



**HAL**  
open science

# Tracking Orthogonal Vanishing Points in Video Sequences for a Reliable Camera Orientation in Manhattan World

Wael Elloumi, Sylvie Treuillet, Rémy Leconge

► **To cite this version:**

Wael Elloumi, Sylvie Treuillet, Rémy Leconge. Tracking Orthogonal Vanishing Points in Video Sequences for a Reliable Camera Orientation in Manhattan World. International Congress on Image and Signal Processing (CISP 2012), Oct 2012, Chongqing, China. pp.5. hal-00771525

**HAL Id: hal-00771525**

**<https://hal.science/hal-00771525>**

Submitted on 8 Jan 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Tracking Orthogonal Vanishing Points in Video Sequences for a Reliable Camera Orientation in Manhattan World

Wael Elloumi, Sylvie Treuillet, Rémy Leconge  
Université d'Orléans, Laboratoire PRISME  
12 rue de Blois, 45067 Orléans cedex

*Abstract*—This paper proposes an algorithm pipeline for estimating the camera orientation based on vanishing points computation targeting pedestrian navigation assistance in Manhattan World. Inspired from some of published methods, the proposed pipeline introduces a novel sampling strategy among finite and infinite vanishing points and a tracking along a video sequence to enforce the robustness by extracting the three most pertinent orthogonal directions while preserving a short processing time for real-time application. Experiments on real images and video sequences show that the proposed heuristic strategy for selecting orthogonal vanishing points is pertinent as our algorithm gives better results than the recently published RNS optimal method [16], in particular for the yaw angle, which is actually essential for navigation task.

*Keywords-component; Vanishing Points; Camera Orientation; Video Sequences; Manhattan World.*

## I. INTRODUCTION

By perspective projection, the parallel lines in 3D scene intersect in the image plane in a so-called vanishing points. If the image plane is parallel to one axis of the 3D world, vanishing lines intersect very far from the image center, that is called infinite vanishing point, unlike the finite ones whose coordinates may be determined in the image plan. In man-made urban environments, many line segments are oriented along three orthogonal directions aligned with the global reference frame. Under this so-called Manhattan world assumption, vanishing lines or points are pertinent visual cues to estimate the camera orientation [1], [7], [10], [11], [12] [14]. This is an interesting alternative to structure and motion estimation based on features matching, a sensitive problem in computer vision. The orientation matrix of a calibrated camera, parameterized with three angles, may be efficiently computed from three noise-free orthogonal vanishing points. Vanishing points are also used for camera calibration [12] [18], [21]. However, these techniques rely on the ability to find a robust orthogonal triplet of vanishing points in a real image. Despite numerous papers dedicated to the straight line clustering to compute adequate vanishing points, this problem remains an open issue for real time application in video sequences. The estimation of the camera orientation is generally computed in a single image. Few works address the tracking along a video sequence [14]. In the context of navigation assistance for blind people in urban area, we address this problem with calibrated camera. Based on

previous work, we propose a pragmatic solution to achieve an accurate estimation of the camera orientation while preserving a short processing time. Our algorithm pipeline relies on two contributions: a novel sampling strategy among finite and infinite vanishing points extracted with a RANSAC-based line clustering, and a tracking along a video sequence.

The paper is organized as follows. A review of related work is proposed in Section 2. The Section 3 presents our method for selecting and tracking three reliable orthogonal vanishing points which are used for estimating the camera orientation. Experimental results are shown in Section 4 and the Section 5 concludes the paper.

## II. RELATED WORK

Since 30 last years, the literature is broad on the subject of vanishing points (VP) computation. In this section we review some of the most relevant. The first approaches used the Hough transform and accumulation methods [2], [4], [5], [13]. The efficiency of these methods highly depends on the discretization of the accumulation space and they are not robust in presence of outliers [20]. Furthermore, they do not consider the orthogonality of the resulting VP. An exhaustive search method may take care of the constraint of orthogonality [19] but it is off-side for real-time application.

Even few authors prefer to work on the raw pixels [8], [14], published methods mainly work on straight lines extracted from image. According to the mathematical formalisation of VP, some variants exist in the choice of the workspace: image plane [5], [19], projective [10], [17], [18] or Gaussian sphere [2], [6], [12], [13]. Using Gaussian unit sphere or projective plane allow to treat equally finite and infinite VP, unlike image plane. This is well suited representation for simultaneously clustering lines that converge at multiple vanishing points by using a probabilistic Expectation-Maximisation (EM) joint optimization approach [1], [7], [12], [15], [17]. These approaches adress the misclassification and optimality issues but the initialization and grouping are the determining factors of their efficiency.

Recently, many authors adopt robust estimation based on RANSAC [9], as the code is fast, easy to implement, and requires no initialization. These approaches consider intersection of line segments as VP hypotheses and then

iteratively clustering the parallel lines consistent with this hypothesis [10], [16], [22]. A variant by J-Linkage algorithm has been used by [21]. By dismissing the outliers, the RANSAC-based classifiers are much more robust than accumulative methods, and give a more precise position of the VP, limited by the size of the accumulator cell. They have been used to initialize EM estimators to converge to the correct VP. Other optimal solutions rely on analytical approach often based on time consuming algorithms [3], [11], [16]. In this last paper, it is interesting to note that, even if they are non deterministic, the RANSAC-based approaches hold comparable results against exhaustive search for the number of clustered lines. So, it remains a very good approach for extracting the VP candidates, in addition with a judicious strategy for selecting a triplet consistent with the orthogonality constraint.

### III. METHOD OVERVIEW

To achieve an accurate estimation of the camera orientation based on three reliable orthogonal vanishing points (VP), we propose an algorithm pipeline inspired from some of published methods, adding a novel sampling strategy among finite and infinite VP and a tracking along a video sequence.

The objective is to enforce the robustness by extracting the three most pertinent orthogonal VP while preserving a short processing time for real-time application. The next sections give some details and justifications about each bloc of the proposed pipeline.

#### A. Vanishing Point Selection

The output of this bloc must provide three VP, each of them aligned with the three main orthogonal directions of the Manhattan world. It is realised in two steps: i) dominant line extraction, ii) line clustering for VP detection. The most intuitive method is to detect the intersection of dominant lines in images. Working directly in the image plan is fast because it does not require a projection in other bounded space like Gaussian sphere. On the other hand, infinite VP need to be detected separately from the finite ones, but we will see in next section that we can take advantage of this differentiation in the good choice of orthogonal VP, with a fast and robust sampling strategy.

1) *Dominant Line Detection*: Some pre-processing are introduced to improve the quality and the robustness of the detected edges in case of embedded camera: first, an histogram equalization harmonizes the distribution of brightness levels in the image, secondly a geometric correction of lenses distortion is done assuming that the camera calibration matrix is known. To find the dominant lines, we detect edges by using a Canny's detector. Then, edge points are projected into sinusoidal curves in polar accumulation space by applying a Hough Transform (HT), where peaks correspond to the dominant clusters of line segments. We use the probabilistic version of HT as it is faster than the classic one. Only 10% or 20% of the edges are randomly selected to obtain statistically good results. Only the straight lines that are long enough are selected as input to estimate multiple VP in an image.

2) *Line Clustering by RANSAC*: Recently, numerous authors adopt RANSAC as a simple and powerful method to provide a partition of parallel straight lines into clusters by pruning outliers. The process starts by randomly selecting two lines to generate a VP hypothesis, then, all lines consistent with this hypothesis are grouped together to optimize the VP estimate. Once a dominant VP is detected, all the associated lines are removed, and the process is repeated to detect the next dominant VP. The principal drawback of this sequential search is that no orthogonality constraint is imposed for selecting a reliable set of three VP to compute the camera orientation. Very recent works propose optimal estimates of three orthogonal VP by an analytical approach based on a multivariate polynomial system solver [16] or by optimization approach based on interval analysis theory [3], but at the expenses of complex time consuming algorithms.

In this work, we introduce an heuristic strategy to extract a limited number of reliable VP while enforcing the orthogonality constraint, in conjunction with RANSAC. In the context of pedestrian navigation, the main orthogonal directions in Manhattan world consist generally in a vertical one (often associated with an infinite VP) and two horizontal ones (associated with finite or infinite VP). So we consider three different possible configurations depending on the alignment of the image plane with the 3D urban scene: i) one finite and two infinite VP, ii) two finite and one infinite VP, iii) three finite VP. The two first configurations are common unlike the third. More details about the computation of the camera orientation depending on these three configurations will be given in section 3.2.

For a robust selection of VP, we detect the three finite candidates and two infinite ones that maximize the consensus set. The criteria used in the consensus score (1) for clustering lines by RANSAC are different depending on each category. Unlike the finite VP whose coordinates may be determined in the image plan, the infinite VP are generally represented as a direction. For finite VP, the consensus score is based on a distance between the candidate straight line and the intersecting point (2). For infinite VP, it uses an angular distance between the direction of the candidate straight line and the direction representing the infinite VP (3).

$$score = \sum_{i=0}^n f(v, l_i) \quad (1)$$

$$f(v, l_i) \begin{cases} 1, & d(v, l_i) < \delta \\ 0, & otherwise \end{cases} \quad (2)$$

where  $n$  is the number of dominant lines and  $d(v, l_i)$  is the Euclidian distance from the finite VP candidate  $v$  to the line  $l_i$ . All lines whose distance is below a fixed threshold  $\delta$  are considered as participants (the threshold  $\delta$  is equal to 4 pixels in our experiments).

$$f(v, l_i) \begin{cases} 1, & \text{Min}(\widehat{(\vec{v}, \vec{l}_i)}, \widehat{(\vec{l}_i, \vec{v})}) < \delta \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where  $\widehat{(\vec{v}, \vec{l}_i)}$  is the angle between the infinite VP direction from the image center and the line  $l_i$  to test in image space (the threshold  $\delta$  is equal to  $4^\circ$  in our experiments).

To avoid redundant VP candidates, we introduce the supplementary constraint to be far enough from each other: we impose on VP to have a minimum angular distance between their directions from the image center (threshold is set to  $30^\circ$  for finite VP and  $60^\circ$  for infinite ones).

By separating finite from infinite VP, the sampling strategy provides the most significant of them without giving more importance to one or other category (we enforce to have at least one candidate finite). Furthermore, this heuristic strategy is faster as we detect only five reliable VP candidates against generally much more for the previous published methods.

3) *Three Orthogonal Vanishing Points*: Among the five candidates selected before, only three VP whose directions from the optical center are orthogonal have to be accepted, included at least one finite VP. We adopt the following heuristic: i) choose the finite VP with the highest consensus score, ii) select two other VP (finite or infinite) based on their orthogonality to the first one, and considering their consensus score as a second criterion. Finally, we identify the vertical VP and the two horizontal ones. In our application, we assume that the camera is kept upright: we identify the vertical VP as which presents the closest direction with the vertical direction from the image center. The two remaining VP are thus horizontal.

### B. Camera Orientation Computation

This part is directly inspired from [4] to compute the camera orientation from the three VP supposed to be orthogonal. We use the directions of the detected VP which correspond to the camera orientation to compute the rotation matrix  $(\vec{u}, \vec{v}, \vec{w})$ . The vectors  $\vec{u}, \vec{v}$  and  $\vec{w}$  to be found represent three orthogonal directions of the scene, respectively the first horizontal direction, the vertical direction and the second horizontal direction. They need to satisfy the following orthonormal relations:

$$\begin{cases} \vec{u} \cdot \vec{v} = \vec{v} \cdot \vec{w} = \vec{w} \cdot \vec{u} = 0 \\ \|\vec{u}\| = \|\vec{v}\| = \|\vec{w}\| = 1 \end{cases} \quad (4)$$

The estimation of these vectors depends on the VP configurations.

1) *One Finite and Two Infinite VP*: This situation is the most frequent one. It occurs when the image plane is aligned with two axis of the world coordinate frame. Let  $V$  be the finite VP and  $f$  the focal length. The direction of  $V$  can be expressed as  $\vec{OV} = (V_{1x}, V_{1y}, -f)^T$  whereas the directions of the infinite VP, in image space, are  $\vec{l}_1 = (I_{1x}, I_{1y}, 0)^T$  and  $\vec{l}_2 = (I_{2x}, I_{2y}, 0)^T$ . The vectors of the rotation matrix are given by the following system of equations:

$$\begin{cases} \vec{u} = \left( I_{1x}, I_{1y}, \frac{I_{1x} V_{1x} + I_{1y} V_{1y}}{f} \right)^T \\ \vec{v} = \left( I_{2x}, I_{2y}, \frac{I_{2x} V_{1x} + I_{2y} V_{1y}}{f} \right)^T \\ \vec{w} = -\vec{OV} = (-V_{1x}, -V_{1y}, f)^T \end{cases} \quad (5)$$

2) *Two Finite and One Infinite VP*: This situation happens when the image plane is aligned with only one of the three axis of the world coordinate frame. Let  $V_1$  and  $V_2$  be the two finite VP of directions  $\vec{OV}_1 = (V_{1x}, V_{1y}, -f)^T$  and  $\vec{OV}_2 = (V_{2x}, V_{2y}, -f)^T$ . Since there are two finite horizontal VP, we set  $\vec{w}$  to the closest VP to the image center. The vector  $\vec{v}$  is obtained by cross product as shown in the system of equations below.

$$\begin{cases} \vec{u} = (-V_{1x}, -V_{1y}, f)^T \\ \vec{w} = (-V_{2x}, -V_{2y}, f)^T \\ \vec{v} = \vec{u} \times \vec{w} \end{cases} \quad (6)$$

3) *Three Finite VP*: This last configuration is the least frequent one. It occurs when there is no alignment between the image plane and the world coordinate frame. Let  $V_1, V_2$  and  $V_3$  be the three finite VP of directions  $\vec{OV}_1 = (V_{1x}, V_{1y}, -f)^T$ ,  $\vec{OV}_2 = (V_{2x}, V_{2y}, -f)^T$  and  $\vec{OV}_3 = (V_{3x}, V_{3y}, -f)^T$ . We start by setting  $\vec{v}$  to the VP whose direction is closest to the vertical direction. We then set  $\vec{w}$  to the closest VP to the image center. In the system of equations (7), we assume that  $V_2$  is the vertical VP and  $V_3$  is closest to the image center.

$$\begin{cases} \vec{u} = (-V_{1x}, -V_{1y}, f)^T \\ \vec{v} = (-V_{2x}, -V_{2y}, f)^T \\ \vec{w} = (-V_{3x}, -V_{3y}, f)^T \end{cases} \quad (7)$$

### C. Vanishing Point Tracker

Once the whole described algorithm is processed for the first frame of the video sequence ( $N > 1$ ), the estimated camera parameters can be tracked from one frame to another. Indeed, VP positions or directions are slightly modified in video-sequences or even in a list of successive frames. So we introduce a tracker to check consistency between the positions of the estimated VP in the frame  $N$  and those estimated in frame  $N - 1$ . For this we use the distance between the positions of the VP for the finite ones  $d(v_N, v_{N-1})$ , and the angle between the VP directions for the infinite ones  $(\widehat{(\vec{v}_N, \vec{v}_{N-1})})$ . When a VP is not coherent with its previous position or direction, it is re-estimated taking into account its previous position or direction and using the remains of unclassified lines. Hence, aberrant VP are discarded and replaced by new VP that are, at the same time, consistent with the previous ones and satisfy the orthogonality constraint. This tracker is efficient since it makes our algorithm much more stable and much robust as will be shown in the next section.

## IV. EXPERIMENTS

This section presents the performance evaluation of the proposed method on real static images and video sequences.

### A. Vanishing Point Estimation

For comparison purpose, we have tested our algorithm on the public York Urban Database (YUD) facing two other approaches for estimating the VP quoted in the related work section:

- an Hough-based accumulation approach proposed by [4].
- an analytical method RNS recently published by [16], that provides optimal least-squares estimates of three orthogonal vanishing points using a RANSAC-based classification of lines.

Each image in the YUD has been hand-labelled to identify the three Manhattan vanishing points and from these to compute the Euler angles relating the camera frame to the scene frame. The database provides the original images, camera calibration parameters, ground truth line segments, and the camera orientation for each image [8]. Fig. 1 illustrates some orthogonal vanishing points and their associated parallel lines extracted by our algorithm on some images pulled out the YUD.

The table 1 presents a comparative study of the methods facing the angular distance from the Ground Truth (GT) of the camera orientation provided in the YUD.

The three first rows of the Table 1 give the number of times the distance exceeds a fixed value of 2, 5 and 10 degrees respectively. The average and standard deviation are reported in the two following rows (in degrees). For the ten tested images, the HT method is obviously the less reliable, as it may

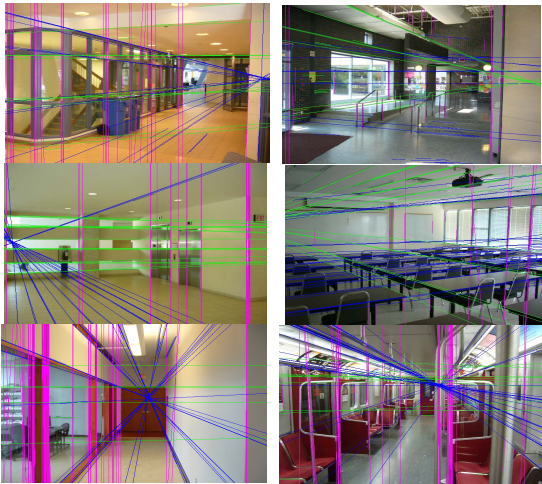


Figure 1: Examples of some triplets of orthogonal vanishing points detected by our algorithm on images from the York Urban Database.

TABLE I. COMPARATIVE STUDY OF THE METHODS FACING THE ANGULAR DISTANCE FROM THE GT (IN DEGREES)

	RNS method			Our method			HT method		
	pitch	yaw	roll	pitch	yaw	roll	pitch	yaw	roll
> 2°	0	4	4	5	0	1	7	7	3
> 5°	0	1	1	2	0	0	6	6	1
> 10°	0	1	1	0	0	0	4	4	1
Average	<b>0,81</b>	2,37	2,52	2,89	<b>0,99</b>	<b>1,01</b>	17,57	13,53	8,01
Std Dev	<b>0,57</b>	2,92	2,90	2,91	<b>0,64</b>	<b>1,08</b>	21,18	15,28	20,92

miss-classify parallel lines in the presence of outliers and does not include the orthogonality constraints between the vanishing points. By using RANSAC-based classification of lines, the other methods remove the outliers from the parallel groups. The RNS method gives the best result for the pitch angle but it is interesting to note that our method is significantly better for the yaw and roll angles. The yaw is actually essential for a pedestrian navigation task since it gives the camera viewing direction. This may be explained by our heuristic strategy for selecting orthogonal vanishing points that are distant enough from each other without confusion between finite and infinite points.

Our method and HT have been implemented by using Visual C++ and OpenCV library. The full results of the RNS method are available in a technical report provided online by the authors (<http://umn.edu/~faraz/vp>). Concerning the processing time for estimating the three orthogonal vanishing points on YUD images of size 640x480 pixels, our method takes ~86 milliseconds, against ~1.12 seconds for HT on a Intel Core 2 Duo processor. For RNS, the authors mention that their current C++ implementation of the symbolic-numeric method takes ~290 milliseconds to find the solutions on a Core 2 Duo processor.

### B. Tracking the Camera Orientation

To show the efficiency of our algorithm for tracking the camera orientation, we acquire real video sequences, with an embedded camera. Our experimental prototype is composed of a camera AVT GUPPY F-033C equipped with a 3.5mm lens and a laptop (Intel core 2 Duo 2.66GHz/4096MB). The video sequence is composed of 350 frames acquired at 25 frames per second in the hallways of our laboratory. As we use a lens with a short focal length, it is recommended to apply a geometric distortion correction before extracting line segments. The camera has been first calibrated using the tool: [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/), a software proposed by Bouguet. Fig. 2 depicts some typical results of vanishing point extraction for this sequence. Fig. 3 compares the evolution of the roll, pitch and yaw angles of the camera orientation along the sequence by applying our method with and without the vanishing point tracker (VPT). The VPT produces a smooth running and a more reliable estimation for camera orientation along the video sequence. Since the VPT removes some aberrant vanishing points, keeping only the points that are consistent, we then obtain a more accurate camera orientation.

The full processing time for estimating the camera orientation takes 16 milliseconds per image of size 320x240 pixels with non-optimized code. Therefore, our algorithm is suitable for real time applications, such as navigation assistance for blind pedestrian.

## V. CONCLUSION

We propose an algorithm pipeline to achieve an accurate estimation of the camera orientation based on three reliable orthogonal vanishing points in the context of visual navigation. Our method introduces a novel sampling strategy among finite and infinite vanishing points and a tracking along a video sequence. Experimental results on real images, show that, even

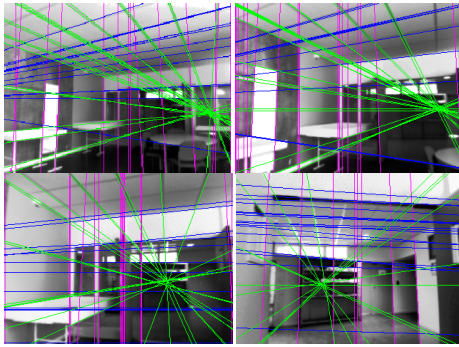


Figure 2: The vanishing points and their associated lines can be robustly detected and tracked along the video sequence during the navigation.

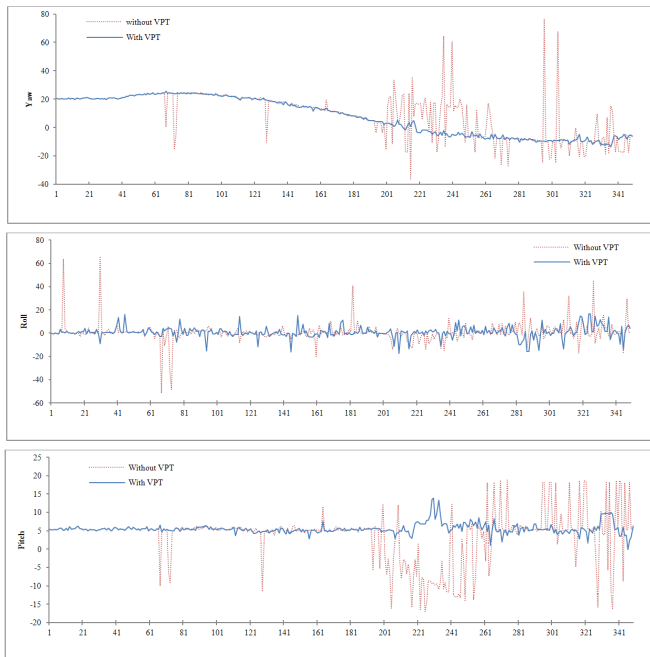


Figure 3: Smoothing effect of the VPT on the estimation of the camera's orientation (pitch, yaw and roll angles).

simple, the adopted strategy for selecting three reliable distant and orthogonal vanishing points in conjunction with RANSAC performs well in practice since the estimation of the camera orientation is better than those obtained with a state-of-art analytical method. Furthermore, the tracker proved to be relevant to dismiss aberrant vanishing points along the sequence, making outmoded refinement or optimization later step and preserving a short processing time for real-time application. This algorithm is devoted to be a part of a localization system that should provides navigation assistance for blind people in urban area.

#### ACKNOWLEDGMENT

This study is supported by HERON Technologies SAS and the Conseil Général du LOIRET.

#### REFERENCES

- [1] M. Antone and S. Teller, "Automatic recovery of relative camera rotations for urban scene," in Proc. of IEEE Conf. Computer Vision and Pattern recognition (CVPR) pp. 282–289, 2000.
- [2] S. T. Barnard, "Interpreting perspective images," *Artificial Intelligence*, 21(4), pp. 435–462, Elsevier Science B.V, 1983.
- [3] J. C. Bazin, Y. Seo, C. Demonceaux, P. Vasseur, K. Ikeuchi, I. Kweon and M. Pollefeys, "Globally optimal line clustering and vanishing points estimation in a Manhattan world," in the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR), 2012.
- [4] K. Boulanger, K. Bouatouch, and S. Pattanaik, "ATIP : A tool for 3D navigation inside a single image with automatic camera calibration," in EG UK Conf on Theory and Practice of Computer Graphics, 2006.
- [5] V. Cantoni, L. Lombardi, M. Porta and N. Sicard, "Vanishing Point Detection: Representation Analysis and New Approaches," in Proc. of Int. Conf. on Image Analysis and Processing (ICIAP), pp. 90–94, 2001.
- [6] R. T Collins and R. S Weiss, "Vanishing point calculation as statistical inference on the unit sphere," in: Proceedings of the 3rd Int. Conference on Computer Vision (ICCV), pp. 400–403, 1990.
- [7] J. M. Coughlan and A. L. Yuille, "Manhattan World: Compass direction from a single image by Bayesian inference," in Int. Conference on Computer Vision (ICCV), 1999.
- [8] P. Denis, J. H. Elder and F. Estrada, "Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery," in European Conference on Computer Vision (ECCV), pp. 197–210, 2008.
- [9] M. A. Fisher and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," in *Comm. of the ACM*, 1981.
- [10] W. Förstner, "Optimal vanishing point detection and rotation estimation of single images from a legoland scene," in Proceedings of the ISPRS Symposium Commission III PCV. S, pp. 157–163, Part A, Paris, 2010.
- [11] M. Kalantari, A. Hashemi, F. Jung and J.P. Guédon, "A New Solution to the Relative Orientation Problem Using Only 3 Points and the Vertical Direction," *Journal of Mathematical Imaging and Vision* archive Volume 39(3), 2011.
- [12] J. Kosecka and W. Zhang, "Video Compass," in Proc. of the 7th European Conf. on Computer Vision (ECCV), 2002.
- [13] E. Lutton, H. Maitre and J. Lopez-Krahe, "Contribution to the determination of vanishing points using Hough transform," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(4), pp. 430–438, 1994.
- [14] A. Martins, P. Aguiar and M. Figueiredo, "Orientation in Manhattan world: Equiprojective classes and sequential estimation," in the *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 27, pp. 822–826, 2005.
- [15] A. Mingawa, N. Tagawa, T. Moriya, and T. Gotoh, "Vanishing point and vanishing line estimation with line clustering," *IEICE Trans. on Inf. & Syst.*, vol.E83-D (7), 2000.
- [16] F. M. Mirzaei and S. I. Roumeliotis, "Optimal estimation of vanishing points in a Manhattan world," in the Proc. of IEEE Int. Conf. on Computer Vision (ICCV), 2011.
- [17] M. Nieto and L. Salgado, "Simultaneous estimation of vanishing points and their converging lines using the EM algorithm," *Pattern Recognition Letters*, vol. 32(14), pp. 1691–1700, 2011.
- [18] R. Pflugfelder and Bischof, "Online auto-calibration in man-made world," in Proc. Digital Image Computing : Techniques and Applications, pp. 519–526, 2005.
- [19] C. Rother, "A new approach for vanishing point detection in architectural environments," in Proc. of the 11th British Machine Vision Conference (BMVC), pp. 382–391, 2000.
- [20] J. A. Shufelt, "Performance evaluation and analysis of vanishing point detection techniques," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(3), pp. 282–288, 1999.
- [21] J.-P. Tardif, "Non-iterative approach for fast and accurate vanishing point detection," in Proc. Int. Conference on Computer Vision (ICCV), pp. 1250–1257, 2009.
- [22] H. Wildenauer and M. Vincze, "Vanishing point detection in complex man-made worlds," in Proc. of Int. Conf. on Image Analysis and Processing (ICIAP), pp. 615–622, 2007.