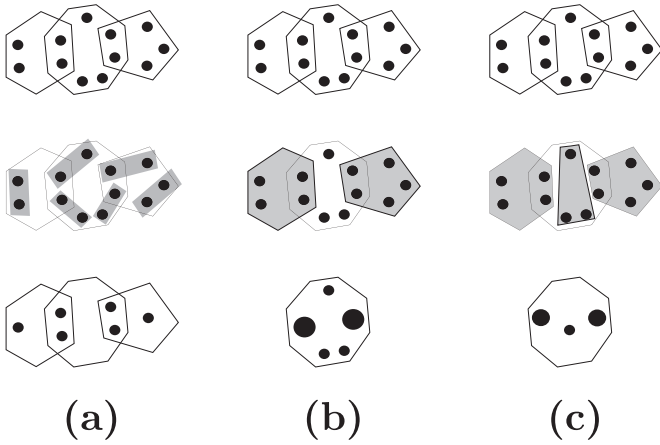


Fig. 3. Hypergraph coarsening illustration, top to bottom: (a) the hypergraph edge coarsening scheme in which pairs of connected vertices are merged together. The vertices are visited in random order and for each vertex v , all vertices that belong to hyperedges incident at v are considered, and the pairs of vertices connected via the largest weight edge are merged [35]; (b) hyperedge coarsening scheme in which a set of hyperedges is selected and the vertices that belong to these hyperedges are contracted together, giving preference to hyperedges with larger weights and smaller sizes [31]; and (c) modified hyperedge coarsening scheme, where after selecting the hyperedges that shall be contracted using the hyperedge coarsening scheme described before, all the uncontracted hyperedges vertices are considered in the contraction step. For each uncontracted hyperedge, the vertices that do not belong to any other hyperedge (that suffered contraction) are contracted together [31].

The weight w_{v_i} is the grayscale intensity of the pixel (i.e. vertex) $v_i \in V$, and the weight w_{h_i} is the mean grayscale intensity of the hyperedge h_i containing v_i . After constructing a neighborhood hypergraph for the foreground objects (see Fig. 1), we obtain K hypergraph partitions using the multilevel partitioning algorithm [31].

The hypergraph partitioning problem is known to be NP-hard, and the algorithms most often used are heuristic [32]. The multilevel paradigm for hypergraph partitioning [33] has become popular since it has advantages as compared to other paradigms, like recursive bisection (RB), which recursively splits the hypergraph into two partitions using heuristics until the desired number of partitions is obtained. However, these heuristics may perform poorly when dealing with large hyperedges or small vertex degrees (the degree of a vertex is the number of hyperedges containing that vertex). Also, these heuristics tend to not provide a global view of the problem, often resulting in sub-optimal solutions [32]. The multilevel paradigm addresses these issues and tends to produce high-quality hypergraph partitionings [34,32,33].

The multilevel approach for image hypergraph segmentation is divided into three main steps [33]: (I) coarsening; (II) initial hypergraph partitioning; and (III) uncoarsening and refinement. In the coarsening step, a sequence of smaller (coarser) hypergraphs is constructed. These smaller (coarser) hypergraphs preserve the structure of the original hypergraph (i.e. the finer hypergraph) as accurately as possible [25], and coarsening is achieved by merging vertices and contracting the hyperedges of the original hypergraph to form new hypergraph vertices in the coarser hypergraph (decreasing the size¹ of the hyperedges). Fig. 3 illustrates three different approaches for the coarsening process. We tested these different coarsening approaches, and the hyperedge coarsening approach (see Fig. 3(b)) provided the best experimental results (see Section 3).

In the next step of the multilevel approach for image hypergraph segmentation, the coarsest hypergraph is partitioned

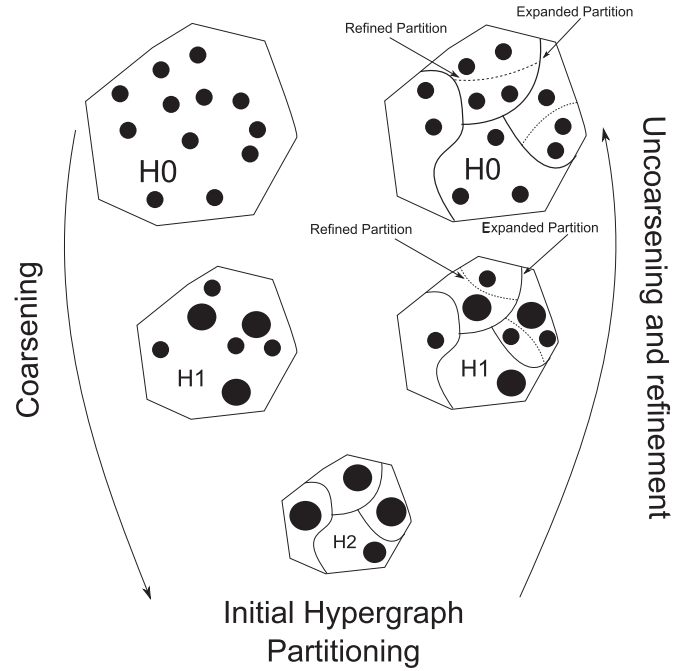


Fig. 4. Illustration of the three steps of the multilevel approach used in this work for image hypergraph segmentation [25].

minimizing the cost of the hypergraph cut, which is followed by the uncoarsening step. The last step of the hypergraph image segmentation process is uncoarsening, and the coarsest hypergraph partition is expanded again to re-introduce the vertices of the original hypergraph in the obtained coarser hypergraph partitioning. Then, a hypergraph partition refinement algorithm is used to further improve this hypergraph partitioning, as discussed next. The steps of the multilevel approach for image hypergraph segmentation are summarized in Fig. 4. The refinement algorithm reduces the cost of the hypergraph cut and improves the image hypergraph partitioning quality. Three refinement algorithms were evaluated for this task, namely: *FM* algorithm [36], which repeatedly moves vertices between partitions to reduce the cut cost [31]; *Early-Exit FM*, which is a simplified version of the FM algorithm that defines a maximum number of passes to be performed, and each pass stops if after moving k vertices the cut cost is reduced [31]; and *Hyperedge Refinement (HER)*, that iteratively moves groups of vertices between partitions until an entire hyperedge is removed from the hypergraph cut set [31]. In our experiments, the FM algorithm provided the best results and is used in this work as the image hypergraph partitioning refinement method.

After an image (i.e. video frame) represented by an image neighborhood hypergraph is segmented using the multilevel hypergraph segmentation approach, a final post-processing step is

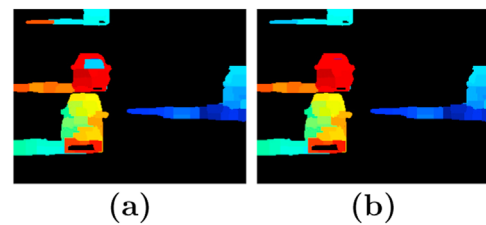


Fig. 5. Example of the splitting process during the image hypergraph partition: (a) partitions consisting of more than one blob (orange and cyan partitions); (b) after the splitting process, all partitions contain only one blob of foreground pixels. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

¹ A hyperedge connecting k nodes has size k .

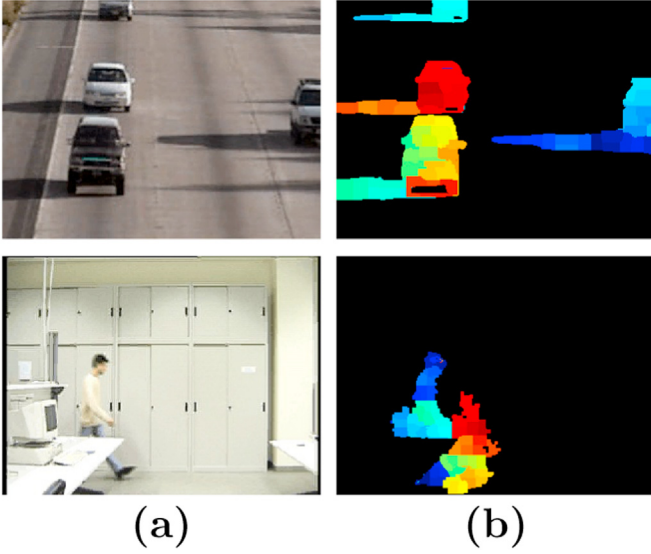


Fig. 6. Hypergraph segmentation results for a frame of the Highway dataset, and for a frame of the Lab dataset: (a) original frame; (b) image hypergraph partitioning result. Each color represents an image hypergraph partition corresponding to a foreground region, and black regions correspond to the background. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

performed. The parameter K indicates the number of hypergraph partitions. Low values of K tends to increase the heterogeneity of the hypergraph partitions, decreasing the shadow detection rate. The number K of hypergraph partitions is user-defined, but each partition may contain more than one group of foreground pixels (blob) (see more details about the parameter K in Section 3.1). Each partition is checked to detect this situation, and partitions that contain more than one blob are re-partitioned (split) until each partition contains only one blob. This splitting process may result in $K + L$ partitions, where L is the number of additional image hypergraph partitions created by the splitting process. An illustration of the splitting process for a video frame is shown in Fig. 5.

An illustration of the proposed image segmentation scheme based on image hypergraph partitioning can be seen in Fig. 6. Since we are only interested in the foreground objects shadows, the foreground mask provided as input to our method is used to build the image hypergraph, which will be used to detect shadows regions in the foreground (see Fig. 1). Details of the proposed shadow region detection process are presented next.

2.3. Shadow detection based on chromatic and spatial continuity information

The next step of our proposed shadow detection scheme is to classify each pixel based on chromatic and spatial continuity information. More formally, let $p(i, j)$ be an image pixel and $x_{p(i,j)}$ its label as follows: $x_{p(i,j)} = s$ if $p(i, j)$ is a shadow pixel, or $x_{p(i,j)} = \bar{s}$ otherwise. The results of this step will later be combined with another classifier (see Sections 2.4 and 2.5) to re-test and validate the detected shadow pixels (or relabel them as non-shadow pixels).

Initially, we detect the shadow pixels simply by comparing their HSV colors in the current frame with their HSV colors in a background reference (see (9)). The HSV color space was chosen because it provides an adequate separation between chromaticity and intensity. Then, we refine these shadow pixels estimates using gradient information [7], by classifying each pixel as a shadow/

non-shadow pixel. This process is described in more detail next.

A pixel $p(i, j)$ may be considered a shadow pixel if it satisfies [1]:

$$\begin{cases} \alpha \leq (F_p^V/B_p^V) \leq \beta, & \text{and} \\ (F_p^S - B_p^S) \leq \tau_s, & \text{and} \\ |F_p^H - B_p^H| \leq \tau_H, \end{cases} \quad (9)$$

where $\{F_p^H, F_p^S, F_p^V\}$ and $\{B_p^H, B_p^S, B_p^V\}$ are the values of the HSV color components of the pixel $p(i, j)$ in the video frame F and in the background reference B , respectively; the thresholds α, β, τ_s and τ_H are set to increase the number of shadow pixels detected in this stage; α is the lower threshold for the V channel frame/background ratio, and β is the higher threshold for the V channel frame/background ratio, with both parameters $\in [0, 1]$; τ_s is the saturation frame/background difference threshold, and τ_H is the hue frame/background difference threshold, and both parameters $\in [0, 255]$. The values of α, β, τ_s and τ_H were optimized to provide the best shadow detection rates (see Section 3.1), and used as constants for processing the video sequences described in Section 3. After that, regions \mathcal{R}_m consisting of shadow connected components constituted by shadow pixels that satisfy (9) are extracted. The gradient magnitude $|\nabla_p|$ and gradient direction θ_p are calculated at each shadow pixel $p(i, j) \in \mathcal{R}_m$, and only the pixels with significant gradient magnitudes $|\nabla_p| \geq \tau_m$, where τ_m is a gradient magnitude threshold $\in [0, 10]$, are further considered, so the effect of noise is reduced [7]. For each shadow pixel $p(i, j)$, the difference of its gradient direction in the frame F and in the background reference B is calculated as follows:

$$\Delta\theta_p = \arccos \left[\frac{\nabla_x^F \nabla_x^B + \nabla_y^F \nabla_y^B}{\left\{ (\nabla_x^F)^2 + (\nabla_y^F)^2 \right\}^{1/2} \left\{ (\nabla_x^B)^2 + (\nabla_y^B)^2 \right\}^{1/2}} \right], \quad (10)$$

and the gradient direction correlation between the frame F and the background reference B at $p(i, j)$ in \mathcal{R}_m is estimated by:

$$c^{\mathcal{R}_m} = \left\{ \sum_{p=1}^N H(\tau_a - \Delta\theta_p) \right\} / N, \quad (11)$$

where N is the number of pixels in \mathcal{R}_m , and $H(\cdot)$ is set to 1 if $\Delta\theta_p \leq \tau_a$, or set to 0 otherwise, where τ_a is the gradient direction threshold $\in [0, \pi]$. Therefore, $c^{\mathcal{R}_m}$ is the fraction of pixels in \mathcal{R}_m that have similar gradient directions in the frame F and in the background reference B [7]. If $c^{\mathcal{R}_m} \geq \tau_c$, the gradients have similar orientations in most pixels of \mathcal{R}_m in the frame F and in background reference B , where τ_c is the correlation threshold $\in [0, 1]$ used to determine if the gradients have similar orientations. These pixels in \mathcal{R}_m satisfying $c^{\mathcal{R}_m} \geq \tau_c$ are validated as shadow pixels, receiving label 's' (i.e. $\forall p(i, j) \in \mathcal{R}_m: x_{p(i,j)} = s$). The parameters values of τ_m, τ_a and τ_c were optimized to maximize the shadow detection rate, and used as constants for processing the video sequences (see Section 3.1).

2.4. Classification of image hypergraph partitions as shadow or non-shadow image regions

This step is simultaneous to the step described in Section 2.3 to create a validation test for the detected shadow pixels. All pixels in each image hypergraph partition \mathcal{P}_n are classified as a shadow (or non-shadow) pixels according to the following criterion:

$$x_{\mathcal{P}_n} = \begin{cases} s & \text{if } \mu_{\mathcal{P}_n}^F \leq \mu_{\mathcal{P}_n}^B \text{ and } \sigma_{\mathcal{P}_n} \leq \tau_\sigma, \\ \bar{s}, & \text{otherwise;} \end{cases} \quad (12)$$

where $x_{\mathcal{P}_n}$ is the label of the pixels in the hypergraph partition \mathcal{P}_n , $\mu_{\mathcal{P}_n}^F$ and $\mu_{\mathcal{P}_n}^B$ are the V channel averages of the HSV color pixels $p \in \mathcal{P}_n$ in the frame F and in the background reference B , respectively, and $\sigma_{\mathcal{P}_n}$ is the standard deviation of the V channel of the HSV color pixels $p \in \mathcal{P}_n$. The reasoning is that pixels belonging to shadow regions tend to be darker in the current frame than in the background reference (which contains no shadows), therefore if the foreground pixels are brighter in average (i.e. $\mu_{\mathcal{P}_n}^F > \mu_{\mathcal{P}_n}^B$), no further validation is required and all pixels in \mathcal{P}_n are classified as non-shadow pixels. Also, image shadow regions tend to be more uniform in the V channel than foreground regions, so the pixels in an image hypergraph partition \mathcal{P}_n that have large intensity variability in average (i.e. $\sigma_{\mathcal{P}_n} > \tau_\sigma$) are classified as non-shadow pixels. The threshold τ_σ is used in the classification of the image hypergraphs partitions, and is an upper limit $\in [0, 1]$ for the standard deviation of the pixels $p \in \mathcal{P}_n$ of the V channel. The pixels of an image hypergraph partition \mathcal{P}_n that have small intensity variability in average (i.e. $\sigma_{\mathcal{P}_n} \leq \tau_\sigma$) are classified as shadow pixels. Increasing the τ_σ value tends to increase the shadow detection rate, and to decrease the shadow discrimination rate (as a consequence). The parameter τ_σ was optimized to provide an adequate balance between the shadow detection and the shadow discrimination rates (see Section 3.1).

Let $x_{p(i,j)}$ be the final label for a pixel $p(i, j)$. If the set of pixels in an image hypergraph partition \mathcal{P}_n is classified as a non-shadow (i.e. $x_{\mathcal{P}_n} = \bar{s}$) then $\forall p(i, j) \in \mathcal{P}_n$ the pixel labels in \mathcal{P}_n are set to $x_{p(i,j)} = \bar{s}$ and the pixels in \mathcal{P}_n are non-shadow pixels and do not need further validation. On the other hand, the set of pixels in an image hypergraph partition \mathcal{P}_n classified as a shadow region (i.e. $x_{\mathcal{P}_n} = s$) may contain some pixels labeled as non-shadow pixels (i.e. $x_{p(i,j)} = \bar{s}$), so the next step in the proposed shadow detection method is to validate the individual shadow or non-shadow pixels labels using the image hypergraph partition labels ($x_{\mathcal{P}_n}$ and $x_{p(i,j)}$), obtaining the final shadow or non-shadow label $x_{p(i,j)}$ for the pixels $p(i, j) \in \mathcal{P}_n$ forming a shadow mask for the image, as detailed next.

2.5. Shadow detection by combining classifiers using stochastic majority voting

The shadow pixels classification based on chromatic and spatial continuity criteria (see Section 2.3) are validated by the hypergraph partitions detected as shadow regions (see Section 2.4) using a stochastic majority rule, and this process generates the final shadow mask, as described next. This validation process is motivated by the fact that at this stage some of the shadow pixels obtained with Eqs. (9)–(11) may be outliers or incorrectly labeled.

The image hypergraph segmentation approach in Section 2.2 generates image segments (hypergraph partitions) consisting of

pixels that are visually similar and spatially close, so these image segments are color homogeneous [37]. If most pixels in an image region \mathcal{P}_n are shadow pixels (or non-shadow pixels), then all pixels in the region \mathcal{P}_n are classified as shadow pixels (or non-shadow pixels). Therefore, pixels that have been estimated previously as shadow pixels are validated (or relabeled as non-shadow pixels) by considering the label of similar pixels in their neighborhoods, resulting in an improved shadow/non-shadow discrimination. This refinement step of the shadow detection process is described in more detail below.








The proportions of shadow/non-shadow pixels are estimated in each image hypergraph partition \mathcal{P}_n labeled as a shadow region (i.e. $x_{\mathcal{P}_n} = s$) (see (12)). If \mathcal{P}_n has more shadow pixels than non-shadow pixels, then \mathcal{P}_n is validated as a shadow region, and the final shadow mask labels $x_{p(i,j)}$ of the pixels $p(i, j) \in \mathcal{P}_n$ are obtained as follows:

$$\forall p(i, j) \in \mathcal{P}_n, \quad x_{p(i,j)} = \begin{cases} s, & \text{if } \frac{\sum_{p(i,j) \in \mathcal{P}_n} \delta(x_{p(i,j)}, s)}{N} \\ & \geq \frac{\sum_{p(i,j) \in \mathcal{P}_n} \delta(x_{p(i,j)}, \bar{s})}{N} - \tau_p, \\ & \text{and } x_{\mathcal{P}_n} = s, \\ \bar{s}, & \text{otherwise,} \end{cases} \quad (13)$$

where $p(i, j)$ is a pixel belonging to a partition \mathcal{P}_n , $x_{p(i,j)}$ is the final shadow mask label of $p(i, j) \in \mathcal{P}_n$, N is the total number of pixels in \mathcal{P}_n ; $x_{p(i,j)} = s$ if the pixel has been classified as a shadow pixel using (9) and $c^{\mathcal{R}_m} \geq \tau_c$ (see (11)), or $x_{p(i,j)} = \bar{s}$ otherwise (see Section 2.3); $\delta(a, b)$ is the Kronecker delta such that $\delta(a, b)$ evaluates to 1 if $a=b$ or to 0 otherwise; and τ_p is a threshold that controls the shadow detection sensitivity and $\tau_p \in [-1, 1]$. It is used to validate the shadow pixels classified based on the chromatic and spatial continuity criteria (see Section 2.3) within the hypergraph partitions detected as shadow regions (see Section 2.4). Positive τ_p values tend to increase the shadow detection rate at the cost of decreasing the shadow discrimination rate. The shadow pixels classification method based on chromatic and spatial continuity criteria (see Section 2.3) tends to produce high shadow discrimination rates at the expense of low shadow detection rates. In our experiments, τ_p was optimized to favor the shadow detection rate, and achieve a better balance between the shadow detection and shadow discrimination rates (see Section 3.1).

The multilevel hypergraph partitioning algorithm is non-deterministic due to the random bisection used in the initial partitioning phase [31] (see Section 2.2, Fig. 4). This occurs because at each run of the multilevel hypergraph partitioning algorithm, a stochastic (or non-deterministic [38,39]) hypergraph partition is obtained. Consequently, the multiple partitions obtained for the same hypergraph may differ, leading to different shadow mask labels $x_{p(i,j)}$ for the same pixels. Therefore, a stochastic majority voting scheme is used as a final step to obtain the shadow mask pixels labels $\hat{x}_{p(i,j)}$. In this stochastic voting scheme, N hypergraph

Table 1
Information about the datasets used in the experiments.

Dataset	Highway 1	Lab	Highway 3	Campus	Room	Hallway	CAVIAR
Frame Example							
Number of frames	8	14	7	53	22	13	1125
Resolution	320 × 245	320 × 240	320 × 240	320 × 240	320 × 245	320 × 240	384 × 288
Scene	Outdoor	Indoor	Outdoor	Outdoor	Indoor	Indoor	Indoor
Objects	Vehicles	People	Vehicles	People	People	People	People
Shadow	Strong	Light	Strong	Light	Light	Light	Light

partitions are obtained stochastically (see Section 2.2), and N classification labels $\mathcal{X}_{p(i,j)}$ are generated for each pixel $p(i, j)$. Let $\mathcal{X}_{p(i,j)}^l$ be the shadow mask label of a pixel $p(i, j)$ (see (13)) obtained in the l -th hypergraph partitioning. The final shadow mask $\hat{\mathcal{X}}_{p(i,j)}^l$ contains the pixels $p(i, j)$ that were classified as shadows in at least τ_r hypergraph partitions, as indicated below:

$$\begin{cases} \hat{\mathcal{X}}_{p(i,j)}^l = 1 & \text{if } \left\{ \sum_{l=1}^N \delta(\mathcal{X}_{p(i,j)}^l, s) \geq \tau_r \right\}, \\ \hat{\mathcal{X}}_{p(i,j)}^l = 0 & \text{otherwise} \end{cases} \quad (14)$$

where $\delta(a, b)$ is the Kronecker delta and τ_r is the voting threshold used in the stochastic majority voting scheme. As mentioned before, N hypergraph partitions are obtained stochastically in this voting scheme, and the final shadow mask is formed by the pixels classified as shadow pixels in at least τ_r hypergraph partitions. The parameter τ_r was optimized to provide an adequate balance between the shadow detection rate and the shadow discrimination rate (see more details in Section 3.1).

2.6. Summary of the proposed stochastic shadow detection method

Below, the proposed shadow detection method is summarized, step by step:

1. Build the weighted hypergraph of the video frame in the Lab color space, using the ΔE_{ab}^* color distance (see (8)) and the provided foreground mask;
2. Obtain K image hypergraph partitions N times, and for each different partitioning do:
 - (a) Split the partitions \mathcal{P}_n that contain more than one blob of pixels;
 - (b) Use the color (see (9)) and gradient correlation information (i.e. $c^{\mathcal{R}_m} \geq \tau_c$, where $c^{\mathcal{R}_m}$ is given by (11)) to classify pixels

$p(i, j)$ as shadow (or non-shadow) pixels (i.e. $\mathcal{X}_{p(i,j)} = s$ or $\mathcal{X}_{p(i,j)} = \bar{s}$);

- (c) Use the mean ($\mu_{\mathcal{P}_n}^F, \mu_{\mathcal{P}_n}^B$) and the standard deviation ($\sigma_{\mathcal{P}_n}$) of the V channel of the HSV color representation of each partition \mathcal{P}_n to classify all pixels in the entire partition as a shadow (or a non-shadow) region pixels (i.e. $\mathcal{X}_{\mathcal{P}_n} = s$ or $\mathcal{X}_{\mathcal{P}_n} = \bar{s}$), according to (12);
- (d) Calculate the proportion of shadow pixels in each shadow hypergraph partition \mathcal{P}_n . If a partition \mathcal{P}_n contains more shadow/non-shadow pixels, classify all pixels $p(i, j) \in \mathcal{P}_n$ as shadow/non-shadow pixels, obtaining the l -th pixel label $\mathcal{X}_{p(i,j)}^l$ according to (13);
3. Obtain the final shadow mask $\hat{\mathcal{X}}_{p(i,j)}^l$ containing all pixels $p(i, j)$ that were classified as shadow pixels (i.e. $\mathcal{X}_{p(i,j)}^l = s$) in at least τ_r of the hypergraph partitionings, according to (14).

3. Experiments and results

In order to validate our proposed method, we performed experiments on seven popular video sequences that are publicly available [10], and measured the shadow detection (η) and shadow discrimination (ξ) rates. We also used the average of these two measures $\left(\frac{\eta + \xi}{2}\right)$ as a third measure [10]. We chose those video sequences because they are commonly used to validate shadow detection methods [10], and have the ground truth data and foreground masks available. Summarizing information about these datasets can be seen in Table 1.

For the comparative tests, the following methods were chosen as representatives of the state-of-the-art in shadow detection [11,10,21,16,24], as detailed next. The method proposed by Huang and Chen [11] uses physically based principles, and tries to learn the appearance of shadow pixels. The method proposed in Huang and Chen [10] uses chromaticity and gradient information to achieve high shadow detection (η) and discrimination (ξ) rates. On

Table 2
Summary of the parameters used in the proposed method with their ranges and suggested values.

Parameter	Description	Range	Parameters values
λ	Maximum color difference allowed to assign two pixels to the same hypergraph partition.	[1, 20]	$\lambda_{min} = 4$ (if $\mu_{\Delta G_L} > \tau_{\Delta G_L}$) or $\lambda_{max} = 10$ (if $\mu_{\Delta G_L} \leq \tau_{\Delta G_L}$) (see discussion about λ below)
$\tau_{\Delta G_L}$	Lightness gradient magnitude threshold used to decide if two pixels should be assigned to the same hypergraph partition.	[0, 255]	24 (see discussion about λ below)
ϵ	Maximum pixel distance allowed to assign two pixels to the same hypergraph partition (i.e. the hyperedge size).	[1, 5]	2
K	Number of desired hypergraph partitions during the hypergraph segmentation	[2, 100]	51
α	Lower threshold for the difference between the foreground and the background model [1] in the V channel (in the HSV color space).	[0, 1]	0.21
β	Upper threshold for the difference between the foreground and the background model [1] in the V channel (in the HSV color space).	[0, 1]	0.99
τ_S	Threshold of the S channel difference between the foreground and the background model [1] (in HSV color space).	[0, 255]	76
τ_H	Threshold of the H channel difference between the foreground and the background model [1] (in HSV color space).	[0, 255]	93
τ_m	Threshold for the V channel gradient magnitude used to decide if pixels should be assigned to shadow regions or to the foreground [7].	[0, 10]	6
τ_a	Threshold for the V channel gradient direction coherence used to decide if pixels should be assigned to shadow regions or to the foreground [7].	[0, π]	$\frac{\pi}{10}$
τ_c	Threshold for the correlation between the V channel gradient directions in the foreground and in the background used to decide if pixels should be assigned to shadow regions or to the foreground [7].	[0, 1]	0.2
τ_σ	Threshold for maximum standard deviation to decide if a partition is a shadow partition. It is measured based on the V channel pixels (in HSV color space).	[0, 1]	0.15
τ_p	Shadow detection sensitivity used to validate shadow pixels within hypergraph partitions.	[-1, 1]	0.24
τ_r	Threshold used to classify a pixel as a shadow pixel after multiple stochastic partitionings.	[1, 10]	3

Table 3

Summary of the average lightness gradient magnitudes estimated for each video sequence, and the λ values used by the proposed shadow detection method.

Dataset	$\mu_{\Delta G_L}$	λ
Highway 1	19.62	10
Highway 3	18.66	10
Campus	20.79	10
Lab	26.87	4
Room	29.29	4
Hallway	35.86	4
CAVIAR	24.62	4

the other hand, the method proposed by Sanin et al. [21] uses the MTM (Matching by Tone Mapping) distance between image patches to detect shadows. The method proposed by Bullick et al. [24] detects shadows in outdoor image from a single image (i.e. no background model) using a Conditional Random Field (CRF) based optimization and a classifier previously trained to detect shadows in common outdoor ground materials, such as asphalt or grass. The method proposed by Lalonde et al. [16] detects shadows from a single-image and uses a paired-regions approach, combined with graph-cuts and a Support Vector Machine (SVM) classifier to partition shadow and non-shadow regions. For more details of the comparative methods see Section 1.

The proposed shadow detection method can adjust to the video lightness variability (in average), which helps improving the performance of the shadow detection process. In order to make the proposed method adaptive to the video average lightness variability, the λ parameter is allowed to adjust to the average gradient magnitude of the lightness (L) channel of each video sequence (in Lab color space), namely $\mu_{\Delta G_L}$, based on a fixed threshold $\tau_{\Delta G_L}$. Consequently, the λ parameter is allowed to assume a maximum λ_{max} or a minimum λ_{min} value as follows: (a) λ_{min} if $\mu_{\Delta G_L} > \tau_{\Delta G_L}$ or (b) λ_{max} if $\mu_{\Delta G_L} \leq \tau_{\Delta G_L}$. The maximum and the minimum values of λ (i.e. λ_{max} and λ_{min}), as well as the threshold $\tau_{\Delta G_L}$ have optimized fixed values which are used for all video sequences.

Next, we discuss the roles and optimized fixed values for the different parameters of the proposed method.

3.1. Summary of the proposed stochastic shadow detection method parameters

Below, we discuss the role of each parameter, and suggest fixed values for the parameters used in the proposed shadow detection method (see Table 2)²:

1. λ : The parameter λ is the only parameter that is adjusted based on the average lightness variability of each video sequence. The parameter λ controls the maximum color difference that two pixels may have to assign both to the same hypergraph partition during the hypergraph partitioning process. It should be observed that a pair of shadow and background pixels may be assigned mistakenly to the same hypergraph partition (i.e. both can be labeled as shadow or as background pixels) during the hypergraph partitioning process if the lightness contrast between shadow regions and their surrounding foreground regions is not controlled. By measuring the average gradient magnitude of the lightness channel $\mu_{\Delta G_L}$ for each video sequence (in Lab color space), and comparing $\mu_{\Delta G_L}$ with a fixed threshold $\tau_{\Delta G_L}$, λ can adjust to the video average lightness

variability. The parameter $\tau_{\Delta G_L} \in [0, 255]$ is the lightness gradient magnitude threshold used to decide if two pixels should be assigned to the same hypergraph partition, and the optimized threshold value is $\tau_{\Delta G_L} = 24$. The average lightness gradient magnitude value ($\mu_{\Delta G_L}$) is estimated based on the shadow regions detected on a few frames of a video sequence (using the approach described in Section 2.3). Therefore, we can classify a cast shadow regions as darker or lighter based on $\mu_{\Delta G_L}$ and $\tau_{\Delta G_L}$, and adjust the value of λ accordingly. The optimized maximum (λ_{max}) and minimum (λ_{min}) values for λ are: (a) $\lambda_{min} = 4$ if $\mu_{\Delta G_L} > \tau_{\Delta G_L}$, and (b) $\lambda_{max} = 10$ if $\mu_{\Delta G_L} \leq \tau_{\Delta G_L}$. By using this adjustable λ scheme, $\lambda = \lambda_{max} = 10$ for the video sequences 'Highway 1', 'Highway 3' and 'Campus' (that contain darker cast shadows), but $\lambda = \lambda_{min} = 4$ for the other sequences containing lighter cast shadows (i.e., 'Lab', 'Room', 'Hallway' and 'CAVIAR'). Table 3 shows the average lightness gradient magnitudes obtained for each video sequence.

2. ϵ : This parameter controls the maximum spatial distance that two pixels may have to assign both pixels to the same hypergraph partition (hyperedge) in the hypergraph partitioning process (i.e. ϵ controls the size of the hyperedge). A reasonable range for this parameter is $[1, 5]$. The optimized parameter value is $\epsilon = 2$, which is used for all video sequences. Higher ϵ values tends to lead to larger hyperedges, decreasing the accuracy of the shadow detection process. Larger hyperedges tend to contain more mixed pixels (i.e. shadow and non-shadow pixels), which may result in more pixels being classified incorrectly as shadow (or non-shadow) pixels. Also, higher values of ϵ may increase the computational cost, because larger neighborhoods must be evaluated for each image pixel. The fixed value $\epsilon = 2$ leads to a satisfactory performance based on our experiments with all video sequences (indoor or outdoor, with light or dark cast shadows).
3. K : This parameter controls the number of hypergraph partitions in the hypergraph segmentation process. If K assumes lower values, the resulting hypergraph segmentation partitions tend to be more heterogeneous (i.e. tend to increase the mix of shadow and non-shadow pixels in the hypergraph segmentation partitions). Also, the size of the cast shadows should be considered when adjusting K , since segmenting larger shadow regions would require a larger number of homogeneous partitions (i.e. more smaller hypergraph partitions), and vice-versa. The optimized value $K=51$ is used by our proposed method to process all video sequences, and this K value tends to provide satisfactory shadow detection results for different scene configurations (indoor or outdoor, with light or dark cast shadows).
4. α and β : These thresholds control the lower and upper intensity difference in the V channel between a shadow region and its corresponding region in the background model (in HSV space), and both α and $\beta \in [0, 1]$. Cucchiara et al. [1] suggest that hue values vary slightly within a shadow region with respect to its corresponding background region, but the saturation and luminance (intensity) color components tend to have smaller values in shadow regions. In our shadow detection method, a rough estimate of the shadow regions is obtained initially, and later these estimates are refined to obtain the final shadow detection mask. Therefore, the thresholds α and β are set in a way that potential shadows pixels are included in larger numbers in these rough (initial) shadow regions estimates. The optimized fixed values $\alpha = 0.21$ and $\beta = 0.99$ are used in our proposed shadow detection method. These fixed values for α and β tend to be efficient for detecting darker and lighter cast shadows. Lowering the value of β would decrease the detection of lighter cast shadow pixels, and increasing the value of α would have the effect of decreasing the detection of darker cast

² It shall be observed that the number of parameters of the proposed method is comparable to the number of parameters of other shadow detection methods that are representative of the state-of-the-art [7].

shadow pixels. These optimized fixed values for α and β were used to test our method in all video sequences, and potentially can provide an adequate shadow detection performance for different scene configurations.

5. τ_s : The threshold $\tau_s \in [0, 255]$ controls the difference in the S saturation channel, between pixels in a shadow region and in its corresponding background region. Cucchiara et al. [1] suggest that pixels have lower saturation values in a shadow region in comparison with their saturation values in the corresponding background region. In the initial stage of the proposed shadow detection process, the number of potential shadow pixels detected is maximized to create rough shadow regions estimates, and τ_s is set to a higher value to allow many pixels to be included in the initial shadow regions estimates. Decreasing τ_s may result in less false positives being detected (i.e. foreground pixels detected as shadow pixels), but also may decrease the number of shadow pixels correctly identified. Video sequences with darker cast shadows may require larger values of τ_s to be detected accurately. The optimized fixed value $\tau_s = 76$ is used for all video sequences, and potentially can provide an adequate shadow detection performance for different scene configurations.
6. τ_H : The threshold $\tau_H \in [0, 255]$ controls the acceptable difference in terms of the hue (H) channel, between pixels in a shadow region and in its corresponding background region. Cucchiara et al. [1] suggest that pixels have similar hue values in shadow regions and in their corresponding background regions. Therefore, if the hue difference between a pixel in the foreground model and in the background model is larger than τ_H , this pixel is not classified as a shadow pixel. To detect darker cast shadows accurately, τ_H should be set to higher values. However, increasing the value of τ_H may increase the false positives (i.e. foreground pixels detected as shadow pixels). The optimized fixed value $\tau_H = 93$ is used for all tested video sequences, and potentially can provide a satisfactory shadow detection performance for different scene configurations.
7. τ_m , τ_a and τ_c : The thresholds τ_m and τ_a control the similarity between the gradient magnitude and direction of pixels in foreground regions and in their corresponding background regions, respectively. Sanin et al. [7] suggest that textural information of the background regions tends to be preserved in their corresponding shadow regions. Therefore, foreground pixels are evaluated if they can be considered shadow pixels, or not, by comparing the local gradient magnitudes and directions of pixels in foreground regions and in their corresponding background regions. This comparison is used to evaluate if foreground pixels can be assigned to shadow regions, and included in the rough (initial) shadow regions estimates. The parameter τ_c controls the maximum difference in the gradients directions for a pixel to be considered a shadow pixel (see Section 2.3). Increasing the τ_c value, more pixels are considered shadow pixels, which increases the false positives rate (i.e. foreground pixels detected as shadow pixels). Reasonable ranges for these parameters are $\tau_m \in [0, 10]$, $\tau_a \in [0, \pi]$ and $\tau_c \in [0, 1]$. The optimized fixed values $\tau_m = 6$, $\tau_a = \frac{\pi}{10}$ and $\tau_c = 0.2$ were used for all video sequences, and potentially can provide a satisfactory shadow detection performance for different scene configurations.
8. τ_σ : The threshold $\tau_\sigma \in [0, 1]$ controls the maximum intensity standard deviation (i.e. intensity variability) that a given hypergraph partition may have to be classified as a shadow region partition. Shadow regions tend to be more homogeneous in terms of intensity values than non-shadow regions, therefore hypergraph partitions with lower intensity standard deviations have a greater chance of corresponding to shadow

Table 4

Parameters used to set the hMETIS tool to perform stochastic hypergraph partitioning within the context of the proposed shadow detection scheme.

Parameter	Value
Number of partitions	51
Number of runs	10
UB factor	1
Matching Scheme	Hyperedge
Refinement Policy	Fiduccia-Mattheyses (FM)
V-cycle Refinement	On good intermediate solution
Reconst	Reconstructs
Color distance (λ)	10 (strong shadows) and 4 (light shadows)
Grid distance (ϵ)	2








regions. Increasing τ_σ may result in increased shadow detection rates, but lower shadow discrimination rates can be obtained as a consequence. The fixed value $\tau_\sigma = 0.15$ is used for all video sequences, and potentially can provide a satisfactory shadow detection performance for different scene configurations.

9. τ_p : The threshold $\tau_p \in [-1, 1]$ controls the shadow detection sensitivity. Ideally, after the hypergraph partitioning, each hypergraph partition would be homogeneous and contain only shadow or non-shadow pixels. But, in practice, each partition may be heterogeneous and contain shadow and non-shadow pixels. In the proposed approach, the hypergraph partition homogeneity information is combined with the initial estimates of shadow and non-shadow pixels obtained using color and gradient information, and the final classification of pixels is obtained. The parameter τ_p controls the percentage of non-shadow pixels that are allowed in a shadow partition (and vice-versa), without impacting significantly on final classification of pixels as shadow or non-shadow pixels (i.e. on the shadow detection performance). It shall be observed that there is a trade-off between the shadow detection rate and shadow discrimination rate in shadow detection methods. Positive values of τ_p tend to increase the shadow detection rate, with the cost of decreasing the discrimination rate (and vice-versa). The optimized fixed value $\tau_p = 0.24$ is used for all video sequences, and potentially can provide a satisfactory shadow detection performance for different scene configurations.
10. τ_r : The threshold τ_r controls the voting process in our stochastic majority voting scheme. Due to the stochasticity of the hypergraph segmentation process, a pixel may receive different shadow/non-shadow classifications in different hypergraph partitionings. In the proposed approach, a pixel is only classified as a shadow pixel if it is classified as a shadow pixel in at least τ_r hypergraph partitionings. We suggest to perform at least 10 hypergraph partitionings, and using the optimized fixed value $\tau_r = 3$ (τ_r has been optimized to favor the shadow detection rate). This τ_r fixed value potentially can provide a satisfactory shadow detection performance for different scene configurations.

It should be emphasized that each parameter used in our shadow detection method has a range. The optimized fixed parameters values suggested for using the proposed method in videos of different scene configurations can be found in Table 2 (e.g. for processing indoor and outdoor scenes, with light and strong cast shadows). The proposed scheme also can be customized for specific applications, and guidelines for re-adjusting the method parameters have been provided above.

Since the proposed method relies on a stochastic hypergraph partitioning algorithm, the hMETIS tool [31] is used in this work for hypergraph partitioning. The parameters of the hMETIS tool are summarized in Table 4.

Table 5
Shadow detection results of the proposed method for all datasets and the average results. For each dataset, the first row is the shadow detection rate η , the second row is the shadow discrimination rate ξ and the third row is the average of the two measures $\frac{\eta + \xi}{2}$.

Datasets	Methods	Bullkich et al. [21]	Huang and Chen [11]	Lalonde et al. [24]	Guo et al. [16]	Sanin et al. [7]	P. Worst	P. Best	P. Voting
 Highway 1	η	72.00%	50.49%	60.54%	67.69%	81.84%	86.75%	87.92%	87.89%
	ξ	95.08%	84.81%	75.56%	75.06%	94.26%	93.50%	94.67%	94.34%
	$\frac{\eta + \xi}{2}$	83.54%	67.65%	68.05%	71.38%	88.05%	90.12%	91.29%	91.11%
 Lab	η	67.94%	82.11%	17.18%	86.43%	85.99%	91.14%	91.13%	92.50%
	ξ	73.69%	84.96%	81.73%	50.07%	96.65%	93.85%	94.46%	93.23%
	$\frac{\eta + \xi}{2}$	70.82%	83.53%	49.45%	68.25%	91.32%	92.49%	92.80%	92.86%
 Highway 3	η	79.61%	43.88%	38.92%	41.60%	62.23%	67.62%	69.10%	64.94%
	ξ	60.88%	70.95%	86.00%	81.85%	90.68%	85.29%	86.78%	90.14%
	$\frac{\eta + \xi}{2}$	70.25%	57.42%	62.46%	61.73%	76.45%	76.46%	77.94%	77.54%
 Campus	η	74.59%	80.90%	13.90%	89.39%	77.74%	80.39%	82.25%	84.36%
	ξ	77.60%	48.22%	86.58%	20.67%	83.39%	80.66%	80.69%	78.95%
	$\frac{\eta + \xi}{2}$	76.10%	64.56%	50.24%	55.03%	80.56%	80.52%	81.47%	81.66%
 Room	η	72.86%	80.72%	16.14%	72.38%	91.36%	91.44%	92.11%	92.47%
	ξ	97.48%	89.91%	88.17%	80.88%	94.50%	93.98%	93.91%	93.47%
	$\frac{\eta + \xi}{2}$	85.17%	85.31%	52.16%	76.63%	92.93%	92.71%	93.01%	92.97%
 Hallway	η	90.43%	82.62%	91.05%	91.51%	95.18%	94.53%	94.71%	95.16%
	ξ	77.06%	87.73%	19.76%	19.09%	95.73%	95.92%	96.87%	95.94%
	$\frac{\eta + \xi}{2}$	83.74%	85.18%	55.41%	55.30%	95.46%	95.22%	95.79%	95.55%
 CAVIAR	η	91.57%	84.52%	30.80%	69.62%	93.77%	93.66%	93.80%	94.76%
	ξ	86.76%	68.22%	78.75%	23.45%	82.07%	82.52%	82.62%	81.15%
	$\frac{\eta + \xi}{2}$	89.16%	76.37%	54.78%	46.54%	87.92%	88.09%	88.21%	87.96%
Average	η	78.43%	72.18%	38.36%	74.09%	83.82%	86.50%	87.28%	87.44%
	ξ	81.22%	76.40%	73.79%	50.15%	90.84%	89.38%	90.00%	89.60%
	$\frac{\eta + \xi}{2}$	79.83%	74.29%	56.08%	62.12%	87.33%	87.94%	88.64%	88.52%

3.2. Experiments

The sequences Highway 1 and Highway 3 contain vehicles on a road and their cast shadows. Some passing vehicles have dark colors similar to their shadows, which tend to cause difficulties for most chromaticity-based shadow detection methods to discriminate between the vehicles and their shadows.

The method proposed in [7] combines color and gradient information and obtains better results, but it requires large color differences between shadows and vehicles to perform well. Our stochastic hypergraph-based shadow detection scheme combines color and gradient information, as well as pixel vicinity information, to classify the segmented image regions as shadow or non-shadow regions, and that may explain its improved shadow detection and shadow/vehicle discrimination rates. Table 5 shows the results of our proposed method.

The video sequences ‘Lab’, ‘Campus’, ‘Hallway’, ‘Room’ and ‘CAVIAR’ contain light shadows cast by people walking in different environments. Despite the shadows being light, the foreground objects and shadows do not have similar colors, avoiding the chromatic problems mentioned before and improving the shadow detection results for all methods (see Table 5).

To illustrate the variability occurring in the shadow masks $\hat{\lambda}_p(i, j)$ due to the stochasticity of the hypergraph partition algorithm, we measured the worst and best results of 10 different hypergraph partitionings obtained in our tests. We also measured the results obtained using the stochastic majority voting scheme expressed by (14) for $N=10$ and $\tau_r = 3$ (see Table 5). Qualitative results obtained by our proposed method and the comparative methods can be seen in Fig. 7. Visually, the shadow masks generated by our method tend to be more accurate than those obtained by the comparative methods, representing the state of the art in shadow detection [11,7,21,16,24].

Quantitative results of the tests with our proposed method and for the comparative methods [11,7,21,24,16] for each dataset, and the average results considering all the seven sequences can be seen in Table 5.

For the seven video sequences our method with the proposed voting scheme (see (14)) obtained better detection rates (η) than all comparative methods and comparable discrimination rates (see Table 5), resulting in a higher average measure $\frac{\eta + \xi}{2}$ for all video sequences, except for the CAVIAR sequence (the method proposed by Bullkich et al. [21] obtained a higher average measure for the CAVIAR sequence). Even by considering the worst results of the 10 executions, our method still performed better than the comparative methods for most sequences. By averaging the results for all video sequences, our method obtained a better detection rate (η) than all comparative methods in the worst case, best case and by using the proposed voting scheme (see (14)). The obtained average discrimination rate ξ was higher than the average discrimination rates obtained by [21] and [11], and comparable to the ξ obtained by [7]. In average, our shadow detection results based on the $\frac{\eta + \xi}{2}$ measure were better than those obtained by all comparative methods. For most sequences, the methods proposed in [24] and [16] provided the worst shadow detection results and do not seem suitable to process the video sequences in our test set. Their poor performance could be explained by observing that these methods do not use a background model, and for the tested video sequences a single image was not enough to provide adequate shadow detection results. Also, the method proposed in [24] is suitable for outdoor scenes, therefore it performed better on the Highway 1 and Highway 3 video sequences.

We also performed another set of experiments to demonstrate how the hypergraph image segmentation compares with other segmentation approaches, by replacing the hypergraph segmentation method by other segmentation techniques on our cascade of



Fig. 7. Shadow masks obtained by the proposed method and by the comparative methods for the datasets. (a) original frame; (b) ground-truth mask; (c) shadow mask obtained by the method proposed by Bullklich et al. [21]; (d) shadow mask obtained by the method proposed by Huang and Chen [11]; (e) shadow mask obtained by the method proposed by Sanin et al. [7]; (f) shadow mask obtained by the proposed method.

shadow classifiers, and performing the experiments on the same datasets. The results of these experiments can be seen in Table 6, and suggest that our proposed hypergraph segmentation approach potentially can provide better shadow detection results (on average, considering the worst and best results, and the proposed voting scheme).

In these comparisons, we evaluated the watershed based image segmentation approach proposed in Jung [40] and the stochastic image segmentation based on Markov Random Fields proposed in Wong et al. [2].

The method proposed in Jung [40] uses a decimated wavelet transform to produce a multiresolution image representation up to a selected scale. At the coarsest resolution, a color gradient magnitude image is computed, and spurious responses are removed using an adaptive threshold. The watershed transform is applied to thresholded magnitudes, obtaining an initial segmentation. The initial segmentation is then projected to finer resolutions using the








inverse wavelet transform until the full resolution image is achieved. Finally, a region-merging post-processing procedure based on the CIELAB color distances is applied to merge regions with similar colors, obtaining the final segmentation.

In the method proposed in Wong et al. [2], each image pixel is attributed to a unique region, and these regions are subsequently merged with other regions stochastically, based on a region merging likelihood function. The regions formed in this initial stage are merged using the same stochastic region merging process to obtain the final segmentation. The method uses a Markov Random Field (MRF) model to approach a Maximum a Posteriori (MAP) problem, where local spatial interactions and densities are used to create a model of the image. Based on this MRF model, a stochastic region merge approach is used to solve the MAP problem and obtain the final segmentation.

We also performed experiments using an adaptive approach similar to the method proposed in Rital et al. [25] to estimate λ ,

Table 6

Shadow detection results for all datasets and the average results obtained for the comparative image segmentation approaches replacing the hypergraph segmentation approach in our cascade of classifiers. For each dataset, the first row is the shadow detection rate η , the second row is the shadow discrimination rate ξ , and the third row is the average of the two measures $\frac{\eta + \xi}{2}$.

Datasets	Comparison of segmentation methods				
	Jung [40]	Wong et al. [2]	P. Worst	P. Best	P. Voting
 Highway 1	94.36%	82.12%	86.75%	87.92%	87.89%
	66.47%	92.44%	93.50%	94.67%	94.34%
	80.42%	87.28%	90.12%	91.29%	91.11%
 Lab	86.43%	85.99%	91.14%	91.13%	92.50%
	50.07%	96.65%	93.85%	94.46%	93.23%
	68.25%	91.32%	92.49%	92.80%	92.86%
 Highway 3	56.43%	40.11%	67.62%	69.10%	64.94%
	80.73%	79.84%	85.29%	86.78%	90.14%
	68.58%	59.98%	76.46%	77.94%	77.54%
 Campus	81.25%	82.03%	80.39%	82.25%	84.36%
	77.96%	75.40%	80.66%	80.69%	78.95%
	79.61%	78.72%	80.52%	81.47%	81.66%
 Room	91.61%	92.98%	91.44%	92.11%	92.47%
	90.85%	91.06%	93.98%	93.91%	93.47%
	91.23%	92.02%	92.71%	93.01%	92.97%
 Hallway	92.52%	92.29%	94.53%	94.71%	95.16%
	93.63%	97.65%	95.92%	96.87%	95.94%
	93.08%	94.97%	95.22%	95.79%	95.55%
 CAVIAR	94.26%	96.36%	93.66%	93.80%	94.76%
	68.85%	75.80%	82.52%	82.62%	81.15%
	81.56%	86.08%	88.09%	88.21%	87.96%
Average	84.19%	81.24%	86.50%	87.28%	87.44%
	81.16%	85.53%	89.38%	90.00%	89.60%
	82.68%	83.39%	87.94%	88.64%	88.52%

and obtained better results than the comparative methods in terms of the average $\frac{(\eta + \xi)}{2}$ of 88.10%. However, as expected, the results were not as good as those obtained by the proposed method were the parameters were set to maximize the shadow detection performance.

3.3. Discussion

Our experiments show that with the exception of the λ parameter (which is allowed to adjust to the video average lightness variability based on a fixed threshold), the fixed sets of parameters were suitable for all video sequences, performing well on different shadow and lighting conditions with minimal setup. Furthermore, the number of parameters is comparable to the methods reported in the literature, such as Lalonde et al. [24], Huang and Chen [11], Guo et al. [16] and Sanin et al. [7]. The parameters of our proposed method allows to fine tune the method for different needs, as explained before. Besides, λ could be estimated adaptively, eliminating the need for setting an adequate λ value for each video sequence. Also, the stochastic nature of the hypergraph segmentation approach, prevents local maxima and reduces the impact of individual parameters.

The hypergraph image segmentation approach performs better integrated to our proposed shadow detection framework (using a cascade of shadow classifiers) than the comparative segmentation approaches. This can be explained by observing that our method

combines approaches that improve the shadow detection scheme based on chromatic and spatial continuity criteria discussed in Section 2.3.

The shadow detection scheme presented in Section 2.3 tends to provide higher shadow discrimination rates at the expense of lower shadow detection rates. Therefore, at this stage the shadow regions detected using texture information may be heterogeneous (i.e., can include shadow and non-shadow sub-regions), and also mistakenly can exclude shadow sub-regions from the detected shadow regions, leading to more mistakes in the subsequent shadow processing stages of our method. By combining this shadow detection stage with the hypergraph image segmentation approach presented in Section 2.4, the shadow detection rate can be increased with a small impact on the shadow discrimination rate, improving the results in average.

In fact, the hypergraph segmentation obtains independently (stochastically) sets of connected pixels, homogeneous in color, that should receive the same classification (shadow or non-shadow). Multiple runs of the stochastic hypergraph segmentation method are obtained, generating multiple shadow or non-shadow segmentations (which reflect different choices of inner parameters and initializations). A majority voting scheme in this stage tends to be more robust to the initial conditions (e.g. it can be verified that different hypergraph segmentations lead to the best and worst results in Table 5). Since the hypergraph image segmentation produces multiple homogenous regions (i.e. shadow and non-shadow partitions), finer shadow and non-shadow sub-regions are obtained, and these sub-regions are used to validate the shadow detection results obtained in the earlier stages (see Section 2.3), improving the overall shadow detection results. Therefore, the hypergraph segmentation stage validates the shadow regions detected in the previous stage by trying to identify and correct heterogeneities within the detected shadow regions. Besides, it shall be observed that the shadow detection method in Section 2.3 tends to increase the shadow discrimination rate at the expense of decreasing the shadow detection rate, while the hypergraph segmentation tends to increase the shadow detection rate at the expense of a decrease in the shadow discrimination rate, causing a counter-balancing effect, which also contributes to improve the overall shadow detection results.

Adaptation to the average lightness contrast of each video sequence appears to improve the discrimination between shadow and other foreground regions. The experimental evidence suggests that our proposed method with the stochastic majority voting rule, potentially helps obtaining more accurate shadow detection in video sequences. Our results show that the proposed method potentially can achieve better qualitative and quantitative results than recent shadow detection methods, that are representative of the state-of-the-art in shadow detection.

Also, it shall be reinforced that all method parameters have fixed values, with the exception of the λ parameter that is allowed to adjust to the video average lightness variability based on a fixed threshold. The experimental results indicate that using the suggested parameter fixed values, our proposed shadow detection method potentially can perform satisfactorily in different scene configurations, such as indoor and outdoor scenes, with light and strong cast shadows. Besides, these parameters can be adjusted and provide flexibility to customize the proposed shadow detection method for specific applications (e.g. traffic monitoring). A description of the role of each parameter is in Table 2, and guidelines for adjusting the parameters of our proposed method for different needs are provided in Section 3.1. Next, we present our conclusions.

4. Conclusion

This work proposed a new stochastic shadow detection method, designed to overcome the limitations identified in most shadow detection methods available in the literature.

The proposed shadow detection method receives as input the segmented foreground objects and their cast shadows, and outputs the detected shadow regions. It relies on chromatic and gradient information, that are integrated with the stochastic hypergraph segmentation of the video frames using a cascade of shadow/non-shadow classifiers. A majority voting scheme is used to detect the shadow regions in the input stream.

The method experimental validation was based on seven publicly available datasets, representing different video capture conditions often found in practice (comprising outdoor and indoor scenes, with light and dark cast shadows). The analysis of these experiments based on the $\frac{\eta + \xi}{2}$ measure indicate that the proposed shadow detection scheme can be more robust to different video acquisition conditions than other shadow detection methods, that are representative of the state-of-the-art. The capability to adjust to the video lightness contrast contributes to its robustness and shadow detection performance.

Future work will focus on real-time implementations and applications of the proposed shadow detection scheme.

Acknowledgments

This project was partially supported by Digicon LTDA. and by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

References

- [1] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, Detecting moving objects, ghosts, and shadows in video streams, *IEEE Trans. Pattern Anal. Mach. Intell.* 25 (10) (2003) 1337–1342, <http://dx.doi.org/10.1109/TPAMI.2003.1233909>, ISSN 0162-8828.
- [2] A. Wong, J. Scharcanski, P. Fieguth, Automatic skin lesion segmentation via iterative stochastic region merging, *IEEE Trans. Inf. Technol. Biomed.* 15 (6) (2011) 929–936, <http://dx.doi.org/10.1109/ITB.2011.2157829>, ISSN 1089-7771.
- [3] L. Silva, J. Scharcanski, Video segmentation based on motion coherence of particles in a video sequence, *IEEE Trans. Image Process.* 19 (4) (2010) 1036–1049, <http://dx.doi.org/10.1109/TIP.2009.2038778>, ISSN 1057-7149.
- [4] J. Scharcanski, A. de Oliveira, P. Cavalcanti, Y. Yari, A. Particle-Filtering, approach for vehicular tracking adaptive to occlusions, *IEEE Trans. Veh. Technol.* 60 (2) (2011) 381–389, <http://dx.doi.org/10.1109/TVT.2010.2099676>, ISSN 0018-9545.
- [5] P. Barcellos, C. Bouvi, F.L. Escouto, J. Scharcanski, A novel video based system for detecting and counting vehicles at user-defined virtual loops, *Expert Syst. Appl.* 42 (4) (2015) 1845–1856, <http://dx.doi.org/10.1016/j.eswa.2014.09.045>, ISSN 0957-4174.
- [6] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent, C. Rosenberger, Comparative study of background subtraction algorithms, *J. Electron. Imaging* 19 (3) (2010) 03–33, <http://dx.doi.org/10.1117/1.3456695>.
- [7] A. Sanin, C. Sanderson, B.C. Lovell, Improved shadow removal for robust person tracking in surveillance scenarios, in: 20th International Conference on Pattern Recognition, 2010, pp. 141–144, <http://dx.doi.org/10.1109/ICPR.2010.43>.
- [8] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, S. Sirotti, Improving shadow suppression in moving object detection with HSV color information, in: Proceedings of the 2001 IEEE International Conference on Intelligent Transportation Systems (ITSC 2001), pp. 334–339, <http://dx.doi.org/10.1109/ICPR.2010.43>, 2001.
- [9] A. Prati, R. Cucchiara, I. Mikic, M. Trivedi, Analysis and detection of shadows in video streams: a comparative evaluation, in: Proceedings of the 2001 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001), vol. 2, ISSN 1063-6919, II-571–II-576, <http://dx.doi.org/10.1109/CVPR.2001.991013>, 2001.
- [10] A. Sanin, C. Sanderson, B.C. Lovell, Shadow detection: a survey and comparative evaluation of recent methods, *Pattern Recognit.* 45 (4) (2012) 1684–1695, <http://dx.doi.org/10.1016/j.patcog.2011.10.001>, ISSN 0031-3203.
- [11] J.-B. Huang, C.-S. Chen, Moving cast shadow detection using physics-based features, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), ISSN 1063-6919, 2009, pp. 2310–2317, <http://dx.doi.org/10.1109/CVPR.2009.5206629>.
- [12] J.-W. Hsieh, W.-F. Hu, C.-J. Chang, Y.-S. Chen, Shadow elimination for effective moving object detection by Gaussian shadow modeling, *Image Vis. Comput.* 21 (6) (2003) 505–516, [http://dx.doi.org/10.1016/S0262-8856\(03\)00030-1](http://dx.doi.org/10.1016/S0262-8856(03)00030-1), ISSN 0262-8856.
- [13] E. ShabaniNia, A. Naghsh-Nilchi, Robust watershed segmentation of moving shadows using wavelets, in: 8th Iranian Conference on Machine Vision and Image Processing (MVIP), ISSN 2166-6776, 2013, pp. 381–386, <http://dx.doi.org/10.1109/IranianMVIP.2013.6780015>.
- [14] N. Al-Najdawi, H.E. Bez, J. Singhai, E. Edirisinghe, A survey of cast shadow detection algorithms, *Pattern Recognit. Lett.* 33 (6) (2012) 752–764, <http://dx.doi.org/10.1016/j.patrec.2011.12.013>, ISSN 01678655.
- [15] D. Xu, J. Liu, X. Li, Z. Liu, X. Tang, Insignificant shadow detection for video segmentation, *IEEE Trans. Circuits Syst. Video Technol.* 15 (8) (2005) 1058–1064, <http://dx.doi.org/10.1109/TCSVT.2005.852402>, ISSN 1051-8215.
- [16] R. Guo, Q. Dai, D. Hoem, Single-image Shadow Detection and Removal Using Paired Regions, in: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011), IEEE Computer Society, Washington, DC, USA, ISBN 978-1-4577-0394-2, pp. 2033–2040, <http://dx.doi.org/10.1109/CVPR.2011.5995725>, 2011.
- [17] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, *IEEE Trans. Pattern Anal. Mach. Intell.* 24 (5) (2002) 603–619, <http://dx.doi.org/10.1109/34.1000236>, ISSN 0162-8828.
- [18] C. Jung, J. Scharcanski, Robust watershed segmentation using wavelets, *Image Vis. Comput.* 23 (7) (2005) 661–669, <http://dx.doi.org/10.1016/j.imavis.2005.03.001>, ISSN 0262-8856.
- [19] C. Jung, J. Scharcanski, Wavelet transform approach to adaptive image denoising and enhancement, *J. Electron. Imaging* 13 (2) (2004) 278–285.
- [20] C. Jung, J. Scharcanski, Robust watershed segmentation using the wavelet transform, in: Proceedings of the XV Brazilian Symposium on Computer Graphics and Image Processing, ISSN 1530-1834, pp. 131–137, <http://dx.doi.org/10.1109/SIBGRA.2002.1167135>, 2002.
- [21] E. Bullklich, I. Ilan, Y. Moshe, Y. Hel-Or, H. Hel-Or, Moving shadow detection by nonlinear Tone-Mapping, in: 19th International Conference on Systems, Signals and Image Processing (IWSSIP), ISSN 2157-8672, 2012, pp. 146–149.
- [22] N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. Syst., Man Cybern.* 9 (1) (1979) 62–66, <http://dx.doi.org/10.1109/TSMC.1979.4310076>.
- [23] M. Khare, R. K. Srivastava, A. Khare, Dual tree complex wavelet transform based shadow detection and removal from moving objects, in: IS&T/SPIE Electronic Imaging, International Society for Optics and Photonics, 2014, pp. 90–97.
- [24] J.-F. Lalonde, A.A. Efros, S.G. Narasimhan, Detecting ground shadows in outdoor consumer photographs, in: European Conference on Computer Vision, 2010.
- [25] S. Rital, S. Miguet, H. Cherifi, Weighted adaptive neighborhood hypergraph partitioning for image segmentation, in: S. Singh, M. Singh, C. Apte, P. Pernier (Eds.), 3rd International Conference on Advances in Pattern Recognition, Pattern Recognition and Image Analysis, LNCS, ISBN 3-540-28833-3, 2005.
- [26] A. Bretto, L. Gillibert, Hypergraph-Based Image Representation, in: L. Brun, M. Vento (Eds.), Graph-Based Representations in Pattern Recognition, Proceedings of the 5th IAPR International Workshop (GbrPR 2005), Poitiers, France, April 11–13, Lecture Notes in Computer Science, vol. 3434, 2005, pp. 1–11, Springer, ISBN 3-540-25270-3.
- [27] A. Bretto, H. Cherifi, Noise detection and cleaning by hypergraph model, in: Proceedings of the International Conference on Information Technology: Coding and Computing, 2000, pp. 416–419, <http://dx.doi.org/10.1109/ITCC.2000.844264>.
- [28] K. Kannan, B.R. Kanna, C. Aravindan, Root mean square filter for noisy images based on hyper graph model, *Image Vis. Comput.* 28 (9) (2010) 1329–1338, <http://dx.doi.org/10.1016/j.imavis.2010.01.013>, ISSN 0262-8856.
- [29] J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (1997) 888–905.
- [30] M. Tkalcic, J. Tasic, Colour spaces: perceptual, historical and applicational background, in: The IEEE Region 8 EUROCON 2003, Computer as a Tool, vol.1, 2003, pp. 304–308, <http://dx.doi.org/10.1109/EURCON.2003.1248032>.
- [31] G. Karypis, R. Aggarwal, V. Kumar, S. Shekhar, Multilevel hypergraph partitioning: applications in VLSI domain, *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* 7 (1) (1999) 69–79, <http://dx.doi.org/10.1109/92.748202>, ISSN 1063-8210.
- [32] C. Aykanat, B.B. Cambazoglu, B. Uar, Multi-level direct K-way hypergraph partitioning with multiple constraints and fixed vertices, *J. Parallel Distrib. Comput.* 68 (5) (2008) 609–625, <http://dx.doi.org/10.1016/j.jpdc.2007.09.006>, ISSN 0743-7315.
- [33] D. Lasalle, G. Karypis, Multi-threaded Graph Partitioning, in: Proceedings of the 2013 IEEE 27th International Symposium on Parallel and Distributed Processing, IPDPS '13, IEEE Computer Society, Washington, DC, USA, 2013, pp. 225–236, ISBN 978-0-7695-4971-2, <http://dx.doi.org/10.1109/IPDPS.2013.50>.
- [34] K. Devine, E. Boman, R. Heaphy, R. Bisseling, U. Catalyurek, Parallel hypergraph partitioning for scientific computing, in: 20th IEEE International Parallel and Distributed Processing Symposium, IPDPS 2006, 2006, pp. 1–10, <http://dx.doi.org/10.1109/IPDPS.2006.1639359>.
- [35] G. Karypis, V. Kumar, Multilevel K-way Hypergraph Partitioning, in: Proceedings of the 36th Annual ACM/IEEE Design Automation Conference, DAC '99, ACM, New York, NY, USA, 1999, pp. 343–348, ISBN 1-58113-109-7, <http://dx.doi.org/10.1145/309847.309954>.

- [36] C. Fiduccia, R. Mattheyses, A linear-time heuristic for improving network partitions, in: 19th Conference on Design Automation, 1982, pp. 175–181. ISSN 0146-7123.
- [37] J. Scharcanski, H. Shen, A.P. Alves da Silva, Colour quantisation for colour texture analysis, *Comput. Digit. Tech., IEE Proc. E* 140 (2) (1993) 109–114.
- [38] J. Scharcanski, C. Dodson, Texture analysis for estimating spatial variability and anisotropy in planar stochastic structures, *Opt. Eng.* 35 (8) (1996) 2302–2309.
- [39] J. Scharcanski, Stochastic texture analysis for monitoring stochastic processes in industry, *Pattern Recognit. Lett.* 26 (11) (2005) 1701–1709.
- [40] C.R. Jung, Unsupervised multiscale segmentation of color images, *Pattern Recognit. Lett.* 28 (4) (2007) 523–533, <http://dx.doi.org/10.1016/j.patrec.2006.10.001>, ISSN 01678655 URL <<http://linkinghub.elsevier.com/retrieve/pii/S0167865506002479>>..

Vitor Gomes currently is working towards his M.Sc. degree in Computer Science at the Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, RS, Brazil. His main areas of interest are image processing and computer vision.

Jacob Scharcanski is a Professor in Computer Science at the Federal University of Rio Grande do Sul (UFRGS), Brazil. He holds a cross appointment with the Department of Electrical Engineering at UFRGS, and also is an Adjunct Professor with the Department of Systems Design Engineering, University of Waterloo, Ontario, Canada. He has authored and co-authored over 150 refereed journal and conference papers, books and book chapters on imaging and measurements. In addition to his academic publications, he has several technology transfers to the private sector. Presently, he serves as an Associate Editor for two journals, and has served on dozens of International Conference Committees. Professor Scharcanski is a licensed Professional Engineer, Senior Member of the IEEE, IEEE IMS Distinguished Lecturer (2015–2017). His areas of expertise are Image Processing, Pattern recognition, Imaging Measurements and their applications.

Pablo Barcellos currently is working towards his p.H.D. degree in Computer Science at the Federal University of Rio Grande do Sul (UFRGS), Porto Alegre, RS, Brazil. His main areas of interest are image processing and computer vision.