

# Automatic search for vanishing points on mobile devices

E V Myasnikov<sup>1,2</sup>

<sup>1</sup>Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

<sup>2</sup>Image Processing Systems Institute of RAS - Branch of the FSRC "Crystallography and Photonics" RAS, Molodogvardejskaya street 151, Samara, Russia, 443001

e-mail: mevg@geosamara.ru

**Abstract.** The vanishing point is a point in the plane of the perspective image, in which the projections of mutually parallel lines of three-dimensional space converge. Automatic search for vanishing points in images is a rather complicated task, the solution of which is not currently possible on mobile devices without introducing additional restrictions. In this paper, a rather simple method is proposed for solving this problem, which operates under two main constraints. One of these limitations is the analysis of the Manhattan scenes. Another is the fusion of information received from mobile device sensors. Modeling was performed using a public open data set to evaluate the characteristics of the proposed solution.

## 1. Introduction

Vanishing points are some of the most important objects in three-dimensional computer vision. The vanishing points are points in the plane of a perspective image, in which the projections of mutually parallel lines of three-dimensional space converge. The search of vanishing points can be one of the stages in assessing the orientation and position of a camera [1], reconstructing three-dimensional scenes [2], and can be used in many applications related to these tasks [3,4].

Images with corrected geometric distortion are used to search for vanishing points. As a rule, the automatic search for vanishing points is performed in several stages. At the first stage, the line segments are selected on images. On the second stage, the lines are divided into groups corresponding to different vanishing points. Finally, the calculation of the positions of the corresponding points is performed.

Unfortunately, the automatic search for vanishing points is quite a difficult task, the solution of which is not currently possible on mobile devices without introducing additional restrictions. In the present paper, a rather simple method is proposed for solving this problem, which operates under two main constraints. One of these limitations is the analysis of only Manhattan scenes. Another is the presence of an accelerometer on a device, which makes it possible to estimate the gravity vector.

The search for vanishing points in images of Manhattan scenes is performed in order to find three vanishing points: the vanishing points of vertical lines of facades, as well as horizontal lines of main and side facades. In this case, the vanishing points are selected in such a way that the vectors corresponding to the found vanishing points are orthogonal. In general, the proposed method is based on the idea described in [5], according to which the search for horizontal vanishing points can be performed along the horizon line, defined by a plane orthogonal to the direction to the vertical

vanishing point. Modeling was performed using a well-known open data set [6-8] to evaluate the characteristics of the proposed solution.

The paper is organized as follows. Section 2 provides a description of the developed method of searching the vanishing points. Section 3 describes the modeling methodology and experimental studies, which were conducted using an open data set. The paper ends with a conclusion and a list of references.

## 2. Method

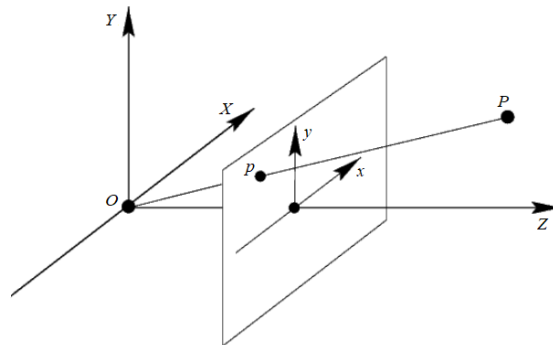
In general, the proposed method consists in the sequential search of three vanishing points, starting from the point corresponding to the vertical lines of the facades.

At the preparatory stage of the method, the image is pre-processed, including the elimination of geometric distortions. After that, a search of line fragments (segments) is performed on the image. For this purpose, the Hough transform can be used; however, in this work, rather a simple approach is used, which is based on the tracing of contours.

Let us assume that the camera is modeled as a regular pinhole camera (see Fig. 1), and the origin of the coordinate system is aligned with the camera. Let the OX and OY axes form a plane parallel to the image plane, and the OZ axis is orthogonal to it. In this case, the relationship between the coordinates of the point  $(X, Y, Z)^T$  in the specified coordinate system and the point  $(x, y)^T$  in the image plane is expressed in the form:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \sim K \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

where  $K$  is the matrix of the internal parameters of the camera, which contains information on the focal length, pixel size, skew, and shift of the image center relative to the optical axis.



**Figure 1.** Camera model and coordinate system.

To find the first vanishing point, we use information from the gravity sensor of the mobile device. At this stage, we select such line segments that are in a good agreement with the gravity vector. This selection is carried out by comparing the angle of deviation of each line  $l_i$  from the gravity vector  $g$  with a predetermined threshold value  $t$ :

$$L = \left\{ l_i \mid \arccos \left( \frac{(l_i^2 - l_i^1)^T g}{\|l_i^2 - l_i^1\| \|g\|} \right) < t \right\}$$

Here  $l_i^1$  and  $l_i^2$  are the coordinates of the ends of the corresponding line segment  $l_i$ .

The selected set of lines  $L_1$  determines the position of the first vanishing point  $V_1$ , corresponding to the vertical lines in the image. After finding the first vanishing point, we exclude all the lines corresponding to it from further consideration:  $L_2 = L / L_1$ .

At the second stage of the method, we determine the plane of the horizon line  $\alpha^h$  in three-dimensional space as a plane orthogonal to the refined gravity vector (and direction to the first vanishing point) and passing through the origin:

$$V_{11}X + V_{12}Y + V_{13}Z = 0$$

Next, we define the horizon line  $\Gamma$  in the image plane as the projection of a line passing, for example, through the points with the following coordinates:

$$\Gamma^1 = \begin{bmatrix} -1 \\ (V_{11} - V_{13})/V_{12} \\ 1 \end{bmatrix}, \quad \Gamma^2 = \begin{bmatrix} 1 \\ -(V_{11} + V_{13})/V_{12} \\ 1 \end{bmatrix}$$

To search for the second vanishing point  $V_2$ , we calculate the points  $p_i$  of the intersection of all the lines  $l_i$ , which were selected on the image  $l_i \in L_2$ , with the horizon line  $\Gamma$ :

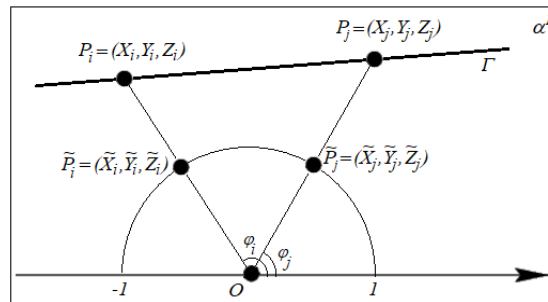
$$p_i = (x_i, y_i), \quad p_i \in l_i \wedge p_i \in \Gamma$$

Then we build the histogram  $h$  of the number of intersection points, which fall into different ranges of angles in the polar coordinate system:

$$h_n = \sum_i I\left(\frac{n-1}{N}\pi \leq \varphi_i < \frac{n}{N}\pi\right), \quad n=1..N$$

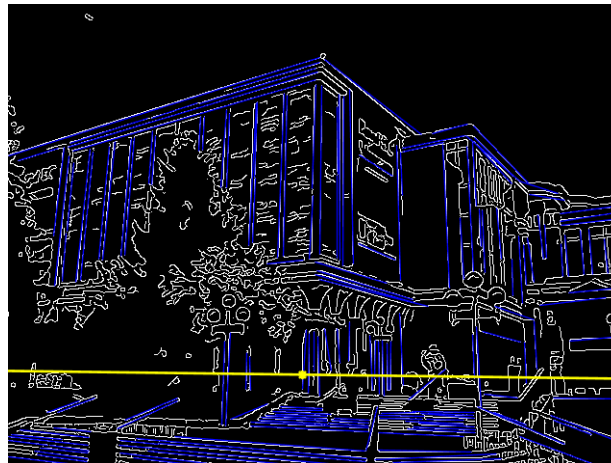
Here  $\varphi_i$  is the radial coordinate of the intersection point  $p_i$  in the plane  $\alpha^f$  (see figure 2),  $\varphi_i = \arccos(\tilde{X}_i)$ , and the normalized coordinates  $\tilde{P}_i$  are related to the coordinates in the image plane as follows:

$$\begin{aligned} (\tilde{X}_i, \tilde{Y}_i, \tilde{Z}_i) &= (X_i, Y_i, Z_i) / \|(X_i, Y_i, Z_i)\| \\ \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} &\sim K^{-1} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \end{aligned}$$



**Figure 2.** The plain of the horizon line.

An example result of the preparatory stage of the method and the found horizon line are shown in figure 3.



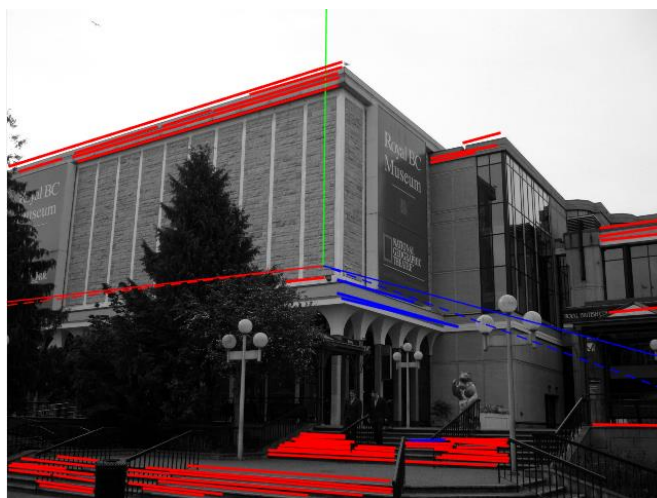
**Figure 3.** A result of the preparatory stage: the detected contours of the image are shown in white; the selected segments of straight lines are shown in blue. The horizon line found is shown in yellow.

We calculate the approximate position of the second vanishing point as the position of the histogram peak on the horizon line. Further, we refine the position of the second vanishing point using the set of line segments related to the peak on the histogram. Then we exclude the set of lines corresponding to the second vanishing point from further consideration.

The position of the third vanishing point can be found in the same way as the position of the peak on the histogram with the subsequent refinement. However, a faster way to guarantee the orthogonality of the vectors corresponding to the vanishing points is to find a vector, which is orthogonal to the vectors corresponding to the first and second vanishing points:

$$V_3 = V_1 \times V_2$$

An example demonstrating the search for the second and third vanishing points is shown in figure 4.



**Figure 4.** An example of the search for vanishing points: the directions to the true points from the center are shown by thin dashed lines; the directions to the estimated vanishing points are shown in thin solid lines; the segments of lines selected for the second and third vanishing points are shown in bold red and blue lines, respectively.

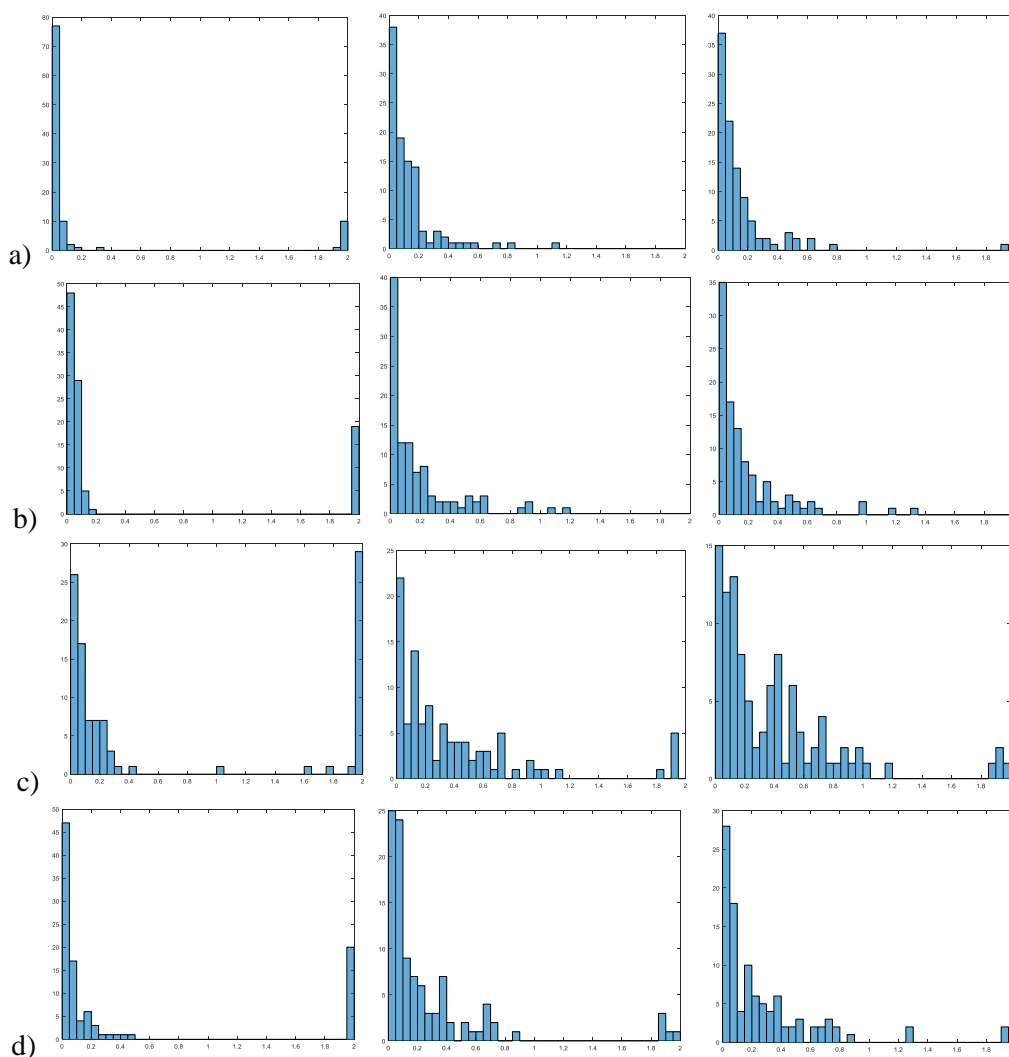
### 3. Experiments

This section describes the results of an experimental study of the method described above. For the experiments, we used the public open data set "The York Urban Database" [6-8]. It contains original images and information about camera calibration parameters as well as the true position of vanishing points. The specified data set includes 102 images of urban areas, obtained mainly on the territory of the University of York and in the city of Toronto, Canada.

To estimate the optimal parameters of the developed method, we performed the simulation according to the following scheme:

- for each image of the above data set, we calculated the true gravity vector using the information about the true position of vanishing points;
- we added random vectors (noise) with specified characteristics (uniform distribution in the range  $[-r; r]$ ) to the obtained gravity vector, and the resulting vector was used as a gravity vector obtained from the sensor of a mobile device;
- we estimated the positions of the vanishing points using the algorithm described above;
- for each vanishing point, we calculated the error as the norm of the vector difference between the estimated and true coordinates of the vanishing points.

The results of the experiments are shown in figure 5 below. Each of the histograms in the figure shows the distribution of the norm of the vector difference between the estimated and true coordinates of the vanishing points. In the ideal case, the histogram has only one leftmost nonzero column, which means minimal deviations of estimated vanishing points from the true values in all experiments. The histogram located in the left column in figure 5.a has a similar appearance.



**Figure 5.** The histogram of deviations of vanishing points. The left column corresponds to the first vanishing point, the middle column corresponds to the second point, and the right column corresponds to the third vanishing point. The rows correspond to different ranges of the noise component:  $r = 0$  (a),  $r = 0.1$  (b),  $r = 0.2$  (c, d). Figures (a)-(c) show the result for more strict selection conditions for line segments:  $t = \pi / 50$ ,  $N = 50$ . Figure (d) shows the result for soft conditions:  $t = \pi / 25$ ,  $N = 25$ .

The presence of a high right column in histograms in figures 5.b and 5.c (left) shows the error in orientation. It means that the direction to the estimated vanishing point is reversed relative to the direction to its true position. Such errors can occur since, for large random deviations of the gravity vector, the algorithm selects lines converging in the opposite direction to the true one.

As can be seen from the results of the experiments, on the one hand, the growth of the noise component causes the expected increase in errors. On the other hand, the growth becomes unacceptable for rather significant random deviations  $r = 0.2$  (see figure 5.c). Such growth can be partially compensated by mitigating the conditions when selecting line segments, as shown in figure 5.d. Here, milder filtering conditions for line segments were used: possible angle deviations were  $t = \pi / 25$  compared to the previously used  $t = \pi / 50$ , and the number of histogram splits was  $N = 25$  compared to the previously used  $N = 50$ .

Thus, the accuracy of the algorithm can be improved in the cases of noisy gravity vector readings by selecting appropriate parameters. Another way to improve the accuracy may consist in the use of previously obtained estimates in the processing of a video stream. This is a subject of future research.

#### 4. Conclusion

The paper proposed and investigated the method of automatic search for vanishing points. This method is based on the combination of information obtained from sensors of mobile devices and images obtained from the camera in the visible range. The simulation was performed using "The York Urban Database" open data set, which allowed evaluating the performance of the method under random deviations in the values of the gravity vector. The method described in the paper is easy to implement and undemanding of computing resources, which allows its use on mobile devices.

In the future, we plan to improve the described method for working with a video stream that allows tracking and refinement of the positions of vanishing points with the aim of evaluating the orientation and position of a camera.

#### 5. References

- [1] Caprile B, Torre V 1990 Using vanishing points for camera calibration *International Journal of Computer Vision* **4(2)** 127-139
- [2] Lee D C, Hebert M and Kanade T 2009 Geometric reasoning for single image structure recovery *International Conference on Computer Vision and Pattern Recognition (CVPR)* 2136-2143
- [3] Xu H, Yang Z, Zhou Z, Shanguan L, Yi K and Liu Y 2016 Indoor localization via multi-modal sensing on smartphones *International Joint Conference on Pervasive and Ubiquitous Computing, ACM* 208-219
- [4] Park S, Lee H, Lee S and Yang H S 2015 Line-based single view 3D reconstruction in Manhattan world for augmented reality *International Conference on Virtual Reality Continuum and its Applications in Industry, ACM* 89-92
- [5] AngladonV, Gasparini S and Charvillat V 2015 The toulouse vanishing points dataset *Proceedings of the 6th ACM Multimedia Systems Conference (MMSys)* (Portland, United States)
- [6] Coughlan J M, Yuille A L 2003 Manhattan world: Orientation and outlier detection by Bayesian inference *Neural Computation* **15(5)** 1063-1088
- [7] Denis P, Elder J H and Estrada F 2008 Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery *Proc. European Conference on Computer Vision* **5303** 197-211
- [8] Denis P 2008 Efficient Edge-Based Methods for Estimating Manhattan Frames in Urban Imagery *M.Sc. Thesis* (York University, Canada)

#### Acknowledgements

The work was partly funded by RFBR according to the research project 17-29-03190 in parts of «2. Method» – «3. Experiments» and by the Russian Federation Ministry of Science and Higher Education within a state contract with the "Crystallography and Photonics" Research Center of the RAS under agreement 007-Г3/Ч3363/26 in part of «1. Introduction».