Vanishing Point Detection by Segment Clustering on the Projective Space

Fernanda A. Andaló[1], Gabriel Taubin[2], and 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. The analysis of vanishing points on digital images provides strong cues for inferring the 3D structure of the depicted scene and can be exploited in a variety of computer vision applications. In this paper, we propose a method for estimating vanishing points in images of architectural environments that can be used for camera calibration and pose estimation, important tasks in large-scale 3D reconstruction. Our method performs automatic segment clustering in projective space – a direct transformation from the image space – instead of the traditional bounded accumulator space. Since it works in projective space, it handles finite and infinite vanishing points, without any special condition or threshold tuning. Experiments on real images show the effectiveness of the proposed method. We identify three orthogonal vanishing points and compute the estimation error based on their relation with the Image of the Absolute Conic (IAC) and based on the computation of the camera focal length.

Key words: Vanishing point detection, Segment clustering, 3D reconstruction

1 Introduction

Large-scale three-dimensional (3D) reconstruction is a challenging task in computer vision and has received considerable attention recently due to the usefulness of the recovered 3D model for a variety of applications, such as city planning, cartography, architectural design, fly-through simulations, and forensic science.

The key task in large-scale 3D reconstruction is to recover high-quality and detailed 3D scene models from two or more unordered and wide-baseline images [1], which may be taken from widely separated viewpoints.

Due to the complexity of the scenes, conventional modeling techniques are very time-consuming and recreating detailed geometry become very laborious. In order to overcome these difficulties, some works have been inclined towards image-based modeling techniques [2], using images to drive the 3D reconstruction [3, 4]. However, in many image-based modeling techniques, the scenes are reconstructed using camera calibrated images or, when this is not the case, it is nontrivial to establish correspondences between different images.

Recent works have focused on using scene constraints to optimize the reconstruction, especially the geometric ones found in almost man-made environments, such as parallelism and orthogonality [5, 6]. Vanishing points are an important geometric constraint widely found in images of man-made objects, that can be used to calibrate the camera [6, 7] and to find the relative pose.

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. The analysis of such vanishing points provides strong cues to make inferences about the 3D structures of a scene, such as depth and object dimension, because they are invariant features.

Each vanishing point corresponds to an orientation in the 3D scene and when the camera geometry is known, these orientations can be recovered. Even without this information, vanishing points can be used to group segments on the image with the same 3D orientation.

Because of its important role in 3D reconstruction, the detection of the vanishing points in a scene has to be effective, especially when no human intervention is required. This work proposes a novel and automated method based on a geometrical approach, in which all finite and infinite vanishing points are estimated in an image of a man-made environment. It does not rely on calibration parameters or thresholds. Our solution is based on the clustering of line segments that are detected in the image, representing points and segments on the projective space. The advantages of our method with respect to previous methods are:

- **Translational and rotational invariance.** Preserves the original distances among points and lines, because it does not operate on a bounded space, such as the Gaussian sphere or the Hough space.
- **Unlimited location accuracy.** It does not use accumulator-space techniques.
- **Unified handling of vanishing points.** It uses projective geometry.
- **Estimates all vanishing points.** It includes orthogonal and non-orthogonal vanishing points.
- **No need for camera calibration.** All camera parameters are unknown.

Figure 1 shows the stages of this method including detection of image line segments, determination of seeds based on a computed quality value for each segment, grouping of the line segments based on the distance among the intersection points of the corresponding lines in projective space (and not relying on any orthogonality assumption). The two later stages iteratively run until convergence to find the vanishing points. Experimental results on real images show that the proposed method can effectively detect all finite and infinite vanishing points. We also compute the estimation error based on the relation of the detected vanishing points with the Image of the Absolute Conic (IAC).

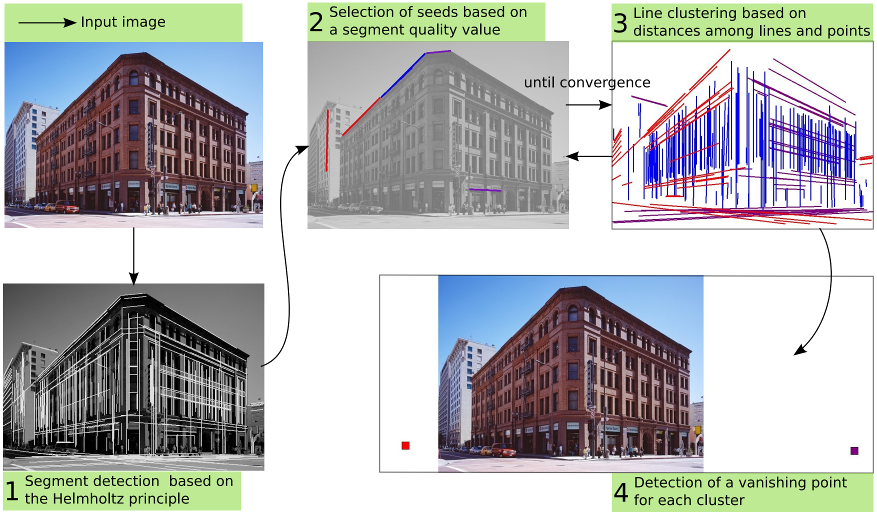

Fig. 1. Flowchart of the proposed vanishing point detection method.

2 Related Work

In recent years, a lot of effort has been devoted to finding vanishing points out of 2D perspective projections and practical methods consider this task as a line intersection detection problem. Due to quantization and error on the detection of segments, the segments corresponding to a specific vanishing point do not intersect at a single point, but they intersect inside an area called **vanishing region**. To address this problem, methods often break the task into three steps:

1. Extraction of line segments on the image plane.
2. Clustering of line segments to groups of lines converging to the same vanishing point.
3. Vanishing point estimation for the extracted line clusters.

The first step is often implemented using a zero-crossing technique to extract edges that are subsequently grouped to form straight segments, e.g. Canny operator [8] followed by Hough transform [9]. For the second and third steps, the methods can be roughly divided in two categories: the ones that use accumulator spaces [10–14] and the ones that perform the clustering directly on the image plane [15, 16].

In the seminal technique due to Barnard [10], a Gaussian sphere is used to represent the orientation space. In this approach, lines from image space are projected onto a sphere that is tangent to the image plane at the center of the image. The projection of lines are circles and the sphere is discretized to compose an accumulator space for these circles; maxima on the sphere represents orientations shared by several line segments, and can be hypothesized as vanishing points for the image.

Since Bernard's work, however, methods for vanishing points detection in digital images have been based on some variation of the Hough transform in a conveniently quantized Gaussian sphere, for mapping the parameters of the line segments into a bounded Hough space [11]. One problem that arises in such methods is categorized as noise: artifacts of digital image geometry and textural effects can combine to produce spurious maxima on the Gaussian Sphere [12]. To address this problem, Almansa et al. [13] use the Helmholtz principle to partition the image plane into Meaningful vanishing regions and use Minimum Description Length to reject spurious vanishing points. Unfortunately, bounded spaces are not translational and rotational invariant (do not preserve distances between lines and points).

In [14], the image plane itself is chosen as the accumulator space and although it is not straight-forward to treat in the same way finite and infinite vanishing points, this method addresses the problem. But since determining local maxima is difficult and expensive, this method imposes an orthogonal criterion – the vanishing points must correspond to the three mutual orthogonal directions of the scene.

The second category of methods use the image plane itself for the clustering process, without the use of any accumulator technique [15, 16]. Generally, the clustering process depends on computations, such as distance among points and lines, that are performed on image space. Such methods have the advantage of not limiting the location accuracy and of preserving distances. It can be difficult, however, to handle infinite vanishing points without additional criterion.

Against this background, this work provides a method for vanishing point estimation that uses the projective space – a direct transformation from the image space – to perform the clustering of segments and to handle all vanishing points without special criterion, despite the fact that the space is unbounded.

3 Large-scale 3D reconstruction from vanishing points

Under perspective projection, a 3D point $x \in \mathbb{R}^3$ is projected to an image point $m \in \mathbb{R}^2$ via a projection matrix $\mathsf{P} \in \mathbb{R}^{3\times 4}$ as

$$\tilde{m} = \mathsf{P}\tilde{x} = \mathsf{K}[\mathsf{R}|\mathsf{T}]\tilde{x} \ , \tag{1}$$

where $\tilde{m}$ and $\tilde{x}$ are the homogeneous form of points m and x, respectively; R is the rotation matrix, T is the translation vector from the world system to the camera system, and K is the camera intrinsic matrix. Matrix K is defined as

$$\mathsf{K} = \begin{bmatrix} f/m_x & \varsigma & p_x \\ 0 & f/m_y & p_y \\ 0 & 0 & 1 \end{bmatrix} \ , \tag{2}$$

where f is the focal length, (m_x, m_y) is the camera pixel dimension, (p_x, p_y) is the camera principal point, and ς refers to the skew factor. For a three-parameter camera, we have to assume square pixels, i.e., $\varsigma = 0$ and $m_x = m_y$; known principal point and known aspect ratio $\gamma = m_x/m_y$.

3.1 Recovering camera matrices

In [7], Wang et al. show that camera parameters can be learned from three orthogonal vanishing points, assuming some restrictions. More specifically, they prove that the camera projection matrix can be uniquely determined from three orthogonal vanishing points, assuming a three-parameter camera. Furthermore, they prove that the global consistent projection matrices can be recovered if an arbitrary reference point in space is observed across multiple views.

To calibrate the camera, we have to recover the image of the absolute conic [7]. The absolute conic $\mathsf{C}_\infty = \mathsf{I}_3$ is a conic on the plane at infinity composed of purely imaginary points. Under perspective projection, the image of the absolute conic (IAC) is defined as

$$\omega = \mathsf{K}^{-T}\mathsf{K}^{-1} \; . \tag{3}$$

It is know that two orthogonal vanishing points v and v^T satisfies

$$v^T \omega v = 0 \; . \tag{4}$$

Consequently, a set of three orthogonal vanishing points can provide three linearly independent constraints to ω and a three-parameter camera can be calibrated.

The projection matrix P is defined as $\mathsf{P} = [s_x\tilde{v}_x, s_y\tilde{v}_y, s_z\tilde{v}_z, s_o\tilde{v}_o]$, where $\tilde{v}_x, \tilde{v}_y, \tilde{v}_z$ are the homogeneous form of the three orthogonal vanishing points, $\tilde{v}_o$ is the world origin; and s_x, s_y, s_z, s_o are unknown scalars.

Given a set of three orthogonal vanishing points v_x, v_y and v_z, the scalars s_x, s_y and s_z can be uniquely determined if the camera is assumed to have three-parameter and if $s_o\tilde{v}_o$ is known [7].

For large-scale 3D reconstruction, when we have multiple views of the scene, the scalars corresponding to the projection matrices of these views must be consistent. Given an arbitrary point in space which can be observed across multiple views, the consistent scalars associated with the translation terms of the projection matrices of these views can be uniquely determined [7].

In [2], the authors solved the inconsistency among the multiple views using digital compass information associated with each view, instead of using a key point in multiple views.

3.2 3D reconstruction

A possible outline for large-scale 3D reconstruction based on vanishing point detection from uncalibrated images is presented in [7]:

(i) For each view:
   (a) Compute three orthogonal vanishing points;
   (b) Compute three scalars s_x, s_y and s_z for a specified world origin.
(ii) Determine the consistent scalars of the projection matrices:
   (a) Select a reference point in the first image and determine its correspondence in other views;

(b) Compute the scalars pair-wisely;
(c) Compute the consistence projection matrices for each view weighted by the scalars;

(iii) Detect and match key features across the images;
(iv) Recover the 3D structure of these key features via triangulations;
(v) Perform global optimization.

4 Effective vanishing point detection

As presented in Figure 1, our method has four main steps. The first step, detection of line segments, is discussed in Section 4.1. The second and the third, that together characterize the clustering process are presented in Section 4.2. The last step is presented in Section 4.3.

4.1 Line segment detection

The line segments are used as primitives of our vanishing point estimator and to detected them, we use a method based on the Helmholtz Principle [17]. The usefulness of this specific method is beyond the task of segment detection. It also 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 the segment.

The Helmholtz principle states that if the expectation in the image of an observed configuration is very small, then the grouping of the objects is a Gestalt:

Definition 1 (ϵ-meaningful event). *An event is ϵ-meaningful, if the expectation of the number of occurrences of this event in an image is less than ϵ.*

Let f be an image of size $N \times N$ and $x_1, \ldots, x_l$ a set of l independents pixels of a line segment A. At each x_i, a random variable X_i equals 1 if the angle between the image gradient $\bigtriangledown f(x_i)$ and the normal to the segment A is less than $p\pi$, where p is the precision level (usually $p \approx 1/16$); and $X_i = 0$ otherwise, assuming a uniform distribution of the gradient orientations.

The random variable that represents the number of points having the same direction as the line is $S_l = X_1 + X_2 + \ldots + X_l$, which has a binomial distribution of parameters p and l.

The method considers a segment of length l_0 to be meaningful when its expected number of occurrences in the image is low (lower than ϵ).

Definition 2 (ϵ-meaningful segment). *A segment of length l is ϵ-meaningful in a $N \times N$ image if it contains at least $k(l, \epsilon)$ points having their direction aligned with that of the segment, where $k(l, \epsilon)$ is given by*

$$k(l, \epsilon) = min \left\{ k \in \mathbb{N}, P[S_l \geq k] \leq \frac{\epsilon}{N^4} \right\} . \quad (5)$$

Let l_i be the length of the i-th segment and e_i the event "the i-th segment is ϵ-meaningful". Let χ_{e_i} denote the characteristic function of the event e_i, so that

$$P[\chi_{e_i} = 1] = P[S_{l_i} \geq k(l_i, \epsilon)] = \sum_{k=k(l_i,\epsilon)}^{l_i} \binom{l_i}{k} p^k (1-p)^{l_i - k} \ . \tag{6}$$

Then the variable representing the number of ϵ-meaningful segments is $R = \chi_{e_1}, \chi_{e_2}, \ldots, \chi_{e_{N^4}}$, and its expectation E(R) gives the expected number of false alarms.

Definition 3 (number of false alarms). *Given a segment of length l_0 in a $N \times N$ image containing k_0 points aligned with the direction of the segment, the number of false alarms for this segment is*

$$NF(k_0, l_0) = N^4 P[S_{l_0} \geq k_0] \ . \tag{7}$$

To avoid spurious responses, the method considers a subset of the ϵ-meaningful segments that are maximal.

The described method depends on two parameters. The meaningful threshold ϵ is necessary and it is not critical. The standard setting $\epsilon = 1$ works well for all images. However, the precision parameters p is not really necessary. Even though $p = 1/16$ works well for most images, a finer p might do better in edges with highly precise gradient orientations [13].

4.2 Line segment clustering

The input of our method is a set $\mathcal{S} = \{s_1, \ldots, s_{|\mathcal{S}|}\}$ of detected image segments on Euclidean space $\mathbb{R}^2$, and the number M of clusters. The output is a classification $cluster(s_i)$ for each segment, representing its assignment to a cluster.

For the segment clustering process, the method constructs three sets: set $\mathcal{L}$ of lines on the real projective space $\mathbb{RP}^2$, corresponding to each segment in $\mathcal{S}$; set $\mathcal{W}$ of the intersection points for each pair of lines in $\mathcal{L}$, where $w_{(a,b)} \in \mathcal{W}$ corresponds to the intersection point between lines a and b; and set $\mathcal{Q}$ of quality values for each segment. For a segment s_i with the number of false alarms NF_i, the quality value q_i is

$$q_i = \left| \frac{NF_i - (max(NF_j) + min(NF_j))}{max(NF_j)} \right| , s_j \in \mathcal{S} \ . \tag{8}$$

The goal of the line segment clustering is to assign a cluster for each one of the segments in $\mathcal{S}$. We denote C_j the j-th cluster. In addition, the following properties corresponds to C_j: a **seed** (d_{1_j}, d_{2_j}), where d_{1_j} and d_{2_j} are lines in $\mathcal{L}$; and a **pseudo-centroid** $t_j = w_{(d_{1_j}, d_{2_j})} \in \mathcal{W}$ that is the intersection point between lines d_{1_j} and d_{2_j}.

The clustering process is divided in three steps: selection of the first seeds, assignment step, and update step. The algorithm aims to minimize an objective function

$$\sum_{j=1}^{M} \sum_{s_i \in C_j} D_{LP}(l_i, t_j) \ , \tag{9}$$

where the function D_{LP} gives the distance between a line and a point. This function is defined in $\mathbb{RP}^2$ and is given by

$$D_{LP}(k, h) = \frac{|k \cdot h|}{\parallel k \parallel \parallel h \parallel} \ . \tag{10}$$

An important property is that the distance between two points in $\mathbb{RP}^n$ is the angle between the corresponding lines in $\mathbb{RP}^{n+1}$ [18]. Using this information, the function D_{LP} gives a value that is relative to the angle between the corresponding line and plane in $\mathbb{RP}^3$. This distance is symmetric, but it is not a full metric – it does not satisfy triangle inequality. However, it is a robust way to measure the amount of symmetry between lines and points.

First Seeds For a number M of vanishing points, we select as seeds $2M$ lines based on the quality of the corresponding segments. More precisely, we select the $2M$ lines with highest corresponding segment quality and distribute these pairs of lines randomly across the clusters.

Assignment step At this step, the algorithm assigns each segment $s \in \mathcal{S}$ to the cluster C that has the closest pseudo-centroid t. The "closest" concept is determined by the distance function D_{LP}. Formally,

$$cluster(s_i) = C \mid t = \underset{t_j, j \in [1,M]}{\operatorname{argmin}} \ D_{LP}(l_i, t_j) \ . \tag{11}$$

Update step When all segments in $\mathcal{S}$ have been assigned to a cluster, we need to recalculate the positions of the pseudo-centroids. To accomplish this task, the method selects a new seed for each cluster. For the cluster C_j, the new seed is (d_{1_j}, d_{2_j}).

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

The line d_{1_j} is the one that the corresponding segment is assigned to the cluster C_j and that minimizes the angular distance to the weighted mean orientation of the cluster. The angular distance is the smallest angle between two orientations. The weighted mean orientation $\overline{\theta}_j$ for the cluster C_j, considering the quality values as the weight, is computed as [19]

$$\overline{\theta_j} = arctan \left(\frac{\sum_{s_i \in C_j} q_i * \sin(2\theta_i)}{\sum_{s_i \in C_j} q_i * \cos(2\theta_i)} \right) \ , \tag{12}$$

where θ_i is the orientation of the line corresponding to the i-th segment assigned to the cluster.

The line d_{2_j} is the one that the corresponding segment is assigned to the cluster C_j and which intersection point with the line d_{1_j}, $w_{(d_{1_j},d_{2_j})}$, minimizes the sum of the distances to all other intersection points $w_{(d_{1_j},i)}$, where s_i is assigned to C_j.

The process of determining d_{1_j} and d_{2_j} on cluster C_j is illustrated on Figure 2. First, the mean orientation of segments assigned to C_j (corresponding to non-dotted lines) is computed. Line d_{1_j} is the one with closest orientation to the mean. Line d_{2_j} is the one that, together with d_{1_j}, forms the intersection point closest to all other intersection points of d_{1_j} (only considering the ones formed by lines corresponding to segments assigned to cluster C_j).

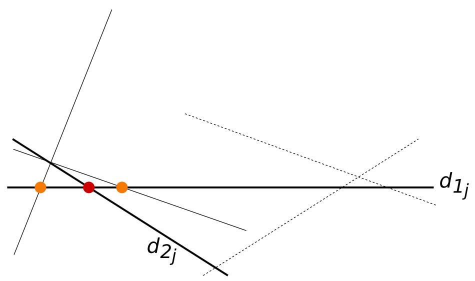

Fig. 2. Determination of the seed (d_{1_j}, d_{2_j}) of the cluster C_j. Non-dotted lines correspond to segments assigned to cluster C_j.

The relative distance between two intersection points in $\mathbb{RP}^2$ is given by the angle between the corresponding lines in $\mathbb{RP}^3$. Figure 3 illustrates the distance on the spherical model of $\mathbb{RP}^2$ between a finite point a and a infinite points b.

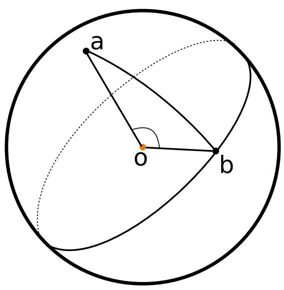

Fig. 3. Relative distance between a finite point a and a infinite points b, on a spherical model of $\mathbb{RP}^2$.

The new pseudo-centroid t_j of cluster C_j is $w_{(d_{1_j},d_{2_j})}$ – the intersection point between lines d_{1_j} and d_{2_j}.

The two last steps – assignment and update – must be computed until convergence is achieved, i.e. until the pseudo-centroids no longer change.

4.3 Vanishing point estimation

The final step is the estimation of the vanishing points location. For each detected cluster C_j, the method selects, as the corresponding vanishing point, the intersection point v_j that is the closest one to all lines in the cluster, according to D_{LP}:

$$v_j = \operatorname*{argmin}_p \sum_{s_i \in C_j} D_{LP}(l_i, p) \ . \tag{13}$$

5 Experiments and Results

We implemented our algorithm in C++ and we conducted the experiments using the York Urban Database [20]. It consists of 102 indoor or outdoor images of man-made environments. Figure 4 illustrates a few obtained results. The first column shows input images with the detected segments. The second row shows the line clustering results and the location of the finite vanishing points. For experimental purposes, the parameter M was set for each image. For real purposes, the parameter M does not need to be tuned. If $M = 3$, the method will actually detect three vanishing points.

Our first experiment to test the effectiveness of the estimated vanishing points was to compute the error associated with their relation with the Image of Absolute Conic (IAC).

The York Urban Database provides the camera intrinsic parameters and therefore it is simple to construct the camera intrinsic matrix K (Equation 2). Given K, the IAC ω is given by Equation 3.

Let $v_i, i = 1, \ldots, M$ be the estimated vanishing points. Our goal is to find the triplet that is more orthogonal, i.e, we want to minimize

$$e_{i,j,w} = (v_i \omega v_j)^2 + (v_j \omega v_w)^2 + (v_w \omega v_i)^2 \ . \tag{14}$$

For all vanishing points estimated by our method, we select the triplet that minimizes Equation 14, the orthogonality error, as the three orthogonal vanishing points. A triplet (v_i, v_j, v_w) of orthogonal vanishing points leads to a zero $e_{i,j,w}$ (Equation 14), the error associated with our estimation procedure. Figure 5 shows the cumulative orthogonality error histogram for our method and for the method provided in York Urban database (hand detected segments and vanishing points detection on the Gaussian sphere), called here as Ground Truth.

The second experiment was to estimate the focal length with the vanishing point triplet that minimized Equation 14 and to compute the focal length error compared to the real focal length provided in the York Urban database.

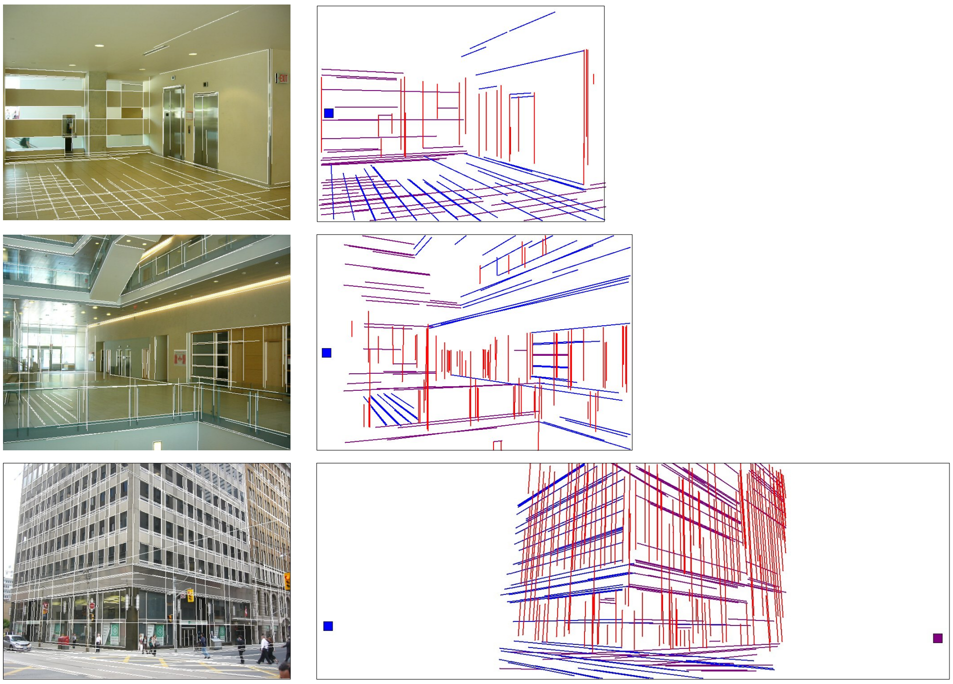

Fig. 4. The first column shows the input image and all detected segments. The second column shows the line clustering result and the estimated finite vanishing points. Each input image has exactly three vanishing points. Parallel lines with the same color represent lines associated with a vanishing point at infinity; the other lines are associated with finite vanishing points.

To compute the focal length, we recovered the camera intrinsic matrix K by decomposing the IAC matrix with unknown focal length.

Our method is compared with three other vanishing point detectors, summarized in Table 1. The method Almansa 2003 detects vanishing regions instead of vanishing points. 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.

Method	**Line detection**	**VP estimation**
Ground Truth [20]	by hand	Gaussian sphere
Tardif 2009 [16]	Canny detector+flood fill	J-Linkage
Almansa 2003 [13]	Helmholtz Principle	Helmholtz Principle

Table 1. Vanishing point detectors used for comparison.

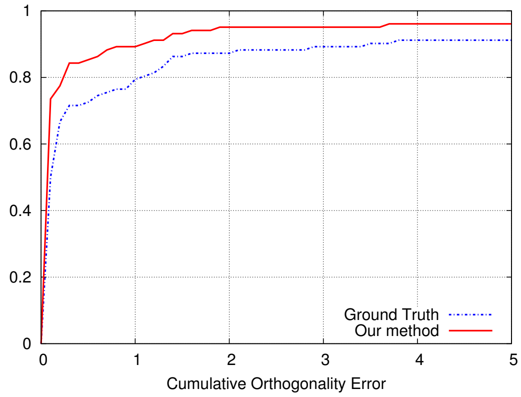

Fig. 5. Cumulative histogram for the estimated errors on York Urban Database. A point (x, y) represents the fraction y of images in the database that have error $e < x$.

Figure 6 shows the cumulative focal length error histogram for our method and for the others methods (Table1) in 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.

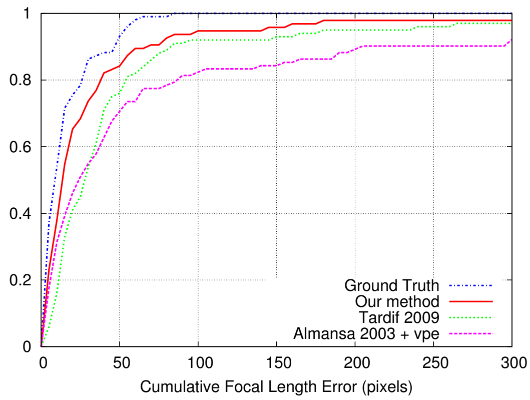

Fig. 6. Cumulative histogram for the focal length errors on York Urban Database. A point (x, y) represents the fraction y of images in the database that have focal length error less than x.

6 Conclusion

This work has examined the problem of estimating vanishing points on an image, a useful tool in large-scale 3D reconstruction, since vanishing points can be used for camera calibration and pose estimation.

We presented a new automated method to detect finite and infinite vanishing point, without any prior camera calibration or threshold tuning. Since the method 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, as illustrated on the experimental results.

The method is effective when applied to images of architectural environments, where there is a predominance of straight lines corresponding to different 3D orientations. This is characterized as a strong perspective. However, if we go to ICCV 2011 in Barcelona, Spain, for example, our pictures will not be good inputs for the method. Most of the buildings in Barcelona have no straight lines, an important characteristic to achieve the detection of the vanishing points.

The results show visually the effectiveness of the vanishing points estimation. The method is also effective when relating the orthogonal vanishing points with the Image of Absolute Conic and for focal length estimation.

Acknowledgments. The authors are grateful to Professor Jorge Stolfi for the valuable discussions on projective space properties, to Jean-Philippe Tardif for providing the source code for his method, and to Patrick Denis for releasing the York Urban database. 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), FAPESP and CAPES.

References

1. Zeng, X., Wang, Q., Xu, J.: MAP Model for Large-scale 3D Reconstruction and Coarse Matching for Unordered Wide-baseline Photos. In: British Machine Vision Conference (2008)
2. Jang, K.H., Jung, S.K.: Practical modeling technique for large-scale 3D building models from ground images. Pattern Recognition Letters 30(10), 861–869 (2009)
3. Lee, S.C., Jung, S.K., Nevatia, R.: Automatic Integration of Facade Textures into 3D Building Models with a Projective Geometry Based Line Clustering. In: USC Computer Vision (2002)
4. Teller, S., Antone, M., Bodnar, Z., Bosse, M., Coorg, S., Jethwa, M., Master, N.: Calibrated, Registered Images of an Extended Urban Area. International Journal of Computer Vision 53(1), 93–107 (2003)
5. Wilczkowiak, M., Sturm, P., Boyer, E.: Using Geometric Constraints through Parallelepipeds for Calibration and 3D Modeling. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(2), 194–207 (2005)
6. Wang, G., Tsui, H.-T., Hu, Z., Wu, F.: Camera calibration and 3D reconstruction from a single view based on scene constraints. Image and Vision Computing 23(3), 311–323 (2005)

7. Wang, G., Tsu, H.-T., Wu, Q.M.J.: What can we learn about the scene structure from three orthogonal vanishing points in images. Pattern Recognition Letters 30(3), 192–202 (2009)
8. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 8(6), 679–698 (1986)
9. Duda, R.O., Hart, P.E.: Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM 15(1), 11–15 (1972)
10. Barnard, S.T.: Interpreting perspective images. Artificial Intelligence 21(4), 435–462 (1983)
11. Tuytelaars, T., Van Gool, L.J., Proesmans, M., Moons, T.: A Cascaded Hough Transform as an Aid in Aerial Image Interpretation. In: International Conference on Computer Vision, pp. 67–72 (1998)
12. Shufelt, J.A.: Performance Evaluation and Analysis of Vanishing Point Detection Techniques. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(3), 282–288 (1999)
13. Almansa, A., Desolneux, A., Vamech, S.: Vanishing Point Detection without Any A Priori Information. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(4), 502–507 (2003)
14. Rother, C.: A New Approach for Vanishing Point Detection in Architectural Environments. In: British Machine Vision Conference (2000)
15. McLean, G.F., Kotturi, D.: Vanishing Point Detection by Line Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(11), 1090–1095 (1995)
16. Tardif, J.-P.: Non-Iterative Approach for Fast and Accurate Vanishing Point Detection. In: International Conference on Computer Vision, pp. 1250–1257 (2009)
17. Desolneux, A., Moisan, L., Morel, J.-M.: Edge Detection by Helmholtz Principle. Journal of Mathematical Imaging and Vision 14(3), 271–284 (2001)
18. Stolfi, J.: Oriented Projective Geometry: A Framework for Geometric Computations. Academic Press (1991)
19. Mardia, K.V., Jupp, P.E.: Directional statistics. John Wiley and Sons (1999)
20. Denis, P., Elder, J.H., Estrada, F.J.: Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery. In: European Conference on Computer Vision, pp. 197–210 (2008)