Image Based 6-DOF Camera Pose Estimation with Weighted RANSAC 3D*

Johannes Wetzel

Fraunhofer Institute of Optronics, System Technologies and Image Exploitation
Fraunhoferstr. 1, 76131 Karlsruhe, Germany

Abstract. In this work an approach for image based 6-DOF pose estimation, with respect to a given 3D point cloud model, is presented. We use 3D annotated training views of the model from which we extract natural 2D features, which can be matched to the query image 2D features. In the next step typically the Perspective-N-Point Problem in combination with the popular RANSAC algorithm on the given 2D-3D point correspondences is used, to estimate the 6-D pose of the camera in respect to the model. We propose a novel extension of the RANSAC algorithm, named *w-RANSAC 3D*, which uses known 3D information to weight each match individually. The evaluation shows that w-RANSAC 3D leads to a more robust pose estimation while needing significantly less iterations.

Keywords: Camera Pose Estimation, Tracking, RANSAC, PnP

1 Introduction

Determining accurate six degree-of-freedom (6-DOF) pose of a camera in respect to the environment is a vital task for many computer vision applications, such as Augmented Reality (AR). We focus on image based marker-less, inside-out-tracking approaches which are based on a point cloud as world model. Gordon and Lowe [5] use structure-from-motion techniques to build a world model and propose a marker-less pose estimation based on natural image features (SIFT [11]). Arth et al.[1] focus on pose estimation of smart phones in respect to huge urban areas. At run-time level a panorama image is stitched on the smart phone. Based on natural image features (SURF [2]) the panorama is matched to the world model. One disadvantage of this approach is that the observer can not change his position during the creation of the panorama image. Our approach is inspired by the work of Irschara et al. [7], who present a method for pose estimation in respect to a sparse structure-from-motion point cloud. Each point in the world model point cloud is associated to a set of 2D SIFT features. These 2D features are extracted from original images of the environment as well as from synthetic projections out of the world model. For final pose estimation the perspective-3-point problem [6] combined with RANSAC [3] is applied. In

*Recommended for submission to YRF2013 by Prof. Dr.-Ing. Astrid Laubenheimer.

contrast to [7] we apply the perspective-n-point problem (PnP) as well as the P3P problem. There are several iterative approaches to solve the PnP like [10],[12] and problem independent meta-heuristics like the *Levenberg-Marquardt-Algorithm*. There are also closed-form solutions like [9],[14] for the PnP and [8],[4] for the P3P, which do not need an initial pose guess. For model estimation we propose an extension of the weighted RANSAC [15] method, named w-RANSAC 3D.

2 System Overview

In the following section we present our approach for the image based 6-DOF pose estimation. Initially the system gets trained with a given point cloud model $PC_M \subset \mathbb{R}^3$ of the environment. This point cloud may be obtained using sensors like the Kinect or stereo approaches. A set V_t of 3D annotated training views v_t is taken from the model by simply projecting the 3D points $\boldsymbol{P}_M \in PC_M$ to a virtual camera sensor. The virtual camera is rotated around the vertical axes with 30°, since common feature descriptors like SIFT and SURF are robust against off-image plane rotations of approximately 30° [13]. For each pixel of the 3D annotated training views $v_t \in V_t$ the mapping $f(u, v) \mapsto \boldsymbol{P}_M(X, Y, Z)$ maps a pixel to its corresponding point $\boldsymbol{P}_M$ in the world. Furthermore a set F_t of training feature descriptors is extracted from all training views V_t. This means each feature descriptor $\boldsymbol{D}_t \in F_t$ can be clearly mapped to a world point $\boldsymbol{P}_M$.

To estimate the pose of the camera by a given query image v_q a set F_q of natural 2D feature descriptors is extracted from v_q. These query features are matched against the pre-build training features F_t to get 2D-3D point correspondences, which are used to solve the PnP problem. To get the 2D-3D point correspondences between a pixel $\boldsymbol{p}_q \in v_q$ from the query image and a world point $\boldsymbol{P}_M$, a brute-force approach finds for every query descriptor $\boldsymbol{D}_q \in F_q$ the best matching training descriptor $\boldsymbol{D}_t \in F_t$. The quality of a match $\boldsymbol{D}_q \leftrightarrow \boldsymbol{D}_t$ is determined by the euclidean norm $d = \|\boldsymbol{D}_q - \boldsymbol{D}_t\|$ of the descriptor vectors. Every correspondence $\boldsymbol{D}_q \leftrightarrow \boldsymbol{D}_t$ implies a correspondence $\boldsymbol{p}_q \leftrightarrow \boldsymbol{P}_M$ between a query image pixel and its corresponding point in world coordinates. To reduce computation time we take only the n best matches into account, given by

$$M_{2D-3D} = \{(\boldsymbol{p}_q, \boldsymbol{P}_M), \dots\} \text{ with } |M_{2D-3D}| = n. \tag{1}$$

To estimate the extrinsic parameters of the camera we solve the perspective-N-point problem in a two-phase approach:

(1) Model estimation: To be robust against outliers we use the model estimator RANSAC [3] in combination with the P3P implementation from Gao et al.[4] to estimate a consensus set M'_{2D-3D} of 2D-3D point correspondences and a first approximation of the extrinsic parameters $\boldsymbol{t}', \boldsymbol{r}'$.

(2) Iterative optimization: A Levenberg-Marquard-optimization is initialized with the given extrinsic parameter guess $\boldsymbol{t}', \boldsymbol{r}'$ and the set M'_{2D-3D} of consistent point correspondences to iteratively optimize the extrinsic parameters.

In the best case, all outliers have been removed in step (1) and step (2) estimates an accurate solution based on all $n = |M'_{2D-3D}|$ robust point correspondences.

3 w-RANSAC 3D

Since the set M_{2D-3D} in general contains a lot of outliers, estimating a good first pose guess can be a challenging task. Based on the weighted RANSAC (w-RANSAC) by Zhang et al.[15], we propose a novel extension of the RANSAC algorithm, named *w-RANSAC 3D*, which uses known 3D information to weight each match individually. Zhang et al. propose that in a w-RANSAC iteration a match $m \in M$ is chosen by the probability

$$P(m) = \frac{w(m)}{\sum_{x \in M} w(x)} \tag{2}$$

while $w(m)$ is anti proportional to the distance d of two matching descriptors. The idea of our approach is to check for every match m if the pixel in the query view can be found on multiple overlapping training views, while referring to the same point in the world model. If this is true it is very likely that the match m fits well to the model. Thus this match gets a strong weight.

Based on this heuristic we propose a more advanced weight function $w(m) \mapsto \mathbb{R}^+$ (see Eq. 7) to determine how good a match fits to the world model. Therefore we first map a query descriptor not only to his best matching training descriptor, but to his k-best matching descriptors. A Match $m \in M$ is then defined as the (2k+2)-tuple

$$m = \left(\boldsymbol{D}_q, \boldsymbol{p}_q,\ \boldsymbol{D}_t^0, \ldots, \boldsymbol{D}_t^{k-1},\ \boldsymbol{P}_M^0, \ldots, \boldsymbol{P}_M^{k-1}\right) \tag{3}$$

while $\boldsymbol{D}_q$ and $\boldsymbol{p}_q$ are the descriptor and the corresponding pixel in the query image. Furthermore $\boldsymbol{D}_t^0, \ldots, \boldsymbol{D}_t^{k-1} \in F_t$ are the training feature descriptors of the best k matches $\boldsymbol{D}_q \leftrightarrow \boldsymbol{D}_t^j$, while $\boldsymbol{D}_t^j$ refers to the j best matching descriptor. Respectively $\boldsymbol{P}_M^j \in \{\boldsymbol{P}_M^0, \ldots, \boldsymbol{P}_M^{k-1}\}$ refers to corresponding world coordinates of the j-best matching feature descriptor. The mapping $v : F_t \mapsto V$ maps a training descriptor $\boldsymbol{D}_t \in F_t$ to the corresponding view $v_t \in V$, from which the descriptor is extracted. The so called *Lowe's Ratio* [11] is calculated for every j-best matching descriptor as $d_r^j = \frac{d^0}{d^j}$ where $d^j = \|\boldsymbol{D}_q - \boldsymbol{D}_t^j\|$ is the euclidean norm between a query descriptor and its j-best matching training descriptor. To get an absolute proportion of the quality of the 2D-2D matching we normalize the distance d_r^j to $d_N^j \in [0..1]$. To determine if two world points are close to each other in 3D space the euclidean norm $d_{3D}^j = \|\boldsymbol{P}_M^0 - \boldsymbol{P}_M^j\|$ is used.

Depending from which training view the training descriptors $\boldsymbol{D}_t^j$ are extracted, the weight function is divided into different parts which are represented by the characteristic variable $c_j \in \{0, 1\}$. Let $j \in [0, \ldots, k-1]$ then for the special case $j = 0$:

$$c_0 = \begin{cases} 1, & v(\boldsymbol{D}_t^0) = v(\boldsymbol{D}_t^1), \\ 0, & \text{else.} \end{cases} \tag{4}$$

and for $j \in [1, \dots, k-1]$:

$$V_j = \left\{ \; v(\boldsymbol{D}_t^0), \dots, v(\boldsymbol{D}_t^j) \; \right\} \tag{5}$$

$$c_j = \begin{cases} 1, & |V_j| = j+1 \\ 0, & \text{else.} \end{cases} \tag{6}$$

Hence c_0 is true if the best and the second-best descriptor are extracted from the same view. The cases c_j for $j \in [1, \dots, k-1]$ are true if the $j+1$ best matching descriptors are all extracted from disjoint views. According to these cases, the weight of a match m is defined as

$$w(m) = \left[c_0 \cdot \alpha \cdot g_3(d_r^1) + \prod_{j=1}^{k-1} \left(1 + c_j \cdot \beta_j \cdot g_1(d_{3D}^j) \cdot g_2(d_r^j) \right) \right] \cdot g_4(d_N^0) \tag{7}$$

To explain the weight function we distinguish between two exclusive cases depending on c_j:

Case 1 ($c_0 = 1$) Here the weight is calculated based on the *Lowes' Ratio* d_r^1 of the best and second-best feature descriptors extracted from the same training view. The Gaussian function g_3 weights a match stronger if d_r^1 gets smaller.

Case 2 ($c_j = 1$ **,** $j \in [1, \dots, k-1]$) The idea is to weight a match stronger if query pixel $\boldsymbol{p}_q$ matches good to similar world points $\boldsymbol{P}_M^0, \boldsymbol{P}_M^j$, each extracted from different training views. Thus the Gaussian functions g_1, g_2 increase the weight of a match if d_r^j approximates to 1 and d_{3D}^j approximates to 0.

Independent of c_j the weight of a match is influenced by the normalized ratio d_N^0 which represents the quality of the 2D-2D matching. If the descriptors match good $g_4(d_N^0)$ approximates to 1, for bad matches respectively to 0. The function g_4 is similar to the weight function used in w-RANSAC. The factors α, β_1, β_2 are parameters to control the influence on the summed weight.

4 Experiments

In the following, we present evaluation results illustrating the performance of our approach in comparison to RANSAC and w-RANSAC. We did experiments based on a set of 12 synthetic query images, reprojected from different poses around an indoor office scene with ground truth. To measure the quality of a pose estimate we apply the ground truth extrinsic parameters $\boldsymbol{t}, \boldsymbol{r}$ from the query image on every world point P_M which is visible on the current query view. As a result we get the point cloud P_g. Accordingly we apply the estimated extrinsic parameters to the same world points and get the corresponding set P_e. Hence we can define the point error as the mean euclidean norm between the corresponding points $\boldsymbol{p}_{gi} \in P_g, \boldsymbol{p}_{ei} \in P_e$. The point error is defined as $error = \frac{1}{n} \sum_{i=1}^{n} |\boldsymbol{p}_{gi} - \boldsymbol{p}_{ei}|$ with $n = |P_g| = |P_e|$. We present the mean point error of 100 iterations for every query view in meters.

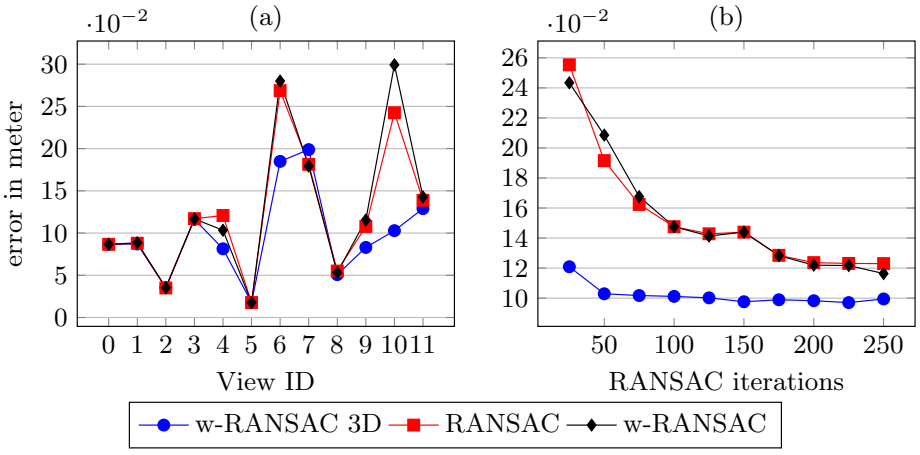

Fig. 1. Comparison of w-RANSAC, RANSAC and w-RANSAC 3D

In Fig. 1(a) the mean point error for every view of the test set is shown. It demonstrates that for the views 4,6,9,10 our approach leads to a more accurate pose estimation while for the other views the error is about the same for all three approaches. For view 7, w-RANSAC 3D performed slightly worse than RANSAC and w-RANSAC. This is due to misalignment of the view, meaning $\boldsymbol{r}, \boldsymbol{t}$ are estimated more accurate but in this special case this leads to a higher point error. Fig. 1(b) shows the performance of the three approaches relative to the number of RANSAC iterations. Here the error is the mean over all test views. The figure illustrates that our approach approximates to its optimum in less than 50 iterations while RANSAC and w-RANSAC both need more than 200 iterations to reach their optimum.

Fig. 2 shows two example query images taken from a standard consumer camera and the reprojection of the estimated extrinsic parameters using w-RANSAC 3D.

(a) (b)

Fig. 2. Real query images with corresponding reprojection from estimated extrinsic parameters. (a) is an example for an inaccurate pose estimation, (b) shows a quite accurate pose estimation.

5 Conclusion

In this work we presented w-RANSAC 3D, a novel extension of the RANSAC algorithm for image based 6-DOF pose estimation based on a reconstructed point cloud. In the evaluation we have shown that w-RANSAC 3D outperforms w-RANSAC and RANSAC in terms of robustness while needing significantly less iterations. For future work, we consider using the weights not only in the RANSAC algorithm, but also in the iterative PnP algorithm for a more robust and accurate solution. Furthermore there is the idea to improve the current weight function with additional heuristics like verifying if the geometric neighborhood suites to a match.

References

1. Arth, C., Klopschitz, M., Reitmayr, G., Schmalstieg, D.: Real-time self-localization from panoramic images on mobile devices. ISMAR pp. 37–46 (2011)
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Comput. Vis. Image Underst. 110(3), 346–359 (Jun 2008)
3. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24(6), 381–395 (Jun 1981)
4. Gao, X.S., Hou, X.R., Tang, J., Cheng, H.F.: Complete solution classification for the perspective-three-point problem. TPAMI pp. 930 – 943 (2003)
5. Gordon, I., Lowe, D.G.: What and where: 3d object recognition with accurate pose. Toward Category-Level Object Recognition pp. 67–82 (2006)
6. Haralick, R., Lee, D., Ottenburg, K., Nolle, M.: Analysis and solutions of the three point perspective pose estimation problem. CVPR pp. 592 –598 (1991)
7. Irschara, A., Zach, C., Frahm, J.M., Bischof, H.: From structure-from-motion point clouds to fast location recognition. CVPR pp. 2599–2606 (2009)
8. Kneip, L., Scaramuzza, D., Siegwart, R.: A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. CVPR pp. 2969–2976 (2011)
9. Lepetit, V., Moreno-Noguer, F., Fua, P.: Epnp: An accurate o(n) solution to the pnp problem. IJCV 81(2), 155–166 (Feb 2009)
10. Lowe, D.G.: Fitting parameterized three-dimensional models to images. TPAMI 13, 441–450 (1991)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60, 91–110 (2004)
12. Lu, C.P., Hager, G., Mjolsness, E.: Fast and globally convergent pose estimation from video images. TPAMI 22(6), 610 –622 (jun 2000)
13. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.: A comparison of affine region detectors. IJCV 65(1), 43–72 (November 2005)
14. Quan, L., Lan, Z.: Linear n-point camera pose determination. TPAMI pp. 774 –780 (1999)
15. Zhang, D., Wang, W., Huang, Q., Jiang, S., Gao, W.: Matching images more efficiently with local descriptors. ICPR pp. 1 –4 (2008)