Calculating Vanishing Points in Dual Space

Yong-Gang Zhao[1], Xing Wang[1], Liang-Bing Feng[1], George Chen[1], Tai-Pang Wu[2], and Chi-Keung Tang[3]

[1] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
{zhao.yg,lb.feng,qian.chen}@siat.ac.cn
[2] Enterprise and Consumer Electronics (ECE) Group, Hong Kong Applied Science and Technology Research Institute
tpwu@astri.org
[3] Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
cktang@cse.ust.hk

Abstract. Vanishing points can be used to exploit the parallel and orthogonal lines in 3D scenes thus the cameras' orientation parameters for vision processing. This paper proposed a vanishing point detection and estimation method in the dual image space. First, edge line segments are extracted. Second, based on the point-line duality theory, lines are transformed into points in the dual space where the transformed points belong to the same vanishing point form collinear clusters. Third, vanishing points are estimated by grouping and fitting straight lines across those clusters. The novel points of our method are: 1) automatically grouping the edge line segments that are the support of a vanishing point; 2) calculating the vanishing points by fitting straight lines in the dual space. Experiment results validated the proposed method.

Keywords: Vanishing Point, Dual Space, Line Detection, Homography.

1 Introduction

Parallel lines in the 3D space are projected into the 2D image space forming the so called pencil-of-lines. The intersection point of a group of convergent lines in the image plane is called a vanishing point which is actually the projection of a point-at-infinity corresponding to the aforementioned parallel 3D lines. The related convergent lines are called the support of the vanishing point. Vanishing points are widely used in parsing 3D scene structures. In 1, a 3D reconstruction approach from a single image was proposed using vanishing points. In 2, Foroosh *et al.* described how 3D Euclidean measurements could be made in a pair of un-calibrated images using vanishing points when only minimal geometric information is available. Vanishing points also play key roles in estimating the camera parameters for camera calibration. To quickly and accurately detect the vanishing points, three main problems need to be solved: i) detecting edge line segments in the image, ii) clustering of the lines belong to a pencil

J. Yang, F. Fang, and C. Sun (Eds.): IScIDE 2012, LNCS 7751, pp. 522–530, 2013.

(or the convergent lines), and iii) estimating the intersection of the lines of a cluster. Sometimes ii) and iii) are intertwined.

Common clustering methods first create a partition of the image then tally the lines that pass through each region. Line clusters are found by identifying the regions with peak counts. Different methods varied in ways of dividing the images. For example, Almansa *et al.* 4 partitioned an image into "vanishing regions" such that the probability that a random line of the image meets a vanishing region is constant for all regions. In the work of Tuytelaars *et al.* 5 and Rother 6, a partition method was used such that the projection of each vanishing region on the Gaussian sphere has a quasi-constant area. The recent work by Li *et al.* 3 suggested replacing the 2D partition by two 1-D partitions thus greatly improved the detection speed. The partition based methods are appropriate for cases where the vanishing points are close to the image center but facing difficulties in dealing with vanishing points at infinity. Schmitt *et al.* 11 used the intersection point neighborhood instead of image plane partition for clustering. The issue of point at infinity is handled in an ad hoc way by substituting the intersection point of two parallel lines (at infinity) by one or two faraway points between the parallel lines.

Other methods perform line counting in some transformed space, often Hough space or Gaussian sphere. Barnard 7 first proposed a vanishing point detection method based on the Gaussian sphere in 1983. Antone *et al.* 14 simplified the idea and used three orthogonal planes in place of the Gaussian sphere. Based on Barnard' work, Lutton *et al.* 8 proposed a method using the Hough transform to calculate the vanishing point. Ebrahimpour *et al.* 10 used the start and end points of the lines to find the vanishing points though Hough transform and K-means clustering. Using the Gaussian mixture model (GMM) of vanishing points and vanishing lines, Akihiro Minagawa *et al.* were able to solve the line clustering problem and vanishing point estimation problem simultaneously 9. The joint point and line random process was modeled in GMM and the MAP solution was found using the standard EM algorithm which produced line clusters, vanishing points as well as vanishing lines all at once.

Most of the previous algorithms either rely on the camera calibration information (e.g. those based on Gaussian sphere) or encounter singularities when dealing with points at infinity (e.g. the partition or GMM methods). In this paper, we propose a new vanishing point detection method based on dual space theory. Our key observation is that pencil-of-lines appear in the dual space as collinear points. Thus vanishing point detection and calculation are solved jointly as robust line fitting in the dual space. Straight line segments belong to distinct vanishing points are automatically classified. There is no need of any form of line accumulation. Furthermore, points at infinity are naturally represented.

Below is the organization of the paper: Section 2 describes the vanishing point calculation theory in dual space. Section 3 illustrates the vanishing point detection algorithm. Section 4 shows the experimental results. Section 5 summarizes the paper and discusses some directions for future work.

2 Vanishing Point Calculation Theory in Dual Space

According to the duality principle, lines in the image space correspond to points in the dual space. Lines meet at one point in the image space are equivalent to collinear points in the dual space. For a quick explanation, we express the 2D line equation as:

$$\begin{pmatrix} a & b & c \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0 \tag{1}$$

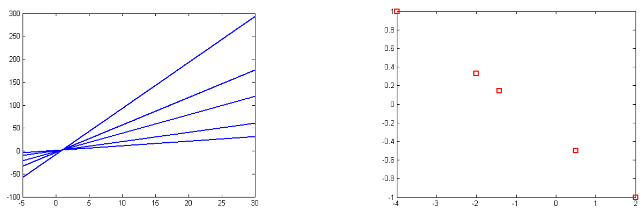
which can be equivalently written as:

$$\begin{pmatrix} x & y & 1 \end{pmatrix} \begin{pmatrix} a/c \\ b/c \\ 1 \end{pmatrix} = 0 \quad \text{for } c \neq 0. \tag{2}$$

Thus the coefficients of the line equation $(a\ b\ c)$ in the image space are equivalent to the point $\left(\frac{a}{c}, \frac{b}{c}\right)^T$ in the dual space. A pencil-of-lines in the image space means that a, b, c are variable while x, y fixed. Therefore in the dual space, points $\left(\frac{a}{c}, \frac{b}{c}\right)^T$ are collinear with the line parameters $(x, y, 1)$. For this representation to hold, the parameter c cannot be zero which in practice can be achieved by a translation of the image and later compensated by applying the inverse translation to the resulted vanishing points.

This paper takes advantage of the principle that the intersecting lines in the image space form a set of collinear points in the dual space. Therefore the problem of calculating intersection of lines is converted to the problem of fitting a straight line across a set of collinear points in the dual space as shown in Figure 1.

Based on the above observation, a new kind of vanishing point detection algorithm is proposed in this paper. First, detect edge line segments in the image space. Second, calculate the line equations of those lines and transform them into the dual space. Third, group dual points and fit lines in the dual space. The coefficients of the resulting line equations are the wanted vanishing points. Using the homogeneous point representation, this method deals with the vanishing point at infinity nicely.



(a) Intersecting lines in image space (b) Collinear points in dual space

Fig. 1. Pencil-of-lines in image space and the corresponding collinear points in dual space

3 The Vanishing Point Detection Algorithm

3.1 Line Segments Detection

In most of the vanishing point detection algorithms, the first step is line detection. Among the line detection approaches, Hough transform 12 and Line Segment Detector (LSD) 13 are widely used. In this paper we adopted the LSD method for line detection.

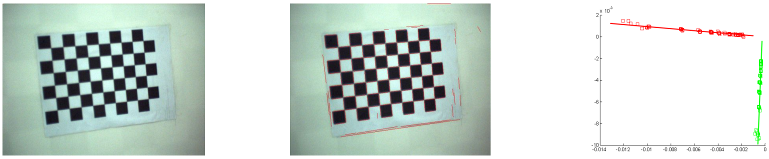
Using polar form for lines:

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

we represent a line in the dual space by

$$\left(-\frac{\cos\theta}{\rho},\quad -\frac{\sin\theta}{\rho}\right)^T \tag{4}$$

To avoid the case of zero denominator, i.e. $\rho = 0$, the image is shifted initially so that none of the lines of our interest passes through the (shifted) image origin. The inverse shifting is later applied to the obtained vanishing points.



(a) Checkerboard image (b) Lines in image space (c) The collinear point clusters in dual space

Fig. 2. Checkerboard image and its collinear dual

In Figure 2, line segments detected by LSD are shown in (b). The vertical and horizontal directions in (a) become the red and green collinear point clusters in (c). Each point cluster in the dual space corresponds to the almost collinear edge segments of the checkerboard. The spread of the cluster is due to the inaccuracy in the LSD process. So the points in dual space do not completely focus at one position but appear scattered.

3.2 Grouping the Dual Points and Getting Vanishing Points

Usually the parallel lines of the architectural buildings are in three axial directions, forming 2 to 3 vanishing points by the camera's perspective projection. In the dual space, the appearance is 2 or 3 collinear point clusters. In this paper, we propose RANSAC and Gustafson-Kessel (GK) 1617 methods for grouping the dual points and use RANAC to fit lines in each group.

RANAC and GK each has its respective advantages and disadvantages. RANSAC is fast but hard to determine the threshold which may produce wrong groupings. GK is relatively more accurate but slow especially when using random initialization. Deciding the number of groups may also be problematic sometimes. In this paper, we combine these two methods as follows: we use RANSAC as an initialization step, and use GK to refine the grouping iteratively. Afterwards, we use RANSAC again within each group to find the straight lines. Below is the pseudo code:

- Loop
  - ✧ Grouping the point clusters using RANSAC
  - ✧ Refine the previous grouping using GK method
  - ✧ Fit a straight line using RANSAC in the largest group
  - ✧ Output the straight line coefficients
  - ✧ Remove the inliers from the point clusters
- Until no more meaningful groups can be found in the remaining points

Let the detected line equations be

$$a_{1,2,3}x + b_{1,2,3}y + c_{1,2,3} = 0 \tag{5}$$

Then, the three vanishing points are $(a_{1,2,3}, b_{1,2,3}, c_{1,2,3})^T$ respectively. Notice the homogeneous formulation here. It allows for dealing with the vanishing point at infinity.

3.3 Calculating the Rotation Matrix and Focal Length from Vanishing Points

The pinhole camera model is

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim M\begin{bmatrix} R & T \end{bmatrix}\begin{pmatrix} X \\ 1 \end{pmatrix} \tag{6}$$

Let r_1, r_2, r_3 be the column vectors of R, $(a_{1,2,3}, b_{1,2,3}, c_{1,2,3})^T$ the vanishing points of the three axial directions in homogeneous coordinates. We have:

$$\begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} \sim \begin{bmatrix} M & 0 \end{bmatrix}\begin{bmatrix} r_1 & r_2 & r_3 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = Mr_1,$$ Thus,

$$r_{1,2,3} \sim M^{-1}\begin{pmatrix} a_{1,2,3} \\ b_{1,2,3} \\ c_{1,2,3} \end{pmatrix} \tag{7}$$

In case only two directions are observed, the third one can be obtained by $r_3=r_1\times r_2$. Since these vectors are recovered up to a scale, they should subsequently be normalized. In practice, it is often safe to assume that the camera's principal axes are orthogonal. Therefore we use the model

$$M = \begin{bmatrix} f & 0 & C_x \\ 0 & f & C_y \\ 0 & 0 & 1 \end{bmatrix},$$ Thus,

$$M^{-1} = \begin{bmatrix} 1/f & 0 & -C_x/f \\ 0 & 1/f & -C_y/f \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Knowing $r_1 \cdot r_2=0$, we have, $\begin{pmatrix} \frac{a_1 - c_1 C_x}{f} & \frac{b_1 - c_1 C_y}{f} & c_1 \end{pmatrix} \begin{pmatrix} \frac{a_2 - c_2 C_x}{f} \\ \frac{b_2 - c_2 C_y}{f} \\ c_2 \end{pmatrix} = 0$

which results in

$$f = \sqrt{-\left(\frac{a_1}{c_1} - C_x\right)\left(\frac{a_2}{c_2} - C_x\right) - \left(\frac{b_1}{c_1} - C_y\right)\left(\frac{b_2}{c_2} - C_y\right)} \tag{9}$$

where $c_1, \; c_2 \neq 0$, i.e., the two vanishing points must not be at infinity which means that the camera must have certain tilt angle with respect to the object in the scene. We assume $C_x=W/2$, $C_y=H/2$ where W, H are the width and height of the image respectively. If all three axial directions are observed, then two additional estimations of the focal length are available from $r_3 \cdot r_2=0$, and $r_1 \cdot r_3=0$. We can use the average as the final estimation. We can use the homography matrix to rectify images as followed (10).

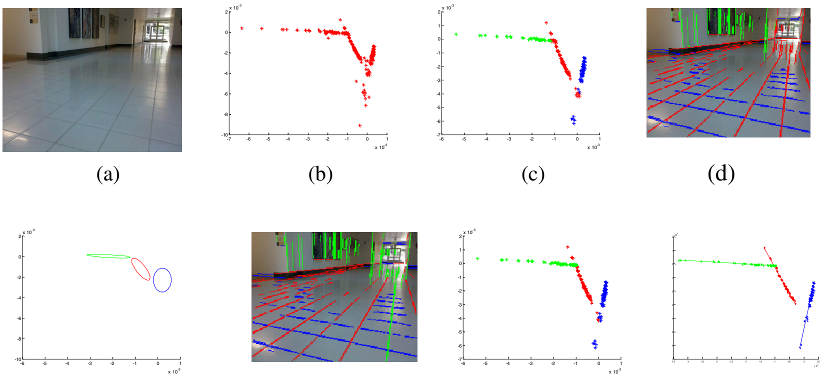
$$H = MR^T M^{-1} \tag{10}$$

The validity of the detected vanishing points can be judged by observing the orthogonality of the rectangular features in the rectified image.

4 Experimental Results

4.1 Dual Points Grouping

We have tested the RANSAC method and the improved GK method to grouping dual vanishing points as follows.



(a) (b) (c) (d)

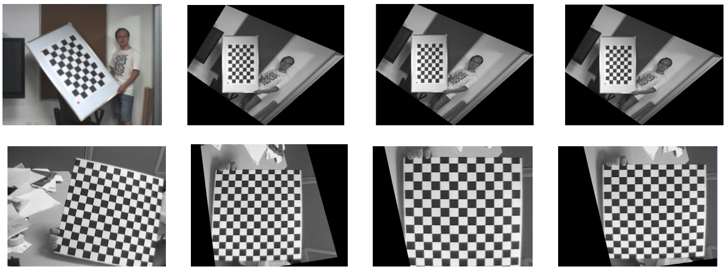
(e) (f) (g) (h)

Fig. 3. Visualization of RANAC and GK grouping. (a) is the input image. (b) is the converted dual points. (c) and (d) show the result of grouping by RANSAC. (e) show the process of grouping by GK initiated by the previous RANSAC. It represent the cluster covariance matrices. (f) is Visualization of GK grouping result. (g) is the result of grouping by GK. (h) shows the final fitted lines. The vanishing points are actually the coefficients of the line equations.

We can see from Figure 3 that the two methods have different grouping effects. Those lines at the upper right of (d) were grouped wrongly (colored in red but should have been green) due to the fact that in the dual space (c), they are close to the intersection of the red and the green groups. Notice that in (c) the red group is atop of the green group near the intersection. This grouping error did not happen in the GK method as shown in (f) and (g). Notice differently from (c), in (g) the green group is atop the red one. However the GK method failed to group the long line towards the right side (green one which should have been grouped as red). But this error somehow "makes sense" because the line indeed appears parallel to those green ones above it.

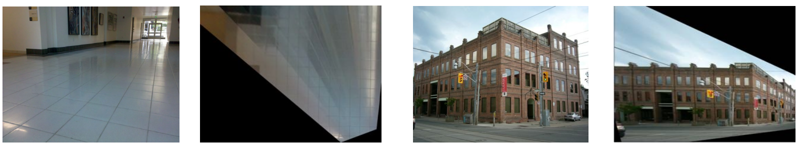
4.2 Image Rectification

In order to validate the detected vanishing points, the experiments of image rectification were carried out using the homography matrix given in (10). We compare our results with the homography matrices obtained from two other methods: 1) M and R provided by camera calibration with the MATLAB calibration toolbox, 2) M and R provided by camera calibration with the method implemented in OPENCV 15. Figure 4 demonstrates that our method using a single image produces comparable results with those calibration methods using multiple images. Figure 5 provides two more rectification examples: one indoor image and one outdoor building image.



(a) Original images (b) By OpenCV (c) By Matlab toolbox (d) By our method

Fig. 4. The image rectification experiment performed on checkerboard images



(a) Indoor image (b) Rectified by our method (c) Outdoor image (d) Rectified by our method

Fig. 5. Image rectification experiment performed on indoor and outdoor images

5 Conclusions

In this paper, we proposed a vanishing point detection method by fitting lines in the dual space based on the point-line duality principle. This method can detect vanishing points in distinct directions and group them automatically. Experiment results verified the validity of the proposed method. Image rectification experiments further confirmed the feasibility of calculating camera intrinsic parameters and rotation matrix using the detected vanishing points.

One shortcoming of the proposed method is the shifting of the image in order to avoid zero denominators in the dual space transform. We plan to address this issue in the future. We also plan to device a more robust grouping algorithm in dealing with more complicated images.

Acknowledgment. This work is supported in part by the National Natural Science Fund of China under grant 61070147 and 61171146, Shenzhen Scientific & Research Development Fund under grant JC201005270331A and Shenzhen Internet Industry Development Fund under grant ZD201006110037A.

References

1. Guillou, E., Meneveaux, D., Maisel, E., Bouatouch, K.: Using Vanishing Points for Camera Calibration and Coarse 3D Reconstruction from a Single Image. The Visual Computer (S0178-2789) 16(7), 396–410 (2000)
2. Foroosh, H., Cao, X., Balci, M.: Metrology in Uncalibrated Images Given One Vanishing Point. In: IEEE International Conference on Image Processing, pp. III-361-4. IEEE, USA (2005)
3. Li, B., Peng, K., Ying, X., Zha, H.: Simultaneous Vanishing Point Detection and Camera Calibration from Single Images. Advances in Visual Computing (2010)
4. Almansa, A., Desolneux, A., Vamech, S.: Vanishing Point Detection without Any A Priori Information. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(4), 502–507 (2003)
5. Tuytelaars, T., Proesmans, M., Van Gool, L.: The Cascaded Hough Transform. In: Proc. Int'l Conf. Image Processing (ICIP 1997), vol. 2, pp. 736–739 (1997)
6. Rother, C.: A New Approach for Vanishing Point Detection in Architectural Environments. In: Proc. British Machine Vision Conf. (2000)
7. Barnard, S.T.: Interpreting perspective images. Artificial Intelligence 21, 435–462 (1983)
8. Lutton, E., Maitre, H., Lopez-Krahe, J.: Contribution to be the Determination of Vanishing Points using Hough Transform. IEEE Transaction on Pattern Analysis and Machine Intelligence 16(4), 430–438 (1994)
9. Minagawa, A., Tagawa, N., Moriya, T., Gotoh, T.: Line Clustering with Vanishing Point and Vanishing Line. In: Proceedings of International Conference on Image Analysis and Processing, pp. 388–393 (1999)
10. Ebrahimpou, R., Rasoolinezhad, R., Hajiabolhasani, Z., Ebrahimi, M.: Vanishing point detection in corridors: using Hough transform and K-means clustering. IET Computer Vision, 40–51 (2012)
11. Schmitt, F., Priese, L.: Vanishing Point Detection with an Intersection Point Neighborhood. In: Brlek, S., Reutenauer, C., Provençal, X. (eds.) DGCI 2009. LNCS, vol. 5810, pp. 132–143. Springer, Heidelberg (2009)

12. Duda, R.O., Hart, P.E.: Use of the Hough Transformation to Detect Lines and Curves in Pictures. Communications of Association for Computing Machinery 15(1), 11–15 (1972)
13. Grompone von Gioi, R., Jakubowicz, J., Morel, J., Randall, G.: LSD: A Fast Line Segment Detector with a False Detection Control. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(4), 722–732 (2010)
14. Antone, M.E., Teller, S.: Automatic Recovery of Relative Camera Rotations for Urban Scenes. In: Proc. CVPR 2000, pp. 282–289 (2000)
15. Bradski, G., Kaehler, A.: Learning OpenCV, pp. 432–437. O'Reilly Media, Inc. (2008)
16. Gustafson, D.E., Kessel, W.C.: Fuzzy clustering with a fuzzy covariance matrix. In: Proc. IEEE CDC, San Diego, CA, USA, pp. 761–766 (1979)
17. Babuska, R., van der Veen, P.J., Kaymak, U.: Improved covariance estimation for Gustafson–Kessel clustering. In: Proc. of the IEEE Internat. Conf. on Fuzzy Systems, pp. 1081–1085 (2002)