The VPLF Method for Vanishing Point Computation

Haris G. Baltzakis and Panos E. Trahianias

Institute of Computer Science
Foundation for Research and Technology – Hellas (FORTH)
P. O. Box 1385, Heraklion, 711 10 Crete, Greece
and
Department of Computer Science, University of Crete
P. O. Box 1470, Heraklion, 714 09 Crete, Greece
email: {xmpalt;trahania}@ics.forth.gr

Abstract

A basic problem in computer vision is the extraction of 3D vanishing points (VP) out of 2D perspective projections. Vanishing point analysis provides strong cues for inferring the 3D structure of a scene from only a single view. In this work we present a novel method for VP computation based on a likelihood function that characterizes the plausibility of a point as VP. The performance of our method has been tested with a variety of artificial and real scenes and has been found very promising.

Key words: Vanishing points, unit sphere, vanishing point likelihood function

1 Introduction

Under perspective projection, any set of parallel lines that are not parallel to the image plane converges to a single image point known as vanishing point (VP). In 3D space, parallel lines meet only at infinity and, therefore, VPs are equivalent to projections of points at infinity. Two or more VPs of lines lying on a single plane in space, define, when projected on the image plane, a vanishing line.

Knowledge of a VP, provided that we know the focal length of the camera, gives enough information to determine the 3D orientation of the corresponding group of space lines [1–4]. Consequently, knowledge of a vanishing line provides enough information to determine the 3D orientation of the corresponding space
plane. Even if we don’t know the exact camera geometry, VPs computed on
the image plane can be used to cluster line segments having the same 3D
orientation. Moreover, camera motion [5] or knowledge of scene structure [6],
combined with information obtained from VP can be used to determine camera
parameters.

In recent years, a lot of effort has been devoted to finding VPs out of single
2D perspective projections. Practical methods addressing this task consist of
three distinct steps: (a) extraction of line segments on the image plane, (b)
clustering of line segments to groups of lines converging to the same VPs, and
(c) VP estimation for the extracted line clusters.

The first step is often implemented using a zero-crossing technique to ex-
tract edges from the 2D projection. Edge pixels are subsequently connected
to small-line segments, which are then grouped so that each group forms a
straight long-line segment. For the second and the third step a number of ap-
proaches have been proposed. Direct line clustering combined with numerical
optimization has been studied in [7]. In Hough transform approaches [8–10]
line segments are mapped onto a discrete partitioned space. VPs are detected
as peaks in the resulting histogram. Magee and Aggarwal [11] computed VPs
directly from line segments using cross product operations. Collins and Weiss
[12] used a combination of the methods described by Barnard [9] and Magee
[11] to present a method that combines the high computational speed ob-
tained by Barnard with the accuracy gained from Magee. The Hough trans-
form methods have been extended in [13] by proposing a statistical method
to detect vanishing points from the histogram. In [14] the orientation error in
line segments and the resulting interpretation planes has been modeled tak-
ing thus uncertainty into account; in the same paper, a primitive-based, VP
detection technique has been presented that uses geometric knowledge about
the objects of interest.

In this work we propose a new method for VP detection based on a vanish-
ing point likelihood function (VPLF). This function estimates, for each point
on the image plane, the likelihood that this point is a vanishing point. The
method has certain advantages when compared with existing methods for VP
detection, such as direct applicability to the image edge data (no need for
line segment extraction), and applicability in cases we want to test only cer-
tain areas of the image plane for vanishing points (e.g. we want to detect
only vanishing points at infinity). By computing the VPLF on points on the
unit sphere (image points that have been projected on the unit sphere), we
effectively detect VPs regardless of their 2D position that may be within or
outside the image area or at infinity. The method has been tested extensively
and performed very satisfactorily in cases of simple, artificial images as well
as real images taken from indoor and outdoor scenes.
2 Method Description

The proposed method for VP detection is based on the definition of a Vanishing Point Likelihood Function (VPLF). VPLF is defined in a way to provide the likelihood of any point expressed in image coordinates to be a vanishing point. It operates on the image edge points, eliminating thus the need for line segment extraction; as a result, the required preprocessing is minimal and the method is robust with respect to inaccuracies introduced by line extraction techniques. Moreover, VPLF results are readily interpretable and VP estimation can be made using a simple peak-detection algorithm.

2.1 Preprocessing

The first step in our method consists in the preparation of the input data to VPLF. Since VPLF operates on the image edge points (c.f. next subsection), an edge detection algorithm is used as a preprocessing step. In the current implementation, edges are detected as points where the norm of the gradient operator exceeds a threshold value. As noted above, the proposed method for VP detection eliminates the need for line segment extraction and the detected edge points are simply kept as image points.

2.2 Vanishing Point Likelihood Function

Let \( e_i, 1 \leq i \leq n \), be an edge point and let \( v : (v_x, v_y) \) be a potential vanishing point. We define the contribution of edge point \( e_i \) for \( v \) to be a vanishing point as:

\[
F_v(e_i) = \frac{E_v(e_i)}{w \cdot h}
\]  

(1)

where, \( E_v(e_i) \) is the shaded area illustrated in Fig.(1), and \( w, h \) the width and the height of the image, respectively. Since \( w \cdot h \) is the total area of the image, that is, the maximum value of \( E_v(e_i) \), it follows easily that \( 0 \leq F_v(e_i) \leq 1 \).

The contribution function \( F_v(e_i) \) is in effect a quantitative way to express the orientation of the line passing through the potential vanishing point \( v \) and the edge point \( e_i \). As can be confirmed, edge points belonging to line segments in the image that point to the potential vanishing point, result in similar values for the contribution function \( F_v(e_i) \). This observation forms the basis of our method, which exploits this fact for VP detection.

3
Fig. 1. Illustration of the contribution function \( F \); (a) the potential vanishing point \( V \) is outside the image, (b) the potential vanishing point \( V \) is inside the image.

A few comments regarding the contribution function \( F_v(e_i) \) are in order. As shown in Fig.1, the contribution function \( F_v(e_i) \) of edge point \( e_i \) for point \( v \) to be a vanishing point, depends only on: (a) the dimensions of the image, (b) the position of point \( v \) and, (c) the angle \( f \) of the line passing through points \( v \) and \( e_i \). Since the dimensions of the image are fixed, the contribution function \( F_v(e_i) \) for a given (fixed) point \( v \) to be a vanishing point, depends only on the orientation of the line segment joining \( e_i \) and \( v \). That is, edges lying on straight lines passing through \( v \) give “similar” \( F_v(e_i) \) contributions. It is noted that the actual angular orientation of the line passing through the point \( v \) and the edge point \( e_i \) could not have been used in the place of \( F_v(e_i) \), since, for points \( v \) lying far away from the image center, this orientation is very unstable and tends to become equal to the orientation of the point \( v \) itself regardless of the position of edge point \( e_i \) under consideration. The contribution function \( F_v(e_i) \) can be considered as a scaled version of the actual line orientation, being a robust feature since it exhibits continuity through the edges of the image. In other words, points on the same edge tend to produce very similar values for \( F_v(e_i) \) when the point \( v \) under consideration is a VP.

Based on the previous observation, potential VPs may be detected as points that result in similar values of the contribution function \( F_v(e_i) \) for large sets of edge points \( e_i \). To exploit this, we introduce the distribution \( DF_v(x) \) of the values of \( F_v(e_i) \). Since \( 0 \leq F_v(e_i) \leq 1 \), the parameter \( x \) of \( DF_v(x) \) takes values in the range \([0,1]\). It follows that strong line segments in the image manifest themselves as single peaks on the histogram of \( DF_v(x) \) when \( v \) is a VP. Histograms that contain two or more prominent peaks—that have resulted from corresponding line segments—indicate potential vanishing points, and the detection of such histograms becomes our next goal.
Since the area under the histogram of $DF_v(x)$ is always equal to the number of edge points, the area of the squared histogram $DF_v^2(x)$ will tend to be larger for histograms containing prominent peaks. Therefore, we formulate the *Vanishing Point Likelihood Function* at point $v$, $VPLF_v$, as:

$$VPLF_v = \int_{x=0}^{1} DF_v^2(x) dx \quad (2)$$

Larger values of $VPLF_v$ will correspond to points more likely to be vanishing points.

It should be noted at this point that, operators other than the square operator could have been used in eq.(2) instead. The square operator was chosen (a) due to its simplicity—and hence computational efficiency—and, (b) through extensive experimentation, which showed that in most cases this operator performed very satisfactorily in correctly rating histograms according to the number and the size of their peaks. Other operators that we have experimented with, such as the exponential operator or feature-based peak detection, showed inferior performance in rating the dominant peaks.

### 2.3 Vanishing Point Detection

A straightforward way for vanishing point detection based on VPLF consists in forming the test-set of all possible VPs and computing the value of VPLF for each member of the set. Points resulting in large values of VPLF will more likely be vanishing points. In our implementation, we form the test-set by considering points from the unit sphere and projecting them on the image plane. The unit sphere is discretely parameterized by assuming polar coordinates on the $\phi, \theta$ grid. For each point on the grid, the corresponding $x, y$ image coordinates are given as

$$ (x, y) = \left( f \frac{\sin \theta \cos \phi}{\cos \theta}, f \frac{\sin \theta \sin \phi}{\cos \theta} \right) \quad (3) $$

where $f$ is the camera focal length. The use of the unit sphere ensures that all possible points are considered. Although the camera focal length $f$ is utilized in eq.(3), in order to project candidate points on the image plane, in practice no need for accurate knowledge of it exists. Projecting using a focal length other than the actual one is equivalent to using a sphere other than the unit sphere and plays no role on the accuracy of the computed vanishing points as long as they are expressed in terms of image coordinates.
After VPLF is computed for each grid point on the unit sphere, vanishing points are detected as peaks on the resulting histogram. Such peaks are simply detected as grid points on the sphere exhibiting VPLF values above a threshold value and larger VPLF values than the grid points that form their 8-neighborhood. The latter criterion is used in order to ensure that only one grid point is detected from each peak that may possibly extend over a number of grid points.

Computing VPLF in all grid points on the unit sphere results in increased computational times, with the obvious effect on the algorithm’s computational efficiency. This can be partly remedied by utilizing coarser grid discretizations, in expense, unfortunately, of VP localization accuracy. In order to simultaneously achieve computational efficiency and VP localization accuracy, a hierarchical approach is adopted for peak detection. That is, the unit sphere is parameterized quite coarsely at the beginning, and, after the first estimations of the vanishing points are computed, grid cells near the VPs are split to smaller sub-cells forming the test-set for a second iteration of the process. In our implementation, the cell-division process takes place twice, and, each time cells are split into four sub-cells.

3 Experimental Results

The VPLF method for vanishing point computation has been extensively tested using a variety of images as test data. Sample results obtained from these experiments are presented here in order to demonstrate its performance.

All experiments were conducted by utilizing both the original algorithm and the hierarchical approach described in the previous section. For the original algorithm a discretization scheme of $1^\circ \times 1^\circ$ cell size was adopted, resulting in a $360^\circ \times 90^\circ, \phi, \theta$ grid. For the hierarchical algorithm the initial cell size was $4^\circ \times 4^\circ$, reducing to $2^\circ \times 2^\circ$ and $1^\circ \times 1^\circ$ after the first and the second iteration, respectively. The proper size of the $DF_v(x)$ distribution histogram was found to depend on both the image size and the grid size in use. For image sizes of $256 \times 256$ and grid sizes as the ones described above, a distribution histogram of 90 cells was employed in our experiments.

The first result presented refers to a simple artificial image in order to illustrate the various steps of the proposed method. Figure 2 shows the edges of the image; as can be verified there exist three sets of line segments in the image: vertical lines with two vanishing points at infinity; horizontal lines with two vanishing points at infinity; (semi)diagonal lines with a vanishing point outside the image.
Fig. 2. A simple, artificial image after the edge extraction process.

Fig. 3. The distribution functions (DFs) calculated at four points (shown in Fig.4a):
(a) VP A, (b) VP B, (c) VP C, (d) point D.

Figures 3a,b,c illustrate the distribution $DF$ of the contribution function $F$ at
the positions of three vanishing points: one VP –point A in Fig.4a– formed by
the vertical lines, one VP –point B in Fig.4a– formed by the horizontal lines,
and the VP –point C in Fig.4a– formed by the diagonal lines. Three peaks
are easily seen in each of these histograms, corresponding to equal number
of line segments pointing to that vanishing point. Figure 3d illustrates the
distribution $DF$ of a randomly selected point –point D in Fig.4a. No line
segment points to point D and, therefore, no clear peaks are visible on the
histogram.

Figures 4a,b illustrate the VPLF values all over the $\phi, \theta$ space. Five different
Table 1
Hierarchical computation of vanishing points.

<table>
<thead>
<tr>
<th>True vanishing point ((\phi, \theta))</th>
<th>Initial estimation of vanishing point</th>
<th>Estimation after first iteration</th>
<th>Estimation after second iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Vanishing Point ((\phi, \theta))</td>
<td>Error (degrees)</td>
<td>Vanishing Point ((\phi, \theta))</td>
</tr>
<tr>
<td>A ((0,90))</td>
<td>(-4.88)</td>
<td>4.471</td>
<td>(0,90)</td>
</tr>
<tr>
<td>B ((90,90))</td>
<td>(88.84)</td>
<td>6.323</td>
<td>(90,90)</td>
</tr>
<tr>
<td>A'((180,90))</td>
<td>(180,88)</td>
<td>2.000</td>
<td>(180,90)</td>
</tr>
<tr>
<td>B'((270,90))</td>
<td>(272,84)</td>
<td>6.323</td>
<td>(270,90)</td>
</tr>
<tr>
<td>C ((314.5,64.75))</td>
<td>(316,64)</td>
<td>1.546</td>
<td>(314,64)</td>
</tr>
</tbody>
</table>

peaks are easily visible corresponding to equal number of vanishing points. For line orientations parallel to the image plane (vertical and horizontal lines), two vanishing points are computed. These vanishing points appear at infinity and their projections on the unit sphere have elevation values equal to \(\pi/2\) and azimuth values differing by \(\pi\). That is, the projection of a vanishing point at infinity on the unit sphere lies on the equatorial of the unit sphere and always has its dual-vanishing point lying at the opposite side of the equatorial.

As described above, an iterative approach is employed for accurate peak localization. The successive refinement of the \(\phi, \theta\) coordinates for the estimated vanishing points A, A', B, B' and C (Fig.4a) is shown in Table 1. In each iteration, VPLF values are calculated in a twice as dense grid near the detected vanishing point and a more accurate estimation of the vanishing point is computed. This cell sub-division process is further illustrated in Fig.5 showing a close-up of Fig.4b in the vicinity of vanishing point C. Initial VPLF values corresponding to a \(\phi, \theta\) grid of \(4^\circ \times 4^\circ\) resolution, are presented in the graph at the bottom of the image. The detected peak at \((\phi, \theta) = (316, 64)\) is marked on the graph with a cross while cells lying within its \(5 \times 5\) vicinity are shown brightened. During the next iteration, brightened cells are divided into four sub-cells and new VPLF values are computed (graph in the middle), resulting in a new estimation for vanishing point C at \((\phi, \theta) = (314, 64)\), similarly marked with a cross. This refinement procedure takes place once more (graph on the top), giving rise to the final computation of the vanishing point at \((\phi, \theta) = (315, 65)\) that presents a very accurate estimation of the true vanishing point.

Table 2 shows a comparison of the original and the hierarchical algorithm in terms of computational efficiency for the same image. All execution times given are measured on a non-optimized implementation of the method running on a
Fig. 4. VPLF values in the $\phi, \theta$ space: (a) VPLF values appear as gray levels on the unit sphere (azimuth-elevation parameterization); the pole of the sphere is assumed to be at the image center; the azimuth $\phi$ is assumed to increase clockwise starting from the left side of the image. For convenience the edge image (Fig.2) is also projected on the sphere. (b) VPLF values appear as surface elevation values.
standard personal computer. Since the actual execution time required for each VPLF computation depends on (a) the number of edge points in the image and (b) the number of the cells in the distribution function histogram, details regarding these parameters are also given in the table. As can be observed, the adoption of the hierarchical algorithm resulted in a drastic decrease of computational time, which dropped by a factor greater than $10^{-1}$ to approximately 30 sec. Such execution times are particularly appealing in applications with no real time requirements, such as camera calibration or 3D model matching. Optimized implementations of the algorithm and use of smaller size images may further increase the algorithm’s computational efficiency. A priori knowledge of areas where potential vanishing points are located can be exploited to confine the number of VPLF computations. Searching for vanishing points at infinity for example, in a $128 \times 128$ image, is performed in 0.5 sec with the current implementation.
Table 2
Comparison of the original and the iterative algorithm in terms of computational times.

<table>
<thead>
<tr>
<th>Computational Time (sec)</th>
<th>Non-Hierarchical</th>
<th>Hierarchical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of VPLF Computations</td>
<td>403.25 sec</td>
<td>30.87 sec</td>
</tr>
<tr>
<td>Breakdown of VPLF Computations</td>
<td>32400</td>
<td>2480</td>
</tr>
<tr>
<td>Initial</td>
<td>1980</td>
<td></td>
</tr>
<tr>
<td>1\textsuperscript{st} Iteration</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>2\textsuperscript{nd} Iteration</td>
<td>250</td>
<td></td>
</tr>
</tbody>
</table>

In additional experiments conducted with images of varying complexity the proposed method performed very satisfactorily. Figures 6, 7 & 8 present the results obtained for an artificial indoor scene and a real indoor scene of similar structure (hallways), and a real outdoor (aerial) scene, respectively. In all cases the dominant line orientations have been correctly identified. Figure 6b shows the VPLF results for the artificial indoor scene (Fig.6a); all three dominant line orientations (five VPs) were successfully detected. Similarly, Fig.7b presents the same result for the real indoor scene (Fig.7a) illustrating again the extraction of the three dominant line orientations. The results referring to the aerial scene (Fig.8a) are given in Fig.8b, where the correct identification of the two dominant line orientations can be observed.

4 Discussion

In this paper a novel method for VP detection from raw image data has been presented. The method requires minimal data preprocessing since, unlike current VP detection methods, only the image edge points need to be extracted from the image. It exhibits, therefore, increased robustness with respect to inaccuracies introduced from line segment extraction techniques. Using the locations of the edge pixels, the method computes a vanishing point likelihood function (VPLF) for each point on the image plane. This function estimates the likelihood that the corresponding point is a vanishing point. By computing the VPLF on points on the unit sphere we effectively detect VPs regardless of their 2D position that may be within or outside the image area or at infinity.
Fig. 6. Artificial scene; (a) perspective image, (b) VPLF values on the unit sphere with vanishing points identified.

Fig. 7. Indoor scene; (a) perspective image, (b) VPLF values on the unit sphere with vanishing points identified.

The method was tested on a variety of images and its performance was found very promising.

By employing an iterative, hierarchical implementation of VPLF computation, computational efficiency and accurate VP localization have been achieved. Moreover, all computations are performed on the image plane, relaxing thus the need for accurate knowledge of the focal length. Knowledge of the focal length is only essential in order to convert vanishing points to exact 3D orientations.
Fig. 8. Outdoor scene; (a) perspective image, (b) VPLF values on the unit sphere with vanishing points identified.

Variations of the method can be implemented by using different contribution functions instead of the area function described in the paper. We have tested the method with angle-based functions both on the unit sphere and on the projection plane but we took poor results in cases where VPs were far away from the image center. In expense of computational cost, more sophisticated peak detection algorithms can also be applied both to the distribution histograms and to the final vanishing point detection step.

References


