A Point and Line Features Based Method for Disturbed Surface Motion Estimation

Xiang Li and Yue Zhou

Institute of Image Processing and Pattern Recognition
Shanghai Jiao Tong University
No 800, Dongchuan Rd, Shanghai, China, 200240
{lostxine,zhouyue}@sjtu.edu.cn

Abstract. Calculating the motion of disturbed surface such as a reflective monochromatic one is often a difficult part, especially when using single feature based method. The error introduced from the feature extraction and matching will gradually accumulate into a larger final error. For a texture-less surface, the number of features makes the situation even more challenging. In this paper, point and line features from stereo sequences are combined to estimate 3D motion of disturbed surfaces. Taking the advantage of feature combination by two-stage iterative optimization and multiple filtering, the motion of surfaces can be estimated accurately, even under little motion blur. This paper also explored the relationship between measurement accuracy and object motion mode. This may provide a reference for the design of a vision based motion measuring system.

Keywords: Motion estimation, combined features, iterative optimization

1 Introduction

Estimating the motion of objects is a classic subject in computer vision. It has many applications such as human-computer interaction, robot navigation. The motion estimation problem shares most mathematical theorems[4] with Simultaneous Localization Mapping(SLAM) or Structure from Motion(SFM). That is, the motion of camera can be estimated by SLAM or SFM algorithms. [11][8][3][13] are several classic and reliable solutions. However, most of solutions depend on single fundamental feature and may not work in texture-less or blur scenes. Reflective objects will also affect the performance. Therefore, motion estimation of disturbed objects, which contains reflective and texture-less surfaces, is still a challenging problem that many authors have started to focus on over the last few years.

The work of this paper is inspired by [3] and [9]. In order to obtain accurate measurement, the most basic features such as points and lines which are supported and proved by mathematic should not be ignored. Tips like filtering and optimization are introduced to overcome disturbed surfaces and unstable features. In this paper, we proposed a novel motion estimation method for stereo system: converting traditional 3D-2D problem into 3D-3D and combining points with lines in 3D space. With the help of constraint from local color and 3D position relationship, an iterative optimization method like 4D ICP algorithm[7], handing both points and lines, are proposed. Bundle adjustment[12] and Random Sample Consensus(RANSAC) filtering were also involved. Our main contributions include: (i) tried to solve estimation problem with the help of 3D feature pairs; and (ii)combined point and line features.

Here comes the outline of this paper. Background, motivation and main idea are proposed first in this section. Then we will introduce the pipeline of proposed method. Detail discussions on point and line constraints come after the pipeline. Post processing containing filtering goes later. Finally, experiments are presented in section 3, followed by analysis and conclusion.

2 Proposed Method

In this section, we present the proposed motion estimation algorithm which using both point and line features from stereo camera. Fig. 1 shows the pipeline. Firstly, image sequence passed through the preprocessing, lines and points in 3D space were extracted and reserved for motion estimation. Then, we applied two-stage iterative optimization, in order to calculate the rotation and translation separately. After that, post processing, including bundle adjustment and RANSAC filtering, gathered several adjacent frames and gave final prediction of 6-dof motion. Finally, 'inertia' kept the result and helped estimation at the next frame.

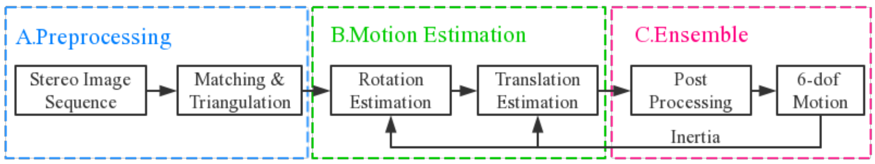

Fig. 1. Pipeline of proposed method

2.1 Preprocessing

The purpose of preprocessing is to extract the features in image sequence and to calculate the specific positions of the features in 3D space. As for stereo camera, with the help of offline calibration, SGBM[5] could generate reliable disparity map, but features needed to be extracted in region that disparity map had covered. The extraction of 3D feature is shown in green zone of Fig. 2. After extraction, matching and filtering began with features from different moments. That is the orange zone in Fig. 2.

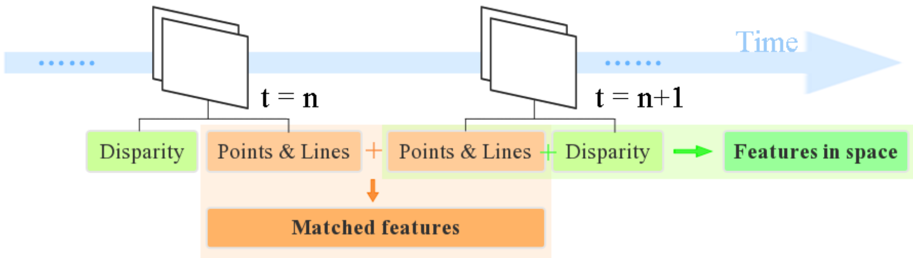

Fig. 2. Flow of preprocessing

Feature points were located by FAST[10] and described by SURF[1] to lower time complexity. Feature lines were generated from Canny[2]. Matching algorithm of lines came from Jia et al[6].

2.2 Motion estimation

In our solution, motion estimation module was designed handle 3D-3D matched points and lines. A new cost function with physical meaning, was introduce to cover both point and line features. In order to reduce the difficulty of solving and achieve convergence faster, the cost function, which includes 6 degrees of freedom, was split into two little ones: rotation cost function and translation cost function. Solving these two sub-problems was what we called two-stage iterative optimization.

Rotation estimation At this stage, the desired solution is rotation vector $\vec{\mathbf{r}}$ containing 3-dof. Fig. 3 shows several variables will be used. Suppose N point and M line features from $t = n$ are marked in blue and corresponding features in $t = n + 1$ are marked in red. The green line between j^{th} point pair is called $d_j(\vec{\mathbf{r}})$. It is a vector from a point in n but after $\vec{\mathbf{r}}$ transformation, pointing to corresponding point in $n + 1$. Rotation cost function was made of four parts like Eq(1).

$$\arg\min_{\vec{\mathbf{r}}} \quad (c_p\theta_p(\vec{\mathbf{r}}) + c_l\theta_l(\vec{\mathbf{r}}) + c_i\theta_{(\vec{\mathbf{r}}-\mathbf{r'})} + c_n\|\vec{\mathbf{r}}\|) \tag{1}$$

In Eq(1), c_p,c_l,c_i,c_n are four constant weights. $\theta_p(\vec{\mathbf{r}})$ is the cost from point features and $\theta_l(\vec{\mathbf{r}})$ comes from line features, which will be introduced later. $\theta_{(\vec{\mathbf{r}}-\mathbf{r'})}$ represent the angle between this rotation vector $\vec{\mathbf{r}}$ and last one $\mathbf{r'}$. That is the representation of 'inertia' part. $\|\vec{\mathbf{r}}\|$ is designed to constrain the range of $\vec{\mathbf{r}}$, otherwise $\|\vec{\mathbf{r}}\|$ may be larger than 2π or smaller than 0.

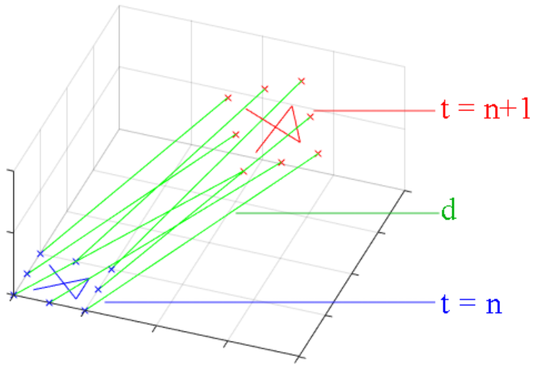

Fig. 3. Variables in the cost function

The cost from point features $\theta_p(\vec{\mathbf{r}})$ can be divided into 2 parts like Eq.(2). c_1, c_2 are both constants. The first part represents the average angle between d_j and their mean. Because when two point clouds towards the same direction, all the connections between corresponding points should be parallel to each other. c_2 is followed by the standard deviation of d_j's length. This part prevents degradation of the first part due to symmetry.

$$\theta_p(\vec{\mathbf{r}}) = c_1 \frac{1}{N} \sum_{j=1}^{N} \|\theta_{(d_j - \bar{d})}\| + c_2 \sqrt{\frac{1}{N} \sum_{j=1}^{N} \|d_j\|} \tag{2}$$

Suppose there were matched lines l_m, l'_m at n and $n+1$. They would contribute their angle $\theta_{(l,l')} = \theta_m$ to the cost from line features $\theta_l(\vec{\mathbf{r}})$. So:

$$\theta_l(\vec{\mathbf{r}}) = c_3 \frac{1}{M} \sum_{m=1}^{M} \|\theta_m\| \tag{3}$$

Translation estimation After the rotation estimation and transformation, features from two moments would share the same orientation. It would be easy to estimate translation vector $\vec{\mathbf{t}}$ using Eq.(4).

$$\underset{\vec{\mathbf{t}}}{\arg\min} \sum_{j=1}^{N} \|d_j(\vec{\mathbf{t}})\| \tag{4}$$

The code of motion estimation are available online[1].

[1] https://github.com/LostXine/motionEstimation

2.3 Ensemble

The ensemble module contains two main parts in order to achieve higher precision. Bundle adjustment[12] helped localize points; RANSAC filtering picked out the frames which held an obvious error. At the same time, number of inner frames reflected the robustness of motion estimation algorithm.

3 Experiment

To evaluate our algorithm, we built a indoor rotating platform with cube model on it and acquired stereo image sequences. Fig. 4 showed design of the experimental device. For every sequence, the cube rotated around a fixed axis at a constant speed. We obtained several different sequences by changing the combination of rotation speed and sample interval. In Fig. 4, baseline of stereo rig $b = 18cm$; distance from the rig to rotating cube $d \approx 200cm$; rotation speed $\omega \in [1, 14]$deg/s and surface of the cube was made of reflective monochrome metallic material. Each camera produces a color picture size of $1280 \times 1024 \times 3$ pixels, and the side length of the cube is approximately 440 pixels in the image.

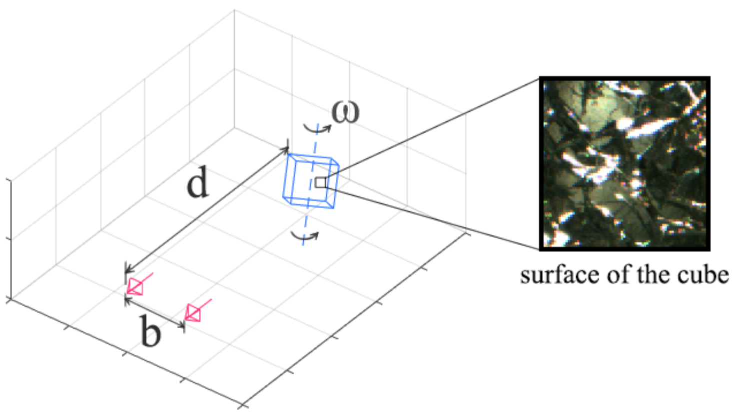

Fig. 4. Design of experimental device

The experiment used five unique sequences, and the error of rotation speed was the highest priority. The basic information for each sequence is shown in Table 1.

For the same pipeline, we had implemented two methods using different features: one used only points and the other used both points and lines. Comparison between two methods are shown in Table 2. The error reflects accuracy of the algorithm, the smaller the better. Inner is the proportion of reserved frames after filtering, reflecting the robustness of the algorithm, the higher the better.

Table 1. Information of test sequences

Index	ω(deg/s)	Sample interval(s)	Rotation speed(deg/frame)	Length (frame)
1	1	1.00	1.0	402
2	6	0.25	1.5	276
3	1	2.00	2.0	201
4	6	0.50	3.0	138
5	14	0.25	3.5	206

Table 2. Comparison of different feature based methods

Rotation speed(deg/frame)	Error(deg/frame)		Inner(%)	
	Point only	Line and point	Point only	Line and point
1.0	0.4688	0.4066	79.05	79.30
1.5	0.2881	0.2631	73.19	80.07
2.0	0.3417	0.2993	82.00	85.00
3.0	0.0679	0.0144	57.66	57.66
3.5	0.3200	0.1896	44.39	56.10

In general, both methods achieved the motion estimation task for the disturbed surface and prove the feasibility of our pipeline. In this experiment, there were about 100-700 available feature point pairs between every two frames. In addition, there were 0-25 pairs of lines, although they were few, but for better performance played a very critical role: (i) absolute error of rotation estimation was reduced. Especially when the motion between frames went larger, this reduction effect is more pronounced. (ii) combined lines helped increase the percentage of inner frames after RANSAC filtering, that means a greater probability of success and better robustness because of the increased signal to noise ratio.

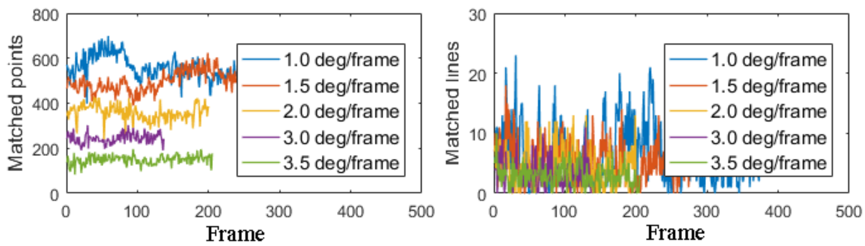

Fig. 5. Number of matched features between frames

An intuitive conclusion is that, with the increase of rotation speed, the reliability(Inner) of algorithm was declining. Because with the increase in angle, reflective surfaces begun to change in brightness even shape. This would increase the difficulty of matching, especially without valid texture information. Fig. 5 supports this view, that degradation of algorithm reliability is actually the decline of matching reliability. But an unexpected phenomenon is that error went up when the rotation between frames was closer to 1.0 deg/frame. Because the angle was tiny, the real distance of the same feature point between two frames was often less than one pixel. At the same time, it would difficult to locate the feature points precisely to superpixel level. Resulting in a larger error could also be understood.

As for the time complexity, our method runs at 0.2-0.5 fps on a PC with Intel i7-6700 CPU @ 3.40GHz. Feature extraction and optimization process took most of the time.

4 Conclusion

In this paper, we proposed a combined line and point feature based method for motion estimation, especially when the surface was disturbed by reflection and other noises. Provided image sequences from stereo camera, all the features would be converted into 3D space after matching and filtering. Then, a brand new cost function including points and lines was optimized. Post processing filtered out invalid frame and gave final output. Experiments showed that our solution could estimate the 6-dof motion stably and accurately without losing much computational efficiency.

However, this method still depends on the fundamental features of the objects. Although we had overcome some of the disturbed reflective surfaces by combining several types of features, for texture-less objects that have a small number of features, this algorithm still has limitations. A method that combines features and shading model may perform better on disturbed objects, and will be done in the future.

Acknowledgments. This work is supported by the National High Technology Research and Development Program of China (863 Program) under Grant 2015AA016402.

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. Computer vision–ECCV 2006 pp. 404–417 (2006)
2. Canny, J.: A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence (6), 679–698 (1986)
3. Feng, Y., Wu, Y., Fan, L.: On-line object reconstruction and tracking for 3d interaction. In: Multimedia and Expo (ICME), 2012 IEEE International Conference on. pp. 711–716. IEEE (2012)

4. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)
5. Hirschmüller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information 2, 807–814 (2005)
6. Jia, Q., Gao, X., Fan, X., Luo, Z., Li, H., Chen, Z.: Novel coplanar line-points invariants for robust line matching across views. In: ECCV (8). pp. 599–611 (2016)
7. Men, H., Gebre, B., Pochiraju, K.: Color point cloud registration with 4d icp algorithm. In: Robotics and Automation (ICRA), 2011 IEEE International Conference on. pp. 1511–1516. IEEE (2011)
8. Nistér, D.: Preemptive ransac for live structure and motion estimation. Machine Vision and Applications 16(5), 321–329 (2005)
9. Pilz, F., Pugeault, N., Krüger, N.: Comparison of point and line features and their combination for rigid body motion estimation. In: Statistical and Geometrical Approaches to Visual Motion Analysis. pp. 280–304. Springer (2008)
10. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. Computer Vision–ECCV 2006 pp. 430–443 (2006)
11. Taylor, C.J., Kriegman, D.J.: Structure and motion from line segments in multiple images. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(11), 1021–1032 (1995)
12. Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle adjustmenta modern synthesis. Vision algorithms: theory and practice pp. 153–177 (2000)
13. Wu, C.: Towards linear-time incremental structure from motion. In: 3DTV-Conference, 2013 International Conference on. pp. 127–134. IEEE (2013)