

Received December 20, 2019, accepted January 21, 2020, date of publication February 11, 2020, date of current version February 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973299

# Dynamic Visual Tracking for Robot Manipulator Using Adaptive Fading Kalman Filter

JIADI QU<sup>1,2</sup>, FUHAI ZHANG<sup>1</sup>, YUNXI TANG<sup>1</sup>, AND YILI FU<sup>1</sup>

<sup>1</sup>State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150001, China

<sup>2</sup>College of Marine Electrical Engineering, Dalian Maritime University, Dalian 116026, China

Corresponding author: Fuhai Zhang (zfhhit@hit.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61673134, and in part by the Natural Science Foundation of Heilongjiang Province of China under Grant LC2017022.

**ABSTRACT** This paper focuses on the problem of visual tracking of a moving target with the temporary occlusion of image feature, a dynamic visual tracking control system for robot manipulator is developed by using adaptive fading Kalman filter (AFKF). The estimation of the residual covariance is used to compute the forgetting factor to automatically adjust the weight of the image observation data for improving the visual state estimation accuracy. When the target features are occluded, the prediction of missing observation sequence are generated by using the predicted compensation noise and preorder observation sequence to determine the forgetting factor for estimating the missing visual states. Then, a parameter adaptive law with projection error compensation is designed to realize the visual tracking with uncertain camera parameters. Finally, the trajectory tracking experiments based on a real robot platform is carried out to verify the performance of the proposed state estimator and tracking controller. The results show that the proposed method can accurately realize the visual tracking with the occluded trajectory and inaccurate camera parameters, which improves the flexibility of dynamic visual tracking of robot manipulator.

**INDEX TERMS** Visual tracking, robot manipulation, Kalman filter, occlusion.

## I. INTRODUCTION


Target tracking is widely used in the navigation of mobile robot, object grasping and capture of robot manipulator. Visual tracking of a moving target uses the camera to acquire the target image features, establish the mapping between visual and task space, and obtain the control command of the robot, which is to realize the dynamic visual servoing control of the robot manipulator [1]. The application of vision sensor can improve the robot tracking performance in complex environment.

Generally, the moving targets can be divided into three types: (1) Identified objects, such as the ground lines, landmarks, which are dynamic objects relative to robots. Such targets are often used in navigation applications of mobile robots [2], [3]. (2) Target with predictable trajectory, which is mostly used for object capture [4], [5]. (3) Dynamic targets, the trajectory is unpredictable, and the motion state can only be obtained through the observer in real time to realize the

robot tracking control. It is often used for grasping tumbling satellites [6]–[9], and face tracking [10], [11].

This paper focuses on the dynamic target tracking of the robot manipulator using visual servo method. The visual tracking control system for dynamic targets can be divided into visual state estimator and visual tracking controller in [12]. The visual state estimator can directly estimate the optimal system state and the object motion state in the image plane through the real-time observer. The visual tracking controller generates the robot's control command directly in the image plane. This design is also adopted in [13], [14], an observer is designed to estimate the position and velocity of the image feature points, and a tracking controller is designed to track the object trajectory using robot manipulator with uncertain model parameters.

In fact, considering the design of both state estimators and tracking controllers is not limited to the study on visual tracking of dynamic target, but it comes from the work on the design of state estimators and controllers for uncertain control variables in more common mechanical systems. Such as in [15], for the tracking problem of the flexible joint manipulator, a composite observer is designed to estimate

The associate editor coordinating the review of this manuscript and approving it for publication was Saeid Nahavandi .

the fault and disturbance, and the sliding mode controller is designed for the fault-tolerant control. And as in [16], for the vibration of space manipulators for on-orbit service, neural network controller and disturbance are designed to suppress the influence of friction and dynamic coupling on joint control performance.

In the study of visual state estimation, the basic problem is to consider the effect of sensor measurement noise on the extraction of visual features. In [17], for visual tracking of micro-agents in minimally invasive surgery, Kalman state estimators are used to estimate the inter-sample states of micro-agents due to the low acquisition rate of medical imaging modalities. In [18], the extended Kalman filter (EKF) is used to estimate the object velocity state with visual measurement noise, which is applied to grasp a tumbling satellite. Reference [19] is also applied to the satellite tracking, where Kalman filter can not only estimate the motion state of the grapple fixture, but also estimate the inertial parameters of the satellite. In the above studies, Kalman filter is adopted as the state estimator, and of course, extended state observer (ESO) is also commonly used [20].

The hot topic of visual tracking is how to track the target features in the case of occlusions, feature losses and limited camera field of view (FOV). In the case of sensor failure (unreliable data) or measurement of feature loss, it is common to use additional sensors as compensation, such as laser rangefinder scanning in mobile robot research [21]. It requires designing a state observer to estimate unmeasurable visual features. For the case of the feature occlusion, [22] summarizes it as visual intermittent measurement problem. It discusses the convergence condition on the dwell time, the minimum time of object visible and maximum time of outside the camera FOV. The results ensure to the image state estimator convergence to reasonable bound under visual intermittent measurement. On the basis of [22], [23] designs an observer and predictor for the image feature outside the camera FOV. When the image features are unavailable, the target motion state can be estimated by switching system between the observer and the predictor. Similarly, the observers and predictors designed by [24] can estimate the position and orientation of 3D moving targets.

On the other hand, for the design of the tracking controller, it needs to consider the parameter uncertainties, which include robot dynamics model, grasp model and camera model. In view of the dynamics parameters and grasp model parameters, it often uses adaptive neural network method to compensate parameter uncertainty. In [25], multi-layer neural networks are used to approximate unknown nonlinear parameter dynamics of autonomous underwater vehicles (AUVs), adaptive robust control are adopted to compensate for environmental disturbances. Reference [26] uses the adaptive neural networks to approximate the unknown nonlinear dynamics of robot manipulator with dead-zone input. Reference [27] develops adaptive neural network-based visual tracking method to solve the problems of an uncertain object grasp position for dual-arm manipulation. The visual

tracking task needs to determine the desired image features in advance, [28] discusses the problem of visual tracking controller without desired image features. The reference frame is defined by visual targets and planar motion constraint, and the pose estimation algorithm is designed for mobile robot.

In addition, inaccurate calibration of camera model parameters will reduce the performance of visual tracking. In [29], an adaptive visual servo tracking controller is designed to compensate unknown external parameters and visual feature depth when the parameters completely unknown. The camera model parameters and depth of image feature are included in the image Jacobian matrix. When there are uncertain for a few parameters, the uncertain parameters are linearly extracted from image Jacobian. The adaptive law can be used to estimate parameters for the performance of the visual tracking system. Reference [30] focuses on the robot visual tracking for working on an unknown constraint surface, where the depth of image feature point is the unknown time-varying. The depth-independent interaction matrix framework and the depth parameter adaptive laws are developed to compensate the unknown depth. In [31], the visual tracking tasks are considered as a nonlinear optimization problem. The depth-independent interaction matrix is used for the linearly extraction of unknown camera model parameters, the model predictive control (MPC) is used to estimate parameters. Furthermore, on the basis of [31], [32] introduces visual constraints to the predictive control methods for avoid the visual features outside the camera FOV.

In this paper, we focus on the problem of visual state estimation and tracking control for a dynamic target. Adaptive fading Kalman filter (AFKF) algorithm is used for the state estimation under the case of image feature temporary occlusion. The forgetting factor is computed to adjust the weight of the image observation data to improve the visual state estimation accuracy. When the target feature lost, the pre-order observation sequence is used to determine the forgetting factor for estimating the missing visual states. Compared with [22] and [23], the proposed method solves the tracking problem in the case of temporary and complete occlusion of image features by solving one parameter (the forgetting factor) without additional design of trajectory predictor. The advantage of this method is convenient for applying to a real robot platform. For the sake of the design of visual tracking controller, based on the depth-independent interaction matrix in [30] and [31], we consider to design the parameter adaptive law including projection error compensation to further improve the estimation accuracy of camera model parameters.

This paper is organized as follows: Section II discusses the design of visual state estimator, where a new computation method of the forgetting factor is proposed in Section II-B. The design of visual tracking controller is presented in Section III, which provides a new adaptive update law for uncalibrated camera parameters in Section III-B. In Section IV and V, simulation and experiment results are presented to demonstrate the effectiveness of the proposed

control method. Finally, the conclusion and future work are provided in Section VI.

## II. VISUAL STATE ESTIMATION OF DYNAMIC TARGET

Since the trajectories of a moving target are unknown, an observer is needed to estimate its motion state. The image position information of the target feature point can be obtained directly through the camera, and its velocity in the image plane can be calculated. However, the differential processing of image position introduces noise, then Kalman filter is used to reduce the noise and improve the accuracy of state estimation.

### A. AFKF ALGORITHM

The normal Kalman filter is a linear Gaussian filtering algorithm based on the state space, which relies on accurate model parameters. However, in practical applications, the model parameters often deviates from the real system to a certain extent, thus the normal Kalman filter cannot guarantee the filter convergence. Kalman filter adopts iterative algorithm, where the state estimation at certain instant is influenced by all observation data [33]. Adaptive fading Kalman Filter (AFKF) uses the forgetting factor to adjust the weights of all observation data in real-time, to improve the utilization of the new observation data, and to reduce the influence of old observation data to state estimation.

Let the Cartesian trajectory of the target point  $\mathbf{o}$  be  $(\mathbf{x}_o(t), \dot{\mathbf{x}}_o(t)) \in \mathbf{R}^3$ , and the corresponding image trajectory be  $(\mathbf{y}_o(t), \dot{\mathbf{y}}_o(t)) \in \mathbf{R}^2$ . AFKF is used to estimate the unknown motion state of the moving target in the image plane, which includes the position, velocity of the feature point.

The state vector of image feature is formed as  $\mathbf{O}$ ,

$$\mathbf{O} = [u_o \ v_o \ \dot{u}_o \ \dot{v}_o]^T \quad (1)$$

where  $\mathbf{y}_o(t) = [u_o \ v_o]^T$  denotes the image coordinate of the feature point  $\mathbf{o}$ ;  $\dot{\mathbf{y}}_o(t) = [\dot{u}_o \ \dot{v}_o]^T$  denotes the velocity in image plane, respectively.

The discrete state equation and observation equation of visual tracking system are given,

$$\mathbf{O}_k = \mathbf{A}\mathbf{O}_{k-1} + \mathbf{G}_{k-1}\mathbf{W}_{k-1} \quad (2)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{O}_k + \mathbf{V}_k \quad (3)$$

where at the instant of  $k$ ,  $\mathbf{y}_k$  is the measurement vector in visual space,  $\mathbf{A}$  and  $\mathbf{C}$  are the state transition matrix and observation matrix, respectively.  $\mathbf{W}_k$  and  $\mathbf{V}_k$  are the sequences of state and measurement noise.

The adaptive fading Kalman filter algorithm (AFKF) is given as follows [33],

$$\bar{\mathbf{O}}_k = \mathbf{A}\bar{\mathbf{O}}_{k-1} \quad (4)$$

$$\tilde{\mathbf{P}}_k = \alpha_k (\mathbf{A}\mathbf{P}_{k-1}\mathbf{A}^T + \mathbf{Q}_{k-1}) \quad (5)$$

$$\tilde{\mathbf{K}}_k = \frac{\alpha_k}{\beta_k} \mathbf{K}_k \quad (6)$$

$$\hat{\mathbf{O}}_k = \bar{\mathbf{O}}_k + \tilde{\mathbf{K}}_k \mathbf{r}_k \quad (7)$$

$$\mathbf{P}_k = \tilde{\mathbf{P}}_k - \tilde{\mathbf{K}}_k \mathbf{C}\tilde{\mathbf{P}}_k \quad (8)$$

Compared with the normal Kalman filter, the forgetting factor  $\alpha_k$  is introduced into the propagation of  $\tilde{\mathbf{P}}_k$  in (5), which is the covariance between predicted and real value, and the update of gain  $\tilde{\mathbf{K}}_k$  in (6), so as to update the predicted  $\bar{\mathbf{O}}_k$  and estimated  $\hat{\mathbf{O}}_k$  of state vector, and the error covariance  $\mathbf{P}_k$  between estimated and real value. Since the predicted error covariance matrix is enlarged by  $\alpha_k$  times, the utilization ratio of the new measurement data is improved. The forgetting factor  $\alpha_k$  is defined by (9)

$$\alpha_k = \frac{\text{trace}(\beta_k \mathbf{C}_k \mathbf{P}_k \mathbf{C}_k^T + (\beta_k - 1) \mathbf{R}_k)}{\text{trace}(\mathbf{C}_k \mathbf{P}_k \mathbf{C}_k^T)}, \quad \alpha_k \geq 1 \quad (9)$$

where  $\beta_k$  is a scalar factor, and  $\mathbf{P}_k$  is the time propagation error covariance of the normal Kalman filter. To compute the  $\alpha_k$ , it needs to compute the  $\beta_k$ .

$$\beta_k = \max \left\{ 1, \frac{1}{m} \text{trace}(\tilde{\mathbf{S}}_k, \mathbf{S}_k^T) \right\} \quad (10)$$

where  $\mathbf{S}_k$  and  $\tilde{\mathbf{S}}_k$  denote the covariance of the residual and its estimation respectively, and they are calculated by the following formula,

$$\mathbf{S}_k = \tilde{\mathbf{C}}\tilde{\mathbf{P}}_k \mathbf{C}^T + \mathbf{R}_k \quad (11)$$

$$\tilde{\mathbf{S}}_k = \beta_k \mathbf{S}_k = \frac{1}{k-1} \sum_{i=1}^k \mathbf{r}_i \mathbf{r}_i^T \quad (12)$$

where  $\mathbf{r}_i$  is the residual, and  $\mathbf{r}_i = \mathbf{y}_i - \mathbf{C}\bar{\mathbf{O}}_i$ .

In AFKF algorithm, the estimation of the residual covariance  $\tilde{\mathbf{S}}_k$  is used to calculate the forgetting factor  $\alpha_k$ , and  $\alpha_k$  is introduced into the predicted error covariance matrix  $\tilde{\mathbf{P}}_k$ , which can increase the weights of new observation data applied in the filtering process and improve the accuracy of state estimation. When the gain  $\mathbf{K}_k$  is ideal,  $\mathbf{r}_k$  is the white noise vector, and its auto-correlation is,

$$E(\mathbf{r}_{k+j} \mathbf{r}_k^T) = \mathbf{C}\mathbf{A}(\mathbf{I} - \mathbf{K}_{k+j-1}\mathbf{C})\mathbf{A}(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{C}) \times \mathbf{A}[\mathbf{P}_k \mathbf{C}^T - \mathbf{K}_k \mathbf{S}_k] \quad (13)$$

When the model parameters are inaccurate, the auto-correlation of  $\mathbf{r}_k$  is not zero. Then, the estimated of the residual covariance at  $k$  is,

$$\tilde{\mathbf{S}}_k = \beta_k (\tilde{\mathbf{C}}\tilde{\mathbf{P}}_k \mathbf{C}^T + \mathbf{R}_k) = \alpha_k \tilde{\mathbf{C}}\tilde{\mathbf{P}}_k \mathbf{C}^T + \mathbf{R}_k \quad (14)$$

The covariance of the residual represents the role of the current error to the system. When the dynamic model is unknown or partially unknown, the covariance of the residual and predicted error will increase due to the action of the unknown part. It can be seen from (14) that the increase of the residual covariance due to  $\beta_k$  can be considered as the increase of the predicted error covariance due to  $\alpha_k$ . Therefore, the incomplete data in the dynamic systems can be compensated by increasing  $\tilde{\mathbf{P}}_k$ . Since  $\beta_k \geq 1$ , then  $\alpha_k \geq 1$ .

When the increase of  $\bar{\mathbf{P}}_k$  is small, the value  $(1 - \beta_k)$  is small, then  $(1 - \beta_k) \mathbf{R}_k$  can be ignored; When the increase of  $\bar{\mathbf{P}}_k$  is greater than  $\mathbf{R}_k$ , it can consider  $(1 - \beta_k) \mathbf{R}_k$  ignored due to  $\mathbf{R}_k$  is small. Thus,  $\alpha_k$  is approximately equal to  $\beta_k$ . Then  $\hat{\mathbf{P}}(k) = \beta(k) \bar{\mathbf{P}}(k)$ , Therefore, this algorithm is called AFKF based on adjusting the covariance of predicted error.

According to the updated state vector  $\hat{\mathbf{O}}_k$ , the position and velocity trajectories of the target point in the image space are obtained,

$$\begin{aligned} \mathbf{y}_d(t) &= [u_o(t) \quad v_o(t)]^T \\ \dot{\mathbf{y}}_d(t) &= [\dot{u}_o(t) \quad \dot{v}_o(t)]^T \end{aligned} \quad (15)$$

$\mathbf{y}_d(t)$  and  $\dot{\mathbf{y}}_d(t)$  will be input into the visual controller in Section III as the desired trajectories of the target point.

### B. COMPUTATION OF THE FORGETTING FACTOR

In Section II-A, we reviewed the AFKF algorithm, whose core is the computation of the forgetting factor. Its computation method is formulated by the specific task. In this study, the above AFKF algorithm is adopted, and a special computation method for the visual tracking task under the target occlusion is proposed. Specifically, we will explain how to compute the forgetting factor without the image observation data.

When the image features of target point are obscured, the AFKF algorithm will fail due to the lack of image observation data. The general strategy is to adopt the linear prediction method to approximate a linear movement for the target over the invisible time, i.e. assuming the velocity is constant. However, when the velocity changes greatly, it cannot achieve the target tracking effectively under this assumption. We use the preorder observation sequence to estimate the target state in the invisible stage to improve the tracking performance.

Assuming the image feature is obscured at  $k$  instant, the state vector cannot be obtained. At this time, the observation equation under the case of image feature occlusion is given,

$$\hat{\mathbf{y}}_k = \mathbf{C}\hat{\mathbf{O}}_k + \mathbf{Y}_k \quad (16)$$

where  $\hat{\mathbf{y}}_k$  denote the prediction of the missing observation vector,  $\hat{\mathbf{O}}_k$  denote the prediction of state vector. According to (16), there is the covariance of an observation noise sequence  $\mathbf{R}_k$  between the observation and real value at  $k$  instant.  $\mathbf{Y}_k$  is not the real observation noise sequence, but it is the compensation noise, which approximates the real observation vector by introducing the covariance  $\mathbf{R}_k$ . Since there is an estimation error with the covariance  $\mathbf{P}_k$  between the estimated state vector  $\hat{\mathbf{O}}_k$  and the real state vector  $\mathbf{O}_k$ , the covariance of the observation noise sequence can be written as

$$\mathbf{U}_k = \mathbf{R}_k + \mathbf{G}_k^T \mathbf{P}_k \mathbf{G}_k \quad (17)$$

Since the current observation state of  $\mathbf{y}_k$  is unknown, its covariance  $\mathbf{P}_k$  is also unknown, which makes  $\mathbf{U}_k$  imponderable. We use preorder observation sequence  $\mathbf{y}_{k-1}$  to

replace  $\mathbf{y}_k$ , and its error covariance  $\mathbf{P}_{k-1}$  to replace  $\mathbf{P}_k$ . The approximate error covariance matrix of the observation noise sequence at  $k$  instant is obtained as

$$\hat{\mathbf{U}}_k = \mathbf{R}_k + \mathbf{G}_k^T \mathbf{P}_{k-1} \mathbf{G}_k \quad (18)$$

By (16), the new residual can be calculated by follows,

$$\hat{\mathbf{r}}_k = \mathbf{C}\hat{\mathbf{O}}_k + \mathbf{Y}_k - \mathbf{C}\bar{\mathbf{O}}_k \quad (19)$$

Considering (18), AFKF update process under occlusion is,

$$\hat{\mathbf{S}}_k = \mathbf{C}\tilde{\mathbf{P}}_k \mathbf{C}^T + \hat{\mathbf{U}}_k + \mathbf{G}_k^T \mathbf{P}_{k-1} \mathbf{G}_k \quad (20)$$

$$\hat{\mathbf{K}}_k = \tilde{\mathbf{P}}_k \mathbf{C}^T \hat{\mathbf{S}}_k^{-1} \quad (21)$$

$$\hat{\mathbf{O}}_k = \bar{\mathbf{O}}_k + [\mathbf{I}_6 - \hat{\mathbf{K}}_k \mathbf{C}]^{-1} \hat{\mathbf{K}}_k \mathbf{Y}_k \quad (22)$$

Substitute (22) into (19), and the new residual is obtained,

$$\hat{\mathbf{r}}_k = \mathbf{C} [\mathbf{I}_6 - \hat{\mathbf{K}}_k \mathbf{C}]^{-1} \hat{\mathbf{K}}_k \mathbf{Y}_k + \mathbf{Y}_k \quad (23)$$

According to the new residual  $\hat{\mathbf{r}}_k$  and the covariance of new residual  $\hat{\mathbf{S}}_k$ , the forgetting factor  $\alpha_k$  can be computed by (9) and (10). Therefore, under the case of image feature occlusion, the state vector  $\mathbf{O}_k$  can keep updating due to the computation of the forgetting factor in AFKF.

## III. UNCALIBRATED VISUAL TRACKING CONTROLLER DESIGN

Inaccurate calibration of camera model parameters will reduce the performance of visual tracking. Therefore, the uncertainty of camera model parameters should be considered in the design of tracking controller. In this section, Image-based visual servoing method (IBVS) method [29] is adopted, where the image Jacobian matrix is linearized, which contains camera internal and external parameters, and the parameter adaptive law is designed to solve the parameter uncertainty of the camera model.

### A. MODEL OF DEPTH-INDEPENDENT INTERACTION MATRIX

The image Jacobian matrix contains the depth parameters of the target feature points, as well as the camera internal and external parameters. Since the depth parameters appear in the form of nonlinearity, the camera model parameters cannot be linearized. This model of depth-independent interaction matrix is introduced to extract depth parameters, and the inaccurate camera parameters could be linearized.

Firstly, the camera perspective projection model is showed, which represents the mapping between the feature point coordinates in the image plane and Cartesian space.

$$\mathbf{y}(t) = \frac{1}{z^c(t)} \mathbf{F} \mathbf{x}(t) \quad \text{with} \quad z^c(t) = \mathbf{m}_3 \mathbf{x}(t) \quad (24)$$

where  $\mathbf{y}(t)$  and  $\mathbf{x}(t)$  are the image coordinate and Cartesian coordinate relative to robot base coordinate system, and  $z^c(t)$  is the depth of the feature point in the camera coordinate system.  $\mathbf{F}$  is the first two rows of the projection



matrix  $\mathbf{M}$ , and  $\mathbf{m}_3$  is the third row. The derivative of (24) is given,

$$\dot{\mathbf{y}}(t) z^c(t) + \mathbf{y}(t) \dot{z}^c(t) = \mathbf{F} \dot{\mathbf{x}}(t) \text{ with } \dot{z}^c(t) = \mathbf{m}_3 \dot{\mathbf{x}}(t) \quad (25)$$

Considering the kinematic constraint of robot arm, (25) can be rewritten as follows,

$$\dot{\mathbf{y}}(t) = \left[ \frac{1}{z^c(t)} \mathbf{N}(\mathbf{y}(t)) \right] \mathbf{J}(\mathbf{q}(t)) \dot{\mathbf{q}}(t)$$

with  $\mathbf{N}(\mathbf{y}(t)) = \mathbf{F} - \mathbf{y}(t) \mathbf{m}_3 = \begin{bmatrix} \mathbf{m}_1 - u(t) \mathbf{m}_3 \\ \mathbf{m}_2 - v(t) \mathbf{m}_3 \end{bmatrix} \quad (26)$

In (26),  $\mathbf{N}(\mathbf{y}(t)) \in \mathbf{R}^{2 \times 4}$  is defined as the depth-independent interaction matrix, and it satisfies the linearization condition, i.e. for any vector  $\boldsymbol{\zeta} \in \mathbf{R}^{4 \times 1}$ ,  $\mathbf{N}(\mathbf{y}(t)) \boldsymbol{\zeta}$  can be linearly parameterized as,

$$\mathbf{N}(\mathbf{y}(t)) \boldsymbol{\zeta} = \mathbf{B}(\mathbf{y}(t), \boldsymbol{\zeta}) \boldsymbol{\theta} \quad (27)$$

where  $\mathbf{B}(\mathbf{y}(t), \boldsymbol{\zeta})$  is the regression matrix, which is independent of camera parameters;  $\boldsymbol{\theta}$  is the parameter vector, which is composed of unknown parameters in the projection matrix  $\mathbf{M}$ . Since the depth  $z^c(t)$  is nonlinear, it needs to separate  $z^c(t)$  from (26) for extracting  $\boldsymbol{\theta}$  linearized. The depth-independent Jacobian matrix  $\mathbf{D}(t) \in \mathbf{R}^{2 \times 3}$  is defined,

$$\mathbf{D}(t) = \mathbf{N}(\mathbf{y}(t)) \mathbf{J}(\mathbf{q}(t)) \quad (28)$$

Then, (26) can be rewritten as,

$$z^c(t) \dot{\mathbf{y}}(t) = \mathbf{D}(t) \dot{\mathbf{q}}(t) \quad (29)$$

The unknown parameter vector  $\boldsymbol{\theta}$  is included in  $\mathbf{D}(t)$ , and its values are needed to designing an adaptive laws to determine, so as to obtain the joint control commands of robot arm.

### B. DESIGN OF ADAPTIVE LAW FOR CAMERA PARAMETERS

When the camera model parameters are uncertain, the elements of the projection matrix  $\mathbf{M}$  are uncertain. In this section, a new adaptive update law for uncertain camera parameters is proposed, which introduces the compensation of projection error estimation. Let the parameter vector be  $\boldsymbol{\theta} = [m_{ij}] \in \mathbf{R}^{12 \times 1}$ ,  $i \in [1, 3], j \in [1, 4]$ . The tracking error of the feature points in the image space is given as follows,

$$\Delta \mathbf{y}(t) = \mathbf{y}(t) - \mathbf{y}_d(t) \quad (30)$$

$$\Delta \dot{\mathbf{y}}(t) = \dot{\mathbf{y}}(t) - \dot{\mathbf{y}}_d(t) \quad (31)$$

$$\dot{\mathbf{y}}_r(t) = \dot{\mathbf{y}}_d(t) - \kappa \Delta \mathbf{y}(t) \quad (32)$$

where the desired image trajectories  $\mathbf{y}_d(t)$  and  $\dot{\mathbf{y}}_d(t)$  are obtained by (15). We defines the reference velocity in the image plane  $\dot{\mathbf{y}}_r(t)$  in (32), which is obtained from the position tracking error  $\Delta \mathbf{y}(t)$  and the desired velocity trajectory  $\dot{\mathbf{y}}_d(t)$ , and  $\kappa$  is the positive constant. For (31), we introduce

the estimation of the velocity tracking error  $\Delta \hat{\mathbf{y}}(t)$  in the image plane,

$$\Delta \hat{\mathbf{y}}(t) = \hat{\mathbf{y}}(t) - \dot{\mathbf{y}}_d(t) \quad (33)$$

The adaptive sliding mode vector  $\mathbf{s}_y(t)$  in the image plane is defined as follows,

$$\begin{aligned} \mathbf{s}_y(t) &= \hat{\mathbf{y}}(t) - \dot{\mathbf{y}}_r(t) = \hat{\mathbf{y}}(t) - \dot{\mathbf{y}}_d(t) + \kappa (\mathbf{y}(t) - \mathbf{y}_d(t)) \\ &= \Delta \hat{\mathbf{y}}(t) + \kappa \Delta \mathbf{y}(t) \end{aligned} \quad (34)$$

where  $\hat{\mathbf{y}}(t)$  denote the estimation of the image velocity trajectory, which is calculated by the estimation of projection matrix  $\hat{\mathbf{M}}(t)$ . By (29), the estimation of the depth-independent Jacobian matrix  $\hat{\mathbf{D}}^+$  is used to map the reference velocity  $\dot{\mathbf{y}}_r(t)$  to the joint space of the robot arm, then the reference in the joint space is given,

$$\dot{\mathbf{q}}_r(t) = \hat{\mathbf{D}}^+(t) \hat{z}(t) \dot{\mathbf{y}}_r(t) \quad (35)$$

where  $\hat{\mathbf{D}}^+(t) \in \mathbf{R}^{7 \times 2}$  is the pseudo-inverse of  $\mathbf{D}(t) \in \mathbf{R}^{2 \times 3}$ , and  $\hat{\mathbf{D}}^+(t) = \hat{\mathbf{D}}^T(t) \left\{ \hat{\mathbf{D}}(t) \hat{\mathbf{D}}^T(t) \right\}^{-1}$ . Similarly, the adaptive sliding mode vector  $\mathbf{s}_q(t)$  in the joint space is defined as follows,

$$\mathbf{s}_q(t) = \dot{\mathbf{q}}(t) - \dot{\mathbf{q}}_r(t) \quad (36)$$

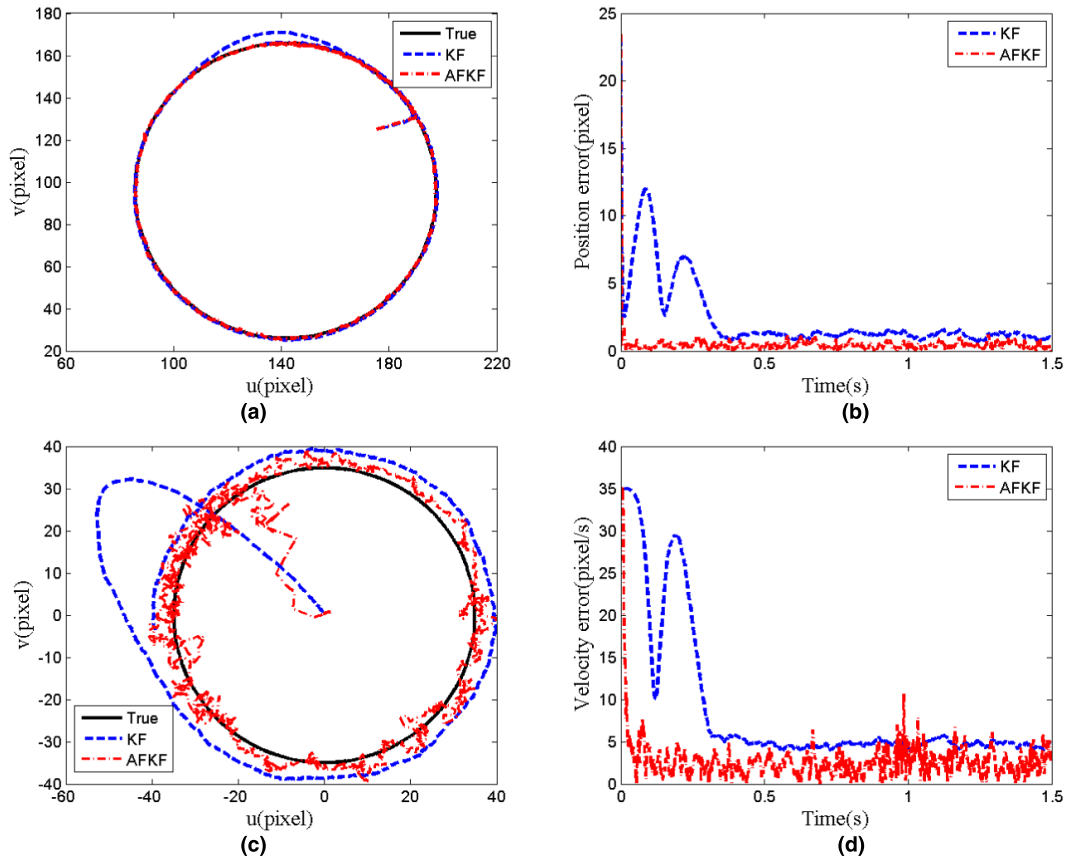
Since  $\hat{\mathbf{D}}(t)$  and  $\hat{z}(t)$  are obtained by  $\hat{\boldsymbol{\theta}}(t)$ , which is linearly extracted from  $\hat{\mathbf{M}}$ . In order to ensure the existence of  $\hat{\mathbf{D}}^+(t)$ ,  $\hat{\mathbf{M}}(t) \in \mathbf{R}^{3 \times 4}$  must be full rank, i.e. the rank of  $\hat{\mathbf{M}}(t)$  is 3. Here a potential field force  $\Psi(\hat{\boldsymbol{\theta}}(t))$  is introduced to keep the  $\hat{\mathbf{M}}(t)$  full rank. The negative gradient function  $\partial \Psi(\hat{\boldsymbol{\theta}}(t)) / \partial \hat{\boldsymbol{\theta}}(t)$  is used to update the  $\hat{\boldsymbol{\theta}}(t)$  to keep it away from the singularity region of  $\hat{\mathbf{M}}(t)$ . It also guarantees the existence of  $\hat{\mathbf{D}}^+(t)$ , referred to [31] for details.

Next, we define the estimation error of the projection matrix, that is  $\|\hat{\mathbf{M}}(t) - \mathbf{M}(t)\|$ . By (24) and (27), the error is given,

$$\begin{aligned} \mathbf{e}_1(t) &= \hat{z}(t) \mathbf{y}(t) - \hat{\mathbf{F}}(t) \mathbf{x}(t) \\ &= \{\hat{\mathbf{m}}_3(t) - \mathbf{m}_3(t)\} \mathbf{x}(t) \mathbf{y}(t) - \{\hat{\mathbf{F}}(t) - \mathbf{F}\} \mathbf{x}(t) \\ &= \boldsymbol{\Xi}(\mathbf{x}(t), \mathbf{y}(t)) \Delta \boldsymbol{\theta}(t) \end{aligned} \quad (37)$$

where  $\|\hat{\mathbf{m}}_3(t) - \mathbf{m}_3(t)\|$  is the estimation error of the third row of  $\hat{\mathbf{M}}(t)$ ,  $\|\hat{\mathbf{F}}(t) - \mathbf{F}\|$  is the estimation error of the first two rows.  $\Delta \boldsymbol{\theta}(t) = \hat{\boldsymbol{\theta}}(t) - \boldsymbol{\theta}$  is the updated error of  $\boldsymbol{\theta}(t)$ . In order to improve the accuracy of parameter estimation, the new estimation errors of depth  $z(t)$  and depth-independent Jacobian matrix  $\mathbf{D}(t)$  are introduced as the compensation of projection error estimation. According to (29), the error is defined as,

$$\begin{aligned} \mathbf{e}_2(t) &= \left\{ (\hat{z}(t) - z(t)) \dot{\mathbf{y}}(t) - (\hat{\mathbf{D}}(t) - \mathbf{D}(t)) \dot{\mathbf{q}}(t) \right\} \mathbf{K}_3 \\ &= \boldsymbol{\Theta}(\dot{\mathbf{y}}(t), \mathbf{x}(t), \dot{\mathbf{x}}(t)) \Delta \boldsymbol{\theta}(t) \end{aligned} \quad (38)$$



**FIGURE 1.** The behavior of motion state estimation of AFKF: (a) position trajectories; (b) position error; (c) velocity trajectories; (d) velocity error.

By the regression matrix  $\Xi(\mathbf{x}(t), \mathbf{y}(t))$  and  $\Theta(\dot{\mathbf{y}}(t), \mathbf{x}(t), \dot{\mathbf{x}}(t))$ , the adaptive update law of  $\hat{\theta}(t)$  is designed as,

$$\begin{aligned} \dot{\hat{\theta}}(t) &= -\Gamma^{-1} \left\{ \begin{aligned} &\Xi^T(\mathbf{x}(t), \mathbf{y}(t))\mathbf{P}_1\mathbf{e}_1(t) + \mathbf{P}_2 \|\mathbf{s}_q(t)\|^2 \frac{\partial \Psi(\hat{\theta}(t))}{\partial \hat{\theta}(t)} \\ &+ \frac{1}{\hat{z}(t)} \Theta^T(\dot{\mathbf{y}}(t), \mathbf{x}(t), \dot{\mathbf{x}}(t))(\Delta \dot{\mathbf{y}}(t) + \kappa \Delta \mathbf{y}(t)) \end{aligned} \right\} \end{aligned} \quad (39)$$

where  $\Gamma$ ,  $\mathbf{P}_1$  and  $\mathbf{P}_2$  are the positive definite weighting matrix. According to the dynamic equation of robot arm, the visual tracking control law is designed as,

$$\begin{aligned} \boldsymbol{\tau} &= \mathbf{H}(\mathbf{q}(t))\ddot{\mathbf{q}}_r(t) + \mathbf{C}(\mathbf{q}(t), \dot{\mathbf{q}}(t))\dot{\mathbf{q}}_r(t) + \mathbf{G}(\mathbf{q}(t)) \\ &\quad - \left( \mathbf{K}_1 + \mathbf{K}_2 \left\| \frac{\partial \Psi(\hat{\theta}(t))}{\partial \hat{\theta}(t)} \right\| \right) \mathbf{s}_q(t) - \frac{1}{\hat{z}(t)} \hat{\mathbf{D}}^T(t) \mathbf{K}_3 \mathbf{s}_y(t) \end{aligned} \quad (40)$$

where  $\mathbf{K}_1$ ,  $\mathbf{K}_2$  and  $\mathbf{K}_3$  denote a symmetric positive definite matrix. In (40), the sliding mode vector  $\mathbf{s}_q(t)$  and  $\mathbf{s}_y(t)$  are regarded as the control error. The gradient function  $\partial \Psi(\hat{\theta}(t)) / \partial \hat{\theta}(t)$  is used to ensure  $\hat{\mathbf{D}}^+(t)$  the existence.

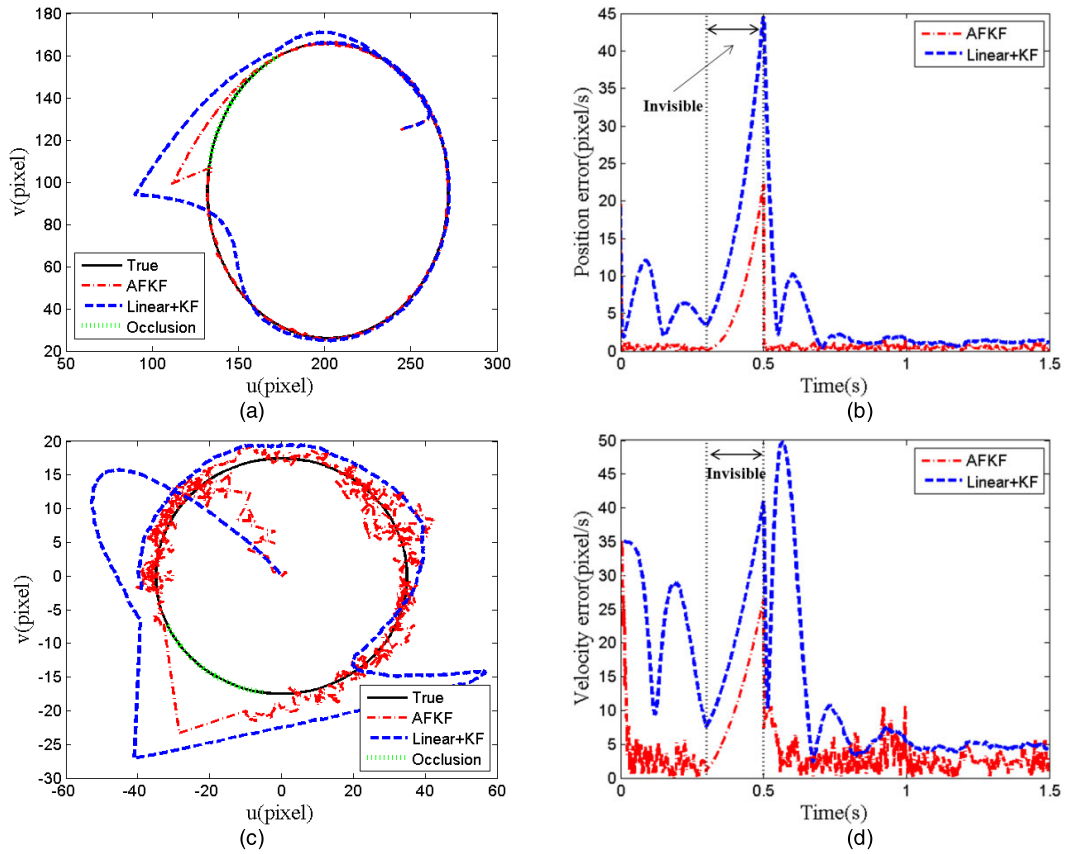
## IV. SIMULATION RESULTS

### A. THE BEHAVIOR OF VISUAL STATE ESTIMATOR

This section verifies the performance of AFKF algorithm in visual state estimation, which can be divided into two parts: the target trajectory with and without occlusion. In simulation, the origin of robot end-effector coordinate system is set as the feature point O of dynamic target. Firstly, the model parameters and initial values of robot arm are shown in Table 1, where the true trajectories  $\mathbf{O}(t)$  of the target are given for comparison. The simulation time is 1.5s, and the sampling interval is 1ms.

In order to indicate the superiority of the proposed method, the results obtained by the normal Kalman filter method are introduced for comparison, where both two methods adopted the same noise parameters. According to the desired trajectory (anticlockwise elliptic) in Table 1, the results obtained by the two methods are shown in Fig. 1. The performance of the estimated position trajectories are shown in Fig. 1(a). Since the initial position of the end-effector point is given inconsistently with the target feature point, both two trajectories move toward the target position rapidly in the initial stage.

In the process of the whole trajectory tracking, it is obvious that the KF trajectory does not coincide with the true trajectory, and there are large deviations in some stages, and the



**FIGURE 2.** The behavior of motion state prediction of AFKF under occlusion: (a) position prediction; (b) position error; (c) velocity prediction; (d) velocity error.

**TABLE 1.** Model parameters and initial values in simulation.

True trajectories	$\mathbf{O}(t) = \begin{cases} u_e(t) \\ v_e(t) \\ \dot{u}_e(t) \\ \dot{v}_e(t) \end{cases} = \begin{cases} 70\cos(5t+0.5)-148 \\ -70\sin(5t+0.5)-64 \\ 350\sin(5t+0.5) \\ -350\cos(5t+0.5) \end{cases} \quad (0 \leq t \leq 14)$
Initial value	$\mathbf{y}(0) = [63.9 \quad 94.4]^T \text{ pixel,}$ $\mathbf{x}(0) = [371 \quad -291.6 \quad -84.1 \quad 1]^T \text{ mm}$ $\mathbf{q}(0) = [-1.72 \quad -2.73 \quad -1.82 \quad 2.79 \quad 0.03 \quad -1.21 \quad -0.39]^T$ $\dot{\mathbf{q}}(0) = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T \text{ rad/s}$
Controller	$\mathbf{P}_1 = \text{diag}\{5 \times 10^{-8} \quad 5 \times 10^{-8}\}, \mathbf{\Gamma} = \text{diag}\{0.5\}_{11 \times 11}, \kappa = 200$ $\mathbf{K}_1 = \text{diag}\{0.3\}_{7 \times 7}, \mathbf{K}_3 = \text{diag}\{5 \times 10^{-7} \quad 5 \times 10^{-7}\}$
Filter	$\mathbf{Q} = \text{diag}\{1 \times 10^{-5} \quad 1 \times 10^{-5}\} \quad \mathbf{R} = \text{diag}\{5 \times 10^{-5} \quad 5 \times 10^{-5}\}$

AFKF trajectory basically coincide with the true trajectory. Fig. 1(b) shows the variation of the estimated error. The error of AFKF trajectory is affected by the large error in the initial stage with exceeding 10 pixels, and the overall tracking average error of AFKF trajectory is limited within 2 pixels. The estimated velocity trajectory is shown in Fig. 1(c). Since the differential operation is adopted which introduced the

noise, and the estimated trajectory presents an oscillating state. The AFKF trajectory moves toward the true trajectory at a practical velocity in the initial stage, and follows the desired velocity trajectory in an oscillation state. In contrast, the KF trajectory moves toward the desired trajectory at a very large velocity in the initial phase, and the maximum reaches 35 pixel/s and the average exceeds 20 pixel/s from 0s to 0.5s, which is impractical for a real applications. After 0.5s, KF trajectory follows the desired trajectory with a certain tracking error, and the error is >5 pixel/s as shown in Fig. 1(d). Although the error of AFKF trajectory reaches 35 pixel/s in the initial stage, it recovers rapidly <5 pixel/s within 0.2s, whose average error is <4 pixel/s. It is in line with a real tracking task of robot arm.

The performance of AFKF is verified in case of target feature occlusion, and the simulation results are shown as Fig. 2. Similarly, the linear prediction method and KF are used for comparison. Linear prediction is used to estimate the motion state under occlusion. The occlusion is lasted from 0.3s to 0.5s, when the observation cannot be obtained. The two methods are used to estimate the invisible motion state. As shown in Fig. 2(a), the two trajectories deviate from the true trajectory under occlusion, and the tracking error of linear + KF is much greater than that of AFKF. This is because the proposed method considers the preorder

observation sequence, and predicts the noise  $\mathbf{Y}_k$  to compensate for the estimated state, thus the deviation is not too large. When the target features become visible at 0.5s, AFKF trajectory rapidly moves toward the true trajectory, due to the update of  $\hat{\mathbf{S}}_k$ . Since the forgetting factor is instantly corrected, the propagation of  $\mathbf{P}_k$  in (5) is rapidly adjusted, and it converges to the true trajectory. After the feature is visible, the linear + KF trajectory slowly is adjusted, to move toward the true trajectory. The position tracking error of AFKF in the occlusion stage is the maximum of 20 pixels, and the average is 10 pixels. And the maximum of linear + KF is 45 pixels and the average is 20 pixels, as shown in Fig.2 (b). In comparison, the velocity tracking error of the two methods is larger than that of the position trajectory due to the larger noise of velocity estimation. As shown in Fig.2(c), the overall trend of the estimated velocity trajectory is similar to that of the position trajectory. When the target feature becomes re-visible, the estimated trajectory moves slowly toward the true trajectory, not immediately. This is also because the propagation of  $\mathbf{P}_k$  is affected by the convergence due to the influence of large noise. In the occlusion stage, the tracking error of AFKF is the maximum 25 pixel/s and the average 15 pixel/s. The maximum of KF is 40 pixel/s and the average is 22 pixel/s. It is indicated that the introduced noise  $\mathbf{Y}_k$  plays an obvious role in the occlusion stage.

**B. THE BEHAVIOR OF VISUAL TRACKING CONTROLLER**

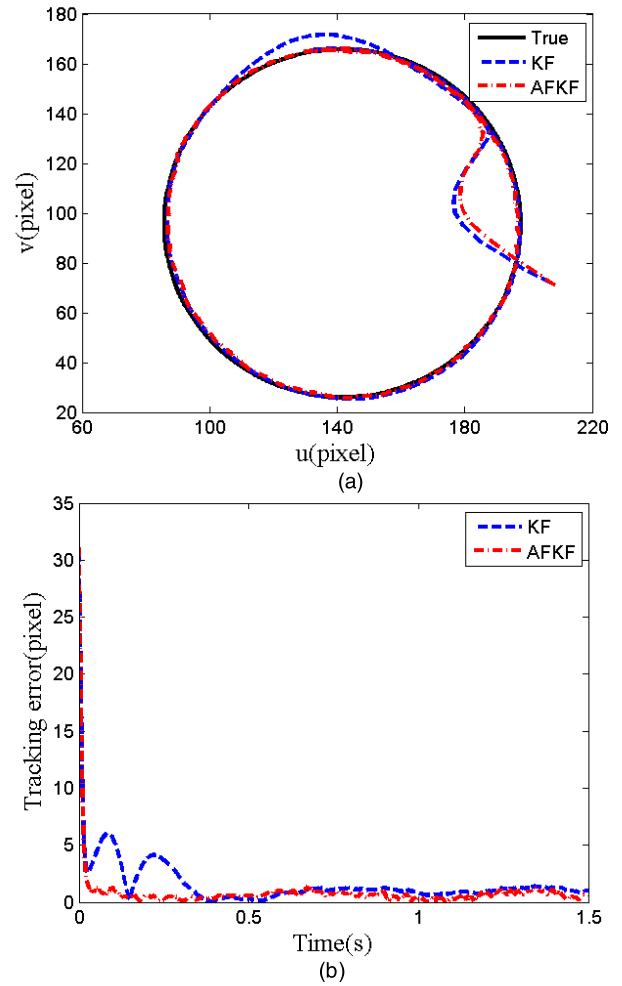
Next, the performance of the proposed uncalibrated visual tracking controller is verified. The true and estimated value of camera projection matrix at the initial time are given as follows,

$$\mathbf{M} = \begin{bmatrix} -1082 & 1662 & -1321 & 697510 \\ 1454 & 842 & -227 & -426860 \\ 0.361 & -0.476 & -0.8 & 868 \end{bmatrix},$$

$$\hat{\mathbf{M}}(0) = \begin{bmatrix} -190 & 1500 & -82 & 200000 \\ 1210 & -885 & -98 & -650000 \\ 0.1 & -0.15 & -0.98 & 1000 \end{bmatrix}$$

In simulation, the estimated trajectories in section 4.1 are used as the desired trajectories, which are obtained by AFKF and linear + KF methods. The visual tracking results without occlusion are shown in Fig. 3. Compared with the desired trajectory in Fig. 1, AFKF trajectory follows the desired trajectory with a very small tracking error. Since the estimation  $\hat{\mathbf{M}}$  of the projection matrix is used, the imprecise parameters are adjusted through the parameter adaptive law (39) in the initial stage in order to reach the tracking requirements. Thus AFKF trajectory slowly converges to the desired trajectory, as shown in Fig. 3(a). In addition, KF trajectory can also be used to follow the desired trajectory through the proposed parameter adaptive law. In the upper half of the true trajectory, KF trajectory deviates significantly, which corresponds to the estimated desired trajectory in Fig. 1(a).

Compared with the results of trajectory estimation, KF trajectory converges to the true trajectory with a smaller error, which indicates that the proposed adaptive method



**FIGURE 3.** The simulation results of uncalibrated visual tracking: (a) trajectory tracking; (b) position tracking error.

**TABLE 2.** The simulation results of tracking error and estimation error.

	Tracking error (pixel)		Position prediction (pixel)		Velocity prediction (pixel)	
	KF	AFKF	KF	AFKF	KF	AFKF
Average	3.11	1.59	2.31	0.43	85.48	25.04
Min	1.53	0.11	0.79	0.01	35.74	0.74
Max	61.9	61.9	21.31	21.31	350	350

can achieve the desired trajectory tracking through parameter update. Although there is some error between the estimated desired and true trajectory.

The tracking error is collected in Table 2, where the maximum value represents the tracking (estimated) error at the initial time, which is the moment with the maximum error in the tracking process. The average error of AFKF position trajectory is 0.43, which is nearly 80% lower than the KF error of 2.31, and the average error of velocity trajectory is also 71% lower. The results indicate that the advantage of AFKF is mainly reflected in the position estimation. Due to



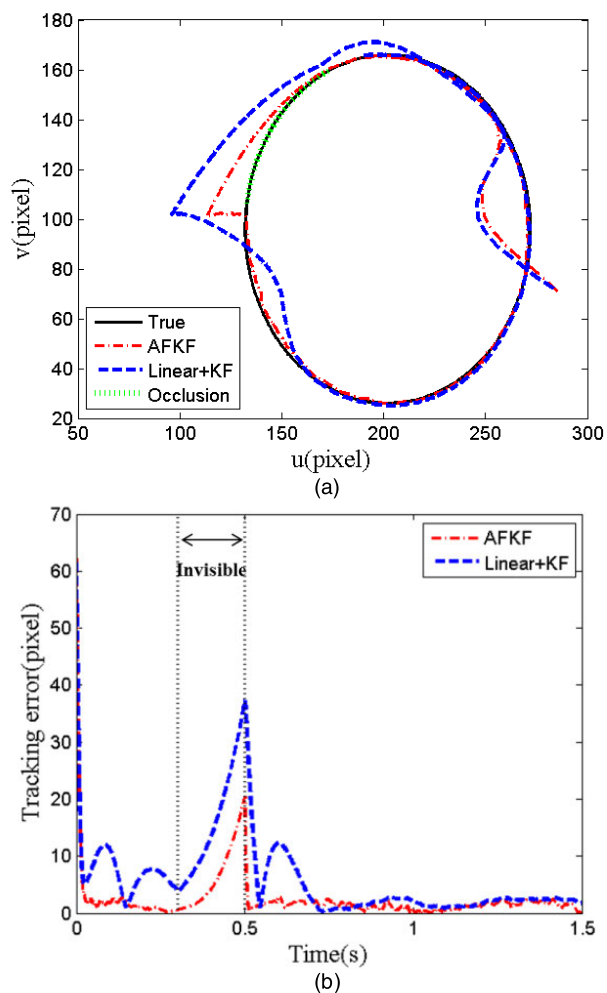


FIGURE 4. The simulation results of visual tracking with occlusion: (a) trajectory tracking; (b) position tracking error.

the oscillation in the velocity estimation, the accuracy is relatively reduced.

The tracking results in the presence of occlusion are shown in Fig. 4. Again, the overall trend of the curve is similar to that of Fig. 2. In the initial stage, the parameter adaptive law is used to adjust the parameters of the inaccurate camera model, and the tracking trajectory move toward the true trajectory. As shown in Fig. 4(a), the trajectory follows the desired trajectory in Fig. 2(a) during the occlusion stage, which indicates that the inaccurate camera parameters do not affect the actual tracking performance. When the features become visible, AFKF trajectory converges to the true trajectory with small fluctuations, which indicates that there is a certain error in the actual tracking control. In the occlusion stage, the average tracking error of AFKF trajectory is <10 pixels, and KF is >15 pixels, as shown in Fig. 4(b).

The tracking and estimation errors with occlusion are collected in Table 3. The average of AFKF position estimation is 6.46 pixels. The estimation error increases significantly compared to the results in Table 2 due to the lack of observation. Compared with the linear prediction and KF methods,

TABLE 3. The results of tracking error and prediction error of occlusion simulation.

	Tracking error (pixel)		Position prediction (pixel)		Velocity prediction (pixel)	
	Linear+KF	AFKF	Linear+KF	AFKF	Linear+KF	AFKF
Average	6.02	2.34	18.90	6.46	221.73	101.45
Min	0.598	0.036	3.34	0.26	77.59	14.39
Max	61.93	61.93	45.77	19.85	421.73	237.09

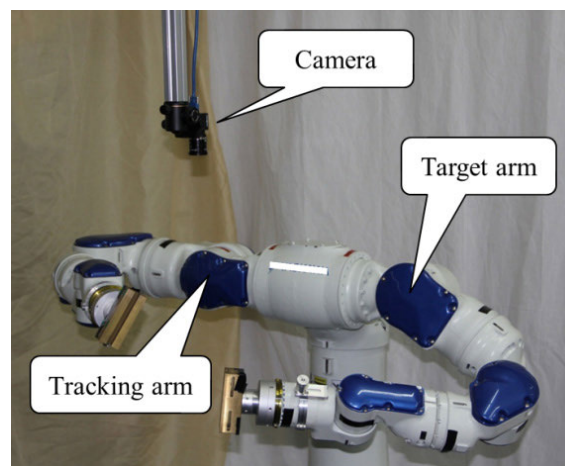


FIGURE 5. The experimental platform of the dual-arm robot.

AFKF reduces the estimation errors of position and velocity by 65.8% and 54.2%, respectively. In the occlusion stage, the estimation error will increase, and the maximum represents the deviation between the estimated and the true value when the target become visible, and it is easy to know that the trajectory estimation of AFKF has a smaller deviation from the true trajectory. In the whole process, AFKF is better to estimate the target trajectory, thus the tracking error is smaller.

### V. EXPERIMENTS AND RESULTS

In the experiment, a monocular color camera with a resolution of 800\*600 pixels is used, which is fixed above the robot’s workspace, as shown in Fig. 5. The images of the working area of the robot manipulator are acquired by the camera in real time, and the feature points pasted on the end-effector are obtained through object recognition and feature extraction. The dual-arm 7DOF robot platform is adopted, in which the left arm is used to simulate the unknown motion of the target feature (target arm), and the right arm is used to follow the motion of the target feature (tracking arm). The feature points are pasted on both the end-effectors of the target arm and the tracking arm, which are red and green squares respectively.

The image coordinates of the square center are obtained by fast image processing algorithm. The number of the feature points on the each end-effector is four, which can avoid the image Jacobian singularity [27]. The coordinate transformation matrix of each feature point relatively to the coordinate system of the end-effector is precisely calibrated in order to

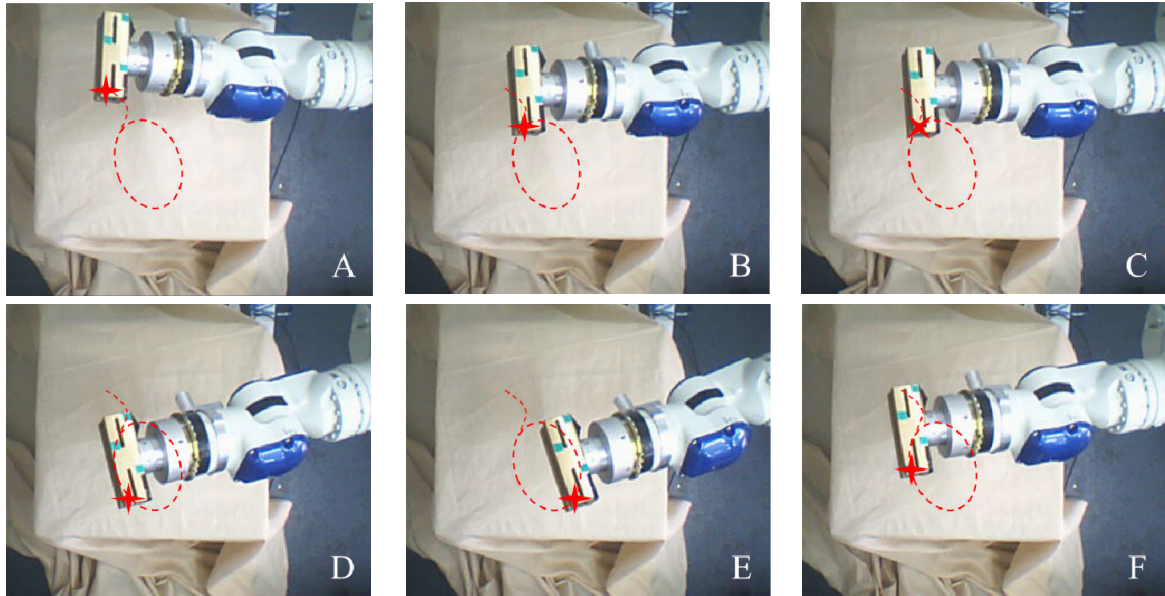


FIGURE 6. The snapshots of experiment 1.

obtain the Cartesian coordinates of the feature points through the robot kinematics.

The experiment is divided into two groups. The first group verifies the performance of the proposed uncalibrated visual tracking controller. Given the desired image trajectory, the end-effector features of the tracking arm are controlled to achieve visual tracking. The second group verifies the overall performance of visual estimator and tracking controller. The features on the target arm move with the given random image trajectory (the trajectory data is recorded as error calculation only), and the features on the end-effector of the tracking arm are controlled to following the motion of the target features.

TABLE 4. The model parameters and initial values in experiment 1.

Camera model	$\hat{\mathbf{M}}(0) = \begin{bmatrix} -1000 & 500 & 500 & 400000 \\ 500 & -500 & -500 & 64000 \\ 0 & 0 & 0 & 600 \end{bmatrix}$
Desired trajectories	$\begin{cases} \mathbf{y}_d(t) = \begin{bmatrix} -20\cos(t) - 100\sin(t) + 106 \\ 100\cos(t) + 20\sin(t) + 337 \end{bmatrix} \\ \dot{\mathbf{y}}_d(t) = \begin{bmatrix} 20\sin(t) - 100\cos(t) \\ -100\sin(t) + 20\cos(t) \end{bmatrix} \end{cases}$
Initial value	$\mathbf{q}(0) = [-2.2 \quad -0.49 \quad 2.494 \quad 0.5 \quad -2.36 \quad 0.23 \quad 1.435]^T$ rad
Controller	$\mathbf{P}_1 = \text{diag}\{2 \times 10^{-7} \quad 2 \times 10^{-7}\}, \Gamma = \text{diag}\{5\}_{11 \times 1}, \kappa = 200$ $\mathbf{K}_1 = \text{diag}\{0.02\}_{7 \times 7}, \mathbf{K}_3 = \text{diag}\{2 \times 10^{-7} \quad 2 \times 10^{-7}\}$

**A. EXPERIMENT 1: GIVEN THE KNOWN VISUAL TRAJECTORIES**

The system parameters and initial state are shown in Table 4. The estimation of the camera projection matrix at the initial

time  $\hat{\mathbf{M}}(0)$  is arbitrarily given, and its true value  $\mathbf{M}$  is unknown. The desired image trajectory of the feature point on the target arm is ellipse, and the image trajectory of one feature point is only given, since the trajectories of the other three feature points are similar, except for the difference in spatial distance.

The snapshots of experiment 1 are shown in Fig. 6. According to of the labeled desired trajectory, the tracking arm move with visual tracking controller. The path point A is the initial position, and the process is divided into two parts: adjustment stage A-C and tracking stage D-F. In the whole process, the camera parameters are continuously updated to achieve an accurate visual tracking task.

The experimental results are shown in Fig. 7. Fig.7 (a) shows the image trajectory tracking results. Due to the influence of motion state estimation error and camera measurement noise, the tracking error in experiment is larger than that in simulation, but the maximum is limited within 6 pixels, and the average is <3 pixels, as shown in Fig.7 (b). Fig.7 (c) represents the tracking trajectory in Cartesian space, and it can be seen that there is an obvious change in the depth direction (z-axis), which indicates that the proposed tracking method achieves the 3-D target tracking. However, the motion along the z-axis of the camera frame is limited to a small scale. This is because the proposed method adopts image-based visual servoing method (IBVS) of the monocular camera, which does not estimate the depth information, thus the proposed method is not suitable for the large scale motion along the z-axis. Fig.7 (d) shows the all joint trajectories, and it can be seen that the range of joint angle variation is small, which reflects the stability of the tracking process.

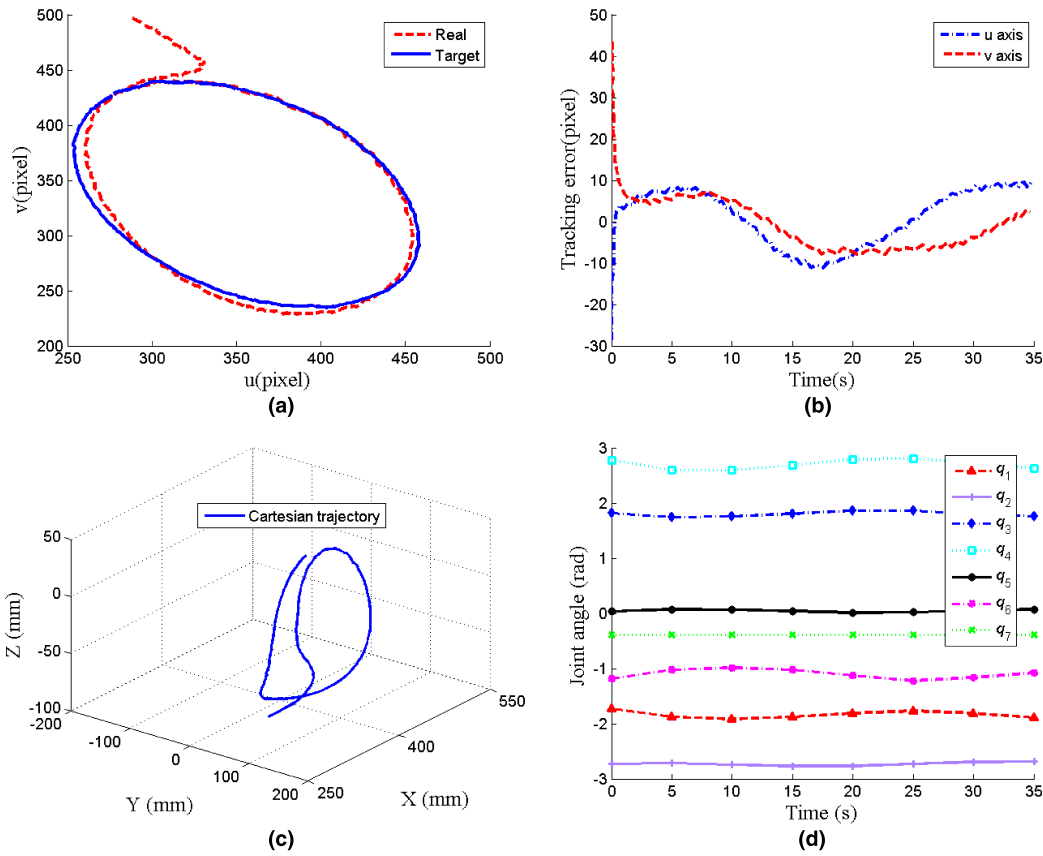


FIGURE 7. The experimental results of uncalibrated visual tracking: (a) image trajectory tracking; (b) tracking errors; (c) Cartesian trajectory; (d) joint trajectories.

**B. EXPERIMENT 2: UNKNOWN AND OCCLUDED VISUAL TRAJECTORIES**

In experiment 2, the feature trajectory of the target arm is shown in Table 5.

TABLE 5. Initial values in experiment 2.

True trajectories	$u = -1.12 \times 10^{-11} (1300 - 100t)^5 + 3.07 \times 10^{-8} (1300 - 100t)^4$
	$-3.03 \times 10^{-5} (1300 - 100t)^3 + 0.01465 (1300 - 100t)^2$
	$-4.283 (1300 - 100t) + 620$
	$v = 1300 - 100t$
Initial value	$\mathbf{q}_t(0) = [-2.29 \quad -0.64 \quad 2.698 \quad 0.243 \quad -0.2855 \quad 0.3513 \quad -0.4888]^T$ rad
	$\mathbf{q}_t(0) = [-0.2574 \quad -0.4667 \quad 0.07 \quad 0.9611 \quad -0.1 \quad -1.21 \quad -0.94]^T$ rad

Again, the image trajectory of one feature point is given. The tracking arm only moves to follow the motion estimated state of the target feature observed by camera. The occlusion of the feature points is carried out randomly. At a certain moment, a white paper is artificially covered over the target feature points, thus the feature points are completely covered. After a proper time interval, the white paper is removed and the feature points are re-visible. This time interval is determined according to the actual situation, i.e. when the relative pose between two arms reaches a certain range or the

features are close to out of the camera field of view, the white paper is immediately removed. The controller parameters are the same as in experiment 1, and the experimental snapshots are shown in Fig. 8.

In order to avoid collisions, a safe distance between the two end-effectors is required. Therefore, the visual tracking error is not zero, but is the constant image distance between the corresponding feature points. The path point **A** is the initial state, **C** is the stable tracking stage, **F** is the start of occlusion, and **H** is the re-visible stage. The process is divided into four stages: in the **A-C** stage, the motion state of the target features are estimated by AFKF, and the visual controller is used to drive the tracking arm to quickly approach the target arm. In **C-F** stage, the uncertain camera parameters are updated to achieve accurate target tracking. In **F-H** stage, the features are not visible, and AFKF estimates the motion state and guides the tracking arm to move. Although the tracking arm deviates significantly from the target trajectory, the general trend remains consistent with the target trajectory, which verifies the effectiveness of AFKF trajectory prediction. In **H-I** stage, the target is re-visible, and the motion state estimated by AFKF becomes accurate, and the tracking arm again rapidly approaches the target trajectory.

The experimental results are shown in Fig.9 (a) and (b), which show the results of image trajectory tracking and error respectively. The experiment lasted closely to 40s.

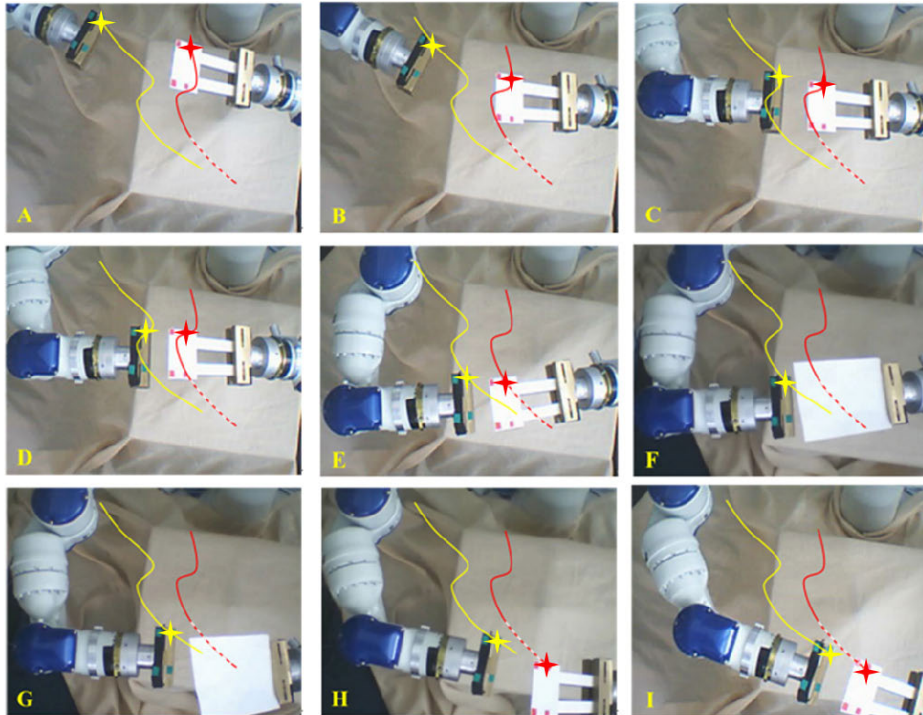


FIGURE 8. The snapshots of experiment 2.

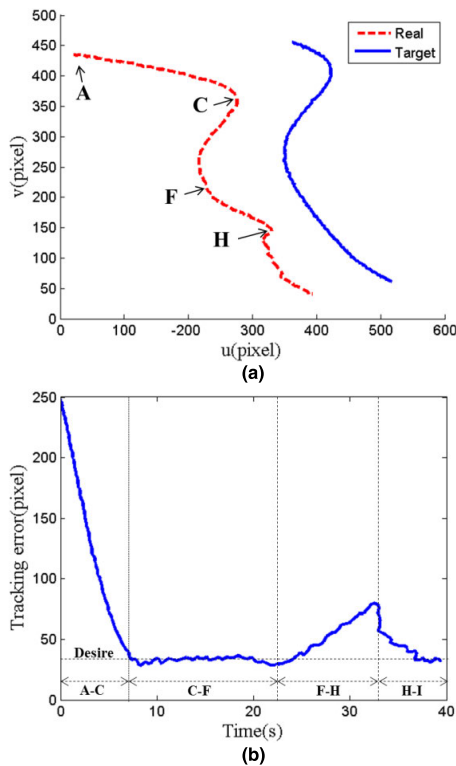


FIGURE 9. The experimental results of visual tracking with occlusion: (a) trajectory tracking; (b) position tracking error.

At the initial state A, the image distance between the two groups of feature points is more than 200 pixels. In A-C stage (0-8s), the tracking arm moves toward the target at 25 pixels per second. In C-F stage (8s-22s), the tracking arm

performs stable tracking with an error within 2 pixels. In F-H stage (22s-33s), the average tracking error is <9 pixels when the target is not visible. In H-I stage (33s-38s), the target is re-visible, and the tracking arm moves toward the target at >10 pixels per second, and finally the tracking error is within 2 pixels in 3s.

TABLE 6. The results of tracking error and estimation error of occlusion experiment.

	Stable tracking C-F (pixel)	Position estimation C-F (pixel)	Occlusion tracking F-H (pixel)	Position prediction F-H (pixel)	Velocity prediction F-H (pixel)
Average	2.76	1.86	14.05	5.37	12.62
Min	1.29	1.02	3.36	2.72	6.29
Max	4.14	3.58	35.84	16.58	30.45

We further analyze the stable tracking stage C-F and occlusion stage F-H. The results are shown in Table 6. In C-F stage, the average error is 2.76 pixels, which comes from many sources. The main source is the position estimation AFKF (1.86 pixels), and the rest is from the measurement and camera parameter update error, which is less than 1 pixel. The maximum denotes the error when the features become not visible. Generally, the range of the tracking error with <3 pixels can be applied to actual robot applications. In F-H stage of the invisible feature, the average error is 14.05 pixels, where the errors from the position and velocity predictions of AFKF contribute 5.37 and 12.62 pixels respectively, and the time of occlusion increase with the maximum (H point) reaching 16.58 and 30.45 pixels. In general, during the



occlusion 11s, the movement distance of the target feature is more than 50 pixels, and the final error is about 8 pixels, which indicates that the proposed method is more suitable for the situations of the feature occluded with short time.

## VI. CONCLUSION

In this paper, an AFKF-based visual tracking method of moving targets for robot manipulation is developed to solve the problems of the visual state estimation under occlusion and visual servoing with parameter uncertainty. The computation method of the forgetting factor is proposed to improve the prediction accuracy of occluded trajectory. The projection error compensation is introduced into the adaptive law of uncertain camera parameters to improve the tracking accuracy. The simulation results indicate the effectiveness of AFKF state estimator and uncalibrated visual tracking controller.

The two experiments are carried out by a real robot platform, and the results show that the ellipse trajectory tracking is achieved with an average error  $<3$  pixels when the real camera parameters are unknown. The unknown and occluded trajectory tracking is successfully achieved with an average error  $<3$  pixels without occlusion and  $<15$  pixels with occlusion. The experimental results can satisfy the requirements of the common robot manipulation applications. In particular, the proposed method is suitable for the extraction of non-robust target feature in a visual tracking task of robot manipulation, which is easy to lose due to environmental interference. Furthermore, the target occlusion in experiment is randomly given, and the occlusion time will be further discussed in future work to expand its application range.

## REFERENCES

- [1] X. Zhang, R. Wang, Y. Fang, B. Li, and B. Ma, "Acceleration-level pseudo-dynamic visual servoing of mobile robots with backstepping and dynamic surface control," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 10, pp. 2071–2081, Oct. 2019.
- [2] L. Li, Y.-H. Liu, T. Jiang, K. Wang, and M. Fang, "Adaptive trajectory tracking of nonholonomic mobile robots using vision-based position and velocity estimation," *IEEE Trans. Cybern.*, vol. 48, no. 2, pp. 571–582, Feb. 2018.
- [3] K. Zhang, J. Chen, Y. Li, and Y. Gao, "Unified visual servoing tracking and regulation of wheeled mobile robots with an uncalibrated camera," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 4, pp. 1728–1739, Aug. 2018.
- [4] Y. Zhu, J. Qiao, and L. Guo, "Adaptive sliding mode disturbance observer-based composite control with prescribed performance of space manipulators for target capturing," *IEEE Trans. Ind. Electron.*, vol. 66, no. 3, pp. 1973–1983, Mar. 2019.
- [5] T. Oliveira, A. P. Aguiar, and P. Encarnacao, "Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems," *IEEE Trans. Robot.*, vol. 32, no. 5, pp. 1062–1078, Oct. 2016.
- [6] J. Virgili-Llop, C. Zagaris, R. Zappulla, A. Bradstreet, and M. Romano, "A convex-programming-based guidance algorithm to capture a tumbling object on orbit using a spacecraft equipped with a robotic manipulator," *Int. J. Robot. Res.*, vol. 38, no. 1, pp. 40–72, Jan. 2019.
- [7] G. Dong and Z. H. Zhu, "Predictive visual servo kinematic control for autonomous robotic capture of non-cooperative space target," *Acta Astronautica*, vol. 151, pp. 173–181, Oct. 2018.
- [8] J. Virgili-Llop, J. V. Drew, R. Zappulla, and M. Romano, "Laboratory experiments of resident space object capture by a spacecraft–manipulator system," *Aerosp. Sci. Technol.*, vol. 71, pp. 530–545, Dec. 2017.
- [9] P. Huang, M. Wang, Z. Meng, F. Zhang, and Z. Liu, "Attitude takeover control for post-capture of target spacecraft using space robot," *Aerosp. Sci. Technol.*, vol. 51, pp. 171–180, Apr. 2016.
- [10] W. Pairo, P. Loncomilla, and J. R. Del Solar, "A delay-free and robust object tracking approach for robotics applications," *J. Intell. Robot. Syst.*, vol. 95, no. 1, pp. 99–117, Jul. 2019.
- [11] D. A. Mercado-Ravell, P. Castillo, and R. Lozano, "Visual detection and tracking with UAVs, following a mobile object," *Adv. Robot.*, vol. 33, nos. 7–8, pp. 388–402, Apr. 2019.
- [12] C.-Y. Tsai, K.-T. Song, X. Dutoit, H. Van Brussel, and M. Nuttin, "Robust visual tracking control system of a mobile robot based on a dual-Jacobian visual interaction model," *Robot. Auto. Syst.*, vol. 57, nos. 6–7, pp. 652–664, Jun. 2009.
- [13] H. Wang, C. C. Cheah, W. Ren, and Y. Xie, "Passive separation approach to adaptive visual tracking for robotic systems," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 6, pp. 2232–2241, Nov. 2018.
- [14] X. Liang, H. Wang, Y.-H. Liu, W. Chen, and T. Liu, "Formation control of nonholonomic mobile robots without position and velocity measurements," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 434–446, Apr. 2018.
- [15] Y. Chen and B. Guo, "Sliding mode fault tolerant tracking control for a single-link flexible joint manipulator system," *IEEE Access*, vol. 7, pp. 83046–83057, 2019.
- [16] H. Jun-Pei, H. Qi, L. Yan-Hui, W. Kai, Z. Ming-Chao, and X. Zhen-Bang, "Neural network control of space manipulator based on dynamic model and disturbance observer," *IEEE Access*, vol. 7, pp. 130101–130112, 2019.
- [17] M. Kaya, A. Denasi, S. Scheggi, E. Agbahca, C. Yoon, D. H. Gracias, and S. Misra, "A multi-rate state observer for visual tracking of magnetic micro-agents using 2D slow medical imaging modalities," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 5386–5393.
- [18] F. Aghili, "A prediction and motion-planning scheme for visually guided robotic capturing of free-floating tumbling objects with uncertain dynamics," *IEEE Trans. Robot.*, vol. 28, no. 3, pp. 634–649, Jun. 2012.
- [19] F. Aghili, M. Kuryllo, G. Okouneva, and C. English, "Fault-tolerant position/attitude estimation of free-floating space objects using a laser range sensor," *IEEE Sensors J.*, vol. 11, no. 1, pp. 176–185, Jan. 2011.
- [20] L. Yang, Q. Wang, and L. Zhang, "Model-free trajectory tracking control of two-particle magnetic microrobot," *IEEE Trans. Nanotechnol.*, vol. 17, no. 4, pp. 697–700, Jul. 2018.
- [21] H. Ben-Said, J. Stephant, and O. Labbani-Igbida, "Sensor-based control using finite time observer of visual signatures: Application to corridor following," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Systems (IROS)*, Daejeon, South Korea, Oct. 2016, pp. 418–423.
- [22] A. Parikh, T.-H. Cheng, R. Licitra, and W. E. Dixon, "A switched systems approach to image-based localization of targets that temporarily leave the camera field of view," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 6, pp. 2149–2156, Nov. 2018.
- [23] A. Parikh, T.-H. Cheng, H.-Y. Chen, and W. E. Dixon, "A switched systems framework for guaranteed convergence of image-based observers with intermittent measurements," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 266–280, Apr. 2017.
- [24] A. Parikh, R. Kamalapurkar, and W. E. Dixon, "Target tracking in the presence of intermittent measurements via motion model learning," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 805–819, Jun. 2018.
- [25] K. Shojaei, "Three-dimensional neural network tracking control of a moving target by underactuated autonomous underwater vehicles," *Neural Comput. Appl.*, vol. 31, no. 2, pp. 509–521, Feb. 2019.
- [26] Y. Zhang, C. Hua, Y. Li, and X. Guan, "Adaptive neural networks-based visual servoing control for manipulator with visibility constraint and dead-zone input," *Neurocomputing*, vol. 332, pp. 44–55, Mar. 2019.
- [27] J. Qu, F. Zhang, Y. Fu, G. Li, and S. Guo, "Adaptive neural network visual servoing of dual-arm robot for cyclic motion," *Ind. Robot.*, vol. 44, no. 2, pp. 210–221, Mar. 2017.
- [28] B. Li, X. Zhang, Y. Fang, and W. Shi, "Visual servoing of wheeled mobile robots without desired images," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 2835–2844, Aug. 2019.
- [29] Y. Qiu, B. Li, W. Shi, and X. Zhang, "Visual servo tracking of wheeled mobile robots with unknown extrinsic parameters," *IEEE Trans. Ind. Electron.*, vol. 66, no. 11, pp. 8600–8609, Nov. 2019.



- [30] Y. Zhang, C. Hua, and J. Qian, "Adaptive robust visual servoing/force control for robot manipulator with dead-zone input," *IEEE Access*, vol. 7, pp. 129627–129636, 2019.
- [31] Z. Qiu, S. Hu, and X. Liang, "Model predictive control for uncalibrated and constrained image-based visual servoing without joint velocity measurements," *IEEE Access*, vol. 7, pp. 73540–73554, 2019.
- [32] Z. Qiu, S. Hu, and X. Liang, "Model predictive control for constrained image-based visual servoing in uncalibrated environments," *Asian J. Control*, vol. 21, no. 2, pp. 783–799, Mar. 2019.
- [33] Z. Wang, S. Gong, D. Li, D. Zhou, and H. Lu, "LED chip accurate positioning control based on visual servo using dual rate adaptive fading Kalman filter," *ISA Trans.*, vol. 87, pp. 163–173, Apr. 2019.



**JIADI QU** received the M.S. degree in information science from Nagoya University, Nagoya, Japan, in 2012, and the Ph.D. degree in mechatronics engineering from the Harbin Institute of Technology, China, in 2019. He is currently a Postdoctoral Fellow with the College of Marine electrical engineering, Dalian Maritime University, China. His research interests include humanoid robot manipulation and robot vision.



**FUHAI ZHANG** received the Ph.D. degree in mechatronics engineering from the Harbin Institute of Technology, China, in 2010. He is currently a Lecturer with the State Key Laboratory of Robotics and System, Harbin Institute of Technology, China. His research interests include dual-arm robotics and rehabilitation robotics.



**YUNXI TANG** received the B.S. degree in mechatronics engineering from the Harbin Institute of Technology, China, in 2019. His research interests include robotics automation and learning control.



**YILI FU** received the Ph.D. degree in mechatronic engineering from the Harbin Institute of Technology, China, in 1996. He is currently a Professor with the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His research interests include space robotics, medical robotics, and mobile robotics.

...