Vanishing Point Detection

A Tai, J Kittler, M Petrou and T Windeatt

Dept. of Electronic and Electrical Engineering,

University of Surrey,

Guildford, Surrey GU2 5XH, United Kingdom

Abstract

Commencing with a review of methods for vanishing point (VP) detection, a new approach is suggested. The proposed approach estimates the location of candidate vanishing points and provides probability measures which reflect the likelihood of those points being the VPs. This new approach allows VPs to be identified in less structured environments compared with its conventional counterparts.

1 Introduction

Geometrical cues and constraints provide valuable information as to how certain image features should be interpreted. For instance, in many man made scenes there exist a number of straight lines which are mutually parallel in 3D. Under perspective projection these lines will meet at a common point known as the vanishing point (VP). Once this point is identified, one can infer 3D structures from 2D features and this constrains the search for other structures. Also, under known camera geometry the orientation of the lines that are grouped together can be determined from the corresponding VP. Furthermore, two or more vanishing points arising from lines which lie on a certain 3D plane give a vanishing line. This property provides an additional constraint which is particularly relevant when analysing, for instance, aerial imagery where one can often assume that structures of interest lie in a common plane – the ground plane.

The relationships amongst camera parameters, structures in 3D scenes and VPs had been established by Haralick [1]. The applications of VP analysis ranges from extracting 3D structures to the calibration of camera parameters.

An obvious approach to locating VPs is to exploit directly the property that all lines with the same orientation in 3D converge to a VP under perspective transformation. Thus the task of VP detection can be treated as locating peaks in a two dimensional array where the intersections of all line pairs in an image plane accumulate. However, the line pairs can intersect anywhere from points within an image to infinity and this poses problem on implementation.

In order to avoid analysing an open space Barnard proposed the projection of image lines onto a Gaussian sphere [2] [3] [4] which neatly represents any 3D orientations. The plane which contains the lens centre and the line segment in the image intersects with the Gaussian sphere centred at the origin to form a great circle. That is a line segment on the image plane is mapped to a great

circle. Hence, VPs can be detected as elements on the surface of the Gaussian sphere which have relatively high votings. Obviously, the Gaussian surface has to be partitioned in order to accumulate votes. One popular parameterisation is in terms of the azimuth and elevation angles of the unit vector in the sphere. Note that a uniform partitioning in the Hough space maps to non-uniform area in the image plane. For example, the elementary areas at the poles are small compared with those at the equator. This implies that some lines may intersect within a larger area and still be grouped together to hypothesise a VP whereas others may not. A crude way of ensuring accuracy is to partition the Hough plane into finer bins. This improvement in accuracy is paid for by an increase in memory requirement and computational load.

Magee *et al* [3] compute the vectors pointing towards the intersection of line segments in the image plane using a series of cross-product operations. Instead of incrementing a discrete parameterisation, the actual values of the azimuth and elevation angles are maintained for comparison using an arc distance as a metric. This circumvents the problem of the non-uniform elemental surface area of the Gaussian sphere. Although this allows one to locate VPs to a higher accuracy, the computation of the vectors pointing at the intersection points has transformed the $O(n)$ order problem to an $O(n^2)$ one.

Quan and Mohr [4] propose an efficient way both in terms of the amount of operations and memory requirement for computing VPs. They employ a pyramidal data structure instead of a straight forward two dimensional array. This algorithm is similar to a Fast Hough Transform method. It consists of subdividing recursively a patch on the sphere into 4 sub-patches from a coarse to fine resolution. By doing so, a coarse to fine algorithm can be used to improve the efficiency of the Hough Transform (HT). However, detailed experimental studies of hierarchical approaches to the vote accumulation in the HT suggest that the steps that need to be taken to ensure the detection of all level features may render the technique computationally inferior to standard HT implementation.

The methods outlined above do not handle the issue of noise directly, instead the locations of the vanishing points are assumed to be the mid-point of the bin. Thus, the error of the estimated locations of VPs is a function of bin dimensions. Collins and Weiss [5] treat the task of VP detection as a statistical estimation problem. They note that a vector pointing towards the VP lies in the projection plane of the line, and is thus perpendicular to the projection plane normal. In other words, the projection plane normals of 3D parallel lines lie in a plane through the origin, perpendicular to the orientation of the 3D lines in a noiseless environment. In reality, these normals cluster around a great circle forming an equatorial distribution and this distribution is then modelled using Bingham's distribution [6]. It turns out that this approach gives the same result as fitting the least squares perpendicular error planes corresponding to the line pairs.

VPs are also widely used for camera calibrations and recovery of rotational component of motion. Camera calibration involves the determination of camera rotation and translation matrix, focal length etc. These rely on the relationships between the VP coordinates and the camera parameters both intrinsic and extrinsic. The usefulness of VPs in motion analysis stems from the fact that they represent 3D orientations and are therefore invariant to 3D translations between the camera and the scene.

Liou and Jain [7] devised a scheme for road tracking in image sequences.

Assuming that the locations of VP remains unchanged in contiguous frames, they then fit a template around this VP. The dimensions of this template are determined by a constant and a tilt angle. Since the true VP must fall inside this area, one can locate the road boundaries by considering a pair of convergent lines which maximises an *ad hoc* measure based upon the lengths of lines, directions of edge points supporting the lines etc.

Shigag *et al* [8] compute camera rotation by first classifying lines on the image into horizontal and non-horizontal groups and then estimating the camera rotation as the difference of the angle between the optical axis and the 3D orientation of a certain horizontal line. This method takes advantage of the property of vanishing line (vanishing line is the locus of VPs for all lines lie on the same plane). To identify whether a line is horizontal or not, two constraints are applied: (1) horizontal lines should not intersect the vanishing line, (2) locus of VPs of horizontal lines are different from that of non-horizontal ones.

Wang and Tsai [9] proposed an approach to camera calibration based upon the use of vanishing lines. This technique requires only a single view of a cube. The three principle vanishing points (i.e. VPs that correspond to the orthogonal directions of the world coordinates) are detected and the orthocentre of the triangle thus formed gives the image plane centre. This provides a neat way for calibration. The camera orientation parameters are determined from the slopes of the lines forming the triangle. In addition to these they also establish the relationship between the area of the vanishing triangle for calibration of focal length.

In summary, all existing methods for extraction of VPs perform some form of accumulation of line pair junctions. With the Gaussian sphere parameterisation being most popular, while this is a valued approach, there are several shortcomings associated with it.

As bins in the Hough plane map to non-uniform area on the image plane, some intersection points are grouped together under a more stringent condition than others depending on the locations of the VPs. Problem also arises when votes fall into neighbouring bins which might cause a significant peak to diminish in strength. However, most important of all is that accuracies of detected lines are ignored. As far as VPs are concerned the positional and orientational errors cause incorrect intersection points to be formed which reduce the strength of the 'true' peaks and give rise to spurious intersection points which might in turn group with other points to produce 'false' vanishing points. Additionally, this would also disperse intersection points (as the bin size has an impact on Hough Transform) which inherently belong to the same VP. The above points show the sensitivity of this approach to noise. Furthermore, due to the nature of the algorithm any convergent group which consists of a relatively small number of 3D parallel lines would be left undetected.

Most papers on the topic, analyse images of scenes such as offices and corridors which are highly structured and have strong perspective. Consequently, there are less potential VPs and the strength of true VPs are significantly higher than the background and are therefore distinguishable from random intersections.

More recently, Brillault-O'Mahony [10] took into consideration the uncertainty in the detection of line segments and designed an isotropic accumulator space where the probability of erroneous VP detection is uniformly distributed
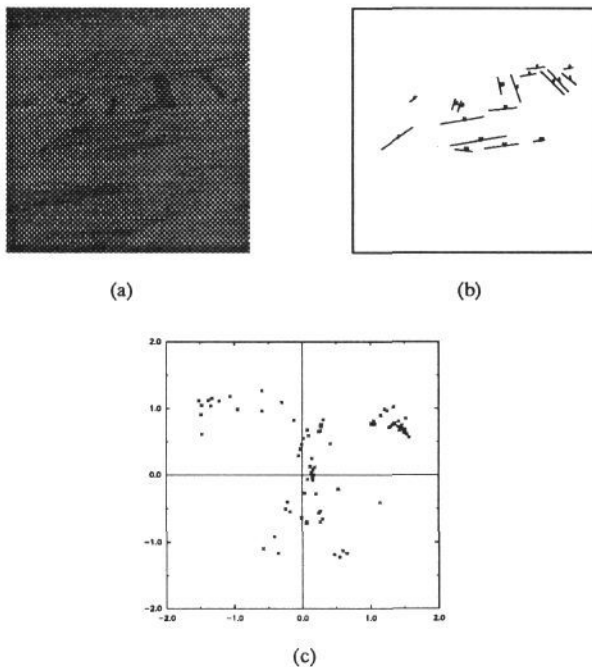
(a)

(b)

(c)

Figure 1: *(a) Original image (b) Hough Transform output (c) Hough Plane accumulation of the azimuth and elevation angles.*

throughout all the cells. In any case, the approach is still based on the accumulator array idea and thus is inappropriate for images with sparse parallel lines.

However, there are many domains of applications where the scene contains only a few lines which are parallel in 3D and on such imagery the existing techniques gave results as illustrated in fig.1. Fig.1(c) shows that there is no dominant peak for the set of lines shown in fig.1(b).

This paper proposes a method which takes a different perspective to detecting vanishing points. Instead of accumulating intersection points, we compute the probability of a group of lines passing the same point. This approach provides a probability measure for discriminating between competing hypotheses irrespective of the size of the vanishing group. In addition, its performance also degrades gracefully in noisy environments. The paper is organised as follows, Section 2 of this paper introduces a novel vanishing point detection method. In Section 3, a probabilistic line representation which is a prerequisite of the method developed. In section 4 we present the experimental results and finally, section 5 offers some conclusions and discussion.

2 Vanishing Point Detection

Let us consider an image with line segments represented by $\rho - \theta$ parameterisation. Due to the geometrical constraints dictated by the image formation process, all perspectively projected line segments having the same orientation

in three space converge to a single point – the vanishing point – in the image under a noise free condition. However, both the imaging and low level edge and straight line extraction processes are inherently noisy resulting in uncertainties in the ρ and θ parameters of the detected lines. Errors in ρ and θ will result in a considerable scatter of the intersection points of the pairs of lines segments which makes it difficult to identify true vanishing points. As pointed out earlier, this problem is particularly pertinent when the scene structure contains only a small number of parallel lines.

In this paper the search for vanishing points makes an explicit use of distribution models of the parameters of the detected lines. With such a probabilistic description for each line we can pose the question of how likely a given point is the common intersection point of a group of lines. In this manner for any selected group of lines we can determine the probability distribution $p(x, y)$ for their mutual intersection point (x, y). A vanishing point is then identified as the point of maximum of this probability distribution function which exceeds some pre-specified threshold.

Let us start by considering a single line with parameters (ρ_i, θ_i) and let the distribution of errors $\delta\rho, \delta\theta$ in ρ_i and θ_i be $p_i(\delta\rho, \delta\theta)$ respectively. Now the probability of the line passing through a point (x, y) in the image will be given by compounding all the combinations of errors $\delta\rho$ and $\delta\theta$ such that the true line with parameters

$$\rho = \rho_i + \delta\rho \tag{1}$$
$$\theta = \theta_i + \delta\theta \tag{2}$$

satisfies the constraint equation

$$\rho = x\cos\theta + y\sin\theta \tag{3}$$

the compound probability $P_i(x, y)$ is thus given by

$$P_i(x, y) = \frac{1}{z_i} \int_s p_i(\delta\rho, \delta\theta) ds \tag{4}$$

where the integration is performed in the parameter space along the sinusoidal line defined by (3) and z_i is the normalising constant to ensure that $p_i(x, y)$ is a probability density function. In terms of parameter errors the compound probability can be expressed as

$$P_i(x, y) = \frac{1}{z_i} \int_{-\pi}^{\pi} p_i(\delta\rho, \delta\theta) \sqrt{1 + (\frac{d\rho}{d\theta})^2} d\theta \tag{5}$$

The compounding process is illustrated in fig.2.

Now let $X = \{e_i | i = 1, 2, ...k\}$ be a group of lines selected from the best of lines output by an image description process, with the measured parameters for each line denoted by vector $\underline{\omega}_i = [\rho_i, \theta_i]^T$ and the associated error distribution by $p_i(\delta\rho, \delta\theta)$. By analogy the probability that the lines jointly pass through a point (x, y) in the image plane (which extends beyond the physical imaging area of the sensor) is given by

$$P(x, y) = \prod_{i=1}^{k} \frac{1}{z_i} \int_{-\pi}^{\pi} p_i(\delta\rho, \delta\theta) \sqrt{1 + (\frac{d\rho}{d\theta})^2} d\theta \tag{6}$$
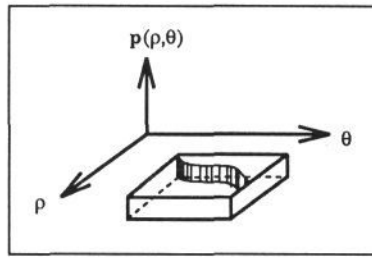
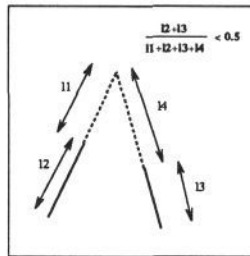**Figure 2:** *Rectangular probability density function.*

Figure 3: *Evidential Support.*

From the knowledge of $p_i(\delta\rho, \delta\theta)$ the probability of $P(x, y)$ can easily be evaluated. Its mode (x_v, y_v) then defines a vanishing point provided $P(x_v, y_v)$ is above the threshold.

In order to develop a practical procedure based on the above idea we first need to select a suitable groups of lines. Regarding these lines, the method is intended for finding vanishing points of small sets of 3D parallel lines, hence the cardinality of the group should be quite small. Moreover, the computational complexity of the problem could potentially grow combinatorially with the number of lines in the group. In the present approach the initial analysis is performed for line triplets. Any larger group of lines is formed after this first analysis stage by considering the proximity of detected vanishing points and the overlap of the two participating line sets.

To prune the set of all possible triplets each candidate group of lines must satisfy a number of criteria. These include

1. angular constraints (similarity of θ_i values)

2. distance constraint (the perpendicular distance of line pair intersection point from the third line)

3. junction quality constraint [11](the lines should intersect at a point which is remote from all participating line endpoints as illustrated in fig.3.)

4. imaging geometry constraints (if known)

3 Probabilistic Line Representation

In the light of the discussion in the previous section we require an appropriate line representation which can associate uncertainties with its parameters. It is important that this representation is easy to convert to and from the standard Hough Transform(HT) $\rho - \theta$ space representation, since we use this method for the extraction of straight lines.

The parametric representations that we adopted for a perfect line are either $v_1 = [\rho, l, \theta, L]^T$ or $v_2 = [x_m, y_m, \theta, L]^T$, where ρ is the distance between the foot of the normal and the origin, l is the distance from the foot of the normal to the line midpoint; θ and L are the line orientation and length respectively. x_m, y_m are the coordinates of the line midpoint.

Deriche and Faugeras [12] also address the issue of finding an appropriate line representation. Their conclusion is that vector v_2 is a more favourable choice than vector v_1 simply because representation v_1 leads to a covariance matrix that strongly depends upon the position of the associated line segment in the image through the effect of ρ and l. Hence, from this standpoint v_1 does not allow different Kalman filters to be applied on each parameter. However, as far as our application is concerned it does not matter what the interactions between various parameters are. We only need the necessary statistical parameters to build the error models which we can utilise for the development of a formal approach to VP detection.

A simple analysis involving Taylor series expansion leads to the following approximate relationship between the errors in line orientation, line midpoint and the distance from the origin to the foot of the normal:

$$\delta\rho = \frac{-\rho\delta\theta^2}{2} - (x_m \sin\theta - y_m \cos\theta)\delta\theta + \delta x \cos\theta + \delta y \sin\theta \tag{7}$$

Note that in deriving the above equation, we assume that any terms involving the cross product of positional and orientational errors can be neglected. For lines four pixels long or more the quadratic terms become negligible. Equation (7) then gives a linear relationship between the errors in orientation θ and line segment midpoint position and the errors in ρ. Thus if $\delta\theta, \delta x$ and δy are normally distributed, so will the errors ρ.

A Monte Carlo experiment was performed to check the validity of the approximate model and its dependence on line length. From table 1 it is apparent that provided the line length $L >= 4$ the linear model yields a distribution of errors $\delta\rho$ with negligible skew and curtosis which can be taken to imply that it closely approximates a Gaussian. Thus if $\delta\theta, \delta x$ and δy are Gaussian, the joint distribution of $\delta\rho, \delta\theta$ and δl will be Gaussian with covariance matrix

$$\begin{pmatrix} l^2\sigma_\theta{}^2 + \sigma^2 & -l\sigma_\theta{}^2 & (sin2\theta)\sigma^2 - l\rho\sigma_\theta{}^2 \\ -l\sigma_\theta{}^2 & \sigma_\theta{}^2 & \rho\sigma_\theta{}^2 \\ (sin2\theta)\sigma^2 - l\rho\sigma_\theta{}^2 & \rho\sigma_\theta{}^2 & \rho^2\sigma_\theta{}^2 + \sigma^2 \end{pmatrix} \tag{8}$$

Note that $\sigma_x{}^2 = \sigma_y{}^2 = \sigma^2$. From (8) the distribution of interest $p(\delta\rho, \delta\theta)$ is normal with zero mean and covariance matrix

$$\Sigma = \frac{1}{N} \begin{pmatrix} l^2\sigma_\theta{}^2 + \sigma^2 & -l\sigma_\theta{}^2 \\ -l\sigma_\theta{}^2 & \sigma_\theta{}^2 \end{pmatrix} \tag{9}$$

Line $x_m = -36.6, y_m = 136.6, \theta = 150°$		mean ($\bar{\rho}$)	stdev (σ_ρ)	skew (σ_3)	curtosis (σ_4)
First	$L = 1.0$	0.06199	8.77310	-0.01172	0.10832
order	$L = 4.0$	0.01441	2.40076	0.00618	-0.07306
approx.	$L = 10.0$	0.00489	1.33047	0.02300	-0.03726

Table 1: *Statistical results of error in ρ ($\delta\rho$) obtained from Monte Carlo experiment.*

where N is the number of points providing evidential support for the line. Note that N may differ form L, the line length as some pixels inside the line segment may be undetected. The covariance matrix in (9) applies if the Hough Transform (HT) line detection scheme has an optimisation facility to estimate the most likely values of ρ and δ. When the standard HT is used with ρ and δ parameter quantisation, the probability distribution $p(\delta\rho, \delta\theta)$ is no longer Gaussian. Instead its shape may vary from the rect to roof function depending on the coarseness of the quantisation process. A study of the relative merits of the various detection schemes and the associated probabilistic line representations is beyond the scope of this paper.

4   Experimental Results

The method was applied to aerial imagery of resolution $256X256$. The rectangular line parameter probability distribution was used in the experiment.
    The implementation involved the following steps:

1. Compute the intersection points of all possible pairs and discard those that fall outside the virtual image (this is an imaginary image of size 512X512 centred at the origin).

2. Combine a line with a pair to create a triplet to see if it satisfies the pursuing constraints of section 2.

3. Set up a window around the line pair intersection point and compute the probability $P(x, y)$ for all (x, y) in the window. Find the mode of $P(x, y) - (x_v, y_v)$.

4. Repeat steps (2) and 3 for all other triplets.

    The result shown in fig.4 demonstrates that the vanishing points which grouped together the lines of interest (i.e. the runways and taxiways) are ranked highly and their corresponding VPs are in close proximity to each other (these VPs can be grouped to extract larger vanishing groups). Fig.5 shows the probability envelope of a VP in image space.
    When the conventional VPs finder was applied on the image shown in fig.4, the runways and taxiways were again detected but there is no way of discriminating between the various hypotheses. If the size of the peak were used as a discriminating criterion, false vanishing groups would be extracted.
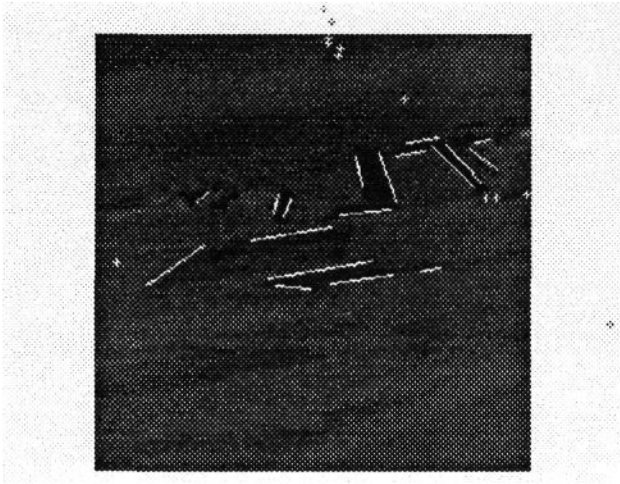
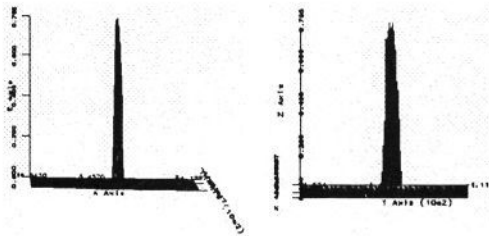**Figure 4:** *New vanishing points finder result on areial imagery.*

Figure 5: *Probability profile of a certain VP.*

5 Discussion and Conclusions

The parameter space in which the algorithm proposed here operates is an open one, since the intersection point of a line pair can lie somewhere between the image and infinity. This, however, should not present much difficulty as lines whose 3D orientations are the same only cease to converge to a point under a very restricted viewpoint. This also demonstrates the relationship between convergent and parallel groups. 3D parallel features when projected onto the image plane can only transform into convergent or parallel groups.

The algorithm proposed in this report provides an estimate of the VP location as well as producing a performance measure. Thus VP hypotheses can be compared and discarded on the basis of their quality measure values rather than the number of lines which converge to a certain point. This offers a common ground for comparison between convergent groups (i.e small convergent groups can also be detected). However, there is a tradeoff between small convergent groups and the probability of accidental coincidence. As the size of the convergent groups reduces, the chance of accidental coincidence increases, which renders the location of true vanishing points more difficult. Therefore some

kind of geometrical cue is needed to recover true vanishing points. Vanishing lines are powerful cues to be exploited for the purpose. They are defined as the locus of VPs formed by 3D parallel groups which lie on the same plane. This property means that the hypothesised points must lie on the vanishing line and thereby provide a constraint for discarding false VPs.

References

[1] R.M. Haralick. Using perspective transformation in scene analysis. *Computer Graphics Image Processing*, 13:191–221, 1980.

[2] S.T. Barnard. Methods for interpreting perspective images. In *Proc. Image Understanding Workshop*, pages 193–203, Palo Alto, California, Sept 1982.

[3] M.J. Magee and J.K. Aggarwal. Determining vanishing points from perspective images. *Computer Vision, Graphics and Image Processing*, 26:256–267, 1984.

[4] L. Quan and R. Mohr. Determining perspective structures using hierachical hough transform. *Pattern Recognition Letters*, 9(4):279–286, 1989.

[5] R.T. Collins and R.S. Weiss. Deriving line and surface orientation by statistical methods. In *Proc. Image Understanding Workshop*, pages 433–438, 1990.

[6] R.T. Collins and R.S. Weiss. Vanishing point calculation as a statistical inference on the unit sphere. *Computer Vision, Graphics and Image Processing*, pages 400–403, 1990.

[7] S. Liou and R. Jain. Road following using vanishing points. *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 41–46, 1986.

[8] L. Shigag, S. Tsuji, and M. Imai. Determining of camera rotation from vanishing points of lines on horizontal planes. *Computer Vision, Graphics and Image Processing*, pages 499–502, 1990.

[9] L. Wang and W. Tsai. Computing camera parameters using vanishing-line information from a rectangular parallelepiped. *Machine Vision and Applications*, (3):129–141, 1990.

[10] B. O'Mahony. *A Probabilistic Approach to 3D Interpretation of Monocular Images*. PhD thesis, City University, London, 1992.

[11] A. Etemadi, J-P. Schmidt, G. Matas, J. Illingworth, and J. Kittler. Low-level grouping of straight line segments. In *Proc. British Machine Vision Conference*, pages 118–126, 1991.

[12] R. Deriche and O. Faugeras. Tracking line segments. *ECCV*, pages 259–268, 1990.

Acknowledgements