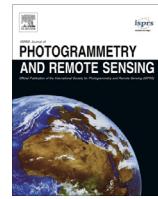




Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

Near real-time shadow detection and removal in aerial motion imagery application

Guilherme F. Silva, Grace B. Carneiro, Ricardo Doth, Leonardo A. Amaral, Dario F.G. de Azevedo*

PUCRS, Av. Ipiranga, 6681, Partenon - Porto Alegre/RS, Brazil

ARTICLE INFO

Article history:

Received 24 March 2017

Received in revised form 26 October 2017

Accepted 7 November 2017

Available online xxxx

Keywords:

Shadow detection

Shadow extraction

Image processing

Specthem ratio

Aerial images

GPU programming

ABSTRACT

This work presents a method to automatically detect and remove shadows in urban aerial images and its application in an aerospace remote monitoring system requiring near real-time processing. Our detection method generates shadow masks and is accelerated by GPU programming. To obtain the shadow masks, we converted images from RGB to CIELCh model, calculated a modified Specthem ratio, and applied multilevel thresholding. Morphological operations were used to reduce shadow mask noise. The shadow masks are used in the process of removing shadows from the original images using the illumination ratio of the shadow/non-shadow regions. We obtained shadow detection accuracy of around 93% and shadow removal results comparable to the state-of-the-art while maintaining execution time under real-time constraints.

© 2017 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

1. Introduction

A Wide-Area Motion Imagery (WAMI) System (Blasch et al., 2014), provides a big-picture of large areas of interest, which can increase the situation awareness, facilitating law-enforcement, tracking and mapping wildfires, borders control and military engagement by monitoring crowd behavior and tracking targets. We present a method to automatically detect and remove shadows in images captured by the WAMI System with local, on-line, near real-time image processing. Our main contributions are the use of a multilevel threshold determination technique to segment a modified Specthem Ratio image to detect shadows, as this ratio elicits shaded pixels. We also present how to effectively remove the shadows by using the statical information of the (unshaded) shadow boundary extracted by a proposed structuring mask. The illumination ratio of shaded/unshaded regions are used to enhance shaded pixels. We provide an implementation under near real-time constraints, i.e. processing each image before the system captures the next one.

In remote sensing applications, shadows are often considered as a nuisance, especially at low resolutions. They are known to modify the form and color of the regions of interest (ROI) (Pan et al., 2014), e.g. a green object may appear black if there's low luminance. The shadow effect can also lead to misidentification within regions where water is present due to the similarity of their spectral signatures. Moreover, the images captured from airspace are subject to change depending on the position of the light source and the movements performed by the acquisition hardware.

In color aerial images, color tone is a powerful descriptor that simplifies and dominates the identifying characteristics of visual interpretation applications. When humans perceive a color object, they describe it in terms of hue, saturation and brightness properties described by several similar color models such as HSL, HSI and HSV (hue-saturation-intensity/value). The amount of different HSL type color spaces across the literature and their device-dependence has caused many authors providing different equations for the same color space (Ford and Roberts, 1998). Although being very good for user interfaces, specially color selection, those HSL related color spaces provide a mere approximation of the illumination information in the image and often confound saturation and lightness or hue and lightness (Brewer, 1994). True lightness calculation requires an appropriate color space such as CIEL*a*b* (CIELAB), or its polar counterpart CIELCh (Ford and Roberts, 1998). This color space describes mathematically all perceivable colors in the three dimensions *L* for lightness and *a* and *b* for the

* Corresponding author.

E-mail addresses: guilherme.froes@acad.pucrs.br (G.F. Silva), grace.braz@acad.pucrs.br (G.B. Carneiro), ricardo.doth@acad.pucrs.br (R. Doth), leonardo.amaral@acad.pucrs.br (L.A. Amaral), dario@pucrs.br (D.F.G. de Azevedo).

color opponents green-red and blue-yellow. In CIELCh, C is the module and h is the angle of the (a, b) coordinate.

The problems in identifying shadows include boundary ambiguity, color variability, variation of lighting, weather effects, and others. According to Jadhav and Jadhav (2016) shadows in aerial images have the following properties:

- Lower luminance (intensity) because the electromagnetic radiation from the sun is blocked (Tsai, 2006).
- Higher saturation with short blue-violet wavelength due to the Rayleigh effect of atmospheric scattering (Polidorio et al., 2003).
- Increased hue values, because the intensity change of a shaded area when compared to an unshaded area is proportional to the wavelength (Huang et al., 2004).
- Increased entropy, which denotes the randomness of the pixels in that region (Zhu et al., 2010).

The most popular approach for shadow detection is to use a variety of variant and invariant cues (features) to capture characteristics of shadows (Khan et al., 2016). A number of authors, Zhu et al. (2010), Lalonde et al. (2010), Guo et al. (2013) and Salvador et al. (2004) focused on the chromatic and textural properties of shadows to determine the illumination conditions in the scene. Another trend is to evaluate the illumination properties as did Xiaoyue Jiang and Wyatt (2011) and Panagopoulos et al. (2010). A blackbody radiator model has also been put forward to detect shadows (Makarau et al., 2011). Finlayson et al. (2006) applied an interactive diffusion process to fill in the derivatives in penumbra region, even though still causing texture loss. Later, Finlayson et al. (2009) have also used inpainting for shadow removal. Shadow invariant approaches attempt to retrieve the true color at each pixel as if there were no shadows in the image (Tian et al., 2009). As shadows are important to build 3D scenes, some authors detect and focus on them to estimate buildings heights as Liasis and Stavrou (2016). In video surveillance applications, techniques take advantage of multiple images (Finlayson et al., 2007) or time-lapse sequences to detect shadows (Joshi and Papanikolopoulos, 2008).

Shadow detection can be broadly classified into model-based methods, property-based methods or even machine learning methods (Adeline et al., 2013). Model-based methods use geometric modeling (Fang et al., 2008; Zhan et al., 2005), for previously known scenes. It has many limitations as the necessity of a point light source, a flat background and different orientations of the object and the cast shadow. Property-based methods involve classification and segmentation (Leone and Distant, 2007; Ying-Li Tian and Hampapur, 2005), which usually involve a slow pixel-per-pixel process, comparing a candidate shadow region's texture to a background reference frame, and histogram thresholding (Tsai, 2006; Chung et al., 2009; Nandini et al., 2014; Chen et al., 2007; Dare, 2005). Different applications make use of different methods. For instance, when we have no prior knowledge of the image, geometric modeling cannot be used.

Histogram thresholding is one of the most used methodology among authors for shadow detection. Based on the properties of shadow pixels, it preserves the pixels that have low luminosity over the pixels with high luminosity. Also, it pursues the color constancy concept, where the color properties of objects do not change with low luminance. Tsai (2006) proposed an efficient shadow detection algorithm to identify shadows converting the RGB color space into several color spaces (as HSI, HSV, HCV, YIQ) and applied threshold on the hue over intensity ratio. The work by Tsai (2006) has poor accuracy on complex high-resolution images. Chung et al. (2009) modified Tsai's method by performing successive local thresholding on shadow candidate pixels, increasing the shadow

detection accuracy but increasing processing time as well. Furthermore, there was the necessity of empirically setting some parameters, which depends on the content of the image, impairing automation. Thresholding is the most efficient methodology in terms of computational resources and the one this paper builds upon, focusing on automation, and real-time implementation, while maintaining values of shadow detection accuracy comparable to the state of the art.

It seems clear that the hue, saturation and intensity characteristics provide relevant information on the shadows of an image. Therefore, this work presents the implementation of a computational method to detect shadows in urban aerial images based on the image representation in the CIELAB color space. Additionally, we use the power that lies in the Graphics Processing Unit (GPU) in order to accelerate complex (or time/resource demanding) calculations. The level of parallelism that can be achieved by GPUs implies several pixels being processed at the same time, causing a reduction in execution time compared to sequential implementation.

To improve the performance in terms of faster responses in the processing of images, we implemented the method making use of CUDA®, or Compute Unified Device Architecture, a parallel computing platform and programming model idealized and provided by NVIDIA™.

In the next section, we discuss our shadow detection methodology in details, going over each step to obtain a final shadow mask. In Section 3, we describe our removal method. Section 4 describes the proposed methodology, discuss CUDA and GPU, and the application for which the method is mainly designed for. The assessment strategy is presented in Section 5. Section 6 shows the computed images and the final shadow masks for each of the three input color aerial urban images as well as the shadow-removed images. We also evaluate the speed improvement and performance regarding correctness and precision of the implemented method using CUDA. Finally, we draw conclusions in Section 7.

Table 1
Shadow detection algorithm.

<i>Mask = Shadow(original)</i>	
1	$L, C, h \leftarrow$ RGB conversion to CIELCh
2	$h_e, L_e \leftarrow$ Mean filter h and L channels
3	Specthem \leftarrow Specthem Ratio $\frac{h_e+1}{L_e+1}$ (Eq. 8)
4	modSpecthem \leftarrow Preprocessing of Specthem
5	Hist \leftarrow Calculate histogram of modSpecthem image
6	Level {1,2,3} \leftarrow Multilevel Otsu's on Hist
7	ShadowBW \leftarrow Threshold Specthem with Level 3
8	Mask \leftarrow Morphological Closing operation on ShadowBW
9	Return Mask

Table 2
Shadow removal algorithm.

<i>Enhanced = Removal(Mask)</i>	
1	$subMask \leftarrow$ Separate <i>Mask</i> into labelled connected regions
2	For each <i>subMask</i> in <i>Mask</i>
3	Get shaded region from <i>subMask</i>
4	<i>borderMask</i> \leftarrow Subtract <i>subMask</i> from dilated <i>subMask</i>
5	Get unshaded borders from <i>borderMask</i>
6	<i>Ratio</i> \leftarrow Calculate illumination ratio between border and shadow region
7	Relight pixels based in <i>Ratio</i>
8	Return Enhanced

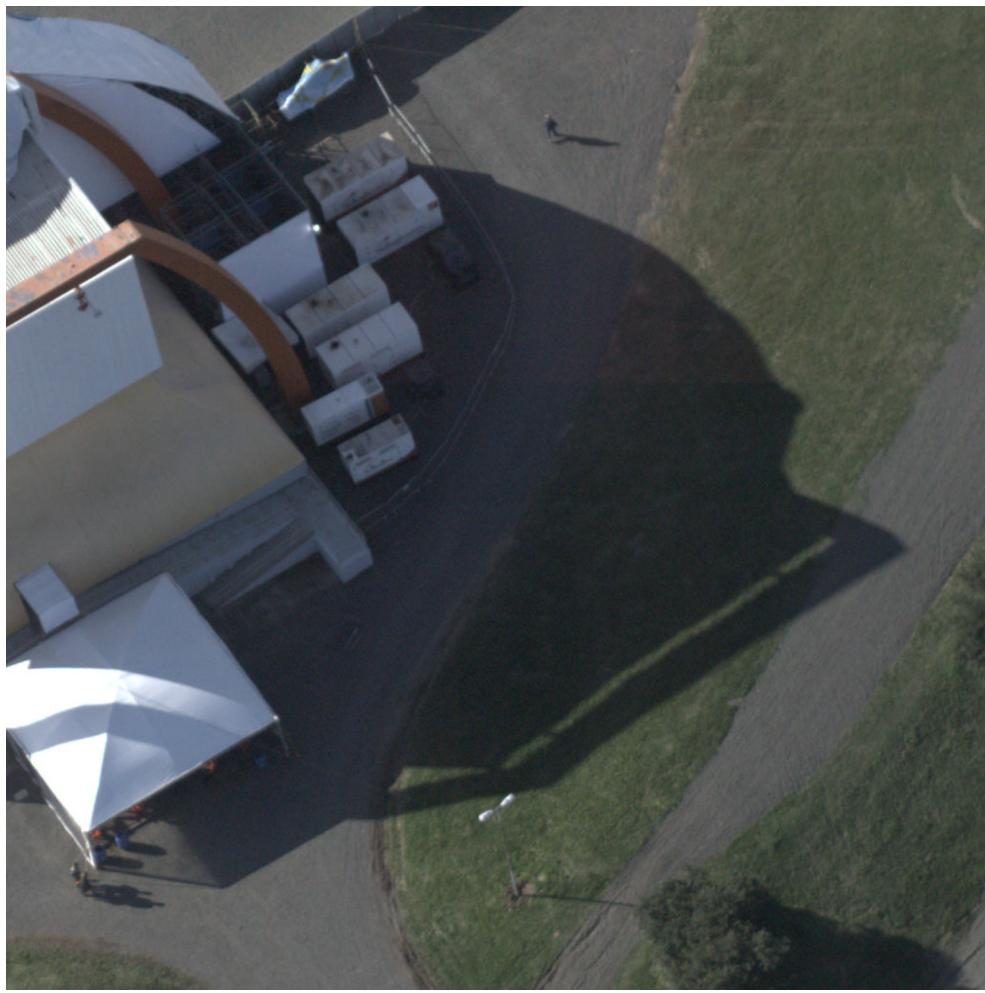


Fig. 1. Original Image 01 acquired by the WAMI System in Porto Alegre, RS, Brazil.

2. Shadow detection

The method for detecting shadows implemented in this paper builds on top of Tsai's method of using color spaces that decouple illumination from chromaticity information (Tsai, 2006). It was adapted in our method to use multilevel global thresholding for segmentation in our modified Specthem Ratio image. This is inspired by the successive local thresholding proposed by Chung et al. (2009). Thus, our proposed method follows these steps:

- (a) Image transformation from RGB model to CIELAB model, in order to separate color from intensity information.
- (b) Conversion from CIELAB color space to its polar representation CIELCh, so we can use the hue channel to exploit the fact that shadows have larger hue values.
- (c) Smoothing of L and h channels, intended to reduce image noise.
- (d) Modified specthem ratio calculation $\frac{h+1}{L+1}$, using CIELCh instead of HSI color space.
- (e) Shadow segmentation by thresholding the Specthem ratio image with the larger threshold found by the multilevel implementation of the Otsu's Method of Thresholding.
- (f) Morphological Erosion of the shadow mask, followed by Dilation for noise reduction and to achieve better delimitation of the shadow regions.

2.1. RGB-space image transformation to CIELCh-space

The color aerial images in the RGB color space are mapped into the CIELCh space, a polar representation of the CIELAB color space, designed by the *Commission Internationale de l'Éclairage* (CIE) to approximate how humans perceive colors. We perceive color as approximately constant irrespective of the illuminant, provided that a sufficient amount of light is available.

The CIELAB color space is device-independent and is calculated from the CIE tri-stimulus CIEXYZ. Despite the L channel having a good correlation with perceived lightness, this color space is not very intuitive (Ford and Roberts, 1998). Alternatively, its polar counterpart, the CIELCh space, describes colors using illumination, chroma, and hue.

The first step is to convert RGB to the CIEXYZ color space, defined such that all visible colors can be defined using only positive values (Ford and Roberts, 1998). This transformation, for the standard 2° observer and D65 illuminant is given by

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124564 & 0.3575761 & 0.1804375 \\ 0.2126729 & 0.7151522 & 0.0721750 \\ 0.0193339 & 0.1191920 & 0.9503041 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

where the 2° standard observer is a definition by the International Commission of Illumination (CIE) to represent an average human's chromatic response within a 2° arc inside the fovea. The D65

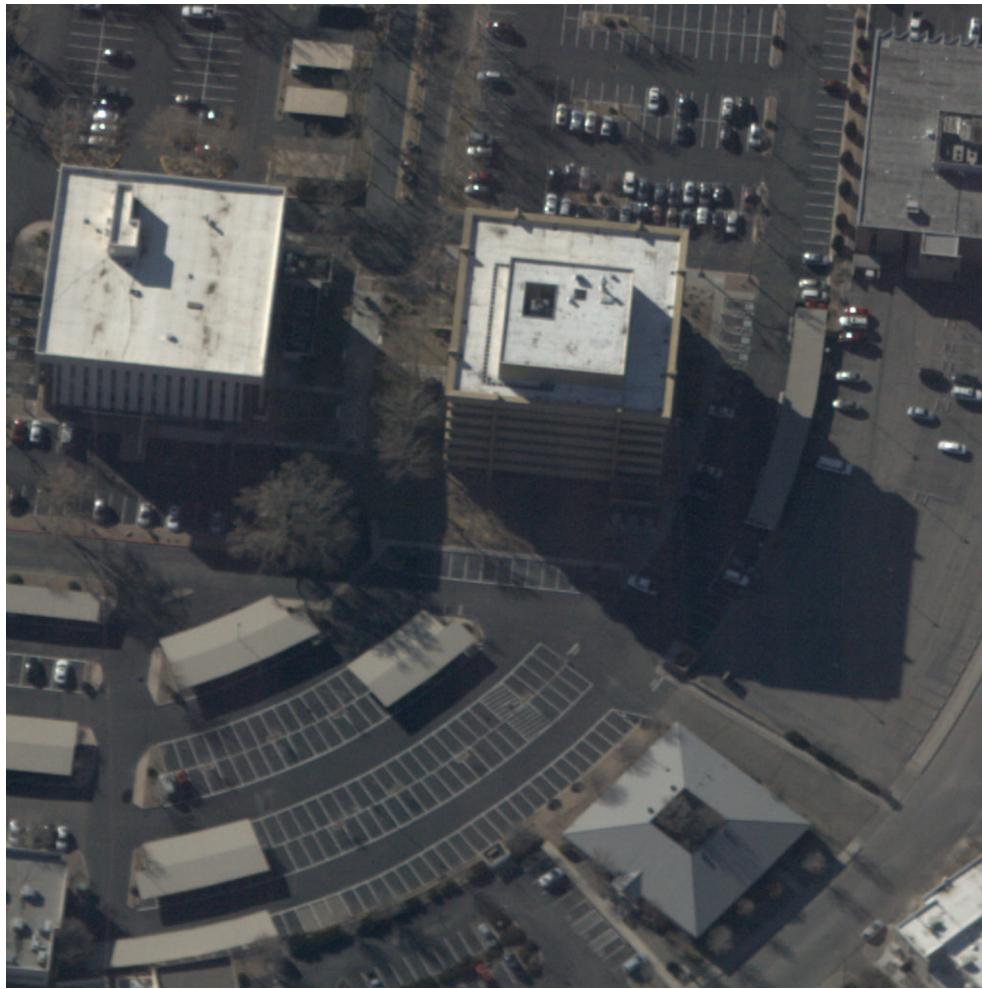


Fig. 2. Original Image 02 acquired by the WAMI System in Albuquerque, NM, US.

illuminant is a commonly used standard illuminant also defined by the CIE. It corresponds roughly to the average midday light in Western Europe/ Northern Europe (comprising both direct sunlight and the light diffused by a clear sky), hence it is also called a daylight illuminant.

From the XYZ tri-stimulus we derive the L , a , and b channels as follows (Ford and Roberts, 1998)

$$L = \begin{cases} 116\left(\frac{Y}{Y_n}\right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856 \\ 903.3\left(\frac{Y}{Y_n}\right) & \text{if } \frac{Y}{Y_n} \leq 0.008856 \end{cases} \quad (2)$$

$$a = 500\left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right) \quad (3)$$

$$b = 200\left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right) \quad (4)$$

where

$$f(x) = \begin{cases} x^{\frac{1}{3}} & \text{if } x > 0.008856 \\ 7.787x + \frac{16}{116} & \text{if } x \leq 0.008856 \end{cases} \quad (5)$$

In the above equations X_n , Y_n , and Z_n refer to the reference white which is $XYZ = \{95.047, 100.00, 108.883\}$ for D65 illuminant with $Y = 100$.

The cartesian coordinates of CIELAB color space can be transformed into the CIELCh color space simply by the geometric transformation

$$C = \sqrt{a^2 + b^2} \quad (6)$$

$$\begin{aligned} h &= \text{atan2}(b, a) \\ h &= \begin{cases} h + 360^\circ & \text{if } h < 0^\circ \\ h - 360^\circ & \text{if } h \geq 360^\circ \end{cases} \end{aligned} \quad (7)$$

where atan2 is a special function in many standard libraries which accounts for when $a = 0$.

2.2. Specthem ratio

In the image segmentation process we calculated the ratio between hue and intensity values of the pixels to elicit shadows. We used L and h channels from the CIELCh space, rather than the H and I channels from HSI proposed by Tsai (2006). Our modified Specthem ratio then becomes:

$$S_r = \frac{(h+1)}{(L+1)} \quad (8)$$

where S_r is the Specthem Ratio image, and the h and L channels were previously normalized to the $[0, 1]$ interval.



Fig. 3. Original Image 03 acquired by the WAMI System in Albuquerque, NM, US.

This image is defined as the ratio between hue and intensity components of the original image and will emphasize the increased hue property of shadows with low luminance (intensity), i.e., the pixels in shaded regions will have higher values than those in unshaded regions.

Before taking the histogram of the Specthem Ratio image as input to the Otsu's Method below, we applied a pipeline of image enhancement techniques to the Specthem Ratio image so better shadow segmentation could be achieved. We calculated the natural logarithm of every pixel of the Specthem Ratio image in order to compress the original values to a narrower scope,

$$S_{rLog} = \log(S_r + 1) \quad (9)$$

note we added 1 to every S_r pixel as to avoid the calculation of $\log(0)$.

Furthermore, we blurred the S_{rLog} image by convolving it with a convolution matrix filter B_f defined by,

$$B_f = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (10)$$

2.3. Otsu's method of thresholding

Otsu's Method (Otsu, 1979) is an automatic threshold determination method, applied to the histogram of an image, which separates the image into two classes. Such is an optimum threshold that maximizes the between-class variance of the two histogram classes. Tsai (2006) used the original one-level Otsu's Method to segment shadows, while Chung et al. (2009) proposed a Successive Thresholding Scheme, also using the one-level Otsu's Method on shadow candidate pixels. They coarsely identified shadows with global thresholding, as did Tsai. However, after that they grouped the candidate shadow pixels and performed local thresholding to extract the true shadow pixels. Unlike Tsai (2006) or Chung et al. (2009), we propose a multilevel global thresholding technique using and adaptation of the Otsu's Method. In our proposal, Otsu's Method is applied to the histogram of the Specthem Ratio image to automatically determine the threshold for segmenting the shadow regions within a logical shadow mask for these pixels with value greater than the segmenting threshold. The classical Otsu's method finds an optimal threshold T that maximizes the equation:

$$V(T) = \frac{(\bar{\mu} \cdot w(T) - \mu(T))^2}{w(T) \cdot \mu(T)} \quad (11)$$

Where:



Fig. 4. Specthem ratio image of Image 01, calculated from CIELCh image.

$$\begin{aligned} w(T) &= \sum_{i=0}^T p_i \\ \mu(T) &= \sum_{i=T+1}^{255} p_i \\ \bar{\mu} &= \sum_{i=0}^{255} i p_i \end{aligned} \quad (12)$$

$$\begin{aligned} \mu_T &= \sum_{i=0}^{L-1} i p_i = \sum_{k=0}^{K-1} u_k \omega_k \\ \sigma_B^2 &= \sum_{k=0}^{K-1} \omega_k (\mu_k - \mu_T)^2 = \sum_{k=0}^{K-1} \omega_k \mu_k^2 - \mu_T^2 \end{aligned} \quad (14)$$

The optimal thresholds can be determined by maximizing the between-class variance as:

$$\{t_0^*, t_1^*, \dots, t_{K-2}^*\} = \arg \max_{0 \leq t_0 < \dots < t_{K-2} < L-1} \sigma_B^2(t_0, t_1, \dots, t_{K-2}) \quad (15)$$

However, as it is a recursive method, each additional class implies one additional for-loop in the algorithm, hugely impacting speed (Liao et al., 2001). Also, the recursive characteristic makes the method unsuitable for parallel implementation.

We propose an automatic multilevel global threshold determination method which finds three thresholds. We threshold the pre-processed Specthem Ratio image into a binary mask using the larger of the determined thresholds. This is necessary as dark objects, water bodies and shadows tend to be in the same side of the histogram. When thresholding shadows using the last threshold level, shadows are better segmented from dark objects and water bodies.

2.4. Noise reduction and morphology operations

Images segmented by thresholding often end up leaving some noise in the form of loose pixels. As the image is binary, one form

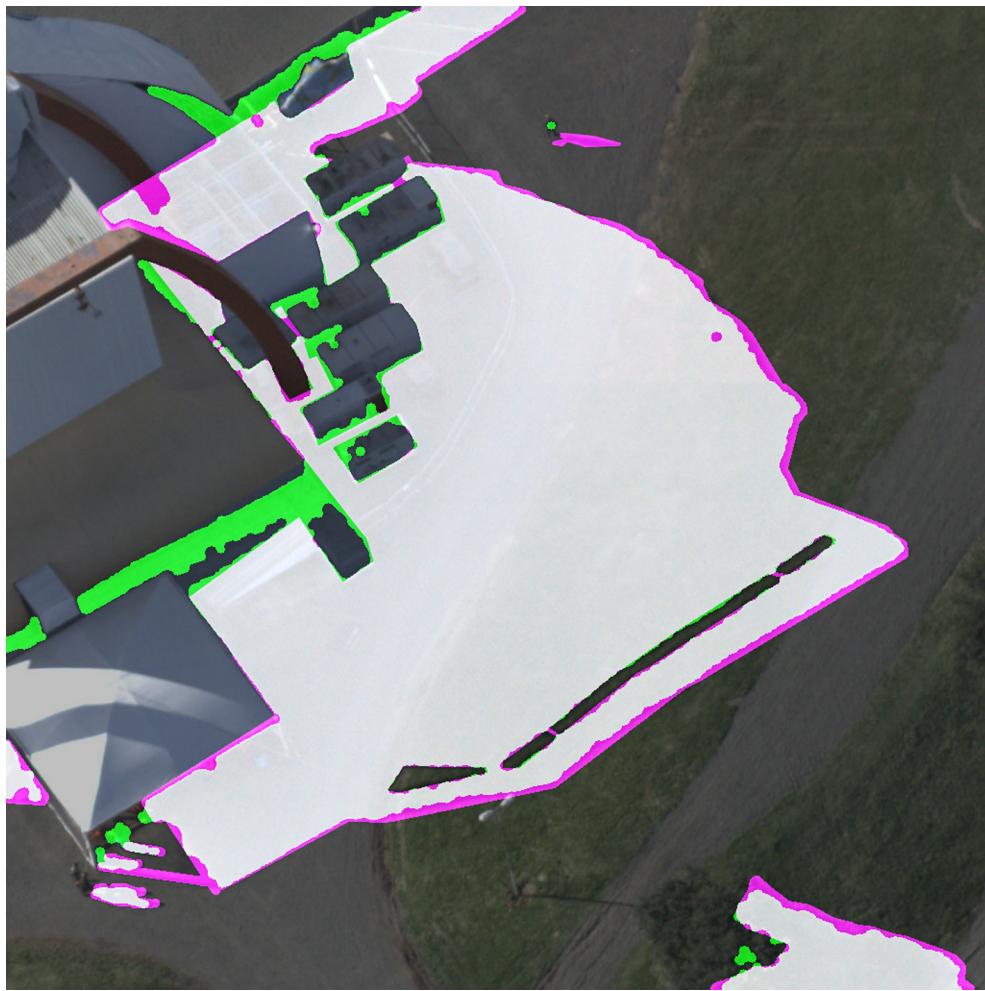


Fig. 5. Overlay of Image 01 with the ground-true shadow mask (in magenta) and the shadow mask detected by our proposed method (in green). White pixels is where both masks intersect. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to remove those pixels is to perform morphology operations, such as Opening and Closing (Gonzalez and Woods, 1992).

We perform morphological Closing to the shadow mask which was segmented from the Specthem ratio in order to remove noise and properly include boundaries (penumbra) in the final shadow mask.

Closing operation is defined as the dilation of the image A by the structuring element B followed by the erosion of the result by B . Closing tends to smooth the object borders, but also eliminates holes. According to Gonzalez and Woods (1992), we define this operation as,

$$A \bullet B = (A \oplus B) \ominus B \quad (16)$$

where dilation \oplus “grows” the objects in a binary image according to the structuring element B . Erosion \ominus is its dual operation, where it “shrinks” the objects according to the structuring element B .

3. Shadow removal

The shadow detection method generated a binary mask in which each pixel is coarsely classified as being shadow or non-shadow. In a real scene, though, the penumbra region causes a graduate change in illumination, which should be accounted for. An image with shadows can be calculated as:

$$I_i = (k_i L_d + L_e) R_i \quad (17)$$

where I_i is the pixel value of the image with shadow, L_d is the intensity of the direct light and L_e is the intensity of the environmental light. R_i is the reflectance of the pixel and k_i indicates whether the pixel is in shade or not.

We assume that the direct light hits all pixels with the same angle, thus illuminating every pixel the same way. In Eq. (17), if k_i is 0, the pixel is only illuminated by environmental light, thus it is in an umbra, if k_i is 1 the pixel is in a nonshadow area. Our shadow detection method provides us with a hard shadow map, in which k_i is either 1 or 0.

Our shadow removal method is an adaptation of the efforts by Guo et al. (2013) and Ye et al. (2012) and was developed to be suitable for parallel computation. It is composed by:

- (a) Connected components labeling of shaded regions, in order to create submasks for each shaded region for parallel local processing.
- (b) Illumination ratio estimation of the shadow regions and their boundaries to find the coefficient needed to compensate the lack of direct illumination in shaded regions.



Fig. 6. De-shadowed Image 01.

- (c) Relighting of the shaded regions by multiplying pixels according to the region illumination ratio and the shadow mask.
- (d) Penumbra compensation.

To determine the overall illumination of both shaded and unshaded areas we define:

$$L_{unshaded,i} = (L_d + L_e)R_i \approx \left(\frac{\sum_{j=1}^N U_j^p}{N} \right)^{1/p} \quad (18)$$

$$L_{shaded,i} = L_e R_i \approx \left(\frac{\sum_{j=1}^M S_j^p}{M} \right)^{1/p} \quad (19)$$

where $L_{unshaded}$ is the overall illumination of an unshaded region, U_i is the i th unshaded pixel and N is the total number of unshaded pixels. L_{shaded} is the overall illumination of a shaded region, S_i is the i th shaded pixel and M is the total number of shaded pixels of that region. Here, p is the exponential parameter of the Minkowski norm, it determines the weight of each grey value in the light source being estimated (Ye et al., 2012).

It was possible to simplify this by choosing $p = 1$, which also gave the best results for our data set. When $p = 1$ the illumination approximation becomes the Gray World Assumption, which states that the average image color is gray (Agarwal and Abidi, 2006).

For the pixel relighting, we calculate the average illumination of shaded and unshaded regions pairs, according to Eqs. (18) and (19) and relight the pixels of the shaded region based on the ratio between direct light and environment light (Guo et al., 2013):

$$r = \frac{L_{unshaded} - L_{shaded}}{L_{shaded}} \approx \frac{L_d}{L_e} \quad (20)$$

the ratio r , as it is defined, is approximately equal to the ratio between the intensity of the direct light and the intensity of the environmental light.

$$I_{ShadowFree,i} = \left(\frac{r+1}{kr+1} \right) I_i \quad (21)$$

the final deshadowed image pixel $I_{ShadowFree,i}$ is multiplied by $(r+1)$ when $k = 0$, that is, the original pixel I_i is under shade, and remains equal to I_i if $k = 1$, that is, the original pixel I_i is not under shade.

4. Proposed methodology

We ran the experiments in an Alienware R3 17 laptop, with an Intel Core i7-6820HK, 16 GB RAM and a 8 GB NVIDIA GeForce

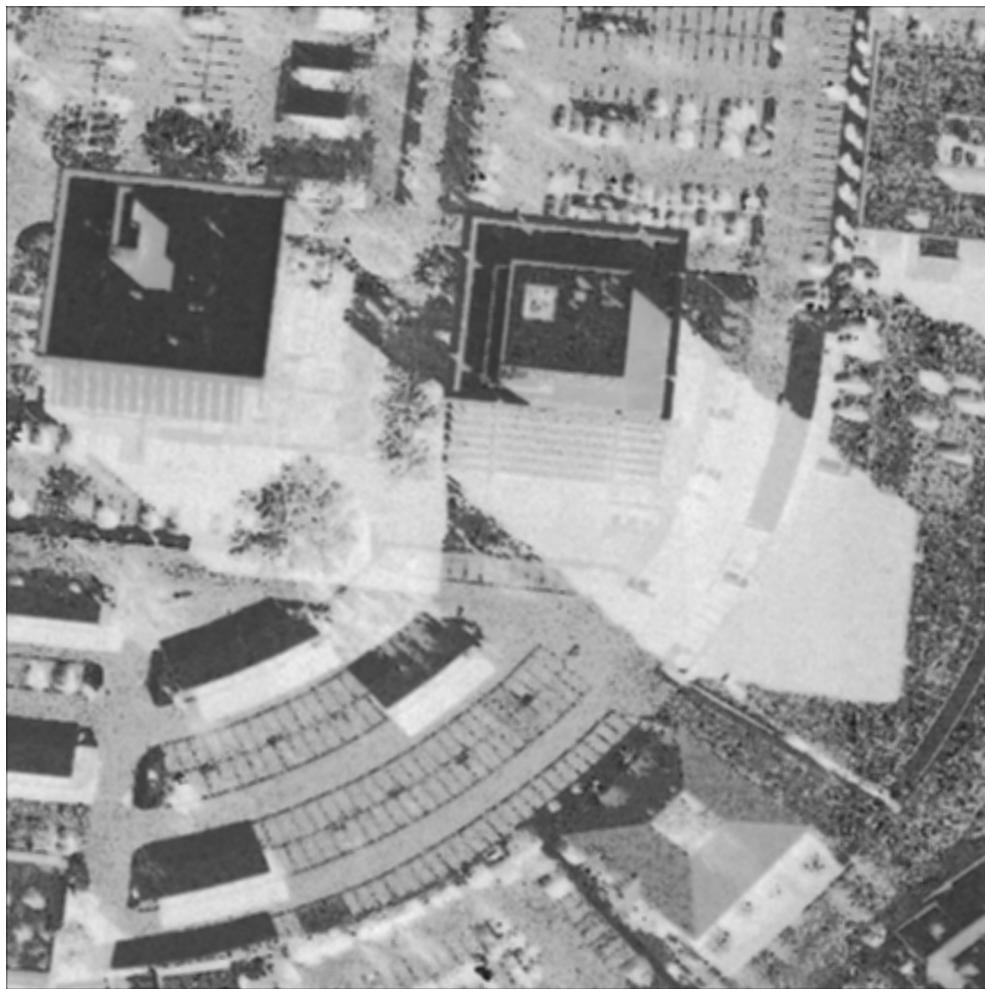


Fig. 7. Specthem ratio of Image 02, calculated from CIELCh image.

GTX 980 M with 1536 CUDA Cores, running Ubuntu 16.04 LTS. In terms of outcome, we show the results for three different images, from Transparent Sky's WAMI, with different sizes and characteristics to evaluate the reliability and performance of the algorithm. We chose CUDA and GPU-based parallel processing due to the aimed application requiring fast processing. With CUDA, we can reach relevant levels of parallelism achieved by GPUs, which implies several pixels of the image being processed at the same time causing a reduction in the execution time compared to sequential implementation. The algorithm of our detection method implementation is presented in [Table 1](#), while the removal method is in [Table 2](#), with high level of abstraction, for simplicity.

Next, we describe the application which this method was designed for, before briefly describing CUDA and GPU architecture.

For speed comparison, we implemented our detection method in sequential C++ and CUDA C++. We also implemented Tsai's method in C++ to compare the correctness and accuracy of our detection method. The work proposed by Chung et al. is intrinsically slower than Tsai's method, as it applies successive thresholding in the specthem image, making it unsuitable for our application.

4.1. WAMI system

Crowd behavior, traffic analysis, and moving images, are examples of requirements of the addressed application that need details of hidden areas. Thus, designed by Transparent Sky™, the system whose images are used in this work is composed of an embedded computer with a sensor imaging earth from an airplane around 3000 feet high. Recently, a drone version is also available. The images are geo-registered (overlapped with satellite earth images) and stabilized. The image processing is done with an NVIDIA GPU, which parallels the work and provides near real-time processing so the throughput of the processed images is faster than the rate of which the sensors capture the original images. This system has a lot of applications, from military to agricultural, and one of its main attractions is the ability to track objects. When an object, say a car, goes under shadow and then back to an unshaded area, the tracking system loses track of the object, due to alteration to its form and color. This is the main motivation of this work, to restore the appearance of an object under shadow to its original appearance under sunlight. The sensor is a high-resolution 29 MP digital camera capturing detailed picture of a 2 square miles area at a rate of once per second, on a gimbal used to stabilize the camera.

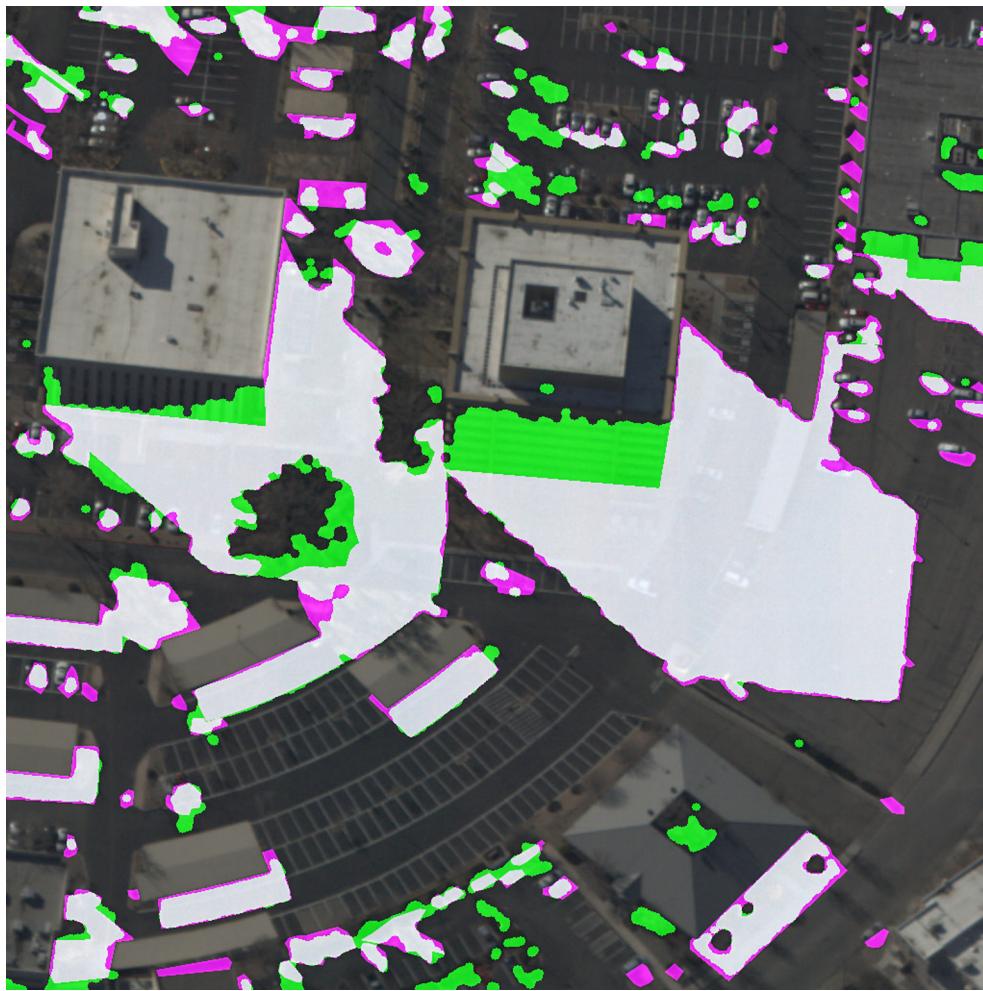


Fig. 8. Overlay of Image 02 with the *ground-true* shadow mask (in magenta) and the shadow mask detected by our proposed method (in green). White pixels is where both masks intersect. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.2. CUDA and GPU architectures

CUDA, or *Compute Unified Device Architecture*, is a parallel computing platform and programming model idealized and provided by NVIDIA. It allows the programmer to write code directly intended to be computed by GPUs (Graphics Processing Units), or to use the API (Application Programming Interface) to port CPU (Central Processing Unit) code directly to GPU code. Moreover, most programming libraries have their version of GPU-enabled functions.

There is a well defined hierarchy of processors in the GPU ([NVIDIA Corporation, 2009](#)). The GPU executes one or more grids of thread blocks. The GigaThread global scheduler distributes these blocks to streaming multiprocessors (SM). SM is a full processor structure containing the actual cores, its own shared memory, internal thread schedulers and registers. Only after work has been scheduled the actual code in the kernel, similar to a CPU function, is executed by a CUDA core or other execution unit in the SM. Each kernel is executed by a number of threads at the same time. In image processing, threads usually have a one-to-one relation with pixels.

5. Assessment strategy

To assess the method regarding the correctness and accuracy, we follow the concept of *error matrix* ([Lillesand and Kiefer, 2000](#)), also known as *confusion matrix* ([Fawcett, 2006](#)), which is a matrix containing the totals of correct decisions along its major diagonal and the totals of errors (confusion) in the secondary diagonal. As ground truth information we used manually interpreted shadows.

We define, then, forms of assessing correctness and accuracy at pixel level classifying pixels that belong to a shadow regions as “shadow” and those that do not, as “non shadow”. The following terminology in [Yao and Zhang \(2004\)](#) is used:

1. Producer's Accuracies:

- (a) Shadow: $\eta_S = \frac{TP}{TP+FN}$
- (b) Non shadow: $\eta_N = \frac{TN}{FP+TN}$

2. User's Accuracies:

- (a) Shadow: $ps = \frac{TP}{TP+FP}$
- (b) Non shadow: $pn = \frac{TN}{TN+FN}$

3. Overall Accuracy: $\tau = \frac{TP+TN}{TP+TN+FP+FN}$



Fig. 9. De-shadowed Image 02.

in which TP (*true positive*) is the number of shadow pixels correctly identified as shadow, FN (*false negative*) is the number of pixels that should have been identified as shadows, but are not. TN (*true negative*) is the number of pixels that were correctly identified as non shadows and FP (*false positive*) is the number of pixels identified as shadows, even though they are not. Also, $TP + TN + FP + FN$ is the sum of all pixels in the image.

Producer's Accuracies are measures of correctness, or specificity of the method, which indicates how well pixels are correctly classified into shadow or non-shadow. While User's Accuracies are measures of the precision of the method, indicating probabilities of the pixels been correctly classified. Also, the Overall Accuracy is the relative effectiveness of the method. The higher those metrics, the better is the method. It is important to note that we are more concerned with User's accuracy, as false-positives are not a problem for our application.

6. Results and discussion

Figs. 1–3 show the three original images used to evaluate the performance of all implementations. They correspond to the input color aerial images obtained from three different urban areas to test the algorithms of our detection method.

Figs. 4–6 show the processing steps for Image 01. Figs. 7–9 show the processing steps for Image 02 and Figs. 10–12 show the processing steps for Image 03. We start processing Image 01 by mapping the aerial image in RGB format (Fig. 1) to CIELCh space. After applying the Specthem ratio, the areas of low luminance are highlighted as shown in Fig. 4. After thresholding and morphological operations, the final mask is obtained. Fig. 5 shows the final mask (in green color) and the ground-truth (in magenta color) blended with the original image for comparison. White pixels represent when both masks coincide. We calculate the illumination ratio between the shaded area and its (unshaded) boundaries using the shadow mask. Finally we relight the shaded pixels, thus, obtaining the de-shadowed image in Fig. 6.

Fig. 1 shows an image with very high resolution and well defined shadows. Even though it contains a fair amount of grass, the proposed technique was able to identify the shadow regions in that area. The process of noise removal eliminated small shaded areas (e.g. people's shadows), because they are not relevant for a WAMI system. It is interesting to notice the sand patches inside the grass area. The green color still poses some challenge to the method, but it is clear that it could successfully differentiate the tree at the bottom from its shadow, while correctly compensating the shadows, as can be seen in Fig. 6.



Fig. 10. Specthem ratio of Image 03, calculated from CIELCh image.

Fig. 3 shows similar results to those obtained in **Fig. 2**. In a complex urban scene, analyzing the effect of the shadows from buildings confounds the detection method, because these images contain similar spectral information (buildings faces and asphalt often have similar color and texture). Some false positives are detected or, in other words, some pixels are classified as belonging to shadow regions even though they do not belong. In addition, we observed that some buildings had at least one face identified as a shadow, also leading to false positives. Regarding deshadowing, we can note, in **Fig. 9**, that the great majority of the shadows present in **Fig. 2** is properly removed, restoring our ability to identify the objects which were under shade. Even the small shadow regions were eliminated. We also note that even the non illuminated faces of the buildings were relighted. We draw attention to the cars that are parked under the shade of the buildings. For **Fig. 3**, the most important aspect to observe is the removal of the trees shadows, as can be seen in **Fig. 12**. We can note that the appearance of the trees remains the same, while the shadows were removed, facilitating car recognition in that area. It is also important to notice that the details of the parking lot and the right-hand side building were enhanced, which were hard to notice before shadow removal.

Fig. 13 shows a comparison between the proposed method and other state of the art methods developed by **Guo et al. (2013)** and

Gong and Cosker (2016), whose implementations were made available by the authors. We also obtained the methods that were implemented, and made available, by **Sanin et al. (2012)**. These were not suitable for comparison as they required a background reference frame, therefore they could not be used to process our data set.

The method by **Guo et al. (2013)** was unable to properly identify most of the shadows, while also overly increasing the intensity of regions which were not under shade, as seen in Image 01 and Image 03. In Image 03, tree shadows were correctly identified and removed but the building shadow in the right hand-side was not. In Image 02, shadows were not removed at all, although an overall increase in the image intensity is noticeable.

We can see that the method developed by **Gong and Cosker (2016)** reduces the presence of artifacts in the shadow boundaries in all Images. However, it increases noise in regions of low light and in regions with misidentified shadows. Most importantly, this method is not automatic as it requires human intervention, while our method does not. The user is asked to manually indicate *lit* and *shadow* regions (those are indicated as blue and red lines, respectively, in **Fig. 13**).

Figs. 14–16 show the histogram of each Specthem Ratio images for Image 01, 02 and 03, respectively. It is easy to notice that the

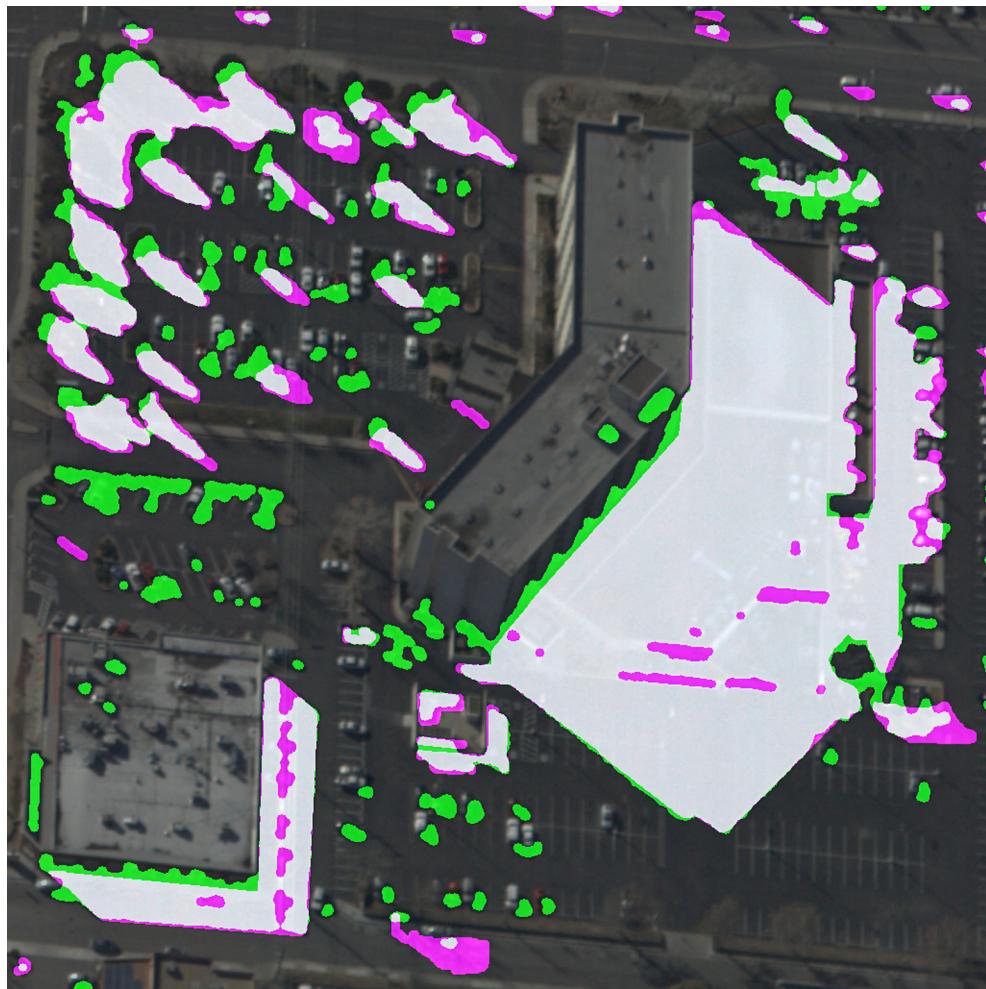


Fig. 11. Overlay of Image 03 with the ground-true shadow mask (in magenta) and the shadow mask detected by our proposed method (in green). White pixels is where both masks intersect. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

definition of a threshold to segment shadows is not trivial. The first valley we have is where the original one level Otsu's Method of threshold determination would find the optimum threshold, but this would segment dark objects, concrete and water bodies as shadows as well. The multilevel thresholding determination guarantees better shadow segmentation.

Table 3 shows the results of each assessed accuracies of shadow detection discussed in Section 5 for Image 01. Table 4 shows the results of each assessed accuracies for Image 02 and Table 5 shows the results of each assessed accuracies for Image 03. The results are compared to Tsai (2006) work. We have also included results for our proposed method using the HSI color space, as to demonstrate that the choice of using CIELCh color space improves User's accuracy regarding shadows, maintaining the overall accuracy of the method. All results are better than the method we compared with, and we can observe our method showed better consistency maintaining accuracies around 90% for most scenes, while Tsai's method is highly dependent on the scene content.

It can be seen that Image 02 presented a smaller overall accuracy than the other images. That is caused due to the complexity and low resolution of the image, caused by the high elevation of

the airplane that acquired the image. Due to the low contrast of the image, cast shadows (shadows of a building on a road) and self shadows (non illuminated faces of a building) often appear as the same region. Our proposed method performed better due to increased noise immunity.

We have also included a comparison of the proposed method with the use of multilevel global thresholding determination and single level thresholding, which can be seen in Tables 6–8. This step is the most important in the increase of detection accuracy.

We can see in Table 9 the time spent for shadow detection processing of each image using the CUDA C++ implementation and the Sequential C++ implementation. We observed speed ups of around 6× times for images with 1024×1024 pixels. We also run experiments on images of size 2048×2048 and observed speed ups of around 7.5×. This is caused by the fact that the main bottlenecks in GPU programming are the memory transfers required between CPU and GPU. When there are more calculations than shadow, CUDA improves the performance even more, as the gains in speed overcome the drawback of memory transfers.

The comparison between the sequential and parallel implementations consider all processing steps, not only shadow segmenta-



Fig. 12. De-shadowed Image 03.

tion. The gain in computational speed comes from parallel implementations of steps like color conversion, Specthem Ratio calculation and morphology operations.

7. Conclusion

We implemented and tested our algorithms in three different aerial real scenes, obtained in different flights and locations, at different times, with different scene complexities and different illumination conditions. We started with the aerial images in RGB format and mapped them into CIELCh space. We then calculated the Specthem Ratio images which highlighted the low luminance areas. We then applied multilevel thresholding to obtain raw masks. We then applied morphology operations to preserve the shapes of the masks and reduce noise, obtaining the final shadow masks. We then used the masks to perform local processing in shadow regions, using the statistical information of their (unshaded) boundaries to relight the shaded pixels.

In this paper we explored the strengths of each technique presented in Chung et al. (2009) and Tsai (2006), covering the

properties of the shadows in terms of luminance and chromaticity by implementing multilevel thresholding processes and morphology. These processes eliminate the need of a priori information of the environment, geometry and light source position to rebuild a 3D model, which can be both time and resource (CPU) consuming to the point of invalidating real-time image processing.

In our proposed method, one can see that the shadows are properly identified and dark areas that are not shadows are minimally identified. Note that false positives, which are pixels being identified as shadows even though they are not, are of minimum consequence in such image application. The most critical situation would be false negatives, or undetected shadows, which are minimized using our method. Nevertheless, additional work in the thresholding technique may be needed for specific applications, in order to reduce false positives.

Next to shadow detection, one of the best results of this work relate to shadow relighting, and can be explained by (1) the local processing of the surrounding area of the shadow mask, and (2) the adoption of the CIELAB remapping, which results in shadow images with less noise than when using the HSI remapping. In

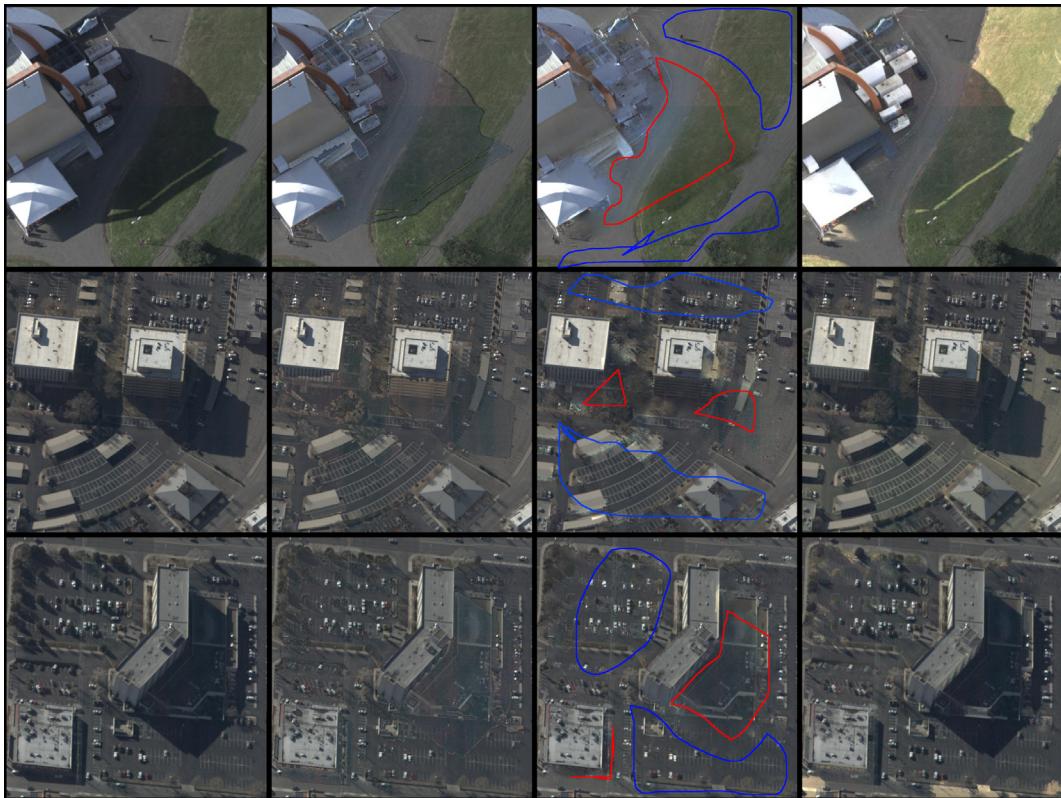


Fig. 13. Shadow removal comparison. Left to right: Original, proposed method, [Gong and Cosker \(2016\)](#) and [Guo et al. \(2013\)](#).

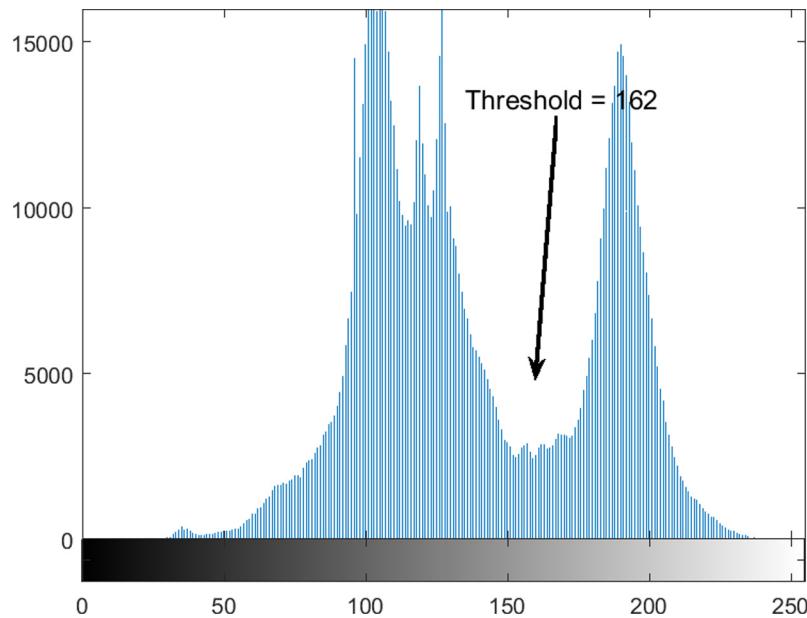
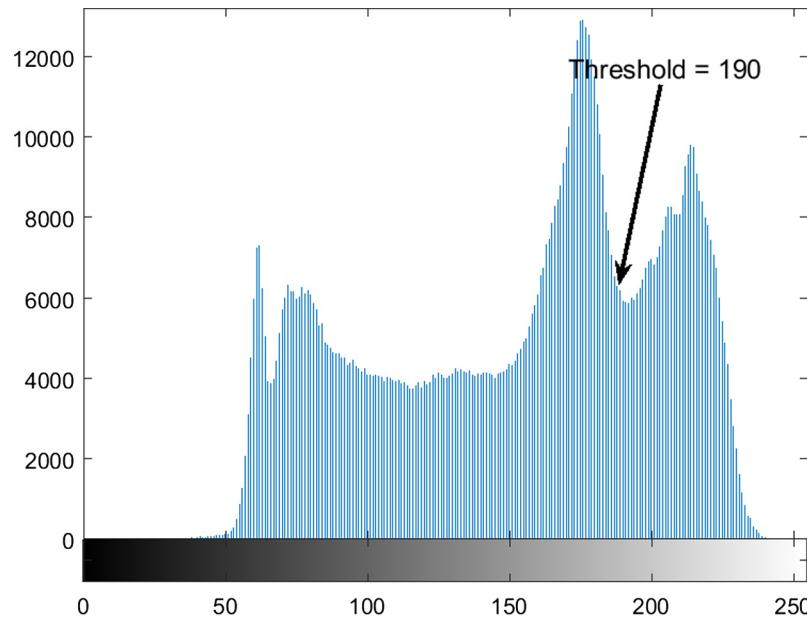
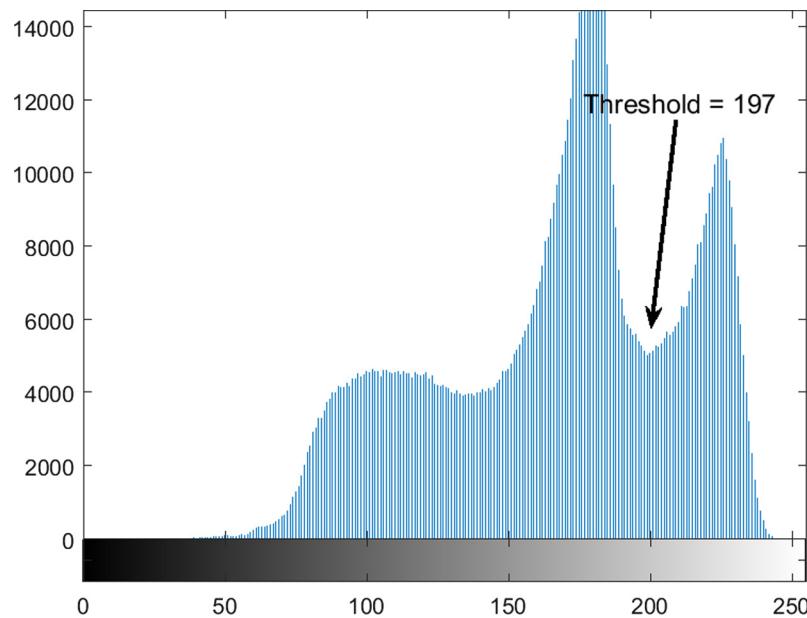


Fig. 14. Histogram of the Specthem Image of Image 01.

other words, our proposed method has an increased noise immunity, as compared to other methods in the literature. We compared our shadow removal results to other state-of-the-art methods. For automatic shadow removal, our method showed the best results.

We observed speed ups of around $6\times$ when comparing GPU implementation with CPU implementation. This increase in processing speed evidences the method possibility of being used in a real-time monitoring system in order to improve other algorithms efficiency.

**Fig. 15.** Histogram of the Specthem Image of Image 02.**Fig. 16.** Histogram of the Specthem Image of Image 03.**Table 3**

Statistics of each assessed accuracy for Fig. 1, comparing Tsai and the proposed method using CIELCh and HSI color spaces.

	Image 01		
	Tsai	Proposed (CIELCh)	Proposed (HSI)
<i>Producer's Accuracies</i>			
· Shadow η_S (%)	95.78	93.52	94.75
· Non shadow η_N (%)	92.50	97.07	95.37
<i>User's Accuracies</i>			
· Shadow p_S (%)	86.62	94.17	91.21
· Non Shadow p_N (%)	97.74	96.73	97.29
<i>Overall Accuracy</i>			
· τ (%)	93.60	95.87	95.16

Table 4

Statistics of each assessed accuracy for Fig. 2, comparing Tsai and the proposed method using CIELCh and HSI color spaces.

	Image 02		
	Tsai	Proposed (CIELCh)	Proposed (HSI)
<i>Producer's Accuracies</i>			
· Shadow η_S (%)	98.61	86.96	87.19
· Non shadow η_N (%)	50.81	93.96	91.89
<i>User's Accuracies</i>			
· Shadow p_S (%)	40.36	82.94	78.41
· Non Shadow p_N (%)	99.08	95.52	95.50
<i>Overall Accuracy</i>			
· τ (%)	62.87	92.19	90.71

Table 5

Statistics of each assessed accuracy for Fig. 3, comparing Tsai and the proposed method using CIELCh and HSI color spaces.

	Image 03		
	Tsai	Proposed (CIELCh)	Proposed (HSI)
<i>Producer's Accuracies</i>			
· Shadow η_s (%)	94.68	86.42	90.51
· Non shadow η_n (%)	80.57	94.36	92.93
<i>User's Accuracies</i>			
· Shadow ps (%)	61.46	83.38	80.73
· Non Shadow pn (%)	97.88	95.50	96.77
<i>OverallAccuracy</i>			
· τ (%)	84.05	92.40	92.33

Table 6

Statistics of each assessed accuracy for Fig. 1, comparing proposed method using multilevel and single level global threshold.

	Image 01	
	Proposed (multi)	Proposed (single)
<i>Producer's Accuracies</i>		
· Shadow η_s (%)	93.52	96.11
· Non shadow η_n (%)	97.07	91.42
<i>User's Accuracies</i>		
· Shadow ps (%)	94.17	85.03
· Non Shadow pn (%)	96.73	97.89
<i>OverallAccuracy</i>		
· τ (%)	95.87	93.00

Table 7

Statistics of each assessed accuracy for Fig. 2, comparing proposed method using multilevel and single level global threshold.

	Image 02	
	Proposed (multi)	Proposed (single)
<i>Producer's Accuracies</i>		
· Shadow η_s (%)	86.96	98.72
· Non shadow η_n (%)	93.96	40.02
<i>User's Accuracies</i>		
· Shadow ps (%)	82.94	35.72
· Non Shadow pn (%)	95.52	98.93
<i>OverallAccuracy</i>		
· τ (%)	92.19	53.84

Table 8

Statistics of each assessed accuracy for Fig. 3, comparing proposed method using multilevel and single level global threshold.

	Image 03	
	Proposed (multi)	Proposed (single)
<i>Producer's Accuracies</i>		
· Shadow η_s (%)	86.42	99.68
· Non shadow η_n (%)	94.36	34.66
<i>User's Accuracies</i>		
· Shadow ps (%)	83.38	33.30
· Non Shadow pn (%)	95.50	99.69
<i>OverallAccuracy</i>		
· τ (%)	92.40	50.69

Table 9

Average time spent in Sequential C++ and CUDA C++ for images of size 1024×1024 pixels.

	Image 01	Image 02	Image 03
Sequential (C++)	232.68 ms	240.00 ms	233.78 ms
CUDA (C++)	36.22 ms	38.78 ms	35.55 ms
Sequential CUDA	6.43 ×	6.19 ×	6.56 ×

Acknowledgment

Our thanks to the United States Air Force Office of Scientific Research (AFOSR) for the funding within the scope of the project number: FA9550-15-1-0141. We also would like to thank the PNPD Program from CAPES/CNPQ. Special gratitude for Transparent Sky for providing the data sets and important insights. Thanks to Professors Alexandre Rosa Franco and Juarez Sagebin Corrêa, from PUCRS, for their incredible job reviewing this paper.

References

- Adeline, K.R.M., Chen, M., Briottet, X., Pang, S.K., Paparoditis, N., 2013. Shadow detection in very high spatial resolution aerial images: a comparative study. *ISPRS Journal of Photogrammetry and Remote Sensing* 80, 21–38.
- Agarwal, V., Abidi, B., 2006. An overview of color constancy algorithms. *Journal of Pattern Recognition Research* 1 (1), 42–54.
- Blasch, E., Seetharaman, G., Suddarth, S., Palaniappan, K., Chen, G., Ling, H., Basharat, A., 2014. Summary of methods in wide-area motion imagery (wami). In: SPIE, Conference on Defense Security+Sensing, Baltimore, United States, pp. 90890C-1–90890C-10.
- Brewer, C.A., 1994. Color use guidelines for mapping. *Visualization in Modern Cartography*, 123–148.
- Chen, Y., Wen, D., Jing, L., Shi, P., 2007. Shadow information recovery in urban areas from very high resolution satellite imagery. *International Journal of Remote Sensing* 28 (15), 3249–3254.
- Chung, K., Lin, Y., Huang, Y., 2009. Efficient shadow detection of color aerial images based on successive thresholding scheme. *IEEE Transactions on Geoscience and Remote Sensing* 47 (2), 671–682.
- Dare, P.M., 2005. Shadow analysis in high-resolution satellite imagery of urban areas. *Photogrammetric Engineering & Remote Sensing* 71 (2), 169–177.
- Fang, L.Z., Qiong, W.Y., Sheng, Y.Z., 2008. A method to segment moving vehicle cast shadow based on wavelet transform. *Pattern Recognition Letters* 29 (16), 2182–2188.
- Fawcett, T., 2006. An introduction to roc analysis. *Pattern Recognition Letters* 27 (8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>.
- Finlayson, G., Fredembach, C., Drew, M.S., 2007. Detecting illumination in images. In: Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, IEEE, pp. 1–8.
- Finlayson, G., Drew, M., Lu, C., 2009. Entropy minimization for shadow removal. *International Journal of Computer Vision* 85 (1), 35–57.
- Finlayson, G., Hordley, S., Lu, C., Drew, M., 2006. On the removal of shadows from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (1), 59–68.
- Ford, A., Roberts, A., 1998. Colour Space Conversions. Westminster University, London, pp. 1–31.
- Gong, H., Cosker, D., 2016. Interactive removal and ground truth for difficult shadow scenes. *Journal of Optical Society of America A* 33 (9), 1798.
- Gonzalez, R.C., Woods, R.E., 1992. Digital Image Processing. Addison-Wesley, Reading, MA.
- Guo, R., Dai, Q., Hoiem, D., 2013. Paired regions for shadow detection and removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (12), 2956–2967.
- Huang, D.Y., Lin, T.W., Hu, W.C., 2011. Automatic multilevel thresholding based on two-stage Otsu's method with cluster determination by valley estimation. *International Journal of Innovative Computing, Information and Control* 7 (10), 5631–5644.
- Jadhav, V.V., Jadhav, S.B., 2016. Survey on shadow detection and shadow removal. *International Journal of Advanced Research in Computer Engineering & Technology* 5 (1), 154–157.
- Huang, J., Xie, W., Tang, L., 2004. Detection of and compensation for shadows in colored urban aerial images. Fifth World Congress on Intelligent Control and Automation (IEEE Cat. No.04EX788), Hangzhou, China, vol. 4. IEEE, pp. 3098–3100.
- Jiang, X. and Schofield, A. J. and Wyatt, J. L., 2011. Shadow detection based on colour segmentation and estimated illumination, 2011. In: Proceedings of the British Machine Vision Conference, Dundee, UK. BMVA Press, pp. 87.1–87.1.
- Joshi, A., Papanikolopoulos, N., 2008. Learning to detect moving shadows in dynamic environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (11), 2055–2063.

- Khan, S.H., Bennamoun, M., Sohel, F., Togneri, R., 2016. Automatic shadow detection and removal from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (3), 431–446.
- Lalonde, J.-F., Efros, A.A., Narasimhan, S.G., 2010. Detecting ground shadows in outdoor consumer photographs. In: European Conference on Computer Vision, Heraklion, Crete, Greece. Springer, pp. 322–335.
- Leone, A., Distante, C., 2007. Shadow detection for moving objects based on texture analysis. *Pattern Recognition* 40 (4), 1222–1233.
- Liao, P., Chen, T., Chung, P., 2001. A fast algorithm for multilevel thresholding. *Journal of Information Science and Engineering* 17, 713–727.
- Liasis, G., Stavrou, S., 2016. Satellite images analysis for shadow detection and building height estimation. *ISPRS Journal of Photogrammetry and Remote Sensing* 119, 437–450.
- Lillesand, T.M., Kiefer, R.W., 2000. *Remote Sensing and Image Interpretation*. Wiley, New York.
- Makarau, A., Richter, R., Muller, R., Reinartz, P., 2011. Adaptive shadow detection using a blackbody radiator model. *IEEE Transactions on Geoscience and Remote Sensing* 49 (6), 2049–2059.
- Nandini, D.U., Leni, A.E.S., 2014. A survey of the existing shadow detection techniques. In: 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCI CCT), Kanyakumari, India, pp. 175–177.
- NVIDIA Corporation, 2009. Nvidia's Next Generation cudaTM Compute Architecture.
- Otsu, N., 1979. A threshold selection method from gray level histograms. *IEEE Transactions on Systems, Man, and Cybernetics SMC-9* (1), 62–69.
- Pan, B., Wu, J., Jiang, Z., Luo, X., 2014. Shadow detection in remote sensing images based on weighted edge gradient ratio. In: 2014 IEEE Geoscience and Remote Sensing Symposium, Québec City, Canada, pp. 505–508.
- Panagopoulos, A., Wang, C., Samaras, D., Paragios, N., 2010. Estimating shadows with the bright channel cue. In: European Conference on Computer Vision, Heraklion, Crete, Greece. Springer, pp. 1–12.
- Polidorio, A.M., Flores, F.C., Imai, N.N., Tommaselli, A.M.G., Franco, C., 2003. Automatic shadow segmentation in aerial color images. In: XVI Brazilian Symposium on Computer Graphics and Image Processing, São Carlos, Brazil, 2003. SIBGRAPI 2003, pp. 270–277.
- Salvador, E., Cavallaro, A., Ebrahimi, T., 2004. Cast shadow segmentation using invariant color features. *Computer Vision and Image Understanding* 95 (2), 238–259.
- Sanin, A., Sanderson, C., Lovell, B.C., 2012. Shadow detection: a survey and comparative evaluation of recent methods. *Pattern Recognition* 45 (4), 1684–1695.
- Tian, Y. and Lu, M. and Hampapur, A., 2005. Robust and efficient foreground analysis for real-time video surveillance. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, USA, vol. 1. IEEE, pp. 1182–1187.
- Tian, J., Sun, J., Tang, Y., 2009. Tricolor attenuation model for shadow detection. *IEEE Transactions on Image Processing* 18 (10), 2355–2363.
- Tsai, V.J.D., 2006. A comparative study on shadow compensation of color aerial images in invariant color models. *IEEE Transactions on Geoscience and Remote Sensing* 44, 1661–1671.
- Yao, J., Zhang, Z., 2004. Systematic static shadow detection. In: Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, vol. 2, pp. 76–79.
- Ye, Q., Xie, H., Xu, Q., 2012. Removing shadows from high-resolution urban aerial images based on color constancy. In: XXII ISPRS Congress, Melbourne, Australia, pp. 525–530.
- Zhan, Q., Shi, W., Xia, Y., 2005. Quantitative analysis of shadow effects in high-resolution images of urban areas. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science* 36 (8/W27).
- Zhu, J., Samuel, K.G.G., Masood, S.Z., Tappen, M.F., 2010. Learning to recognize shadows in monochromatic natural images. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, USA. IEEE, pp. 223–230.