An Approach to Finding and Refinement Planes in 3D Points Cloud, Obtained Under 3D Recovery from Image Set

Ekaterina V. Semeikina[*], Dmitry V. Yurin[**]

[*] Moscow State University, Faculty of Computational Mathematics and Cybernetics, Lab. of Mathematical Methods of Image Processing.
[**] Institute of Computing for Physics and Technology

semeikinae@gmail.com, yurin_d@inbox.ru

Abstract

An algorithm of structure analysis of an input 3D points cloud and planes discrimination is proposed. The algorithm is based on the hierarchical and randomized Hough transform. The algorithm allows detecting image regions corresponding to planes instead of separate points, and partially converting 3D model from cloud of points to refined mesh.

Keywords: 3D recovery, points cloud structure analysis, planes detection, Hough transform, homography refinement, mesh refinement.

Figure 1. *Example of stereo pair of images used to find 3D cloud of points.*
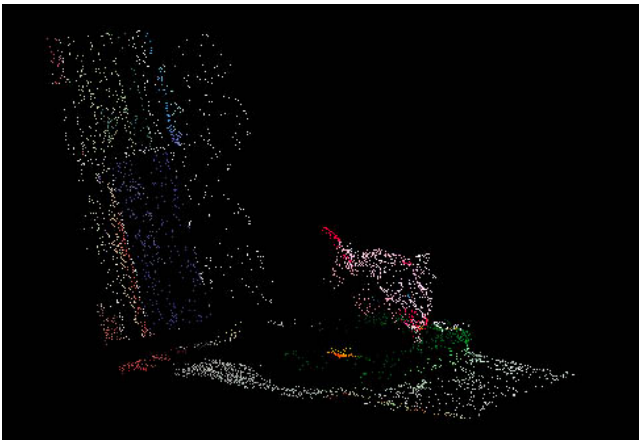

Figure 2. *Input 3D points cloud.*

1. INTRODUCTION

There are a lot of algorithms that allow finding feature points on images, establishing correspondences between such points on different images and determining 3D coordinates of them [1-5]. In the visualization systems three-dimensional scenes are generally represented as mesh surfaces because a set of 3D points describes an object insufficiently. But direct application of Delaunay triangulation [6, 7] or construction of convex hulls [6] brings to an inadequate mesh model. Particularly, errors of 3D coordinates of points and a lack of feature points near dihedral angles bring to effect of a "wrinkled" model.

In this work the following approach is proposed. Inputs of the algorithm are a set of images (Figure 1), feature points founded on them, established correspondences between points, recovered 3D coordinates of these points (Figure 4) and their errors estimates. To form hypotheses of planes the Hough transform [8] for planes was applied to 3D cloud of points. Then the homography matrices [1] were found by feature points on images related to the hypothesis. The homography matrices are refined to subpixel accuracy using algorithms based on optical flow [9]. Using these matrixes images are transformed to the same position of the hypothetic plane and the difference images are computed. If intensity on difference image is low and dispersion on initial images is high in vicinity of feature points related to the plane, hypothesis can be considered to be right. For such hypotheses satisfying connected domains are picked out. Points related to such plane fragment are replaced in 3D model by polygons.

Numerical experiments were executed with the data had been found by the algorithm [5] from stereo pair of images (Figure 1).

2. HYPOTHESIZING

To form hypotheses of planes the Hough transform **[8]** was applied to 3D cloud of points. To increase a speed of calculations and to save a memory the hierarchical and randomized version of it was used.

2.1 Parameterization of plane

A plane in three-dimensional space

$$Ax + By + Cz + D = 0 \quad , \quad S = \sqrt{A^2 + B^2 + C^2} \neq 0 \quad (1)$$

can be determined by three points which do not situated on one line.

$$
\begin{aligned}
A &= y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2) \\
B &= x_1(z_3 - z_2) + x_2(z_1 - z_3) + x_3(z_2 - z_1) \\
C &= x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)
\end{aligned}
\quad (2)
$$

$$D = x_1 y_2 z_3 + x_3 y_1 z_2 + x_2 y_3 z_1 - x_3 y_2 z_1 - x_1 y_3 z_2 - x_2 y_1 z_3,$$

where (x_i, y_i, z_i), $i = \overline{1,3}$ - coordinates of points situated on the plane.

Parameters A, B, C, D can be determined up to scale, therefore it's possible to parameterize a plane by directional cosines.

$$x \cos\alpha_x + y \cos\alpha_y + z \cos\alpha_z = \rho$$
$$\cos\alpha_z = \pm\sqrt{1 - \cos^2\alpha_x - \cos^2\alpha_y} \qquad (3)$$

where $\cos\alpha_x$, $\cos\alpha_y$ are directional cosines, ρ is the distance between the origin of coordinate system and the plane.

Direction of a plane normal (choice of its surface) is not essential, therefore sign «+» was used in formula (3). Thus, a plane can be parameterized by the following values:

$$\cos\alpha_x = \frac{A}{S} \quad , \quad \cos\alpha_y = \frac{B}{S} \quad , \quad \rho = \frac{D}{S} \qquad (4)$$

The parameter space is bounded:

$$|\cos\alpha_x|, |\cos\alpha_y| \le 1, \quad |\rho| < \sqrt{2}\max_i(|x_i|, |y_i|, |z_i|), \quad i = 1..N,$$

where i – index of an any feature point

therefore it can be divided into cells and each of them corresponds to range of parameters. The division of parameters space is uniform. Plane searching was executed by consecutive detailing of division of parameter space, in other words by consecutive approximation of the sought plane parameters.

2.2 Detailing of parameter space division

If a number of points is large, searching of all sets of three points is too long. Therefore, on each detailing step, if $N > M$ (N – a number of input or found on the previous step points), then βN sets of three points were considered, where M: $M(M-1)(M-2) \approx M^3 = \beta \cdot N$.

For each such set, if the points were not close to each other and one line, plane parameters were calculated with the formulas (2), (1), (4). If the appropriate cell of accumulator hadn't contained these points yet, they were placed into it.

For each cell, containing more, then $T_m < T_{m-1}$ points (m – a number of step), next detailing step was executed. The threshold T_n (n – number of the last step) describes a minimal size of a plane that should be found, and it makes sense to be not less than $\max(5, 0.1 \cdot N)$. An accumulator, detailing the cell, was build only for a range of parameters, in which planes, related to cell on the previous step, occurred.

A number of steps n and a size of an accumulator $\left\{S_{\cos_x}, S_{\cos_y}, S_\rho\right\}$ are correlated in the following way (5):

$$\sqrt{1 - \left(S^n_{\cos_x}\right)^2} \max_{1 \le i \le N}(|x_i|) \approx \varepsilon_x,$$
$$\sqrt{1 - \left(S^n_{\cos_y}\right)^2} \max_{1 \le i \le N}(|y_i|) \approx \varepsilon_y, \qquad (5)$$

$$\frac{\sqrt{2}\max_{1 \le i \le N}(|x_i|, |y_i|, |z_i|)}{S^n_\rho} \approx \max(\varepsilon_x, \varepsilon_y, \varepsilon_z)$$

where i – index of an any feature point.

In the numerical experiments three detailing steps were executed, the threshold $T_3 = 0,1N$, $\beta = 25$, and a size of accumulators was $2^2 \times 2^2 \times 2^3$ on every step.

2.3 Eliminating of outliers

Let's denote a plane found by three point as P_3. A centre of mass of parameters of planes $P_{3,i}$ (i – a number of plane), which had fallen into accumulator cell, corresponds to an average plane. Let's denote it as $P_{cm} = \left\{\cos\alpha_x, \cos\alpha_y, \rho\right\}$.

From accumulators of the last detailing step, which contain more than the threshold T_n points, points were deleted, if the distance between them and the plane P_{cm} is larger than $\max(\varepsilon_x, \varepsilon_y, \varepsilon_z)$, where $\varepsilon_x, \varepsilon_y, \varepsilon_z$ - errors of points coordinates. If after eliminating of all such points a number of points in the cell remained larger than T_n, then P_{cm} can be considered as a hypothetic plane.

3. CHECKING OF HYPOTHESES

3.1 Finding of a homography matrix

To check hypothesis a homography matrix was found by points, related to this hypothesis. Normalized DLT algorithm [1, alg .4.2, p. 109] was used for it. Only points, which were related by Hough transform to the hypothesis, were used to find homography, therefore using of the robust algorithms like RANSAC for eliminating of outliers is not required.

3.2 Image registration

The first image was transformed using obtained homography matrix. Bilinear interpolation was used for it to increase computational effectiveness and to avoid artifacts like the Gibbs effect.

Figure 3. *Difference images corresponding to the different hypothetic planes.*

Then a difference image (Figure 3) of the second and transformed first images was computed and final matching of plane areas was executed using method Shi-Tomasi [9]. Planes had been found

sufficiently precisely on the Hough transformation stage, therefore sought correction was little enough to be approximated by the affine model [9]. If the hypothetic 3D plane really exists on the images, then corresponding area on the difference image was almost black unlike areas, witch are not on this 3D plane.

4. EXTRACTION OF PLANE AREAS ON IMAGES

To pick out areas, corresponding to the found plane, the Bresenham filling algorithm [10] was executed on difference image. Previously difference image was smoothed by convolution with the gaussian function with the large σ ($\sigma = 15$ was used) to smooth an edge of a black area and at the same time to smooth small inaccuracy of registration of this area. Then filling was carried out for areas of difference image where intensity was close to zero and dispersion on initial image was large (Figure 4).

As output of the Bresenham algorithm edge points of filled area was also obtained and some of the edge points were selected to be apexes of a polygon in the model. In the given example (Figure 6) every 20th point was selected.
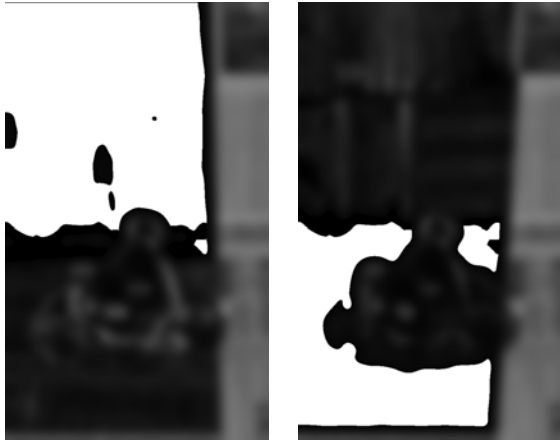

Figure 4. *Extraction of image areas corresponding to the planes.*

To include the area, found in image coordinates, into the model it was necessary to find 3D coordinates of its edge point.

At first homography matrix, transforming coordinates of the feature points on the image into 2D coordinates on the found plane $P = \{\cos\alpha_x, \cos\alpha_y, \rho\}$ in three-dimension space, was found in the following way. The feature points on image are known, their 3D coordinates are also known. But because of errors in coordinates feature points (x, y, z) in 3D space don't lie on the found plane P exactly. Therefore firstly they were projected to this plane P. Then their coordinates (x', y') on the plane P was calculated, for it coordinates system was graded into such system where the plane P coincides with the plane Oxy:

$$x' = t_{11}(x - \rho\cos\alpha_x) + t_{21}(y - \rho\cos\alpha_y) + t_{31}(z - \rho\cos\alpha_z)$$
$$y' = t_{12}(x - \rho\cos\alpha_x) + t_{22}(y - \rho\cos\alpha_y) + t_{32}(z - \rho\cos\alpha_z) \quad (6)$$

where $\quad t_{31} = 0, \qquad t_{32} = -\sqrt{\cos\alpha_x^2 + \cos\alpha_y^2}$.

$$t_{11} = \frac{\cos\alpha_y}{\sqrt{\cos\alpha_x^2 + \cos\alpha_y^2}}, \quad t_{21} = -\frac{\cos\alpha_x}{\sqrt{\cos\alpha_x^2 + \cos\alpha_y^2}},$$

$$t_{12} = -\frac{\cos\alpha_x \cdot \cos\alpha_z}{\sqrt{\cos\alpha_x^2 + \cos\alpha_y^2}}, \quad t_{22} = -\frac{\cos\alpha_y \cdot \cos\alpha_z}{\sqrt{\cos\alpha_x^2 + \cos\alpha_y^2}}.$$

Then homography matrix cuold found as in 3.1, and coordinates of edge point $(\overline{x}, \overline{y})$ on the plane P could be found using this matrix.

Plane parameters were found, so 3D coordinates (x, y, z) of the edge point could be obtained by the formulas (7):

$$x = t_{11}\overline{x} + t_{12}\overline{y} + \rho\cos\alpha_x,$$
$$y = t_{21}\overline{x} + t_{212}\overline{y} + \rho\cos\alpha_y, \quad (7)$$
$$z = t_{31}\overline{x} + t_{32}\overline{y} + \rho\cos\alpha_z$$

Now the corresponding polygon could be included into the model (Figure 6).
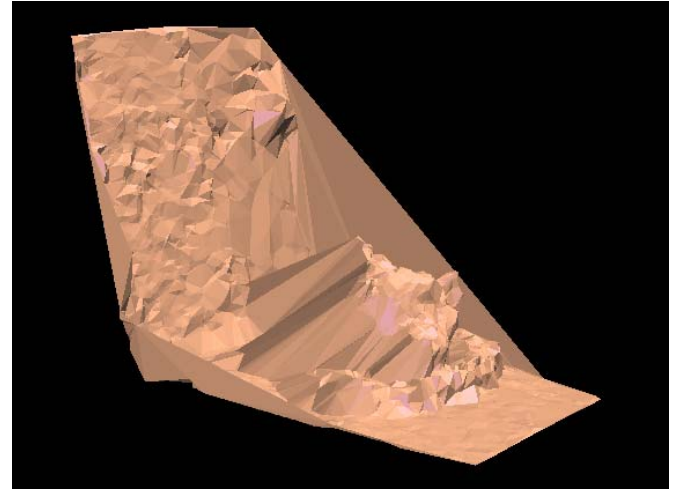

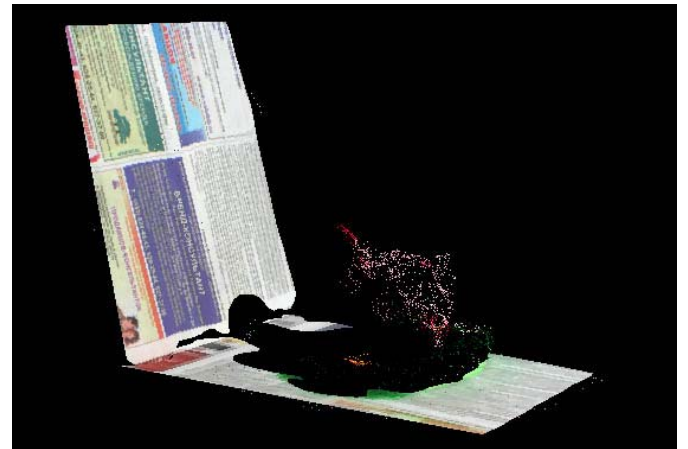***Figure 5.*** *Model, obtained by triangulation.*

Figure 6. *Model, constructed using proposed algorithm.*

5. CONCLUSION

5.1 Complexity

Complexity of the algorithm depends on a number of input feature points (because their searching was executed while hypothesis building) and on a real number of planes on images (because the last stages were executed for every plane). While testing the number of points were 1500-4000 and the number of planes was 1-3. A typical time of work of the non-optimized algorithm was 0.5-1.5 minutes.

5.2 Results

Testing on the data, obtained by the method [5], showed that the algorithm determines belonging of points to planes correctly. Refinement of registration parameters and analysis of a difference image allow removing of model «wrinkleness» and widening of an area, corresponding to each plane, in comparison with a triangulation (Figure 5). Points, that don't satisfy a hypothesis, often turn out to be separated into clusters, allowing fragment-serial refinement of model.

As a further development of the work finding of more complicated models (like B-splines) for such clusters of point is intended. After a building of approximations of surfaces, containing practically all 3D point, solving of the segmentation problem is intended like [11, 12], but starting from the good initial approximation and without confining within a case of rectify stereo. In a case of processing of more than two images, if more than two non-parallel planes were found, refinement of epipolar geometry and camera pose [1,13,14], based on refined homographies (3.2), is a subject of interest.

6. ACKNOWLEDGEMENTS

7. LITERATURE

[1] R. Hartley, A. Zisserman. *Multiple View Geometry in Computer Vision. Cambridge University Press 2004, 672 p., ISBN: 0521540518.*

[2] David G. Lowe. *"Object recognition from local scale-invariant features," International Conference on Computer Vision, Corfu, Greece, September 1999, -P. 1150-1157.* http://www.cs.ubc.ca/~lowe/papers/iccv99.pdf.

[3] A. Baumberg. *Reliable feature matching across widely separated views. //In Proc. CVPR 2000, -P. 774--781.* http://citeseer.ist.psu.edu/baumberg00reliable.html.

[4] K. Mikolajczyk, C. Schmid. *Scale & Affine Invariant Interest Point Detectors. //International Journal of Computer Vision, October 2004, –V. 60, -No. 1, –P. 63—86.* http://www.robots.ox.ac.uk/~vgg/research/affine/det_eval_files/mikolajczyk_ijcv2004.pdf.

[5] D.B. Volegov, D.V. Yurin. *Finding disparity map via image pyramid. //In proc. of 17-th International Conference on Computer Graphics and Vision GraphiCon'2007. Moscow State University, Moscow, Russia, June 23-27, 2007, -P. 223-226.* http://www.graphicon.ru/2007/proceedings/Papers/Paper_81.pdf.

[6] C.B.Barber, D.P.Dobkin, H.Huhdanpaa. *The Quickhull Algorithm for Convex Hulls //ACM Transactions on Mathematical Software 1996, -V. 22, -No. 4, -P. 469-483.* http://citeseer.ist.psu.edu/article/barber95quickhull.html

[7] J.R.Shewchuk. *Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator //Proc. 1st Worksh. Appl. Computational Geometry, ACM, 1996, pp.124-132.* http://www.cs.cmu.edu/~quake/tripaper/triangle0.html

[8] R.Gonzalez, R.Woods. *Digital Image Processing. Prentice Hall 2002, 793p., ISBN: 0201180758.*

[9] J.Shi, C.Tomasi. *Good features to track //IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94) (Seattle), June 1994, -P. 593-600.* http://citeseer.ist.psu.edu/shi94good.html.

[10] R.Wilton. *Programmer's Guide to PC and Ps/2 Video Systems: Maximum Video Performance Form the Ega, Vga, Hgc, and McGa //Microsoft Press, December 1987, 544 p. ISBN-10: 1556151039, ISBN-13: 978-1556151033.*

[11] S. Birchfield, C. Tomasi. *Multiway cut for stereo and motion with slanted surfaces. //Proc. of the IEEE International Conference on Computer Vision, September 1999, -P. 489-495.* http://vision.stanford.edu/%7ebirch/publications/multiwaycut_iccv1999.pdf

[12] M.Lin, C.Tomasi. *Surfaces with Occlusions from Layered Stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence, -V. 26, -No. 8, August 2004.*

[13] H.C.Longuet-Higgins *A Computer Algorithm for Reconstructing a Scene from Two Projections*, *Nature, -V. 293, -P. 133–135, September 1981.*

[14] B. K. P. Horn. *Recovering Baseline and Orientation from Essential Matrix , 1990.* http://ocw.mit.edu/NR/rdonlyres/Electrical-Engineering-and-Computer-Science/6-801Fall-2004/B9D82753-1D93-42BD-AB30-9CCFBF7073AC/0/essential.pdf.

About the authors

Ekaterina V. Semeikina is 4th-year student at Laboratory of Mathematical Methods of Image Processing, Chair of Mathematical Physics, Faculty of Computational Mathematics and Cybernetics, Moscow Lomonosov State University. Her contact email is semeikinae@gmail.com

Dmitry V. Yurin, PhD, is a senior scientist at Institute of Computing for Physics and Technology and at Laboratory of Mathematical Methods of Image Processing, Chair of Mathematical Physics, Faculty of Computational Mathematics and Cybernetics, Moscow Lomonosov State University. His contact email is yurin_d@inbox.ru