2015 IEEE/SICE International Symposium on System Integration (SII)
December 11-13, 2015. Meijo University, Nagoya, Japan

Real-time 3D Model-based Tracking of Work-piece with Monocular Camera

Wenjun Zhu[1], Peng Wang[1], Fudong Li[2], Jianhua Su[1] and Hong Qiao[1]

Abstract—**The tracking of 3D work pieces with real-time speed is becoming more and more important for some industrial tasks, such as work pieces grasping and assembly, especially in complex environment. In this paper, we propose a novel real-time 3D model- based tracking method for work pieces with monocular camera, which can provide accurate 3D location information of the tracking object continuously. Three processes are designed in the proposed method, i.e., the offline global model library generation process, the online dynamic library updating and selection process, and the 3D work piece localization process. The method is suitable for the texture-less work-pieces in industrial applications. In the offline global model library generation process, the CAD models of the work piece are used to generate a set of discrete 2D views matching libraries. In the online dynamic library updating and selection process, the previous 3D location information of the work piece is used to predict the following location range, and a dynamic discrete views library with a small number of models is selected for localization. Then, the work piece is localized with high-precision and real time speed in the 3D work piece localization process. The small range of the library enables a real-time matching. Experimental results demonstrate the high accuracy and high efficiency of the proposed method.**

I. Introduction

Vision system is becoming more and more popular in industrial applications because of the automation level demand and the improvements of the machine vision technologies. In industrial area, vision system was first applied mainly in 2D defects detection applications or planar assembly in circuit board manufacturing. Recently, vision systems are widely used in robotic arms or other manipulators to detect and localize the target object. 2D information of the complex shape of the target object is not enough for many industrial tasks, such as grasping and assembly. Usually, 3D information of the target object can be obtained using stereo vision with binocular vision system or multi-camera system, which are expensive and not rigid enough for industrial environment.

3D object detection and localization with monocular camera is a challenging technology, and many methods have been proposed. Feature-based approaches [1-3] use gray value edges, intersections of straight lines which approximate gray value edges, or more complex features that from grouping extracted primitives. The extracted features are matched to the corresponding 3D object features, and 3D pose of the object can be calculated directly from the corresponding features. Feature-based approaches are vulnerable to occlusion and blur. View-based approaches [4，5] were popular for industrial application. In these approaches, 2D views of the target object were pre-computed, and then they were compared with the image obtained by the camera. 3D position and pose information of the target object in the image could be drawn from the matched pre-computed 2D view, which always corresponded to a specific pose during its generation. However, these approaches tried to cover the full geometric search space, the generated 2D view library was huge which made the matching speed slow, and these methods were not acceptable for tracking in practice

Recently, the detection and localization of 3D work pieces continuously with real-time speed is becoming more and more important for some industrial tasks, such as work pieces grasping and assembly, especially in complex environment.

Since view-based 3D object detection approaches suffer the slow speed of matching, they are not popularly seen in real-time tracking applications. Two main popular kinds of 3D object tracking techniques have been used in real applications, i.e., marker-based 3D object tracking techniques and marker-less natural features-based 3D object tracking approaches. In [6], the concentric contrasting circle fiducials were used to track the 3D object in augmented reality application. Method in [7] uses color-coded fiducials to identify 3D object more reliably. An inner dot and a surrounding outer ring formed one fiducial, and their colors are different. Planar, rectangular fiducials were used to estimate the pose of the 3D object [8, 9], which showed favorable performance because a single fiducial is enough to estimate the pose. This kind of tracking techniques requires one or more fiducials are visible at all times, and this character determines that it is not robust to occlusion. Besides, the need for attaching fiducials to the 3D object limited the range of application of marker-based tracking methods. Marker-less natural features-based 3D object tracking approaches use features naturally present in the images, such as edge-based tracking methods [10-12]. These methods have a fairly low computational complexity.

Although view-based methods are usually not suitable for tracking because of their low speed matching, but good pose and position accuracy can be obtained with these methods, besides, they are robust to occlusion and light changing. These features are desirable for industrial applications, so they are attracting more attention from some researchers. Improvements for speeding up the view-based approaches have been investigated [13-15]. During these improvements, CAD model-based methods showed promising prospect for the recognition of 3D work piece with monocular camera and

1 The authors are with the Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, PRC (e-mail: wenjun.zhu@ia.ac.cn, peng_wang@ia.ac.cn, jianhua.su@ia.ac.cn, hong.qiao@mail.ia.ac.cn).

2 The author is with School of Information Engineering, Yangzhou university, Yangzhou, 225000, PRC (e-mail: fdli@yzu.edu.cn)

Peng Wang is the corresponding author.

also the potential of real-time tracking. In [16], a hierarchy of 2D views of the CAD model is derived offline and the work piece can be efficiently recognized by minimizing a geometric distance between the image and the derived 2D views. This method is robust to noise, occlusions and complex environment. The 3D pose of the work piece can be detected with a position accuracy of up to 0.12 percent with respect to object distance, and an orientation accuracy of up to 0.35 degree. But the recognition time is still not up to a real-time level, and typical runtimes are in the range of a few hundred milliseconds.

In this paper, we propose a novel real-time 3D model-based tracking method for work pieces with monocular camera, which can provide accurate 3D location information of the tracking object continuously. Three processes are designed in the proposed method, i.e., the offline global model library generation process, the online dynamic library updating and selection process, and the 3D work piece localization process. The method is suitable for the texture-less work-pieces in industrial applications. In the offline global model library generation process, the CAD models of the work piece are used to generate a set of discrete 2D views matching libraries. In the online dynamic library updating and selection process, the previous 3D location information of the work piece is used to predict the following location range, and a dynamic discrete views library with a small number of models is selected for localization. Then, the work piece is localized with high-precision and real time speed in the 3D work piece localization process.

The outline of the paper is as follows: Section II gives a framework of the proposed approach. In section III, the details of the proposed method are given. The experimental tracking results and accuracy verification results are given in section IV. Section V gives the conclusions and prospects.

II. Framework

Framework of the proposed tracking approach is shown in Fig. 1.The introduced real-time 3D model-based tracking approach can be divided into offline phase and online phase.

In the offline phase, i.e., the offline global model library generation process, a hierarchical model is generated with the CAD model of the object and the intrinsic parameters of the camera. The main task of the model generation is to derive a set of discrete libraries. Each discrete library contains a small range of 2D hierarchy views of the object that can be used to search the object efficiently in a frame. All 2D hierarchical views of the set of the discrete libraries cover full range of poses, within which the object might appear in front of the camera. The set of these discrete hierarchical libraries, named static global library, are stored in hard disk of computer. The detail of this process will be explained in section III-A.

The online phase contains two processes, i.e., the online dynamic library updating and selection process, and the 3D work piece localization process. In the online dynamic library updating and selection process, a dynamic local library is introduced which is composed of specific discrete hierarchical libraries from the static global library. The searching library used for the 3D object recognition is selected from the dynamic local library, based on the state (location and pose) of the previous frame. Meanwhile, the dynamic local library is updated. The purpose of updating the

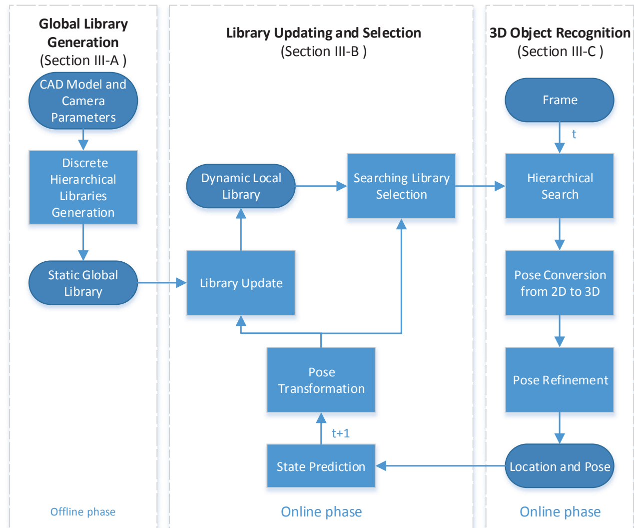


Fig. 1. Framework of the proposed tracking approach

dynamic library is to select those discrete 2D views matching libraries, views of which are within a specific range from the pose of the previous frame, and to abandon those exceed the limits. The dynamic library is stored in random access memory (RAM). Using the selected matching library, the pose of the current frame can be calculated within a short time in the 3D work piece localization process. This phase will be explained in section III-B and section III-C.

III. Algorithm

The proposed real-time 3D tracking method with monocular camera is based on a view-based localization method using the CAD model of the object as priori knowledge. One of the key factors for the efficiency of localization is the generation of the 2D views library. The less number of 2D views the suitable library contained, the more efficient the localization will be. The core concept is to set and select a suitable searching library for localization, and then to track the 3D object efficiently in the frames.

A. Global Library Generation

In the matching process, six degrees of freedom of an object in 3D space lead to a huge number of 2D views that must be compared to the image. While an exhaustive search guarantees that the best fitting 2D view is always found, it is much too slow for practical applications. Therefore, most view-based approaches try to reduce the complexity by pre-computing views of which the camera is placed on the surface of a virtual viewing sphere. With this approach, only three degrees of freedom need to be sampled offline (distance, latitude and longitude). The resulting views are compared online to the image, where a remaining degree of freedom must be considered by rotating the 2D view around the optical axis of the camera. Unfortunately, for time-critical applications there are still too many views that must be transformed and compared to the image. Hierarchical view generation method has been studied in [16] and it is used to speed up the localization step in our tracking method.

In the view-based localization approaches, only one 2D views library with large ranges is generated. The larger ranges the matching library contains, the longer time the matching takes. When the ranges of a library exceed certain level, real-time matching will be impossible.

To reduce the matching time during the tracking, we split the united library into a group of discrete libraries, combined with the hierarchical view generation method. Since each discrete library only contains small ranges of the searching space, the matching time can be reduced dramatically.

The group of discrete libraries is generated in the following way. Suppose the united template library is $L\{[d_{min}, d_{max}], [\phi_{min}, \phi_{max}], [\lambda_{min}, \lambda_{max}]\}$.The united template library is split into a group of discrete libraries using the range intervals of distance Δd, latitude $\Delta\phi$ and longitude $\Delta\lambda$. A discrete library only contains small ranges of the searching space. The generation process of the discrete libraries is as follows.

The starting discrete library is $L\{[d_{min}, d_{min}+\Delta d], [\phi_{min}, \phi_{min}+\Delta\phi], [\lambda_{min}, \lambda_{min}+\Delta\lambda]\}$, the next one is $L\{[d_{min}, d_{min}+\Delta d], [\phi_{min}, \phi_{min}+\Delta\phi], [\lambda_{min}+\Delta\lambda, \lambda_{min}+2\Delta\lambda]\}$...until the whole searching ranges are covered, and the last one is $L\{[d_{max}-\Delta d, d_{max}], [\phi_{max}-\Delta\phi, \phi_{max}], [\lambda_{min},-\Delta\lambda,\lambda_{max}]\}$.

This kind of discrete library group has a potential defect: when the previous corresponding pose of the work piece lies on the edge of the library, the corresponding pose of the work piece may fall out of the searching range the library contains at this circle, and the matching will fail. Steps of the discrete library are introduced to solve this problem. Take the longitude range for example, the corresponding libraries generated are illustrated in Fig. 2.

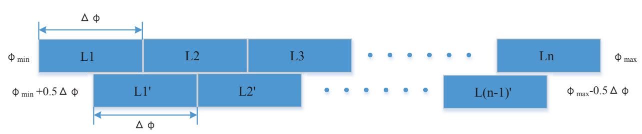


Fig. 2. Overlapped discrete libraries

The overlapped discrete libraries ensure that the corresponding pose of the work piece never lies on the edge of the chosen discrete library.

B. Dynamic Library Updating and Selection

In industrial application, the work piece always moves consecutively. Thus, previous poses can be used to predict the pose of the current frame.

The pose variation from frame *n-1* to frame *n* can be estimated from previous poses in frame (*n-1*) and (*n-2*).

$$\Delta P_n^{estimate} = P_{n-1} - P_{n-2} \quad (1)$$

where, $\Delta P_n^{estimate}$ is the pose variation from frame *n-1* to frame *n*.

The above equation is based on an assumption that the pose changing velocity of the work piece is constant, and since the interval between two frames is less than one hundred milliseconds, this assumption is reasonable. The estimated pose $P_n^{estimate}$ of current frame is then used to select a suitable matching library from the dynamic local library. The distance between the estimated pose of current frame and the center poses of neighbouring individual discrete library in the dynamic library are calculated. The individual discrete library in the dynamic library with the shortest distance toward the estimated pose of current frame is selected as the matching library in the following 3D localization step.

To overcome the problem that the static library always occupy huge storage space, a dynamic library is set up, which is stored in RAM (Random-Access Memory) and contains libraries not only the selected match library but also the libraries around. The process of updating dynamic library is to prepare those discrete 2D views libraries, whose center poses are within a specific distance from the estimated pose of the current frame, and to abandon those exceed the limits at the same time.

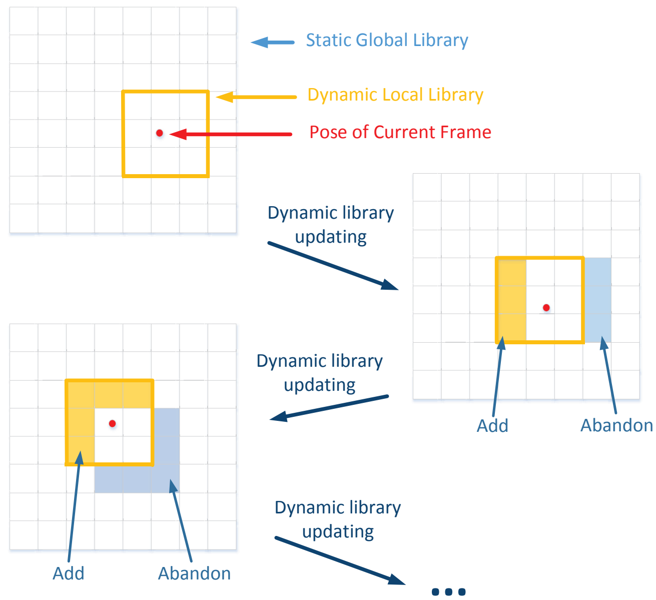


Fig. 3. Overlapped discrete libraries

A major problem in model-based 3D recognition is that if we try to deal with the six degrees of freedom of the object in 3D space, a huge number of 2D templates will be generated, and this makes the matching process very time consuming. In our approach, spherical quadrilateral space range [16] is adopted with a restriction that the starting pose of the virtual camera coordinates are always fixed to be exactly the same with the pose of the CAD model coordinates. Fig. 4 shows the starting poses of the above mentioned two coordinates.

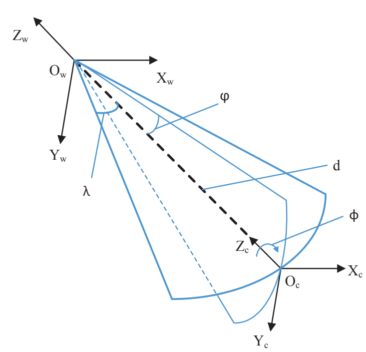


Fig. 4.Illustration of the space ranges of the views

{W} is the coordinates of the CAD model of the work piece, and {C} is the virtual camera coordinates. The parameters of the view ranges are defined as follows. The distance between the origins of the virtual camera coordinates and the CAD model coordinates is d; Latitude φ is the rotation angle around the axis X_w; Longitude λ is the rotation angle around the axis Y_w; ϕ is the rotation angle around the optical axis of the camera, also the axis Z_c. The 2D view of the work piece in the virtual camera image is $S\{d, \varphi, \lambda\}$, which is generated in the off-line phase. Corresponding 2D view $T\{d, \varphi, \lambda, \phi\}$ is generated in the on-line matching process.

It is convenient to represent the 2D view with $T\{d, \varphi, \lambda, \phi\}$. But when the match completes, if we still use the spherical coordinates to describe the pose and position of the work piece, it would become inconvenient for the manipulation of the robot to grab the work piece. So, the pose and position of the work piece is expressed with 3 position coordinates and 3 rotation angles $W\{x, y, z, \alpha, \beta, \gamma\}$, where, (x, y, z) is the 3D coordinates of the origin of the CAD coordinates { W } in the camera coordinates { C }, and (α, β, γ) are the roll, pitch, yaw angles of the CAD coordinates with respect to the camera coordinates.

The initial poses of { W } and { C } are identical. Pose of camera coordinates { C } is obtained from { W } in the following steps. First, { C } is rotated aroud X_w for angle φ, and then it is rotated around Y_w for angle λ, finally it is rotated around the new Z_c for angle ϕ. If we reverse this process, { W } can be obtained from { C }. First, { C } is rotated around Z_c for angle $-\phi$, then it is rotated around the new Y_c for angle $-\lambda$, finally it is rotated around the new X_c for angle $-\varphi$. The coordinates obtained from { C } in the above reverse process is { W }.

{ W } can be obtained with the roll, pitch, yaw method from { C }. First, { C } rotated around Z_c for angle γ, then it is rotated around Y_c for angle β, finally it is rotated around X_c for angle γ.

The pose of coordinates { W } obtained in the reverse process of the 2D view generation process and the pose of { W } obtained with the roll, pitch, yaw style from { C } is the same. The two processes expressed in the rotation matrix is as follows.

$$\mathrm{Rotz}(\gamma)\mathrm{Roty}(\beta)\mathrm{Rotx}(\alpha) = \mathrm{Rotz}(-\phi)\mathrm{Roty}(-\lambda)\mathrm{Rotx}(-\varphi) \quad (2)$$

where, $\mathrm{Rotx}(\theta)$, $\mathrm{Roty}(\theta)$, $\mathrm{Rotz}(\theta)$ are rotation matrix around X, Y, Z axis（$c\theta=\cos(\theta)$, $s\theta=\sin(\theta)$.

$$\mathrm{Rotx}(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\theta & s\theta \\ 0 & -s\theta & c\theta \end{bmatrix}$$

$$\mathrm{Roty}(\theta) = \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix}$$

$$\mathrm{Rotz}(\theta) = \begin{bmatrix} c\theta & s\theta & 0 \\ -s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

When the matching completes, $\{\phi, \lambda, \varphi\}$ are the known parameters, and note:

$$\mathrm{Rotz}(-\phi)\mathrm{Roty}(-\lambda)\mathrm{Rotx}(-\varphi) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Left side of equation (2) can be expressed as

$$\mathrm{Rotz}(\gamma)\mathrm{Roty}(\beta)\mathrm{Rotx}(\alpha) = \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ c\beta s\gamma & s\alpha s\beta s\gamma + c\alpha c\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma \\ -s\beta & s\alpha c\beta & c\alpha c\beta \end{bmatrix}$$

Substitute left side and right side expressions of (2), we can solve for:

$$\beta = \mathrm{atan2}(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2})$$

$$\gamma = \mathrm{atan2}(\frac{r_{21}}{c\beta}, \frac{r_{11}}{c\beta})$$

$$\alpha = \mathrm{atan2}(\frac{r_{32}}{c\beta}, \frac{r_{33}}{c\beta})$$

As for the position (x, y, z) of the work piece, obviously, $z = d$. x and $y$are computed from the image.

$$x = (c_x - c_{x0})f_x$$

$$y = (c_y - c_{y0})f_y$$

where, (c_x, c_y) is the center of the matched work piece in the frame, and (c_{x0}, c_{y0}) is the optical center of the image, f_x and f_y are pixel scales in width and height.

C. 3D Object Recognition

Huge number of 2D templates used for the matching of the work piece in the image can be a very time consuming process if they are compared one by one in an ordinary way. In our real-time tracking approach, we adopt a hierarchical searching strategy [16].

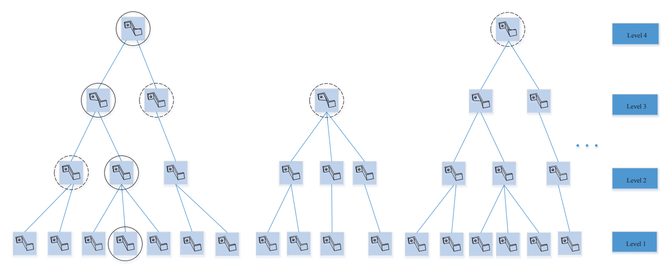


Fig. 5.Matching process with pose refinement

The hierarchical searching process is shown in Fig. 5. The circle around the 2D model means a matching process is conducted to the 2D model. While the solid line circle indicates a match candidate is found, the dash line circle means the match is discarded because of low similarity.

During the matching, similarity measures between the 2D models of the views and the current image pyramid are computed using the criterion in (3)

$$c = \left| \frac{1}{n} \sum_{i=1}^{n} \frac{\langle m_i, s_i \rangle}{\|m_i\| \cdot \|s_i\|} \right| \quad (3)$$

where, $m_i(i=1, \ldots, n)$ are the gradients of the transformed 2D model points, and $s_i(i=1,\ldots, n)$ are the gradients of pixels in the corresponding pyramid of the search image. A threshold T_{score} is predefined between 0 (no similarity) and 1 (perfect similarity) to determine whether a match candidate is find.

Exhaustive matching begins on the highest pyramid level of the hierarchy which only contains several 2D models. Before the matching begins, the 2D models are firstly rotated to cover 360 degrees with an appropriate step, and then scaled with a suitable step to cover the scales adopted in the 2D model generation process which merged the initial views into the current view. The matching will be conducted at each position of the rotated and scaled 2D models in the image.If similarity measure c of the matching exceeds T_{score}, a match candidate is considered to appear, and the 2D pose of the match labeled as a match candidate. On the next lower pyramid level, the 2D models that have no parent node are searched in the same way like the highest pyramid level. The 2D models on this level that have a match candidate node are also searched, while all the 2D models under a invalid node are discarded. The searching process is repeated until all match candidates have been tracked down to the lowest pyramid level.

After the searching process, all the matches are found in the search image, and their 2D poses (image position, discrete rotation, discrete scale) are determined at the same time. 3D pose $H_{d, \phi, \lambda, \varphi}$ of each match can be computed with the stored corresponding parameters of the 2D matching pose during the 2D views generation process and the rotation angle of the 2D model in the matching process. But the accuracy of the 3D pose is limited to the sampling of the views and the rotation steps and scaling steps during the 2D matching. Such 3D pose is insufficient for industrial application. Refinement of the 3D pose is conducted using an iterative nonlinear optimization using the Levenberg-Marquardt algorithm. Four steps are taken to complete the 3D pose refinement.

(1) 3D CAD model with the obtained 3D pose $H_{d, \phi, \lambda, \varphi}$ is projected into the search image.

(2) Corresponding sub-pixel image edge points are searched for each visible projected CAD model edge points.

(3) Minimization of the squared distances of the image edge points to their corresponding projected CAD model edge points over the six pose parameters. A refined 3D CAD model is obtained after the minimization.

(4) Iteration of (1)~(3), until the correspondences or the refined pose parameters between two iterations no longer change.

After pose refinement, the recognized object with a precise 3D posecan be determined.

IV. EXPERIMENTS AND RESULTS

This section shows the performance of the proposed 3D model-based tracking method. The proposed algorithm is tested on a computer system with the configuration of Quad-Core CPU and 16G RAM. The resolution of the image used for tracking is 640 ×480. The frame rate of the camera used is set to 12 frames/s.

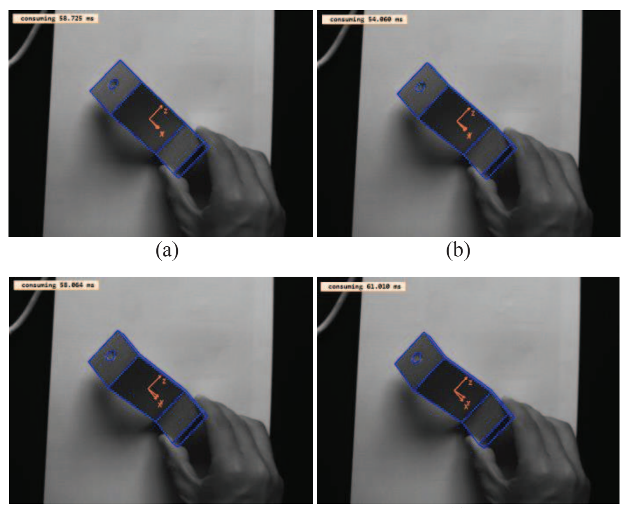
In this experiment, the tracking object is one metal bracket. The full possible searching ranges of the object are specified as: the longitude λ is from -60 degree to 120 degree, the latitude φ is from -60 to 120 degree, the camera roll range ϕ is -180 to 180 degree, the minimum and maximum distance of the camera to the object is from 68cm to 75cm.

Range intervals for the discrete hierarchical libraries are correspondingly specified as the longitude interval $\Delta\lambda$ is10 degree，the latitude interval $\Delta\varphi$ is 10 degree, the camera roll intervalΔr is 10 degree, the distance interval Δd is 2cm. The steps of each parameters are set to half of the corresponding value.

In this experiment, the object is moved with hand. For each frame, the location and pose of the object can be calculated out efficiently. In figure 6, four frames matched during the tracking process are shown. The pose of the bracket is visualized by the blue edges, and the consuming time of each frame are listed. The result shows that the pose of the bracket can be recognized accurately with the proposed approach. The consuming time for each frame is coarsely 50 ~ 70ms. The variation of location and pose in this experiment is shown in Fig. 7.

TABLE 1
VALUE OF ARGUMENTS

Symbo l	Quantity	Value	Unit
λ	Longitude	-60~120	Degree
φ	Latitude	-60~120	Degree
ϕ	Camera roll	0~360	Degree
d	Distance	68~75	cm
$\Delta\lambda$	Longitude interval	10	Degree
$\Delta\varphi$	Latitude interval	10	Degree
Δr	Camera roll interval	10	Degree
Δd	Distance interval	2	cm

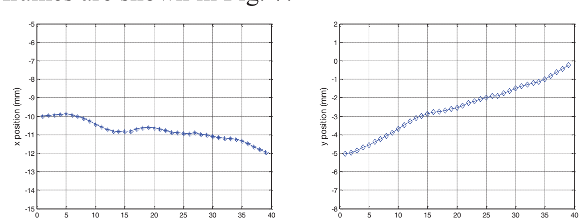


(a) (b)

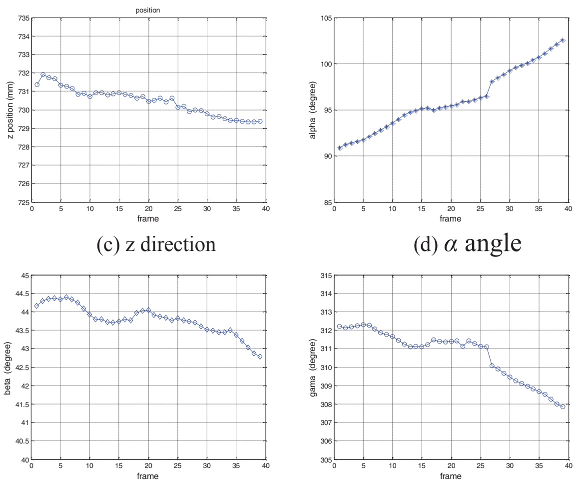
(c) (d)

Fig. 6. The bracket can be tracked successfully with the proposed approach. (a)-(d) show tracking results for the metal bracket. The pose of the bracket is visualized by the blue edges, and the consuming time of each frame are listed.

Six degrees of freedom trajectories changing with respect to frames are shown in Fig. 7.



(a) x direction (b) y direction



(c) z direction (d) α angle

(e) β angle (f) γ angle

Fig. 7.The variation of location and pose in this experiment

V. CONCLUSIONS AND FUTURE WORK

A novel method of real-time 3D tracking of work piece based on CAD model with monocular camera is proposed. Three processes are designed in the proposed method, i.e., the offline global model library generation process, the online dynamic library updating and selection process, and the 3D work piece localization process. The method is suitable for the texture-less work-pieces in industrial applications. The experiment results show that only 60ms~70ms is needed for each frame of the tracking process of the object. The proposed framework is not suitable for the situation when the object miss for a while, that is a problem to solve for our future work.

ACKNOWLEDGMENT

This work was partly supported by the NNSF (National Natural Science Foundation) of China under the grants 61379097, 61100098, 61033011 and 61210009, and the Youth Innovation Promotion Association of CAS (2015112).

REFERENCES

[1] J. H. M. Byne, J. A. D. W. Anderson. A CAD-based computer vision system[J]. Image and Vision Computing, 1998, 16(8): 533-539.

[2] H. Borotschnig, L. Paletta, M. Prantl, et al. Appearance-based active object recognition[J]. Image and Vision Computing, 2000, 18(9): 715-727.

[3] M. S. Costa, L. G. Shapiro. 3D object recognition and pose with relational indexing[J]. Computer Vision and Image Understanding, 2000, 79(3): 364-407.

[4] D. G. Lowe. Distinctive image features from scale-invariant keypoints [J]. International journal of computer vision, 2004, 60(2): 91-110.

[5] W. E. Grimson, D. P. Huttenlocher. On the verification of hypothesized matches in model-based recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 1991 (12): 1201-1213.

[6] W. A. Hoff, K. Nguyen, T. Lyon. Computer-vision-based registration techniques for augmented reality[C]. Photonics East'96. International Society for Optics and Photonics, 1996: 538-548.

[7] A. State, G. Hirota, D. T. Chen, W. F. Garrett, et al. Superior augmented reality registration by integrating landmark tracking and magnetic tracking[C]. Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. ACM, 1996: 429-438.

[8] H. Kato, M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system[C]. Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality, 1999: 85-94.

[9] H. Kato, M. Billinghurst, I. Poupyrev, et al. Virtual object manipulation on a table-top AR environment[C]. Proceedings. IEEE and ACM International Symposium on Augmented Reality, 2000: 111-119.

[10] A. I. Comport, É. Marchand, F. Chaumette. A real-time tracker for markerless augmented reality[C]. Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality, 2003: 36-45.

[11] E. Marchand, P. Bouthemy, F. Chaumette. A 2D–3D model-based approach to real-time visual tracking[J]. Image and Vision Computing, 2001, 19(13): 941-955

[12] L. Vacchetti, V. Lepetit, P. Fua. Combining edge and texture information for real-time accurate 3d camera tracking[C]. Proceedings of the 3rd International Symposium on Mixed and Augmented Reality IEEE, 2004: 48-56.

[13] P. David, D. DeMenthon, R. Duraiswami, et al. Simultaneous pose and correspondence determination using line features[C] Proceedings of Conference on Computer Vision and Pattern Recognition, 2003: 424-431

[14] P. David, D. DeMenthon. Object recognition in high clutter images using line features[C] Proceedings of the 10th International Conference on Computer Vision, 2005, 2: 1581-1588.

[15] R. Strzodka, I. Ihrke, M. Magnor. A graphics hardware implementation of the generalized hough transform for fast object recognition, scale, and 3d pose detection[C] Proceedings. 12th International Conference on Image Analysis and Processing, 2003: 188-193.

[16] M. Ulrich, C. Wiedemann, C. Steger. Combining scale-space and similarity-based aspect graphs for fast 3d object recognition[J], IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(10): 1902-1914.