Monocular Visual Odometry Approach for Trajectory Estimation of a Moving Object Using Ground Plane Geometry

S. Sajikumar
Department of Mathematics
College of Engineering, Trivandrum
Thiruvananthapuram, India
sajikumar.s@cet.ac.in

P. Bhanumathy
Vikram Sarabhai Space Centre
Indian Space Research Organisation
Thiruvananthapuram, India
pb_bhanumathy@vssc.gov.in

A. K. Anilkumar
DSSAM
Indian Space Research Organisation
Bengaluru, India
ak_anilkumar@isro.gov.in

***Abstract*—Trajectory estimation of a moving object using monocular visual odometry approach is an important research direction in computer vision. The moving object can be a car or a robot or anything which is carrying camera need to be tracked for autonomous navigation. Scale estimation of the translational motion is tedious in monocular visual odometry system. This paper proposes a new trajectory estimation algorithm in which Nister's 5-point algorithm is used as the baseline algorithm. We have adapted the method discussed in [1] for the scale estimation of trajectory with the additional input of camera height. Estimated scale is nothing but the ratio between the known camera height and the estimated camera height. The camera height shall be found by decomposing the Homography matrix. The proposed algorithm is tested with KITTI standard benchmark dataset [2]. Experimental results on KITTI dataset shows significant improvement in the trajectory estimation of moving object.**

***Index Terms*—Visual Odometry (VO); scale estimation; trajectory estimation; ground plane geometry.**

I. Introduction

Vision-based structure from motion (SFM) is a rapidly increasing research area which has major applications in autonomous driving, robot navigation, unmanned aerial vehicles (UAVs) systems etc. In order to get SFM we rely on two vision systems such as monocular and stereo. When we are using just one camera, it is called monocular Visual Odometry (VO) and if two or more cameras are used it is referred to as stereo VO. Odometry in computer vision refers to estimating the entire trajectory of a moving agent (robot or vehicle or human). The vector $(x_t, y_t, z_t, \alpha_t, \beta_t, \gamma_t)$ describes the complete pose (Cartesian and Euler angles) of the agent at any instant of time t. In monocular VO approach we have a camera fixed on a vehicle or a robot and the video stream coming from this camera is used for the construction of 6-degrees-of-freedom (DoF) trajectory. The translation vector coming from monocular VO is up to a scale. To get the scale information we need one more parameter. We have considered camera height from the ground plane as the extra parameter.

Motion estimation of moving object from the images captured using a camera attached with the object is not a new problem. It started since 1980 . This problem was motivated by NASA Mars exploration program when they were searching for an alternate to the 6 DoF motion estimation of the rover in the presence of uneven terrains [3].

Motion estimation using monocular VO approach shall be done in three ways. It can be based on the corner points extracted from the consecutive images or from the gray scale information or from both corner points and gray scale information. We have used Nister's 5-point algorithm as as the baseline algorithm to estimate rotation and translation matrix between consecutive images which is basically a feature point based algorithm.

Monocular systems are attractive because it is cheap and easy to implement. But they suffer from scale drift which is a serious drawback. This can be corrected by adding some extra information to the monocular VO systems. Even though stereo VO is usually much more robust than monocular VO, it become poorer than monocular VO in certain situations. Several methods are available for correcting the scale drift in monocular VO ([4], [5], [6], [7]). Prior geometric information such as height of the camera and normal direction of the ground plane are used in [8]. Some other prior information methods can be seen in ([9], [10]). The above mentioned methods do not work always because they have their own applicable scenarios.

In order to get the trajectory of the moving object from monocular VO we have to estimate the translational and rotational motion followed by scale estimation. Dingfu et al. [1] corrected and suppressed scale drift using Bundle Adjustment (BA) and they recovered the absolute scale using the prior knowledge of camera height. Usually direct decomposition of homography matrix $\mathbf{H}$ [11] is applied to estimate the camera height ([12], [13], [14]). As $\mathbf{H}$ is constructed from feature matches of noisy data, the estimated camera height obtained from it will be sensitive numerically. Song and Chandraker [15] have shown the instability of estimated scale obtained in this manner. They overcome this by applying multiple-cue fusion approach. Instead of direct decomposition of $\mathbf{H}$, Dingfu et al. proposed to decompose $\mathbf{H}$ from the structure parameters of the ground plane. Also they applied Kalman filter to get good initial values of the scale and then applied nonlinear optimization using Nelder-Mead simplex method [16] for refining camera height. But in our method we do not drop any initial scale values using filtering operation and the optimized scale value is directly applied to the trajectory estimation.

This paper is organized as follows. Motion estimation framework using Nister's 5-point algorithm is described in section II. Section III briefly describes ground plane model for scale estimation. Proposed method is explained in section IV. Experimental results and comparison of proposed method is shown in section V. Section VI shows conclusion and future works.

II. Motion estimation framework

We have a camera rigidly attached to a moving object, and our aim is to construct a 6-DoF trajectory using the video stream coming from this camera. For this construction we need the following inputs. To get the scale using ground plane geometry for trajectory estimation we have adopted the methodology described in [1] which is explained in section III.

A. Inputs for Trajectory Estimation

A camera which is attached with the moving object will be giving us the consecutive images of the environment. Let $I_t,\ I_{t+1}$ respectively represent the images, captured by the camera at time t and $t+1$. Let us assume the the camera is calibrated and we have the camera matrix with us. Let the camera matrix obtained using calibration technique be

$$\mathbf{K} = \begin{pmatrix} f/sx & \gamma & ox \\ 0 & f/sy & oy \\ 0 & 0 & 1 \end{pmatrix}$$

Here f represents the focal length of the camera, sx and sy are the pixel lengths in x and y directions respectively. (ox, oy) is the principal point of the image plane and γ is the skewness between x and y axes of the image plane. As the Nister 5-point solver [17] outputs the rotation and translation matrix up to a scale, we have to separately estimate the absolute scale for the computation of trajectory. For this purpose, additional input considered here is the camera height from the ground plane.

B. Algorithm description

Step 1 Detect feature points in image I_t using FAST algorithm [18].

Step 2 Track those features to I_{t+1} using KLT tracking algorithm ([19], [20]).

Step 3 Estimate essential matrix $\mathbf{E}$ using Nister's 5-point algorithm with RANSAC [21].

Step 4 Estimate $\mathbf{R}, \mathbf{t}$ from $\mathbf{E}$.

Step 5 Compute the relative scale between the frames using the additional information of camera height.

Step 6 Estimate the trajectory using $\mathbf{R}$, $\mathbf{t}$ and the relative scale.

C. KITTI Dataset

KITTI dataset [2] provides real-world computer vision benchmark obtained from an autonomous driving platform by driving around a mid-size city, rural areas and on highways. This benchmark dataset can be used for validating visual odometry experiments. Ground truth accuracy is guaranteed by this dataset. We have extracted the relative scale information from the ground truth trajectory supplied by the KITTI dataset ([2], [22]).

D. Algorithm flow chart for real benchmark images available in KITTI dataset

```
Step 1  Define camera matrix K, maximum number of
        frames (max_frame) and threshold for minimum
        number of tracked features (min_point).
Step 2  Detect FAST corners in the first image with
        appropriate threshold.
Step 3  Track the features in the second image using KLT
        tracker method with appropriate window size,
        pyramid level and threshold.
Step 4  Find the essential matrix E using five point
        algorithm with RANSAC.
Step 5  Find [$\mathbf{R}_{ref}, \mathbf{t}_{ref}$] from E.
        (i)     For $i = 3$: max_frame
        (ii)    Track the features in the $i^{th}$ image with
                appropriate window size, pyramid level and
                threshold.
        (iii)   Find the essential matrix E.
        (iv)    Find [$\mathbf{R}, \mathbf{t}$] from E.
        (v)     Find the scale between the frames using the
                ground plane geometry detailed in section
                III.
        (vi)    Construct trajectory of the moving object:
                $\mathbf{t}_{ref} = \mathbf{t}_{ref} + scale * \mathbf{R}_{ref} * \mathbf{t}$;
                $\mathbf{R}_{ref} = \mathbf{R} * \mathbf{R}_{ref}$.
        (vii)   If the number of features tracked in the $i^{th}$
                image $\leq$ min_point,
                detect new features in $i^{th}$image.
        (viii)  End if
Step 6  End loop
```

The algorithm assumes that all the feature points are extracted from rigid objects present in the image. But sometimes there may be a situation where the moving object/camera is stationary and other objects moving in the opposite direction. This kind of scenario may mislead the algorithm to believe that the object of interest is moved, but it is incorrect. We can avoid this kind of wrong estimations in our algorithm by imposing the following conditions:

If forward motion is dominant, then accept $[\mathbf{R}, \mathbf{t}]$ if (scale>0.1 and $z > x$, $z > y$).
If downward motion is dominant, then accept $[\mathbf{R}, \mathbf{t}]$ if (scale$>$ 0.1 and $y > x$, $y > z$).

Above conditions are valid for KITTI dataset because in KITTI dataset, X-axis represents left/right motion, Y-axis represents up/down motion, and Z-axis represents front/back (forward) motion.

III. Ground Plane Model

Let h be the height of the camera and $\mathbf{n}$ be the normal vector as shown in Fig. 1 [1]. In monocular VO we can reconstruct the camera motion up to a scale factor s only. The scale factor s is the ratio of ground true length l' and measured length l. If h' is the true camera height, the scale factor in our method is $s = h'/h$. Using this s we can recover the absolute translation $\mathbf{t}$ as $\mathbf{t}' = s\mathbf{t}$. Assume that world coordinate and camera coordinate coincide for the first frame. For any point $\mathbf{X} = (X, Y, Z)^T$ in the road plane, it will satisfy the equation:

$$\mathbf{n}^T\mathbf{X} + h = 0 \tag{1}$$

Using $[\mathbf{R}, \mathbf{t}]$ and plane geometry information $\{h, \mathbf{n}\}$ [12] $\mathbf{H}$ can be represented as:

$$\mathbf{H} = \mathbf{K}(\mathbf{R} + \mathbf{t}\mathbf{n}^T/h)\mathbf{K}^{-1} \tag{2}$$

From this representation, camera height h can be computed by decomposing $\mathbf{H}$. Optimization of three parameters related to $\mathbf{n}$ in Eq. (3) results in further scale refinement.

$$\min_{\mathbf{n}} \sum_{i=1}^{N} (X_i' - \mathbf{H}X_i) + (X_i - \mathbf{H}^{-1}X_i') \tag{3}$$

The camera height h is indirectly contained in vector $\mathbf{n}$ as $h^{-1} = ||\mathbf{n}||$.

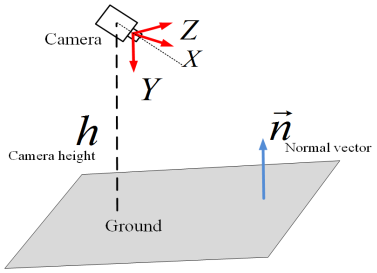
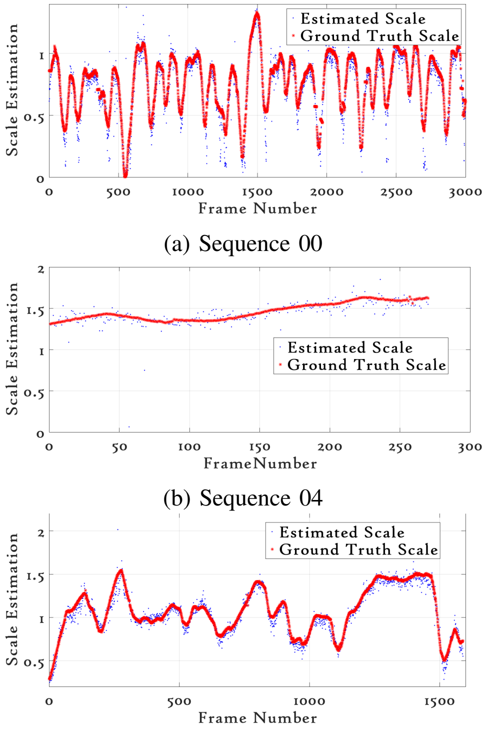


Fig. 1: Ground plane geometry



(a) Sequence 00

(b) Sequence 04

(c) sequence 09

Fig. 2: Estimated scales (blue color) and Ground truth scales (red color) for KITTI sequences 00 , 04 and 09.

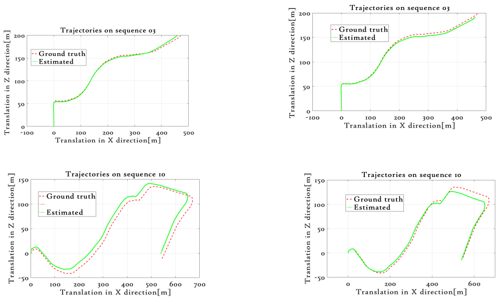
IV. Proposed method for scale estimation

In this paper we propose a feature based monocular VO algorithm with Nister's 5-point solver for trajectory estimation of a moving object. Camera height and normal direction of the ground plane are used as prior geometric information. Using the camera height and ground plane normal we can estimate scale by the decomposition of the homography matrix $\mathbf{H}$ between consecutive frames [12].

In [1], Dingfu et al. suggested to drop outliers scales after applying Kalman filter. When we follow that approach for scale estimation we ended up with severe scale drift in the trajectory estimation. So we are proposing here to consider all the initial scales for optimization and directly use this optimized value in the trajectory estimation. This may also lead to some scale drift and we have observed that it is less in the trajectory as compared to the method described in [1]. The scale estimation algorithm in short is given below:

Detect features and match them between frames for estimating the five degrees of freedom relative pose. Get the feature matches inside the region of interest (flat region in the terrain) and fit a homography matrix using them. Compute rough initial camera height using the estimated homography and relative pose. Refine the camera height by minimizing the re-projection error (Eq. (3)) of the inliers feature matches between two frames. Finally, estimate the relative scale by:

$$\text{Relative scale} = \frac{\text{camera height}}{\text{estimated camera height}}$$



(a) (b)

Fig. 3: (a) and (b) compares Visual odometry results for sequence 03 and 10 using method described in [1] and proposed method.

V. Experimental results

We have tested our algorithm with KITTI standard benchmark dataset. Results of scale estimation and ground truth for sequence $00, 04$ and 09 are given in Fig. 2. Ground truth scales are in red color and estimated ones are indicated in blue. Estimated trajectories (green color) with the proposed method is compared with ground truth trajectories (red color) for sequences 03 and 10 are given in Fig. 3. The trajectories computed using the scale estimation method described in [1] are given in Fig. 3 (a) and the results of proposed method are given in Fig. 3 (b). Comparison of the performance in trajectory estimation by the two methods shows significant improvement in the proposed method.

VI. Conclusion and future work

A new trajectory estimation algorithm for a moving object is presented. Nister's 5-point algorithm is used as the baseline algorithm with an additional input of camera height. As no scale correction method is required in our algorithm, amount of computations can be reduced. Comparison of our algorithm with the one presented in [1] shows significant improvement in the trajectory estimation. The advantage of our method is that we do not apply filtering of initial scales before optimization. In future work, we will use Continuous Wavelet Transform (CWT) technique for obtaining trajectory of the moving object. Different types of scale estimation algorithms shall be studied to get improved trajectories. Instead of using KITTI dataset for scale estimation and trajectory computation, real time data provided by the moving agent shall be used.

Acknowledgment

The authors would like to thank Deepak Negi, Head, Applied Mathematics Division, VSSC and Abhay Kumar, Group Director, AFDG, VSSC, Thiruvananthapuram for encouragement and support and also for providing computational facilities.

References

[1] D. Zhou, Y. Dai, and H. Li, "Reliable scale estimation and correction for monocular visual odometry," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 490–495.

[2] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[3] M. O. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, "Review of visual odometry: types, approaches, challenges, and applications," *SpringerPlus*, vol. 5, no. 1, p. 1897, 2016.

[4] Z. Hu and K. Uchimura, "Real-time data fusion on tracking camera pose for direct visual guidance," in *IEEE Intelligent Vehicles Symposium, 2004*. IEEE, 2004, pp. 842–847.

[5] D. P. Shepard and T. E. Humphreys, "High-precision globally-referenced position and attitude via a fusion of visual slam, carrier-phase-based gps, and inertial measurements," in *2014 IEEE/ION Position, Location and Navigation Symposium-PLANS 2014*. IEEE, 2014, pp. 1309–1328.

[6] J. Zhang, S. Singh, and G. Kantor, "Robust monocular visual odometry for a ground vehicle in undulating terrain," in *Field and Service Robotics*. Springer, 2014, pp. 311–326.

[7] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of imu and vision for absolute scale estimation in monocular slam," *Journal of intelligent & robotic systems*, vol. 61, no. 1-4, pp. 287–299, 2011.

[8] B. M. Kitt, J. Rehder, A. D. Chambers, M. Schonbein, H. Lategahn, and S. Singh, "Monocular visual odometry using a planar road model to solve scale ambiguity," 2011.

[9] T. Botterill, S. Mills, and R. Green, "Correcting scale drift by object recognition in single-camera slam," *IEEE transactions on cybernetics*, vol. 43, no. 6, pp. 1767–1780, 2013.

[10] J. Gräter, T. Schwarze, and M. Lauer, "Robust scale estimation for monocular visual odometry using structure from motion and vanishing points," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 475–480.

[11] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[12] E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," Ph.D. dissertation, INRIA, 2007.

[13] O. D. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 03, pp. 485–508, 1988.

[14] Z. Zhang and A. R. Hanson, "3d reconstruction based on homography mapping," *Proc. ARPA96*, pp. 1007–1012, 1996.

[15] S. Song and M. Chandraker, "Robust scale estimation in real-time monocular sfm for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1566–1573.

[16] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder–mead simplex method in low dimensions," *SIAM Journal on optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[17] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 0756–777, 2004.

[18] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European conference on computer vision*. Springer, 2006, pp. 430–443.

[19] J. Shi and C. Tomasi, "Good features to track," Cornell University, Tech. Rep., 1993.

[20] C. Tomasi and T. Kanade, "Shape and motion from image streams: a factorization method: full report on the orthographic case," Cornell University, Tech. Rep., 1992.

[21] K. G. Derpanis, "Overview of the ransac algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2–3, 2010.

[22] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.