Dissecting scale from pose estimation in visual odometry

Rong Yuan
rong_yuan@brown.edu
Hongyi Fan
hongyi_fan@brown.edu
Benjamin Kimia
benjamin_kimia@brown.edu

School of Engineering
Brown University
Providence, USA

Abstract

Traditional visual odometry approaches often rely on estimating the world in the form a 3D cloud of points from key frames, which are then projected onto other frames to determine their absolute poses. The resulting trajectory is obtained from the integration of these incremental estimates. In this process, both in the initial world reconstruction as well as in the subsequent PnP projection, a rotation matrix and a translation vector are the unknowns that are solved via a numerical process. We observe that the involvement of all these variables in the numerical process is unnecessary, costing both computational time and accuracy. Rather, the relative pose of pairs of frames can be independently estimated from a set of common features, up to scale, with high accuracy. This scale parameter is a free parameter for each pair of frames, whose estimation is the only obstacle in the integration of these local estimates. This paper presents an approach for relating this free parameter for each neighboring pair of frames and therefore integrating the entire estimation process, leaving only a single global scale variable. The odometry results are more accurate and the computational efficiency is significantly improved due to the analytic solution of the relative pose as well as relative scale.

1 Introduction

Odometry, the procedure for the construction of the trajectory of a moving platform from sensor has become an increasingly important problem, mainly due to an increasing range of applications, *e.g.* robotics [14], autonomous driving cars [5], drones or unmanned aerial vehicles(AUV) [7], and personal navigation. In recent years, visual odoemtry (VO), the procedure for the construction of the trajectory of a moving platform from videos captured by one or more camera mounted on moving platform has become more popular, when compared to alternatives such as radar and GPS-based odometry, mainly because it is more affordable and applies to a longer range of environments.

The key problem in visual odometry is to estimate the pose of a moving platform, namely 3 parameters for rotation and 3 parameters for translation, from correspondences between pairs in input image sequences. Specifically, after computing the correspondences, the pose parameters of each image was estimated by solving an optimization problem, associated with

an outlier rejection scheme such as RANSAC. Such methods have reached an impressive level of performance in benchmarks. However, (i) the optimization process can sometimes suffers numerical issue which prevent it from reaching the global minimum, and (ii) the cost of RANSAC, or alternative scheme, is relatively high.

This paper proposes a novel scheme when the computation of the magnitude of the translation vector T, scale λ, is separated from the computation of the rotation matrix R and the direction of the translation vector $\hat{T} = \frac{T}{|T|}$. The latter five parameters are solved analytically while the former single parameter of scale is computed using RANSAC. This procedure avoids an explicit reconstruction of 3D points thus avoiding both triangulation errors and the computation time necessary for it. The analytic computation of $(R, \hat{T})$ and the 1D RANSAC computation of λ results in a significant improvement in computation time as well as a significant improvement in robustness.

The rest of the paper organized as follows: in Section 2, related literature is presented. In Section 3, the proposed method is described. Finally, in Section 4, experiments are described and results are presented.

2 Related Works

The main problem of visual odometry is to estimate the relative pose of a moving camera from frames of a video sequence. Nister [17] use the "five-point algorithm" to solve the relative rotation and translation, up to a scale, among two consecutive frames. Note that scale remains ambiguous because scale cannot be computed from correspondences, since both the embedding spaces and the distance to camera can be scaled leading to same correspondence (metric ambiguity). Scale ambiguity has been traditionally resolved through sensor-based estimation of depth at feature points, typically either through calibrated stereo pairs [14] or RGB-D cameras [9]. Once depth is available, feature points in each image become a cloud of 3D points which can be put in correspondence with another cloud of 3D points from another image, thus estimating the scale, or the magnitude of the translation vector [5]. Alternatively, the cloud of 3D points reconstructed from one image can be projected onto a second image, and the pose is varied to minimize the reprojection error among corresponding points [3]; or with RANSAC-based outlier rejection [10].

The computation of pose from a pair of frames can be erroneous. Further more, pose estimation error from a pair of frames can propagate to subsequent frames. A set of methods aimed at using a large number of frames to robustly estimate pose, typically through bundle adjustment, which then gives global pose and global 3D reconstruction. This has been successfully employed by SLAM systems [2, 11, 15], which compute both the trajectory as well as the 3D world. An analogous system that is solely focused on the trajectory [18] combines feature tracking, pose estimation and local bundle adjustment which reached state-of-the-art performance.

Bundle adjustment is a powerful tool in preventing drift in pose estimation. its computational cost is relatively high. An alternative approach to regularizing pose estimation among frames is the use of a motion model which typically uses one parameter [8, 19, 20] or two parameters [22] to estimate the pose among consequent frames. This type of approach is appropriate for cars which satisfy this motion model.

The idea of decoupling scales was also inspired by the works in network localization area. [21] uses the composition of rigid motions and the graph cycle basis to calculate the global scale within a sensor network given the bearing vector between sensor pairs only. [1] deploy the

similar idea onto Structure from Motion problem for computing the over all scale given only the rotation matrix and the direction of translation vector among camera pairs. Our approach has two main differences comparing to these works. (i) In our application, rigid motion can only be reliably estimated locally between adjacent frames but face extreme challenges in estimating pose between more distant frames as the number of correspondence drops exponentially. Thus the composition of rigid motion is not always possible. (ii) Our scale estimation procedure links together the observation space and the parameter space, *i.e.*, unlike the bearing-only problem, our method make use of the image correspondence as clue of scale estimation.

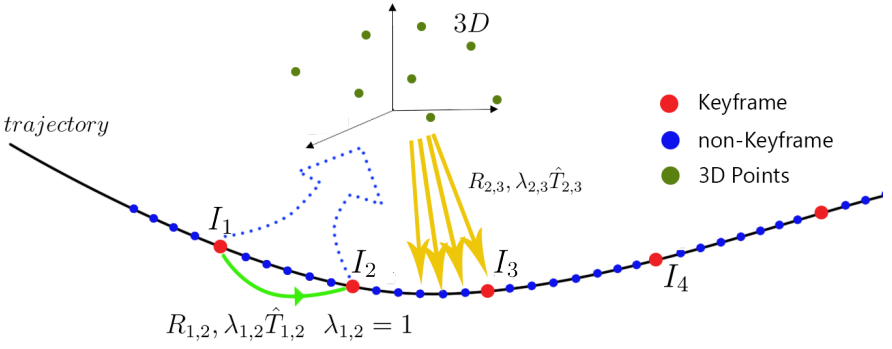

Figure 1: The traditional approach begins with a reconstruction of the 3D world in the form of an unorganized cloud of points by matching features from two initial keyframes, I_1 and I_2. The reconstructed 3D world is then used to determine poses of non-keyframes and keyframes by first tracking features and determining the pose using PnP. The reconstruction is incrementally enriched and adjusted with each new frames.

3 Our Approach: Dissecting Scale

Let the relative pose of two cameras generating two views, I_i and I_j, be denoted by the rotation matrix $R_{i,j}$ and by the translation vector $T_{i,j} = \lambda_{i,j}\hat{T}_{i,j}$, where $\lambda_{i,j}$ is the magnitude of $T_{i,j}$ and $\hat{T}_{i,j}$ is the unit vector representing the direction of $T_{i,j}$. This means that the expression of a 3D point Γ in the coordinate frame of camera i, Γ_i, is related to its expression in the coordinate frame of camera j, Γ_j, as

$$\Gamma_j = R_{i,j}\Gamma_i + \lambda_{i,j}\hat{T}_{i,j}. \tag{1}$$

It is a standard exercise in multiview reconstructions the Essential Matrix can be computed from a sufficient number of corresponding features in the two images. The Essential Matrix then gives $R_{i,j}$ and $\hat{T}_{i,j}$, leaving it as a free parameter. This implies a family of reconstructions which can scale linearly as determined by the parameter $\lambda_{i,j}$, commonly referred as the metric ambiguity in multiview reconstruction. A standard approach to visual odometry selects a number of keyframes, say every other five frames, and computes the relative pose of adjacent keyframes as described above, Figure 1. For example, for keyframes I_1 and I_2, the relative pose is computed from corresponding features, in the form of $(R_{1,2}, \hat{T}_{1,2})$, This reconstruction also gives a 3D cloud of points, which is necessary to compute the relative pose in the

intervening frames, as the yellow arrows shown in Figure 1. Specifically, once the 3D cloud of points is computed from keyframes I_1 and I_2, features are tracked in the frames between I_2 and I_3 and the correspondence between tracked features and their 3D reconstruction is used in a standard PnP pose estimation [13] to give $(R_{2,3}, \lambda_{2,3}\hat{T}_{2,3})$, with no additional scale ambiguity beyond the assumed initial scale $\lambda_{1,2}$. The pose of each new frame is determined using numerical optimization to solve for $(R_{2,3}, \lambda_{2,3}\hat{T}_{2,3})$, $(R_{3,4}, \lambda_{3,4}\hat{T}_{3,4})$, *etc*.

We observe that the optimization to solve from these six unknowns can be decomposed into estimating $(R_{n,n+1},\hat{T}_{n,n+1})$ and estimating $\lambda_{n,n+1}$. The first step is straight forward since $(R_{n,n+1},\hat{T}_{n,n+1})$ can be computed directly from "five-points algorithm" [16]. What remains is the estimation of a single scale $\lambda_{n,n+1}$. This approach has two distinct advantages. First, the estimation of $(R_{n,n+1},\hat{T}_{n,n+1})$ should improve when we estimate it from the large number of matched features between the two keyframes (A_2, A_3), typically, 2000 features, as opposed to those matched features of (A_1, A_2) which can be tracked to A_3, say 500 features. This improvement is demonstrated in Figure 3. Second, the independent estimation of $(R_{n,n+1},\hat{T}_{n,n+1})$ leaves a single variable $\lambda_{n,n+1}$ between frames A_n and A_{n+1}, *i.e.*, $\lambda_{1,2}$, $\lambda_{2,3}$, $\lambda_{3,4}$, *etc*. We now show below that the scale between each adjacent frames can be related to that of a previous adjacent pair of frames using only a single tracked features.

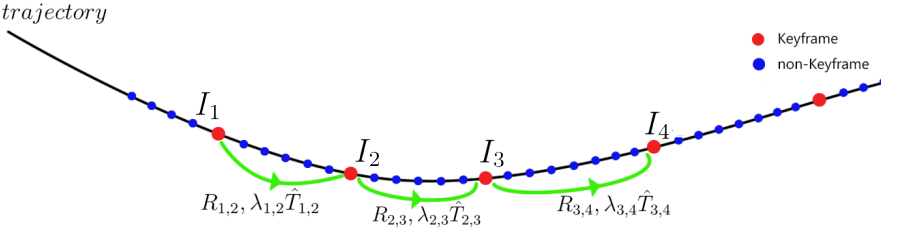

Figure 2: The proposed approach determines the relative pose of each pair of adjacent keyframes from matching features, giving $(R_{n,n+1}, \hat{T}_{n,n+1})$ but leaving a free scale variable $\lambda_{n,n+1}$ for each pair of frames. In this paper we show how to calculate scale for adjacent pairs of frames. We show that scale-independent estimation of $(R_{n,n+1}, \hat{T}_{n,n+1})$ from the full set of features improves robustness.

Proposition 1. ***(Linking metric ambiguity between adjacent pairs):*** *Consider three frames I_1, I_2 and I_3, and three corresponding image points γ_1, γ_2 and γ_3, one per frame, all arising from a single 3D point Γ. Let a pair of images I_i and I_j be related by relative pose $R_{i,j}$, $\lambda_{i,j}\hat{T}_{i,j}$ as in Equation 1, $i, j \in \{1,2,3\}$, Figure 2. When $(R_{i,j}, \hat{T}_{i,j})$ are available but the scale of translation $\lambda_{i,j}$ is unknown, the ratio of scales satisfies*

$$\frac{\lambda_{2,3}}{\lambda_{1,2}} = \frac{[(e_1^T\hat{T}_{2,1}) - (e_3^T\hat{T}_{2,1})(e_1^T\gamma_1)][(e_3^TR_{2,3}\gamma_2)(e_1^T\gamma_3) - (e_1^TR_{2,3}\gamma_2)]}{[(e_1^T\hat{T}_{2,3}) - (e_3^T\hat{T}_{2,3})(e_1^T\gamma_3)][(e_3^TR_{2,1}\gamma_2)(e_1^T\gamma_1) - (e_1^TR_{2,1}\gamma_2)]}, \tag{2}$$

where $e_1^T = [1,0,0]$, $e_2^T = [0,1,0]$ and $e_3^T = [0,0,1]$.

Proof. Let $\Gamma_1 = \rho_1\gamma_1$ and $\Gamma_2 = \rho_2\gamma_2$, where ρ_1 and ρ_2 are depths of point Γ in cameras 1 and 2, respectively. Then the inner product of coordinate vectors e_1^T, e_2^T and e_3^T with Equation 1 gives three equations for the three unknowns ρ_1, ρ_2 and $\lambda_{1,2}$:

$$\begin{cases} \rho_2(e_1^T\gamma_2) = \rho_1(e_1^TR_{1,2}\gamma_1) + \lambda_{1,2}(e_1^T\hat{T}_{1,2}) \\ \rho_2(e_2^T\gamma_2) = \rho_1(e_2^TR_{1,2}\gamma_1) + \lambda_{1,2}(e_2^T\hat{T}_{1,2}) \\ \rho_2(e_3^T\gamma_2) = \rho_1(e_3^TR_{1,2}\gamma_1) + \lambda_{1,2}(e_3^T\hat{T}_{1,2}). \end{cases} \tag{3}$$

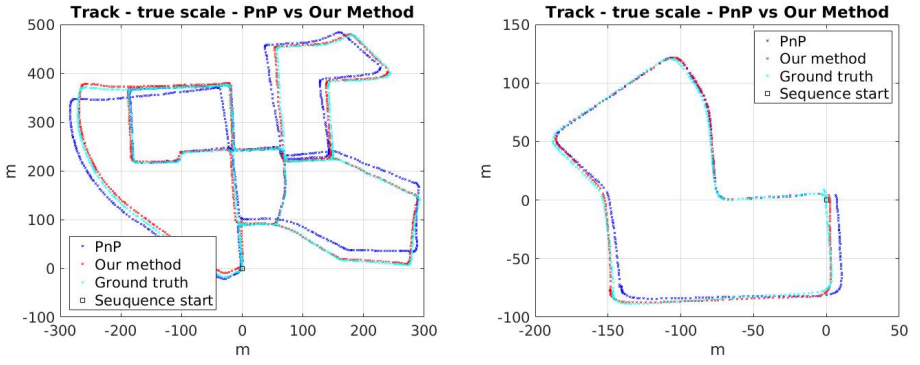

Figure 3: Direct pose estimation between two frames based on direct matching of features is more accurate than pose estimation based on tracking those features matched from a previous frame. We use the ground truth scale in comparing PnP to direct estimation to illustrate the improvement in isolation. This is shown for two distinct tracks.

Since $e_3^T \gamma_2 = 1$, the last equation relates ρ_2 to ρ_1

$$\rho_2 = (e_3^T R_{1,2} \gamma_1)\rho_1 + (e_3^T \hat{T}_{1,2})\lambda_{1,2}. \tag{4}$$

Substituting this into the first equation of Equations 3 we have:

$$\rho_1 = \left[\frac{(e_1^T \hat{T}_{1,2}) - (e_3^T \hat{T}_{1,2})(e_1^T \gamma_2)}{(e_3^T R_{1,2} \gamma_1)(e_1^T \gamma_2) - (e_1^T R_{1,2} \gamma_1)} \right] \lambda_{1,2} \tag{5}$$

$$\rho_2 = \left[\frac{(e_1^T \hat{T}_{2,1}) - (e_3^T \hat{T}_{2,1})(e_1^T \gamma_1)}{(e_3^T R_{2,1} \gamma_2)(e_1^T \gamma_1) - (e_1^T R_{2,1} \gamma_2)} \right] \lambda_{1,2}, \tag{6}$$

which gives ρ_1 and ρ_2 in terms of $\lambda_{1,2}$. Thus for two images, I_1 and I_2, depth ρ_1 and ρ_2 can be computed in terms of $\lambda_{1,2}$. Similarly, for two images I_2 and I_3, depth ρ_2 and ρ_3 can be computed in terms of $\lambda_{2,3}$. This gives two expressions for ρ_2, one in terms of $\lambda_{1,2}$ and one in terms of $\lambda_{2,3}$, *i.e.*,

$$\begin{cases} \rho_2 = \lambda_{2,1} \frac{(e_1^T \hat{T}_{2,1}) - (e_3^T \hat{T}_{2,1})(e_1^T \gamma_1)}{(e_3^T R_{2,1} \gamma_2)(e_1^T \gamma_1) - (e_1^T R_{2,1} \gamma_2)} \\ \rho_2 = \lambda_{2,3} \frac{(e_1^T \hat{T}_{2,3}) - (e_3^T \hat{T}_{2,3})(e_1^T \gamma_3)}{(e_3^T R_{2,3} \gamma_2)(e_1^T \gamma_3) - (e_1^T R_{2,3} \gamma_2)}. \end{cases} \tag{7}$$

Then the ratio between two scales is computed as

$$\frac{\lambda_{23}}{\lambda_{12}} = \frac{[(e_1^T \hat{T}_{2,1}) - (e_3^T \hat{T}_{2,1})(e_1^T \gamma_1)][(e_3^T R_{2,3} \gamma_2)(e_1^T \gamma_3) - (e_1^T R_{2,3} \gamma_2)]}{[(e_1^T \hat{T}_{2,3}) - (e_3^T \hat{T}_{2,3})(e_1^T \gamma_3)][(e_3^T R_{2,1} \gamma_2)(e_1^T \gamma_1) - (e_1^T R_{2,1} \gamma_2)]}. \tag{8}$$

□

The proposition states that for a given triplet of images I_1, I_2 and I_3, any triplet of correspondences γ_1, γ_2 and γ_3 gives $\frac{\lambda_{2,3}}{\lambda_{1,2}}$. Theoretically, any other triplet of corresponding features,

say $\bar{\gamma}_1$, $\bar{\gamma}_2$ and $\bar{\gamma}_3$ should give the same value. Practically, however, there is a distribution of estimates over this space of feature triplets, as demonstrated in Figure 4. One might be able to compute an optimal estimate among the pool of scale ratios. However, there is no foundations for minimizing error in the parameter space. Rather, the meaningful error is the observation space, *i.e.*, given a scale ratio, the extent of reprojection error when a pair of correspondence is reprojected onto a third under this scale ratio. Our goal is to find an optimal scale ratio $\lambda_{2,3}/\lambda_{1,2}$ that minimizes the trinocular reprojection error. Specially, given triplets $(\gamma_1, \gamma_2, \gamma_3)$, this scale ratio $\lambda_{2,3}/\lambda_{1,2}$ gives $\bar{\gamma}_3$ as trinocular reprojection of γ_1 and γ_2 and the error is $||\gamma_3 - \bar{\gamma}_3||$. An efficient optimization for $\lambda_{2,3}/\lambda_{1,2}$ is based on a highly efficient one-parameter RANSAC where any given triplet suggests a value for $\lambda_{2,3}/\lambda_{1,2}$ and the extent of inlier, where $||\gamma_3 - \bar{\gamma}_3|| < \varepsilon$, where ε is a distance threshold typically $\varepsilon = 1$ pixel. Figure 5 demonstrates the performance of this algorithms in determining the relative scale with respect to ground-truth.

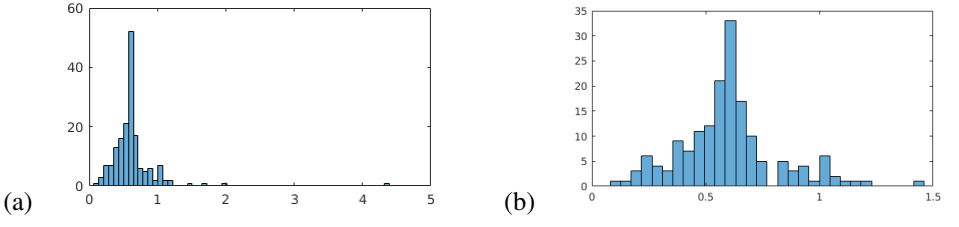

Figure 4: The distribution of $\frac{\lambda_{n,n+1}}{\lambda_{n-1,n}}$ for all features in the triple of image (I_{n-1}, I_n, I_{n+1}) shown in full in (a) and magnified in (b). Observe that while the distribution is narrow, there is a range of possible values. Also note there are some outliers which can easily be identified. Disregarding these outliers we can examine the rest of the distribution.

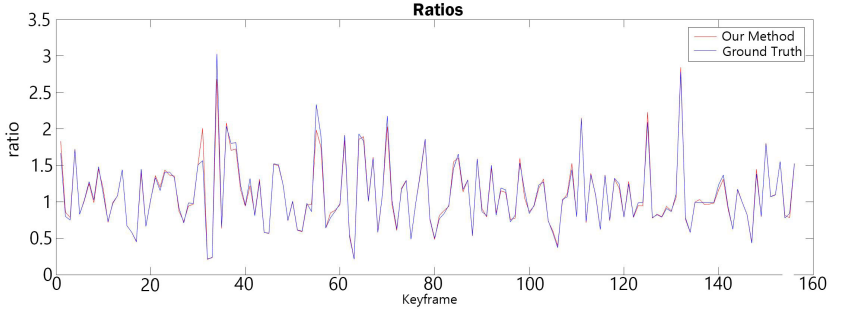

Figure 5: The comparing between ground truth relative scale $\frac{\lambda_{n,n+1}}{\lambda_{n-1,n}}$ and estimated relative scale with our method. In this figure, our method can estimate correct relative scale between keyframes. There are several wrong estimations, which will propagate.

Trinocular Reprojection Error: The traditional approach to trinocular reprojection is to first recover the 3D point by triangulation from two views and reproject onto a third view, and measure the reprojection error, *e.g.*, Figure 6, where two points γ_1 and γ_2 triangulate to give Γ and the reprojection of Γ into a third view gives $\bar{\gamma}_3$. The distance between γ_3

and $\bar{\gamma}_3$ is the trinocular reprojection error. The difficulty in this approach is that rays from two corresponding points γ_1 and γ_2 rarely meet in 3D due to calibrations and discretization errors. As such Γ is typically taken as the a point minimizing the distance from these non-intersecting rays.

Alternatively, $\bar{\gamma}_3$ can be obtained by intersecting the two epipolar lines in the third view, one arising from γ_1 and γ_2 in the third view.

$$\bar{\gamma}_3 = \begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} \frac{(a_2b_1-a_1b_2)(\bar{a}_3\bar{b}_1-\bar{a}_1\bar{b}_3)-(a_3b_1-a_1b_3)(\bar{a}_2\bar{b}_1-\bar{a}_1\bar{b}_2)}{(a_2b_3-a_3b_2)(\bar{a}_3\bar{b}_1-\bar{a}_1\bar{b}_3)-(a_3b_1-a_1b_3)((\bar{a}_2\bar{b}_3-\bar{a}_2\bar{b}_3)} \\ \frac{(\bar{a}_2\bar{b}_1-\bar{a}_1\bar{b}_2)-(\bar{a}_2\bar{b}_3-\bar{a}_3\bar{b}_2)}{(\bar{a}_3\bar{b}_1-\bar{a}_1\bar{b}_3)} \frac{(a_2b_1-a_1b_2)(\bar{a}_3\bar{b}_1-\bar{a}_1\bar{b}_3)-(a_3b_1-a_1b_3)(\bar{a}_2\bar{b}_1-\bar{a}_1\bar{b}_2)}{(a_2b_3-a_3b_2)(\bar{a}_3\bar{b}_1-\bar{a}_1\bar{b}_3)-(a_3b_1-a_1b_3)((\bar{a}_2\bar{b}_3-\bar{a}_2\bar{b}_3)} \end{pmatrix} \tag{9}$$

where $a_i = e_i^T R_{1,3}\gamma_1$, $b_i = e_i^T T_{1,3}$, $\bar{a}_i = e_i^T R_{2,3}\gamma_2$ and $\bar{b}_i = e_i^T T_{2,3}$, as derived supplementary material. This avoids an unnecessary compromise in accuracy as a result of approximation in the 3D reconstruction and the subsequent projection.

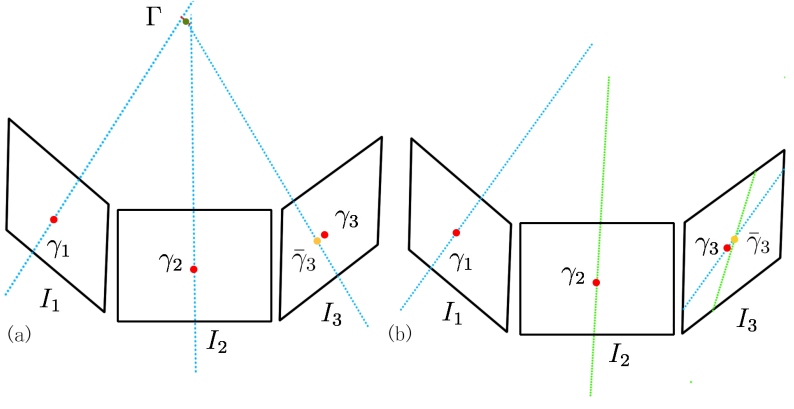

Figure 6: (a) The 3D reprojection error is typically computed by first triangulating points and then reprojecting into another view. (b) Alternatively, the point can be directly estimated without triangulation, effectively as the intersection of epipolar lines.

Monocular Visual Odometry: Our approach is that all $(R_{n,n+1}, \hat{T}_{n,n+1})$ between adjacent keyframes are computed without involving the scale $\lambda_{n,n+1}$. In a addition, the previous proposition and 1-D RANSAC approach gives the ratio $\frac{\lambda_{n,n+1}}{\lambda_{n-1,n}}$ for all adjacent keyframe pairs. What remains is to compute each scale $\lambda_{n,n+1}$. One approach is to assume a single unknown, say $\lambda = \lambda_{1,2}$, compute all $\lambda_{n,n+1}$ in terms of λ in cascade and then resort to techniques for determining this global scale λ, for example estimating the ground plane and known camera height [12]. However, observe that any error in any stage of this sequential process propagates errors, compounding the error in each subsequent stage.

Stereo Visual Odometry: Alternatively, when stereo imaging is available, instead of a single frame I_i we have a pair of frames from the left and right cameras, denoted by (I_i^-, I_i^+). Observe that the stereo camera is typically calibrated so that the relative pose is known, with a rotation matrix identity, $R = I$ and absolute translation $|T| = b$. Thus, taking a triplet of cameras, namely, (I_1^-, I_2^-, I_1^+), allows for computing the ratio of scales $\lambda_{1,2}/b$ which is in turn gives λ_2. This local computes of scale avoids a global, cascaded scale determination, thus preventing propagation and compounding of errors. Observe that this can be done in a

"forward" manner, Figure 7(b) or in a "backward" manner, Figure 7(c), allowing for a regularization of local scale based on two distinct and independent set of images. In addition, the continuity of absolute scale can be gauged by measuring relative scale in a sequential fashion, Figure 7(d). The overall method for doing stereo odometry is shown in Algorithm 1.

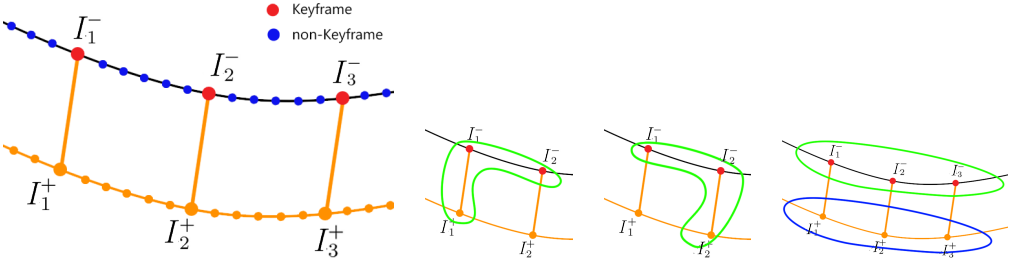

Figure 7: (a) Stereo visual odometry provides absolute scale and prevents propagation of scale errors. (b), (c) different ways of establishing absolute scale. (d) regularization of absolute scale.

Input: Correspondence between a calibrated stereo keyframe I_i^+ and I_i^- with baseline b and between frame I_1^- and subsequent frame I_2^-.
Output: Relative Pose $R_{1,2}$ $\lambda_{1,2}\hat{T}_{1,2}$ and r_{max}.
Parameters: Max number of iterations $N_{max} = 50$, $\varepsilon = 1$.
Initialization: Apply the five point algorithm to I_1^- and I_2^- to calculate $R_{1,2}$ and $\hat{T}_{1,2}$; $r_{max} = 0$, $N = 0$.
while $N \not> N_{max}$ **do**
  Randomly select a triplet of points correspondence on I_1^-, I_2^-, I_1^+, *i.e.*, $\gamma_1^- \in I_1^-$, $\gamma_2^- \in I_2^-$, and $\gamma_1^+ \in I_1^+$;
  Calculate scale $\frac{\lambda_{12}}{b}$ using Equation 2 and find $\lambda_{1,2}$;
  Calculate inlier ratio r of points with $|\gamma_2^- - \bar{\gamma}_2^-| < \varepsilon$ over all points for $R_{1,2}$ and $\lambda_{1,2}\hat{T}_{1,2}$;
  $r_{max} := \max(r_{max}, r)$;
  $N := N + 1$;
end

Algorithm 1: Stereo Odometry Method.

4 Experiments & Results

Dataset and Experiments: Evaluation was performed on the KITTI benchmark [4], where dataset are captured at 10Hz by driving around the city, organized in ten tracks covering scenes from urban areas, rural areas, and highways. Our method is evaluated on tracks 00-10 with the exception of 01, which does not contain sufficient features to track. Odometry results for track 00 and track 07 are reported here, while the odometry results for the remain tracks are reported in the supplementary materials.
Results and Discussion: KITTI benchmark contains accurate ground truth of both rotation

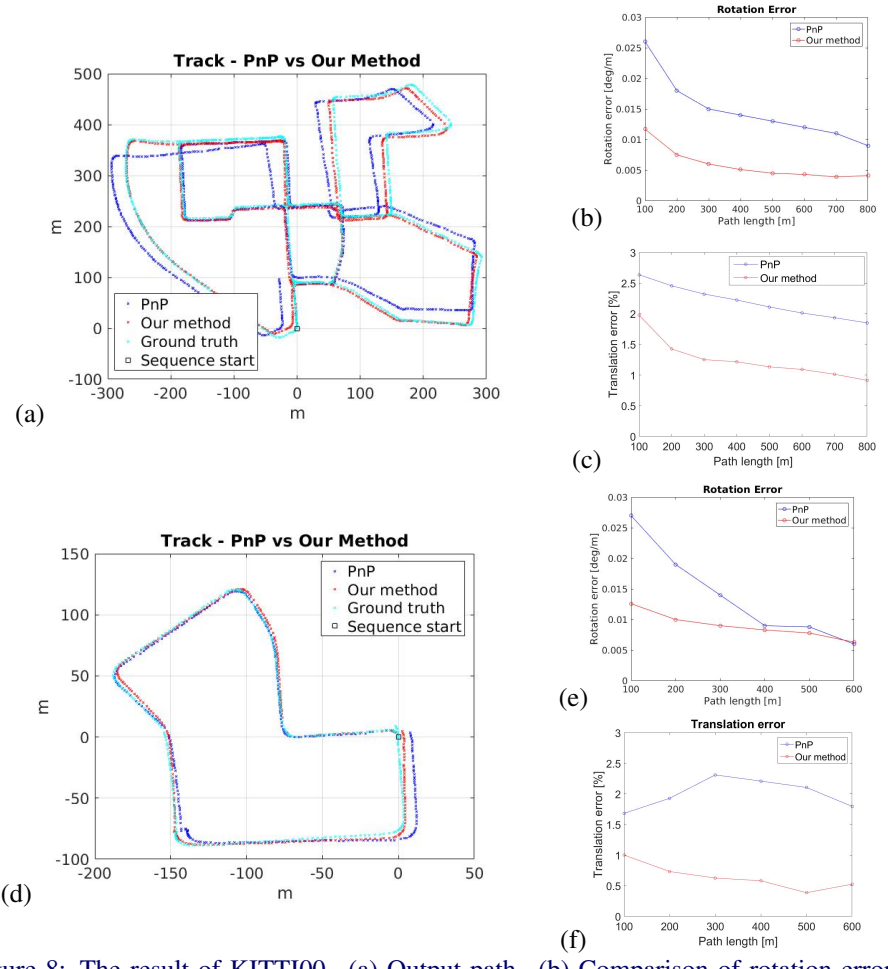

Figure 8: The result of KITTI00. (a) Output path. (b) Comparison of rotation error. (c) Comparison of translation error. The result of KITTI07. (d) Output path. (e) Comparison of rotation error. (f) Comparison of translation error.

and translation frame by frame, This allowing for a detailed comparison of results from the standard PnP approach and ours. We have avoided bundle adjustment or other refinement approach for the comparison.

The computed trajectories for track 00 and track 07 are shown for the PnP approach and our approach in comparison to ground-truth, in Figure 8(a) and (d). Observe that the trajectories computed from our approach is closer to ground truth throughout the path, especially near the end. We has also plotted rotation error and translation error for track 00 in Figure 8(b) and (c), respectively, and the same for track 07 in Figure 8(e) and (f), respectively. It is clear that there is a significant improvement in each for our method as compared to classical PnP. In addition to improving the accuracy of the trajectory computation, our method also improves in the computational requirements. Since our method and classical PnP method share the same feature correspondence method, we compare the pose estimation computation time. For the PnP method, the average running time is 0.65s per pair of keyframe, as compared to

0.38s for our method, a 41% improvement.

Estimating speed: The scale $\lambda_{n,n+1}$ is effectively the distance between two keyframes, which when normalized by the time difference between them gives the speed of the vehicle. We compare speed using the PnP method and ours against ground truth and also plot the distribution of error. These results clearly indicate that our approach estimates scale more accurately, Figure 9. Further details on other tracks can be found in supplementary materials. As an overview of the result on other tracks, Figure 10 plots the average translation, rotation, and speed error on all tracks we have experimented on.

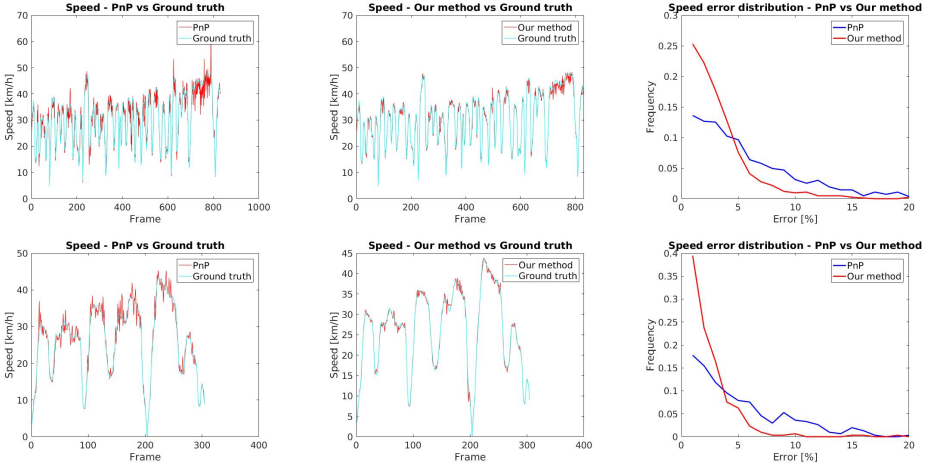

Figure 9: A comparison of scale/speed estimation of the PnP-based algorithm (left column) and ours (middle column) against the ground truth. The distribution of errors are plotted in the right column. Rows corresponds to tracks 00 and 07 of KITTI.

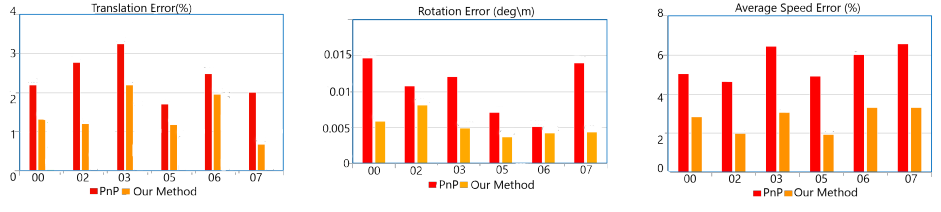

Figure 10: All tracks we experimented on. Left: Average translation Error. Middle: Average Rotation Error. Right: Average Speed Error.

5 Conclusions

In this paper, we have shown that the key problem in visual odometry is to estimate the relative scale between two frames. The rotation matrix and unit translation vector can be easily solved with existing techniques. This paper proposed a closed form solution to estimate the relative scale ratio between three views. The experimental result on KITTI benchmark shows that our method is effective and efficient.

References

[1] Federica Arrigoni, Andrea Fusiello, and Beatrice Rossi. On computing the translations norm in the epipolar graph. In *3D Vision (3DV), 2015 International Conference on*, pages 300–308. IEEE, 2015.

[2] Alessandro Chiuso, Paolo Favaro, Hailin Jin, and Stefano Soatto. 3-d motion and structure from 2-d motion causally integrated over time: Implementation. *Computer Vision, ECCV 2000*, pages 734–750, 2000.

[3] Andreas Geiger, Julius Ziegler, and Christoph Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 963–968. Ieee, 2011.

[4] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3354–3361. IEEE, 2012.

[5] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[6] Gim Hee Lee, Friedrich Faundorfer, and Marc Pollefeys. Motion estimation for self-driving cars with a generalized camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2746–2753, 2013.

[7] Albert S Huang, Abraham Bachrach, Peter Henry, Michael Krainin, Daniel Maturana, Dieter Fox, and Nicholas Roy. Visual odometry and mapping for autonomous flight using an rgb-d camera. In *Robotics Research*, pages 235–252. Springer, 2017.

[8] Yanhua Jiang, Huiyan Chen, Guangming Xiong, and Davide Scaramuzza. Icp stereo visual odometry for wheeled vehicles based on a 1dof motion prior. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 585–592. IEEE, 2014.

[9] Christian Kerl, Jürgen Sturm, and Daniel Cremers. Robust odometry estimation for rgb-d cameras. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 3748–3754. IEEE, 2013.

[10] Bernd Kitt, Andreas Geiger, and Henning Lategahn. Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. In *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 486–492. IEEE, 2010.

[11] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 225–234. IEEE, 2007.

[12] Bhoram Lee, Kostas Daniilidis, and Daniel D Lee. Online self-supervised monocular visual odometry for ground vehicles. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 5232–5238. IEEE, 2015.

[13] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009.

[14] Mark Maimone, Yang Cheng, and Larry Matthies. Two years of visual odometry on the mars exploration rovers. *Journal of Field Robotics*, 24(3):169–186, 2007.

[15] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5): 1147–1163, 2015.

[16] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence*, 26(6):756–770, 2004.

[17] David Nistér, Oleg Naroditsky, and James Bergen. Visual odometry. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–I. Ieee, 2004.

[18] Mikael Persson, Tommaso Piccini, Michael Felsberg, and Rudolf Mester. Robust stereo visual odometry from monocular techniques. In *Intelligent Vehicles Symposium (IV), 2015 IEEE*, pages 686–691. IEEE, 2015.

[19] Davide Scaramuzza, Friedrich Fraundorfer, and Roland Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4293–4299. Ieee, 2009.

[20] Davide Scaramuzza, Andrea Censi, and Kostas Daniilidis. Exploiting motion priors in visual odometry for vehicle-mounted cameras with non-holonomic constraints. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 4469–4476. IEEE, 2011.

[21] Roberto Tron, Luca Carlone, Frank Dellaert, and Kostas Daniilidis. Rigid components identification and rigidity control in bearing-only localization using the graph cycle basis. In *American Control Conference (ACC), 2015*, pages 3911–3918. IEEE, 2015.

[22] Menglong Zhu, Srikumar Ramalingam, Yuichi Taguchi, and Tyler Garaas. Monocular visual odometry and dense 3d reconstruction for on-road vehicles. In *Computer Vision–ECCV 2012. Workshops and Demonstrations*, pages 596–606. Springer, 2012.