Detecting vanishing points by segment clustering on the projective plane for single-view photogrammetry

Fernanda A. Andaló [1], Gabriel Taubin [2], Siome Goldenstein [1]

[1] *Institute of Computing, University of Campinas (Unicamp)*
CEP 13083-852, Campinas, SP, Brazil
{feandalo,siome}@ic.unicamp.br

[2] *Division of Engineering, Brown University*
Providence, RI 02912 USA
taubin@brown.edu

***Abstract*—In this paper, we describe how an effective vanishing point detector can be applied to photogrammetry when only a single view of an architectural environment is available. Our method performs automatic segment clustering in projective space – a direct transformation from the image space – instead of the traditional bounded accumulator space. Experiments on real images show the effectiveness of the proposed detector in finding all vanishing points, as well as its application in a photogrammetry algorithm, by recovering the vertical direction of the scene and the vanishing line for the ground plane.**

I. INTRODUCTION

Images provide strong cues about the three-dimensional (3D) geometry of the scenes depicted. These cues can be exploited in a variety of computer vision applications, such as forensics, image-based rendering, and surveillance. The advantages of using images for such applications, instead of other specific devices (laser scanners, ultrasonic devices, etc.), is that images can be applied to measure the distance between the camera and the objects, as well as two other points in space, areas of surfaces and angles [1]. Furthermore, since images are the only input data, the depth cues can be analysed even if the scene is no longer accessible, using archived images or old footage.

The problem of taking world measurements using images – **photogrammetry** – has one of its main application in forensic science, where image measurements are being used to provide useful information about the course of events and the size of items and persons in a crime scene [2]. The two most commonly reported forensic activities that involve photogrammetry are human height analysis and accident analysis of motor vehicles [2]. In a reliable system, the estimated height of people can be used as a corroborative evidence, for example.

However, when dealing with images, the essential problem is how to recover the third dimension or take measures of the scene, knowing that the required information was not captured and stored in the process of projecting the 3D space to the planar image.

WIFS'2010, December 12-15, 2010, Seattle, WA, USA.

The early methods rely on knowing the internal parameters of the camera and its position with respect to the viewed scene. Even when this information is not available, camera calibration can be performed if the depth of some key points is available [3] or using self-calibration techniques [4]. However, the necessary data is not always available or involves a much bigger and ill-posed problem, the 3D reconstruction.

In this scenario, previous works show that vanishing lines and points are useful features [1], [5]. A **vanishing point** is defined as the convergence point of a set of lines in the image plane that is produced by the projection of parallel lines in real space, under the assumption of perspective projection, e.g. with a pin-hole camera. Because they are invariant features, the analysis of vanishing points provides strong cues to make inferences about the 3D structures of a scene, such as depth and object dimension.

In this work, we propose a novel and automated method for vanishing point detection based on a geometrical approach, in which all finite and infinite vanishing points are estimated in a single image of a man-made environment. We use the vanishing points to detect the vertical direction of the scene and the vanishing line of the ground plane. For estimating heights of objects in the scene, we insert our method in the algorithm proposed by [1].

This paper is organized as follows. Section II presents the problem of measuring heights in an image using vanishing points and lines. Section III describes our method for vanishing point estimation. Section IV describes how to estimate the vertical direction in an image and also shows how to detect the vanishing line associated with the ground plane. Section V discusses experimental results and Section VI concludes the paper.

II. BACKGROUND

In [1], Criminisi addresses the problem of measuring distances of points from planes in an uncalibrated framework. The aim is to compute the height of an object relative to a reference. The author assumes that the vertical direction and the vanishing line for the ground plane have been computed by some method.

If v is the vanishing point for the vertical direction, m is the vanishing line of the ground plane, t_r and b_r are the top and base points of the reference, respectively, and t_x and b_x are the top and base points of the object to be measured, then [1]

$$\alpha Z_i = -\frac{\|b_i \times t_i\|}{(m \cdot b_i)\,\|v \times t_i\|}, \forall i = r, x, \quad (1)$$

where Z_x is the height of the object to be measured, Z_r is the reference height and α is a scalar quantity herein referred to as *metric factor*. If α is known, then a metric value for the height Z is obtained. If the height Z is known, then α can be computed.

The method outline [1] is presented in Algorithm 1.

Algorithm 1 Measuring heights in an image.

1) Estimate the vanishing point v for the vertical direction;
2) Estimate the vanishing line m of the reference plane;
3) Select top and base points of the reference segment (points t_r and b_r);
4) Compute the metric factor α by applying Equation 1 with t_r and b_r;

loop
- (a) Select top and base of the object to be measured (points t_x and b_x);
- (b) Compute the metric factor α by applying Equation 1 with t_x and b_x.

end loop

According to Criminisi [1], the key to the success of this calculation is an accurate estimation of the vertical vanishing point v and the vanishing line m of the reference plane.

The following sections describe our technique for the automatic computation of vanishing points and how to obtain, from them, the vanishing point for the vertical direction and the vanishing line for the ground plane.

III. Estimating the Vanishing Points

Our method for vanishing point detection is presented in Algorithm 2.

Algorithm 2 Vanishing point estimation

1) Extraction of line segments on the image plane;
2) Clustering of line segments to groups of lines converging to the same vanishing point;

loop
- (a) determination of seeds based on a computed quality value for each segment;
- (b) grouping of the line segments based on the distance among lines and intersection points in projective space;

end loop

3) Vanishing point estimation for the extracted segment clusters.

The outline of the proposed method is illustrated in Figure 1. The following subsections describe each stage.

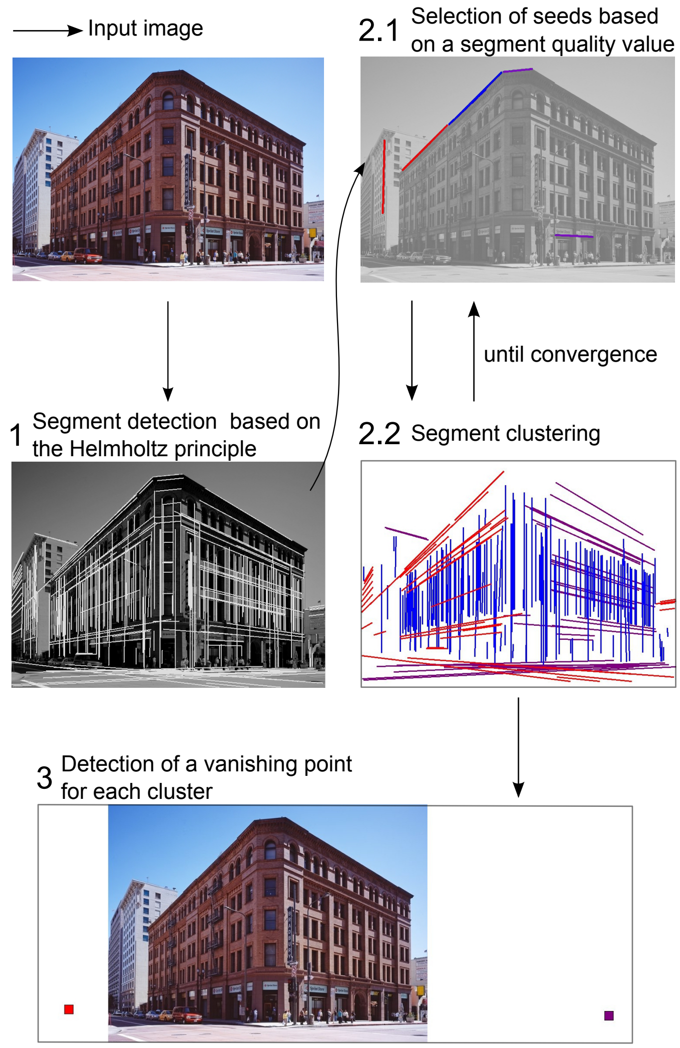


Fig. 1. Stages of the proposed vanishing point detector.

A. Line segment detection

Our vanishing point detector considers line segments as primitives. To detect these primitives, we use a method based on the Helmholtz Principle [6]. This principle states that if the expectation of an observed configuration in an image is very small, then the grouping of the objects in this configuration is valid.

Aside from detecting line segments, this method provides an important value – the number of false alarms for a segment – that is useful in the next steps to compute a quality value for each segment. The number of false alarms of a segment represents an upper-bound of the expectation, in an image, of the number of segments of probability less than the one of the considered segment [6].

In the following, we denote the set of the detected line segments by $\mathcal{S} = \{s_1, \ldots, s_{|S|}\}$, the number of false alarms for the segment $s_i \in \mathcal{S}$ by NF_i, and the intersection point between the lines corresponding to segments s_i and s_j by $w_{(l_i,l_j)}$.

B. Line segment clustering

The goal of the line segment clustering is to assign a cluster for each one of the line segments in $\mathcal{S}$. To achieve this goal, the clustering process is subdivided in three steps: selection of the first seeds, assignment step, and update step.

The number M of clusters to be detected is informed by the user. However, this parameter does not need to be tunned, since the method works independently of the chosen number, detecting the corresponding number of vanishing points.

1) First seeds: The first seeds are chosen based on the quality value of the segments. The quality value q_i of segment $s_i \in \mathcal{S}$ is defined as

$$q_i = \left| \frac{NF_i - (max(NF_j) + min(NF_j))}{max(NF_j)} \right|, j \in \mathcal{S}. \quad (2)$$

We select the $2M$ lines with highest corresponding segment quality and randomly choose pairs of these lines as seeds for the M clusters. We denote d_{1_i} and d_{2_i} the two seeds of cluster i.

2) Assignment step: Each segment $s \in \mathcal{S}$ has to be assigned to a cluster. The assignment step is based on the distance between the lines corresponding to the segments and the pseudo-centroids $c_i = w_{(d_{1_i}, d_{2_i})}, i = 1, \ldots, M$.

The distance between a line l and a pseudo-centroid c is defined as

$$D_{LP}(l, c) = \frac{|l \cdot c|}{\| l \| \| c \|}. \quad (3)$$

D_{LP} is symmetric, but it is not a full metric. However, it is a robust way to measure the amount of symmetry between lines and points.

The segment s_i is assigned to the cluster with the closest pseudo-centroid. Formally,

$$cluster(s_i) = j \mid c_j = \underset{c_k, k \in [1,M]}{\text{argmin}} D_{LP}(l_i, c_k), \quad (4)$$

where l_i is the line corresponding to the segment s_i.

3) Update step: The update step comprises the selection of the new seeds d_{1_i} and d_{2_i}, for each cluster $i = 1, \ldots, M$.

The choice of the new seeds is such that they minimize the error to the lines that would pass through the real corresponding vanishing point. The seed d_{1_i} minimizes the distance to the mean line of cluster i and d_{2_i} is chosen so that the new pseudo-centroid c_i minimizes the distance to some key intersection points.

The seed d_{1_i} is the line corresponding to a segment assigned to cluster i, that minimizes the angular distance to the weighted circular mean orientation of the cluster. The angular distance is the smallest angle between two orientations; and the weighted circular mean orientation $\overline{\theta}_i$ for cluster i, considering the quality values as the weights, is computed as [7]

$$\overline{\theta_i} = arctan\left(\frac{S}{C}\right), \quad (5)$$

where S and C correspond to

$$S = \sum_{j=1}^{n} q_j * \sin(2\theta_j), \quad (6)$$

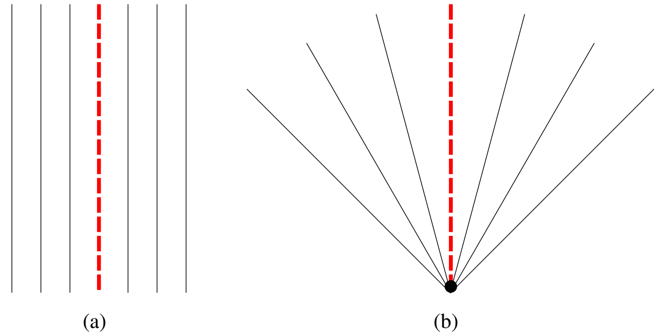


Fig. 2. Example of possible clusters. The dashed line represents the mean line of the clusters.

$$C = \sum_{j=1}^{n} q_j * \cos(2\theta_j). \quad (7)$$

In Equations 6 and 7, n is the number of segments assigned to cluster i, and q_j and θ_j are, respectively, the quality value and orientation of the j-th segment assigned to the cluster i.

The seed d_{2_i} is the line corresponding to a segment assigned to cluster i and which intersection point with seed d_{1_i}, $w_{(d_{1_i}, d_{2_i})}$, minimizes the sum of the distances to all other intersection points $w_{(d_{1_i}, l_j)}$, where s_j is assigned to cluster i.

C. Vanishing point detection

The vanishing point v_i for the cluster i is the intersection point that is the closest one to all lines in the cluster:

$$v_i = \underset{p}{\text{argmin}} \sum_{s_j \in \text{ Cluster } i} D_{LP}(l_j, p). \quad (8)$$

IV. Estimating the vanishing point for the vertical direction and the vanishing line for the ground plane

As shown in [8], the vanishing points for the vertical direction are constrained to fall on the image y-axis, under normal camera positions. Furthermore, they tend to be well separated from the vanishing points for the horizontal directions [8].

Considering this constraint, the first condition to detect the cluster associated to the vertical direction is for the cluster to have a little to no offset between the mean orientation of its segments (Equation 5) and the image y-axis. But this condition is not sufficient. Observe, for example, the clusters illustrated in Figure 2.

Both mean orientations, represented by the dashed lines in Figure 2, have the same orientation and have no offset to the image y-axis. However, if they occur in the same image, we want to choose the cluster in Figure 2(a) to be the one associated with the vertical direction.

To accomplish this, we have to consider the distribution of the lines' orientation in the same cluster. In Figure 2(a), the standard deviation of the lines' orientation is lower than in Figure 2(b). Considering this observation, the second condition to detect the cluster associated to the vertical direction is for the cluster to have low circular standard deviation, defined as [7]:

$$\sigma = \frac{\sqrt{-2ln\left(\overline{R}\right)}}{2}, \tag{9}$$

where $\bar{R}$ corresponds to

$$\overline{R} = \frac{\sqrt{S^2 + C^2}}{\sum_{j=1}^{n} q_j}. \tag{10}$$

The two conditions together are sufficient to select the cluster associated to the vertical direction. Formally, we want to select the cluster i that minimizes

$$D_{ang}\left(\overline{\theta_i}, \frac{\pi}{2}\right) + \sigma_i, \tag{11}$$

where D_{ang} is the smallest angle between the orientations, $\overline{\theta_i}$ is the circular mean of the lines' orientation in cluster i (Equation 5) and σ_i is the circular standard deviation in cluster i (Equation 9).

The vanishing point v associated with the cluster that minimizes Equation 11 is the vertical vanishing point.

Following the same idea, to select the vanishing line for the ground plane, first we consider all lines passing through all detected vanishing points. We select the line that minimizes the angular distance to the image x-axis. If only two vanishing points are available, the vanishing line for the ground plane is the line that passes through the non-vertical vanishing point and which orientation is the same as the circular mean orientation of the respective cluster.

V. Experiments

The method was implemented in C++ and we conducted the experiments in the York Urban database [9]. It consists of 102 indoor and outdoor images of man-made environments. The York Urban database also provides the camera intrinsic parameters as well as vanishing points computed with hand-detected segments.

Figure 3 illustrates a few obtained results for the detection of vanishing points. The first column shows input images with the segments and the second column shows the line clustering results and the location of the finite vanishing points. For experimental purposes, the number M of vanishing points was set for each image.

To show the effectiveness of the method, we executed two experiments in the York Urban database:

- Computation of the orthogonality error of the detected vanishing points. This experiment shows how much the more orthogonal vanishing points[1] deviate from the real orthogonality on the image plane.
- Computation of the focal length error based on the detected vanishing points and the intrinsic parameters matrix for the camera. This experiment compares the focal length computed from the more orthogonal vanishing points with the expected focal length.

[1]The term *more orthogonal vanishing points* refers here to the detected vanishing points that minimizes the orthogonality error (Equation 13).

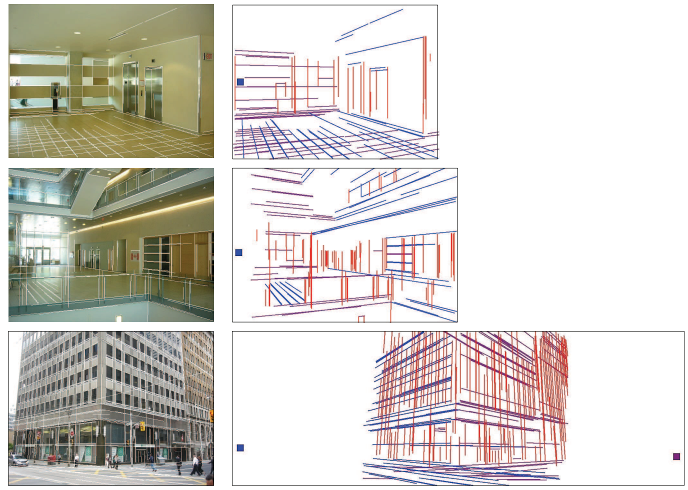


Fig. 3. The first column shows the input image and all detected segments. The second column shows the line clustering result and the detected finite vanishing points. Parallel lines with the same color are associated with a vanishing point at infinity; the other lines are associated with finite vanishing points.

Given the camera pixel dimension, the principal point and the skew factor, provided in the York Urban Database, we can construct the intrinsic parameters matrix K of the camera. Given K, the Image of the Absolute Conic (IAC) ω is given by

$$\omega = \mathsf{K}^{-T}\mathsf{K}^{-1}. \tag{12}$$

Let $v_i, i = 1, \ldots, M$, be the estimated vanishing points. The more orthogonal triplet minimizes the orthogonality error

$$e_{p,q,r} = (v_p \omega v_q)^2 + (v_q \omega v_r)^2 + (v_r \omega v_p)^2. \tag{13}$$

A perfectly orthogonal triplet leads to a zero orthogonality error. To select the more orthogonal vanishing points detected by our method, we choose the triplet that minimizes Equation 13.

Figure 4 shows the cumulative orthogonality error histogram for our method and for the method provided in York Urban database, called here as Ground Truth. In the database, the segments are detected by hand and the vanishing points are computed by a Gaussian Sphere method.

Considering the triplet of vanishing points that minimizes the orthogonality error, we can compute the focal length and compare it to the estimated focal length provided in the York Urban database. To compute the focal length, we recovered the camera intrinsic matrix K by decomposing the IAC matrix with unknown focal length.

We compared our method with three other vanishing point detectors by comparing the focal length error generated by each one. The method Almansa 2003 [10] detects line and vanishing regions based on the Helmholtz principle. For comparison purposes, we have selected the center of the detected regions as the vanishing points location. We called this extension as Almansa 2003 + vpe. The method Tardif 2009 [11] uses the Canny edge detector and flood fill for line detection and an algorithm called J-Linkage for vanishing point estimation.

Figure 5 shows the cumulative focal length error histogram for our method and for the other methods, using the York Urban database. We can see that for the critical part of the histogram, where the focal length error is low, our method provides significant superior results.

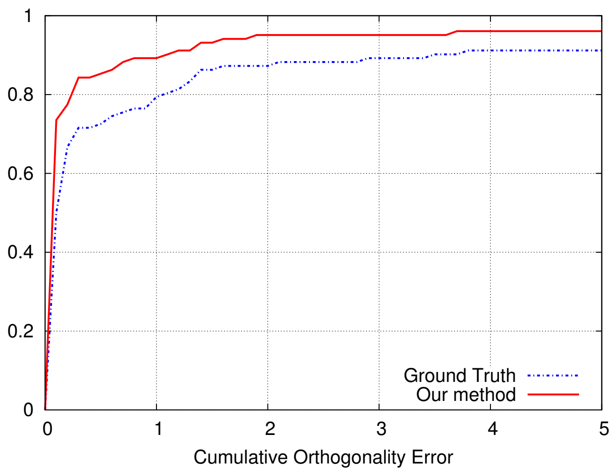


Fig. 4. Cumulative histogram for the estimated errors on York Urban Database. A point (x, y) represents the fraction y of images in the database that have error $e < x$.

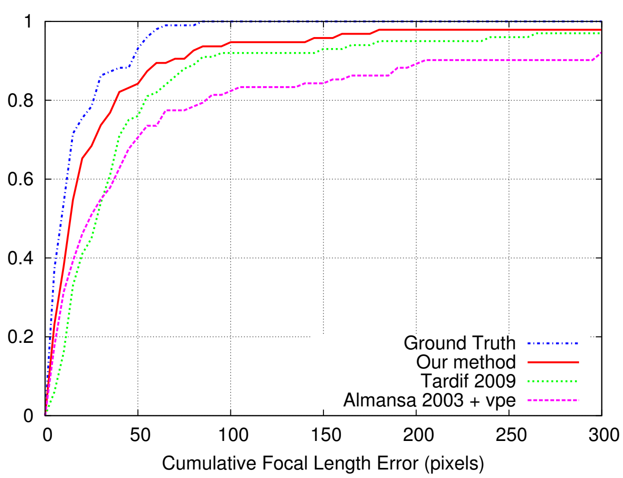
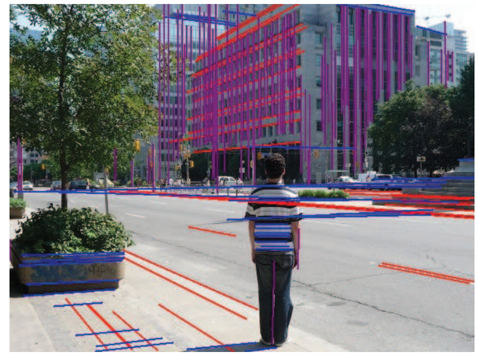
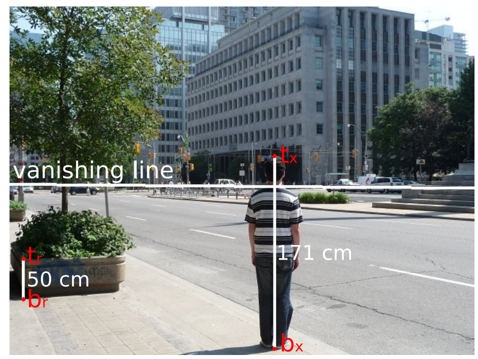


Fig. 5. Cumulative histogram for the focal length errors on York Urban Database. A point (x, y) represents the fraction y of images in the database that have focal length error less than x.

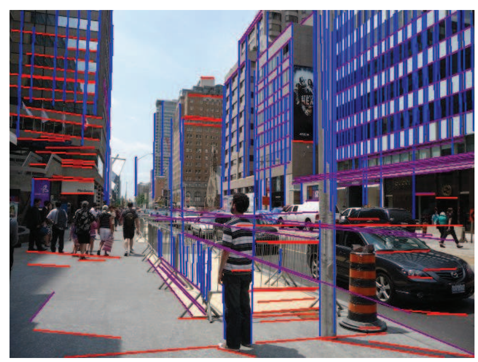
The two experiments shows the effectiveness of the proposed method for vanishing point estimation. To measure heights in an image, we need to find the vertical direction and the vanishing line for the ground plane. This is done as described in Section IV. If only one view of the scene is available, we also need the height of a reference object to compute an absolute value.



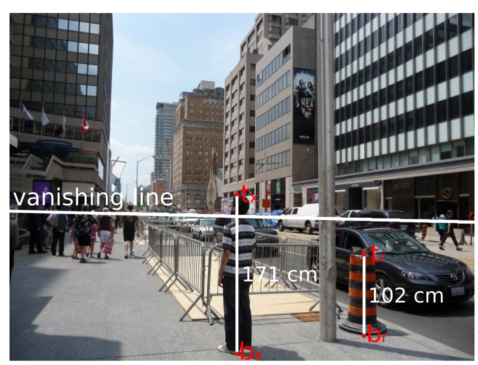
(a)



(b)



(c)



(d)

Fig. 6. Measuring heights in single images. (a) and (c) input images and the line clustering result. (b) and (d) the vanishing line, the reference object and the object measured.

Figure 6 shows some examples of heights measured in single images. The aim is to compute the height of a person relative to the height of a reference object. The mean error in estimating the height in Figures 6(b) and 6(d) is 0.18 cm.

VI. Conclusion

In this work, we presented a vanishing point detector and its application to photogrammetry. It is a useful tool for forensic science, since it allows the measurement of heights in images of man-made environments.

Our method detects finite and infinite vanishing points, without any prior camera calibration, and since it is performed on an unbounded space – the projective plane – all vanishing points can be accurately estimated with no loss of geometrical information from the original image.

The method is effective when applied to images of architectural environments, where there is a predominance of straight lines corresponding to different 3D orientations.

It is important to notice that when only one view of the scene is available, the scale factor α cannot be recovered uniquely. So to compute heights of objects in the scene, such as people, we need an additional reference in the scene. If an reference height is not provided, the estimated height will not be absolute.

However, if multiple views are available, it is possible to compute α uniquely by observing one arbitrary point across the multiple views [12].

The experiments show the effectiveness of the vanishing points estimation and its application to photogrammetry, where the height of a person was estimated within a small error range.

Acknowledgment

This work is primarily supported by CNPq grant 141248/2009-2, with additional funding from NSF (grants CNS-0729126, CNS-0721703, IIS-0808718, and CCF-0915661), CNPq (grants 309254/2007-8, 551007/2007-9, 551623/2009-8 and 200717/2010-3), FAPESP, and CAPES.

References

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

[2] S. Bramble, D. Compton, and L. Klasn, "Forensic image analysis," in *13th INTERPOL Forensic Science Symposium*, 2001.

[3] O. Faugeras, *Three-dimensional computer vision: a geometric viewpoint*. Cambridge, MA, USA: MIT Press, 1993.

[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[5] B. Caprile and V. Torre, "Using vanishing points for camera calibration," *International Journal of Computer Vision*, vol. 4, no. 2, pp. 127–140, 1990.

[6] A. Desolneux, L. Moisan, and J.-M. Morel, "Edge detection by helmholtz principle," *Journal of Mathematical Imaging and Vision*, vol. 14, no. 3, pp. 271–284, 2001.

[7] K. Mardia and P. Jupp, *Directional statistics*, ser. Wiley Series in Probability and Statistics. John Wiley and Sons, Nov. 1999.

[8] 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.

[9] P. Denis, J. Elder, and F. Estrada, "Efficient edge-based methods for estimating manhattan frames in urban imagery," in *European Conference on Computer Vision*, 2008, pp. 197–210.

[10] A. Almansa, A. Desolneux, and S. Vamech, "Vanishing point detection without any a priori information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 4, pp. 502–507, 2003.

[11] J.-P. Tardif, "Non-iterative approach for fast and accurate vanishing point detection," *International Conference on Computer Vision*, pp. 1250–1257, 2009.

[12] H.-T. T. G. Wang and Q. Wu, "What can we learn about the scene structure from three orthogonal vanishing points in images," *Pattern Recognition Letters*, vol. 30, no. 3, pp. 192–202, 2009.