Computer Vision and Image Understanding 138 (2015) 51–60

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Efficient height measurements in single images based on the detection of vanishing points[☆]

Fernanda A. Andaló[a,*], Gabriel Taubin[b], Siome Goldenstein[c]

[a] SAMSUNG Research Institute Brazil (SRBR), CEP 13097-160 Campinas, SP, Brazil
[b] School of Engineering, Brown University, Providence, RI 02912 USA
[c] Institute of Computing, University of Campinas (Unicamp), CEP 13083-852 Campinas, SP, Brazil

ARTICLE INFO

Article history:
Received 30 June 2014
Accepted 28 March 2015
Available online 16 April 2015

Keywords:
Photogrammetry
Vanishing point
Height measurement
Projective space

ABSTRACT

Surveillance cameras have become a customary security equipment in buildings and streets worldwide. It is up to the field of Computational Forensics to provide automated methods for extracting and analyzing relevant image data captured by such equipment. In this article, we describe an effective and semi-automated method for detecting vanishing points, with their subsequent application to the problem of computing heights in single images. With no necessary camera calibration, our method iteratively clusters segments in the bi-dimensional projective space, identifying all vanishing points – finite and infinite – in an image. We conduct experiments on images of man-made environments to evaluate the output of the proposed method and we also consider its application on a photogrammetry framework.

1. Introduction

By analyzing certain image properties – known object size, occlusion, stereoscopic vision, focus, gradient, texture and vergence – we can infer three-dimensional (3D) information of the geometry portrayed in the images. This inferred 3D data can be employed in various computer vision applications: image-based rendering [1,2], automated driving [3], object detection [4], and forensic science [5].

The process of extracting, from images, geometric properties, such as heights, areas, and angles, is denominated by *photogrammetry*. Photogrammetry methods are widely used in forensic investigations, where it can help corroborate pieces of evidence [5]. Analysis of car accidents and of human height are the two main examples of forensic activities that involve photogrammetry [5].

Considering images captured by surveillance cameras, for instance, with the aid of photogrammetry we can estimate the height of objects and people, useful identification characteristic when the face of the suspects cannot be identified or the details in their clothes are not relevant. Conversely, the suspects are often not on the crime scene at the time of the investigation, or the scene has changed. In these cases, the images themselves are the only source of information.

When only one image depicting the scene is available, the essential problem is the recovery of the third dimension, given that this information was not captured in the acquisition process, in which the 3D scene was projected onto the 2D image plane. In particular, perspective distortions also occur. For example, objects that are away from the camera appear smaller in the image than objects that are closer.

To solve this problem, the first photogrammetrists assumed some *a priori* information: internal camera parameters (focal length, optical center, scale, distortions, and skew factor), and the camera position in relation to the scene. However, this is only valid when the accuracy of these values is high, since any deviation can generate large measurement errors [6].

Recent works on photogrammetry can be generally classified into two categories: the ones that exploit 3D information from multiple images and the ones that analyze geometric properties in a single image. The first category deals with 3D reconstruction of the scene, from multiple views, to estimate homography or to calibrate the camera [7–9]. Single-view based methods can only rely on the analysis of geometric properties [10–13] but, in this scenario, they often use previously positioned markers for calibration purposes [11,12]. Although these methods attain high accuracy, they cannot be used in forensic applications, where often a previously taken image is the only source of information.

Here we propose a single-view method that detects vanishing points – invariant geometric features that can aid photogrammetry. A *vanishing point* can be defined as an image point where the projection of a set of real-world parallel lines converges, assuming perspective

[☆] This paper has been recommended for acceptance by Yasutaka Furukawa.

[*] Corresponding author.

E-mail addresses: f.andalo@samsung.com, fernanda@andalo.net.br (F.A. Andaló), taubin@brown.edu (G. Taubin), siome@ic.unicamp.br (S. Goldenstein).

http://dx.doi.org/10.1016/j.cviu.2015.03.017
1077-3142/© 2015 Elsevier Inc. All rights reserved.

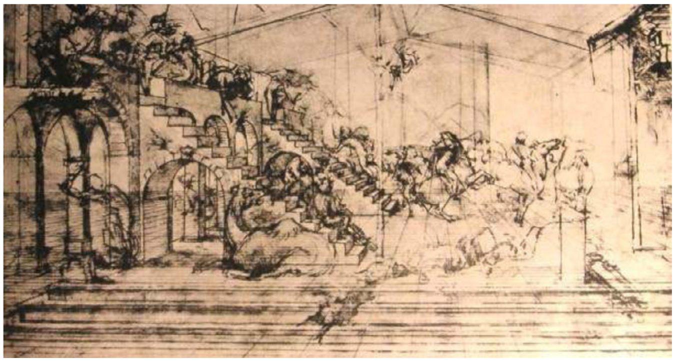

Fig. 1. Study for the painting *Adoration of Magi*, by Leonardo da Vinci showing the use of perspective.

projection. A *vanishing line* is a line that contains two vanishing points.

The credit for discovering linear perspective is given to renaissance artist Leon Battista Alberti and architect Filippo Brunelleschi [14]. Alberti's treatise, *De pictura* [15], published in 1435, encloses the first scientific study of perspective. By the 1470s, several artists were able to produce their works of art demonstrating a full understanding of the principles of linear perspective. Leonardo da Vinci, beginning in 1481, also studied and employed perspective in his earlier paintings (Fig. 1).

In this article, we use a geometric approach to effectively estimate the location of vanishing points – finite and infinite ones – in images of urban and indoor spaces, with the Manhattan-world assumption. We are assuming that the scene has a natural cartesian 3D coordinate system, which is plausible for indoor, outdoor city, and even some country scenes [16].

By representing segments, initially in the image plane, in the bidimensional projective space $\mathbb{RP}^2$, our method clusters them into groups of segments that converge to a unique vanishing point locality. The detected features are then used to identify the ground plane and the vertical direction of the scene. This information is finally inserted into a photogrammetry algorithm proposed by Criminisi [10], with the ultimate goal of measuring the height of objects and people in single images.

This article starts by presenting, in Section 2, a background on the measurement of heights in single images based on vanishing points. In Section 3, we present our vanishing point detector. In Section 4, we show how to estimate the scene vertical direction and also how to detect the ground plane vanishing line. The experiments and their results are provided in Section 5. Finally, Section 6 states the conclusions of this work.

2. Background

The projection of a world point $X \in \mathbb{R}^3$ into an image point $x \in \mathbb{R}^2$, considering perspective projection, is described by projection matrix $\mathsf{P} \in \mathbb{R}^{3\times4}$ as

$$\tilde{x} = \mathsf{P}\tilde{X} = \mathsf{K}[\mathsf{R}|\mathsf{T}]\tilde{X} = [p_1\ p_2\ p_3\ p_4]\tilde{X}, \qquad (1)$$

where $\tilde{X}$ and $\tilde{x}$ are the points X and x in homogeneous coordinates, respectively; K is the matrix representing the intrinsic parameters of the camera; the extrinsic parameters are R – rotation matrix – and T – translation vector from the world to the camera system; p_1, p_2, p_3 and p_4 are the columns of P; and the equality is up to scale.

In [17], the authors prove that p_1, p_2 and p_3 are the orthogonal vanishing points corresponding to the world coordinate system, and that p_4 is the image of the world origin. Here we denote these orthogonal vanishing points as v_x, v_y, v_z.

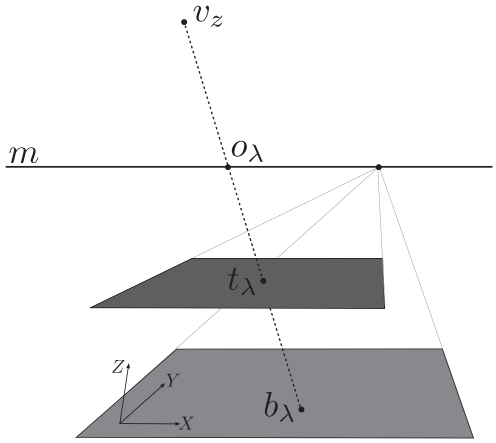

Fig. 2. Distance between the planes that contain points t_λ and b_λ. Point o_λ represents the intersection between vanishing line m and line that contains points t_λ and b_λ.

Considering that v_x and v_y are the two vanishing points on the vanishing line, we can say that p_4 must not be on the same line. If it does, then v_x, v_y, and p_4 are linearly dependent. Hence the fourth column can be set to $p_4 = m/\|m\| = \overline{m}$, where m is the vanishing line [6]. The final projection matrix is

$$\mathsf{P} = [v_x\ v_y\ \alpha v_z\ \overline{m}], \qquad (2)$$

where α is an unknown scalar referred as *metric factor*. If v_z and m are available, then the metric factor α is the only unknown value.

To measure heights in images, we must compute the distance between points in two different planes. Let v_z be the vanishing point that indicates the scene vertical direction, and m the ground vanishing line. Considering the projection matrix P (Eq. 2), Criminisi [18] proved that for an arbitrary object λ, with height Z_λ and delimited by image points t_λ and b_λ (top and bottom points), it holds that

$$\alpha Z_\lambda = -\frac{\|b_\lambda \times t_\lambda\|}{(\overline{m}\cdot b_\lambda)\,\|v_z \times t_\lambda\|}. \qquad (3)$$

Fig. 2 illustrates Eq. (3). Point o_λ represents the intersection between vanishing line m and the line that contains t_λ and b_λ. This intersection point helps to define a ratio of distances between planes and, using this value, one can compute $\frac{Z_\lambda}{Z_c}$, where Z_c is the camera's distance. However, it is simpler to compute Z_λ via a reference measurement in the image with a known length [6].

Thus, if we want to measure height Z_{obj} of an object *obj*, and Z_{ref} is a known distance on the same image, i.e., it is a reference distance between points t_{ref} and b_{ref}, then Eq. (3) allows the computation of the scale factor α and subsequently the distance Z_{obj} between t_{obj} and b_{obj}. The following steps can be used to compute height Z_{obj}:

1. Detect vanishing points in image I.
2. Identify vanishing point v_z associated with the scene vertical direction.
3. Estimate vanishing line m associated with the ground plane.
4. Compute scale factor α by applying Eq. (3) with object *ref*, i.e., $\lambda = ref$.
5. Compute Z_{obj} by applying Eq. (3) with object *obj*, i.e., $\lambda = obj$, and α.

The process of computing heights in images is illustrated in Fig. 3. According to Criminisi [10], efficient measurements in images can be done by accurately detecting the vanishing points, and estimating vertical vanishing point v_z and vanishing line m that represents the ground plane (steps 1, 2 and 3). Therefore, we can use the output of our detector to estimate v_z and m.

In the subsequent sections, we describe our method to detect vanishing points and how to estimate, from them, vertical vanishing point v_z, and ground vanishing line, m.

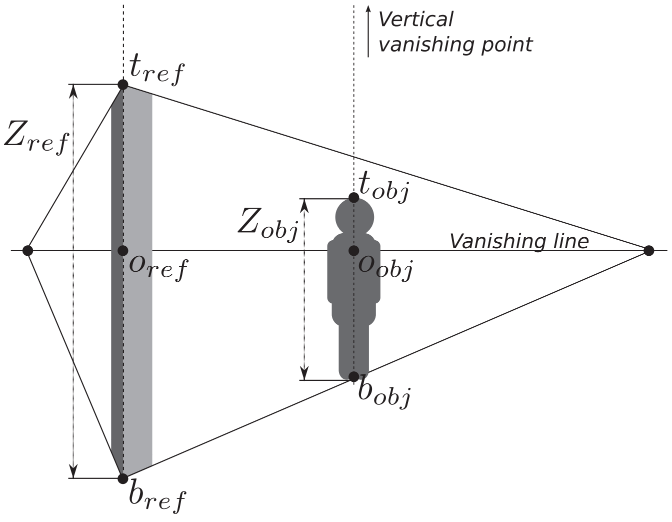

Fig. 3. Measuring the height of a real object depicted in an image. Height Z_{obj} can be estimated with the aid of known height Z_{ref} of a reference object.

3. Estimating vanishing points in single images

The problem of locating vanishing points in 2D perspective projection images has been studied since the 80s. This task is generally seen as the computation of line intersections but, due to quantization errors, lines that correspond to a single vanishing point intersect inside an area denominated *vanishing region*.

To confront the *vanishing region* problem, methods often divide the vanishing point detection in three main stages: detection of segments in the image, clustering of the these segments in groups that converge to a vanishing point, and vanishing point estimation for each cluster.

The first stage can be accomplished using an edge detector subsequently grouping the edges to form segments, e.g., Canny operator [19] and Hough transform [20]. Methods can perform the other stages in two ways: using accumulator spaces [21–25] or using the image plane directly [26,27].

Since Barnard's seminal work on the detection of vanishing points [21], methods have been employing different Hough transform techniques in quantized Gaussian spheres [22]. The problem in such methods are the artifacts often present in digital images, producing erroneous maxima on the quantized space [23].

The methods that use the image plane directly do not need accumulator techniques [26,27]. In this case, the accuracy of the vanishing point location is not limited by the space and the distances are preserved. However, it may be necessary to incorporate additional criteria to work with the infinite vanishing points.

Against these works, our method uses the bi-dimensional projective space $\mathbb{RP}^2$, or projective plane, transformed directly from the image space, to cluster the segments and detect all vanishing points, including the infinite ones, without additional criteria. Moreover, the space is not bounded, meaning it does no limit the location accuracy.

The proposed detector also has three stages:

1. Extracting line segments from the input image.
2. Clustering of the segments that converge to the same vanishing point (repeated until convergence):
   2.1. Selection of seeds.
   2.2. Grouping of segments based on the seeds and the intersection points.
3. Detection of a vanishing point for each final cluster.

Fig. 4 illustrates the three steps to estimate the vanishing points, numerated as shown above. Each one of the steps is described in a following subsection.

3.1. Line segment detection

We start by extracting line segments from the input image using a method proposed by Desolneaux et al. [28,29]. They deal with the extracting problem by exploring the Helmholtz Principle [29]. Besides line segments, their method also produce, for each segment, its number of false alarms, which will be used to compute the quality of this segment.

The Helmholtz principle declares that if, in an image, the expectation of a certain observed configuration is small, then this grouping is a Gestalt, i.e, it is meaningful [29].

Definition 1 (ε-meaningful configuration)**.** A configuration is ε-meaningful if it occurs in an image in a number less than ε.

Let I be a $N \times N$ image, and $A \in I$ a segment formed by a set of pixels $\{x_i\}$, $i = 1, \ldots, l$. Consider a random variable X_i where $X_i = 1$

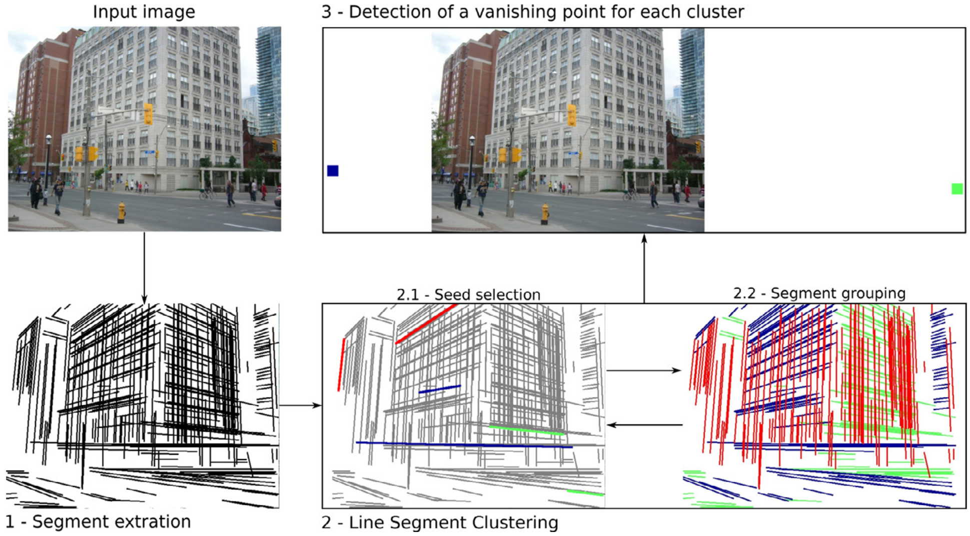
Input image
3 - Detection of a vanishing point for each cluster
2.1 - Seed selection
2.2 - Segment grouping
1 - Segment extration
2 - Line Segment Clustering

Fig. 4. Illustration of the three steps to detect vanishing points in an image. Each color (red, green and blue) represents a different cluster. In step 3, the vertical vanishing point is not shown, because it lies at infinity. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

if the direction of pixel[1] x_i is aligned to the direction of A, and $X_i = 0$ otherwise. Therefore, X_i has the following distribution with precision level p:

$$P[X_i = 1] = p \quad \text{and} \quad P[X_i = 0] = 1 - p. \tag{4}$$

The direction of x_i is aligned to the direction of A with precision p when

$$D_{ang}(dir(x_i), dir(A)) \leq \pi p, \tag{5}$$

where function $dir(\alpha)$ outputs the direction of a pixel or segment α, and D_{ang} outputs the smallest angle between the two directions.

The number of aligned pixels in segment A is represented by random variable $S_l = X_1 + X_2 + \ldots + X_l$. Because random variables X_i are independent, S_l has the following binomial distribution:

$$P[S_l = k] = \binom{l}{k} p^k (1-p)^{l-k}. \tag{6}$$

To consider segment A as a primitive for our method, we need to know if A is ε-meaningful among all segments in I.

Definition 2 (ε-meaningful segment)**.** A l-length segment A is ε-meaningful if it contains a minimum of $k(l)$ aligned pixels, where

$$k(l) = min\left\{k \in \mathbb{N}, P[S_l \geq k] \leq \frac{\epsilon}{N^4}\right\}. \tag{7}$$

The value N^4 is the number of oriented segments (defined by their initial and end pixels) in a $N \times N$ image.

Consider the ith segment, with length l_i, and the event e_i meaning “the ith segment is ε-meaningful”. Let χ_{e_i} denote the characteristic function of this event, so that

$$P[\chi_{e_i} = 1] = P[S_{l_i} \geq k(l_i)] = \sum_{k=k(l_i)}^{l_i} \binom{l_i}{k} p^k (1-p)^{l_i-k}. \tag{8}$$

Then the total of ε-meaningful segments is represented by variable $R = \chi_{e_1} + \chi_{e_2} + \cdots + \chi_{e_{N^4}}$, and expectation $E(R)$ relates to the number of false alarms.

Definition 3 (number of false alarms in a segment)**.** Considering a l_0-length segment A with at least k_0 aligned pixels, the number of false alarms of A is

$$F(k_0, l_0) = N^4 P[S_{l_0} \geq k_0] = N^4 \sum_{k=k_0}^{l_0} \binom{l_0}{k} p^k (1-p)^{l_0-k}. \tag{9}$$

If we take into account all ε-meaningful segments as primitives for our method, we will have several low quality segments. To avoid these equivocal segments, we finally take into account only some of the ε-meaningful segments, the maximal ones.

Definition 4 (Maximal segment)**.** A segment A is maximal if:

1. $\forall B, B \subset A \Rightarrow F(B) \geq F(A)$.
2. $\forall B, B \supset A \Rightarrow F(B) > F(A)$.

The two parameters of the method by Desolneux et al. [29], threshold ε and precision level p, are not critical. Values $\varepsilon = 1$ and $p = \frac{1}{16}$ work well for all images [24].

We group the detected maximal segments in a set $\mathcal{S} = \{s_1, \ldots, s_{|S|}\}$, and each maximal segment $s_i \in \mathcal{S}$ has a number of false alarms denoted by F_i.

Consider the end points (x_1, y_1) and (x_2, y_2), in the image plane $\mathbb{R}^2$, that specify an arbitrary maximal segment. If we represent these points in $\mathbb{RP}^2$, we get $(x_1, y_1, 1)$ and $(x_2, y_2, 1)$, respectively, using homogeneous coordinates. Here we are also interested in the lines that correspond to the maximal segments, $(x_1, y_1, 1) \times (x_2, y_2, 1)$. So, for each maximal segment s_i, we compute the correspondent line as shown above, resulting in l_i. The computed lines form a set $\mathcal{L} = \{l_1, \ldots, l_{|S|}\}$.

3.2. Segment clustering

The clustering process consists in assigning each segment $s_i \in \mathcal{S}$ to a cluster h. Each cluster h is a set formed by line segments assigned to it.

The clustering process has three main steps: determination of the first seeds, assignment of segments to clusters, and update of the seeds.

3.2.1. First seeds

The selection of the first seeds is done by considering a quality value for segments. The quality value q_i of segment s_i is defined as

$$q_i = 1 - \left(\frac{F_i - minF}{maxF - minF}\right), \tag{10}$$

where $maxF$ and $minF$ are the maximum and minimum numbers of false alarms among segments in $\mathcal{S}$, respectively. Note that the quality value of a segment is inversely proportional to its number of false alarms, i.e., the lower the number of false alarms, higher its quality.

The number of clusters, H, is chosen by the user. The $2H$ higher quality segments are randomly selected, in pairs, to be the seeds for the H clusters. The two segment seeds of cluster h are denoted by α_h and β_h.

The method converges in fewer steps when the first seeds are selected based on the quality of the segments. If random segments are chosen as the first seeds instead, the method yields the same results, except that it takes more steps.

3.2.2. Assignment step

The assignment step is responsible for selecting a cluster for each segment in $\mathcal{S}$. This is done by computing the distance between pseudo-centroids $c_h = l_{\alpha_h} \times l_{\beta_h}$, defined for each cluster $h = 1, \ldots, H$, and lines in $\mathcal{L}$.

The distance between an arbitrary point c and an arbitrary line l in $\mathbb{RP}^2$ can be computed as

$$D_{proj}(c, l) = \frac{|c \cdot l|}{\| c \| \| l \|}. \tag{11}$$

The formula for distance D_{proj} is obtained by considering the angle between the line that corresponds to c and the plane that corresponds to l in $\mathbb{RP}^3$. Distance D_{proj} only measures the symmetry between points and lines, because it is not a complete metric. Note that D_{proj} can be used arbitrarily also between two points in $\mathbb{RP}^2$.

The cluster selected for line segment s_i is the one with the closest pseudo-centroid, i.e,

$$cluster(s_i) = \underset{j \in [1,H]}{\text{argmin}}\, D_{proj}(c_j, l_i). \tag{12}$$

3.2.3. Update step

In this step, the method selects new seeds α_h and β_h for each cluster $h = 1, \ldots, H$.

New seed α_h is chosen as the segment in cluster h with closest orientation to the weighted circular mean [30] orientation $\overline{\theta_h}$ of the cluster, computed as

$$\overline{\theta_h} = arctan\left(\frac{S_h}{C_h}\right), \tag{13}$$

where S_h and C_h correspond to

$$S_h = \sum_{s_i \in \text{ cluster } h} q_i \sin(2\theta_i), \tag{14}$$

[1] The direction of a pixel is defined by Desolneux et al. [29] as the direction orthogonal to the direction of the gradient at the pixel.

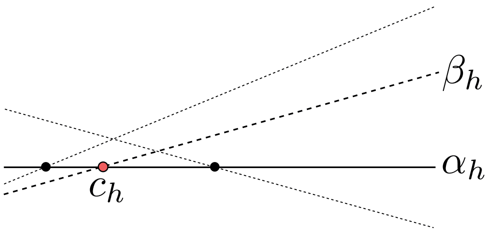

Fig. 5. Illustration of a cluster h with seeds α_h and β_h, and pseudo-centroid $c_h = l_{\alpha_h} \times l_{\beta_h}$. The dots represent all possible pseudo-centroids prior to the selection of β_h. The dashed segments are the candidates for seed β_h prior to its selection. Seed β_h is chosen based on the distance between the possible pseudo-centroids.

$$C_h = \sum_{s_i \in \text{ cluster } h} q_i \cos(2\theta_i). \tag{15}$$

In Eqs. (14) and (15), θ_i is the orientation of segment s_i.

Formally,

$$\alpha_h = \underset{s_i \in \text{ cluster } h}{\text{argmin}} \, D_{ang}(\overline{\theta_h}, \theta_i), \tag{16}$$

where D_{ang} gives the minimum angle between the orientations.

By selecting α_h, the possible new pseudo-centroids for cluster h are the points $l_{\alpha_h} \times l_i$, with $s_i \in$ cluster h. New seed β_h is selected as the one that permits new pseudo-centroid c_h to be the closest point to all other possible pseudo-centroids of the cluster. Formally,

$$\beta_h = \underset{s_i \in \text{ cluster } h}{\text{argmin}} \sum_{\substack{s_j \in \text{ cluster } h \\ i \neq j}} D_{proj}(l_{\alpha_h} \times l_j, l_{\alpha_h} \times l_i). \tag{17}$$

Fig. 5 illustrates a cluster h with its new seeds α_h and β_h. The dashed segments are the candidates prior to the selection of β_h.

The method converges when pseudo-centroids are the same in two consecutive steps.

3.3. Vanishing point detection

After the convergence, the possible vanishing points associated with cluster h are all intersection points in the cluster: $l_i \times l_j$, with s_i, $s_j \in$ cluster h. The vanishing point v_h is the closest intersection point to all segments in the cluster, i.e.,

$$v_h = \underset{\substack{l_i \times l_j \\ s_i, s_j \in \text{ cluster } h}}{\text{argmin}} \sum_{s_k \in \text{ cluster } h} D_{proj}(l_k, l_i \times l_j). \tag{18}$$

4. Estimating vertical vanishing point and ground vanishing line

According to [31], vertical vanishing points have two characteristics:

1. They usually are observed near the image y-axis, considering typical camera positions;
2. They are well separated from non-vertical vanishing points.

By considering these two characteristics as our premise, we can determine the vertical vanishing v_z, and the ground vanishing line m.

Regarding the first characteristic studied by [31], v_z should be related to the cluster that has the closest mean orientation (Eq. (13)) to the y-axis of the image. But this condition alone does not suffice.

Fig. 6 shows two possible clusters with their mean orientation represented by a dashed line. Note that the mean orientation in both clusters has no deviation from the y-axis. Nonetheless, a cluster associated with a vertical direction should have all segments with similar orientation, eliminating the possibility of choosing the first cluster.

Consequently, another characteristic that has to be analyzed is the distribution of the segments orientation in a cluster. For example, considering the segments orientation in Fig. 6, the first cluster has greater standard deviation than the second cluster. This observation leads us to the second necessary condition for associating a cluster with the vertical direction: the segments orientation in the cluster must have low circular standard deviation [30], defined as

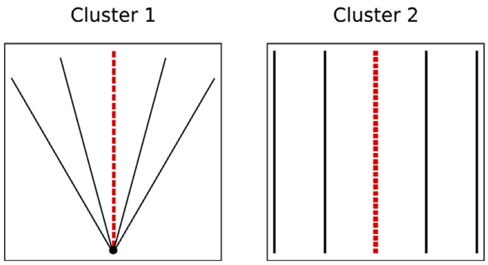

Fig. 6. Illustration of two possible clusters. The mean orientation of each cluster is represented by a dashed line.

$$\sigma_h = \frac{\sqrt{-2 ln(\overline{R_h})}}{2}, \tag{19}$$

where $\overline{R_h}$ corresponds to

$$\overline{R_h} = \frac{\sqrt{{S_h}^2 + {C_h}^2}}{\sum_{s_i \in \text{ cluster } h} q_i}. \tag{20}$$

Combining these two conditions, we can select cluster z associated with the vertical direction. Formally,

$$z = \underset{h \in [1,H]}{\text{argmin}} \, D_{ang}\left(\overline{\theta_h}, \frac{\pi}{2}\right) + \sigma_h, \tag{21}$$

where $\overline{\theta_h}$ is the circular mean orientation of cluster h (Eq. (13)) and σ_h is the circular standard deviation of cluster h (Eq. (19)).

Then vertical vanishing point v_z is the one associated with cluster z.

To estimate the ground vanishing line m, we recall the second characteristic studied by [31]: all vanishing points are well separated from v_z. So, there are three cases to be considered, depending on the number of clusters chosen by the user:

- Two: the ground vanishing line is the one that contains the non-vertical vanishing point, and with orientation given by the mean orientation of its cluster.
- Three: the ground vanishing line is the one that connects the two non-vertical vanishing points.
- Four or more: in this case, it is not clear through which two vanishing points the ground vanishing line passes. By graphically showing the possible vanishing lines, the user can select the one that corresponds to the ground plane.

5. Experiments

5.1. Vanishing point detector

We tested the effectiveness of the vanishing point detector in the York Urban database [32], an image database with a hundred images of man-made environments. Together with the images, the authors of the database also provided the intrinsic camera parameters, and the vanishing points computed with hand-detected segments and a Gaussian sphere method.

In Fig. 7, we provide some visual examples of the obtained results. The first column presents the input images together with the line segments in black, detected as shown in Section 3.1. The second column presents the results of the segment clustering algorithm associated with each input image of the first column, computed as shown in Section 3.2. Each cluster is represented in a different color

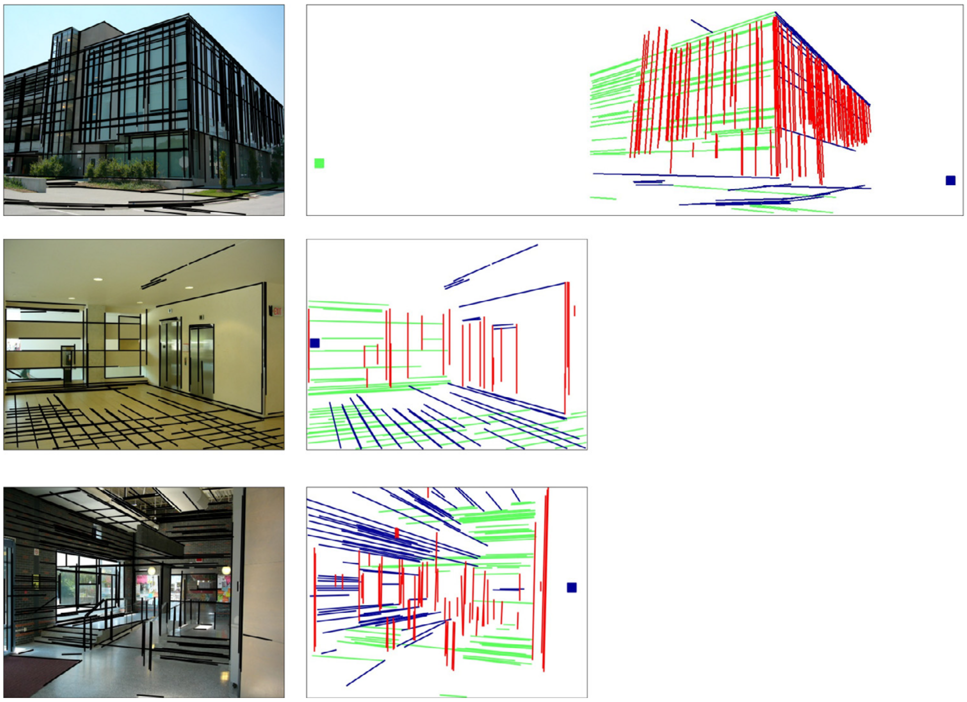

Fig. 7. The first column presents three input images together with the detected segments in black. The second column shows the correspondent final segment clusters and colored squares representing the detected finite vanishing points. Red parallel segments represent a cluster associated with an infinite vanishing point. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article).

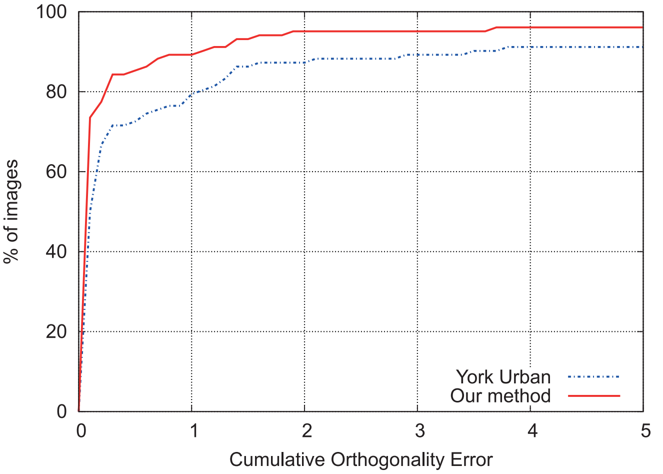

Fig. 8. Orthogonality error cumulative histogram computed in the York Urban Database, with the most orthogonal vanishing point triplet in each image. Point (x, y) represents y% of images with orthogonality error lower than x.

(red, green, and blue) and the colored squares represent the finite vanishing points, detected as shown in Section 3.3.

Note that we are considering the Manhattan-world assumption, i.e., the images must depict scenes with a cartesian 3D coordinate system. Scenes with few parallel lines or lines not aligned with the coordinate system axis cannot be considered here.

Besides the visual inspection of the results, we conducted two experiments. First, we computed the *orthogonality error* to quantify the deviation of the most orthogonal vanishing points[2] from the actual orthogonality.

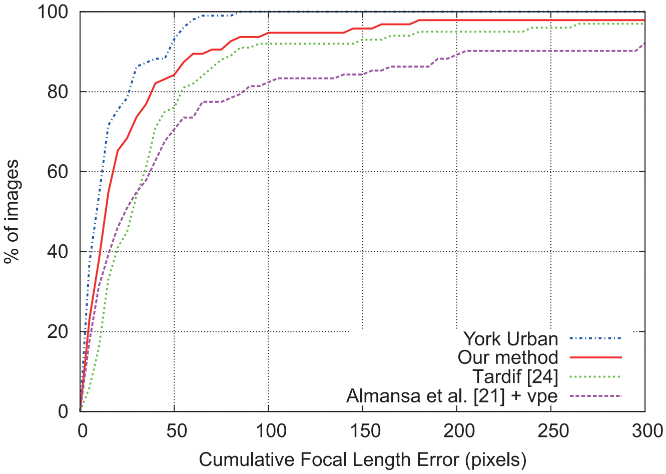

Fig. 9. Focal length error cumulative histogram computed in the York Urban Database, with the most orthogonal vanishing point triplet in each image. Point (x, y) represents y% of images with focal length error lower than x.

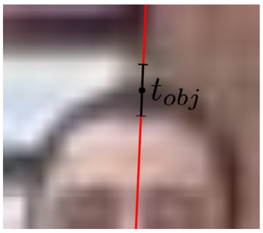

Fig. 10. Range of 1% · |$\overline{t_{obj}b_{obj}}$| around t_{obj}.

The second experiment measured the *focal length error* using the camera intrinsic parameters, provided by the database. We compared the focal length computed with the most orthogonal vanishing points to the real expected one.

[2] The *most orthogonal vanishing points* are the ones with the lowest orthogonality error (Eq. (25)).

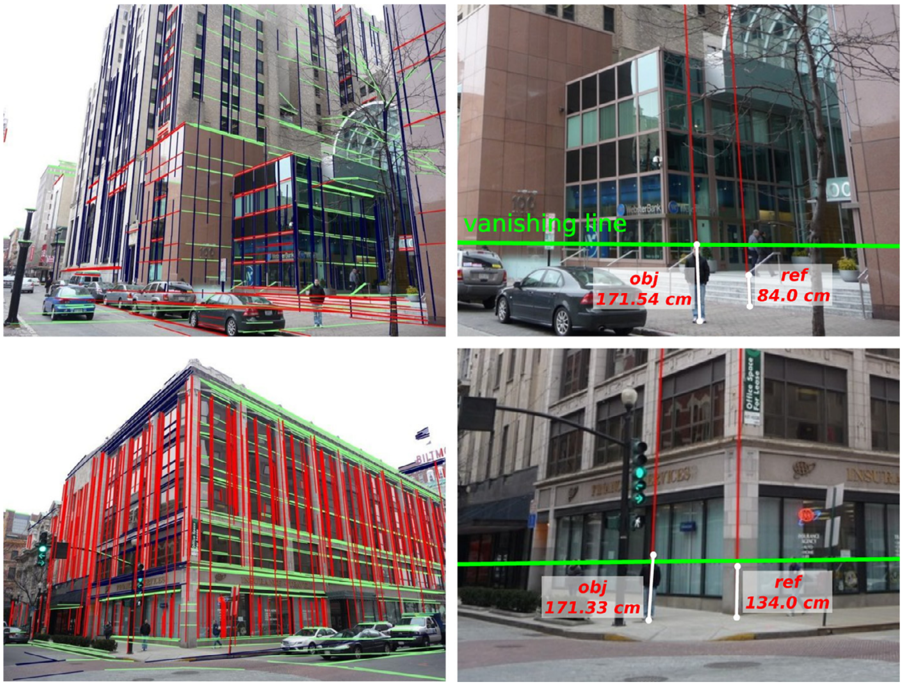

Fig. 11. Measuring the height of a person. The first column contains the input image with the segment clustering result. The second column presents a crop with the computed height.

5.1.1. *Orthogonality error*

The authors of the database provided the intrinsic parameters of the camera: focal length f, pixel dimension (m_x, m_y), principal point (p_x, p_y), and skew factor ς. With this information, we can construct the camera intrinsic matrix:

$$\mathsf{K} = \begin{bmatrix} f/m_x & \varsigma & p_x \\ 0 & f/m_y & p_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{22}$$

Given that we have matrix K, it is trivial to obtain the Image of the Absolute Conic ω:

$$\omega = \mathsf{K}^{-T}\mathsf{K}^{-1}. \tag{23}$$

Considering two arbitrary orthogonal vanishing points v_p and v_q, it is true that

$$v_p \omega v_q = 0. \tag{24}$$

Therefore, to find the most orthogonal vanishing points among all detected vanishing points $v_i, i = 1, \dots, H$, we select the triplet (v_p, v_q, v_r) that minimizes the orthogonality error

$$e_{p,q,r} = (v_p \omega v_q)^2 + (v_q \omega v_r)^2 + (v_r \omega v_p)^2. \tag{25}$$

For each image on the York Urban database, we selected the most orthogonal vanishing point triplet. We then constructed a histogram of the cumulative orthogonality error for these triplets, shown in Fig. 8. We compared our method with the one provided in the database [32], named here as *York Urban* method, where the segments are hand detected and the computation is done by considering a cumulative space represented in a Gaussian Sphere.

5.1.2. *Focal length error*

For this experiment, consider that the focal length f is unknown, but instead we have the most orthogonal vanishing point triplet for each image. With this information, we can estimate the unknown f and compare it with the real value. In order to do this, it is necessary to retrieve matrix K by decomposing matrix ω.

For each image of the database, we selected the most orthogonal vanishing point triplet, and estimated a focal length from them. We then constructed a histogram of the cumulative focal length, shown in Fig. 9. We compared our obtained focal length with the one obtained by *York Urban* method and two other detectors:

- Almansa et al. [24] considers the Helmholtz principle to detect segments and estimate vanishing regions. We selected the center of these regions to locate the vanishing points. This extension is named here as *Almansa et al. + vpe*.
- Tardif [27] considers a Canny edge detector and a flood fill algorithm to extract segments. Vanishing points are detected by a J-Linkage algorithm.

Note that *York Urban* method [32] provides in general a better estimation of the focal length. This is justified by the fact that, in this method, the segments are hand detected, eliminating a major source of errors.

5.2. *Measuring heights*

With the knowledge of the vanishing points locality, we can select which one is vanishing direction v_z and also determine the ground line m, as detailed in Section 4. This information can be inputed into Criminisi's photogrammetry framework [6], as described in 2, to make height measurements in single images.

Because we are dealing only with a single view of the scenes, it is necessary to obtain the height of reference objects in order to compute absolute values. This cannot be done with images of scenes we don't have access to make reference measurements. Therefore we constructed a database of twenty images (750 × 563 pixels) with reference objects in the real world. The vanishing points detected on these images generate focal length errors of less than 25 pixels.

In order to measure the real height of an object *obj* depicted in the image, the user must inform its top and bottom points, t_{obj} and b_{obj}, and the top and bottom points of a reference *ref*, t_{ref} and b_{ref}, with known height Z_{ref}.

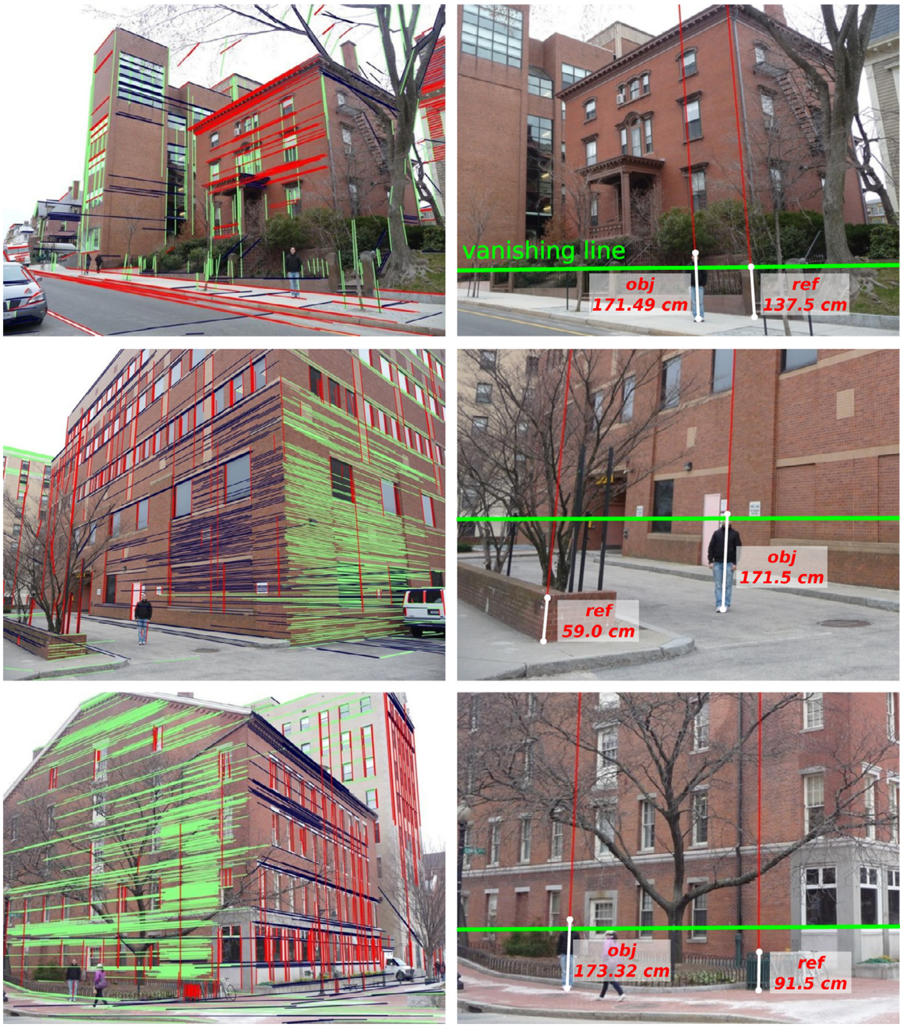

Fig. 12. Measuring the height of a person. The first column contains the input image with the segment clustering result. The second column presents a crop the computed height.

In an error-free case, the points t_α and b_α, $\alpha = obj, ref$, have to be aligned with v_z [6]. Therefore, to compute height Z_{obj}, we replace the location of the selected points b_{obj} and b_{ref} with their perpendicular projection onto lines $\overleftrightarrow{v_z t_{obj}}$ and $\overleftrightarrow{v_z t_{ref}}$, respectively, generating the scenario depicted in Fig. 3.

The errors associated with the choice of points t_{obj}, b_{obj}, t_{ref} and b_{ref} were considered by computing the height using all possible combinations of points inside a range of $1\% \cdot |\overline{t_\alpha b_\alpha}|$ around each point, for $\alpha = obj, ref$. For example, Fig. 10 shows the considered range around t_{obj} (top of object *obj*). Each point inside the depicted range was considered once as t_{obj}. The final height is the average between every obtained height.

The height of a person was measured comparatively to a known reference height in twenty images taken in urban environments. The ground truth height is 171.5 cm and the mean observed error across all images was ±0.58 cm. Some of the results are shown in Figs. 11, 12, and 13. In each one of these figures, the first column contains the input image with the segment clustering result, where each color represents a cluster associated with a vanishing point. The second column shows a zoomed crop with the vanishing line in green, the reference object height, and the computed height.

6. Conclusion

In this article, we described a method to detect vanishing points in an image and showed how to apply it to make efficient height measurements. The proposed method works with uncalibrated cameras, and can detect all vanishing points. Since it is performed in a bidimensional projective space, the detected points have accurate locations with no loss of information in transformations between spaces.

However, the method is only effective when applied to images of man-made environments, in which it is possible to extract straight segments corresponding to different 3D orientations.

Several applications can benefit from the proposed method. In forensic investigations, for example, images from CCTV cameras – Closed-circuit television – can be used to estimate the height of suspects. CCTV cameras are widely employed in streets of major cities like New York and London.

The results show visually and experimentally the effectiveness of the method and its application to a photogrammetry framework, allowing the estimation of heights in images.

The presented method is useful in computing the height of rigid objects or people standing in a straight position. However, the height of animated objects tend to change while they move. This study will be included in a future work.

Acknowledgments

This work was primarily supported by CNPq (grants 201238/2010-1 and 308882/2013-0), with additional support by FAPERJ and CAPES (grant E-26/103.665/2012), NSF (grants IIS-0808718, CCF-0915661, and IIP-1330139), and FAPESP (grant 2012/50468-6).

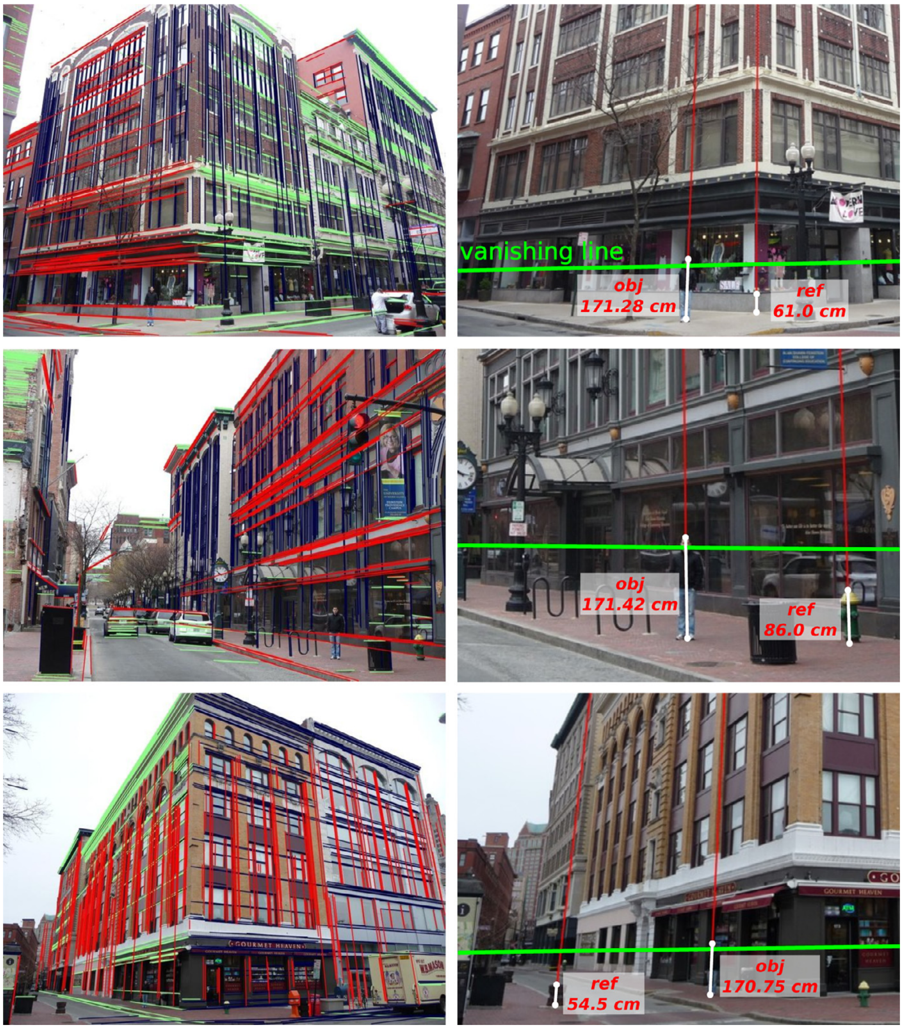

Fig. 13. Measuring the height of a person. The first column contains the input image with the segment clustering result. The second column presents a crop the computed height.

References

[1] A. Saxena, M. Sun, A. Ng, Learning 3-D scene structure from a single still image, in: Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV), 2007, pp. 1–8.

[2] A. Saxena, M. Sun, A. Ng, Make3D: learning 3D scene structure from a single still image, IEEE Trans. Pattern Anal. Mach. Intell. 31 (5) (2009) 824–840.

[3] J. Michels, A. Saxena, A. Ng, High speed obstacle avoidance using monocular vision and reinforcement learning, in: Proceedings of the 22nd International Conference on Machine Learning (ICML), 2005, pp. 593–600.

[4] J. Nascimento, J. Marques, Performance evaluation of object detection algorithms for video surveillance, IEEE Trans. Multimedia 8 (4) (2006) 761–774.

[5] S. Bramble, D. Compton, L. Klasén, Forensic image analysis, in: Proceedings of the 13th INTERPOL Forensic Science Symposium, 2001.

[6] A. Criminisi, Accurate Visual Metrology from Single and Multiple Uncalibrated Images, Springer-Verlag, New York, USA, 2001.

[7] C. Madden, M. Piccardi, Height measurement as a session-based biometric for people matching across disjoint camera views, in: Proceedings of the Conference on Image and Vision Computing New Zealand (IVCNZ), 2005, pp. 282–286.

[8] Z. Chen, N. Pears, B. Liang, A method of visual metrology from uncalibrated images, Pattern Recognit. Lett. 27 (13) (2006) 1447–1456.

[9] S. Khan, M. Shah, A multiview approach to tracking people in crowded scenes using a planar homography constraint, in: Proceedings of the 9th European Conference on Computer Vision (ECCV), (2006) 133–146.

[10] A. Criminisi, Single-view metrology: algorithms and applications, in: Proceedings of the 24th DAGM Symposium on Pattern Recognition, 2002, pp. 224–239.

[11] S.-W. Park, T.-E. Kim, J.-S. Choi, Robust estimation of heights of moving people using a single camera, in: Proceedings of the International Conference on IT Convergence and Security (ICITCS), 2011, pp. 389–405.

[12] K.-Z. Lee, A simple calibration approach to single view height estimation, in: Proceedings of the Ninth Conference on Computer and Robot Vision (CRV), 2012, pp. 161–166.

[13] N. Nguyen, R. Hartley, Height measurement for humans in motion using a camera: a comparison of different methods, in: Proceedings of the International Conference on Digital Image Computing Techniques and Applications (DICTA), 2012, pp. 1–8.

[14] A. Cole, Eyewitness Art: Perspective, Dorling Kindersley, London, UK, 1992.

[15] L. Alberti, De Pictura, Reproduced by Laterza (1980), 1435.

[16] J. Coughlan, A. Yuille, Manhattan world: orientation and outlier detection by bayesian inference, Neural Comput. 15 (5) (2003) 1063–1088.

[17] G. Wang, H.-T. Tsui, Q. Wu, What can we learn about the scene structure from three orthogonal vanishing points in images, Pattern Recognit. Lett. 30 (3) (2009) 192–202.

[18] A. Criminisi, I. Reid, A. Zisserman, Single view metrology, Int. J. Comput. Vis. 40 (2) (2000) 123–148.

[19] J. Canny, A computational approach to edge detection, IEEE Transact. Pattern Anal. Mach. Intell. 8 (6) (1986) 679–698.

[20] R. Duda, P. Hart, Use of the Hough transformation to detect lines and curves in pictures, Commun. ACM 15 (1) (1972) 11–15.

[21] S. Barnard, Interpreting perspective images, Artif. Intell. 21 (4) (1983) 435–462.

[22] T. Tuytelaars, L.V. Gool, M. Proesmans, T. Moons, The cascaded Hough transform as an aid in aerial image interpretation, in: Proceedings of the Sixth International Conference on Computer Vision (ICCV), 1998, pp. 67–72.

[23] J. Shufelt, Performance evaluation and analysis of vanishing point detection techniques, IEEE Trans. Pattern Anal. Mach. Intell. 21 (3) (1999) 282–288.

[24] A. Almansa, A. Desolneux, S. Vamech, Vanishing point detection without any a priori information, IEEE Trans. Pattern Anal. Mach. Intell. 25 (4) (2003) 502–507.

[25] C. Rother, A new approach to vanishing point detection in architectural environments, Image Vis. Comput. 20 (9) (2002) 647–655.

[26] G. McLean, D. Kotturi, Vanishing point detection by line clustering, IEEE Trans. Pattern Anal. Mach. Intell. 17 (11) (1995) 1090–1095.

[27] J.-P. Tardif, Non-iterative approach for fast and accurate vanishing point detection, in: Proceedings of the IEEE 12th International Conference on Computer Vision (ICCV), 2009, pp. 1250–1257.

[28] A. Desolneux, L. Moisan, J.-M. Morel, Meaningful alignments, Int. J. Comput. Vis. 40 (1) (2000) 7–23.

[29] A. Desolneux, L. Moisan, J.-M. Morel, Maximal meaningful events and applications to image analysis, Ann. Stat. (2003) 1822–1851.

[30] K. Mardia, P. Jupp, Directional Statistics, Wiley Series in Probability and Statistics, John Wiley and Sons, 1999.

[31] A. Gallagher, Using vanishing points to correct camera rotation in images, in: Proceedings of the 2nd Canadian Conference on Computer and Robot Vision, 2005, pp. 460–467.

[32] P. Denis, J. Elder, F. Estrada, Efficient edge-based methods for estimating manhattan frames in urban imagery, in: Proceedings of the 10th European Conference on Computer Vision (ECCV), 2008, pp. 197–210.