Ren B, Wu JC, Lv YL *et al.* Geometry-aware ICP for scene reconstruction from RGB-D camera. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 34(3): 1–13 May 2019. DOI 10.1007/s11390-015-0000-0

Geometry-Aware ICP for Scene Reconstruction from RGB-D Camera

Bo Ren, *Member*, *CCF*, *ACM*, Jia-Cheng Wu, Ya-Lei Lv, Ming-Ming Cheng, *Member*, *CCF*, *ACM* and Shao-Ping Lu, *Member*, *CCF*, *ACM*

College of Computer Science, Nankai University, Tianjin 300350, China

E-mail: rb@nankai.edu.cn; wjcskqygj@gmail.com; lyuyalei@126.com; cmm.thu@gmail.com; slu@nankai.edu.cn

Received December 29, 2018; revised March 15, 2019.

Abstract The Iterative Closest Point (ICP) scheme has been widely used for the registration of surfaces and point clouds. However, when working on depth image sequences where there are large geometric planes with small (or even without) details, existing ICP algorithms are prone to tangential drifting and erroneous rotational estimations due to input device errors. In this paper, we propose a novel ICP algorithm that aims to overcome such drawbacks, and provides significantly stabler registration estimation for simultaneous localization and mapping (SLAM) tasks on RGB-D camera inputs. In our approach, the tangential drifting and the rotational estimation error are reduced by: 1) updating the conventional Euclidean distance term with the local geometry information, and 2) introducing a new camera stabilization term that prevents improper camera movement in the calculation. Our approach is simple, fast, effective, and is readily integratable with previous ICP algorithms. We test our new method with the TUM RGB-D SLAM dataset on state-of-the-art real-time 3D dense reconstruction platforms, i.e., ElasticFusion and Kintinuous. Experiments show that our new strategy outperforms all previous ones on various RGB-D data sequences under different combinations of registration systems and solutions.

Keywords ICP (iterative closest point), RGB-D, tangential drifting, rotational estimation, covariance matrix

1 Introduction

The Iterative Closest Point (ICP) algorithm[1] has been widely used for the registration of surfaces and point clouds, and it plays a central role in various computer vision and robotics applications, ranging from simultaneous localization and mapping (SLAM) to object recognition and detection and to augmented and virtual reality.

At its core, ICP aims to recover the transformation between two point clouds by alternating a point matching phase with a minimization of the total squared error between matches. This simple idea has seen many variations introduced over the years, e.g., by [2-7]. For example, the point cloud data can be cleaned up with filtering[8−11]. Different types of energy functions, capturing different strategies of generating registration candidates can be employed, such as frame-to-frame[12] or frame-to-model[8]. Specific data structures such as the truncated signed distance function (TSDF)[13] and kd-trees[14] can help and hasten the discovery of point correspondences. Additional information collected from the input data, such as color[15,16] or salient key-points[10], can also be used to increase robustness.

When the input data has large planes with few (or even without) details, existing ICP algorithms are prone to tangential drifting and erroneous rotational estimations. In such case the energy function does not successfully penalize erroneous camera movements. In this paper, we propose a novel energy function formulation for scene reconstruction from RGB-D camera inputs, as an alternative to the standard point-to-point energy of [1], its improved variant of point-to-plane[17], and the generalized probabilistic distribution based model of [2, 18, 19]. In our approach, the rotational estimation error and the tangential drifting are reduced by: 1) updating the conventional Euclidean

Regular Paper

Special Section of CVM 2019

This work was supported by Tianjin Natural Science Foundation of China under Grant Nos. 18JCYBJC41300 and 18ZXZNGX00110, and the National Natural Science Foundation of China under Grant No. 61620106008.

©2019 Springer Science + Business Media, LLC & Science Press, China

distance term to take local geometry information into account, and 2) introducing a new camera stabilization term that prevents improper camera movement in the calculation.

Our modification is simple, fast yet effective. Experiments on the TUM RGB-D SLAM dataset show that our strategy outperforms all the matching energy function variations stated above. Furthermore, our solution is readily integratable with previous ICP algorithms, through a simple substitution of the corresponding terms in the energy function; thus all variations (e.g., using color or filtering) still apply.

Using our novel ICP energy function requires little extra computational overhead compared with the standard ICP pipeline. This enables us to generate better and more robust registration results without harming the usability of the ICP algorithm in real-time scenarios. We showcase this by integrating and benchmarking our novel ICP formulation in two state-of-the-art real-time 3D dense reconstruction platforms, ElasticFusion[20] and Kintinuous[21].

Our main contributions are as follows.

1) We provide insight into the fundamental drawbacks of the conventional ICP algorithm that lead to erroneous rotational estimations and tangential drifting results.

2) Based on our analysis, instead of only using the conventional Euclidean distance, we introduce a geometry-aware energy function that leads to stabler registration estimation, especially in datasets containing less-detailed large planes.

3) Our algorithm is readily integratable with state-of-the-art SLAM systems and also achieves higher performance when combined with orthogonal strategies such as using color information or loop closure.

The remainder of this paper is structured as follows. Section 2 describes related work. In Section 3, based on a novel analysis that directly reveals the reasons causing failures of original ICP algorithm on real-world datasets, we develop our approach that targets at overcoming the drawbacks. Experimental results are shown in Section 4 and finally we conclude and discuss future work in Section 5.

2 Related Work

The ICP algorithm is one of the most popular methods used in the local refinement stage of the extensively studied geometric registration problem[5,6,22,23]. The problem in this stage of registration is mainly to obtain a tight registration between surfaces[24,25], where an initial estimate of the rigid motion is computed from a former global alignment stage. Recently, the ICP algorithm and its variances have found more popularity in most real-time SLAM applications due to their simplicity and efficiency[26−30].

However, the ICP algorithm is not perfect by itself, with many unresolved difficulties affecting its performance in practice, which leads to hundreds of research attempts to improve its versatility. The ICP algorithm has to match a new pose to a previous pose, which is sensitive to how precise and noise-free the previous and current poses are. There are filtering methods for the input point cloud data proposed to reduce noise from different practical sensors[8−11]. In contrast to the original frame-to-frame strategy[1,12], frame-to-model strategies[8,31] are proposed to further average out the erroneous noise. Meanwhile, other visual features are often used to assist matching in pose estimation, such as considering matching of the color from RGB-D sensor inputs[3,24], adding various keypoint descriptors[11,32], which stabilize estimated camera motions and produce more robust registration compared with vanilla ICP. The cumulative errors from each estimation between frames can result in the failure of loop closure, and many researches tackle this problem by introducing global[33] or local[34,35] optimizations. A recent work uses re-localizing algorithm to enhance performance when data contains fast camera motion[36]. Other studies explored the utility of relaxed assignments[37−39], distance field representations[40], mixture models[41,42], and local reference frame[43] for increasing the robustness of local registration, or using different data structures such as kd-tree[14,15] and TSDF[13] or being assisted with a spatial hashing scheme[44] to enhance computational performance of the ICP algorithm. Based on variations of ICP, integrated systems such as Kinectfusion[8], Kintinuous[21], InfiniTAM[45], and ElasticFusion[20] are widely used in computer vision. The evaluations of the related methods are usually performed on benchmarks[46,47].

The distance metric of ICP also evolves over time. Originally the ICP algorithm is based on a point-to-point calculation of distance metric[1], while later on the point-to-plane method is proved to be more efficient with better results[17]. Mainly adopted in robotics for laser-beam scanned data inputs, the normal-distributions transform (NDT) algorithm and its variants (such as Generalized-ICP (GICP)[2], 3D-NDT[48] and Color-NDT[49]) further use a probabilistic distribution based model for distance metric. How-

ever, the point-to-point and point-to-plane methods do not consider local anisotropic geometry, and favor eliminating normal-aligned displacement over tangential displacements. On the other hand, the NDT-based algorithms assume normal-aligned errors are much smaller, and their weight should be much higher than tangential weights in matching optimization, which is true for laser-beam scanned input in robotics but not true for general RGB-D cameras widely used in computer vision. Contrary to the previous methods, we consider local geometry and input error from RGB-D cameras, enhancing camera pose registration by reducing rotational estimation errors and tangential drifting.

During the pose estimation between input frames, outlier handling is another problem that affects total performance of the ICP algorithm. These outlier points are usually recognized according to its mismatching distance[7], rank in all pairs based on some metric[50], or consistency with neighboring pairs[51]. Many studies consider it suffice to discard the outliers[8,24], and a few researches such as GICP[2] indirectly include them in the computation with its distribution calculation. Instead of discarding the outliers, in our approach we utilize them in the pose estimation to help stabilize camera motion.

3 Approach

In this section, we first recap the conventional ICP algorithm and analyze the reason of its registration error. Then we introduce our novel ICP algorithm that is able to overcome the drawbacks in previous strategies.

3.1 Point-to-Plane ICP

As a widely adopted ICP strategy, the point-to-plane[17] ICP is generally considered to be an upgradation of the point-to-point[1] ICP, making it more suitable in various SLAM applications. Therefore, we settle to briefly recap the point-to-plane ICP in this subsection.

The point-to-plane ICP registers two point clouds from sequential inputs of two frames (or one frame and the global model in the frame-to-model case), and gives the estimation of camera pose change. An initial guess, usually using identity (or the camera pose of the last frame in the frame-to-model case), is first provided to generate an initialized set of matching-point pairs. Assuming that the point $\boldsymbol{v}_t^k$ from the current frame t matches with the point $\boldsymbol{v}^k$ from the previous frame, and the normal vector of $\boldsymbol{v}^k$ is denoted as $\boldsymbol{n}^k$, the camera pose estimation problem is equivalent to minimizing the following energy function[20]:

$$E = \sum_k \|(\boldsymbol{v}^k - \exp(\hat{\boldsymbol{\xi}})\boldsymbol{T}\boldsymbol{v}_t^k) \cdot \boldsymbol{n}^k\|^2, \tag{1}$$

where $\boldsymbol{T}$ is the transformation matrix, and $\hat{\boldsymbol{\xi}}$ corresponds to a small change of the camera pose. The summation consists of all Euclidean distances from each transformed $\boldsymbol{v}_t^k$ position to the corresponding mapping on the tangential plane of $\boldsymbol{v}^k$. Given $\hat{\boldsymbol{\xi}}$ is small, such an energy function can be further linearized with respect to ξ using the Rodriguez equation, i.e., $\boldsymbol{J}_{icp}\boldsymbol{\xi} + \boldsymbol{r}_{icp}$, where $\boldsymbol{J}_{icp}$ is a combined measurement Jacobian form and $\boldsymbol{r}_{icp}$ is a residual[20]. Following this idea, an optimal solution of $\boldsymbol{\xi}$ can be obtained by the typical least square method.

3.2 Failure Case Analysis on Point-to-Plane ICP

The point-to-plane ICP is widely used in computer vision applications whose inputs are captured with various depth cameras. Working on depth image sequences, however, this ICP algorithm is easy to generate erroneous camera pose estimations. This is much worse when the captured view has only a few large planes with few or even without details. A key observation is that such failure cases consist of two kinds of wrong camera pose estimation: 1) erroneous rotational estimation of the camera movement, and 2) tangential drifting of camera along a certain plane.

Generation of the rotational error can be analyzed with Fig.1. In this simple scene, suppose the ideal 3D model consists of a large plane and a small planar square perpendicular to it. Due to the precision of the input device (e.g., an RGB-D camera), the "current" and "previous" frames used in the calculation do not perfectly match with the "real object" and have small errors. Consider the case that their errors are both of a checker-board-like distribution but have reversed signs on the big plane as shown in Fig.1(a). The point-to-plane method evaluates the mismatch energy of different matching results and finds an optimized result with the minimum mismatch energy. In Fig.1 we select two possible matches that can be evaluated by point-to-plane ICP: one ideal match corresponding to the correct pose estimation (upper-right) and one 90° rotated match which is wrong (lower-right). We will show in the following that as the point-to-plane ICP

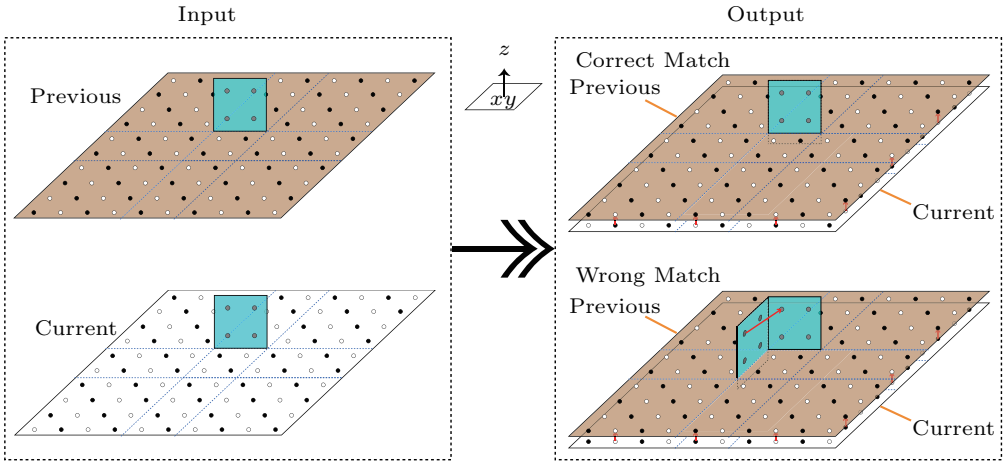


Fig.1. Erroneous rotational estimation caused by point-to-plane ICP. The scene consists of a large plane and a small square perpendicular to it. Small circles indicate sample points with input errors from device. White color means an error towards positive-z direction and black towards negative-z direction. Arrows indicate point match in the registration. When the plane is large enough compared with the square, point-to-plane ICP will lead to erroneous 90° rotated match that eliminates the mismatch error on the large plane. Gaps between two large planes in this figure are drawn purely for visual clarity and do not exist in the registration.

minimizes the perpendicular mismatching distance between two frames, when the plane is sufficiently larger (as explained below) than the small square, the optimized result will be a wrong match that rotates the current frame by 90°s. We need only simple calculation to see the reason. Suppose we have one sample point in each 0.01 m^2 area, the small square has the side length of 0.2 m (4 samples) and the large-enough plane has the side length of 2 m (400 samples), the input error from device is 0.01 m. Using the energy function of point-to-plane ICP, in the ideal match, each point pair on the large plane has a mismatch distance of 0.02 m and each point pair on the small square has 0 m mismatch distance, thereby the total mismatch energy is $(0.02\ \text{m})^2 \times 400 = 0.16\ \text{m}^2$, where 400 is the number of sample points. In the 90° rotated wrong match, each point pair on the large plane has a mismatch distance of 0 m. Two of the four sample points on the small square have mismatch distance of 0.2 m, and the other two have that of 0.1 m. Therefore the total mismatch energy is $2 \times (0.2\ \text{m})^2 + 2 \times (0.1\ \text{m})^2 = 0.1\ \text{m}^2$. The point-to-plane ICP will thus be in favor of the wrong match compared with the ideal match, resulting in erroneous rotation. Note that in the latter wrong match, the vector mismatch-distance of the square has a tangential opponent measured at its estimated position. If we take this into consideration and put a larger weight on such a tangential mismatching distance based on local geometry, the energy of the wrong match may be able to scale up and surpass the energy of the ideal match, leading to correct estimation toward the ideal match. We will discuss this in more detail in Subsection 3.3, and a real example on real-world dataset is provided there.

On the other hand, it is not hard to see that the tangential drifting along a large plane in the scene is a direct result of the point-to-plane ICP algorithm, since the dot product with $\boldsymbol{n}^k$ in (1) totally omits tangential mismatches in the calculation. However, such tangential drifting will create points (outliers) that are unable to find a good-enough match in the calculation. In state-of-the-art SLAM systems such as in ElasticFusion, such points will be treated as outliers, which means a large tangential drifting will produce many outlier points as shown in Fig.2. This inspires us to penalize outlier points to prevent tangential drifting, more details will be described in Subsection 3.3.

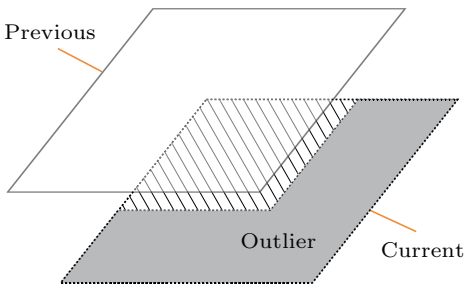


Fig.2. Tangential drifting along a finite plane creates a large number of outliers (gray region).

Plane size has influence in the above two aspects. Generally, in data sequences the scene contains many

planes of different sizes and alignments. While using (1), the optimization loop will be in favor of giving an estimation that strictly fits objects in their surface normal direction. In other words, if the noise or error from the input device generates competing mismatch between surfaces aligned in various directions, (1) will prefer to match larger planes in their normal directions first, over their tangential drift and the resulting mismatch of other smaller objects.

3.3 Geometry-Aware ICP

Based on the above observation, we propose a novel geometry-aware ICP solution to resolve the specific problem of erroneous rotational estimation and tangential drifting.

3.3.1 Geometry-Aware Energy Function

For improving rotational estimation, the Euclidean energy term is substituted by a new term as follows:

$$E_1 = \sum_k \|\boldsymbol{D}^{\mathrm{T}}\tilde{\boldsymbol{G}}\boldsymbol{D}\|, \tag{2}$$

where $\boldsymbol{D} = ||(\boldsymbol{v}^k - \exp(\hat{\boldsymbol{\xi}})\boldsymbol{T}\boldsymbol{v}_t^k)\cdot\boldsymbol{n}^k||\boldsymbol{n}^k$ is the mismatching distance vector with symbol meanings as described in Subsection 3.1.

Intuitively, we no longer use the Euclidean distance between two points, but modify it with a matrix $\tilde{\boldsymbol{G}}$. To explain $\tilde{\boldsymbol{G}}$, firstly a matrix $\boldsymbol{G}$ can be defined from the covariance matrix computed within a point's neighborhood as:

$$\boldsymbol{G} = \frac{\sum_{\boldsymbol{x}\in N}(\boldsymbol{x}-\bar{\boldsymbol{x}})(\boldsymbol{x}-\bar{\boldsymbol{x}})^{\mathrm{T}}}{|N|},$$

where N denotes the set of neighboring points, and $\bar{\boldsymbol{x}}$ is the average position vector.

The effect of the above $\boldsymbol{G}$ matrix is serving as an ellipsoidal kernel to the distance measurement, with its shortest axis along the geometric local normal direction, and longer axis along tangential directions. Due to the property of the covariance matrix, if $\boldsymbol{X}$ is a mismatching distance vector, $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{G}\boldsymbol{X}$ effectively scales distance in tangential directions proportional to the corresponding eigen values. Therefore, such energy function is geometry-aware in comparison with the Euclidean distance based formulation.

$\boldsymbol{G}$ can be easily computed from a 5×5 pixel window on the input frame centered at each point. We also normalize $\boldsymbol{G}$ utilizing $(\sum_{\boldsymbol{x}\in N}\|\boldsymbol{x}-\boldsymbol{x}_0\|)/|N|$, i.e., the average of distances from the current point to its neighboring points, in order to count for scale change over depth. The final computation of kernel matrix $\boldsymbol{G}$ is:

$$\boldsymbol{G} = \begin{cases} \frac{|N|^\gamma}{(\sum_{\boldsymbol{x}\in N}\|\boldsymbol{x}-\boldsymbol{x}_0\|)^\gamma}\left(\frac{\sum_{\boldsymbol{x}\in N}(\boldsymbol{x}-\bar{\boldsymbol{x}})(\boldsymbol{x}-\bar{\boldsymbol{x}})^{\mathrm{T}}}{|N|}\right), \text{ if } |N| > k_r, \\ k_n\boldsymbol{I}, \text{if } |N| \leqslant k_r, \end{cases} \tag{3}$$

where $\boldsymbol{x}_0$ is the position of current point. For the few stand-alone points that have too few neighbors (less than k_r), to avoid unreliable covariance matrix calculation, we set $\boldsymbol{G} = k_n\boldsymbol{I}$ with k_n as a constant value. Since far-away points have larger errors, in order to reduce far-away-point influence, γ is set to 2 in frame-to-model systems where a depth cut-off method is usually integrated, and is set to 4 in frame-to-frame systems where no such cut-off is applied.

Then $\tilde{\boldsymbol{G}}$ should be $\tilde{\boldsymbol{G}} = \boldsymbol{R}\boldsymbol{G}\boldsymbol{R}^{\mathrm{T}}$, where $\boldsymbol{R}$ is the rotation matrix between the current frame and the estimated pose, i.e., $\boldsymbol{x}_{\mathrm{e}} = \boldsymbol{R}\boldsymbol{x}_{\mathrm{c}}$ with $\boldsymbol{x}_{\mathrm{e}}, \boldsymbol{x}_{\mathrm{c}}$ which refer to a direction vector in estimated and current coordinate respectively. This ensures distance scaling is correctly calculated in the estimated-pose local coordinate but not in camera screen coordinate. Note this also makes the optimization problem no longer quadratic. A solution is, in each ICP iteration, treating $\tilde{\boldsymbol{G}}$ as a known value calculated from the estimated rotation from the last iteration. We use this strategy in our approach, and in our experiments it always converges under real-time performance.

Note if $\boldsymbol{G} = \boldsymbol{I}$, (2) is exactly the distance metric in previous point-to-plane ICP. Then, in (3) when k_n is used, we effectively calculate the standalone points using the point-to-plane ICP distance metric with a weight of k_n. This weight should be smaller than the smallest eigen value of $\boldsymbol{G}$, i.e., the eigen value along normal direction, computed from input data (approximately no larger than 0.015 for datasets we use).

For tangential drifting, since it will generate outlier points, a natural consideration is to make use of it for a penalty term. We introduce a novel stabilization term that utilizes the outlier points to stabilize camera motion instead of simply discarding them. The stabilization term is given as

$$E_2 = \sum_{k\in\Omega}\left\|\boldsymbol{v}_t^k - \exp(\hat{\boldsymbol{\xi}})\boldsymbol{v}_t^k\right\|^2,$$

where Ω is the set of outlier points. As mentioned before, a point is treated as an outlier when it cannot find a match or the distance to its supposed match exceeds a threshold.

The effect of this term can be observed from an extreme case, where all points are outliers. In this case the lowest penalty energy is given when camera pose remains unchanged. Thus, this term utilizes outlier points to stabilize camera motion. This strategy is also reasonable in that between frames that contain many outliers, it is more proper to temporarily keep camera motion stable instead of allowing random registration solely by the remaining points. In practical SLAM tasks, this simple camera stabilization term can effectively reduce tangential drifting in the calculation. More details will be given in Subsection 4.1.

With the above-mentioned tangential weighting and stabilization terms, the final energy function of our approach is given as:

$$E = \sum_{k\in\Phi} \|\boldsymbol{D}^{\mathrm{T}}\tilde{\boldsymbol{G}}\boldsymbol{D}\| + t \cdot \sum_{k\in\Omega} \|\boldsymbol{v}_t^k - \exp(\hat{\boldsymbol{\xi}})\boldsymbol{v}_t^k\|^2, \quad (4)$$

where Φ contains all points except outliers, Ω contains the outlier points, and t is a weight factor of the stabilization term.

Our approach computes the energy in two following aspects: reconsidering normal and tangential mismatch distances by adaptively weighting the tangential mismatch distance through kernel matrix $\boldsymbol{G}$, and finding the trade-off between stabilization and movement of camera pose through the stabilization term.

3.3.2 Discussion Compared with NDT Methods

In our approach, (2) is different from the distance metric used in the NDT-based algorithms[2,18]. The NDT-based algorithms assume the normal-aligned error is much smaller than tangential errors from the input devices and their algorithms are designed to strengthen bias toward eliminating normal-aligned mismatch. As a result, they use the Mahalanobis distance instead of the Euclidean distance, which is effectively always the invert of the covariance matrix. The basic assumption of NDT-based algorithms applies well in robotics, since the main focus there is to recover registration from laser-scanned data, which has high confidence on the normal direction. However, for general RGB-D cameras used in computer vision, certain amount of input error or noise within any direction is inevitable, and mismatching displacement along different directions should be treated equally. We show in Subsection 4.2 that in such data sequences our approach performs better than the NDT-based assumption that uses the Mahalanobis distance.

It is also interesting to note that the covariance matrix in (2) is assumed to be always invertible in the NDT-based methods. Since the covariance matrix is symmetric and positive-semidefinite, if the assumption that it is invertible is followed, then (2) is a generalized quadratic distance metric that weights mismatch distances in different directions with an ellipsoidal kernel ($\tilde{\boldsymbol{G}}$) instead of an isotropic kernel ($\boldsymbol{I}$) used in the point-to-plane distance metric[52]. Note this also means our term, developed from a completely different view aspect that directly analyzes drawbacks of original ICP algorithm, is not mathematically equivalent to variants of NDT-based terms, since we do not need any approximation or invertible assumption to strictly cover original ICP term by setting $\boldsymbol{G} = \boldsymbol{I}$.

4 Experiments

We choose the state-of-the-art dense visual SLAM system ElasticFusion[20,28] as our main base system for the comparison. Their online codes not only provide better original-ICP registration results when only using depth information than other Lie-algebra based systems such as InfiniTAM[45], but also provide many options such as loop closure. In addition to experiments on Elasticfusion, we also test our algorithm on Kintinuous[21]. We run the online codes of these open-source systems on an NVIDIA GeForce GTX 970 graphics card and obtain the comparison data from their default parameters in order to evaluate the overall performances, and we assign $k_r = 5$, $k_n = 0.01$, $t = 0.3$ in all our experiments unless otherwise explicitly stated. We perform the experiments on TUM RGB-D benchmark[46] and evaluate the registration performance using the absolute trajectory (ATE) root-mean-square error metric (RMSE)[46] criteria.

In the comparisons, we first only use the depth information in the experiments in Subsection 4.1 and Subsection 4.2 to demonstrate the effectiveness of our approach upon itself. However, other features of the input data or improvement technique over ICP can be easily integrated by simply substituting our energy terms for previous distance metric term, and we show the comparison of integration results with some common methods in Subsection 4.3. In the following, "default ICP" refers to the online ICP code of ElasticFusion and Kintinuous using point-to-plane ICP. For speed performance, calculating ICP using the default code of ElasticFusion runs at 4.5 ms/frame; after substituting our energy terms for the default distance metric term into the ElasticFusion

code, our approach runs at 12.1 ms/frame, achieving good convergence with real-time performance.

4.1 Comparison with Point-to-Plane ICP

We compare our approach with the point-to-plane ICP in both frame-to-frame strategy and frame-to-model strategy.

Table 1 shows the comparison result in frame-to-frame SLAM tasks. The stabilization term is not included in our approach in this comparison. Our approach achieves better performance in almost all the data sequences used in the test, with an average 50% improvement of pose estimation accuracy.

The frame-to-model strategy is generally considered to be more robust than the frame-to-frame strategy, and is adopted by most recent SLAM systems. Our following experiments will be mainly with this strategy. In this subsection, we use ElasticFusion as our base system.

Table 1. Comparison of ATE RMSE on the TUM RGB-D Benchmark[46] Between Point-to-Plane ICP (PPICP) and Our ICP in Frame-to-Frame Manner

	1_360	1_desk	1_room	1_rpy	2_xyz	2_desk	3_cab	3_lcab	3_loh	3_snf	3_snn
PPICP	0.175 2	0.078 0	0.229 2	0.075 6	0.183 3	1.150 3	0.586 2	1.203 7	0.519 4	0.180 7	0.178 8
Ours	**0.149 1**	**0.066 4**	**0.189 2**	**0.053 4**	**0.087 0**	**0.610 5**	**0.366 0**	**0.708 9**	**0.317 5**	**0.076 7**	**0.144 1**

Note: Stabilization term is not included. Bold numbers are the better ones.

Table 2 and all following tables illustrate the result of pose estimation on different data sequences using frame-to-model manner. In Table 2, on most data sequences, the performance of our approach largely exceeds that of the point-to-plane ICP. Table 2 also shows quantitatively the separate improvement of each of our modifications when applied alone. The "metric" column shows the results applying $\tilde{\boldsymbol{G}}$ but without the stabilization term. The "stabilization" column shows the results directly adding the stabilization term to the energy function of default ICP. One can observe that enhancements in rotational estimation and tangential drifting have different extent of importance depending on data sequences, but each of them as well as their combination has better performance than the default ICP on almost all the datasets.

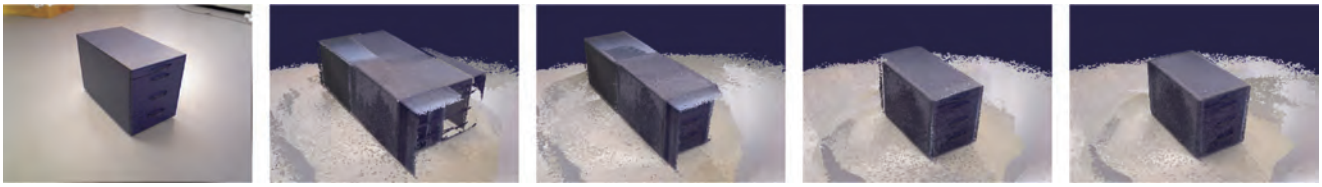
Table 2. Performance Comparison of Original Point-to-Plane ICP and Our Improved ICP Energy Function Formulations

Sequence	Default	Metric	Stabilization	Both
1_360	0.170 2	0.175 1	0.170 9	**0.116 8**
1_desk	1.006 4	**0.049 7**	0.051 4	0.057 5
1_room	0.431 2	0.190 1	**0.186 0**	0.188 5
1_rpy	0.032 4	0.031 3	0.030 7	**0.030 7**
2_xyz	0.019 9	0.019 7	**0.019 4**	0.019 7
2_desk	0.133 0	0.117 6	0.121 1	**0.117 3**
3_cab	0.394 8	0.350 2	0.064 1	**0.035 3**
3_lcab	0.176 0	**0.089 6**	0.120 7	0.118 2
3_loh	0.122 7	**0.093 2**	0.095 6	0.097 0
3_snf	0.068 8	0.030 9	0.030 6	**0.030 3**
3_snn	**0.020 3**	0.025 4	0.025 2	0.024 6

Note: default: the ATE values of the original ICP; metric: our ICP with only distance metric modified; stabilization: our ICP with only stabilization term; both: our ICP with both modification applied. Bold numbers are the better ones.

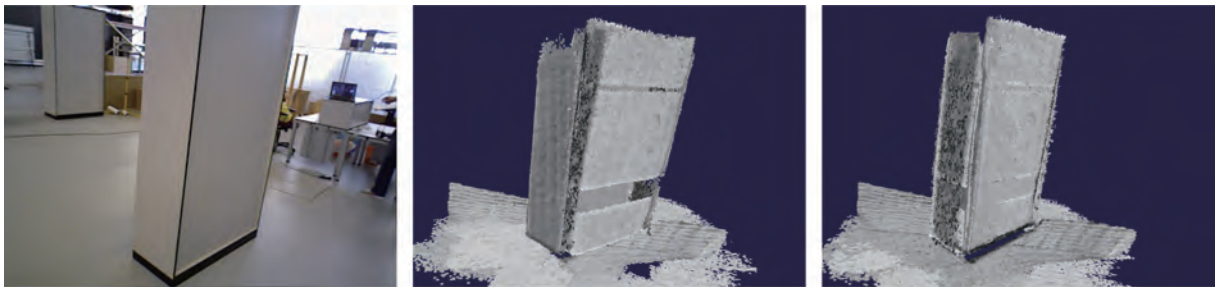
Figs.3 and 4 qualitatively show the registration results generated by our approach and the point-to-plane ICP. In Fig.3, both the modified distance metric and the stabilization term contribute to a better result. Note how the modified distance metric corrects erroneous rotational estimation in default ICP, and the stabilization term suppresses tangential drifting. Fig.4 is another data sequence easy to cause camera pose drift. Again our approach is capable of correctly registering with depth information alone.

We note that in Table 2 some ATE RMSE values are slightly higher when both modification to distance metric and stabilization term are applied than when only one of them is applied. We take the result on 1_desk depth input as an example. In Fig.5 we show the reconstruction results around the critical frames that point-to-plane ICP fails. It can be observed that at that time the input frame contains only a few large planes, and as a result erroneous rotational estimation and large drifting both occur using point-to-plane ICP, which exactly corresponds to our analysis in Subsection 3.1. Applying either part of our approach can correct the estimation at this critical point. This example shows that one reason for better results obtained by our approach is that it automatically enhances camera pose estimation especially at critical input frames where point-to-plane ICP is easy to fail. When the two aspects of our approach complement each other, such as in 1_360, and 3_cab, the result can improve significantly. When the two aspects correlate as in 1_desk, or one of them is improper to apply (e.g., in the extreme case, no worry of tangential drifting leads to no need of camera stabilization), applying only one of them may be a better choice,



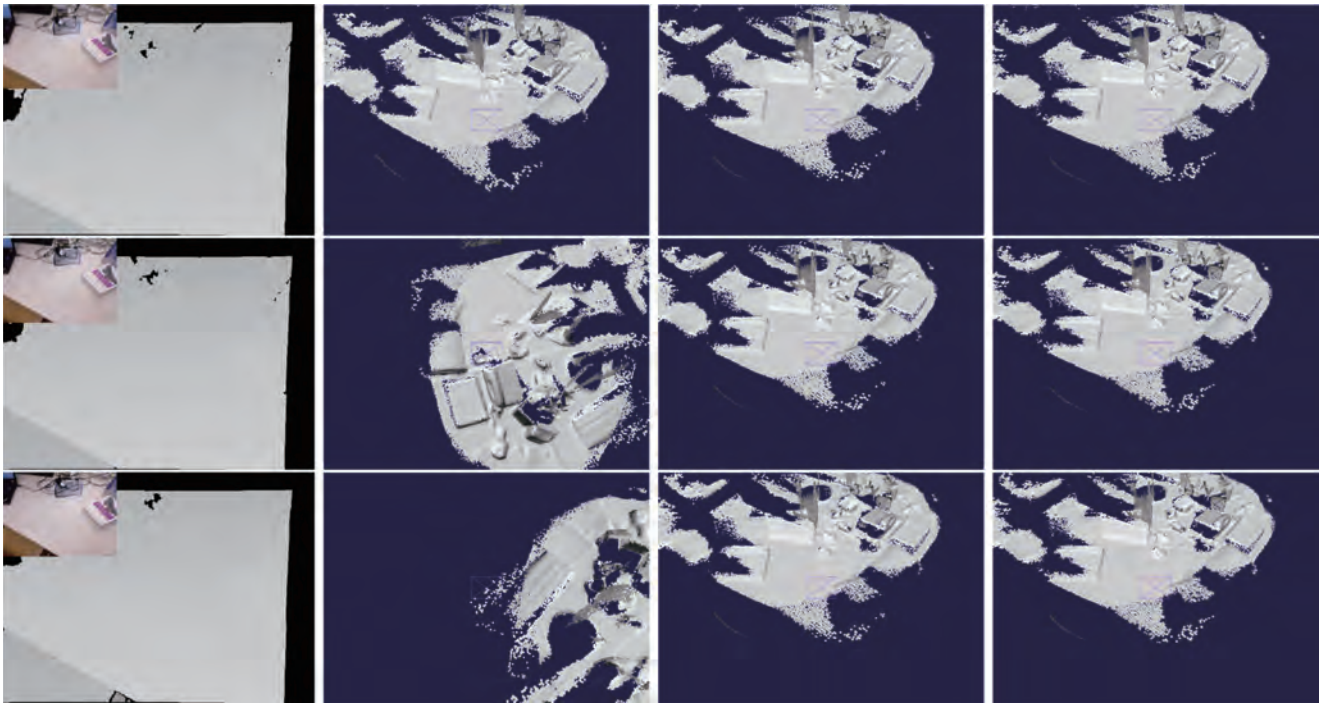
(a) (b) (c) (d) (e)

Fig.3. Qualitative comparison of scene models reconstructed by different algorithms on 3_cabinet sequence. (a) Reference original color image from the data sequence. (b) Reconstruction result of ElasticFusion. (c)–(e) Results of our method with only modified distance metric, only adding stabilization term and combination of the two factors, respectively.



(a) (b) (c)

Fig.4. Qualitative comparison of scene models of 3_large_cabinet data sequence reconstructed by default ICP and our approach on ElasticFusion. (a) Reference original image from the data sequence. (b) Result of default ICP. (c) Result of our approach.



(a) (b) (c) (d)

Fig.5. Illustration of critical frames using 1_desk. The top, middle and bottom rows are the inputs and reconstruction outputs using depth information at three sequential frames (252–254) by ElasticFusion. (a) Input depth data (with RGB input at upper-left corner as reference). (b) Camera pose estimation of default ICP. (c)(d) Camera pose estimation applying our modification to distance metric and applying the stabilization term, respectively. It is clear that the default ICP behaves just as our analysis in Subsection 3.1, and both of our two enhancement aspects are able to provide correct camera pose estimation at this critical point.

though in Table 2 we can observe in most of such times the difference between "both" and using either single strategy is small compared with the improvement to point-to-plane ICP.

It is worth noting that datasets in the TUM RGB-D benchmark all contain more or less some planar features. Our approach has shown performance from being at least acceptable to large improvements in all experiments. This includes datasets that contain many small objects with non-aligning surface normals (e.g., 1_desk, 1_rpy, and 1_360) and shows the generality of our approach.

4.2 Comparison with Distance Metric from NDT

We also test the performance of Mahalanobis distance used by the NDT-based methods on the data sequences both with and without our stabilization term. Shown in Table 3, our approach applies much better in more than 70% of the data sequences tested, with about half of the data sequences having large improvements, and is hardly worse in the remaining data sequences.

4.3 Further Integration with Existing Methods

In this subsection we demonstrate the flexibility of our approach in combination with existing variations of the ICP algorithm. We test the performance on both surfel-based ElasticFusion and voxel-based Kintinuous. The latter integrates color information and loop closure optimization. In experiments of this subsection, we always use our full energy function (4) in our approach. Shown in Table 4, in general, our approach produces better results under various combinations of strategies, and has comparable results on certain datasets that already show small ATE values (less than 0.04).

Combination with Color Information. Color information is proved to be useful in SLAM problems, and is popularly used by recent registration algorithms. Table 4 shows a comparison result using Kintinuous system with the color information combined in the optimization. It is to be noted that the modification to the distance metric in our approach also affects the choice of weight factor w before color energy term. In our experiments w is set to 250. For the Kintinuous system we set $t = 0.000\,5$, and our result has generally smaller ATE RMSE values on most of the data sequences. Table 4 shows a comparison result using ElasticFusion, where our approach also shows better results.

Table 3. Comparison of Our Approach Using Covariance (C) Matrix and Using Mahalanobis Distance Which Uses Inverse Covariance (IC) Matrix Adopted by NDT-Based Methods

Sequence	C	IC	C+S	IC+S
1_360	**0.175 1**	0.504 1	**0.116 8**	0.282 5
1_desk	**0.049 7**	1.932 3	**0.057 5**	0.738 5
1_room	0.190 1	**0.163 9**	0.188 5	**0.164 8**
1_rpy	**0.031 3**	0.031 4	**0.030 7**	0.031 3
2_xyz	0.019 7	**0.018 5**	0.019 7	**0.018 5**
2_desk	0.117 6	**0.115 9**	0.117 3	**0.115 8**
3_cab	**0.350 2**	0.500 3	**0.035 3**	0.523 9
3_lcab	**0.089 6**	0.140 0	**0.118 2**	0.139 2
3_loh	**0.093 2**	0.119 0	**0.097 0**	0.121 9
3_snf	**0.030 9**	0.085 4	**0.030 3**	0.085 1
3_snn	**0.025 4**	0.065 1	**0.024 6**	0.188 9

Note: We show both situations with and without the stabilization (S) term. Our approach applies much better in more than 70% of the data sequences tested, and is hardly worse in the remaining data sequences. Bold numbers are the better ones.

Table 4. Five Groups (a)–(e) of Comparisons Between Different Options of Kintinuous (Kint) and ElasiticFusion (Elas) Measured by ATE RMSE

Setting			Sequence										
			1_360	1_desk	1_room	1_rpy	2_xyz	2_desk	3_cab	3_lcab	3_loh	3_snf	3_snn
Kint	(a)	c, d	0.147 0	0.083 3	0.217 4	**0.035 4**	0.057 8	0.115 4	0.200 6	0.089 1	0.045 9	**0.021 4**	0.031 3
		c, o	**0.119 9**	**0.074 0**	**0.180 4**	0.044 4	**0.022 5**	**0.078 6**	**0.029 8**	**0.062 3**	**0.029 9**	0.022 4	**0.030 9**
Elas	(b)	d	0.170 2	1.006 4	0.431 2	0.032 4	0.019 9	0.133 0	0.394 8	0.176 0	0.122 7	0.068 8	**0.020 3**
		o	**0.116 8**	**0.057 5**	**0.188 5**	**0.030 7**	**0.019 7**	**0.117 3**	**0.035 3**	**0.118 2**	**0.097 0**	**0.030 3**	0.024 6
	(c)	c, d	0.273 0	**0.025 6**	0.224 2	0.040 8	0.012 9	0.073 2	1.008 4	0.597 0	**0.022 6**	**0.027 9**	0.840 5
		c, o	**0.252 3**	0.030 7	**0.174 6**	**0.034 2**	**0.012 2**	**0.070 2**	**0.809 1**	**0.066 9**	0.032 4	0.037 2	**0.030 2**
	(d)	l, d	0.170 2*	1.006 4*	0.431 2*	0.032 4*	**0.018 7**	**0.094 2**	0.526 6	0.176 0*	0.106 5	0.068 8*	**0.020 3***
		l, o	**0.116 8***	**0.058 1**	**0.188 5***	**0.030 7***	0.019 3	0.117 3*	**0.035 3***	**0.118 2***	**0.097 0***	**0.030 3***	0.024 6*
	(e)	c, l, d	0.273 0*	**0.025 5**	0.224 2*	0.040 8*	0.012 0	0.077 4	1.008 4*	0.597 0*	**0.023 7**	**0.027 9***	0.538 3
		c, l, o	**0.252 3***	0.031 5	**0.174 6***	**0.034 2***	**0.011 9**	**0.070 2***	**0.809 1***	**0.066 9***	0.032 4*	0.037 2*	**0.030 2***

Note: These options include: with color (c), with loop closure (l), using our ICP (o), and using default ICP (d). The star on the number means that the data sequences have not triggered the loop closure. Bold numbers are the better ones in each group.

Combination with Loop Closure Optimization. To make full use of the visual information, recent visual SLAM systems have integrated loop closure detection as an important technique to reduce accumulated drift when the camera observes old landmarks. We utilize the loop closure algorithm in ElasticFusion and test the performance of our approach combined with loop closure optimization. Shown in Table 4, our approach also performs better on most of the data sequences.

Combination with Both Color and Loop Closure. An integration of combining both color information and loop closure optimization, based on the ElasticFusion system, is also tested in our experiments. Again our performance has smaller ATE RMSE values in most of the data sequences shown in Table 4.

4.4 Limitation Discussion

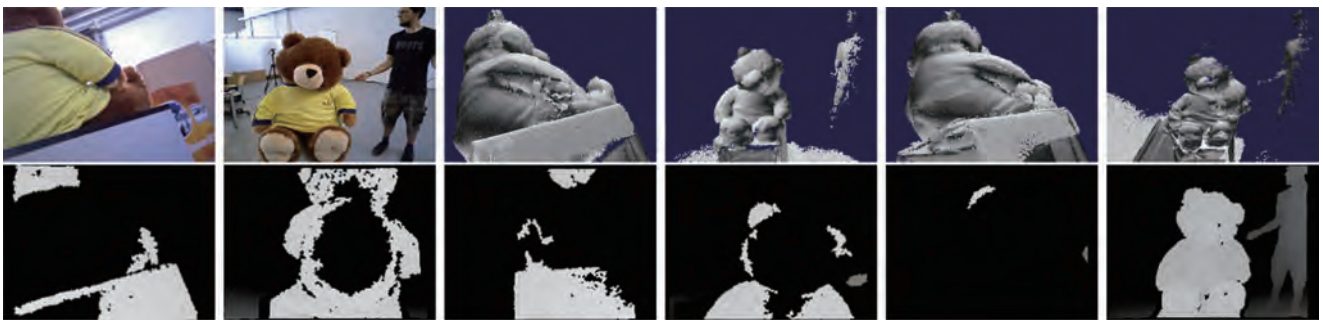
In general, our approach concentrates on handling the erroneous rotation and tangential drifting due to the default ICP algorithm. It does not enhance performance over failure cases due to other issues such as rapid jittering, fast motion of the camera, or missing data inputs. That is, our algorithm follows the common assumption that small value of ξ is not much violated, and the stabilization term can be less effective when input sequence contains rapid jittering or fast motion of the camera. Local geometric estimation in these cases can also be worse, resulting in inaccurate computation of distance metric. In such cases our approach on depth alone can give only comparable results (e.g., fast rotation case in 1_rpy, jittering in 3_snn). We note that in the cases on which our approach does not perform well, the default ICP usually does not perform well either. Alternative assistant techniques such as providing better initial guesses will be further needed to improve the performance on related datasets.

It is to be noted that in general cases, the stabilization term has a limitation that it can count in newly appeared points when there is no match for them. Theoretically, these points should not prevent the camera to move. However, if we assume the camera moves slowly, new points are revealed under the following two situations. 1) New points are revealed on the "boundary" of existing planes. Such newly revealed points are much fewer than existing points, and the stabilization term is always 10^2–10^4 times smaller than the other term in good matching, thereby the possible induced error is small. 2) A new plane is revealed. The new plane is "new" only at one single frame, and the possible temporary influence can be corrected by later calculations. Our experiments have shown that the small possible drawbacks during good matching are out-weighted by the total benefit of preventing large tangential drifting.

A failure case of our approach is shown in Fig.6. We perform this experiment on the 3_teddy dataset on an NVIDIA GeForce 1080Ti GPU. In this dataset, the depth input sequence contains lots of frames with excessive missing depth values. Unlike in other datasets, in 3_teddy the outlier points are so many that they begin to prevent camera motion in a long time (more than 500 frames), leading to pose estimation failure before the depth input finally becomes good again.

5 Conclusions

In this paper a novel energy function formulation is proposed for the ICP algorithm, which reduces rotational estimation error and tangential drifting of camera pose. Experiments based on state-of-the-art real-time



(a) (b) (c) (d) (e) (f)

Fig.6. Failure case due to excessive missing input data. Upper row: (a)(b) the 1 500th and the 2 048th frame of input video, respectively. (c)(d) Reconstructed models of default ICP at those frames, respectively. (e)(f) Reconstructed models of our approach at those frames, respectively. Bottom row: (a)–(f) Depth data from intermediate frames 1 500, 1 560, 1 708, 1 898, 1 942, 2 048, respectively. Excessive missing data causes the stabilization term to prevent camera moving, resulting in the askew shape of our approach at frame 2 048.

3D dense reconstruction platforms, e.g., ElasticFusion and Kintinuous, showed that our method outperforms previous matching energy function variations on the TUM RGB-D SLAM dataset using the ATE RMSE criteria. The proposed method is simple, fast yet effective. Moreover, our solution is readily integratable with previous ICP algorithms using color or loop closure.

For future work, we would like to in-depth investigate the problem of adaptively integrating our distance metric with previous ICP algorithms for the best registration result on different data sequences, e.g., adaptively determining the best color weighting when using the color information.

References

[1] Besl P J, McKay N D. Method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, 14(2): 239-256.

[2] Segal A, Hähnel D, Thrun S. Generalized-ICP. In *Proc. Robotics: Science and Systems*, June 2009, Article No. 21.

[3] Steinbrücker F, Sturm J, Cremers D. Real-time visual odometry from dense RGB-D images. In *Proc. the 2011 IEEE International Conference on Computer Vision Workshops*, November 2011, pp.719-722.

[4] Kerl C, Sturm J, Cremers D. Robust odometry estimation for RGB-D cameras. In *Proc. the 2013 IEEE International Conference on Robotics and Automation*, May 2013, pp.3748-3754.

[5] Tam G K, Cheng Z Q, Lai Y K, Langbein F C, Liu Y, Marshall D, Martin R R, Sun X F, Rosin P L. Registration of 3D point clouds and meshes: A survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics*, 2013, 19(7): 1199-1217.

[6] Salvi J, Matabosch C, Fofi D, Forest J. A review of recent range image registration methods with accuracy evaluation. *Image and Vision Computing*, 2007, 25(5): 578-596.

[7] Rusinkiewicz S, Levoy M. Efficient variants of the ICP algorithm. In *Proc. the 3rd International Conference on 3D Digital Imaging and Modeling*, May 2001, pp.145-152.

[8] Newcombe R A, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison A J, Kohi P, Shotton J, Hodges S, Fitzgibbon A. KinectFusion: Real-time dense surface mapping and tracking. In *Proc. the 10th IEEE International Symposium on Mixed and Augmented Reality*, October 2011, pp.127-136.

[9] Izadi S, Kim D, Hilliges O *et al.* KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera. In *Proc. the 24th Annual ACM Symposium on User Interface Software and Technology*, October 2011, pp.559-568.

[10] Henry P, Krainin M, Herbst E, Ren X, Fox D. RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments. *The International Journal of Robotics Research*, 2012, 31(5): 647-663.

[11] Huang A S, Bachrach A, Henry P, Krainin M, Maturana D, Fox D, Roy N. Visual odometry and mapping for autonomous flight using an RGB-D camera. In *Proc. the 15th International Symposium on Robotics Research*, December 2017, pp.235-252.

[12] Rusinkiewicz S, Hall-Holt O, Levoy M. Real-time 3D model acquisition. *ACM Transactions on Graphics*, 2002, 21(3): 438-446.

[13] Curless B, Levoy M. A volumetric method for building complex models from range images. In *Proc. the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, August 1996, pp.303-312.

[14] Simon D A. Fast and accurate shape-based registration [Ph.D. Thesis]. Robotics Institute, Carnegie Mellon University, 1996.

[15] Johnson A E, Kang S B. Registration and integration of textured 3D data. *Image and Vision Computing*, 1999, 17(2): 135-147.

[16] Jin H, Favaro P, Soatto S. Real-time feature tracking and outlier rejection with changes in illumination. In *Proc. the 8th International Conference on Computer Vision*, July 2001, pp.684-689.

[17] Chen Y, Medioni G. Object modelling by registration of multiple range images. *Image and Vision Computing*, 1992, 10(3): 145-155.

[18] Biber P, Straßer W. The normal distributions transform: A new approach to laser scan matching. In *Proc. the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2003, pp.2743-2748.

[19] Magnusson M, Lilienthal A, Duckett T. Scan registration for autonomous mining vehicles using 3D-NDT. *Journal of Field Robotics*, 2007, 24(10): 803-827.

[20] Whelan T, Leutenegger S, Salas-Moreno R F, Glocker B, Davison A J. ElasticFusion: Dense SLAM without a pose graph. In *Proc. Robotics: Science and Systems XI*, July 2015, Article No. 1.

[21] Whelan T, Kaess M, Fallon M, Johannsson H, Leonard J, McDonald J. Kintinuous: Spatially extended kinectFusion. In *Proc. Robotics: Science and Systems Workshop on RGB-D: Advanced Reasoning with Depth Cameras,* July 2012.

[22] Pomerleau F, Colas F, Siegwart R, Magnenat S. Comparing ICP variants on real-world data sets — Open-source library and experimental protocol. *Autonomous Robots*, 2013, 34(3): 133-148.

[23] Holz D, Ichim A E, Tombari F, Rusu R B, Behnke S. Registration with the point cloud library: A modular framework for aligning in 3-D. *IEEE Robotics & Automation Magazine*, 2015, 22(4): 110-124.

[24] Whelan T, Johannsson H, Kaess M, Leonard J J, McDonald J. Robust real-time visual odometry for dense RGB-D mapping. In *Proc. the 2013 IEEE International Conference on Robotics and Automation*, May 2013, pp.5724-5731.

[25] Choi S, Zhou Q Y, Koltun V. Robust reconstruction of indoor scenes. In *Proc. the 2015 IEEE Conference on Computer Vision and Pattern Recognition*, June 2015, pp.5556-5565.

[26] Valentin J, Vineet V, Cheng M M, Kim D, Shotton J, Kohli P, Nießner M, Criminisi A, Izadi S, Torr P. SemanticPaint: Interactive 3D labeling and learning at your fingertips. *ACM Transactions on Graphics*, 2015, 34(5): Article No. 154.

[27] Kähler O, Prisacariu V A, Ren C Y, Sun X, Torr P, Murray D. Very high frame rate volumetric integration of depth images on mobile devices. *IEEE Transactions on Visualization and Computer Graphics*, 2015, 21(11): 1241-1250.

[28] Whelan T, Salas-Moreno R F, Glocker B, Davison A J, Leutenegger S. ElasticFusion: Real-time dense SLAM and light source estimation. *The International Journal of Robotics Research*, 2016, 35(14): 1697-1716.

[29] Hu R, Wen C, van Kaick O, Chen L, Lin D, CohenOr D, Huang H. Semantic object reconstruction via casual handheld scanning. *ACM Trans. Graph.*, 2018, 37(6): Article No. 219.

[30] Cheng M, Hou Q, Zhang S, Rosin P L. Intelligent visual media processing: When graphics meets vision. *J. Comput. Sci. Technol.*, 2017, 32(1): 110-121.

[31] Whelan T, Kaess M, Leonard J J, McDonald J. Deformation-based loop closure for large scale dense RGB-D SLAM. In *Proc. the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, November 2013, pp.548-555.

[32] Pirker K, Rüther M, Schweighofer G, Bischof H. GPSlam: Marrying sparse geometric and dense probabilistic visual mapping. In *Proc. the 22nd British Machine Vision Conference*, August 2011, Article No. 102.

[33] Konolige K, Agrawal M. FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 2008, 24(5): 1066-1077.

[34] Davison A J. Real-time simultaneous localisation and mapping with a single camera. In *Proc. the 9th IEEE International Conference on Computer Vision*, October 2003, pp.1403-1410.

[35] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces. In *Proc. the 6th IEEE/ACM International Symposium on Mixed and Augmented Reality*, November 2007, pp.225-234.

[36] Dai A, Nießner M, Zollhöfer M, Izadi S, Theobalt C. BundleFusion: Real-time globally consistent 3D reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics*, 2017, 36(3): Article No. 24.

[37] Granger S, Pennec X. Multi-scale EM-ICP: A fast and robust approach for surface registration. In *Proc. the 7th European Conference on Computer Vision*, May 2002, pp.418-432.

[38] Liu Y. A mean field annealing approach to accurate free form shape matching. *Pattern Recognition*, 2007, 40(9): 2418-2436.

[39] Rangarajan A, Chui H, Mjolsness E, Pappu S, Davachi L, Goldman-Rakic P, Duncan J. A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis*, 1997, 1(4): 379-398.

[40] Bylow E, Sturm J, Kerl C, Kahl F, Cremers D. Real-time camera tracking and 3D reconstruction using signed distance functions. In *Proc. Robotics: Science and Systems IX*, June 2013, Article No. 35.

[41] Jian B, Vemuri B C. Robust point set registration using Gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, 33(8): 1633-1645.

[42] Tsin Y, Kanade T. A correlation-based approach to robust point set registration. In *Proc. the 8th European Conference on Computer Vision*, May 2004, pp.558-569.

[43] Song P. Local voxelizer: A shape descriptor for surface registration. *Computational Visual Media*, 2015, 1(4): 279-289.

[44] Nießner M, Zollhöfer M, Izadi S, Stamminger M. Real-time 3D reconstruction at scale using voxel hashing. *ACM Trans. Graph.*, 2013, 32(6): Article No. 169.

[45] Prisacariu V A, Kahler O, Cheng M M, Ren C Y, Valentin J, Torr P H S, Reid I D, Murray D W. A framework for the volumetric integration of depth images. arXiv: 1410.0925, 2014. https://arxiv.org/abs/1410.0925, March 2019.

[46] Sturm J, Engelhard N, Endres F, Burgard W, Cremers D. A benchmark for the evaluation of RGB-D SLAM systems. In *Proc. the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2012, pp.573-580.

[47] Kraft M, Nowicki M, Schmidt A, Fularz M, Skrzypczyński P. Toward evaluation of visual navigation algorithms on RGB-D data from the first- and second-generation Kinect. *Machine Vision and Applications*, 2017, 28(1/2): 61-74.

[48] Magnusson M. The three-dimensional normal-distributions transform: An efficient representation for registration, surface analysis, and loop detection [Ph.D. Thesis]. Örebro University, 2009.

[49] Huhle B, Magnusson M, Straßer W, Lilienthal A J. Registration of colored 3D point clouds with a kernel-based extension to the normal distributions transform. In *Proc. the 2008 IEEE International Conference on Robotics and Automation*, May 2008, pp.4025-4030.

[50] Pulli K. Multiview registration for large data sets. In *Proc. the 2nd International Conference on 3D Digital Imaging and Modeling*, October 1999, pp.160-168.

[51] Dorai C, Wang G, Jain A K, Mercer C. Registration and integration of multiple object views for 3D model construction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, 20(1): 83-89.

[52] Abou-Moustafa K, Ferrie F P. Local generalized quadratic distance metrics: Application to the k-nearest neighbors classifier. *Advances in Data Analysis and Classification*, 2018, 12(2): 341-363.

Bo Ren received his Ph.D. degree in computer science from Tsinghua University, Beijing, in 2015. He is currently a lecturer in the College of Computer Science, Nankai University, Tianjin. His research interests include physically-based simulation and rendering, scene geometry reconstruction and analysis. His recent research focuses on multi-fluid and multi-phase simulations in computer graphics.

Jia-Cheng Wu received his B.S. degree in computer science from Nankai University, Tianjin, in 2017. His research interests include 3D SLAM, scene reconstruction, databases, and computer architecture.

Ya-Lei Lv is a senior undergraduate majoring computer science at Nankai University, Tianjin. Her research interests lie in computer graphics and computer vision.

Ming-Ming Cheng received his Ph.D. degree in computer science from Tsinghua University, Beijing, in 2012, supervised by Prof. Shi-Min Hu. Then he did two years research fellow, with Prof. Philip Torr in Oxford. He is currently an associate professor at Nankai University, Tianjin. His research interests include computer graphics, computer vision, and image processing. He has received the Google Ph.D. Fellowship Award, the IBM Ph.D. Fellowship Award, etc.

Shao-Ping Lu is currently an associate professor at Nankai University, Tianjin. In 2013–2017, he worked as a postdoc and senior researcher at Vrije Universiteit Brussel (VUB). He received his Ph.D. degree in computer science at Tsinghua University, Beijing, in 2012. His research interests lie primarily in the intersection of visual computing, computer vision and computer graphics, with particular focus on 2D&3D image and video processing, computational photography and representation, visual scene analysis, machine learning, and mathematical optimization.