DOI: 10.1111/cgf.14976
Pacific Graphics 2023
R. Chaine, Z. Deng, and M. H. Kim
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 42 (2023), Number 7

3D Object Tracking for Rough Models

Xiuqiang Song[1,6] and Weijian Xie[2,4] and Jiachen Li[2] and Nan Wang[4,5] and Fan Zhong[3,6] and Guofeng Zhang[†2] and Xueying Qin[†1,6]

[1] School of Software, Shandong University, China [2] State Key Lab of CAD&CG, Zhejiang University, China
[3] School of Computer Science and Technology, Shandong University, China [4] SenseTime Research, China [5] Tetras.AI, China
[6] Engineering Research Center of Digital Media Technology, Ministry of Education, China

Abstract

Visual monocular 6D pose tracking methods for textureless or weakly-textured objects heavily rely on contour constraints established by the precise 3D model. However, precise models are not always available in reality, and rough models can potentially degrade tracking performance and impede the widespread usage of 3D object tracking. To address this new problem, we propose a novel tracking method that handles rough models. We reshape the rough contour through the probability map, which can avoid explicitly processing the 3D rough model itself. We further emphasize the inner region information of the object, where the points are sampled to provide color constrains. To sufficiently satisfy the assumption of small displacement between frames, the 2D translation of the object is pre-searched for a better initial pose. Finally, we combine constraints from both the contour and inner region to optimize the object pose. Experimental results demonstrate that the proposed method achieves state-of-the-art performance on both roughly and precisely modeled objects. Particularly for the highly rough model, the accuracy is significantly improved (40.4% v.s. 16.9%).

CCS Concepts

• ***Computing methodologies*** → *Augmented reality; Object tracking;*

1. Introduction

Model-based 3D object tracking aims to continuously estimate precise 6DoF (Degree of Freedom) poses of rigid objects from monocular video frames. This fundamental problem in computer vision is widely used in various fields, such as augmented reality, robot grasping, automatic navigation, and education [LF05, LSFK10, MUS16].

For tracking textureless or weakly-textured objects, a precise 3D CAD model is essential. Since there are typically few features in the inner region of objects, tracking methods usually focus on the regions surrounding the object contours. Pose optimization constraints can be established by aligning the projected contour of the 3D model with the implicit segmentation contour of the input frame, such as region-based [TSSC19, SPS*22] or edge-based [WZQ17, TLZQ22] methods.

Unfortunately, precise CAD models are not always available or reconstructable. For example, when reconstructing artifacts in a museum, reflections and obstructions from glass cases can introduce inaccuracies. Similarly, in scenarios with only RGB sensors and low computational power, reconstructing textureless objects can be challenging. In the tracking process, rough models easily lead to tracking drift and failure due to the lack of precise contour constraints, limiting the wide application of 3D object tracking.

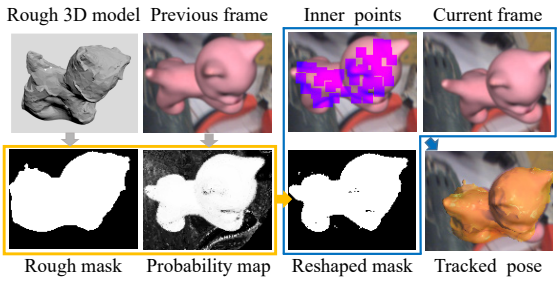

Figure 1: *Imprecise contour of the rough model will result in false countour matches. We use probability maps to reshape the contour while incorporating inner points to track the object accurately.*

One intuitive approach for handling rough models of textureless or weakly-textured objects is to operate directly with the imprecise object contour, ensuring that the "incorrect" projected contour correctly matches the object contour in the frame. Correcting the rough model's projected contour is crucial in this process. Additionally, the inner region information can provide additional constraints to

† Corresponding authors.
e-mail: zhangguofeng@zju.edu.cn
e-mail: qxy@sdu.edu.cn

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

weaken the negative effects of imprecise contours. However, due to the lack of distinct features in the inner region of textureless objects, directly sampling inner points is difficult to establish a stable correspondence between frames.

To address the issues above, we propose a novel 3D object tracking method that bridges the gap between rough and precise models. The key ideas are reshaping the projected rough contour (mask) to make it more precise and combining the color constraints of the pixels within the object. The main process is shown in Fig. 1. By employing the probability map to reshape the projected contour of the rough 3D model, we can get a more precise contour to mitigate the impact of model inaccuracies, and the inner points of the object are also used to provide additional constraints for better pose estimation. The contributions of this paper are listed as follows:

1. We propose a 3D object tracking method that is robust to the modeling error of the objects. To the best of our knowledge, it is the first 3D object tracking method that can deal with rough models, and for precise models it also works well.
2. We introduce a method to correct the projected contours of the rough model based on probability map. The corrected contours then are used for establishing new contour constraints so that the object can be tracked accurately.
3. We propose a multi-region sampling strategy in order to leverage the constraints of inner regions, and a 2D region pre-search strategy is also introduced to deal with large displacements.

2. Related Works

Recent textureless 3D object tracking algorithms can be classified into three major categories: edge-based methods, region-based methods, and direct methods. Due to space limitations, we will only introduce the methods that are closely related to our work.

Region-based and Edge-based Methods. Region-based methods [PR12, ZWS*14, HH16, TSS16, TSS17, TSSC19, ZZZ*20, SPS*20, SPS*22, HZQ22] have demonstrated outstanding tracking performance in recent years. These approaches employ color statistical segmentation models to capture the implicit contour of the object and align it with the contour projected by the 3D model to optimize the pose. The quality of the statistical segmentation model is crucial to tracking accuracy. Common statistical segmentation models include the global models [PR12, SPS*20], multiple local circular models [TSS17, TSSC19], and fan-shaped models [ZZZ*20]. Tracking is particularly challenging when the foreground and background colors are similar.

Edge-based methods [HS90, DC02, SPP*14, IP15, WZQ17, CRV*18, WZQ19, HZSQ20] generally begin by detecting the object's edges and then matching them with the contours projected by the 3D model to optimize the object's pose. However, these methods can be easily disrupted by chaotic backgrounds. Consequently, color information is often incorporated to aid in identifying the object's contour in the image. To achieve more robust tracking results, some methods [LSZQ21, LZXQ21, HZQ22, TLZQ22] combine both region and edge constraints.

However, neither region-based nor edge-based methods are effective in dealing with rough models. Establishing color statistical segmentation models requires precisely counting pixel colors in the foreground and background regions. For rough 3D models, the foreground and background segmented by the imprecise projected contour may not align with the actual object contour in the image, negatively affecting the quality of the color statistical segmentation models. Moreover, when optimizing the object's pose, it is necessary to project the 3D model's contour based on the initial pose and align it with the actual object's (implicitly) contour in the image. The projected rough contour is likely to result in false matches, leading to tracking failure.

Direct Methods. The direct methods optimize the pose by matching points between two adjacent frames through point color or local descriptors. These methods assume illumination invariance on the inner object points and small displacement of object motion. Many methods are dedicated to obtaining features that are more robust to illumination. For example, [CL14] calculates the gradient descriptor at each point and [SW16] proposed using the surface normal vector of the object to model the change of point under the Lambertian assumption. One problem of direct methods is that it produces cumulative errors. [ZZ19] combines region-based and direct methods, and uses a gradient descriptor that is more robust to illumination. However, this method does not significantly improve tracking accuracy. In addition, these methods all assume that the small displacement hypothesis holds, and therefore they have not made efforts to better meet this assumption.

Category-level 3D Object Tracking. Category-level 3D object tracking [WMX*20, WB21, WWZ*21] uses template models to track the different instance objects. The template model need to has some specific features, for example, a cup with a "handle". The generalization of category-level tracking methods is very limited due to the specific feature requirement. Overall, category-level object tracking and rough model tracking are two different problems.

Model-free Tracking. Some model-free methods [SWZ*22, HSW*22] require a set of RGB images with known object pose as priors to reconstruct a 3D point cloud of the object via Structure-from-Motion [SF16, SZFP16]. Then pose estimation is performed by matching the 3D point cloud with 2D images. Other methods, such as [LWP*22], directly use these priors to estimate the pose without reconstruction, and [YYH*22] also uses depth data. [WLP*22, NHX*22] are trained using some objects and then generalized to unseen objects, but [NHX*22] still need a precise 3D model of the unseen object as a prior during pose estimation. [WTB*23] performs reconstruction and tracking simultaneously with a RGBD sequence and an object mask as input, but it can only run offline. Typically, most RGB-based model-free methods rely on a series of priors, and often struggle to estimate the scale information of the object due to lack of a 3D model. Some methods [SWZ*22] can not handle textureless objects. In contrast, the scale information is naturally in the pose estimated by our method with the 3D model, and our method can handle textureless objects, does not suffer generalization issues, and does not need to reconstruction. In general, the model-free methods and our research are fundamentally different and applicable in different scenarios.

3. Preliminaries

In this section, we will first introduce the fundamental mathematical concepts in 3D object tracking, followed by an introduction

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

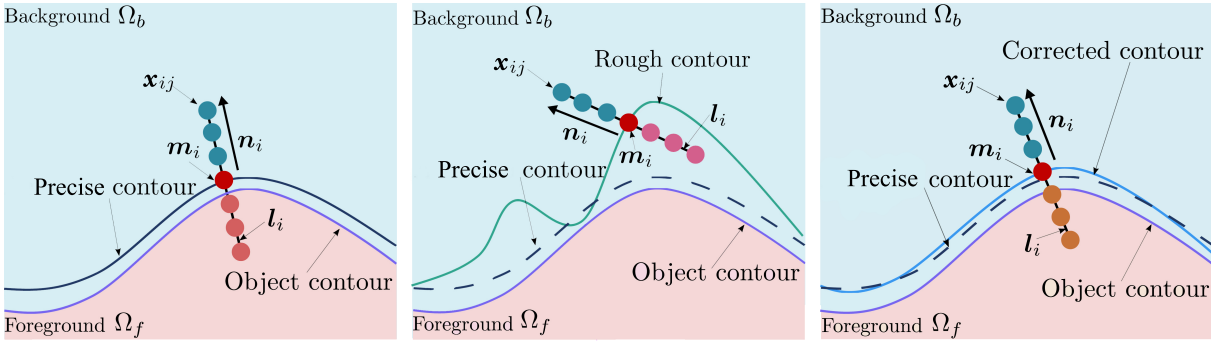

Figure 2: *Illustration of search lines. $\boldsymbol{l}_i \in L$ is a search line consists of center $\boldsymbol{m}_i$ and other points $\boldsymbol{x}_{ij}$, and $\boldsymbol{n}_i$ is the normal direction of the contour point. (a) A search line on the projected contour of an precise 3D model. (b) The search line $\boldsymbol{l}_i$ wrongly centered at $\boldsymbol{m}_i$ due to the rough model. (c) The corrected contour generated by the corrected depth map.*

of the probability map and then a brief overview of region-based constraints and our baseline method [HZQ22].

Fundamental Concepts. The 6DoF pose of an object can be represented by a Lie algebra $\boldsymbol{\xi} = [\omega_1, \omega_2, \omega_3, \upsilon_1, \upsilon_2, \upsilon_3]^\top \in \mathbb{R}^6$ [Var13]. The conversion between Lie algebra $\boldsymbol{\xi}$ and rigid body transformation matrix $\boldsymbol{T} \in \mathbb{SE}(3)$ can be achieved through exponential mapping $\exp(\cdot)$ and logarithmic mapping $\ln(\cdot)$:

$$\boldsymbol{T} = \exp(\hat{\boldsymbol{\xi}}) \tag{1}$$

$$\hat{\boldsymbol{\xi}} = \ln(\boldsymbol{T}) \tag{2}$$

where $\hat{\boldsymbol{\xi}} \in \mathbb{R}^{4\times4}$ is the anti-symmetric matrix corresponding to $\boldsymbol{\xi}$, and further description can be found in [AK08, Var13]. A 3D point $\boldsymbol{X} = [X, Y, Z]^\top$ in the world coordinate can be projected onto a 2D point $\boldsymbol{x} = [x, y]^\top$ in the image plane through $\boldsymbol{\xi}$ and the pre-calibrated camera intrinsic parameter matrix $\boldsymbol{K} \in \mathbb{R}^{3\times3}$:

$$\boldsymbol{x} = \pi(\boldsymbol{K}(\exp(\hat{\boldsymbol{\xi}})\tilde{\boldsymbol{X}})_{3\times1}) \tag{3}$$

where $\tilde{\boldsymbol{X}}$ represents the homogeneous form of $\boldsymbol{X}$, and $\pi(\boldsymbol{x}) = [X/Z, Y/Z]^\top$. Similarly, the corresponding 3D point $\boldsymbol{X}$ of a 2D point $\boldsymbol{x}$ can be determined with a known depth value Z:

$$\boldsymbol{X} = (\exp(\hat{\boldsymbol{\xi}})^{-1}(\boldsymbol{K}^{-1}(\pi^{-1}(\boldsymbol{x}, Z))_{4\times1})_{3\times1} \tag{4}$$

where $(\cdot)_{4\times1}$ represents the homogeneous form.

Probability Map. In 3D object tracking, the probability map is used to calculate the posterior probability of a pixel $\boldsymbol{x}$ belonging to the foreground $P_f(\boldsymbol{x})$ or the background $P_b(\boldsymbol{x})$ given the color of $\boldsymbol{y}$ for that pixel $\boldsymbol{x}$. The probability map is typically computed using Bayesian principles. This involves pre-computing the color distributions of both the foreground and background, which are then used to obtain the foreground statistical model M_f and the background statistical model M_b. Then $P_f(\boldsymbol{x})$ and $P_b(\boldsymbol{x})$ can be calculated:

$$\begin{aligned} P_f(\boldsymbol{x}) &= P(M_f \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_f)}{\eta_f P(\boldsymbol{y} \mid M_f) + \eta_b P(\boldsymbol{y} \mid M_b)} \\ P_b(\boldsymbol{x}) &= P(M_b \mid \boldsymbol{y}) = \frac{P(\boldsymbol{y} \mid M_b)}{\eta_f P(\boldsymbol{y} \mid M_f) + \eta_b P(\boldsymbol{y} \mid M_b)} \end{aligned} \tag{5}$$

where η_i is a smoothed function. There are various types of statistical segmentation models, as described in Sec. 2. We adopt local statistical segmentation models [TSSC19] to calculate the probability map. For further information of probability maps, please refer to [PR12, TSSC19].

Region-based Constraints. Region-based constraints are constructed by pixel-wise posterior probabilities surrounding the projected contours of the 3D model. The baseline method [HZQ22] builds search lines around the projected contours to utilize these probabilities. A search line $\boldsymbol{l}_i$ is centered at the projection contour point of the 3D model, as shown in Fig. 2(a). The posterior probability that the j-th sample point $\boldsymbol{x}_{ij}$ on the i-th search line $\boldsymbol{l}_i$ belonging to the foreground and background are denoted by $P_f(\boldsymbol{x}_{ij})$ and $P_b(\boldsymbol{x}_{ij})$, respectively. The objective is to find the best implicit segmentation for the foreground and background, i.e. the precise projected contour that aligns with the actual object contour in the current frame. This is achieved by defining an energy function as follows:

$$E_R(\boldsymbol{\xi}) = \frac{1}{2} \sum_{\boldsymbol{x}_{ij} \in L} w(\boldsymbol{x}_{ij}) \psi(\boldsymbol{x}_{ij}) F^2(\boldsymbol{x}_{ij}, \boldsymbol{\xi}) \tag{6}$$

where $L = \{\boldsymbol{l}_1, \boldsymbol{l}_2, ..., \boldsymbol{l}_n\}$ is the set of search lines. $F(\boldsymbol{x}_{ij}, \boldsymbol{\xi})$ is the pre-defined loss function for pixel-wise posterior probabilities, $\psi(\boldsymbol{x}_{ij})$ is the corrected term, and $w(\boldsymbol{x}_{ij})$ is the weighted term. The level-set function is embedded into the $F(\boldsymbol{x}_{ij}, \boldsymbol{\xi})$ to represent the object contour, which is a commonly used technique in region-based methods [PR12, TSSC19]. For a detailed description, please refer to [HZQ22]. When a precise model is available, search lines established on the precise contour points are used to find the actual object contour in the frame for pose optimization.

4. Method

We first analyze the impact of rough models on region-based methods and next introduce the proposed strategy of contour *reshaping* and *remapping*. The key idea of *reshaping* is to utilize the probability map to refine the projected contour of rough models, as detailed in Sec. 4.1. Finally, we utilize the information within the object to add the constraints of color consistency.

© 2023 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

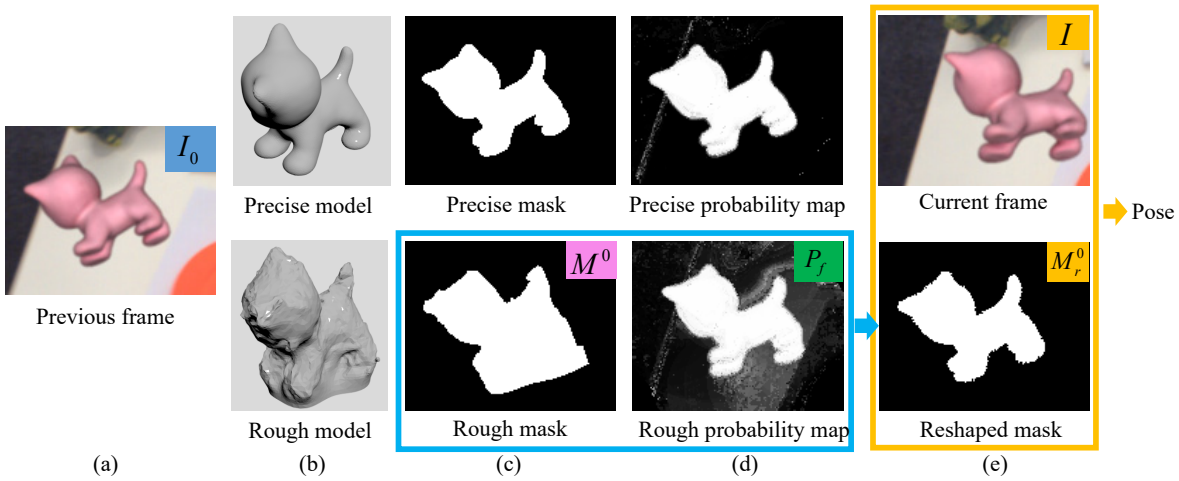

Figure 3: *Contour reshaping by the probability map. (a) The previous frame. (b-c) In the case of rough models, the projected rough mask is quite different from the precise mask, so their contours differ significantly. (d) The probability map calculated from the rough model and the previous frame contains more noise, but it can still reflect the object's contour well. (e) The probability map is used to reshape the rough mask, and a reshaped mask and contour are obtained, and the object pose is optimized by aligning the reshaped contour with the actual (implicit) object contour in the current frame.*

4.1. Region Constraints with Contour Reshaping

For a rough model, the projected contour may contain many wrong parts, as shown in Fig. 3(c), which is unfavorable for tracking. Take the baseline method [HZQ22] as an example, the center of the search line may be wrongly positioned, resulting in failure to find foreground contour points, as shown in Fig. 2(b). In fact, the real center position of the search line should be in the corrected contour, as shown in Fig. 2(c), and this can be achieved through the proposed contour *reshaping* strategy.

As a result of the implicit segmentation, the probability map can provide valuable information as an additional reference for the object contour. The probability map is calculated based on the color distribution of the foreground and background, and the division of the foreground and background is based on the projected mask of the 3D model. As shown in Fig. 3(d), when a precise model is available, the probability map has high accuracy and contains only a small amount of noise. When the model is rough, the probability map may contain more noise, but it can still reveal the precise object contour to some extent, and can be used to refine the rough contour.

The projected rough contour of the 3D model can be extracted from the projected mask, therefore, reshaping the mask is equivalent to reshaping the object contour. Assuming successful tracking of the previous frame I_0 and obtaining the initial pose $\boldsymbol{\xi}_0$, we project the 3D rough model using $\boldsymbol{\xi}_0$ to obtain the mask M^0. Then, the mask M^0 can be reshaped using the calculated foreground probability map P_f:

$$M_r^0 \leftarrow M^0 \cap H_e(P_f) \tag{7}$$

where $H_e(\cdot)$ represents the classical *Heaviside* function, which returns 1 when the input larger than 0, else returns 0. The contour can be reshaped effectively in this way, and then can be used to optimize the pose with the current frame, as shown in Fig. 3. Meanwhile, the search lines L in Eq. (6) will be re-established in the reshaped contour and change to $L' = \{\boldsymbol{l}'_1, \boldsymbol{l}'_2, .., \boldsymbol{l}'_n\}$.

4.2. Contour Updating by Remapping

After each iteration of the 3D tracking algorithm with the initial pose $\boldsymbol{\xi}_0$, an updated pose $\boldsymbol{\xi}^k$ is obtained, where k denotes the iteration number. Projecting the 3D model using the updated pose produces a new mask M^k. However, since the previous frame I_0 remains unchanged, the re-projected M^k will be mismatched with the probability map, leading to errors in contour correction. To address this issue, we obtain an updated reshaped mask M_r^k after iteration by remapping the initial reshaped mask M_r^0 in Eq. (7).

Before the pose iteration, we use the initial pose $\boldsymbol{\xi}_0$ to back-project 2D points set $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ in the reshaped mask M_r^0 to obtain the corresponding 3D points set $\{\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_n\}$ using Eq. (4). After each iteration, the pose is updated to $\boldsymbol{\xi}^k$, and $\{\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_n\}$ is re-projected to 2D points under pose $\boldsymbol{\xi}^k$ to obtain a new mask M_r^k:

$$M_r^k = \Upsilon\left(\{\boldsymbol{X}_1, \boldsymbol{X}_2, ..., \boldsymbol{X}_n\}, \boldsymbol{\xi}^k\right) \tag{8}$$

where Υ represents the projection process from 3D to 2D achieved through Eq. (3). This strategy allows the contour to maintain a good shape during iterations, which we call *remapping*. The process of *reshaping* and *remapping* is equivalent to making an implicit adjustment to the 3D model geometry via the 2D mask. During this procedure, small blank areas may appear in the mask, which we will fill with the closest pixel value.

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

4.3. Color Consistency Constraints

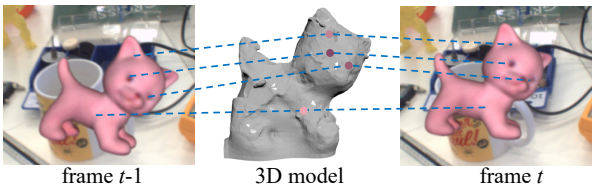

Figure 4: *The 3D points on the model exhibit highly similar colors in the projected points between adjacent frames. Therefore, the objects' colors are consistent between frames and can be used to provide constraints for pose estimation.*

When the projected object contour becomes unreliable, the internal information of the object becomes more crucial. We use internal information to apply additional constraints to pose optimization. The key idea is that the 3D model can be used as a guide to ensure color consistency, as the colors of corresponding points on an object should remain consistent across adjacent frames, as shown in Fig. 4. Although textureless objects may lack discernible feature points, their color still can create a distinct visual change in scenes, and importantly, often differs from the background. Therefore, the color consistency can provide constraints.

We sample points in the previous frame I_0 and find their corresponding points in the current frame I by color alignment. We eroded the rough mask by a distance of 10 pixels inwards to limit the sampling range. Even absence of texture, the color still change visually, so some Harris corner points [HS88] with low threshold can still be calculated. Noting that the role of Harris corners in this case is to locate the areas where the color change is most pronounced, rather than to obtain some points with the maximum gradient.

Then the Harris corner points along with the points within their corresponding square neighborhoods are selected as candidate points, as shown in Fig. 1. Since different neighborhoods may have overlapping regions where color changes sharply, points within such regions are sampled multiple times and will contribute more to the constraint force. The energy function is formulated as follows:

$$E_C(\boldsymbol{\xi}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{\boldsymbol{c}_{ij}\in\Omega_i}\left(I(\boldsymbol{c}_{ij}(\boldsymbol{\xi})) - I_0(\boldsymbol{c}_{ij}(\boldsymbol{\xi}_0))\right)^2 \tag{9}$$

where n is the number of Harris corners, and Ω_i is the neighborhood of the i-th corner (including the corner itself). $\boldsymbol{c}_{ij}(\boldsymbol{\xi}_0)$ is the j-th point in the neighborhood Ω_i in frame I_0, and $\boldsymbol{c}_{ij}(\boldsymbol{\xi})$ is the corresponding point to be found in the current frame I, which location is related to the pose $\boldsymbol{\xi}$. $I(\boldsymbol{c}_{ij}(\boldsymbol{\xi}))$ and $I_0(\boldsymbol{c}_{ij}(\boldsymbol{\xi}_0))$ represent the photometric values of points $\boldsymbol{c}_{ij}(\boldsymbol{\xi})$ and $\boldsymbol{c}_{ij}(\boldsymbol{\xi}_0)$, respectively.

5. Optimization

We first utilize a proposed pre-search strategy to perform 2D optimization and obtain a better initial pose, which strengthens the constraint of color consistency. Subsequently, we optimize the pose iteratively by incorporating both the region and color constrains.

5.1. Pre-Search Strategy

The direct method is based on the assumption of small displacement between consecutive frames, and we propose a simple but effective 2D pre-search strategy to better satisfying the assumption. The key to this strategy is to count the colors of foreground points in the previous frame I_0, and then using a sliding window search in the current frame I to find an area with the smallest color difference. The objective function can be formulated as follows:

$$\Delta\boldsymbol{x} = \arg\min_{\Delta\boldsymbol{x}} \sum_{\boldsymbol{x_i}\in\Omega_f} |(I(\boldsymbol{x}_i + \Delta\boldsymbol{x}) - I_0(\boldsymbol{x}_i))| \tag{10}$$

where Ω_f denotes the foreground region in I_0, and $\Delta\boldsymbol{x} = [\Delta x, \Delta y]^\top$ is the 2D displacement. By mapping $\Delta\boldsymbol{x}$ to 3D space, we can derive the pose increment $\Delta\boldsymbol{\xi}_x$:

$$\Delta\hat{\boldsymbol{\xi}}_{\boldsymbol{x}} = \ln\left(\begin{bmatrix} \boldsymbol{I} & \boldsymbol{K}^{-1}\Delta\tilde{\boldsymbol{x}} \\ \boldsymbol{0} & 1 \end{bmatrix}\right) \tag{11}$$

where $\Delta\tilde{\boldsymbol{x}} = [Z\Delta x, Z\Delta y, 0]^\top$, and Z is the depth value of the model center. Then the initial pose $\boldsymbol{\xi}$ in Eq. (9) is updated by $\Delta\boldsymbol{\xi}_x$:

$$\hat{\boldsymbol{\xi}} \leftarrow \ln(\exp(\Delta\hat{\boldsymbol{\xi}}_{\boldsymbol{x}})\exp(\hat{\boldsymbol{\xi}}_0)) \tag{12}$$

In this way, the distance between the points in previous and current frames is narrowed, which better satisfies the assumption and makes the optimization easier.

5.2. Pose Optimization

We combine region-based constraints (6) with color consistency constraints (9), and the energy function is defined as follows:

$$E(\boldsymbol{\xi}, L') = \lambda_1 E_R(\boldsymbol{\xi}, L') + \lambda_2 E_C(\boldsymbol{\xi}) \tag{13}$$

where λ_1 and λ_2 are balanced parameters, and due to the *reshaping* strategy, the search lines L in Eq. (6) are regenerated and becomes L'.

We first utilize the *pre-search* strategy described in Sec. 5.1 to acquire a better initial pose, and then perform Gauss-Newton method to optimize the energy function iteratively:

$$\Delta\boldsymbol{\xi} = -(\boldsymbol{H})^{-1}\boldsymbol{J}^\top \tag{14}$$

$$\hat{\boldsymbol{\xi}} \leftarrow \ln(\exp(\Delta\hat{\boldsymbol{\xi}})\exp(\hat{\boldsymbol{\xi}})) \tag{15}$$

where $\boldsymbol{J}$ and $\boldsymbol{H}$ are the Jacobian and Hessian matrix:

$$\boldsymbol{J} = \lambda_1\boldsymbol{J}_R + \lambda_2\boldsymbol{J}_C \tag{16}$$

$$\boldsymbol{H} = \lambda_1\boldsymbol{H}_R + \lambda_2\boldsymbol{H}_C \tag{17}$$

where $\boldsymbol{J}_R$ and $\boldsymbol{H}_R$ are the Jacobian and Hessian matrix computed from Eq. (6), more details please refer to [HZQ22]. $\boldsymbol{J}_C$ and $\boldsymbol{H}_C$ are the Jacobian and Hessian matrix computed from Eq. (9):

$$\boldsymbol{J}_C = \sum_{i=1}^{n}\sum_{\boldsymbol{c}_{ij}\in\Omega_i} h(\boldsymbol{c}_{ij})\left(\frac{\partial I(\boldsymbol{c}_{ij}(\boldsymbol{\xi}))}{\partial\Delta\boldsymbol{\xi}}\right)^\top \tag{18}$$

$$\boldsymbol{H}_C = \sum_{i=1}^{n}\sum_{\boldsymbol{c}_{ij}\in\Omega_i} \left(\frac{\partial I(\boldsymbol{c}_{ij}(\boldsymbol{\xi}))}{\partial\Delta\boldsymbol{\xi}}\right)^\top \left(\frac{\partial I(\boldsymbol{c}_{ij}(\boldsymbol{\xi}))}{\partial\Delta\boldsymbol{\xi}}\right) \tag{19}$$

© 2023 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

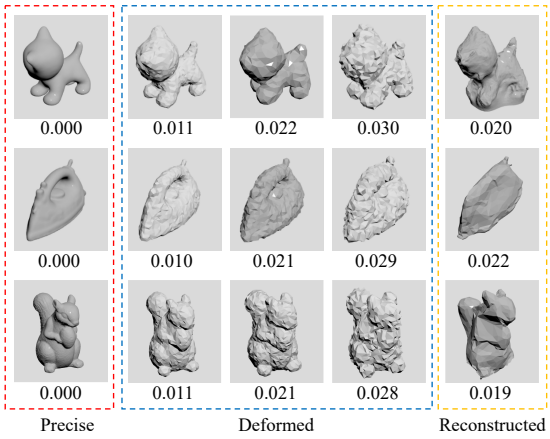

Figure 5: *Some precise models, deformed rough models and reconstructed rough models used in our experiments. The value below the model marks the roughness.*

Parameter	Value
λ_1 in energy function	0.20
λ_2 in energy function	$1/255^2 \times 0.52$
Number of points in remapping	All points in the mask
Number of pixels in erosion	10
Threshold of Harris corners	0.0001
Min distance of Harris corners	8
Max number of Harris corners	8000
Side length of Ω in Eq. (9)	11

Table 1: *The parameters in our experiments.*

Method	Iterations	Regular	Dynamic	Noisy	Occlusion	Avg. ↑
Precise Models (Roughness = 0)						
[TSSC19]	7	79.9	81.2	56.6	73.3	72.8
[HZQ22]	7	89.9	90.7	69.6	88.9	84.8
[SPS*20]	14	90.0	90.6	71.5	85.6	84.4
[SPS*22]	14	94.2	94.6	81.7	93.2	90.9
[TLZQ22]	30	95.2	**95.4**	83.2	**94.9**	92.2
Ours	7	**95.4**	94.9	**86.2**	93.2	**92.4**
Slightly Rough Models (Roughness ≈ 0.01)						
[TSSC19]	7	70.7	73.0	45.6	62.9	63.1
[HZQ22]	7	75.4	77.4	51.0	73.0	69.2
[SPS*20]	14	79.0	78.1	59.2	71.3	71.9
[SPS*22]	14	78.5	79.1	64.5	73.5	73.9
[TLZQ22]	30	77.2	77.5	61.8	72.4	72.2
Ours	7	**86.1**	**84.6**	**72.6**	**81.9**	**81.3**
Moderately Rough Models (Roughness ≈ 0.02)						
[TSSC19]	7	23.1	23.0	11.9	16.4	18.6
[HZQ22]	7	19.5	20.0	10.5	16.1	16.5
[SPS*20]	14	33.0	32.1	23.2	26.5	28.7
[SPS*22]	14	33.2	33.0	25.0	28.2	29.9
[TLZQ22]	30	24.7	24.9	18.5	21.2	22.3
Ours	7	**52.3**	**51.1**	**41.5**	**49.1**	**48.5**
Highly Rough Models (Roughness ≈ 0.03)						
[TSSC19]	7	0.3	4.7	1.9	3.8	2.7
[HZQ22]	7	9.4	9.2	10.9	8.0	9.4
[SPS*20]	14	17.1	17.2	13.1	15.7	15.8
[SPS*22]	14	17.7	18.1	14.8	17.0	16.9
[TLZQ22]	30	9.3	10.0	7.5	9.2	9.0
Ours	7	**44.7**	**43.8**	**32.6**	**40.5**	**40.4**

Table 2: *Tracking results on RBOT dataset with precise and deformed rough models. "Regular" etc. represent different scenarios of the dataset. The numerical value represents the tracking success rate, and the best result is highlighted in bold. Roughness is calculated by Hausdorff distance. The baseline method is [HZQ22].*

where $h(\boldsymbol{c}_{ij}) = I(\boldsymbol{c}_{ij}(\boldsymbol{\xi})) - I_0(\boldsymbol{c}_{ij}(\boldsymbol{\xi}_0))$ is the photometric differ, and:

$$\frac{\partial I\left(\boldsymbol{c}_{ij}(\boldsymbol{\xi})\right)}{\partial \boldsymbol{\Delta\xi}} = \frac{\partial I\left(\boldsymbol{c}_{ij}(\boldsymbol{\xi})\right)}{\partial \boldsymbol{c}_{ij}} \frac{\partial \boldsymbol{c}_{ij}}{\partial \boldsymbol{C}_{ij}} \frac{\partial \boldsymbol{C}_{ij}}{\partial \boldsymbol{\Delta\xi}} \tag{20}$$

where $\boldsymbol{C}_{ij} = [X, Y, Z]^\top$ is the 3D point corresponding to $\boldsymbol{c}_{ij}$ in the camera coordinate system. For detailed derivation of the formula, please refer to our supplementary materials.

The unit of the color consistency constraints in Eq. (9) is the photometric value, which is generally larger than the probability value in Eq. (6). Therefore, it is necessary to assign a lower weight to the color consistency constraints term to balance its importance.

We adopt the coarse-to-fine strategy for pose optimization, consisting of first iterating 4 times on the 1/4 size frame, then iterating 2 times on the 1/2 size frame, and finally iterating 1 time on the original frame.

6. Experiments

The proposed method can track both rough and precise models, and experiments are conducted on each of these model types. Our method can achieve a high frame rate of 30-40 FPS on a desktop computer equipped with an Intel i7-11700 CPU and a Nvidia GTX3080 GPU.

Table1 lists the parameters in our experiments. The balanced parameters λ_1 and λ_2 are determined empirically. The term $1/255^2$ in λ_2 is used for regularization, as the probability range in the region constraint is 0~1.0, while the color range in the color constraint is 0~255, and squaring is done for dimensional consistency. Harris corners are computed using OpenCV, and the required parameters for the implementation of OpenCV are listed, where a low threshold of 0.0001 is used to generate a sufficient number of points. The parameters of the search lines are the same as our baseline [HZQ22].

6.1. Dataset, Metrics and Models

Dataset. We evaluate the proposed method on the RBOT dataset [TSSC19], which consists of 18 objects of different sizes and shapes moving in four different scenes (Regular, Dynamic light, Noisy, and Occlusion). There are a total of 72 sequences, each with 1000 frames for testing, and the frame size is 640×512.

Metrics of Tracking and Roughness. Like most 3D object tracking algorithms [TSSC19, LZXQ21, ZZZ*20, HZQ22, SPS*20, SPS*22, TLZQ22] in recent years, we evaluated the tracking results using the popular 5 cm/5° metric, which means tracking success if the translation error is less than 5 cm and the rotation error is less than 5°. If the errors exceed the thresholds, the tracking is considered a failure and reinitialized with the ground truth pose. The

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

Method Roughness	driller 0.007	clown 0.010	bakingsoda 0.010	phone 0.013	squirrel 0.019	cat 0.020	ape 0.021	iron 0.022	duck 0.023	Avg.↑ 0.016
Regular										
[TSSC19]	72.9	50.4	16.7	38.6	31.1	16.1	30.2	7.4	29.2	32.5
[HZQ22]	76.7	58.6	15.3	42.6	32.3	32.6	36.5	7.6	33.9	37.3
[SPS*20]	75.6	53.7	28.2	44.5	37.8	30.1	42.7	26.7	30.5	41.1
[SPS*22]	75.5	60.3	26.1	50.1	51.4	36.4	**45.5**	23.2	45.0	45.9
[TLZQ22]	75.8	64.3	16.7	51.1	38.2	32.9	50.0	13.8	41.9	42.7
Ours	**82.1**	**74.2**	**35.9**	**54.1**	**53.5**	**50.2**	43.6	**40.9**	**45.8**	**53.4**
Dynamic light										
[TSSC19]	71.9	50.4	16.8	41.1	35.5	11.9	29.8	7.5	28.4	32.6
[HZQ22]	74.5	59.2	14.3	42.9	35.5	29.5	37.4	9.5	35.5	37.6
[SPS*20]	71.9	53.7	25.9	40.4	36.2	27.7	40.1	25.1	30.2	39.0
[SPS*22]	74.0	60.2	25.6	51.4	50.8	36.3	**47.3**	23.1	45.2	46.0
[TLZQ22]	74.9	61.7	16.0	49.9	37.4	32.3	48.7	14.3	38.8	41.6
Ours	**79.7**	**74.7**	**27.1**	**53.7**	**53.5**	**50.1**	42.8	**38.5**	**46.0**	**51.8**
Noisy										
[TSSC19]	43.8	28.1	20.0	20.3	29.2	9.4	29.5	6.0	27.6	23.8
[HZQ22]	57.3	33.5	15.5	23.3	25.6	25.7	34.7	7.4	33.8	28.5
[SPS*20]	63.9	39.4	25.4	31.6	28.9	21.4	30.7	22.9	26.2	32.3
[SPS*22]	65.5	47.8	22.6	**45.0**	41.0	28.7	42.3	19.6	38.3	39.0
[TLZQ22]	**66.3**	46.0	15.6	42.6	33.1	25.3	44.4	13.5	32.8	35.5
Ours	65.5	**61.8**	**27.0**	40.5	**47.9**	**44.4**	**44.6**	**31.5**	**45.2**	**45.4**
Occlusion										
[TSSC19]	60.3	43.2	16.4	33.6	25.1	8.7	28.9	6.8	23.7	27.4
[HZQ22]	72.0	54.0	14.7	38.8	24.4	26.5	33.6	8.6	31.2	33.8
[SPS*20]	68.6	48.3	20.3	38.8	33.0	25.4	35.5	22.5	25.7	35.3
[SPS*22]	69.3	53.8	20.3	47.0	41.6	32.3	43.6	22.1	37.0	40.8
[TLZQ22]	69.4	55.8	19.4	45.1	33.8	29.7	**46.1**	14.3	32.2	38.4
Ours	**75.9**	**69.8**	**29.3**	**50.1**	**52.1**	**47.3**	42.9	**37.8**	**43.8**	**49.9**

Table 3: *Tracking results on RBOT dataset with reconstructed rough models with a 5 cm/5° metric in four scenarios (Regular, Dynamic light, Noisy, Occlusion). The numbers above each model represent its roughness calculated by the Hausdorff distance. The best result is highlighted in bold. The baseline method is [HZQ22].*

ratio of the number of successful frames is calculated as the success rate. We calculate the Hausdorff distance between the rough and precise model, which we then normalize by dividing it by the diagonal length of the BBX of the rough model. This normalized value is used to quantify the roughness of the model, with higher values indicating greater roughness.

Models. The precise models are provided by the RBOT dataset. We deform the precise models using techniques such as fractal displacement, random vertex displacement, smoothing, etc., to obtain corresponding rough models. Additionally, we attach texture patches to some precise objects in RBOT dataset and use the COLMAP [SF16, SZFP16] algorithm to reconstruct them. Meshlab software is employed to align the reconstructed rough models with the precise models.

We categorize the models into four types based on their level of roughness. The first type is the *precise* models, which have a roughness of 0. The second type is the *slightly rough* models, with a roughness of about 0.01, which can be considered as an approximate version of the precise models. The third type is the *moderately rough* models, with a roughness of about 0.02, and the fourth type is the *highly rough* models, with a roughness of about 0.03. Figure 5 shows some of the models, and all models can be found in the supplementary materials.

6.2. Results and Discussion

The results on the precise and deformed rough models are shown in Tab. 2 and the experimental results on the reconstructed rough models are shown in Tab. 3. In cases where the model is moderately rough or highly rough, the *reshaping* strategy and color consistency constraints are used to optimize the pose. When the model is precise or approximately precise (slightly rough), the projected contour is reliable enough, so *reshaping* is not adopted in this case.

Precise and Deformed Rough Models As shown in Tab. 2, our method achieves state-of-the-art results. As the roughness of the model increases, our method exhibits a more significant improvement in tracking success rate compared to other methods. The success rate under the four roughnesses are 0.2%, 7.4%, 18.6% and 23.5% higher than previous state-of-the-art methods, respectively. In the case of highly rough models, [SPS*22] achieves a tracking success rate of only 16.9%, while our method a significantly higher success rate of 40.4%. It is worth noting that we only perform 7 iterations, whereas [TLZQ22] performs 30 iterations and [SPS*22] perform 14 iterations.

In dynamic lighting scenarios, our method exhibits almost no decrease in tracking success rate. This is because the frames are continuous during tracking, so lighting does not change drastically between adjacent frames. And in Sec. 6.3, our ablation experiment shows that the hypothesis of small displacements has a greater impact compared to the assumption of light invariance. In noisy scenarios, the quality of the probability map may decrease, but the accuracy of our method still far exceeds that of other methods. In occlusion scenarios, the accuracy has a slight decrease, indicating that the proposed method has good robustness to occlusion.

Reconstructed Rough Models For the reconstructed rough models, the roughness ranges from 0.007 to 0.023, and the average roughness is 0.016. As shown in Tab. 3, our method outperformed

© 2023 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

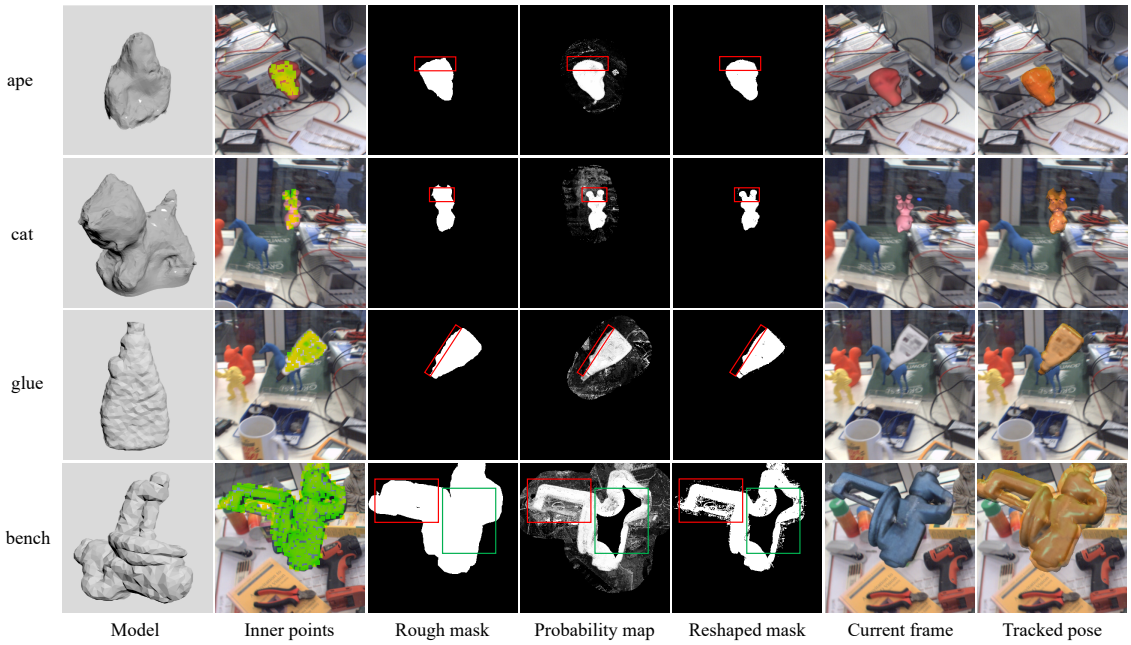

Figure 6: *Some intermediate results in tracking. "ape" and "cat" are obtained by reconstruction. "glue" and "bench" are obtained by deform. The sampling regions of the inner points are marked with green squares in the previous frame (second column). A reshaped mask can be obtained by reshaping the rough mask with a probability map. The reshaped mask is more refined and accurate (marked with a red rectangle). The last column visualizes the rendering of the 3D model in the current frame under the tracked pose.*

the previous state-of-the-art method by 7.2% accuracy in average. And on most models, we have achieved the highest accuracy.

Intermediate Results Figure 6 displays some intermediate results during the tracking process of rough models. The proposed *reshaping* strategy can effectively refine the rough mask, leading to the generation of a more accurate and reliable contour of the mask, thus establishing a more robust contour constraint, as indicated by the red rectangle. Additionally, the color consistency of multiple regions can provide extra constraints, which further enhance the tracking performance of our method. As described in Sec. 4.3, we set the selection threshold for Harris corner points low enough, resulting in numerous candidate points even on textureless surfaces. These candidate points, along with the points in their neighborhoods, are marked with green squares.

In some cases, such as the "bench" shown in Fig. 6, there are holes (indicated by green retangle) in the probability map. This is due to the statistical segmentation models used to calculate the probability map being locally built on the object contour points. Therefore, when the object occupies a large area in the frame, the object center may not be covered by the statistical segmentation models. Region-based constraints are primarily built around contours, so holes inside the probability map have only a tiny impact. However, for the color consistency constraints, the hole parts does not generate candidate points, reducing the constraint force. Therefore, we sample points on the rough mask without *reshaping*, and although some external points may be sampled in this case, we have found in experiments that this is a better approach. More results can be found in our supplementary materials.

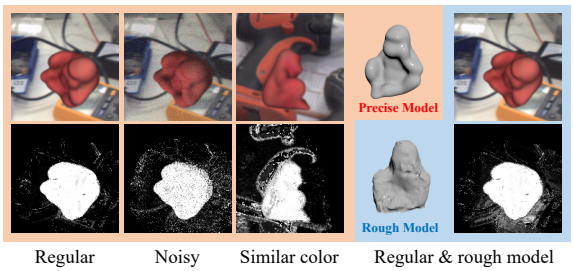

Figure 7: *Compared to regular scenes, noise, similar colors, and rough models can decrease the quality of probability maps.*

Sensitivity of the Probability Map A probability map is calculated by the color distribution of the foreground and background. As a result, if the foreground colors are similar to the background colors or if there is random noise in the image, the quality of the probability maps would decrease. For rough models, the projected mask of the 3D model may not align well with the object, resulting in a small portion of erroneous colors sampling near the boundary between foreground and background, which can also decrease the quality of the probability maps. These cases are as shown in Fig. 7. Unreliable probability maps may lead to a decrease in tracking accuracy, as shown in Tab. 2, compared to regular scenes, the

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

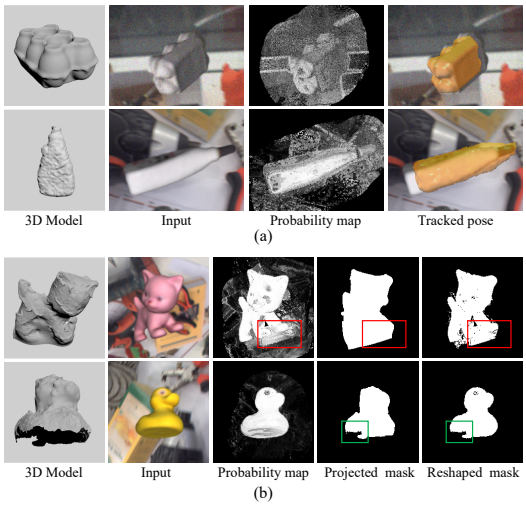

Figure 8: *Failure cases. (a) Failures of tracking both for precise and rough models due to unreliable probability maps. (b) Failures of reshaping due to an unreliable probability (red rectangle) or a model with large missing parts (green rectangle).*

noise and the rough models decrease the tracking accuracy to some extent. In some cases, unreliable probability maps may lead to failure, which we will discuss in the following sections. Failures due to unreliable probability maps are inevitable, and all region-based methods would face this issue.

Detection and Segmentation The probabilistic map plays a role in implicit segmentation. Therefore, a natural thought is to use learning-based detection or segmentation methods to obtain an explicit foreground mask. However, as a real-time AR application that can be applied to any object, using detection or segmentation methods may lead to real-time and generalization problems. Foundation models, like SAM(Segment Anything Model) [KMR*23], have excellent generalization and segmentation capabilities but suffer from poor real-time performance. In our equipment, the default SAM model takes around 400ms, and the *ViT-B* SAM model takes around 100ms to segment one frame, which is hard to meet real-time needs. As a comparison, computing a probability map only takes around 2ms. Compared to SAM, lightweight methods may achieve faster speeds but suffer from training and generalization problems. In contrast, our method can track various objects without suffering the generalization problems and achieves 30~40FPS.

Failure Cases Generally, unreliable probability maps may lead to tracking failures for both precise and rough models, as shown in Fig. 8(a). Additionally, in the case of a rough model, an unreliable probability map may undermine the effectiveness of the *reshaping* strategy. Moreover, when a large part of the model is missing, it is hard for the *reshaping* strategy to obtain a precise and complete contour, as shown in Fig. 8(b).

6.3. Ablation Study

To explore the role played by the proposed strategies, we conduct experiments on the deformed rough models with a roughness about 0.02. The experimental results are shown in Tab. 4. The *reshaping* strategy significantly improves tracking accuracy (16.5% v.s. 43.8%), which shows the strategy's effectiveness and the importance of contours in tracking. The color consistency constraints also lead to a large improvement (16.5% v.s. 29.4%), showing that the internal information of the rough model also has great importance. The best tracking results are produced combined with the two strategies (16.5% v.s. 48.5%), proving the effectiveness of the proposed strategies

Baseline	Reshaping	Color consistency	Success Rate(%)
✓			16.5
✓	✓		43.8
✓		✓	29.4
✓	✓	✓	**48.5**

Table 4: *Ablation study on RBOT dataset with deformed rough models (Roughness ≈ 0.02). The baseline is [HZQ22].*

Baseline	Pre-search	CC-	Success Rate(%)
✓			84.8
✓	✓		87.0
✓		✓	84.7
✓	✓	✓	**92.4**

Table 5: *Ablation study of the pre-search strategy on RBOT dataset with precise models. "CC-" denotes color consistency constraints without pre-search. The baseline is [HZQ22].*

We strive to satisfy the assumption of small displacement between frames, and therefore propose a pre-search strategy. We also conduct ablation experiments on the precise models to explore this strategy. As shown in the Tab. 5, when only using the pre-search strategy, the tracking accuracy has a small improvement; when only using color consistency constraints without pre-search, there is little change in tracking accuracy; when both strategies are used simultaneously, there is a considerable improvement in tracking accuracy. This shows that the color consistency constraints yield a considerable improvement when the assumption of small displacement between frames is better satisfied.

6.4. Application

We applied the proposed tracking algorithm to a Buddha statue artifact in a museum, which was difficult to reconstruct due to its placement against a wall and being located inside a glass case. In addition, we have also reconstructed a "cat" model for tracking in common AR scenes. While more accurate reconstruction can be achieved through devices such as scanners, in cases where quick augmented reality (AR) interaction is desired and equipment is limited, reconstruction is often rough. The scenes and reconstructed models and tracking results are shown in Fig. 9. For more detailed results, please refer to our supplementary materials.

6.5. Limitations

Although our method achieves good performance for both rough and precise models, there are still some shortcomings that need to be addressed. Firstly, tracking objects relies on the probability maps, an unreliable probability may result in tracking failures, and

© 2023 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

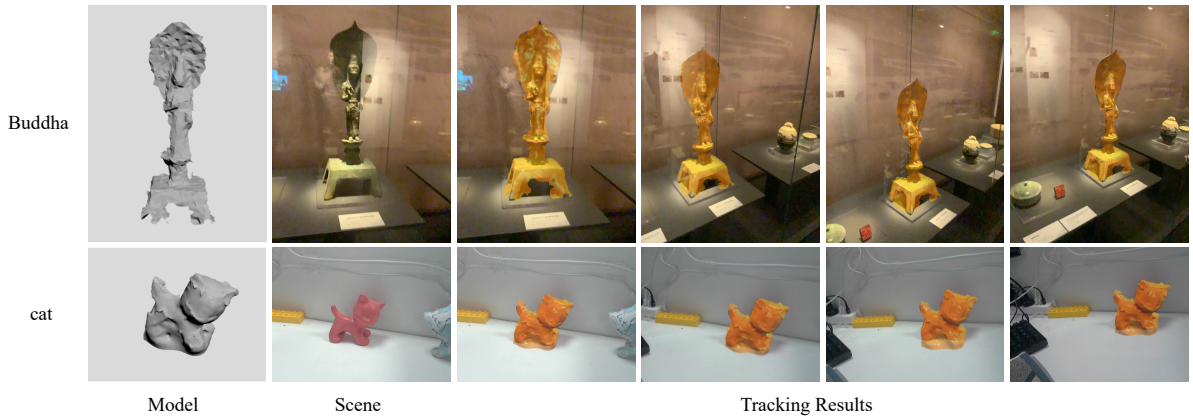

Figure 9: *Applications of our method. Buddha is an artifact in the museum, while "cat" is a model used for AR in daily life.*

if the rough model has significant missing parts, the *reshaping* strategy may also struggle to get a precise and complete contour. Secondly, the *reshaping* strategy is designed for the rough model to make the projected rough contour become precise. If the model is precise, the projected contour is relatively accurate, so the *reshaping* is unnecessary for precise models. Therefore, we need to empirically evaluate the roughness of the object before tracking and deciding whether to enable the *reshaping* module. Finally, we do not optimize the topology of the 3D model during tracking, which is our future direction.

7. Conclusions

For rough models tracking, projected contours may no longer provide valid constraints. To address this issue, we propose using the probability map to reshape the contour, and we combine this with a multi-region sampling strategy to emphasize the inner features. To utilize color consistency constraints for inner points, the assumption of the small displacement between frames needs to be satisfied. To address this problem, we have proposed a fast 2D space search strategy that is an independent module that can be applied to other 3D tracking methods. Experimental results demonstrate that the proposed method can track both rough and precise models effectively, and the rougher the model, the greater the improvement compared to other tracking methods. In addition, our method can be applied to any 3D model without requiring parameter adjustments and real-time, which makes it highly generalizable. In future work, our goal is to explore the use of deep learning networks, such as SAM [KMR*23], to predict object contours to supplement the shortcomings of traditional probability models, and to explore how to achieve real-time results when using such networks.

8. Acknowledgments

This work is partially supported by the National Key R&D Program of China under grant (No. 2022YFB3303203), NSF of China (No. 62172260, 62202425), and also sponsored by SenseTime.

References

[AK08] ALEXANDER KIRILLOV J.: *An Introduction to Lie Groups and Lie Algebras*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2008. 3

[CL14] CRIVELLARO A., LEPETIT V.: Robust 3D tracking with descriptor fields. In *Proc. CVPR '27* (2014), pp. 3414–3421. doi:10.1109/CVPR.2014.436. 2

[CRV*18] CRIVELLARO A., RAD M., VERDIE Y., YI K. M., FUA P., LEPETIT V.: Robust 3D object tracking from monocular images using stable parts. *IEEE TPAMI 40*, 6 (June 2018), 1465–1479. doi:10.1109/TPAMI.2017.2708711. 2

[DC02] DRUMMOND T., CIPOLLA R.: Real-time visual tracking of complex structures. *IEEE TPAMI 24*, 7 (July 2002), 932–946. doi:10.1109/TPAMI.2002.1017620. 2

[HH16] HEXNER J., HAGEGE R. R.: 2D-3D pose estimation of heterogeneous objects using a region based approach. *IJCV 118*, 1 (May 2016), 95–112. doi:https://doi.org/10.1007/s11263-015-0873-2. 2

[HS88] HARRIS C. G., STEPHENS M.: A combined corner and edge detector. In *Proc. Alvey Vision Conference '4* (September 1988), pp. 147–151. doi:10.5244/c.2.23. 5

[HS90] HARRIS C., STENNETT C.: Rapid - a video rate object tracker. In *Proc. BMVC '1* (September 1990), pp. 1–6. doi:10.5244/C.4.15. 2

[HSW*22] HE X., SUN J., WANG Y., HUANG D., BAO H., ZHOU X.: Onepose++: Keypoint-free one-shot object pose estimation without CAD models. In *Proc. NeurIPS '36* (November 2022). doi:10.48550/arXiv.2301.07673. 2

[HZQ22] HUANG H., ZHONG F., QIN X.: Pixel-wise weighted region-based 3D object tracking using contour constraints. *IEEE TVCG 28*, 12 (December 2022), 4319–4331. doi:10.1109/TVCG.2021.3085197. 2, 3, 4, 5, 6, 7, 9

[HZSQ20] HUANG H., ZHONG F., SUN Y., QIN X.: An occlusion-aware edge-based method for monocular 3D object tracking using edge confidence. *Computer Graphics Forum 39*, 7 (November 2020), 399–409. doi:10.1111/cgf.14154. 2

[IP15] IMPERIOLI M., PRETTO A.: D2CO: fast and robust registration of 3D textureless objects using the directional chamfer distance. In *Proc. ICVS '10* (July 2015), vol. 9163, pp. 316–328. doi:10.1007/978-3-319-20904-3_29. 2

[KMR*23] KIRILLOV A., MINTUN E., RAVI N., MAO H., ROLLAND C., GUSTAFSON L., XIAO T., WHITEHEAD S., BERG A. C.,

© 2023 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

LO W., DOLLÁR P., GIRSHICK R. B.: Segment anything. *CoRR abs/2304.02643* (2023). doi:10.48550/arXiv.2304.02643. 9, 10

[LF05] LEPETIT V., FUA P.: *Monocular Model-Based 3D Tracking of Rigid Objects: A Survey*, vol. 1. Now Foundations and Trends, 2005. doi:10.1561/0600000001. 1

[LSFK10] LIMA J. P., SIMÕES F., FIGUEIREDO L. S., KELNER J.: Model based markerless 3d tracking applied to augmented reality. *SBC Journal on Interactive Systems 1* (November 2010), 2–15. doi:10.5753/jis.2010.560. 1

[LSZQ21] LI J., SONG X., ZHONG F., QIN X.: Fast 3D texture-less object tracking with geometric contour and local region. *Computers and Graphics 97* (June 2021), 225–235. doi:10.1016/j.cag.2021.04.012. 2

[LWP*22] LIU Y., WEN Y., PENG S., LIN C., LONG X., KOMURA T., WANG W.: Gen6d: Generalizable model-free 6-dof object pose estimation from RGB images. In *Proc. ECCV '17* (2022), vol. 13692, pp. 298–315. doi:10.1007/978-3-031-19824-3_18. 2

[LZXQ21] LI J., ZHONG F., XU S., QIN X.: 3D object tracking with adaptively weighted local bundles. *JCST 36*, 3 (May 2021), 555–571. doi:10.1007/s11390-021-1272-5. 2, 6

[MUS16] MARCHAND E., UCHIYAMA H., SPINDLER F.: Pose estimation for augmented reality: A hands-on survey. *IEEE TVCG 22*, 12 (December 2016), 2633–2651. doi:10.1109/TVCG.2015.2513408. 1

[NHX*22] NGUYEN V. N., HU Y., XIAO Y., SALZMANN M., LEPETIT V.: Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions. In *Proc. CVPR '35* (June 2022), pp. 6761–6770. doi:10.1109/CVPR52688.2022.00665. 2

[PR12] PRISACARIU V. A., REID I. D.: PWP3D: Real-time segmentation and tracking of 3D objects. *IJCV 98*, 3 (January 2012), 335–354. doi:10.1007/s11263-011-0514-3. 2, 3

[SF16] SCHÖNBERGER J. L., FRAHM J.: Structure-from-motion revisited. In *Proc. IEEE/CVF CVPR '29* (June 2016), pp. 4104–4113. doi:10.1109/CVPR.2016.445. 2, 7

[SPP*14] SEO B., PARK H., PARK J., HINTERSTOISSER S., ILIC S.: Optimal local searching for fast and robust textureless 3D object tracking in highly cluttered backgrounds. *IEEE TVCG 20*, 1 (January 2014), 99–110. doi:10.1109/TVCG.2013.94. 2

[SPS*20] STOIBER M., PFANNE M., STROBL K. H., TRIEBEL R., ALBU-SCHÄFFER A.: A sparse gaussian approach to region-based 6DoF object tracking. In *Proc. ACCV '15* (November 2020), vol. 12623, pp. 666–682. doi:10.1007/978-3-030-69532-3_40. 2, 6, 7

[SPS*22] STOIBER M., PFANNE M., STROBL K. H., TRIEBEL R., ALBU-SCHÄFFER A.: SRT3D: A sparse region-based 3D object tracking approach for the real world. *IJCV 130*, 4 (February 2022), 1008–1030. doi:10.1007/s11263-022-01579-8. 1, 2, 6, 7

[SW16] SEO B., WUEST H.: A direct method for robust model-based 3D object tracking from a monocular RGB image. In *Proc. ECCV Workshops (3) '14* (November 2016), vol. 9915, pp. 551–562. doi:10.1007/978-3-319-49409-8_48. 2

[SWZ*22] SUN J., WANG Z., ZHANG S., HE X., ZHAO H., ZHANG G., ZHOU X.: Onepose: One-shot object pose estimation without cad models. In *Proc. IEEE/CVF CVPR '35* (June 2022), pp. 6825–6834. doi:10.48550/arXiv.2205.12257. 2

[SZFP16] SCHÖNBERGER J. L., ZHENG E., FRAHM J., POLLEFEYS M.: Pixelwise view selection for unstructured multi-view stereo. In *Proc. ECCV '14* (September 2016), vol. 9907, pp. 501–518. doi:10.1007/978-3-319-46487-9_31. 2, 7

[TLZQ22] TIAN X., LIN X., ZHONG F., QIN X.: Large-displacement 3D object tracking with hybrid non-local optimization. In *Proc. ECCV '17* (October 2022), pp. 627–643. doi:10.1007/978-3-031-20047-2_36. 1, 2, 6, 7

[TSS16] TJADEN H., SCHWANECKE U., SCHÖMER E.: Real-time monocular segmentation and pose tracking of multiple objects. In *Proc. ECCV '14* (September 2016), pp. 423–438. doi:10.1007/978-3-319-46493-0_26. 2

[TSS17] TJADEN H., SCHWANECKE U., SCHÖMER E.: Real-time monocular pose estimation of 3D objects using temporally consistent local color histograms. In *Proc. ICCV '16* (October 2017), pp. 124–132. doi:10.1109/ICCV.2017.23. 2

[TSSC19] TJADEN H., SCHWANECKE U., SCHÖMER E., CREMERS D.: A region-based gauss-newton approach to real-time monocular multiple object tracking. *IEEE TPAMI 41*, 8 (December 2019), 1797–1812. doi:10.1109/TPAMI.2018.2884990. 1, 2, 3, 6, 7

[Var13] VARADARAJAN V. S.: *Lie groups, Lie algebras, and their representations*. Springer, 2013. doi:10.1007/978-1-4612-1126-6. 3

[WB21] WEN B., BEKRIS K. E.: Bundletrack: 6D pose tracking for novel objects without instance or category-level 3D models. In *Proc. IEEE IROS' 31* (September 2021), pp. 8067–8074. doi:10.1109/IROS51168.2021.9635991. 2

[WLP*22] WEN Y., LI X., PAN H., YANG L., WANG Z., KOMURA T., WANG W.: Disp6d: Disentangled implicit shape and pose learning for scalable 6d pose estimation. *Proc. ECCV '17* (October 2022), 404–421. doi:10.1007/978-3-031-20077-9_24. 2

[WMX*20] WANG C., MARTÍN-MARTÍN R., XU D., LV J., LU C., FEI-FEI L., SAVARESE S., ZHU Y.: 6-PACK: Category-level 6d pose tracker with anchor-based keypoints. In *Proc. IEEE ICRA '37* (August 2020), pp. 10059–10066. doi:10.1109/ICRA40945.2020.9196679. 2

[WTB*23] WEN B., TREMBLAY J., BLUKIS V., TYREE S., MÜLLER T., EVANS A., FOX D., KAUTZ J., BIRCHFIELD S.: Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. *Proc. IEEE/CVF CVPR '36* (2023). doi:10.48550/arXiv.2303.14158. 2

[WWZ*21] WENG Y., WANG H., ZHOU Q., QIN Y., DUAN Y., FAN Q., CHEN B., SU H., GUIBAS L. J.: CAPTRA: Category-level pose tracking for rigid and articulated objects from point clouds. In *Proc. IEEE/CVF ICCV '18* (October 2021), pp. 13189–13198. doi:10.1109/ICCV48922.2021.01296. 2

[WZQ17] WANG B., ZHONG F., QIN X.: Pose optimization in edge distance field for textureless 3D object tracking. In *Proc. of CGI '34'* (January 2017), no. 32, pp. 1–6. doi:10.1145/3095140.3095172. 1, 2

[WZQ19] WANG B., ZHONG F., QIN X.: Robust edge-based 3D object tracking with direction-based pose validation. *Multimedia Tools and Applications 78*, 9 (May 2019), 12307–12331. doi:10.1007/s11042-018-6727-5. 2

[YYH*22] YISHENG H., YAO W., HAOQIANG F., QIFENG C., JIAN S.: Fs6d: Few-shot 6d pose estimation of novel objects. *Proc. CVPR '35* (June 2022), 6804–6814. doi:10.1109/CVPR52688.2022.00669. 2

[ZWS*14] ZHONG S., WANG L., SUI W., YU WU H., PAN C.: 3D object tracking via boundary constrained region-based model. In *Proc. IEEE ICIP '21* (October 2014), pp. 486–490. doi:10.1109/ICIP.2014.7025097. 2

[ZZ19] ZHONG L., ZHANG L.: A robust monocular 3D object tracking method combining statistical and photometric constraints. *IJCV 127*, 8 (August 2019), 973–992. doi:10.1007/s11263-018-1119-x. 2

[ZZZ*20] ZHONG L., ZHAO X., ZHANG Y., ZHANG S., ZHANG L.: Occlusion-aware region-based 3D pose tracking of objects with temporally consistent polar-based local partitioning. *IEEE TIP 29* (February 2020), 5065–5078. doi:10.1109/TIP.2020.2973512. 2, 6

© 2023 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.