Pattern Recognition Letters 32 (2011) 1691–1700

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Simultaneous estimation of vanishing points and their converging lines using the EM algorithm

Marcos Nieto [a,*], Luis Salgado [b]

[a] Vicomtech-ik4, Mikeletegi Pasealekua 57, 20009, San Sebastián, Spain
[b] Grupo de Tratamiento de Imágenes, Universidad Politécnica de Madrid, Madrid, Spain

ARTICLE INFO

Article history:
Received 6 September 2010
Available online 22 August 2011
Communicated by B. Kamgar-Parsi

Keywords:
Vanishing points
Expectation–Maximisation
Projective plane
Mixture models
Non-linear optimisation

ABSTRACT

This paper introduces a new method for the simultaneous computation of sets of lines meeting at multiple vanishing points through the use of the Expectation–Maximisation (EM) algorithm. The proposed method is based on the formulation of novel error functions in the projective plane between lines and points which involve the use of non-linear optimisation procedures that allow to treat equally finite and infinite vanishing points. These functions are included into the EM framework, which handles the multi-modality of the problem naturally with mixture models. Results show that the proposed method of joint computation of vanishing points and lines converging at such points enhances the robustness and the accuracy of locating these points.

1. Introduction

Vanishing points are elements of great interest in the computer vision field. They are the main source of information about structured environments in images. A vanishing point is a point in the image to which parallel lines of the scenario appear to converge. Their importance lies in their correspondence to three-dimensional directions in the space. Hence, they can be used to retrieve information about the camera projection parameters or the geometry of the scene. There are a number of applications that can take advantage of the computation of these points, such as plane rectification (Liebowitz and Zisserman, 1998), camera calibration (Wang et al., 2009), single view metrology (Criminisi et al., 2000), compass estimation (Coughlan and Yuille, 1999), and others. Besides, some of these applications require the computation of lines meeting at vanishing points. For instance, Schaffalitzky and Zisserman (2000) have shown that the detection of imaged lines meeting at a common vanishing point, that actually correspond to equally spaced parallel lines in the scene, can be used to retrieve the line at the infinity. Other authors, like Trinh and Jo (2006), have used lines meeting at vanishing points to estimate the volume of buildings, or to estimate the geometry of road lanes in traffic applications like Rasmussen (2004) or Lai and Yung (2000). The estimation of vanishing points in images has been studied by the computer vision research community for decades, yielding a very significant number of noteworthy works (Almansa et al., 2003; Barnard, 1983; Košecká and Zhang, 2003; Liebowitz, 2001; Lutton et al., 1994; McLean and Koyyuri, 1995; Pflugfelder, 2008; Schindler and Dellaert, 2004). Although most of these works compute vanishing points through optimisation procedures, lines are typically obtained with additional processing steps such as the Hough transform. Therefore these works do not offer a closed solution for both problems and the combination of different nature methods does not offer the robustness and reliability of a joint optimisation procedure. In this paper a new method for calculating multiple vanishing points is introduced based on the EM algorithm. The major contribution of this work is the definition of a closed solution in the projective plane for the simultaneous estimation of vanishing points and the most significant lines that converge in those vanishing points by means of the introduction of a mixture model into the EM algorithm. The estimation of these elements in a single optimisation framework offers excellent estimation results, in terms of accuracy and robustness, improving the results of approaches that do not consider lines within their models. This paper is structured as follows: Section 2 summarises the literature related to vanishing point estimation to clarify the contribution of our work. Section 3 defines the proposed likelihood models as well as the required data calibration and normalisation stages. Section 4 describes the proposed steps of the EM algorithm, and Section 5 presents a performance evaluation and the obtained results.

2. Related work

The following subsections describe and classify the most relevant vanishing point estimation methods according to their workspace,

* Corresponding author. Tel.: +34 943 30 92 30; fax: +34 943 30 92 93.
E-mail addresses: mnieto@vicomtech.org (M. Nieto), L.Salgado@gti.ssr.upm.es (L. Salgado).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.07.018

the estimation technique used and the number of detected vanishing points. Table 1 summarises the most relevant properties of these methods.

2.1. Workspace

Vanishing points can be described mathematically in different ways. On the one hand, points are 2D image entities but on the other hand they correspond to 3D space directions. Therefore, we carry out a first classification of vanishing point detection techniques according to the workspace they use and associated parameterisation of vanishing points. Some authors have chosen the image plane itself as workspace such as Caprile and Torre (1990), McLean and Koyyuri (1995), Sekita (1994) and more recently Minagawa et al. (2000) and Suttorp and Bücher (2006). For instance, Suttorp and Bücher (2006) estimates the position of the main vanishing point under the assumption that in road scenes it typically lies inside the limits of the image and, therefore, the image plane can be then used as the workspace. Not restricted to one single vanishing point, other authors such as McLean and Koyyuri (1995), Sekita (1994) or Minagawa et al. (2000) define error measurements on the image plane to optimally estimate the position of several vanishing points. Specifically, Minagawa et al. (2000) propose as error measurement the normal distance between the vanishing points and the lines that contain the line segments. Barnard (1983) proposed to project the image plane, which is by definition an unbounded space, on a unit sphere centered at the optical center of the camera. The target is to work with a bounded space on which operations are simplified and thus finite and infinite vanishing points are treated equally. Most of subsequent works (Magee and Aggarwal, 1984; Quan and Mohr, 1989; Lutton et al., 1994) made explicit use of the unit sphere using its surface, properly tessellated, as the accumulation space where to search for maxima that determine the presence of dominant vanishing points. As an alternative to the image plane, and as a mathematical formalisation of the unit sphere parameterisation, other works emerged that use the projective plane to analytically handle infinite points in a natural way (Liebowitz, 2001; Pflugfelder, 2008; Rother, 2000). Analogously to the unit sphere, this search requires to augment the dimension of vanishing points by adopting the use of homogeneous coordinates. These works overcome the drawbacks of accumulation processes on the unit sphere by proposing error functions between vanishing points and image features that do not depend on the position of the vanishing point, and also applying optimisation techniques such as RANSAC or EM (Košecká and Zhang, 2003; Antone and Teller, 2000). Other authors proposed alternative transformed spaces to handle vanishing points at infinity (Cantoni et al., 2001; Seo et al., 2006; Tuytelaars et al., 1998). Tuytelaars et al. (1998) defined an intelligent partition of the parametric line space (computed as the Hough transform) by considering three bounded sub-spaces that were used to detect vanishing points and also vanishing lines. Similar approaches were proposed by Almansa et al. (2003) and Seo et al. (2006). The work by Cantoni et al. (2001) focused on the polar space obtained as the polar Hough transform of the points on the image. In this space, vanishing points at infinity are represented as sinusoids which are estimated through linear least squares optimisation.

2.2. Type of approach

Vanishing point estimation strategies are typically composed of two fundamental steps named clustering and estimation. The former classifies the image features into groups, or clusters, that share a common vanishing point; the latter considers the information of the cluster and estimates the position of the vanishing point. Many authors have converged to the use of iterative joint strategies, which contain the grouping and estimation stages as an alternating process. Specifically, the most used techniques are those based on robust estimation tools, such as RANSAC or its variants, and optimisation strategies based on mixture models using the EM algorithm. The methods based on RANSAC (RANdom Sampling And Consensus) (Pflugfelder, 2008; Rother, 2000) work iteratively selecting a minimal sample set of image features, which is used to compute a candidate vanishing point, and then finding the set of features that are coherent with it. Once this consensus set has been determined, the vanishing point can be re-estimated using the information of the elements of the whole consensus set. To determine more vanishing points, the features already used to compute the vanishing point are removed and the RANSAC iterations are restarted to search for a new vanishing point Pflugfelder (2008). The procedure is repeated until there are not enough features to determine new vanishing points. The EM (Expectation–Maximisation) algorithm has been shown to be a very powerful tool for recursive estimation and clusterisation. As opposed to RANSAC, EM allows to assign each line segment to different vanishing points with particular probabilities. The iterative process runs until the position of all vanishing points has been determined (Antone and Teller, 2000; Košecká and Zhang, 2003; Minagawa et al., 2000; Pflugfelder, 2008; Sekita, 1994). Typically, these methods are devised and used to refine results obtained with faster

Table 1
Classification of vanishing point estimation methods.

Ref.	Year	Workspace	Contribution	Num. Vps.	Lines
Barnard (1983)	1983	Sphere	Hough accumulation on sphere surface	3	No
Magee and Aggarwal (1984)	1984	Sphere	Search for clusters of intersections.	–	No
Quan and Mohr (1989)	1989	Sphere	Hierarchical Hough transform	–	No
Caprile and Torre (1990)	1990	Image plane	Vps. matching	–	No
Lutton et al. (1994)	1994	Sphere	Semi-regular sphere quantification	3 orth.	No
McLean and Koyyuri (1995)	1995	Image plane	Cluster and estimate approach	1	No
Tuytelaars et al. (1998)	1998	Hough	Cascaded Hough transform	N and Vls.	Yes
Antone and Teller (2000)	2000	Sphere	EM algorithm, angle error model	N	No
Minagawa et al. (2000)	2000	Image plane	EM algorithm for lines and Vps. and Vl.	N and Vl.	Yes
Ribeiro and Hancock (2000)	2000	Sphere	Spectral voting and search for maxima	N	No
Rother (2000)	2000	Image plane	Mid-point error model, unbounded search	3	No
Cantoni et al. (2001)	2001	Hough/image plane	Voting and least squares sinusoid	1	No
Liebowitz (2001)	2001	Projective plane	Optimal MLE solution for LSegs.	1	No
Košecká and Zhang (2003)	2003	Unc. Sphere	EM algorithm, outliers control and refinement	N	No
Almansa et al. (2003)	2003	Image plane	Equiprobable vanishing regions	N	No
Schindler and Dellaert (2004)	2004	Projective plane	EM algorithm, Atlanta world	N	No
Seo et al. (2006)	2006	ICIS	Inverted Coordinate Image Space	N	No
Pflugfelder (2008)	2008	Projective plane	Online scheme with EM algorithm	N	No

LSegs.: Line segments; Vps.: Vanishing points; Vl.: Vanishing line; -: Unknown or unspecified; Unc.: Uncalibrated.

approaches such as RANSAC, thus providing very accurate results although at the cost of significantly increasing the computational requirements. The work by Schindler and Dellaert (2004) is noteworthy as it uses the EM algorithm with on–off mixture models for edge points on the projective plane. Additionally, as opposed to many works that search for three vanishing points that correspond to the three orthogonal directions of the Euclidean space (Lutton et al., 1994; Quan and Mohr, 1989; Rother, 2000) (typically known as Manhattan worlds (Coughlan and Yuille, 1999)), this work does not require additional information from the scenario (known as Atlanta world).

2.3. Number of vanishing points and lines

The number of vanishing points obtained by each approach is strongly related to their use. For instance, methods that compute vanishing points for camera autocalibration need at least three vanishing points, while those for plane rectification just require two. There are methods that fit models assuming the existence of a single vanishing point (Cantoni et al., 2001; Liebowitz, 2001; McLean and Koyyuri, 1995), although they can be used to obtain several vanishing points (i.e. the image features used to compute the first vanishing point are removed from the feature set to compute the next vanishing point). However, other approaches do actually consider multiple vanishing points in their models so that they are computed simultaneously (Antone and Teller, 2000; Košecká and Zhang, 2003; Minagawa et al., 2000; Schindler and Dellaert, 2004; Seo et al., 2006; Tuytelaars et al., 1998).

2.4. Comparison with other joint strategies

Nevertheless, only few works do actually assume that vanishing points are the intersection of dominant lines meeting at them and compute the parameters of these lines. For instance, Tuytelaars et al. (1998) proposed a cascade of Hough transforms to identify lines as dominant clusters of edge points, vanishing points as their intersection, and finally vanishing lines. The use of the EM algorithm for this purpose was introduced by Minagawa et al. (2000), which defined a solution in the image plane for the simultaneous estimation of clusters of lines and their vanishing points. Nevertheless, in contrast to our proposal, it can not treat properly vanishing points far from the image, and it proposes a limited model to obtain a linear solution: only the position of the edge points is modeled within the EM algorithm and no solution to handle outliers is considered.

3. Projective-plane solution

This section describes in detail the workspace of the proposed approach, including concepts related to the projective plane such as data normalisation and likelihood models definition. These concepts are further used in the description of the proposed EM algorithm described in Section 4.

3.1. Image features

The input image information required for vanishing point estimation can be any feature with orientation, such as gradient-pixels (pixels with enough gradient magnitude) or line segments. These features are equivalent since they can be both expressed as a line in homogeneous coordinates (this concept is illustrated in Fig. 1). Let $\boldsymbol{r} = (x, y, 1)^\top$ be the homogeneous coordinates of a point of the image plane with significant gradient magnitude defined by the components of the gradient vector $\boldsymbol{g} = (g_x, g_y)^\top$. Any such gradient-pixel can be expressed as a line $\boldsymbol{l} = \boldsymbol{r} \times (\boldsymbol{r} + \lambda \boldsymbol{m})$, where $\boldsymbol{m}$ is the normal to the gradient vector, and λ is any real number. Besides, a line segment is defined by a pair of end-points $\{\boldsymbol{a}, \boldsymbol{b}\}$ such that they define the line $\boldsymbol{l} = \boldsymbol{a} \times \boldsymbol{b} = (l_1, l_2, l_3)^\top$ in homogeneous coordinates. Its associated reference point is the mid-point $\boldsymbol{c} = \frac{\boldsymbol{a}+\boldsymbol{b}}{2}$. As a generalisation, let us consider from here on a data sample as the set $\boldsymbol{x} = \{\boldsymbol{r}, \boldsymbol{l}\}$, whose elements are derived either from a gradient-pixel or a line segment (in this latter case, let use a common notation for the reference point: $\boldsymbol{c} = \boldsymbol{r}$).

3.2. Data calibration and normalisation

Each data sample on the image plane defines a plane that passes through the optical center of the camera model. This plane is denominated "Edge Plane" by Antone and Teller (2000), although it is more frequently identified as the "Interpretation plane" as done by Pflugfelder (2008). This plane intersects a sphere centered as well at the optical center generating the so-called "great circles". The intersections between these circles correspond to the intersection of the corresponding lines on the image plane. Fig. 2 illustrates these concepts. An interesting property we get from these definitions is that a vanishing point can be treated as a 3D direction, instead of just as a 2D point on an image. The vanishing direction is defined by the vector that joins the optical center and the point on the surface of the sphere which is the intersection of two or more great circles that correspond to lines meeting at a common vanishing point. Vanishing points are represented in homogeneous coordinates as $\boldsymbol{v} = (v_1, v_2, v_3)^\top$ (Hartley and Zisserman, 2004). This representation allows handling projective relationships with linear matrix operations. For instance, the projection of a point in the space $\boldsymbol{X} = (X, Y, Z, 1)^\top$ into the image plane is:

$$\boldsymbol{x}_{\text{hom}} = K[I|\boldsymbol{0}]\boldsymbol{X} \tag{1}$$

where K is the camera calibration matrix, I is the identity 3×3 matrix, and $\boldsymbol{0}$ is a column 3×1 vector of zeros. The image coordinates of this projected point are then obtained by dividing $\boldsymbol{x}_{\text{hom}}$ by its third coordinate, obtaining the inhomogeneous coordinates of the point in the image plane $\boldsymbol{x}_{\text{inh}} = \left(f\frac{X}{Z}, f\frac{Y}{Z}\right)^\top$. One dimension is lost projecting a space point onto the image plane since any point lying on the ray that joins $\boldsymbol{X}$ with the optical center $\boldsymbol{C}$ is projected in the same image point (Hartley and Zisserman, 2004). From a geometric point of view, if the camera calibration matrix is known, the aforementioned ray can be retrieved from any 2D image point. For the case

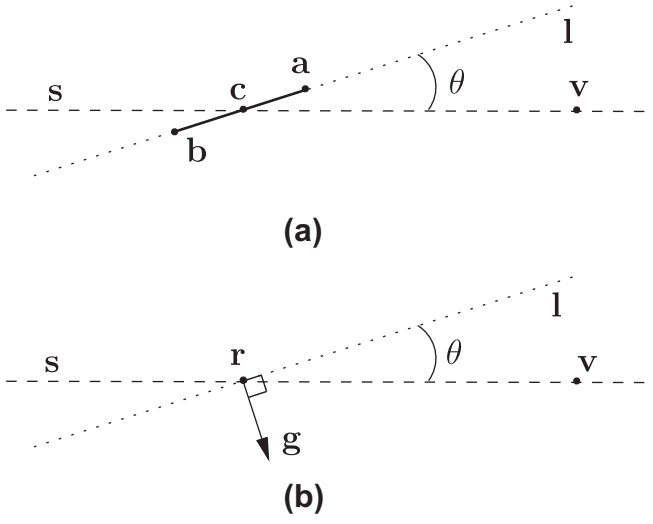

Fig. 1. Relative orientation θ between the orientation of the measured feature and the line that joins the vanishing point $\boldsymbol{v}$ and the feature ((a) line segments and (b) edge-points). When the vanishing point lies on $\boldsymbol{l}$ the error is minimum.

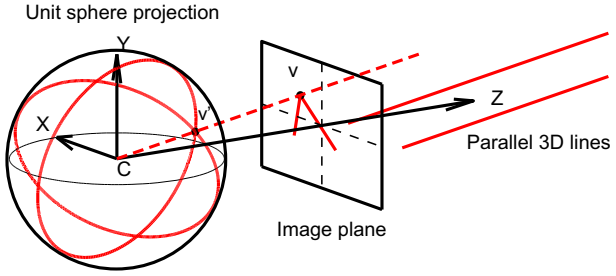

Fig. 2. Two parallel 3D lines are projected onto the image plane meeting at the vanishing point $\boldsymbol{v}$. Its calibrated and normalised version $\boldsymbol{v}'$ can be seen either as the ray joining the optical center C and $\boldsymbol{v}$ or as the intersection of the two great circles defined by the projected lines.

of a vanishing point, this ray actually corresponds to the 3D direction of the parallel lines in the 3D world. Indeed, when parallel lines are projected onto the image they converge at a vanishing point. This idea is illustrated in Fig. 2. To explain this concept, let us suppose that there are two parallel lines in the 3D world whose common director vector is $\boldsymbol{D} = (X, Y, Z)^\top$. The intersection of these lines is at some point in the infinity that we can represent in homogeneous coordinates as $\boldsymbol{V} = (X, Y, Z, 0)^\top$. This point is projected as the vanishing point $\boldsymbol{v} = K[I|\boldsymbol{0}]\boldsymbol{V}$. If we left-multiply both members of this expression by K^{-1} we obtain $K^{-1}\boldsymbol{v} = [I|\boldsymbol{0}]\boldsymbol{V} = (X, Y, Z)^\top$, which renders a new 3D vector $\boldsymbol{v}' = K^{-1}\boldsymbol{v}$. We will refer to this transformation as "data calibration" as other authors do (Košecká and Zhang (2003)). Hence, the calibration of a vanishing point yields the actual 3D direction of the parallel lines whose projections onto the image plane converge at it. This transformation can also be understood as the transformation of the 2D coordinates of the image features (points and lines) into the 3D coordinates under the camera coordinate system. Data calibration is carried out in the following way: points are normalised with K^{-1} as $\boldsymbol{x}' = K^{-1}\boldsymbol{x}$ and lines with $K^\top$ as $\boldsymbol{l}' = K^\top\boldsymbol{l}$. Besides, to be able to represent these elements on the unit sphere, an additional scale normalisation must be applied such that the norm of the 3D vectors is the unity. When the camera calibration is not available working with the homogeneous vector $\mathbf{v}$ entails an implicit distortion in the angular information. For instance, the cosine of the angle between two uncalibrated directions is dependent on an unknown distortion given by $\omega = K^{-\top}K^{-1}$:

$$\cos\alpha = \frac{\boldsymbol{v}_1'^\top \boldsymbol{v}_2'}{\|\boldsymbol{v}_1'\|\|\boldsymbol{v}_2'\|} = \frac{\boldsymbol{v}_1^\top \omega \boldsymbol{v}_2}{\sqrt{\boldsymbol{v}_1^\top \omega \boldsymbol{v}_1}\sqrt{\boldsymbol{v}_2^\top \omega \boldsymbol{v}_2}} \tag{2}$$

Nevertheless, the camera calibration matrix can be easily approximated by a default projective transformation based on the image dimensions, which reduces the aforementioned inherent distortion, as done by some authors (Košecká and Zhang (2003)):

$$\widetilde{K} = \begin{pmatrix} W & 0 & W/2 \\ 0 & H & H/2 \\ 0 & 0 & 1 \end{pmatrix} \tag{3}$$

where W and H are the width and height of the image in pixels, respectively.

3.3. Likelihood models

Given a data set $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^N$, vanishing point estimation techniques typically focus on only the optimisation of the parameters of the vanishing point $\boldsymbol{v}$ (or the set of vanishing points). In our approach the dimensionality of the problem is augmented since the set of parameters to be optimised is now given by a set of vanishing points $\{\boldsymbol{v}_k\}_{k=1}^V$ and a set of supporting or main lines that pass through them $\{\boldsymbol{s}_{jk}\}_{j=1,k=1}^{M,V}$. Let us define the relationship between data samples and the set of supporting lines. The likelihood model is governed by two equations that link each data sample $\boldsymbol{x}_i = \{\boldsymbol{r}_i, \boldsymbol{l}_i\}$ with lines meeting at a common vanishing point (the subindex k has been removed for the sake of clarity), $\{\boldsymbol{s}_j\}_{j=1}^M$, as:

$$p(\boldsymbol{x}_i|\boldsymbol{s}_j) = p(\boldsymbol{r}_i|\boldsymbol{s}_j)p(\boldsymbol{l}_i|\boldsymbol{s}_j) \tag{4}$$

where the point-line and line-line distances are given by three-vector dot products:

$$p(\boldsymbol{r}_i|\boldsymbol{s}_j) \propto \exp\left(-\frac{1}{2\sigma_{\rho j}^2}\left(\boldsymbol{r}_i^\top \boldsymbol{s}_j\right)^2\right) \tag{5}$$

$$p(\boldsymbol{l}_i|\boldsymbol{s}_j) \propto \exp\left(-\frac{1}{2\sigma_{\phi j}^2}\left(1 - \left(\boldsymbol{l}_i^\top \boldsymbol{s}_j\right)^2\right)\right) \tag{6}$$

Fig. 3 illustrates these functions. The point-line distance, $\boldsymbol{r}_i^\top \boldsymbol{s}_j$ in (5) assumes that vectors are normalised such that $\|\boldsymbol{r}_i\| = \|\boldsymbol{s}_j\| = 1$. Therefore, this distance corresponds to the cosine of the angle defined by the vector that joins the optical center C with the point $\boldsymbol{r}_i$ and the normal vector of the plane defined by the great circle that corresponds to the line $\boldsymbol{s}_j$. This way, if the point belongs to the line on the image plane, its associated 3D vector is orthogonal to the normal vector of the interpretation plane and (5) is maximum. The line–line distance, considering also normalised vectors, is related to the sine of the angle between the interpretation planes corresponding to $\boldsymbol{l}_i$ and $\boldsymbol{s}_j$. Particularly, given that $\cos(\theta) = \left(\boldsymbol{l}_i^\top \boldsymbol{s}_j\right)$ and $\sin^2(\theta) = (1 - \cos^2(\theta))$ then $\sin^2(\theta) = 1 - \left(\boldsymbol{l}_i^\top \boldsymbol{s}_j\right)^2$. Therefore, if the lines are equal, these normal vectors are coincident and the sine of their angle is zero so that the likelihood as in (6) is maximum. The result is a likelihood function defined by a mixture model of M supporting lines grouped in V sets that correspond to different vanishing points in the scene plus an additional component that handles outliers:

$$p(\boldsymbol{x}_i|\Theta, \Omega) = \sum_{k=1}^{V}\sum_{j=1}^{M} \omega_{jk} p(\boldsymbol{x}_i|\theta_{jk}) + \omega_{\text{out}} p_{\text{out}}(\boldsymbol{x}_i) \tag{7}$$

where $\Theta = \{\theta_{jk}\}_{j=1,k=1}^{M,V}$ and $\theta_{jk} = \{\boldsymbol{s}_{jk}, \sigma_{\phi jk}, \sigma_{\rho jk}\}$, and the mixture model components are $\Omega = \left\{\{\omega_{jk}\}_{j=1,k=1}^{M,V}, \omega_{\text{out}}\right\}$. This expression already includes a component of the mixture devoted to outliers such that the weights sum to one:

$$\left(\sum_{k=1}^{V}\sum_{j=1}^{M} \omega_{jk}\right) + \omega_{\text{out}} = 1 \tag{8}$$

The outlier distribution can be defined as a normal distribution with fixed large variance which approximately behaves as a uniform distribution. The use of such a normal distribution has been already addressed by other authors like Košecká and Zhang (2003). The reason of choosing this approximation instead of the uniform itself is because the EM algorithm for a mixture of Gaussians is much simpler to compute. Note that, although the standard deviation of the supporting lines could be different according to the indexes j and k, we will consider, for the sake of simplicity, that there is a common value $\sigma_{\phi jk} = \sigma_\phi$ and $\sigma_{\rho jk} = \sigma_\rho$ $\forall j = 1 \ldots M$, $k = 1 \ldots V$. The presence of multiple i.i.d. data samples yields the incomplete likelihood function $p(\mathcal{X}|\Theta) = \prod_{i=1}^N p(\boldsymbol{x}_i|\Theta, \Omega)$, and the log-likelihood function which includes the summation on clusters of lines indexed by k:

$$\log \mathcal{L}(\Theta|\mathcal{X}) = \sum_{i=1}^{N} \log\left(\sum_{k=1}^{V}\sum_{j=1}^{M} \omega_{jk} p(\boldsymbol{x}_i|\theta_{jk}) + \omega_{\text{out}} p_{\text{out}}(\boldsymbol{x}_i)\right) \tag{9}$$

Given the definition of the likelihood model for data samples and supporting lines let us now consider the relationship between vanishing points and data samples. For this purpose, we could use any error function, like the calibrated point-line distance $\boldsymbol{l}_i^\top \boldsymbol{v}$. This

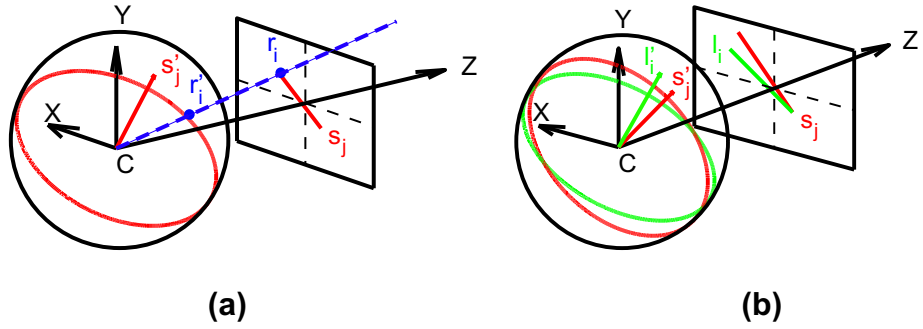

Fig. 3. Point–line and line–line distances in the projective plane. The unit sphere visualisation helps to understand the proposed distances as relationships between vectors joining the optical center, C, and points in the surface of the sphere. For clarity, in this figure, calibrated coordinates are depicted with a tilde.

distance is equivalent to the point-line distance $\boldsymbol{s}_j^\top \boldsymbol{r}_i$ that was illustrated in Fig. 3. Nevertheless, it has been shown Nieto and Salgado (2010) that more accurate results can be found using the orientation deviation between $\boldsymbol{l}_i$ and a line joining the vanishing point and a reference point, which is the mid-point of the line segment.

$$d(\boldsymbol{x}, \boldsymbol{v}) = |\sin(\theta)| = \frac{|-l_2 s_1 + l_1 s_2|}{\sqrt{l_1^2 + l_2^2}\sqrt{s_1^2 + s_2^2}} \tag{10}$$

This distance represents the error between a vanishing point $\boldsymbol{v}$ in homogeneous coordinates and the orientation of the data sample $\boldsymbol{x}$. It is related to the absolute value of the sine of the angle θ between $\boldsymbol{l}$ and a line $\boldsymbol{s} = (s_1, s_2, s_3)^\top$ that joins $\boldsymbol{v}$ and $\boldsymbol{r}$ (as shown in Fig. 1) and that can be computed as $\boldsymbol{s} = \boldsymbol{v} \times \boldsymbol{r}$. It is important to realise that this error function is defined on uncalibrated coordinates. Hence, in this case, data has not to be calibrated. The likelihood of a data sample $\boldsymbol{x}_i$ to belong to a vanishing point $\boldsymbol{v}$ is given by

$$p(\boldsymbol{x}_i|\boldsymbol{v}) \propto \exp\left(-\frac{1}{2\sigma^2} d^2(\boldsymbol{x}_i, \boldsymbol{v})\right) \tag{11}$$

where $d(\boldsymbol{x}_i, \boldsymbol{v})$ is defined as in Eq. (10).

4. EM algorithm

The EM procedure can be described once we have defined the likelihood models. The process has to be done gradually, i.e. with several M-steps and, for this case, also several E-steps since there is not a single expression to update at once all the model parameters. The following paragraphs summarise the proposed way to proceed, which updates in the first M-step the position of the vanishing points $\{\boldsymbol{v}_k\}_{k=1}^{V}$ and in the second one the parameters of the supporting lines $\{\boldsymbol{s}_{jk}\}_{j=1,k=1}^{M,V}$. Although it is possible to proceed the other way round, i.e. updating first the parameters of the supporting lines and then the position of the vanishing points, empirical tests have shown that this approach is more prone to errors. Updating the vanishing point position first allows the E-step to classify data samples according to the vanishing point they meet and then update the supporting lines.

4.1. E-step (1)

This step computes the conditional probabilities of the set of data samples with respect to the set of vanishing points and their corresponding supporting lines. Given the current best estimation of the parameters of the model Θ^* and Ω^* these probabilities are computed as

$$p(j, k|\boldsymbol{x}_i, \Theta^*) = \frac{\omega_{jk}^* p(\boldsymbol{x}_i|\theta_{jk}^*)}{\omega_{\text{out}} p_{\text{out}}(\boldsymbol{x}_i) + \sum_{k'j'=1}^{V,M} \omega_{j'k'}^* p(\boldsymbol{x}_i|\theta_{j'k'}^*)} \tag{12}$$

where $p(\boldsymbol{x}_i|\theta_{jk}^*)$ is computed as defined in (4). For the sake of simplicity, $p(j, k|\boldsymbol{x}_i, \Theta^*)$ will be denoted as γ_{ijk} from here on.

4.2. M-step (1)

Once the conditional probabilities have been computed, the first maximisation step estimates the position of the set of vanishing points minimising the selected error function. In case of using the error function in uncalibrated coordinates, defined in Eq. (10), the conditional probabilities γ_{ik} can be used as the scale factor so that the error function is redefined as $d'(\boldsymbol{x}, \boldsymbol{v}) = \gamma_{ik} d(\boldsymbol{x}, \boldsymbol{v})$. Note that the index j has disappeared from γ_{ijk} as for the estimation of each vanishing point the dimension corresponding to the supporting lines is marginalised. The set of data samples is classified according to γ_{ik}, i.e. according to the vanishing point they most likely belong. Then, for each vanishing point $\boldsymbol{v}_k$ and its corresponding subset of samples $\mathcal{X}_k$ the function to be minimised is

$$\log \mathcal{L}(\boldsymbol{v}_k|\mathcal{X}_k) = \sum_{\boldsymbol{x}_i \in \mathcal{X}_k} d^2(\boldsymbol{x}_i, \boldsymbol{v}_k) \tag{13}$$

which can be solved with the Levenberg–Marquardt algorithm that updates the vanishing point $\boldsymbol{v}_k$. The initialisation of this procedure can be random, although we have found better results if the initialisation is carried out using the calibrated point-line minimisation. This is a linear approach that can be solved in a least squares sense: as a linear optimisation problem, the least squares solution for each $\boldsymbol{v}_k$ is the eigenvector with smallest eigenvalue of the problem (Košecká and Zhang, 2003)

$$(L^\top \Gamma^\top \Gamma L)\boldsymbol{v}_k = \lambda \boldsymbol{v}_k \tag{14}$$

where L is a $N \times 3$ matrix, in which each row corresponds to the data sample $\boldsymbol{l}_i$ in row format, and Γ is a $N \times N$ diagonal matrix of weights that corresponds to the conditional probabilities γ_{ik}.

4.3. M-step (2)

After M-step (1), as the position of the vanishing points have been updated, it is required to re-compute the parameters of their supporting lines so that they meet at their new positions. This concept is illustrated in Fig. 4. The problem is stated for each supporting line $\boldsymbol{s}_{jk}$ as

$$\hat{\boldsymbol{s}}_{jk} = \underset{\boldsymbol{s}_{jk}^*}{\text{argmax}} \left(\boldsymbol{s}_{jk}^\top \boldsymbol{s}_{jk}^*\right)^2 \quad \text{subject to } \boldsymbol{s}_{jk}^\top \boldsymbol{v}_k = 0 \tag{15}$$

The solution to this problem can be found using iterative methods for non-linear minimisation with restrictions. For instance, a sequential quadratic programming (SQP) method has been used.

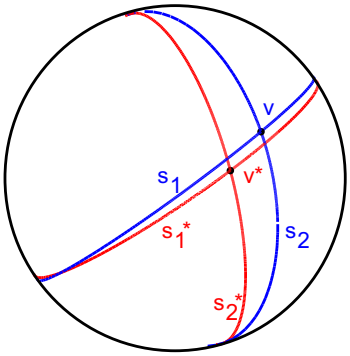

Fig. 4. Given the new position of the vanishing point, $\boldsymbol{v}$, the lines $\boldsymbol{s}_1^*$ and $\boldsymbol{s}_2^*$ are re-estimated yielding $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ such that they actually pass through $\boldsymbol{v}$ and the angular distances $\boldsymbol{s}_j^\top \boldsymbol{s}_j^*$ are minimised.

4.4. E-step (2)

After the re-estimation of the supporting lines the conditional probabilities have to be updated according to the new parameters of the supporting lines. Therefore, the values of γ_{ijk} are re-computed as in (12).

4.5. M-step (3)

Finally, using the updated conditional probabilities and the likelihood function defined in (9), the parameters of the set of supporting lines have to be updated. The problem to be solved for each line $\boldsymbol{s}_{jk}$ is defined as:

$$\hat{\boldsymbol{s}}_{jk} = \underset{\boldsymbol{s}_{jk}^*}{\operatorname{argmax}} \log \mathcal{L}(\Theta|\mathcal{X}) \tag{16}$$

subject to $\boldsymbol{s}_{jk}^\top \boldsymbol{v}_k = 0$ and $\|\boldsymbol{s}_{jk} = 1\|$. This problem can be again solved using iterative methods for non-linear minimisation with restrictions as in M-step (2).

5. Testing and discussion

In this section we present several tests that evaluate different aspects of the proposed method. We use the York Urban Data Base (YUDB) (Denis et al., 2008), which contains a set of 102 images of structured environments including ground truth vanishing points and line segments with available camera calibration information. First, we analyse the dependency of the proposed EM algorithm with respect to the initialisation of the vanishing points. Second, we compare two different approaches for automatic initialisation: a RANSAC-based method Nieto and Salgado (2010) and a variation of the Hough transform proposed by Almansa et al. (2003). Finally, we present the results of the execution of the proposed strategy on the set of images of the YUDB. For the tests, we have used two different input data sources: (i) the ground truth line segments of the YUDB, which contain no outliers (i.e. line segments that do not meet any vanishing point); and (ii) automatically computed line segments using the SSWMS algorithm Nieto et al. (2011), which renders relevant segments in the image, regardless if they meet or not at any vanishing point. This algorithm is fast (can be used in online applications) and yields accurate line segments while keeping a low false alarm ratio. Fig. 5 shows an example of the line segments computed using this method.

5.1. Initialisation analysis

The described EM method requires initialisation of the target parameters: the vanishing points and their corresponding supporting lines. As described in the literature Minagawa et al. (2000), EM

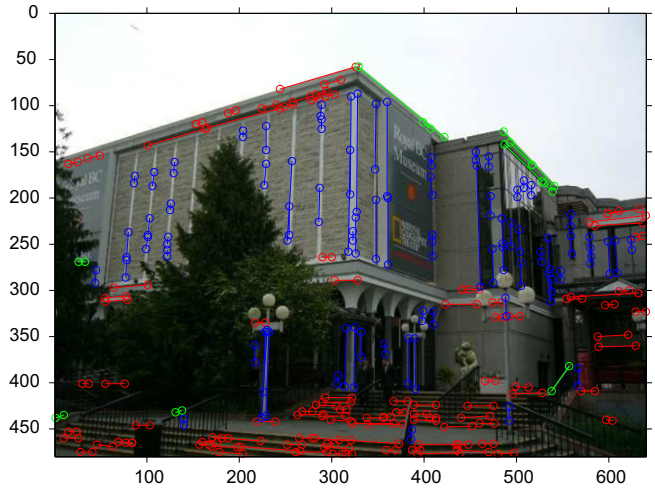

Fig. 5. Line segments obtained with the automatic line segment detector. The colour code indicates the classification of the segments according to the vanishing point they most likely belong to according to the RANSAC method.

based strategies are quite sensitive to the initialisation process: they are able to compensate errors in the initialised target parameters up to certain deviation from their actual values. Therefore, any initialisation procedure should take this constraint into account to reach the desired results. To evaluate the sensitivity of the proposed strategy we have executed the EM method for all the images of the YUDB using as initialisation vanishing points with increasing angular error in their position (since they can be treated as 3D vectors, the error between two vanishing points can be computed as an angle, as described in Eq. (2)). Angular errors between 1° and 15° have been used for the analysis, although it should be noticed that errors above 5° are in many cases too severe, and they are particularly relevant for vanishing points close to the image frame. The results of the test are illustrated in Fig. 6. In (a) the results correspond to the execution of the proposed method using ground truth line segments while (b) shows the results obtained using the SSWMS line segments. Each figure shows three curves, each one corresponding to one of the three vanishing points of the images according to the YUDB criterion Denis et al. (2008) (VP #1 is the most significant horizontal vanishing point, VP #2 is the vertical vanishing point, and VP #3 is the second most significant vanishing point). These curves depict the number of images for which the EM method has found the correct solution, i.e. reaching estimates of the vanishing points close to the ground truth. Using ground truth line segments, the proposed method shows excellent results (above 90%) for deviations up to 6°, with slightly better results for VP #2 as it is the vertical vanishing point and it is typically supported by more line segments than the other vanishing points (an example is shown in Fig. 7). As expected, the performance decreases when the SSWMS segments are used: the main reason is the presence of clutter or outliers, i.e. line segments not meeting at any of the ground truth vanishing points. When the initialisation error increases, the clutter can more easily drive the estimation to local minima and thus make the system fail to find the correct vanishing points. The figures show that the best performance corresponds to the estimation of the VP #2, while VP #3 is more sensitive to the clutter, as it is the vanishing point located typically near or inside the limits of the image frame.

5.2. Automatic initialisation of vanishing points

As noted in the previous subsection, the EM method needs initializations with relatively low error (up to 5° approximately).

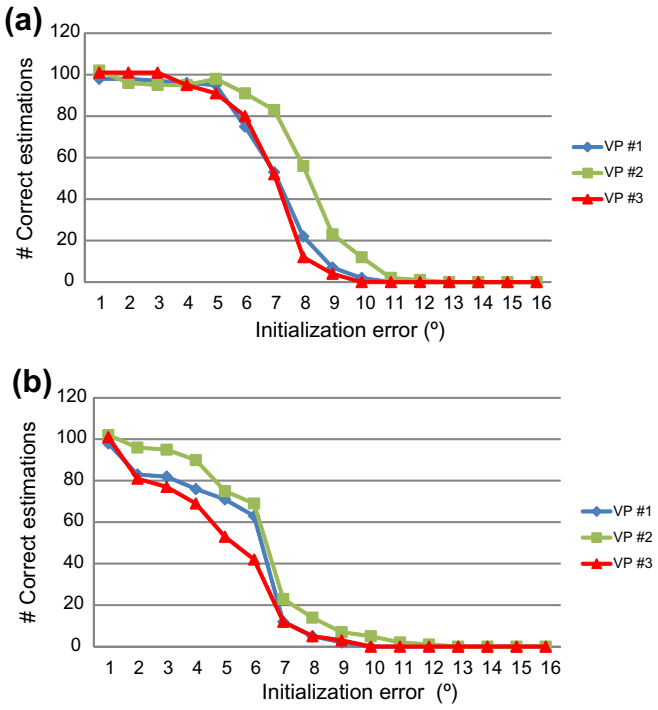

Fig. 6. Sensitivity to initialisation error: (a) using ground truth line segments (YUDB); (b) using automatically computed line segments with the SSWMS algorithm.

There are several methods in the literature that can generate these estimates. We have selected two, and we have compared their performance: the method by Almansa et al. (2003), which is a Hough-based algorithm with an intelligent partition of the image plane, and a method based on a variation of the RANSAC algorithm, the MSAC (M-estimator Sample And Consensus), which has been shown to provide excellent classification results (Nieto and Salgado, 2010). As mentioned by Almansa et al. (2003), their method worked on input data with a very reduced false alarm rate, which we have confirmed by using the available ground truth line segments of the YUDB. In Fig. 8 we illustrate how this method tessellates the image plane, then selects line segments according to their orientation and finds their vanishing point. In (a) we show the result using the ground truth line segments, achieving angular errors below 0.1° (with $d\theta = \frac{\pi}{1024}$ as defined by the authors). However, in a more realistic situation, with non quasi-error free line segments and outliers, the error increases significantly for the detected vanishing points and some miss-detections occur. In (b) we see how outliers affect the estimation, rising the error to 2.2° and 3.5° for VP #1 and VP #2, while VP #3 is not detected. This sensitivity to outliers and noise is what motivates our proposal of using the RANSAC-based approach. The MSAC algorithm proceeds iteratively in two main steps: (i) a minimal subset of data samples is randomly selected in order to generate a hypothesis of the vanishing point. In this case, the minimal set is composed of two data samples whose associated estimate is $\boldsymbol{v}^* = \boldsymbol{l}_i \times \boldsymbol{l}_j$; and (ii) the consensus set, $s(\boldsymbol{v}^*)$, i.e. the subset of data samples that are coherent with the hypothesis, is computed as $s(\boldsymbol{v}^*) \triangleq \{\boldsymbol{l}_i \in \mathcal{L} : d^2(\boldsymbol{l}_i, \boldsymbol{v}^*) \leqslant \delta\}$; where δ is a certain threshold governed by the statistics of the data. These two steps are repeated until the probability of finding a better consensus set is below a certain threshold. Here resides the difference between MSAC and RANSAC: while basic RANSAC ranks the consensus sets according to their cardinality, i.e. the number of elements of the set, the MSAC version ranks them according to a global cost function that gathers individual contributions of inliers and outliers. The total cost of a hypothesis in MSAC is given by $\mathcal{C}(\mathcal{L}, s(\boldsymbol{v}^*)) = \sum_{i=1}^{N} \rho(\boldsymbol{l}_i, \boldsymbol{v}^*)$, where $\rho(\boldsymbol{l}_i, \boldsymbol{v}^*)$ is defined as:

$$\rho(\boldsymbol{l}_i, \boldsymbol{v}^*) = \begin{cases} d^2(\boldsymbol{l}_i, \boldsymbol{v}^*), & d^2(\boldsymbol{l}_i, \boldsymbol{v}^*) \leqslant \delta \\ \delta, & \text{otherwise} \end{cases} \tag{17}$$

Fig. 9 shows the results of this method applied on the example image shown in Fig. 5 using, as in the case of the Hough-based method, the ground truth line segments and the automatically computed line segments. As can be observed in Fig. 9(a), analogously to the Hough-based method, MSAC achieves very accurate results for almost noise and outliers free input data: the estimation error ranges between 0.2° and 1°. However, in (b), we can see that MSAC is also able to compute vanishing points using the SSWMS line segments, achieving errors below 4°. In summary, the MSAC method shows to be robust in the presence of outliers and obtains average estimation errors of 3°. Therefore, MSAC can be safely used as initialisation for the EM method.

5.3. Initialisation of supporting lines

As the supporting lines must also be initialised, a line segment clustering strategy is carried out. For this purpose, the Hough transform can be used on the set of data samples $\mathcal{X} = \{\boldsymbol{r}_i, \boldsymbol{l}_i\}_{i=1}^{N}$ so that each point $\boldsymbol{r}_i$ is transformed into a sinusoid $\rho = x_i \sin\theta + y_i \cos\theta$. Then, the votes of data samples are accumulated on the quantised cells defined by the sinusoid. Each cell value is increased by the length of the line segment. Therefore, the votes of longer line segments are more important than those of shorter ones. Typically it is enough to search for two separated maxima that correspond to the two most significant clusters of line segments.

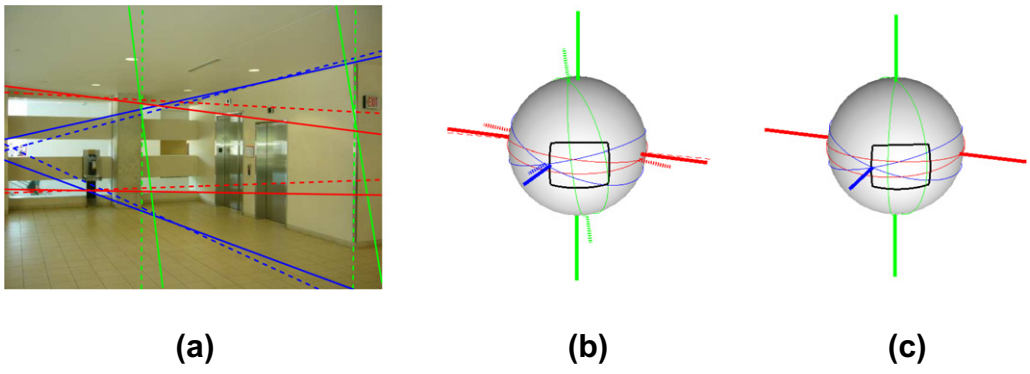

Fig. 7. Example initialisation of vanishing points with 5° angular error with respect to ground truth: (a) initialisation of lines (solid lines) and estimated lines after the application of the proposed EM method (dashed lines); (b) and (c) visualisation of the initialisation and estimation on the unit sphere. Solid axes represent ground truth vanishing points. Thin arcs on the sphere surface represent the projection of the supporting lines. Dashed axes represent the initialised position of the vanishing points.

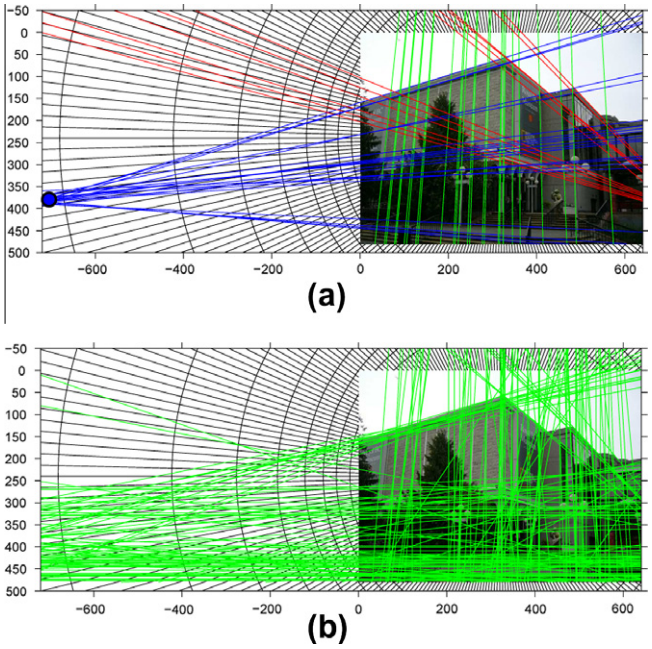

Fig. 8. The Hough-based method is significantly sensitive to the presence of outliers and noise in the input data set (segments): (a) results (for VP #1) of the method using the ground truth line segment set for an example image of the YUDB (the lines are coloured since the association between line segments and vanishing points is also available as ground truth); (b) using an automatic line segment detector makes the method fail to find vanishing points (mainly affected by the presence of outliers).

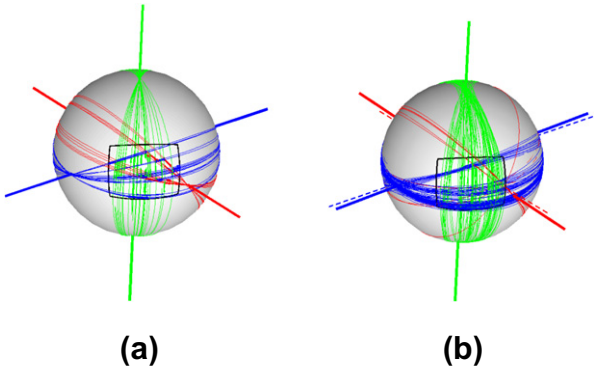

Fig. 9. Unit sphere visualisation of the classification of line segments according to the MSAC procedure and the obtained vanishing points (as thick axes): (a) using the ground truth line segments, the obtained vanishing points visually coincide with the ground truth vanishing points; (b) using the automatically extracted line segments, the obtained vanishing points are shown as dashed axes, which slightly differ from the ground truth vanishing points (solid axes).

5.4. Algorithm complexity

The use of an EM scheme, whose nature is recursive, makes the amount of required operations proportional to the size of the input data and the number of iterations of the algorithm. The complexity of the algorithm is $O(kN)$, where k is the number of iterations of the EM algorithm, and N is the size of the input data set (in the case of line segments computed with SSWMS, N ranges between 100 and 500 depending on the image contents and size). At each iteration the most consuming stages of the algorithm are the E-steps, where the conditional probabilities γ_{ijk} have to be computed (two times since there are two E-steps). In comparison, the cost of the M-steps is negligible. In average, we have found that the proposed method tends to converge in 5 to 10 iterations.

5.5. Performance analysis

The images of the YUDB are used to test the performance of the proposed projective-plane EM algorithm. Some examples of the results are presented in Fig. 10. These examples show, in dashed lines, the initialisation of the supporting lines according to the Hough transform, coloured according to the vanishing point they meet at (which has been computed using the MSAC initialisation approach). In solid lines, the estimate of the lines after running the proposed EM algorithm is presented. Regarding the initialisation, the upper row shows examples for which the initialised supporting lines are close to actual significant lines of the image; the projective-plane EM algorithm applied on these cases provides a refinement of the initial supporting lines parameters and accurately fit them to the observed ones in the image. The second row addresses the ability of the EM algorithm to correct inaccurate initialisations. As shown, some of the initialised supporting lines have incorrect orientation mainly due to a low accuracy in the vanishing point initialisation. These inaccuracies can be due to two main factors: (i) the set of detected features is not dense enough to provide enough line segments for each vanishing points (such as for the bottom image in (a)); or (ii) there is a large proportion of inliers that are not clustered into main lines and that have higher orientation error than samples actually clustered into main lines. The MSAC processes all the inliers and thus the estimation can be affected by the presence of such false-inliers (examples shown in the second row, (b) and (c)). In fact, some of these inliers are actually noisy line segments accidentally meeting at the vanishing point under consideration, such as those in the trees or the floor. Their presence heavily disturbs the correct estimation of vanishing points as they show high error values (although within the bounds of acceptance for the MSAC procedure). The application of the proposed EM-based method corrects these initialisations and rectifies the position of the vanishing points and the corresponding supporting lines. The result, as shown in the examples of Fig. 10 is that the proposed method accurately determines the three main directions of the scene as well as the main supporting lines passing through them, including vanishing points inside and outside the limits of the image, and those in the infinity (for instance, case (b) of upper row shows, respectively, in green and red, two parallel supporting lines meeting at the infinity). The computation of the supporting lines during the optimisation process is the reason for such a good performance. The supporting lines act as a sub-selection process that filters which line segments will contribute to the estimation of the vanishing point. This way, line segments are selected only if they are clustered around significant lines, discarding other line segments (that could also meet at the vanishing point) if they do not belong to significant clusters. The advantage comes from the hypothesis that line segments that are clustered into lines have less error with respect to vanishing points than isolated or sparse line segments, which could be in fact outliers accidentally meeting at the vanishing point, and thus having higher error than true inliers. The proposed method, by using this line clustering approach, discards these false inliers enhancing the accuracy of the estimation. This is precisely what makes the MSAC to fail (and actually any estimation method that does not sub-select inliers according to this criterion) in the cases shown in Fig. 10 second row, (b) and (c). This property of the proposed projective-plane EM algorithm is exemplified in Fig. 11. Column (a) shows the classification of line segments obtained applying the MSAC algorithm for two example images of the YUDB. In (b), an example pair of supporting lines for one vanishing point is shown in solid black lines. The corresponding line segments

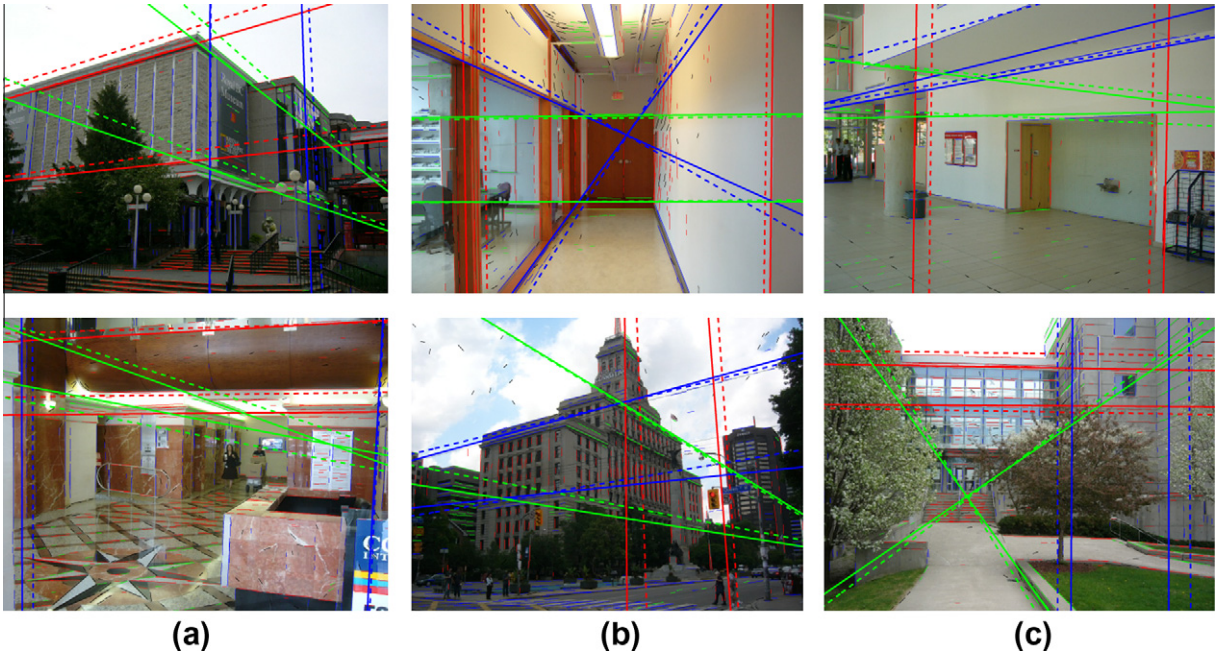

Fig. 10. Examples of the application of the proposed projective-plane EM algorithm for the estimation of multiple vanishing points. The upper row show cases in which the initialised vanishing points (the intersection of the dashed lines) is quite correct and thus the EM algorithm just works as a refinement step. The bottom row shows more difficult cases, in which the initialisation of some vanishing points is significantly incorrect, like the blue vanishing point in (a), or the green one in (b). In these cases, the proposed strategy corrects these errors and provides highly accurate estimates of the vanishing points. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

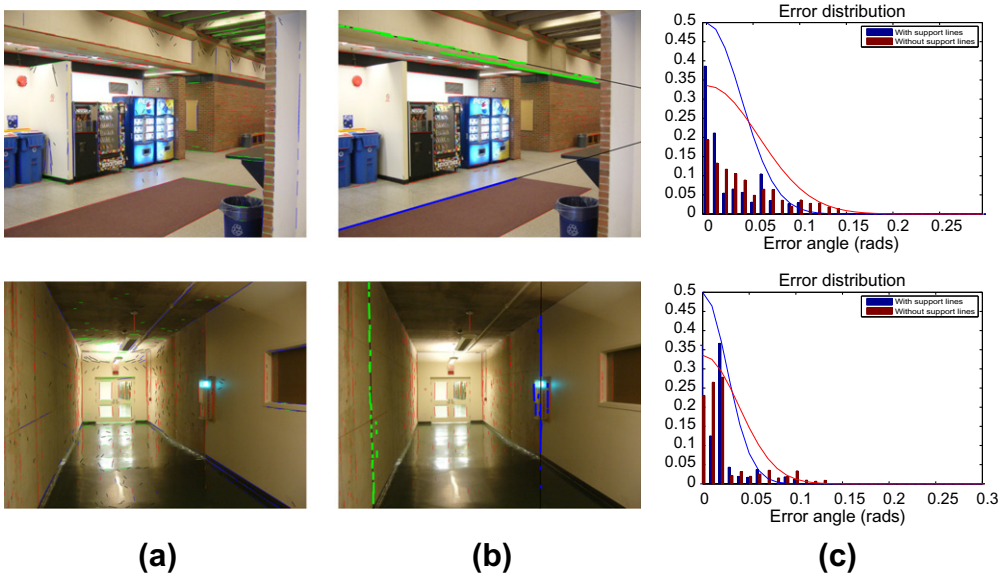

Fig. 11. Error distribution with and without using the supporting lines: (a) classification of line segments given by the MSAC algorithm; (b) line segments corresponding to two supporting lines computed by the projective-plane EM algorithm; (c) comparison of the normalised histograms (weighted according to the length of the segments) of the orientation error.

associated to these supporting lines (whose conditional probabilities γ_{ijk} are higher than that of any other component of the mixture model) are highlighted in thick green[1] and blue lines, while the rest of line segments associated to the same vanishing point are shown in red. The plots of column (c) show the normalised histogram of the orientation error weighted according to the length of the line segments. A normal fit is also provided for the two datasets: one corresponding to the complete set of inliers provided by MSAC, and the other corresponding to the subset of line segments selected according to the supporting lines computed by the EM algorithm. As shown, the use of supporting lines provides narrower error distributions for the two cases. The upper row shows a case in which the supporting lines actually correspond to two very long, well defined lines in the scene, and thus the orientation error histogram is much narrower than that computed with the complete set of line segments, which contain a large number of what we call

[1] For interpretation of colour in Figs. 11, the reader is referred to the web version of this article.

false inliers. The bottom row shows an example in which line segments are all of similar length, and the supporting lines can not be fitted to a well defined set of long line segments in the image. Although here supporting lines are fitted to a slightly sparse set of aligned line segments, there is still a significant gain in the reduction of the error. Finally, we have analyzed the accuracy improvement that the proposed projective-plane EM algorithm provides to the initial estimates which are obtained using the proposed MSAC strategy for the whole set of images of the YUDB. Provided that there are 102 images in the database and a total number of 301 vanishing points, the MSAC algorithm obtains correct detections (below 10° with respect the ground truth vanishing points) for 284/301 = 94.35% vanishing points. The average error achieved by this method is 3.5°. For the rest of vanishing points 17/301, the MSAC algorithm generate estimates with errors above 10°. In these cases, the method fails due to the low number of line segments meeting at the vanishing point which, sometimes, is even lower than the remaining outliers or clutter line segments of the scene. Considering these 284 correct initialisations, the projective-plane EM algorithm refines the obtained vanishing points and reduces the error to 1° for 247/284 = 86.97% cases. For only 28/284 the EM method does not improve the MSAC results. Finally, the projective-plane EM algorithm also commits some errors, and for 9/284 = 3.17% of the cases, the MSAC estimation is better than that provided by the EM strategy. All these estimation errors refers to the cases in which either there are not enough line segments supporting the vanishing point or they are not clustered in dominant lines but actually sparsely distributed along the whole image.

6. Conclusions

The proposed method, defined on the projective plane, provides excellent vanishing point detection results for general problems, since it treats equally all vanishing points even if they are at the infinity or within the image bounds. Up to our knowledge, no other work in the literature solves the simultaneous estimation of multiple vanishing points with their converging lines in the projective plane. The use of these lines in the EM framework has been shown to enhance the accuracy of estimates, since it allows taking advantage of the clustering of line segments into dominant lines in structured scenarios, which are less noisy than sparse sets of line segments. This way, the EM algorithm automatically selects only the line segments that belong to these supporting lines and discard the rest of information, thus leading to significantly more accurate results

Acknowledgments

This work has been partially supported by the Ministerio de Ciencia e Innovación of the Spanish Government under projects TEC2007-67764 (SmartVision) and TEC2010-20412 (Enhanced 3DTV), and by the Basque Government under the ETORGAI project berriTRANS.

References

Almansa, A., Desolneux, A., Vamech, S., 2003. Vanishing point detection without any a priori information. IEEE Trans. Pattern Anal. Machine Intell. 25 (4), 502–507.

Antone, M.E., Teller, S., 2000. Automatic recovery of relative camera rotations for urban scenes. In: Proc. Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 282–289.

Barnard, S.T., 1983. Interpreting perspective images. Artif. Intell. J. 21 (4), 435–462.

Cantoni, V., Lombardi, L., Porta, M., Sicard, N., 2001. Vanishing point detection: Representation analysis and new approaches. In: Proc. Internat. Conf. on Image Analysis and Processing, pp. 26–28.

Caprile, B., Torre, V., 1990. Using vanishing points for camera calibration. Internat. J. Comput. Vision (3), 127–140.

Coughlan, J., Yuille, A., 1999. Manhattan world: Compass direction from a single image by bayesian inference. In: Proc. Internat. Conf. on Computer Vision, pp. 941–947.

Criminisi, A., Reid, I., Zisserman, A., 2000. Single View Metrology. Internat. J. Comput. Vision 40 (2), 123–148.

Denis, P., Elder, J.H., Estrada, F.J., 2008. Efficient edge-based methods for estimating Manhattan frames in urban imagery. In: Proc. European Conf. on Computer Vision, vol. 2, pp. 197–210.

Hartley, R.I., Zisserman, A., 2004. Multiple View Geometry in Computer Vision. Cambridge University Press.

Košecká, J., Zhang, W., 2003. Video Compass. In: Proc. European Conf. on Computer Vision, LNCS, vol. 2350, pp. 476–491.

Lai, A.H.S., Yung, N.H.C., 2000. Lane detection by orientation and length discrimination. IEEE Trans. Systems Man Cybernet. – Part B: Cybernetics 30 (4), 539–548.

Liebowitz, D., 2001. Camera calibration and reconstruction of geometry, Ph.D. Thesis, University of Oxford.

Liebowitz, D., Zisserman, A., 1998. Metric rectification for perspective images of planes. In: IEEE Proc. Computer Vision and Pattern Recognition, pp. 482–488.

Lutton, E., Maître, H., Lopez-Krahe, J., 1994. Contribution to the determination of vanishing points using Hough transform. IEEE Trans. Pattern Anal. Machine Intell. 16 (4), 430–438.

Magee, M.J., Aggarwal, J.K., 1984. Determining vanishing points from perspective images. CVGIP 26, 256–267.

McLean, C.F., Koyyuri, D., 1995. Vanishing point detection by line clustering. IEEE Trans. Pattern Anal. Machine Intell. 17 (11), 1090–1095.

Minagawa, A., Tagawa, N., Moriya, T., Gotoh, T., 2000. Vanishing point and vanishing line estimation with line clustering. IEICE Trans. Inf. Syst. E83-D (7).

Nieto, M., Salgado, L., 2010. Non-linear optimization for robust estimation of vanishing points. In IEEE Proc. Internat. Conf. on Image Processing, pp. 1885–1888.

Nieto, M., Cuevas, C., Salgado, L., García, N., 2011. Line segment detection using weighted Mean Shift procedures on a 2D Slice sampling strategy. Pattern Anal. Appl. 14 (2), 149–163, 201.

Pflugfelder, R., 2008. Self-calibrating cameras in video surveillance, Ph.D. Thesis, Graz University of Technology.

Quan, L., Mohr, R., 1989. Determining perspective structures using hierarchical Hough transform. Pattern Recognition Lett. 9, 279–286.

Rasmussen, C., 2004. Grouping dominant orientations for ill-structured road following. In: Proc. Computer Vision and Pattern Recognition, Vol. 1, pp. 470–477.

Ribeiro, E., Hancock, E.R., 2000. Perspective pose from spectral voting. In: IEEE Proc. Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 656–662.

Rother, C., 2000. A new approach for vanishing point detection in architectural environments. In: Proc. 11th British Machine Vision Conf., pp. 382–391.

Schaffalitzky, F., Zisserman, A., 2000. Planar grouping for automatic detection of vanishing lines and points. Image Vision Comput. 18, 647–658.

Schindler, G., Dellaert, F., 2004. Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In Proc. Conf. on Computer Vision and Pattern Recognition, pp. 203–209.

Sekita, I., 1994. On fitting several lines using the EM algorithm, In: CVVC'94, pp. 107–109.

Seo, K.-S., Lee, J.-H., Choi, H.-M., 2006. An efficient detection of vanishing points using inverted coordinates image space. Pattern Recognition Lett. 27 (2), 102–108.

Suttorp, T., Bücher, T., 2006. Robust vanishing point estimation for driver assistance. In: IEEE Proc. Intelligent Transportation Systems Conf., pp. 1550–1555.

Trinh, H.-H., Jo, K.-H., 2006. Image-based structural analysis of building using line segments and their geometrical vanishing points. In: Proc. SICE-ICASE Internat. Joint Conf., pp. 566–571.

Tuytelaars, T., Proesmans, M., Van Gool, L., 1998. The cascaded Hough transform. In: IEEE Proc. Internat. Conf. on Image Processing, pp. 736–739.

Wang, G., Tsui, H.-T., Jonathan Wu, Q.M., 2009. What can we learn about the scene structure from three orthogonal vanishing points in images. Pattern Recognition Lett. 30 (3), 192–202.