



# A ground truth based vanishing point detection algorithm

Andrew C. Gallagher\*

*Electronic Imaging Products, Eastman Kodak Company, 1700 Dewey Avenue, Rochester, NY 14650-01816, USA*

Received 24 August 2000; accepted 18 June 2001

## Abstract

A Bayesian probability-based vanishing point detection algorithm is presented which introduces the use of multiple features and training with ground truth data to determine vanishing point locations. The vanishing points of 352 images were manually identified to create ground truth data. Each intersection is assigned a probability of being coincident with a ground truth vanishing point, based upon conditional probabilities of a number of features. The results of this algorithm are demonstrated to be superior to the results of a similar algorithm where each intersection is considered to be of equal importance. The advantage of this algorithm is that multiple features derived from ground truth training are used to determine vanishing point location. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

*Keywords:* Vanishing point detection; Bayes network; Gaussian sphere; Perspective; Line segments; Projection

## 1. Introduction

Perspective projection describes the mapping of a three-dimensional scene (a set of real world objects) onto a two-dimensional image plane (a photograph.) Detailed descriptions of perspective projection may be found in Ballard and Brown [1], and Kanatani [2]. As shown in Fig. 1, perspective distortion maps a point in the three-dimensional scene  $(x_o, y_o, z_o)$  to a point on the image plane  $(x'_o, y'_o, z'_o)$  by the following equation:

$$(x'_o, y'_o, z'_o) = \left( \frac{x_o f}{z_o}, \frac{y_o f}{z_o}, f \right), \quad (1)$$

where  $f$  is the focal length of the optical system. The point  $\mathbf{p} = (x_o, y_o, z_o)$  may also be represented as a vector from the origin. Thus,  $\mathbf{p}$  may be written in terms

of sums of multiples of unit vectors in the directions of the  $x$ -,  $y$ -, and  $z$ -axis ( $\mathbf{i}$ ,  $\mathbf{j}$ , and  $\mathbf{k}$ , respectively.) In this representation, the point  $\mathbf{p} = (x_o, y_o, z_o)$  may be represented as  $\mathbf{p} = x_o \mathbf{i} + y_o \mathbf{j} + z_o \mathbf{k}$ .

Since the 15th century, it has been known that the perspective projection of lines that are parallel in a three-dimensional scene meet at a common vanishing point on the image plane [3]. The vanishing point  $\mathbf{v}$  may be represented as a point on the image plane  $\mathbf{v} = x_{vp} \mathbf{i} + y_{vp} \mathbf{j} + f \mathbf{k}$ .

For example, Fig. 2 shows an image of a room with several sets of parallel lines. This scene contains at least two sets of mutually orthogonal lines: the vertical lines of the windows, doors, and framing and the horizontal lines of the windows, ceiling, and couch. Each of these two sets of lines has a corresponding vanishing point, shown as \*, and +, respectively, in Fig. 3. The vanishing point of vertical scene lines must fall on the vertical axis of the image, providing that the camera is rotated about the  $x$ -axis only. The real image in this example shows that the vanishing point marked by \* falls almost on the vertical image axis. The reason the vanishing point does

\* Fax: +1-716-722-0160.

*E-mail address:* andrew.gallagher@kodak.com  
(A.C. Gallagher).

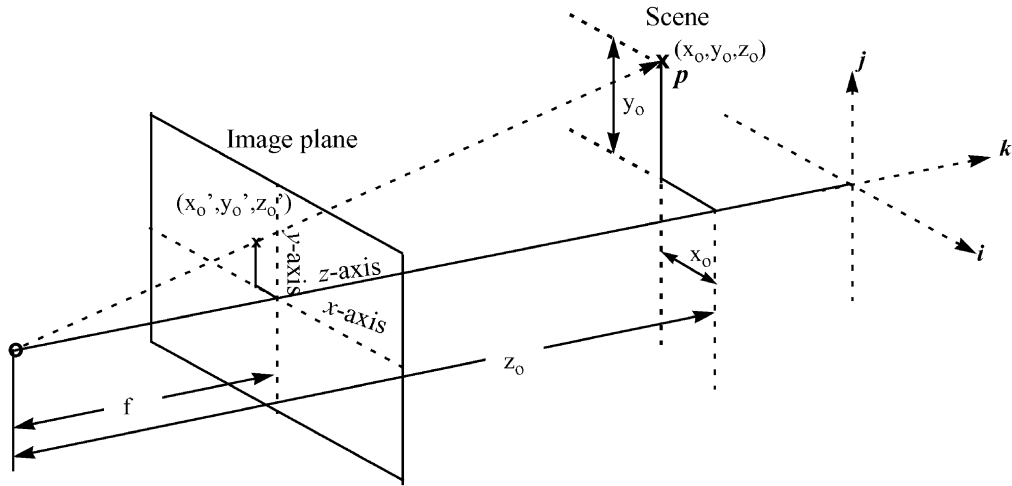


Fig. 1. Perspective projection of a single point  $x$  in space to a point on the image plane. The point of projection is marked with  $(o)$ .



Fig. 2. An image with multiple sets of mutually orthogonal parallel scene lines. The location of a vanishing point is determined by finding the intersection of a set of parallel scene lines.

not fall exactly on the vertical image axis is that there is some small amount of rotation of the camera about the  $z$ -axis (the optical axis).

Barnard [4] first proposed using a Gaussian mapping [5] to represent the vanishing point  $v$  as a point on a unit sphere (also called a Gaussian sphere) centered at the origin (the focal point) of the optical system. This mapping of a point on the image plane to the Gaussian sphere is accomplished simply by normalizing the vector from the origin to the point  $v$  to have magnitude 1.0, as

$$v_G = \frac{v}{|v|}. \tag{2}$$

Any vanishing point  $v = (x_{vp}, y_{vp}, f)$  may be represented as some vanishing point vector  $v_G = (x_G, y_G, z_G)$ . In fact, any point in space (except the origin) may utilize this Gaussian mapping. The vanishing point vector  $v_G$  has been found to be a useful representation because the entire image plane extending to infinity in all directions may be mapped onto the  $z_G \geq 0$  half of the Gaussian sphere. For example, the point on the image plane  $v = (0, \infty, f)$  is mapped to  $v_G = (0, 1, 0)$ . In order to find the location on the image plane referenced by a vector on the Gaussian sphere, simply scale the vector to have the focal length  $f$ , given as

$$v = \frac{f v_G}{z_G}. \tag{3}$$

The vanishing point  $v$  occurs at the intersection of the image plane and the vector  $v_G$ .

Fig. 3 shows the mapping of the two vanishing points of the image first shown in Fig. 2 from the image plane to the unit sphere.

Notice that Fig. 3 also labels the image axes as the vertical and horizontal axes. In this example, the horizontal axis is parallel to the  $x$ -axis, and the vertical axis is parallel to the  $y$ -axis. It is important to note that the vertical and horizontal axes are labeled with respect to the orientation of the particular image under consideration, rather than the optical system frame of reference. The vertical and horizontal axes always pass through the image origin and are always parallel to either the  $x$ -axis or the  $y$ -axis of the optical system frame. The vertical axis is considered to be collinear with “top” and “bottom” on the image. The horizontal axis is orthogonal to the vertical axis and runs from “left” to “right” on the image.

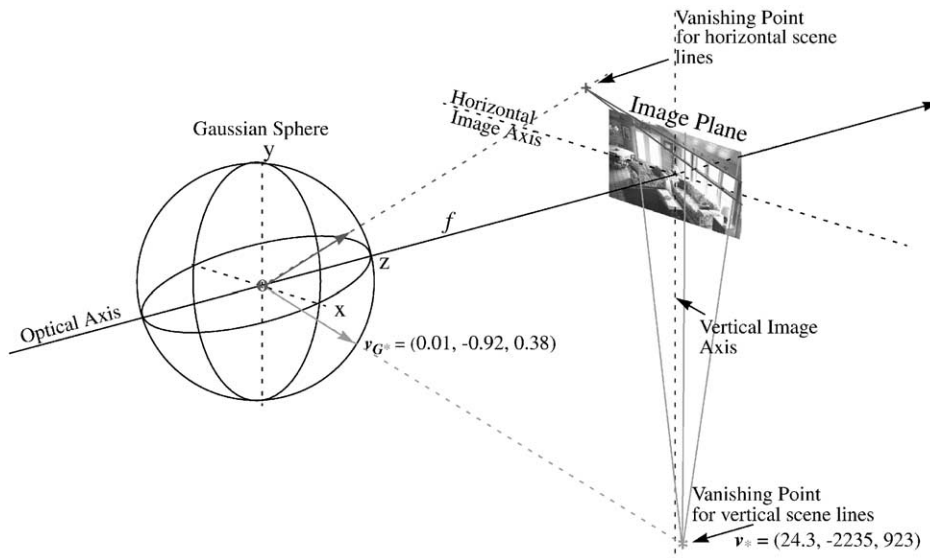


Fig. 3. A typical arrangement used to represent a vanishing point with a Gaussian sphere. The optical system is centered at the origin, shown by  $o$ .

The convenient vanishing point vector notation made possible by the Gaussian mapping is commonly used by researchers who study vanishing points. Strictly speaking, the center of the unit sphere is located a distance  $f$  equal to the focal length of the camera from the image plane. However, for the purpose of vanishing point representation, Magee and Aggarwal [6] have shown that the distance  $f$  need not be the true focal length of the camera, as discussed in Section 2. Throughout the work presented in this paper, the focal length  $f$  is assumed to be equal to the length of the frame diagonal, which is a fairly common camera design. In the case of a 35 mm negative, the focal length is assumed to be 43.3 mm. The images used in this research are digitized at  $512 \times 768$  pixels, so the focal length is assumed to be 923 pixels. As an example, the vertical vanishing point  $*$  of Fig. 3 may be represented on the image plane as  $v_* = (24.3, -2235, 923)$ , or as a vanishing point vector  $v_{G*} = (0.01, -0.92, 0.38)$ .

Throughout this paper, it should be assumed that points in space (including the ground truth vanishing points and the automatically detected vanishing points) are represented by the corresponding Gaussian mapping (2), unless otherwise noted.

## 2. Past vanishing point work

Barnard [4] first proposed the idea of using a Gaussian sphere as an accumulator space for vanishing point detection. The “plane of interpretation” is identified

as the plane passing through the center of the sphere (the origin or focal point of the optical system) and both ends of a particular line segment. Each bin of the Gaussian sphere accumulator that falls on the intersection of the Gaussian sphere and the interpretation plane (this intersection forms a great circle) is incremented. After this procedure is completed for all line segments, the vanishing points may be found by searching for local maxima on the sphere. The position of the local maximum represents the vanishing point vector. The location of the vanishing point in the image plane may be determined by projecting this vector back onto the image plane (3). One difficulty with Barnard’s approach is that the partitioning of the Gaussian sphere causes nonuniform bin sizes that affect the final results.

Brillault-O’Mahony [7] and Quan and Mohr [8] build on Barnard’s work by describing vanishing point detection using an alternative accumulator space and an alternative quantization of the Gaussian sphere, respectively.

Magee and Aggarwal [6] propose a vanishing point detection scheme that is similar to Barnard’s method in that the Gaussian sphere is again utilized. However, the Gaussian sphere is not used as an accumulator. With their method, cross products operations are performed in order to identify the intersections of line segments on the Gaussian sphere, forming a list of all intersections of each pair of line segments. Clustering is performed to find common intersections that are identified as possible vanishing points. Also, they show that the algorithm is

insensitive to focal length. This method has several advantages. First, the accuracy of the estimated vanishing point is not limited to the quantization of the Gaussian sphere. Secondly, this method allows for the consideration of each intersection individually. Thus, intersections that do not make sense (such as an intersection occurring on one of the two component line segments) as a vanishing point may be rejected. This was not possible with the accumulator space methods, which consider only line segments rather than intersections.

Collins and Weiss [9] devised a scheme of statistical inference which also utilizes a Gaussian sphere where the normal to each plane of interpretation is determined. A statistical fitting is performed in order to find the polar axis of an equatorial distribution. Shufelt [10] reports that the method of statistical inference had poor performance on test data when compared with Barnard's method.

Kanatani [2] describes much of the theory of vanishing points and perspective projection. However, rather than using a Gaussian sphere as an accumulator space, he describes the vanishing points with  $n$ -vectors, vectors of length 1 originating from the perspective point and pointing to a location in the image plane. These  $n$ -vectors are similar to the vectors used by Magee.

Lutton et al. [11] introduce a probability mask on the Gaussian sphere in order to compensate for the finite extent of the image. The probability mask attempts to account for this effect by considering the probability of two random lines (noise) intersecting at any given place on the Gaussian sphere. However, the authors fail to take into consideration the prior probability of the actual vanishing point locations, and instead assume that all vanishing points are equally likely. This assumption is far from the actual case for typical consumer imagery (see Fig. 5). In addition, the authors describe an effort to account for errors in line segment identification. Rather than incrementing only the bins falling on the great circle, the authors propose incrementing the bins falling in a swath about the great circle. The swath size is determined by possible interpretation planes passing through the endpoints of the line segments. However, this weighting scheme is based upon assumption rather than actual ground truth data.

Shufelt [10] describes the method of primitive modeling in order to increase the robustness of the vanishing point detection. A model is used to determine the relative position of multiple vanishing points. Knowledge of the primitive model assists the vanishing point detection by allowing peak detection to include sampling multiple bins simultaneously, but requires an accurate estimate of the focal length of the imaging system. In addition, Shufelt reviews the performance of several of the other vanishing point detection methods with a database of aerial images.

In the present work, ground truth data is utilized to establish conditional probabilities in a cross product scheme similar to Magee's. Features pertaining to the two line segments forming each intersection are used to determine the conditional probability that the intersection is coincident to a ground truth vanishing point. Weighted clustering is then performed to determine the most likely vanishing point location, considering the collection of intersections and conditional probabilities. This algorithm is based upon the ground truth vanishing point data derived from the 86 images in the training set. This algorithm represents the first reported use of ground truth data for training an automatic vanishing point detection algorithm.

### 3. Ground truth collection

A user interactive tool was created for the purpose of manually identifying the vanishing points of an image. The author used this tool to identify the vanishing points for two image sets consisting of typical consumer images: 86 images from a training database referred to as "TSD", and a 266 image testing database referred to as "JBLL", for a total of 352 images. Incidentally, both of these databased contain approximately 50% indoor images and 50% outdoor images. The following "rules of thumb" were used in the ground truth collection:

- (a) Vanishing points are identified in order of significance. The most significant vanishing point is identified first.
- (b) Vanishing points from human-made structures only are identified. No attempt was made, for instance, to assume that trees in a forest are substantially parallel or that rows of crops are parallel.
- (c) After a first vanishing point is identified, a second vanishing point is identified only if the set of line segments appear to have been approximately orthogonal in the scene to the line segments corresponding to the first vanishing point. Likewise, a third vanishing point is identified only if there exists another set of line segments that appear to have been orthogonal in the scene to the line segments corresponding to both the first and second vanishing points.

It should be noted that this ground truth selection is somewhat subjective, especially in light of the term "significance" in rule (a) above. It can be imagined that a variety of subjects would, on a given image, disagree about which vanishing point is "most significant". In addition, subjects could conceivably disagree about whether a significant vanishing point appears at all in the image. Thus, if the vanishing point ground truth data reported in this document is found to be compelling in nature, it may be desirable to corroborate the data with additional subjects.

Table 1  
Frequency of images with a specific number of significant vanishing points

	0	1	2	3	Total
TSD 94 v.p.'s	36 41.2%	16 11.6%	24 27.9%	10 11.6%	86
JBIL 200 v.p.'s	129 48.5%	79 29.7%	53 19.9%	5 1.9%	266
Total	165 46.6%	95 28.8%	77 21.7%	15 4.2%	354

#### 4. Results of ground truth collection

As previously mentioned, the vanishing point ground truth collection was performed over two databases. The metric of these images is a logarithmic scaling of exposure. A total of 94 vanishing points were identified in the TSD database, and 200 in the JBIL database. A complete breakdown of the number of images containing a specified number of vanishing points is shown in Table 1. The statistics vary somewhat between the databases. For each database, slightly more than half of the images contained at least one significant vanishing point. However, the TSD database contained a greater percentage of images with 2 and 3 vanishing points than did the JBIL database. Assuming that these databases reflect an appropriate sampling of image space, it appears safe to state that approximately  $\frac{1}{2}$  of all consumer images contain at least one significant vanishing point.

In order to gain insight into the ground truth data, each vanishing point vector  $\mathbf{v}_G = (x_G, y_G, z_G)$  is further transformed to the angles  $\phi$  and  $\theta$ , calculated as follows:

$$\phi = \cos^{-1}(z_G), \quad (4)$$

$$\theta = \tan^{-1}(y_G/x_G). \quad (5)$$

Note that the angle  $\theta$  is unwrapped so that its value is between 0 and  $2\pi$ . For example, a vanishing point corresponding to the vector (0.05, -0.86, 0.51) has values of  $\phi = 1.0356$  rad and  $\theta = 4.7705$  rad. The angle  $\phi$  is the angle from the optical axis to the vector, and ranges from 0 (when the vector falls directly on the optical axis and points to the image center) to  $\pi/2$  (when the vector is orthogonal to the optical axis and intersects with the image plane at infinity).

The angle  $\theta$  is the angle between the projection of the vector onto the  $xy$ -plane and the positive  $x$ -axis. Fig. 4 shows a graphical representation of the angles  $\phi$  and  $\theta$

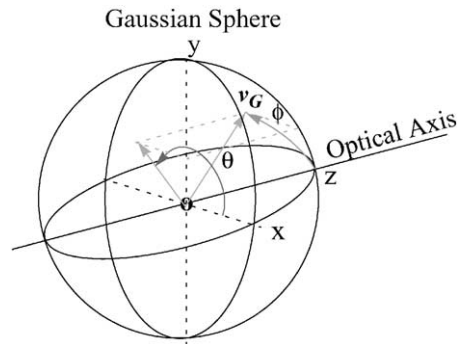


Fig. 4. An illustration of angles  $\phi$  and  $\theta$  on the Gaussian sphere.

with respect to the Gaussian sphere. Thus (assuming that the horizontal axis of the image is parallel with the  $x$ -axis and the vertical axis of the image is parallel with the  $y$ -axis), vanishing point vectors which have  $\theta = 0$  radians correspond to the vanishing point along the right side of the horizontal axis of the image. Likewise, vanishing point vectors which have  $\theta = \pi/2$  correspond to the vanishing point along the upper portion of the vertical axis of the image, etc.

Fig. 5 shows an image of the two-dimensional histogram of the 294 vanishing point vectors from both databases, quantized into bins of  $\phi$  and  $\theta$ . (For this discussion, the vanishing point of each image was determined with respect to the image in upright orientation. That is, the vertical axis of the image is parallel to the  $y$ -axis. In addition, the “top” of the image is in the direction of the positive  $y$ -axis.) In this representation, the darker bins represent bins that contained more counts. Black represents a bin having 25 or more counts, and white represents 0 counts. The interesting features of this histogram occur especially for large  $\phi$ , that is, vanishing points far from the image center (large  $\phi$ ) or at infinity ( $\phi = \pi/2$ ). It is clear that, contrary to Lutten et al. [11], every vanishing point is not equally likely. Vanishing points with large  $\phi$  tend to coincide with the vertical or horizontal axis of the image (since there are peaks at  $\theta = \{0, \pi/2, \pi, 3\pi/2\}$ .) When a vanishing point is far from the image center, it is extremely probable that the vanishing point will fall near either the horizontal or vertical axis of the image. Additionally, the peaks at  $\theta = \{\pi/2, 3\pi/2\}$  are much stronger than the peaks at  $\theta = \{0, \pi\}$ . This data shows that it is far more likely that the vanishing point will fall on the vertical axis of the image than the horizontal axis of the image.

This observation is logical for several reasons. First of all, there is a predominance of vertical lines in man-made structures. When a photograph is taken with the camera held perfectly level and straight (i.e., the optical axis ( $z$ -axis) and the  $x$ -axis are orthogonal to the vertical lines

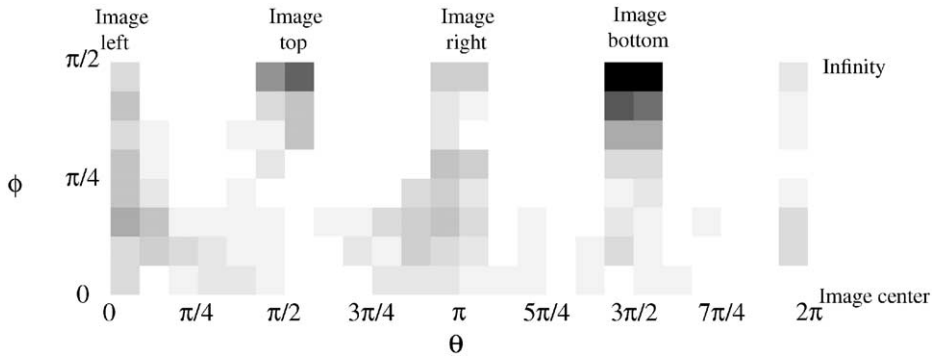


Fig. 5. A histogram of the locations of the vanishing points determined by the ground truth study. The density of each bin corresponds to the number of ground truth vanishing points falling within the bin. These bins are not equal size partitions of the Gaussian sphere; however, along each row (fixed  $\phi$ ) the bin sizes are equivalent. A majority of vanishing points coincide with  $\theta = 3\pi/2$ .

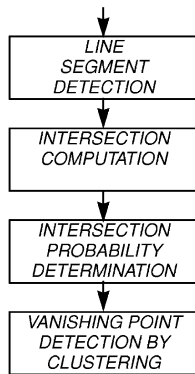


Fig. 6. A high-level block diagram of the Bayesian probability based vanishing point algorithm.

of the scene), the vanishing point will occur on the vertical axis of the image at infinity. However, in consumer photography it is common for the photographer to pivot the camera slightly up or down in order to achieve a more pleasing image composition (i.e., the  $x$ -axis remains orthogonal to the vertical lines in the scene, but the optical axis does not). In this case, it can be shown that the vanishing point of vertical scene lines will still fall on the vertical axis of the image, but will not occur at infinity. Thus, for typical camera positions, the vanishing point of vertical scene lines will fall on the vertical axis of the image, independent of the camera tilt about the  $x$ -axis.

**5. A vanishing point detection algorithm based on ground truth**

Fig. 6 shows a high-level block diagram of the ground truth based vanishing point detection algorithm. First,

image line segments are detected. Next the intersections of these line segments are calculated. Each intersection is then analyzed to determine its probability of being coincident (as will be defined below) to a true vanishing point. Finally, the intersections are clustered based on the probabilities to determine the algorithm’s guess at vanishing point location.

*5.1. Line segment detection*

Because vanishing points relate to the projection of scene lines to image lines, all vanishing point detection algorithms have some inherent component of straight line detection. In fact, it is difficult to compare the results of various vanishing point algorithms unless the line detection method is also described.

A number of authors have addressed the problem of straight line detection [12–15]. A line in an image exhibits a collection of local edges. Thus, line detection algorithms generally work by detecting edges (and/or gradients) in the image, then determining collections of these edges that occur in the image to form lines.

Kahn et al. [14] expands on the approach taken by Burns [12]. The line support region clustering is essentially unchanged except that a threshold is placed upon the gradient magnitude to exclude a major portion of image pixels from further processing. In addition, rather than using a planar fit of the line support region to determine the parameterization of the line, lines are directly fitted to the line support region by using principal component analysis. Kahn also mentions that the ratio of eigenvectors provides a useful estimate of line quality  $Q$ . In an effort to keep complexity to a minimum, the line segment detection algorithm used in the present work is essentially a hybrid of the algorithms described by Burns [12] and Kahn [14].

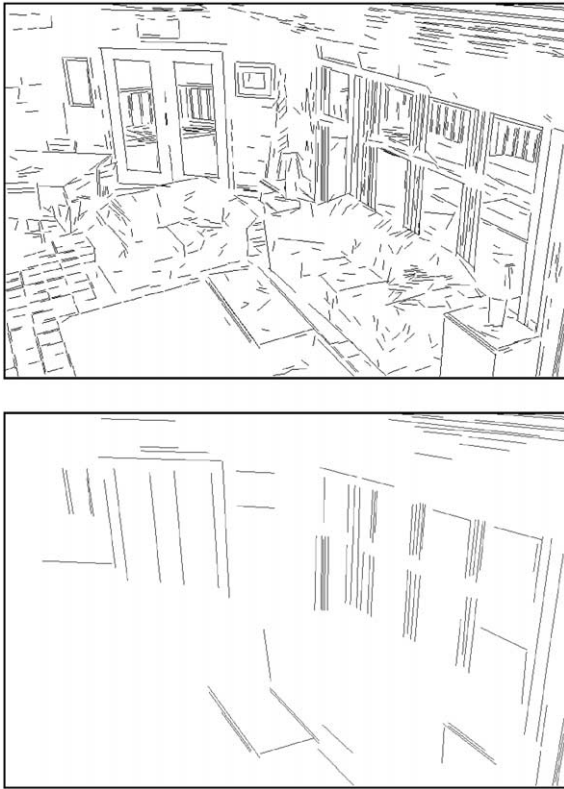


Fig. 7. Images of the lines passing different criterion. Top: All lines (997) with at least 17 pixels in the line support region and a  $Q$  of at least 10. Bottom: All lines (92) with at least 100 pixels in the line support region and a  $Q$  of at least 200. This setting is used for the vanishing point detection.

As a final stage of the line segment detection, line thresholds are applied to remove short line segments and line segments of low quality. Any line segment with  $N$ , the number of pixels belonging to the line support region, less than a threshold  $t_1$  is excluded from further processing. The threshold  $t_1$  is currently set at 100. In addition, any line segment having a quality  $Q$  less than another threshold  $t_2$  is excluded from further processing. Currently the threshold  $t_2$  is set at 200. Examples of the detected lines passing certain threshold criteria are shown in Fig. 7.

After the line detection has been completed, an image has an associated list of  $M$  line segments that meet the requirements to be considered a valid line segment.

## 5.2. Intersection computation

Most vanishing point detection algorithms are somehow related to finding points where a relatively large number of line segments intersect.

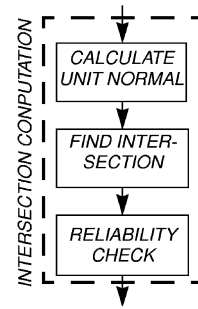


Fig. 8. Line segment intersection algorithm.

For each possible pair of lines from the  $M$  lines detected from the image, the intersection is computed using a series of cross products. Thus,  $M$  valid lines results in a total of  $M(M-1)/2$  intersections. Each intersection is considered to be a possible vanishing point location. Each intersection is assigned a probability of coinciding with a correct vanishing point, based upon characteristics of the two lines used to compute the intersection. Finally, the most probable vanishing point is determined by examining the Bayesian probabilities associated with the intersection.

Fig. 8 is a block diagram describing the process of determining the intersection of two line segments,  $L_i$  and  $L_j$ . Each line segment contains two endpoints that may be represented by location in the image plane. As a note on nomenclature, for each image there exists  $M$  line segments, referred to by the labels  $L_i$ , where  $i = 1, 2, \dots, M$ . A specific attribute of a given line  $i$  is referred to by using the same line subscript. For example,  $A_i$  refers to the attribute  $A$  associated with line segment  $L_i$ .

As shown in Fig. 9, line segment  $L_i$  has endpoints  $p_{1i}(x_1, y_1)$  and  $p_{2i}(x_2, y_2)$ . These endpoints may be represented as vectors, by mapping onto the Gaussian sphere as previously described by Eq. (2). (Estimated focal length  $f = 923$  pixels.) Thus, the vector  $p_{Gi}$  is a unit vector in the direction from the origin to  $p_{1i}$ .

As described by Barnard [4], an image line segment can be thought of as forming an interpretation plane, the plane which contains the origin and the line segment. The unit normal  $N_i$  to the interpretation plane associated with line segment  $L_i$  may be determined by computing the cross product of the unit vectors from the origin to each of the line segment endpoints, as given in the following equation:

$$N_i = \frac{p_{Gi1} \times p_{Gi2}}{|p_{Gi1} \times p_{Gi2}|}. \quad (6)$$

Thus, the unit normal  $N_i$  of the interpretation plane corresponding with each line segment  $L_i$  is determined with Eq. (6).

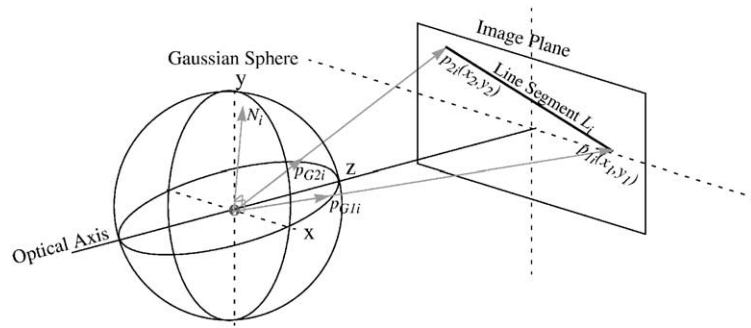


Fig. 9. A typical arrangement used to represent a vanishing point with a Gaussian sphere. The optical system is centered at the origin, shown by *o*.

The Gaussian vector representation of the intersection  $I_{ij}$  of any two interpretation planes associated with any two line segments  $L_i$  and  $L_j$  can then be determined by the cross product of the unit normals

$$I_{ij} = \frac{N_i \times N_j}{|N_i \times N_j|} \quad (7)$$

As shown in Fig. 8, once the intersection  $I_{ij}$  is computed, a reliability check is performed. If  $N_i = N_j$  (i.e., line segments  $L_i$  and  $L_j$  are collinear), the cross product has magnitude zero, which causes a singularity for Eq. (7). In addition, the intersection  $I_{ij}$  is checked to ensure that it does not occur interior to either of the line segments. Intersections formed from collinear line segments and intersections occurring interior to either line segment or the other are ignored. Thus, after this stage, there are a maximum of  $M(M - 1)/2$  intersections. The number of rejected intersections is generally on the order of 2–4% of the total number of intersections.

Typical values for  $M$  range from 0 to 130, meaning that for some images, up to 8385 intersections are calculated.

### 5.3. Intersection probability determination

Let  $V$  be the event that an intersection  $I_{ij}$  is coincident with a ground truth vanishing point  $v_G$ . The definition of coincident will be discussed in further detail below.

Rather than assume that each intersection has equal importance as the prior research does, a probability  $p_{ij}$  is assigned to each intersection. This probability is the probability of  $V$ , given what is known about the corresponding line segments  $L_i$  and  $L_j$ . Bayesian reasoning is used to determine  $p_{ij}$  from an analysis of the ground truth data. Several types of features are utilized to establish the value of  $p_{ij}$ , including line segment length, difference in line segment

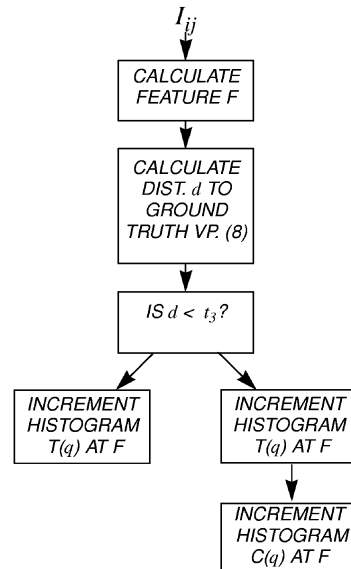


Fig. 10. Block diagram of the procedure used to establish the conditional probabilities for the Bayesian probability.

angle  $\theta$ , difference in line color, and line intersection location.

The ground truth data from the TSD database was used to develop the posterior probabilities in the manner described by Fig. 10. Consider the feature  $F$ . The value of the feature  $F_{ij}$  for the intersection  $I_{ij}$  is calculated based upon the attributes of the lines  $L_i$  and  $L_j$ . Additionally, the intersection  $I_{ij}$  is computed from  $L_i$  and  $L_j$  as described by Eq. (7).

Next, the intersection  $I_{ij}$  is compared with the ground truth vanishing points for the image. The distance between two unit normal vectors on the Gaussian sphere is measured by the arc length between the two vectors, which is equivalent to the angle between the two vectors.



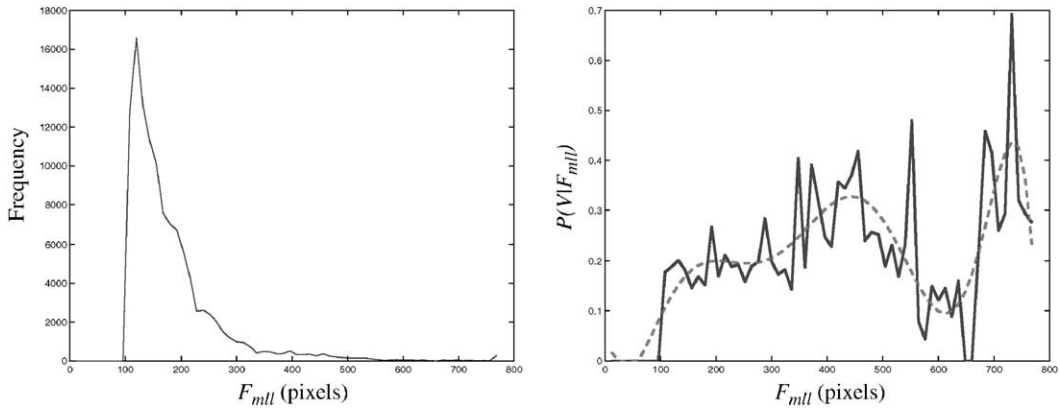


Fig. 11. Left: A histogram of the feature  $F_{mll}$ , the minimum length of the two lines used to calculate the intersection. Right: The probability  $P(V|F_{mll})$  (solid) and a smoothed version (dashed) of the same. Note that the fluctuations in the function at high values of  $F_{mll}$  result from insufficient data in the training set.

The calculation of this distance is

$$d_{ij} = \min_x [\text{acos}(I_{ij} \cdot \mathbf{v}_{Gx})]. \quad (8)$$

The distance  $d_{ij}$  is computed as the minimum between the intersection  $I_{ij}$  and any of the  $X$  ground truth vanishing points  $\mathbf{v}_{Gx}$  associated with the image. If the distance  $d$  is less than a pre-defined threshold  $t_3$  (currently,  $t_3 = 0.1$ ), then the intersection  $I_{ij}$  is considered to be coincident to the ground truth vanishing point. Thus the “correct” histogram  $C(q)$  is incremented at the location of the feature  $q = F_{ij}$ . Additionally, regardless of  $d_{ij}$ , the “total” histogram  $T(q)$  is also incremented at the location of the feature  $q = F_{ij}$ .

When this analysis is performed over a large number of intersections and images, the desire is to get a fairly accurate estimate of the probability that an intersection  $I_{ij}$  is coincident with a vanishing point, given feature  $F_{ij}$ . This posterior probability is approximated by

$$P(V|F_{ij}) = \frac{C(q = F_{ij})}{T(q = F_{ij})}. \quad (9)$$

If several features  $F_1, F_2, \dots, F_n$  are given for an intersection, then the probability of the intersection being a valid vanishing point given these features may be estimated with the following, based on Bayes theorem:

$$p_{ij} = P(V|F_{1ij}, F_{2ij}, \dots, F_{nij}) \\ = \frac{P(V|F_{1ij})P(V|F_{2ij}) \dots P(V|F_{nij})}{P(V)^{n-1}}. \quad (10)$$

Note that Eq. (10), requires that the individual features  $F_{1ij}, F_{2ij}, \dots, F_{nij}$  are independent. This equation is the basis for a single layer Bayesian network [16].

Several features specific to the line segment detection algorithms were developed for use with this Bayesian approach.

The first feature is the minimum line length  $F_{mll}$ . The value of this feature is determined by analyzing the line segments  $L_i$  and  $L_j$  from which the intersection  $I_{ij}$  is determined. The value of the  $F_{mll}$  feature for intersection  $I_{ij}$  is the minimum of the lengths of the line segments  $L_i$  and  $L_j$ .

The total histogram for the feature  $F_{mll}$  is shown in Fig. 11 based on an analysis of the TSD database. In this database, there are 116,133 valid intersections total. Of these, 19.1% (~22,000) of the intersections occur within the threshold distance of  $t_3 = 0.1$  radians from a ground truth vanishing point. Thus, the prior probability  $P(V) = 0.191$ . Fig. 11 shows a plot of  $P(V|F_{mll})$ , calculated with Eq. (9). Note that a smoothed version of this probability is also created with a polynomial fit and shown with a dashed line. Notice that as the value of feature  $F_{mll}$  increases, so does the probability that the intersection  $I_{ij}$  will be coincident with a vanishing point  $\mathbf{v}_G$ . For very high values of  $F_{mll}$  (above ~450), the probability  $P(V|F_{mll})$  exhibits noisy behavior due to the few number of samples (about 2% of the population), as can be seen in the histogram  $T(F_{mll})$ . Note that this noise behavior may also be observed with regard to the remaining features as well.

A second feature,  $F_{cA}$ , is based on the average color of the line support regions from which  $L_i$  and  $L_j$  are determined. The color based feature  $F_{cA}$  is the Mahalanobis distance between the average color of the line segment  $L_i$  and the average color of the line segment  $L_j$ . As shown in Fig. 12, an intersection has a higher probability of being a vanishing point if the line segments which determine the intersection are of similar color. This result seems logical because parallel lines often are edges of the same object (e.g., sides of a building, window panes,

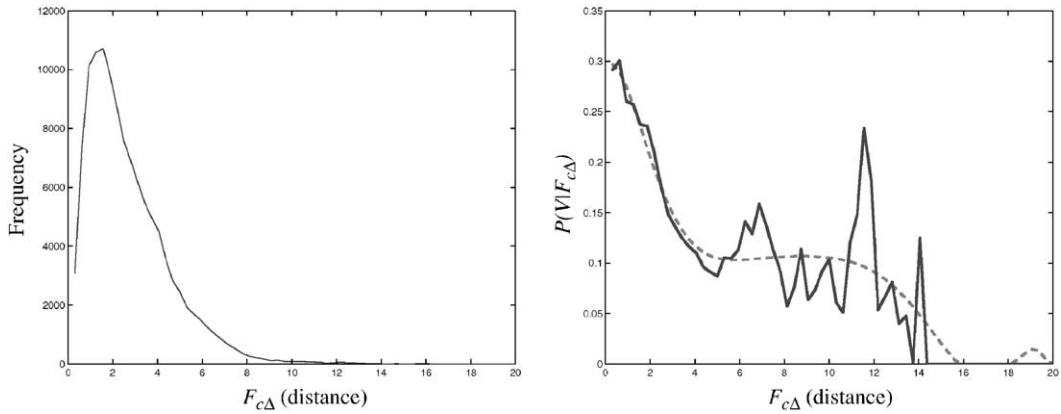


Fig. 12. Left: A histogram of the feature  $F_{c\Delta}$ , the Mahalonobis distance between the average colors of the two line support regions. Right: The probability  $P(V|F_{c\Delta})$  (solid) and a smoothed version (dashed) of the same.

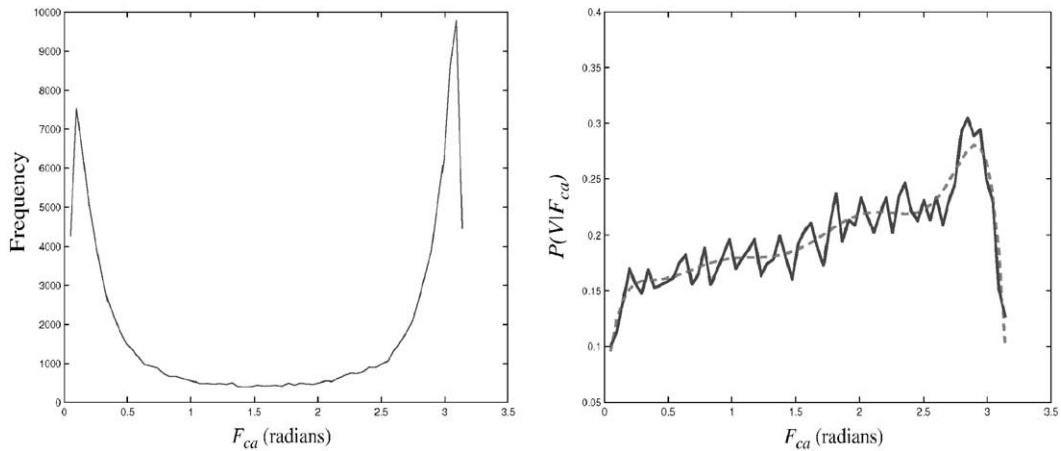


Fig. 13. Left: A histogram of the feature  $F_{ca}$ , the angle between the neutral axis and the vector connecting the average color of the two line support regions. Right: The probability  $P(V|F_{ca})$  (solid) and a smoothed version (dashed) of the same.

etc.). Additionally, a single object is often a consistent color throughout its surface. Parallel lines will likely be of similar color because the lines result from a single uniformly colored object.

A third feature,  $F_{ca}$ , is also based on the difference in color of the line support regions of  $L_i$  and  $L_j$ . However, in this case, the distance is the angular distance in radians between the neutral axis (red = green = blue) and the vector adjoining the mean color of the line support regions of  $L_i$  and  $L_j$ . It should be expected that the colors of parallel lines may differ by only a shift in the luminance direction. As shown in Fig. 13, the value of  $F_{ca}$  is zero when the color of two lines differ by only a luminance shift. (Note that 0 and  $2\pi$  are identical in this metric.) It is quite possible for two lines to have a high value of  $F_{c\Delta}$  and a small value

of  $F_{ca}$  when the lines differ in color by a luminance shift. When  $F_{ca}$  has a value of  $\pi$ , that indicates that both line segments have similar luminances but different colors.

A fourth feature,  $F_{\Delta a}$ , is the difference in angle between the two line segments  $L_i$  and  $L_j$ . The feature  $F_{\Delta a}$  is computed with the following simple equation:

$$F_{\Delta a} = |\theta_i - \theta_j|. \tag{11}$$

Fig. 14 shows that the intersection formed by line segments having similar angle is more likely to be a valid vanishing point that the intersections of lines with greatly different angles.

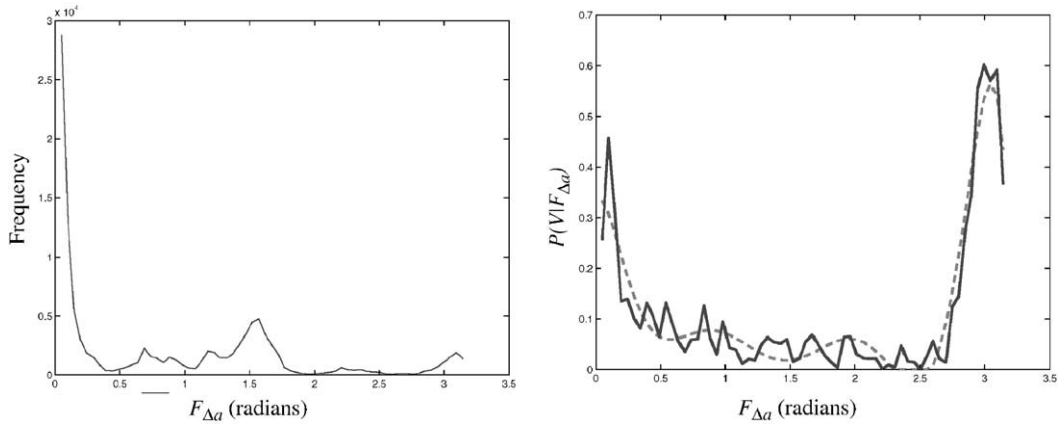


Fig. 14. Left: A histogram of the feature  $F_{\Delta a}$ , the difference in radians between the two line segments on the image plane. Right: The probability  $P(V|F_{\Delta a})$  (solid) and a smoothed version (dashed) of the same.

A fifth feature  $F_{loc}$  is based upon the location of the intersection  $I_{ij}$  on the Gaussian sphere alone. This location is expressed in terms of  $\phi$  and  $\theta_w$ .

$$\phi = \cos^{-1}(z_{GI}) \tag{12}$$

$$\theta_w = \begin{cases} \theta_v & \text{if } \theta_v < \frac{\pi}{4}, \\ \frac{\pi}{2} - \theta_v & \text{if } \theta_v \geq \frac{\pi}{4}, \end{cases} \tag{13}$$

where  $\theta_v = |\tan^{-1}(y_{GI}/x_{GI})|$  and  $I_{ij}$  is represented by the gaussian mapping  $(x_{GI}, y_{GI}, z_{GI})$ .

The angle  $\phi$  is the angle from the optical axis (the  $z$ -axis) to the Gaussian representation of the intersection. The angle  $\phi$  ranges from 0 to  $\pi/2$ . The angle  $\theta_w$  is the smallest angle from the projection of the intersection onto the  $xy$ -plane to the nearest of the  $x$ - or  $y$ -axis. Note that the angle  $\theta_w$  is wrapped so that its value is limited to between 0 and  $\pi/4$ .

Fig. 15 shows a two-dimensional display of the histogram of all 116,133 intersection locations. (Black corresponds with a high number of occurrences, and white corresponds with no occurrences.) Additionally, Fig. 15 shows the probability of an intersection being coincident with a ground truth vanishing point, given the intersection location. (Black corresponds with high probability.) Again, this prior probability contradicts the Lutton et al.'s [11] assumption that each point on the Gaussian sphere has equal likelihood of being a vanishing point. The equal probability assumption would be true if the camera were held at random positions to the scene; however, we know that this is not the case. This feature is independent of the distribution of the orientations of the images in the training database, since the value of  $\theta_w$  is image orientation independent. Another version of the feature  $F_{loc}$  could be made if the orientation of the image is known. However, this variation has not yet been explored.

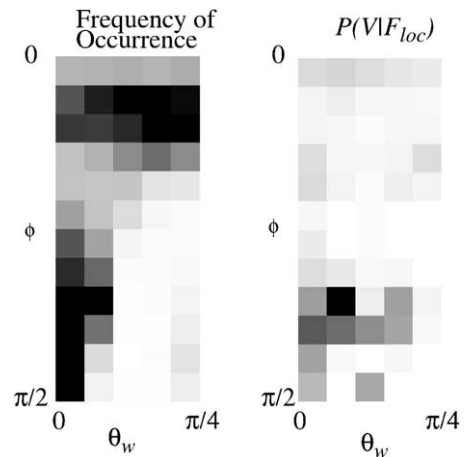


Fig. 15. Left: A histogram of the feature  $F_{loc}$ , the location of the intersection expressed in terms of  $\phi$  and  $\theta_w$ . Black corresponds with a high number of occurrences, and white corresponds with no occurrences. Right: The probability  $P(V|F_{loc})$ . Locations having a higher probability are black, and lower probability is lighter. Notice that although a large number of intersections occur near the image center (small  $\phi$ ), there is a low probability that these intersections coincide with vanishing points.

Note that Lutton et al. [11] suggests a probability masking to account for the effect of the restricted retina (finite image size). He suggests that the accumulator cells be scaled by the inverse of this mask. However, as previously noted, Lutton assumes that all vanishing points are equally likely and does not base the probability on any ground truth data. Additionally, Lutton's result is not directly applicable for any particular line detection scheme.

Thus, for each intersection  $I_{ij}$  determined from line segments  $L_i$  and  $L_j$ , there is an associated probability that the intersection location is coincident (within some tolerance) with a ground truth vanishing point. In the algorithm, look-up tables (LUTs), each with 64 indexes, are used to determine the posterior probabilities as shown in Figs. 11–14. The smoothed version of each probability is used in the application of the algorithm. Eq. (10) describes how the probabilities from individual features may be combined. Although the five features are probably not totally independent as Eq. (10) requires, this assumption does not greatly affect the utility of the estimate of  $P(V|F_{1ij}, F_{2ij}, \dots, F_{nij})$ . The results will illustrate that the estimation of the vanishing point location is improved with multiple features, despite the approximation that the features are independent. Removing this correlation between features could be accomplished by the creation of intermediate features in the form of a multilevel Bayes network [16].

The use of all features is not required. Any of the 32 possible subsets of the five features may be used with the current implementation of the algorithm without any retraining of the algorithm.

Determining the relative importance of the five features is difficult, but may be approximated with a weighted measure of variance. The more that the probability changes over the range of the feature, the better that feature is for differentiating some intersections from others. One measure of the ability of a feature to differentiate amongst the intersections may be given as a variance of the posterior probability that is weighted by the histogram of the feature. This ability  $A$  may be calculated as:

$$A = \frac{\sum_{F_i} T(F_i)[P(V|F_i) - P(V)]^2}{\sum_{F_i} T(F_i)} \quad (14)$$

For the five features given, the value of  $A$  is 0.0018, 0.0046, 0.0033, 0.0275, and 0.0616, respectively. Thus, we expect the most benefit from the location feature  $F_{loc}$ .

#### 5.4. Vanishing point detection by clustering

At this point in the algorithm we have a list of (at most)  $M(M - 1)/2$  intersections  $I_{ij}$ , each with associated probability  $p_{ij}$  that intersection  $I_{ij}$  is coincident with a vanishing point. Each intersection  $I_{ij}$  may be viewed as a classifier, and the associated probability  $p_{ij}$  represents the associated confidence. The question is: How should this large number of classifiers be combined to determine an overall estimate of the vanishing point? Fig. 16 shows a high-level diagram of the procedure to determine the most probable vanishing point from the collection of intersections  $I_{ij}$  and the associated probabilities  $p_{ij}$ . Fig. 17

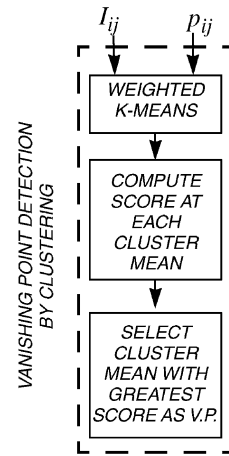


Fig. 16. Block diagram of the process of combining the probabilities  $p_{ij}$  for each intersection  $I_{ij}$  to form an estimated vanishing point.

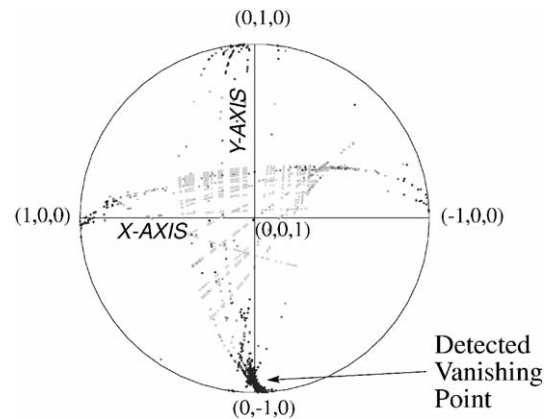


Fig. 17. The 4080 intersections resulting from the 92 detected line segments shown in Fig. 7. The shade of each intersection corresponds to the probability  $p_{ij}$  determined for that intersection, with dark corresponding to high probability of coinciding with a vanishing point and light corresponding with low probability. This view is looking from the image plane down the optical axis toward the Gaussian sphere. The cluster of high-probability intersections on the negative  $y$ -axis corresponding to a vanishing point of the vertical scene lines.

illustrates a graphical example of the intersections and associated probabilities for the image shown in Fig. 2.

First, a constrained weighted K-means algorithm using up to 20 clusters is applied (up to a maximum of 20 iterations) to the  $M(M - 1)/2$  intersections for the purpose of determining cluster means.

After the weighted K-means clustering, the cluster means are examined to select the most likely vanishing

point. For each cluster mean  $C_m$ , the intersections  $I_{ij}$  occurring within some distance of the cluster mean  $C_m$  are examined. The distance  $\delta_2$  between each intersection and cluster mean is measured as an arc length on the Gaussian sphere, as given in Eq. (8). Those intersections having a distance  $\delta_2$  less than  $t_4$  ( $t_4 = 0.1$  was used for this research) from a cluster mean  $C_m$  are considered to be “nearby intersections”. A weight factor  $w_m = 1$  when the intersection  $I_{ij}$  is a nearby intersection to cluster mean  $C_m$ , and  $w_m = 0$ , otherwise. Note that the clusters determined by the K-means clustering are not of uniform size, so the K-means clusters are not used for further analysis. Based on the associated probability  $p_{ij}$  of each intersection  $I_{ij}$  nearby to cluster mean  $C_m$ , an overall score  $S_m$  for the corresponding cluster mean is generated. The cluster mean  $C_m$  with the greatest score  $S_m$  is determined to be the most likely vanishing point for the image.

Each intersection is considered to be a random event with a probability  $p_{ij}$  of being coincident with a ground truth vanishing point  $v_G$ . In order for  $C_m$  to be coincident with vanishing point, at least one of the nearby intersections must correspond with a vanishing point. Thus, determining the probability that any one of the intersections nearby to  $C_m$  is coincident with a vanishing point is a score that may be used to determine the most likely vanishing point.

$$S_{1m} = 1.0 - \prod_{i,j} w_m(1 - p_{ij}). \quad (15)$$

When a cluster mean  $C_m$  has a large number of nearby intersections (as is often the case), the quantity is often extremely close to 1.0, beyond the accuracy of the numerical representation of the computer. In fact, a computer often has trouble distinguishing between the scores of two cluster means. In order to avoid this problem, a logarithmic approach is taken

$$S_{2m} = - \sum_{i,j} w_m \log(1 - p_{ij}). \quad (16)$$

This scoring function is a monotonically increasing transformation of Eq. (15). Therefore, the vanishing point detected as the cluster mean with the maximum scoring value will be identical in the case of Eqs. (15) or (16).

By the preceding method, the most likely vanishing point for an image is selected as the cluster mean with the highest score. Obviously, if an image has fewer than two detected lines meeting the requirements, then no intersections and no vanishing points can be computed. However, it is highly likely that such an image has no vanishing points.

After a first vanishing point has been detected, subsequent vanishing points may be detected. Each line

segment  $L_i$  is compared with all previously identified vanishing points. If the normal of line segment  $L_i$  is nearly orthogonal to a vanishing point already detected, then that line segment is assigned to that vanishing point and is omitted from further analysis. After each line segment has been reviewed, the process of computing intersections, performing K-means clustering, and selecting the most likely vanishing point may be repeated. This process may be repeated as many times as desired, until fewer than two lines remain unassigned to a vanishing point. Each repetition generates another vanishing point estimate. Generally it does not make sense to attempt to detect more than three vanishing points.

## 6. Pseudocode

The overall flow of the algorithm may be described in pseudocode as follows:

*Step 1.* Detect line segments (Section 5.1).  $M$  line segments  $L$  are detected from the input image.

*Step 2.* For each of the  $M(M - 1)/2$  possible line segment pairs  $L_i$  and  $L_j$

*Step 3.* Compute intersection (Section 5.2). The intersection  $I_{ij}$  of the line segment pair  $L_i$  and  $L_j$  is found as follows:

*Step 3.1.* Calculate unit normals  $N_i$  and  $N_j$  for each line segment of the pair (6).

*Step 3.2.* Find intersection  $I_{ij}$  using the cross product of the unit normals  $N_i$  and  $N_j$  (7).

*Step 3.3.* If reliability check fails, proceed to next line segment pair.

*Step 4.* Determine intersection probability (Section 5.3) as follows:

*Step 4.1.* Calculate features  $F = \{F_{m11}, F_{ca}, F_{ca}, F_{\Delta a}, \text{ and } F_{loc}\}$  for intersection  $I_{ij}$ .

*Step 4.2.* Determine (by LUT) posterior probabilities for each of the intersection’s features  $P(V|F)$ .

*Step 4.3.* Combine the intersection’s posterior probabilities with Bayes theorem (10) to determine the probability  $p_{ij}$  that the intersection  $I_{ij}$  is coincident with an actual image vanishing point.

*Step 5.* Detect vanishing point by clustering intersections and associated probabilities (Section 5.4).

*Step 5.1.* Apply weighted K-means clustering to find a set of cluster means.

*Step 5.2.* For each cluster mean, compute score  $S$  (16).

*Step 5.3.* Select cluster mean with greatest score  $S$  as the vanishing point.

## 7. Results

As an example of this algorithm applied to a real image, Fig. 18 shows the line segments corresponding to a

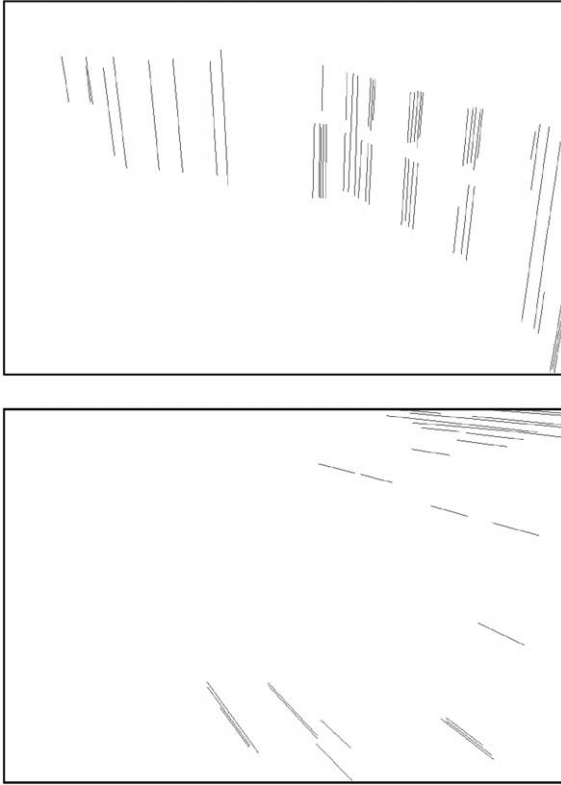


Fig. 18. The first and second algorithm output for the image from Fig. 2.

first (top image) and a second (bottom image) vanishing point, detected automatically. In this case, the algorithm correctly identifies a first and second vanishing point for the image.

In order to assess feature robustness, the conditional probabilities were computed on the JBJL testing database. The results are shown in Fig. 19. Generally, the results appear similar to the conditional probabilities computed for the TSD training set, with the exception of the conditional probability  $P(V|F_{ca})$  for small  $F_{ca}$ .

As a measure of the algorithm performance, the algorithm was trained on the 86 images from the TSD database, and then tested on the 266 images from the JBJL database. The algorithm was configured to output a maximum of two vanishing point estimates per image, and over the 266 images, 483 vanishing points were detected by the algorithm. If no features are used, then all probabilities  $p_{ij}$  default to 1.0 and the algorithm essentially performs the algorithm described by Magee and Aggarwal [6], where the most likely vanishing point is based upon a local maximum of the intersections  $I_{ij}$ . In this scheme, each intersection has equal weight and

the algorithm cannot take advantage of the ground truth data.

In order to evaluate the algorithm's performance, the algorithm results are compared with the manually determined ground truth data. The arc distance  $\delta_3$  between each ground truth vanishing point and each estimated vanishing point is determined. When this distance is less than threshold  $t_5$ , the estimated vanishing point is determined to be correct (a true positive). If the distance  $\delta_3$  from an estimated vanishing point to each ground truth vanishing point is greater than  $t_5$ , then the estimated vanishing point is a false positive. Ground truth vanishing points not within a distance of  $t_5$  from any estimated vanishing point are classified as false negatives.

The vanishing point detection algorithm may be applied using any combination of the five features. Since five features have been described, there are a total of 32 combinations of those features. Five specific tests were performed on the testing set. By varying the threshold  $t_5$ , an idea of the algorithm's performance can be gathered. The results are shown in Fig. 20. Note that since the number of vanishing point estimates was fixed over the whole database (at an average of 1.82 vanishing points per image), an increase in the number of true positives corresponds identically to a decrease in the number of false positives.

The baseline algorithm where no features are utilized (i.e., each intersection is of equal importance as in Ref. [6]), is called Trial 0. This version had the poorest performance among the five variations tested. Fig. 20 shows the performance of the algorithm within an allowable error ( $t_5$ ) from 0.01 to 0.3 rad. For example, only about 30% of the vanishing points are identified within a tolerance of 0.1 radians between the Gaussian representation of the ground truth vanishing point and the algorithm result vanishing point.

Trial 24, where both of the color features ( $F_{ca}$  and  $F_{caA}$ ) are used, provides a slight improvement over the Trial 0 results.

Trial 4, where the intersection location feature ( $F_{loc}$ ) is used provides a large boost over the baseline results, as anticipated by the result of Eq. (14).

Trial 28, which combines the intersection location feature ( $F_{loc}$ ) used with Trial 4 and the two color features ( $F_{ca}$  and  $F_{caA}$ ) used with Trial 24 creates a slight improvement over the Trial 4 results.

Finally, Trial 31, which utilizes all five of the described features, performs best overall, providing a small performance increase over Trial 28, and a performance increase on the order of 10% when compared with Trial 0. This combination of features allows for the identification of 39% of the ground truth vanishing points within a tolerance  $t_5 = 0.1$ , a significant improvement in performance over Trial 0, which uses no features.

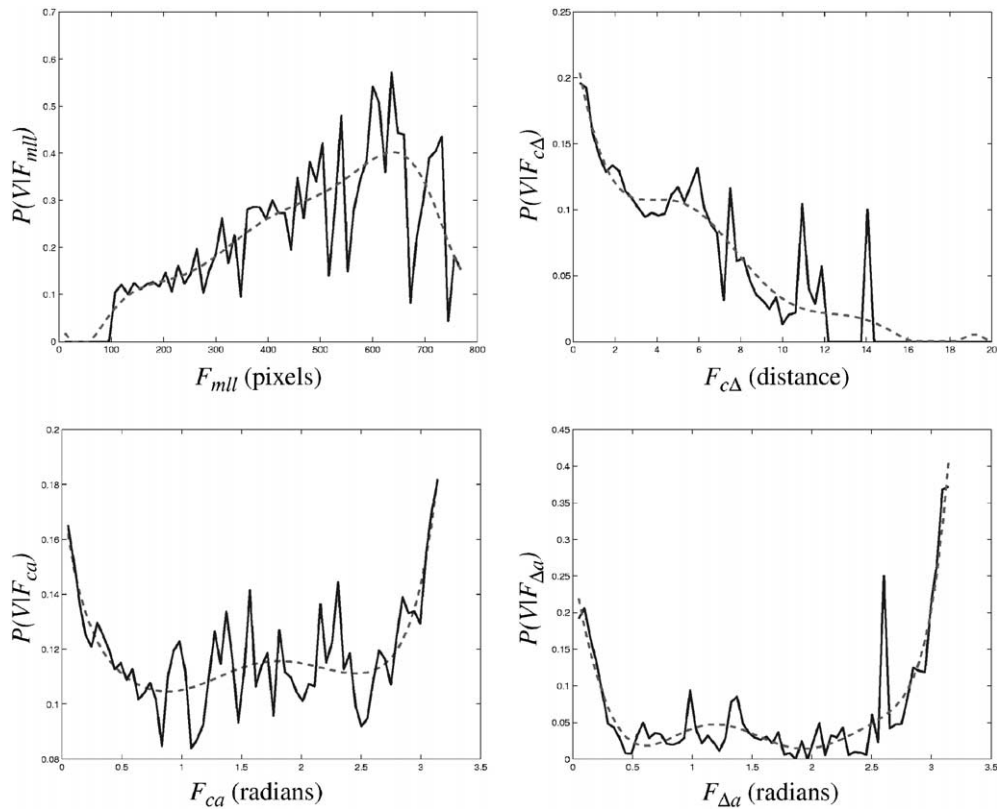


Fig. 19. The probabilities derived from the ground truth of the 266 images of the JBJL database. In general, these probabilities appear similar (in regions where there are a significant number of intersections, as indicated by the histograms) to the ground truth of the 86 image of the TSD database, shown in Figs. 11–14. However, the probability  $P(V|F_{ca})$  for the JBJL database does appear different from the  $P(V|F_{ca})$  for the TSD database in the region of small  $F_{ca}$ . This may indicate that the  $F_{ca}$  feature is not a reliable feature across different databases.

The number of false positives may be dramatically reduced by placing a threshold on the score  $S$  associated with each detected vanishing point. True positive vanishing point detections tend to have high scores  $S$ . For example, any detected vanishing point with a score  $S$  less than a threshold may be ignored. An operating curve using this decision boundary for trial 31 where  $t_5 = 0.1$  is shown in Fig. 21. About 35% of all vanishing points may be detected with 0.5 false positives per image.

## 8. Conclusions

Vanishing point ground truth data has been established for 352 images. Slightly more than half of these images contain one or more significant vanishing points. As distance from the center of the image increases, vanishing points are more likely to occur near an axis of the image.

A Bayesian probability based vanishing point detection algorithm using multiple features with ground truth training is presented. The vanishing points of 352 images were manually identified to create ground truth data. Each intersection is assigned a probability of being coincident with a ground truth vanishing point, based upon conditional probabilities of a number of features. The results of this ground truth based algorithm are demonstrated to be superior to the results of a similar algorithm where each intersection is considered to be of equal value. About 35% of all vanishing points may be detected with 0.5 false positives per image.

## 9. Summary

Perspective projection governs the mapping of the three-dimensional scene to the two-dimensional image. The perspective projection of lines that are parallel in a three-dimensional scene meet at a common vanishing point on the image plane.

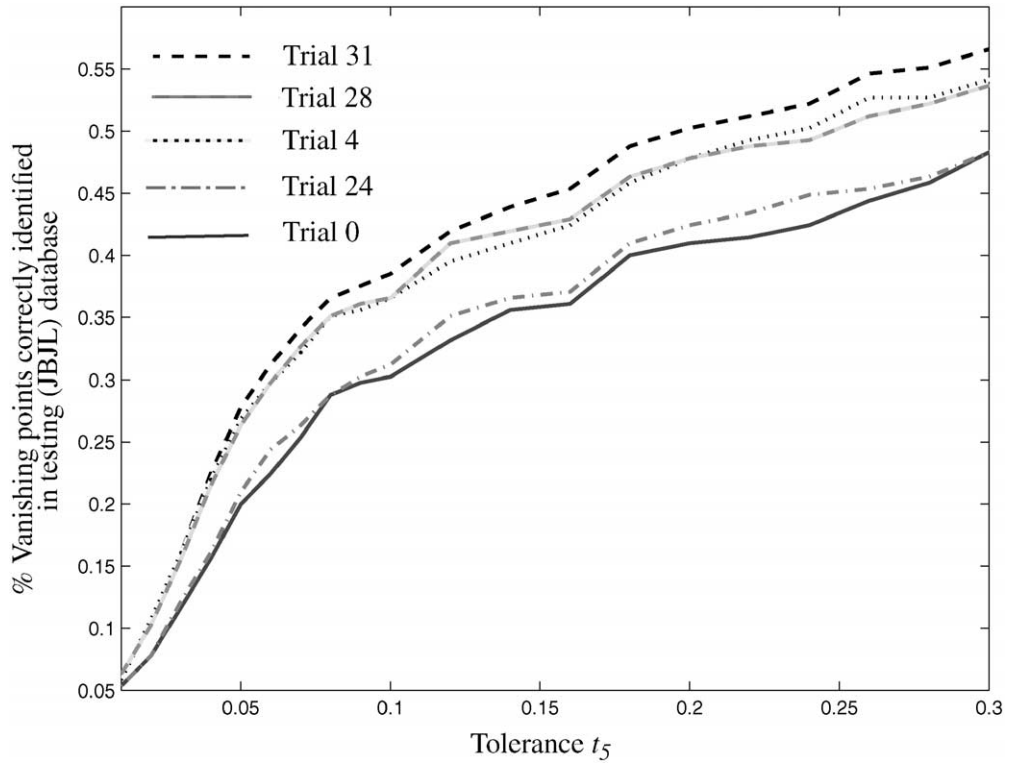


Fig. 20. Results of Bayesian based automatic vanishing point detection. Note that the baseline of using no features is similar to the method of Magee and Aggarwal [6].

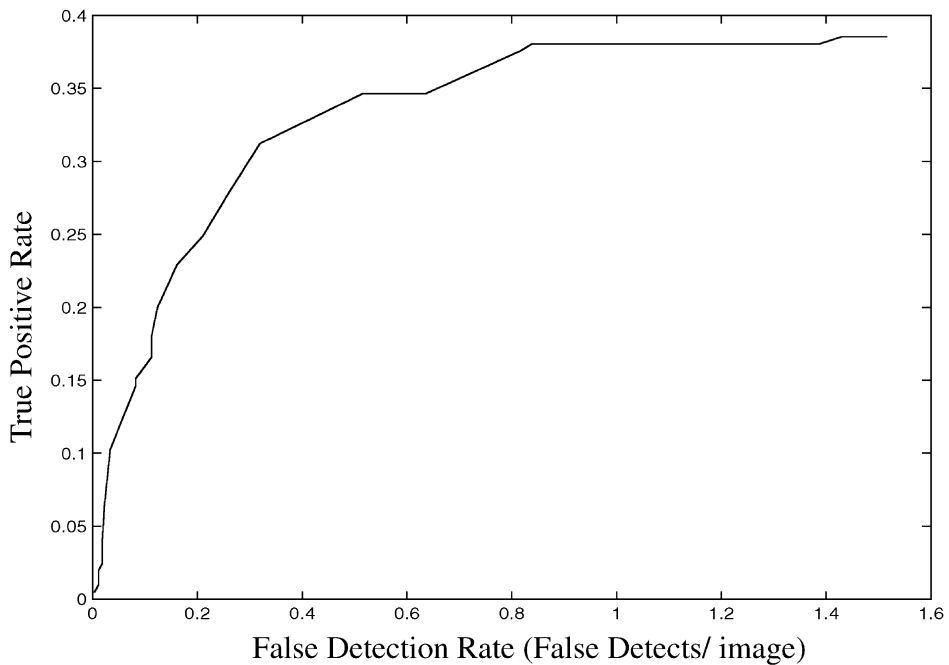


Fig. 21. The ROC (on the 266 images in the JBJL test set) for the automatic vanishing point detection algorithm. This curve is created by thresholding the scoring value  $S$  of the detected vanishing point (all detected vanishing points with  $S <$  the threshold are ignored.) Also, it is assumed that the detected vanishing point must be within 0.10 rad of the ground truth vanishing point to be detected as a true positive. The vertical axis is the true positive rate and the horizontal axis is the average number of false positives per image, when the algorithm is configured to output a maximum of two estimates per image.



Automatic vanishing point detection schemes rely on line segment detection followed by a means to determine the intersections of these line segments. The vanishing point is determined to be at the point with the most intersections. The prior art methods generally weight all features and classifiers equally, although there have been attempts to compensate for the bias of intersections occurring near the image center and the uncertainty of the line segment accuracy. However, these compensations did not rely on ground truth data for the determination of an appropriate level of compensation.

A Bayesian probability-based vanishing point detection algorithm is presented which introduces the use of multiple features and training with ground truth data to determine vanishing point locations. The vanishing points of 352 images were manually identified to create ground truth data. Each intersection is assigned a probability of being coincident with a ground truth vanishing point, based upon conditional probabilities of a number of features. The results of this algorithm are demonstrated to be superior to the results of a similar algorithm where each intersection is considered to be of equal importance. The advantage of this algorithm is that multiple features derived from ground truth training are used to determine vanishing point location.

## References

- [1] D.H. Ballard, C.M. Brown, *Computer Vision*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1982.
- [2] K. Kanatani, *Geometric Computation for Machine Vision*, Oxford University Press, Oxford, 1993.

- [3] B. Ernst, *The Magic Mirror of M.C. Escher*, Random House, New York, 1995.
- [4] S. Barnard, Interpreting perspective images, *Artif. Intell.* 21 (1983) 435–462.
- [5] D. Hilbert, S. Cohn-Vossen, *Geometry and the Imagination*, Chelsea, New York, 1952.
- [6] M. Magee, J. Aggarwal, Determining vanishing points from perspective images, *Comput. Vision, Graphics, Image Process.* 26 (1984) 256–267.
- [7] B. Brillault-O'Mahoney, New method for vanishing point detection, *Comput. Vision, Graphics, Image Process.: Image Understanding* 54 (2) (1991) 289–300.
- [8] L. Quan, R. Mohr, Determining perspective structures using hierarchical Hough transform, *Pattern Recognit. Lett.* 9 (4) (1989) 279–286.
- [9] R. Collins, R. Weiss, Vanishing point calculation as a statistical inference on the unit sphere, *Proceedings of the Third International Conference on Computer Vision*, 1990, pp. 400–403.
- [10] J. Shufelt, Performance evaluation and analysis of vanishing point detection techniques, *IEEE Trans. PAMI* 21 (3) (1999) 282–288.
- [11] E. Lutton, H. Maitre, J. Lopez-Krahe, Contribution to the determination of vanishing points using Hough transform, *IEEE Trans. PAMI* 16 (4) (1994) 430–438.
- [12] J. Burns, A. Hanson, E. Riseman, Extracting straight lines, *IEEE Trans. PAMI* 8 (4) (1986) 425–455.
- [13] P. Hough, Method and means for recognizing complex patterns, US Patent No. 3069654, 1962.
- [14] P. Kahn, L. Kitchen, E. Riseman, A fast line finder for vision-guided robot navigation, *IEEE Trans. PAMI* 12 (11) (1990) 1098–1102.
- [15] W. Yu, G. Chu, M. Chung, A robust line extraction method by unsupervised line clustering, *Pattern Recognit.* 32 (1999) 529–546.
- [16] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufman Publishers, Inc., San Francisco, 1988.

**About the Author**—ANDREW C. GALLAGHER was born in Erie, Pennsylvania, on January 13, 1974. He received his B.S.E.E. in electrical engineering in 1996 from Geneva College in Beaver Falls, Pennsylvania, and the M.S. degree in electrical engineering from Rochester Institute of Technology in Rochester, New York, in May 2000. Since 1996, he has worked as a research scientist for the Eastman Kodak Company in Rochester, NY in the Imaging Science Technology Laboratory. Current research interests include image enhancement, pattern recognition, and image understanding.