Reflective texture-less object registration using multiple edge features for augmented reality assembly

Zaixing He
Jindong Zhao
Xinyue Zhao (✉ zhaoxinyue@zju.edu.cn)
Zhejiang University

Wuxi Feng
Quanyou Wang
Huilong Jiang

Research Article

Keywords: virtual assembly, object registration, multiple edge features, template sampling

Posted Date: May 5th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1578869/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License
Abstract

Virtual assembly technology has developed rapidly in recent years. As one of its key technologies, object registration has been widely studied, whose main task is to register the target object. However, the existing methods cannot effectively register reflective texture-less objects such as metal parts. In this paper, an object registration method based on multiple edge features is proposed, which includes a sparse uniform template sampling method and a multiple edge feature matching method. At the same time, we also used two oil pumps to carry out virtual assembly experiments and verify the accuracy of object registration. The experimental results show that this method can achieve high accuracy which is in line with the requirements of the virtual assembly algorithm in the industry.

1 Introduction

Augmented reality is a technology that renders virtual objects in the real environment and integrates virtual information with the real world. In recent years, augmented reality technology has developed rapidly, and there are a lot of researches on the implementation and application of augmented reality [1–3]. Among them, the application of augmented reality technology in the manufacturing industry is very extensive, where the most extensive application is virtual assembly technology [4–6]. The virtual assembly technology based on augmented reality mainly studies how to assemble models of parts on a physical part or part of the assembly virtually, so as to complete the assembly of a virtual part with real parts.

The aim of augmented reality assembly is to align the added information which is the virtual object precisely with the real world. Therefore, to achieve the combination of the real world and the virtual world, augmented reality assembly needs to make the position of the camera relative to the object in the real image the same as the position of the camera relative to the object in the virtual image exactly. Therefore, object registration is the main step in the whole process of augmented reality assembly.

In object registration, the main task is to obtain the pose information of the object, which is the position relationship with the viewpoint. There are many methods for object registration. According to the image type used, object registration methods can be divided into methods based on three-dimensional information (3D point cloud) and methods based on two-dimensional information (2D image). In recent years, in both 2D and 3D image-based object registration, deep learning becomes a useful tool [7, 8]. Methods based on deep learning could achieve high robustness. However, they need lots of templates to train the model which requires a difficult data acquisition process and many calculation resources.

The method based on 3D information is to obtain the depth information of the image. Because it utilizes more information, it could achieve more accurate results than methods based on 2D information. Park [9] et al. proposed a novel multi-task Template Matching (MTTM) framework to find the Template image of the nearest target object. Iterative closest point (ICP) [10, 11] algorithm is a classical algorithm, which can calculate the pose relationship between two coordinates according to two sets of point clouds. Konishi
et al. combined PCOF-MOD (Multimodal PCOF) and balanced Pose Tree (BPT) and optimal memory rearrangement into object registration to optimize data storage structure and search speed. Zhang et al. transformed sliding windows to scale-invariant RGB-D patches and applied a hash voting-based hypothesis generation scheme to register reflective texture-less objects. Pan et al. [14] segmented the object in the point cloud coarsely and registered the object in the gray image precisely with a view-based matching method. He et al. [15] proposed a deep Hough voting network to detect 3D keypoints of objects and then estimate the 6D pose parameters within a least-squares fitting manner.

For ordinary objects, many existing 3D image-based object registration methods can achieve high performance. As for the metal parts that are common in manufacturing, they are easy to fail due to the reflective surface of the metal parts. It is difficult to obtain accurate depth information from their surfaces, which is the basis for the success of 3D image-based methods. Therefore, it is more suitable to use 2D images for reflective object registration.

Methods based on 2D information utilize plane information such as geometric information, texture information, and so on, to register objects. Because of using less data, the time cost and the resource cost of methods based on 2D information are lower. Scale-invariant feature transform (SIFT) feature [16] and speeded up robust features, SURF [17] are early and classic features of attitude estimation based on object texture. Konishi [18] proposed a kind of Perspective Cumulated Orientation Feature (PCOF), which is based on the Orientation histogram extracted from the randomly generated 2D projection image. Khosla proposed the Fine Pose Parts-based Model (FPM) [19], which is used to locate objects in the image and register objects by using the given CAD model. Pan et al. [20] utilized multiple appearance features including the color, size, and aspect ratio to distinguish objects from the cluttered environment and measure the 6D pose. BB8 [21] used CNN to predict the two-dimensional projection of the eight vertices of the three-dimensional boundary box of the object. Crivellaro [22] trained CNN to predict the 6D posture of an object when it was only partially visible. PVNet [23] predicted the direction from each pixel to each key point, so the spatial probability distribution of two-dimensional key points can be obtained just like RANSAC. Zhang et al. [24] detected the object in the RGB image via a 2D bounding box and then registered the object in the edge image. Posenet [25] is a monocular 6D pose recognition system, which is used to train CNN to regress for regression 6D pose. In this network, the 6D pose estimation problem transforms into a regression problem. A single RGB image is input and the 6D pose of the camera is output by an end-to-end method.

For reflective texture-less objects, the performance of existing 2D image-based methods is also not high, since it is difficult to obtain reliable feature points of 2D images of reflective texture-less objects. Many current methods based on 2D images use image feature points for object registration, but the surface of metal parts is smooth and without texture, so there are almost no reliable image feature points to extract. Moreover, due to its reflective properties, some fake textures may even be generated. Many researches have been made to improve the robustness and accuracy, where deep learning technology plays an important role. While progress has been made, the problem is far from resolved.
In order to solve this problem, we propose a new object registration method based on multiple edge features which will not be seriously affected by reflective and texture-less surfaces. Besides, memorized sampling method will be introduced as a preparation step. We apply the method to the augmented reality assembly of mechanical metal parts. The surface of mechanical metal parts lacks stable textures and it sometimes even generates some fake textures due to its reflective characteristic. In this object registration method, we use more reliable multiple edge features and match them between real images and template images. Moreover, edge feature points could be calculated by matching multiple edge features. Then we could obtain matched edge feature point pairs according to the matching relationship of multiple edge features and the object registration can be completed by EPnP [26].

2 Method

The process of the proposed method can be divided into the following three parts (Fig. 1): Firstly, the CAD model is uniformly sampled to generate a large number of template images, and the corresponding relationship between 2D points on template images and 3D points on the CAD model is saved by vertex memorization method. Next, we select a template that best matches the real image, and then extract and match multiple edge features with edge feature points on the real image and the template image, and later register the object by EPnP. Finally, we complete the augmented reality according to the registration result which includes the transformation between two registration results and the rendering of the CAD model. The implementation of this method is described in detail below:

2.1 Sparse templates sampling and generation

In the process of augmented reality assemble, object registration is a primary and necessary step. The main step of object registration is matching templates and real images. Therefore, a certain number of template images are needed. To improve the utilization rate of the template and avoid the excessive number of templates in a certain perspective, sampling the model uniformly is quite significant. However, since the proposed method ultimately uses EPnP method to solve the pose, and EPnP method requires many matched 2D-3D point pairs, we need to preserve the 3D information of the model. Therefore, the Fibonacci mesh method is adopted to sample the template on the sphere, and the method of memorization of vertex information is used to save the three-dimensional information of the template image.

a) Model vertex information memorization

Because the template image could only save 2D information when we convert the CAD model to it, the 3D information is included in the CAD model but is discarded during rendering. Thus, we use memorization techniques to retain this information during rendering:

The original vertex coordinates are defined in local space. Then, the model matrix is used to transform the vertex coordinates into world space, which is a much larger scale. These coordinates will then share a new coordinate system with other objects. Next, we transform the vertex coordinates into the view space.
so that each coordinate is viewed from the camera's point of view. The view space is the result of converting the coordinates of the world space into the coordinates in front of the camera's field of vision. This series of coordinate transformations are saved in the view matrix. We then use the projection matrix to project the coordinates into the clip space, filtering out the vertices that will appear on the screen.

The projection matrix creates a truncated body that defines the visible space. Anything outside this truncated body will not end up in the clip space and will be clipped. Each coordinate inside the truncated body is mapped to a point on the clip space.

Finally, a perspective division is performed on points in the clip space to convert them into screen space and rasterize them for display on the screen.

Vertex coordinates in clip space can be calculated from vertex coordinates in local space using Eq. (1):

\[
V_{\text{clip}} = M_{\text{projection}}M_{\text{view}}M_{\text{model}}V_{\text{local}}
\]  

where \( V_{\text{clip}} \) and \( V_{\text{local}} \) represent vertex coordinates in clipping space and local space.

Since the subsequent perspective division is irreversible, the information in screen space cannot be used to calculate the original vertex coordinates. To solve this problem, we compute all vertices with the same flow and parameters and saved the data, while using Z-Buffer to determine whether a vertex is visible, that is, if the current position pixel has been rendered and its Z-Buffer value is less than the distance from the calculated vertex to the viewpoint, then the vertex is invisible. At last, we pair the vertex coordinates in screen space with the vertex coordinates in local space and save them as part of the current template image.

b) template sampling based on Fibonacci mesh

When sampling parts, template sampling should be carried out from different angles, and sampling points should be evenly distributed. There should be no over-dense or over-sparse sampling in one place. Given that the part is at the center of the unit ball, the problem of uniform sampling of the template can be transformed into the problem of the uniform distribution of points on the sphere. A relatively simple and direct method is to take sampling points according to longitude and latitude, that is, the longitude and latitude are divided into several equal parts, and then the intersection point is taken as the sampling point. This method is relatively easy to implement and understand, but it will lead to over-dense sampling points on the north and south poles of the sphere, and over-sparse sampling points near the equator, which does not achieve uniform sampling. Therefore, the Fibonacci mesh method was used to conduct uniform sampling (Fig. 2), with the formula as follows:

\[
z_i = \frac{2i - 1}{N} - 1,
\]
\[ x_i = \sqrt{1 - z_i^2} \cdot \cos(2\pi i\phi), \]

where \((x, y, z)\) is the coordinate of the \(i\)-th sample point. \(N\) is the amount of the sample points. 

\[ \varphi = \frac{\sqrt{5} - 1}{2}, \] which is the golden ratio. The distribution results are shown below.

### 2.2 Object registration based on extraction and matching of multiple edge features

In industrial metal parts, there are many holes and straight lines which are projected to ellipses and lines in the 2D real images. Therefore, we choose lines and ellipses as the edge features for the object registration. The overall idea of the algorithm is roughly described as follows. Firstly, template rough matching is carried out to select a template image that is close to the real image. Then, multiple edge features of the template image and the real image are extracted and matched respectively. Edge feature points are then extracted from the matched multiple edge features for pose registration. Since there is a matching relationship to edge features, these edge feature points naturally have the same matching relationship. In addition, we use the method of memorizing sampling to generate templates, so we indirectly obtain the point-pair relationship between 2D coordinates in the real image screen space and 3D coordinates in the model space. Finally, the registration can be done by EPnP method.

c) matching of multiple edge features

For edge features, we mainly choose straight lines and circles. For the matching of straight lines, we have ever proposed a method [27] which utilizes the position relationship among extracted lines to generate descriptors for lines. Then it could match the lines between real images and template images by these descriptors. The method proposed in [28] is used to extract ellipse. For the matching of circle features, we adopt a method based on Distance Map. In general, the distance between the circle and the surrounding line is used to generate a descriptor of the circle, and then the circle is matched according to the descriptor. Specifically, we define the feature of a line coplanar with a circle as a coplanar line of that circle. As shown in Fig. 3, in the distance map-based circle matching process, we first calculate the distance map of each circle in the real image and template image respectively. The key of the distance map is the hash value of a coplanar line of a circle, and the value is the distance from the center of the circle to the line. The distance between a coplanar line and a circle can be calculated by Eq. (5):

\[
\text{distance} = \frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}}
\]
where $Ax + By + C = 0$ is the equation of the coplanar line, $(x_0, y_0)$ is the coordinate of circle center.

The lines which are matched share the same hash value. The comparison of two descriptors includes two steps: unication and normalization. In the unication step, both maps need to remove keys that are not contained in the other.

After the unication step, both descriptors have the same key set which is beneficial to the calculation of the distance between them.

In the normalization step, both descriptors need to be scaled where the scaled factor is the sum of all values in the descriptors. And the distance can be obtained by calculating the absolute value of the difference between the values in the map that have the same key and summing them up.

When the difference is less than the ratio of two times the length of the short axis to scaled factor of descriptors in the template, it can be considered that the two circles correspond to the same assembly reference circle, that is, the two circles match each other.

d) matching of edge feature points

Multiple edge features are divided into two types in the paper: straight line and circle. From these two types of edge features, two kinds of edge feature points can be calculated respectively. One is the crosspoint of straight lines and the other is the center of circles. The matching of feature points is based on the matching of edge features. The center of circles that matches successfully can be matched naturally according to this matching relationship. The matching of the crosspoint of the straight lines is also completed according to the matching relation of lines. Since it is impossible to judge whether lines intersect in three-dimensional space on a plane image, we first use the three-dimensional line information recorded by memorization sampling to judge whether lines are coplanar. If the lines are coplanar, the intersecting operation is carried out by Eq. (6). If the lines are not coplanar, the equation could not be solved.

$$
\begin{align*}
    x &= x_a^s k + (1 - k) x_a^f = (x_b^s - x_b^f) r + x_b^f \\
    y &= y_a^s k + (1 - k) y_a^f = (y_b^s - y_b^f) r + y_b^f \\
    z &= z_a^s k + (1 - k) z_a^f = (z_b^s - z_b^f) r + z_b^f
\end{align*}
$$

6

where $r$ and $k$ are parameters, superscript $s$ represents the beginning of the line, and $f$ represents the end of the line. The two random lines are $L_a$ and $L_b$. $P_1(x_a^s, y_a^s, z_a^s)$ and $P_2(x_a^f, y_a^f, z_a^f)$ are the two endpoints of $L_a$ and $P_3(x_b^s, y_b^s, z_b^s)$ and $P_4(x_b^f, y_b^f, z_b^f)$ are the two endpoints of $L_b$. 
The two-dimensional lines corresponding to the real images of three-dimensional lines corresponding to the coplanar are also carried out by Eq. (7). Several groups of 2D-3D point pairs generated are saved and EPnP can be used for pose registration.

\[
\begin{align*}
x &= x_s^a + \tau (x_f^a - x_s^a) = x_s^b + \mu (x_f^b - x_s^b) \\
y &= y_s^a + \tau (y_f^a - y_s^a) = y_s^b + \mu (y_f^b - y_s^b)
\end{align*}
\]

where the two lines are \(L_a\) and \(L_b\), \(P_1(x_s^a, y_s^a)\) and \(P_2(x_f^a, y_f^a)\) are the two endpoints of \(L_a\), and \(P_3(x_s^b, y_s^b)\) and \(P_4(x_f^b, y_f^b)\) are the two endpoints of \(L_b\).

### 2.3 Rendering of CAD model

According to the last section, the pose of an object has been obtained, which is also the translation matrix between object coordinate and camera coordinate \(M_{OC}\). To render the CAD model of parts on the real image of an object, we need to calculate the translation matrix between part coordinate and camera coordinate \(M_{PC}\). And through the CAD model, we could obtain the \(M_{PC}\) by the following equation:

\[
M_{PC} = M_{PO} \times M_{OC}
\]

where \(M_{PO}\) is the translation matrix between part coordinate and object coordinate and can be calculated by CAD model.

And to increase the authenticity of rendering, we render the color of CAD models by the direction of the triangles’ vector. For a vector \(\vec{a} = [x, y, z]\), its triangular patch will be rendered by the following equation:

\[
[R, G, B] = 255 \times \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} / \begin{bmatrix}
|\vec{a}| \\
|\vec{a}| \\
|\vec{a}|
\end{bmatrix}
\]

### 3 Experiment

#### 3.1 experimental setup

In order to verify the performance of the proposed method, we set up the following two experiments. In the first experiment, we obtained image information of two reflective texture-less parts and registered them to verify the accuracy and robustness of the proposed method. To test the performance of the proposed method better, another two existing methods were compared. [27] is a method based on
geometric features. [29] utilizes deep learning to solve the problem of object registration. In the second experiment, we obtained the image information of a part of an assembly and completed the process of virtual assembly to verify the performance of the proposed method on augmented reality intuitively. The camera used in the first experiment is a 2048*2448 grayscale camera. To make the visual effect of virtual assembly more real, the camera used in the second camera is a 1280*1024 three-channel industrial camera. The equipment's intrinsic parameters and the distortion coefficients of the camera were calibrated by the method mentioned in [30]. The experimental objects used are oil pumps. One is a horizontal plunger pump, and the other is a vertical plunger pump. Both pumps are reflective and texture-less.

3.2 results

e) Experiment I

The ground truths of the registration results are denoted by $x_t, y_t, z_t, Rx_t, Ry_t, Rz_t$, while the measured values of the 6D poses are denoted by $x_m, y_m, z_m, Rx_m, Ry_m, Rz_m$. Two indexes are used to evaluate the accuracy of the methods, including $\Delta \text{pos}[\text{mm}]$ and $\Delta \text{rot}[^\circ]$, which are defined following.

$$\Delta \text{pos} = \frac{\Delta \text{pos}, x + \Delta \text{pos}, y + \Delta \text{pos}, z}{3}$$

$$\Delta \text{rot} = \frac{\Delta \text{rot}, x + \Delta \text{rot}, y + \Delta \text{rot}, z}{3}$$

Where $\Delta \text{pos}, x = |x_t - x_m|$, $\Delta \text{pos}, y = |y_t - y_m|$, $\Delta \text{pos}, z = |z_t - z_m|$, $\Delta \text{rot}, x = |Rx_t - Rx_m|$, $\Delta \text{rot}, y = |Ry_t - Ry_m|$, $\Delta \text{rot}, z = |Rz_t - Rz_m|$.

We ran twenty tests on each pump and the average error of registration results is recorded in Table 1. Due to space constraints, we choose three results of each pump which are shown in Table 2. When the $\Delta \text{pos}$ is over 15mm or $\Delta \text{rot}$ is over 20°, we consider the result is incorrect. The symbol “/” in the table indicates an incorrect result. To visualize the results, we render the CAD models according to the registration result and overlap them on the real images as shown in Fig. 4.

As shown in Tables 1 and 2, He-Jiang’s method is a little weaker than the proposed method and the result of Sundermeyer’s method is worst. This is because He-Jiang’s method utilizes the endpoints of straight lines to measure pose, but when the straight lines are not extracted completely, the accuracy of endpoints will be affected, which will increase the error of the method. Sundermeyer’s method is based on deep learning which needs to be trained by lots of templates. If the image in the experiment does not exist in the training set, the accuracy of the result will be affected. However, the features used in the proposed
method are multiple edge features which are also stable on the reflective and texture-less objects. The feature points used in the proposed method are edge feature points which will not be affected by the completeness of the extraction of edge features. As a whole, the proposed method is suitable for the registration of reflective and texture-less objects. It can also be seen that the performance of the proposed method is superior to that of the other two methods.

**f) Experiment II**

To verify the performance of the proposed method on virtual assembly based on augmented reality, we set up an experiment which applied the proposed method on virtual assembly. The visual results are presented in Fig. 5 and Fig. 6. The real part represents the current assembly state of the assembly and parts to be assembled. The virtual part represents the part that needs to be assembled currently. For instance, as shown in Fig. 5, the first image shows all the parts of the assembly and the second image shows the first step of assembling where we need to assemble two bearings and a shaft in the base. The third image shows that the end cover should be assembled on the base in the second step. And the last image shows the assembly that has been assembled completely. In these renderings, parts to be assembled are rendered exactly where they should be assembled. And through a series of renderings, we can clearly understand the assembly process of the entire assembly.

| Table 1 | The average error of the correct registration result |
|---------|-----------------------------------------------|
|         | The proposed method | He - Jiang [27] | Sundermeyer [29] |
|         | Δpos/mm | Δrot/° | Δpos/mm | Δrot/° | Δpos/mm | Δrot/° |
|         | 1.44    | 1.06   | 2.01    | 1.68   | 6.08    | 7.61   |
|         | 1.85    | 1.24   | 2.44    | 2.01   | 4.80    | 8.16   |

| Table 2 | Comparison of the results of the object registration |
|---------|-----------------------------------------------------|
| part    | pose | The proposed method | He - Jiang [27] | Sundermeyer [29] |
|         |      | Δpos/mm | Δrot/° | Δpos/mm | Δrot/° | Δpos/mm | Δrot/° |
| 1       | 1    | 1.28    | 0.85   | 1.99    | 1.48   | 4.12    | 4.07   |
|         | 2    | 1.53    | 1.09   | 2.30    | 1.91   | 7.32    | 13.25  |
|         | 3    | 1.27    | 1.04   | 1.77    | 1.38   | 4.80    | 6.36   |
| 2       | 1    | 1.89    | 1.35   | 2.58    | 1.98   | 4.34    | 9.07   |
|         | 2    | 2.12    | 1.20   | 2.43    | 1.97   | /       | /      |
|         | 3    | 1.75    | 1.84   | /       | /      | 3.80    | 6.36   |

4 Conclusion
In this paper, we proposed a reflective texture-less object registration method for virtual assembly based on augmented reality. Firstly, a convenient and rational method for templates sampling and generation was applied. Next, an object registration method based on multiple edge features was used to calculate the 6D pose of objects. Finally, the virtual model was rendered and combined with the real image. We also set up an experiment to verify the performance of the proposed method. Experiment shows that the proposed method could achieve high accuracy and robustness when registering the reflective texture-less objects. According to the object registration experiment results, the position error of the proposed method is lower than 2mm and the rotation error is lower than 1.5°. According to the virtual assembly experiment, the proposed method could accurately render the part to be assembled and exactly show the assembly process. The future research of this paper could focus on recovering 3D information from 2D images. This will be beneficial to the judgment of occlusion relation between a rendered model and a real part, which will improve the authenticity of virtual assembly greatly.

Declarations

Author contribution Conceptualization, Zaixing He and Xinyue Zhao; methodology, Zaixing He; software, Jindong Zhao, and Wuxi feng; validation, Xinyue Zhao and Quanyou Wang; Data Curation, Huilong Jiang. All authors have read and agreed to the published version of the manuscript.

Funding This study was funded by the National Natural Science Foundation of China (51775497 and 51775498) and the Zhejiang Provincial Natural Science Foundation (LY21E050021).

Data availability Data will be available from corresponding author upon reasonable request.

Ethical approval Not applicable

Consent to participate Not applicable

Consent to publish Not applicable

Conflict of Interest The authors declares that they have no conflict of interest.

References

1. M. Gattullo, A. Evangelista, A. E. Uva, M. Fiorentino and J. L. Gabbard, "What, How, and Why are Visual Assets Used in Industrial Augmented Reality? A Systematic Review and Classification in Maintenance, Assembly, and Training (From 1997 to 2019)," in IEEE Transactions on Visualization and Computer Graphics, vol. 28, no. 2, pp. 1443-1456, 1 Feb. 2022.

2. L. F. d. S. Cardoso and E. R. Zorzal, "An Augmented Reality Review on Production Environments," 2018 20th Symposium on Virtual and Augmented Reality (SVR), Foz do Iguacu, Brazil, 2018, pp. 143-149.
3. K. Mühlan, K. A. Przybysz, F. Lindner, D. Akrmanová, D. Winkler and S. Keil, "A Review and Implementation Framework of Industrial Augmented Reality," *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Västerås, Sweden, 2021, pp. 01-04.

4. C. Botto et al., "Augmented Reality for the Manufacturing Industry: The Case of an Assembly Assistant," *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, Atlanta, GA, USA, 2020, pp. 299-304.

5. D. Nguyen and G. Meixner, "Gamified Augmented Reality Training for An Assembly Task: A Study About User Engagement," *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Leipzig, Germany, 2019, pp. 901-904.

6. A. Luxenburger et al., "Extended Abstract: Augmented Reality for Human-Robot Cooperation in Aircraft Assembly," *2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, San Diego, CA, USA, 2019, pp. 241-2411.

7. S. Hoque, M. Y. Arafat, S. Xu, A. Maiti and Y. Wei, "A Comprehensive Review on 3D Object Detection and 6D Pose Estimation With Deep Learning," in *IEEE Access*, vol. 9, pp. 143746-143770, 2021.

8. R. Xiaogang, Y. Wenjing, H. Jing, G. Peiyuan and G. Wei, "Monocular Depth Estimation Based on Deep Learning: A Survey," *2020 Chinese Automation Congress (CAC)*, Shanghai, China, 2020, pp. 2436-2440.

9. K. Park, T. Patten, J. Prankl, and M. Vincze, "Multi-Task Template Matching for Object Detection, Segmentation and Pose Estimation Using Depth Images," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7207-7213.

10. Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145-155, 1992/04/01/ 1992.

11. P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," IEEE Trans. Pattern Anal. Mach. Intell., vol. 14, no. 2, pp. 239–256, 1992.

12. Y. Konishi, K. Hattori, and M. Hashimoto, "Real-Time 6D Object Pose Estimation on CPU," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 3451-3458.

13. H. Zhang and Q. Cao, "Texture-less object detection and 6d pose estimation in RGB-D images," *Robotics And Autonomous Systems*, vol. 95, pp. 64-79, Sep 2017.

14. W. Pan, F. Zhu, Y. Hao, and L. Zhang, "Fast and precise 6d pose estimation of textureless objects using the point cloud and gray image," *Applied Optics*, vol. 57, no. 28, pp. 8154-8165, Oct 1 2018.

15. Y. He, W. Sun, H. Huang, J. Liu, H. Fan and J. Sun, "PVN3D: A Deep Point-Wise 3D Keypoints Voting Network for 6DoF Pose Estimation," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11629-11638, doi: 10.1109/CVPR42600.2020.01165.

16. D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

17. H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proceedings of the 9th European conference on Computer Vision - Volume Part I*, 2006.
18. Y. Konishi, Y. Hanzawa, M. Kawade, and M. Hashimoto, "Fast 6D Pose Estimation from a Monocular Image Using Hierarchical Pose Trees," in *Springer, Cham*, 2016.

19. J. J. Lim, A. Khosla, and A. Torralba, "FPM: Fine Pose Parts-Based Model with 3D CAD Models," Springer, Cham, 2014.

20. W. Pan, F. Zhu, Y. Hao, and L. Zhang, "6D Pose Estimation Based on Multiple Appearance Features from Single Color Image," in 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), 2017, pp. 406-411.

21. M. Rad and V. Lepetit, "BB8: A Scalable, Accurate, Robust to Partial Occlusion Method for Predicting the 3D Poses of Challenging Objects without Using Depth," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3848-3856.

22. A. Crivellaro, M. Rad, Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit, "A Novel Representation of Parts for Accurate 3D Object Detection and Tracking in Monocular Images," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4391-4399.

23. S. Peng, Y. Liu, Q. Huang, H. Bao, and X. Zhou, "PVNet: Pixel-wise Voting Network for 6DoF Pose Estimation," 2018.

24. X. Zhang, Z. Jiang, H. Zhang, and Q. Wei, "Vision-Based Pose Estimation for Textureless Space Objects by Contour Points Matching," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 5, pp. 2342-2355, 2018.

25. A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," in 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2938-2946.

26. Lepetit V , Moreno-Noguer F , Fua P . EPnP: An Accurate O(n) Solution to the PnP Problem[J]. International Journal of Computer Vision, 2009, 81(2):155-166.

27. Z. He, Z. Jiang, X. Zhao, S. Zhang and C. Wu, "Sparse Template-Based 6-D Pose Estimation of Metal Parts Using a Monocular Camera," in *IEEE Transactions on Industrial Electronics*, vol. 67, no. 1, pp. 390-401, Jan. 2020, doi: 10.1109/TIE.2019.2897539.

28. Fornaciari M , Prati A , Cucchiara R . A fast and effective ellipse detector for embedded vision applications[J]. Pattern Recognition, 2014, 47(11):3693-3708.

29. Sundermeyer, M., Marton, Z. C., Durner, M., Brucker, M. & Triebel, R. Implicit 3D orientation learning for 6D object detection from RGB images. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (2018). doi:10.1007/978-3-030-01231-1_43.

30. Z. Zhang, "A flexible new technique for camera calibration," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pp. 1330-1334, Nov. 2000, doi: 10.1109/34.888718

**Figures**
Figure 1

The overall workflow of the proposed method
Figure 2

Fibonacci mesh
Figure 3

Calculation of distance map of ellipses

\[ \text{dis} = |A_1 - A_2| + |B_1 - B_2| + |C_1 - C_2| \]

Figure 4

proposed method  He-Jiang’s method [27]  Sundermeyer’s method [29]  proposed method  He-Jiang’s method [27]  Sundermeyer’s method [29]
Real image overlapped by the CAD template in registration results of three algorithms.

Figure 5

Augmented reality assembling of horizontal plunger pump
Figure 6

Augmented reality assembling of vertical plunger pump