Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A hybrid visual servo control method for simultaneously controlling a nonholonomic mobile and a manipulator*

Wei LI, Rong XIONG‡

State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China

E-mail: li_wei_666@163.com; rxiong@zju.edu.cn

Received Aug. 31, 2019; Revision accepted Oct. 9, 2019; Crosschecked May 18, 2020; Published online June 26, 2020

Abstract: Visual servo control rules that refer to the control methods of robot motion planning using image data acquired from the camera mounted on the robot have been widely applied to the motion control of robotic arms or mobile robots. The methods are usually classified as image-based visual servo, position-based visual servo, and hybrid visual servo (HVS) control rules. Mobile manipulation enhances the working range and flexibility of robotic arms. However, there is little work on applying visual servo control rules to the motion of the whole mobile manipulation robot. We propose an HVS motion control method for a mobile manipulation robot which combines a six-degree-of-freedom (6-DOF) robotic arm with a nonholonomic mobile base. Based on the kinematic differential equations of the mobile manipulation robot, the global Jacobian matrix of the whole robot is derived, and the HVS control equation is derived using the whole Jacobian matrix combined with position and visual image information. The distance between the gripper and target is calculated through the observation of the marker by a camera mounted on the gripper. The differences between the positions of the markers' feature points and the expected positions of them in the image coordinate system are also calculated. These differences are substituted into the control equation to obtain the speed control law of each degree of freedom of the mobile manipulation robot. To avoid the position error caused by observation, we also introduce the Kalman filter to correct the positions and orientations of the end of the manipulator. Finally, the proposed algorithm is validated on a mobile manipulation platform consisting of a Bulldog chassis, a UR5 robotic arm, and a ZED camera.

Key words: Mobile manipulation; Hybrid visual servo; Eye-in-hand; Global Jacobian; Kalman filter
https://doi.org/10.1631/FITEE.1900460 **CLC number:** TP13

1 Introduction

Visual servo control rules refer to the methods that control a robot to move to the target position according to the image acquired from a camera mounted on the robot. At present, many methods have been proposed and are usually classified into three categories: image-based visual servo (IBVS), position-based visual servo (PBVS), and hybrid visual servo (HVS) control rules. The so-called IBVS control refers to driving the robot to move along a path to gradually reduce the distance between the current and target positions of the image feature points in an image coordinate system. Therefore, this method can make the feature points move along smooth and straight paths close to the target positions of the image feature points in the image coordinate system. However, this may cause the actual robot's movement to be very distorted in the world coordinate system. PBVS control calculates the three-dimensional (3D) position of the target through the camera, and then drives the robot to

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. U1609210) and Science and Technology Project of Zhejiang Province, China (No. 2019C01043)

ORCID: Wei LI, https://orcid.org/0000-0001-8446-5427; Rong XIONG, https://orcid.org/0000-0001-9318-9014

move along a path that narrows the distance between the current and target positions of the gripper. This may cause the feature points in the image to move out of the field of view. To overcome the shortcomings of the two methods, combining the advantages of IBVS and PBVS, the HVS control rule has been proposed so that the distortion of the robot's movement and the movement of the target out of the field of view of the camera can be avoided at the same time.

According to the relationship between the camera and robot, visual servo control can be divided into two cases: eye-in-hand and eye-to-hand configurations. The eye-in-hand configuration means that the camera is mounted at the end of the robotic arm, and the camera moves with the arm. The eye-to-hand configuration means that the camera is mounted in a relatively fixed position out of the arm, and the field of view always covers the arm. The camera can remain motionless while the arm is moving. In general, due to the limited range of vision, the eye-to-hand configuration cannot adapt to robots with a large range of motion, such as mobile robots and mobile manipulators. However, the eye-in-hand configuration can make the target remain within the field of view of the camera during moving, therefore making it applicable to these kinds of robots.

At present, most algorithms used in mobile manipulation separate the planning of the mobile bases and the robotic arms. That means, the mobile bases are driven to the designated positions first, and then with the aid of camera vision, the robotic arms complete their tasks. Visual servo control methods are used mainly for the latter process. In this study we attempt to find a visual servo method that controls the robotic arm and the mobile base as a whole, so that the robot can grasp while moving, and perform a smoother path with a shorter working time.

The main contributions of this study are as follows:

1. Based on the existing visual servo control methods of robotic arms, combined with the control laws of a moving nonholonomic chassis, an HVS method is proposed for the whole mobile manipulation. The global Jacobian matrix which covers not only the six degrees of freedom (DOF) of the arm, but also the mobile nonholonomic base is derived using the kinematic differential equation of the whole robot. Then it is substituted into a convergence control law to obtain the visual servo control rule that covers the whole robot.

2. An HVS control rule that covers the whole mobile manipulation robot is proposed based on PBVS and IBVS. This rule considers both visual observation constraints and convergence speeds, thereby improving the overall performance of the visual servo control rule. The stability of the HVS control rule is proved by Lyapunov analysis.

3. Real-time position and orientation of the gripper are calculated using a Kalman filter, making the visual servo control more precise.

2 Related work

Classical visual servo control methods are divided into IBVS, PBVS, and HVS. The basic principles of the three methods were described in detail in Chaumette and Hutchinson (2006, 2007). The IBVS control rule subtracts the target position vector of the marker's feature points from the present position vector, and multiplies it by the inverse of the image Jacobian matrix. The result is substituted into a Lyapunov speed control law to derive the velocity and angular velocity that the camera should take when it is moving and converging to the target position and orientation. The PBVS control rule uses the projected image of the object to calculate the position and orientation of the object in the camera coordinate system. Then the position and orientation vector is multiplied by the position Jacobian inverse. The result is substituted into the Lyapunov speed control law to derive the velocity and angular velocity of the camera. Both of the above methods control the movement of the position and orientation of the camera at the same time. Some HVS control rules divide the Jacobian matrix into a position portion and an orientation portion to separately control the camera's position and orientation (Chaumette and Hutchinson, 2007). First, the angular velocity is calculated by the difference between the present and target orientations, and then multiplied by the Jacobian matrix of the orientation portion. Finally, the result is substituted into the control scheme of the position portion to calculate the velocity of the camera combined with the position of the object. Hafez et al. (2008) pointed out that the IBVS method may cause distortion of the actual camera's motion in the world coordinate

system, while the PBVS method may make the target's feature points leave the camera's field of view during motion. Therefore, the HVS control rule which can switch between the two visual servo methods is proposed. By constructing the switching function, the control scheme is switched to PBVS when the robotic arm is about to be twisted to the limit, and switched back to IBVS when the feature points are about to move out of the camera's field of view. Kermorgant and Chaumette (2011) addressed the issue of hybrid 2D/3D visual servoing by adding 2D information to the PBVS method to ensure the visibility constraint. Agravante and Chaumette (2017) proposed an active vision method based on the IBVS method. By optimizing the Fisher information matrix defined in the study, a speed control rule can be used to improve the pose estimation accuracy and obtain a better servo effect.

Since the position estimation has a large error, the filtering method is generally used to correct the visual observation error. The extended Kalman filter is used to correct the observation error of the camera's position and orientation in the PBVS method (Dong and Zhu, 2015), in which the position, orientation, velocity, and angular velocity of the target are used as the state quantities, and the observations of these are used as the measurement quantities to be optimized. Particle filter has been introduced in the PBVS method to correct the observation error of the camera's position and orientation (Fantacci et al., 2018). The method relies solely on the visual reconstruction of the target shape by matching with the target's model to initially determine the pose of the camera relative to the target. Kalman filter has been introduced in the IBVS control method (Zhong et al., 2015), taking the image Jacobian matrix as the state quantity to be optimized and taking the observed pixel positions of the feature points as the measurement. The state and measurement equations were built, the neural network was used to optimize the residuals of the equations, and the output results were added to the values of the state equations to output a closest real-world image Jacobian matrix. Recently, researchers have used deep learning for visual servo control of robotic arms (Levine et al., 2016; Bateux et al., 2017), and achieved a good grasp accuracy.

Visual servo control was first used for the operation control of the robotic arms and has achieved a great success. Then it was gradually applied to the motion control of mobile robots and mobile manipulators. Benhimane and Malis (2007) and Hu et al. (2008) proposed a visual servo control strategy based on homography matrix decomposition. This HVS control method has been widely used in mobile robots. A camera and inertial navigation were installed at the gripper of a mobile manipulation robot (Sandy and Buchli, 2017). Kalman filter was used to obtain the position and velocity of the gripper. The desired motion trajectory of the robot gripper was substituted into the decoupled arm dynamic equation to obtain the control torque of each joint angle. Finally, we have the effect that the end of the robotic arm remained motionless when the robot base was moving. Lang et al. (2016) installed a camera at the end of a robotic arm of a mobile manipulator, and used it to collect data of the target marker to perform IBVS control. Wang et al. (2014) proposed a visual servo control method based on the visual information synthesis of two cameras (a monocular camera on the arm end and a binocular camera on the base), and successfully applied this technique to mobile manipulators. However, the above methods are not servo control rules that regard the moving base and the robotic arm as a whole. Tsakiris et al. (1998) proposed a method which takes a nonholonomic base and a 3-DOF arm as a whole. However, it uses only IBVS, and the structure of the robotic arm is very simple. The above methods all adopt IBVS methods while ignoring the advantages of PBVS methods. The method proposed in this study regards the base and the arm as a whole to do visual servo controlling, and the method combines the respective strengths of PBVS and IBVS under different conditions, greatly improving the comprehensive performance of the visual servo method of mobile manipulation.

3 Theoretical derivation and algorithm implementation

3.1 Derivation of the global Jacobian matrix of mobile manipulation robots

Fig. 1 shows the task scenario of this study. The marker which is used for visual positioning of the target object and the mobile manipulator is perpendicular to the ground. Our purpose is to make the mobile manipulator recognize the target marker

and then to guide the robot to the vicinity of the target object to grab the target object with the help of the visual servo method. The mobile manipulator is generally composed of a differential drive base and a 6-DOF robotic arm with a camera and a gripper at the end. The traditional method first drives the base to a fixed position near the target object, and then makes the robotic arm grab the target object. The effect of the algorithm realized in this study is that the robotic arm and the base simultaneously move to the target object and grasp it.

As shown in Fig. 1, the world coordinate system $(X^{\mathrm{W}}, Y^{\mathrm{W}}, Z^{\mathrm{W}})$ is defined as follows: the origin is the projection of the marker center on the ground, the X^{W} axis is parallel to the marker plane to the right, the Y^{W} axis is perpendicular to the marker plane, and the Z^{W} axis is perpendicular to the ground and passes through the center of the marker. $(X^{\mathrm{R}}, Y^{\mathrm{R}}, Z^{\mathrm{R}})$ is the robot base coordinate system. Its origin is defined at the center of the robot base and is on the ground. The X^{R} axis points to the front of the robot base, the Y^{R} axis points to the left side of the robot, and the Z^{R} axis is perpendicular to the base. (X, Y, Z) is the robotic arm coordinate system. Its three axes are parallel to the three axes in the robot base coordinate system respectively, and its origin which lies on the base of the robotic arm translates a $(a, b, c)^{\mathrm{T}}$ vector from the origin of the robot base coordinate system. In addition, $(X^{\mathrm{C}}, Y^{\mathrm{C}}, Z^{\mathrm{C}})$ is the camera coordinate system, with its origin at the center of the camera, and its direction is also shown in Fig. 1. $(X^{\mathrm{G}}, Y^{\mathrm{G}}, Z^{\mathrm{G}})$ is the gripper coordinate system, with its origin in the center of the gripper and its X^{G}, Y^{G}, and Z^{G} directions parallel to the $-X^{\mathrm{C}}$, Z^{C}, and Y^{C} directions, respectively. The origin of the robot base coordinate system is at the position of $(x_0, y_0, 0)$ of the world coordinate system, and there is an angle α between its X^{R} axis and X^{W} axis. Fig. 2 shows the relationships between the above-mentioned coordinate systems and the physical quantities. As shown in Fig. 2a, a grid plane is projected from the camera. This represents the pixel coordinate system (u, v). The u axis is parallel to the X^{C} axis, and the v axis is parallel to the Y^{C} axis. In this study we bring the distance between the gripper and the target gripper, and the distance between the positions of the current and the target marker's feature points in the pixel coordinate system into the servo control equation to obtain the control rule after processing the camera data. The control rule then moves the robot to the vicinity of the marker and drives the gripper to grab the target object.

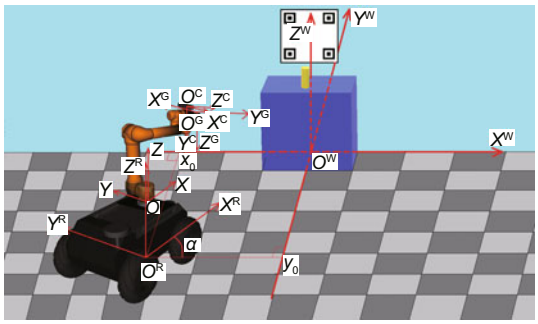

Fig. 1 Scene for the hybrid visual servo (HVS) of mobile manipulation, showing the different coordinate systems and the marker used for image processing

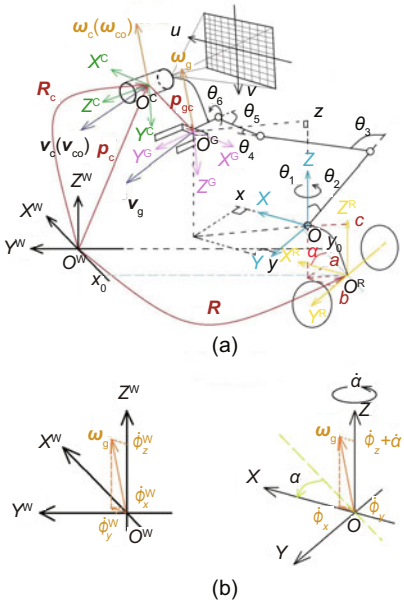

Fig. 2 General diagram of coordinate systems and physical quantities: (a) transformations of coordinate systems; (b) decompositions of the angular velocity in the world (left) and the robotic arm (right) coordinate systems

The following equations can be derived from the relationship between the position of the gripper in the robotic arm coordinate system (x, y, z) and the position of the gripper in the world coordinate system $(x^{\mathrm{W}}, y^{\mathrm{W}}, z^{\mathrm{W}})$ as shown in Fig. 2a:

$$x^{\mathrm{W}} = (x + a)\cos\alpha - (y + b)\sin\alpha + x_0, \quad (1)$$

$$y^{\mathrm{W}} = (x + a)\sin\alpha + (y + b)\cos\alpha + y_0, \quad (2)$$

$$z^{\mathrm{W}} = z + c. \quad (3)$$

Differentiate the left and right sides of Eqs. (1)–(3) with respect to time to obtain the transformation between the velocity of the gripper relative to the world coordinate system and the velocity of the gripper relative to the robotic arm coordinate system:

$$\begin{aligned}\dot{x}^{\mathrm{W}} =&\dot{x}\cos\alpha-(x+a)(\sin\alpha)\dot{\alpha}-\dot{y}\sin\alpha\\&-(y+b)(\cos\alpha)\dot{\alpha}+\dot{x}_0,\end{aligned}\tag{4}$$

$$\begin{aligned}\dot{y}^{\mathrm{W}} =&\dot{x}\sin\alpha+(x+a)(\cos\alpha)\dot{\alpha}+\dot{y}\cos\alpha\\&-(y+b)(\sin\alpha)\dot{\alpha}+\dot{y}_0,\end{aligned}\tag{5}$$

$$\dot{z}^{\mathrm{W}} =\dot{z}.\tag{6}$$

This transformation means that the velocity of the gripper in the world coordinate system is related to the movement of the robotic arm, the velocity of the robot base, and also the angle α between the robot base and the world coordinate system. In the same way, according to kinematics knowledge, we can obtain the transformation between the angular velocity of the gripper in the world coordinate system and that in the robotic arm coordinate system as shown in Fig. 2b:

$$\dot{\phi}_x^{\mathrm{W}}=\dot{\phi}_x\cos\alpha-\dot{\phi}_y\sin\alpha,\tag{7}$$

$$\dot{\phi}_y^{\mathrm{W}}=\dot{\phi}_x\sin\alpha+\dot{\phi}_y\cos\alpha,\tag{8}$$

$$\dot{\phi}_z^{\mathrm{W}}=\dot{\phi}_z+\dot{\alpha},\tag{9}$$

where $(\phi_x,\ \phi_y,\ \phi_z)$ and $(\phi_x^{\mathrm{W}},\ \phi_y^{\mathrm{W}},\ \phi_z^{\mathrm{W}})$ are the Euler angles of the gripper in the robotic arm coordinate system and the world coordinate system, respectively. In addition, in the nonholonomic robot, the velocity of the base v and angular velocity of the base ω can be introduced:

$$\dot{x}_0=v\cos\alpha,\tag{10}$$

$$\dot{y}_0=v\sin\alpha,\tag{11}$$

$$\dot{\alpha}=\omega.\tag{12}$$

Substituting Eqs. (10)–(12) into Eqs. (4)–(9) can derive a new set of equations. We may wish to express this new set of equations in the form of differential equations of the matrix for subsequent processing:

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}x^{\mathrm{W}} & y^{\mathrm{W}} & z^{\mathrm{W}} & \phi_x^{\mathrm{W}} & \phi_y^{\mathrm{W}} & \phi_z^{\mathrm{W}}\end{bmatrix}^{\mathrm{T}}\\&=\begin{bmatrix}&&T_1&T_2\\&&T_3&T_4\\\boldsymbol{R}&\boldsymbol{0}_{3\times3}&0&0\\\boldsymbol{0}_{3\times3}&\boldsymbol{R}&0&0\\&&0&0\\&&1&0\end{bmatrix}\\&\quad\cdot\begin{bmatrix}\frac{\mathrm{d}x}{\mathrm{d}t}&\frac{\mathrm{d}y}{\mathrm{d}t}&\frac{\mathrm{d}z}{\mathrm{d}t}&\frac{\mathrm{d}\phi_x}{\mathrm{d}t}&\frac{\mathrm{d}\phi_y}{\mathrm{d}t}&\frac{\mathrm{d}\phi_z}{\mathrm{d}t}&\omega&v\end{bmatrix}^{\mathrm{T}}\\&=\begin{bmatrix}&&T_1&T_2\\&&T_3&T_4\\\boldsymbol{R}&\boldsymbol{0}_{3\times3}&0&0\\\boldsymbol{0}_{3\times3}&\boldsymbol{R}&0&0\\&&0&0\\&&1&0\end{bmatrix}\begin{bmatrix}\boldsymbol{J}_{0(6\times6)}&\boldsymbol{0}_{6\times2}\\\boldsymbol{0}_{2\times6}&\boldsymbol{I}_{2\times2}\end{bmatrix}\\&\quad\cdot\begin{bmatrix}\frac{\mathrm{d}\theta_1}{\mathrm{d}t}&\frac{\mathrm{d}\theta_2}{\mathrm{d}t}&\frac{\mathrm{d}\theta_3}{\mathrm{d}t}&\frac{\mathrm{d}\theta_4}{\mathrm{d}t}&\frac{\mathrm{d}\theta_5}{\mathrm{d}t}&\frac{\mathrm{d}\theta_6}{\mathrm{d}t}&\omega&v\end{bmatrix}^{\mathrm{T}},\end{aligned}\tag{13}$$

$$\boldsymbol{R}=\begin{bmatrix}\cos\alpha&-\sin\alpha&0\\\sin\alpha&\cos\alpha&0\\0&0&1\end{bmatrix},\tag{14}$$

$$\boldsymbol{J}_{0(6\times6)}=\begin{bmatrix}\boldsymbol{a}_1\times(\boldsymbol{p}_6-\boldsymbol{p}_1)&\boldsymbol{a}_2\times(\boldsymbol{p}_6-\boldsymbol{p}_2)\\\boldsymbol{a}_1&\boldsymbol{a}_2\\\cdots&\boldsymbol{a}_5\times(\boldsymbol{p}_6-\boldsymbol{p}_5)&\boldsymbol{0}\\\cdots&\boldsymbol{a}_5&\boldsymbol{a}_6\end{bmatrix},\tag{15}$$

where $T_1=-(x+a)\sin\alpha-(y+b)\cos\alpha$, $T_2=\cos\alpha$, $T_3=(x+a)\cos\alpha-(y+b)\sin\alpha$, and $T_4=\sin\alpha$. $\boldsymbol{J}_{0(6\times6)}$ is the Jacobian matrix of the robotic arm at the moment, and $\boldsymbol{a}_i$ and $\boldsymbol{p}_i$ are the normalized rotation and displacement vectors of the i^{th} joint of the robotic arm, respectively. This can be easily deduced from the supposition principle. We split the Jacobian matrix $\boldsymbol{J}_0$ into two parts: the translation part $\boldsymbol{J}_{0\mathrm{t}}$ which comprises the first three rows of $\boldsymbol{J}_0$ and the rotation part $\boldsymbol{J}_{0\mathrm{r}}$ which comprises the last three rows of $\boldsymbol{J}_0$. From Eq. (13) we obtain the form of the global Jacobian matrix:

$$\boldsymbol{J}=\begin{bmatrix}&&T_1&T_2\\&&T_3&T_4\\\boldsymbol{R}&\boldsymbol{0}_{3\times3}&0&0\\\boldsymbol{0}_{3\times3}&\boldsymbol{R}&0&0\\&&0&0\\&&1&0\end{bmatrix}$$

$$\cdot \begin{bmatrix} \boldsymbol{J}_{0(6\times6)} & \boldsymbol{0}_{6\times2} \\ \boldsymbol{0}_{2\times6} & \boldsymbol{I}_{2\times2} \end{bmatrix} = \begin{bmatrix} & T_1 & T_2 \\ & T_3 & T_4 \\ \boldsymbol{R}\cdot\boldsymbol{J}_{0\mathrm{t}} & 0 & 0 \\ \boldsymbol{R}\cdot\boldsymbol{J}_{0\mathrm{r}} & 0 & 0 \\ & 0 & 0 \\ & 1 & 0 \end{bmatrix}. \tag{16}$$

It represents the overall partial differential of the velocity and angular velocity of the gripper in the world coordinate system with respect to the velocities of eight degrees of freedom. So, it is a 6×8 matrix. The above differential equation can be reversed to obtain Eq. (17):

$$\begin{aligned} &\begin{bmatrix} \dfrac{\mathrm{d}\theta_1}{\mathrm{d}t} & \dfrac{\mathrm{d}\theta_2}{\mathrm{d}t} & \dfrac{\mathrm{d}\theta_3}{\mathrm{d}t} & \dfrac{\mathrm{d}\theta_4}{\mathrm{d}t} & \dfrac{\mathrm{d}\theta_5}{\mathrm{d}t} & \dfrac{\mathrm{d}\theta_6}{\mathrm{d}t} & \omega & v \end{bmatrix}^{\mathrm{T}} \\ =&\boldsymbol{J}^{+}\cdot\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x^{\mathrm{W}} & y^{\mathrm{W}} & z^{\mathrm{W}} & \phi_x^{\mathrm{W}} & \phi_y^{\mathrm{W}} & \phi_z^{\mathrm{W}} \end{bmatrix}^{\mathrm{T}}. \end{aligned} \tag{17}$$

In Eq. (17), the left side of the equation is composed of the robot motion control quantities and $\boldsymbol{J}^{+}$ (an 8×6 matrix) is the pseudo inverse of the global Jacobian matrix. The specific form is

$$\boldsymbol{J}^{+} = \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{J}\boldsymbol{J}^{\mathrm{T}})^{-1}. \tag{18}$$

3.2 Derivation of hybrid visual servo control of a mobile manipulation robot

As shown in Fig. 2a, assume that in the camera coordinate system, the camera has a general velocity ($\boldsymbol{V}_{\mathrm{co}}$) which includes the velocity ($\boldsymbol{v}_{\mathrm{co}}$) and the angular velocity ($\boldsymbol{\omega}_{\mathrm{co}}$), and the position of the gripper can be represented with a translation vector ($\boldsymbol{p}_{\mathrm{gc}}$). In the world coordinate system, the rotation matrix of the camera is $\boldsymbol{R}_{\mathrm{c}}$, the translation vector of the camera is $\boldsymbol{p}_{\mathrm{c}}$, the velocity of the camera is $\boldsymbol{v}_{\mathrm{c}}$, the angular velocity of the camera is $\boldsymbol{\omega}_{\mathrm{c}}$, the velocity of the gripper is $\boldsymbol{v}_{\mathrm{g}}$, and the angular velocity of the gripper is $\boldsymbol{\omega}_{\mathrm{g}}$. The general velocity of the gripper is $\boldsymbol{V}_{\mathrm{g}} = [\boldsymbol{v}_{\mathrm{g}}, \boldsymbol{\omega}_{\mathrm{g}}]^{\mathrm{T}}$. The general velocity of the camera is $\boldsymbol{V}_{\mathrm{c}} = [\boldsymbol{v}_{\mathrm{c}}, \boldsymbol{\omega}_{\mathrm{c}}]^{\mathrm{T}}$. There is a transformation between them:

$$\boldsymbol{V}_{\mathrm{g}} = \boldsymbol{H}\cdot\boldsymbol{V}_{\mathrm{co}}. \tag{19}$$

To obtain the expression of $\boldsymbol{H}$, the translation vector of the gripper $\boldsymbol{p}_{\mathrm{g}}$ in the world coordinate system is

$$\boldsymbol{p}_{\mathrm{g}} = \boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{gc}} + \boldsymbol{p}_{\mathrm{c}}. \tag{20}$$

Differentiating both sides of the equation with respect to time yields

$$\dot{\boldsymbol{p}}_{\mathrm{g}} = \dot{\boldsymbol{p}}_{\mathrm{c}} + \dot{\boldsymbol{R}}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{gc}} = \dot{\boldsymbol{p}}_{\mathrm{c}} + \boldsymbol{\omega}_{\mathrm{c}}\times(\boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{gc}}). \tag{21}$$

Therefore, the relationship between the velocity and angular velocity vectors of the gripper and the camera in the world coordinate system is

$$\begin{aligned} \begin{bmatrix} \boldsymbol{v}_{\mathrm{g}} \\ \boldsymbol{\omega}_{\mathrm{g}} \end{bmatrix} &= \begin{bmatrix} \boldsymbol{I} & -[\boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{gc}}]_{\times} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_{\mathrm{c}} \\ \boldsymbol{\omega}_{\mathrm{c}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{I} & -[\boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{g}}]_{\times} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{\mathrm{c}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}_{\mathrm{c}} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_{\mathrm{co}} \\ \boldsymbol{\omega}_{\mathrm{co}} \end{bmatrix}, \end{aligned} \tag{22}$$

where the subscript "$\times$" changes a vector to its cross product matrix. The expression of the conversion relationship $\boldsymbol{H}$ from camera velocity to gripper velocity is

$$\begin{aligned} \boldsymbol{H} &= \begin{bmatrix} \boldsymbol{I} & -[\boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{g}}]_{\times} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_{\mathrm{c}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R}_{\mathrm{c}} \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{R}_{\mathrm{c}} & -[\boldsymbol{R}_{\mathrm{c}}\cdot\boldsymbol{p}_{\mathrm{g}}]_{\times}\cdot\boldsymbol{R}_{\mathrm{c}} \\ \boldsymbol{0} & \boldsymbol{R}_{\mathrm{c}} \end{bmatrix}. \end{aligned} \tag{23}$$

In the PBVS method, there is a transformation between the general velocity of the gripper $\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{p})}$, the general displacement between the initial and target positions $\boldsymbol{e}_{\mathrm{p}}$, and its derivative with respect to time $\dot{\boldsymbol{e}}_{\mathrm{p}}$:

$$-\kappa_0\boldsymbol{e}_{\mathrm{p}} = \dot{\boldsymbol{e}}_{\mathrm{p}} = \boldsymbol{L}_{\mathrm{p}}\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{p})}, \tag{24}$$

where κ_0 is a positive constant that adjusts the convergence speed, $\boldsymbol{e}_{\mathrm{p}}$ is the vector that represents $[\boldsymbol{t}-\boldsymbol{t}_0, \theta\boldsymbol{u}]^{\mathrm{T}}$ with $\boldsymbol{t}$ and $\boldsymbol{t}_0$ the current and target position vectors of the gripper in the world coordinate system respectively, and $\theta\boldsymbol{u}$ the angular vector of the rotation between the current and target orientations of the gripper. How $\boldsymbol{t}$ and $\theta\boldsymbol{u}$ are calculated from images will be described in Section 4.1. The position Jacobian $\boldsymbol{L}_{\mathrm{p}}$ can be represented as follows:

$$\boldsymbol{L}_{\mathrm{p}} = \begin{bmatrix} -\boldsymbol{I}_3 & [\boldsymbol{t}-\boldsymbol{t}_0]_{\times} \\ \boldsymbol{0} & \boldsymbol{L}_{\mathrm{w}} \end{bmatrix}, \tag{25}$$

where

$$\boldsymbol{L}_{\mathrm{w}} = \boldsymbol{I}_3 - \frac{\theta}{2}\boldsymbol{u}_{\times} + \left(1 - \frac{\operatorname{sinc}\theta}{\operatorname{sinc}^2\frac{\theta}{2}}\right)\boldsymbol{u}_{\times}^2. \tag{26}$$

Then the formula can be rewritten as follows:

$$\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{p})} = \boldsymbol{L}_{\mathrm{p}}^{-1}\dot{\boldsymbol{e}}_{\mathrm{p}} = -\kappa_0\boldsymbol{L}_{\mathrm{p}}^{-1}\boldsymbol{e}_{\mathrm{p}}. \tag{27}$$

In the IBVS method, there is a transformation between the general velocity of the camera $\boldsymbol{V}_{\mathrm{c}}$, the

general displacement between the initial and target pixel positions of the feature points $\boldsymbol{e}_{\mathrm{i}}$, and the derivative of the general displacement with respect to time $\dot{\boldsymbol{e}}_{\mathrm{i}}$:

$$-\lambda_0\boldsymbol{e}_{\mathrm{i}}=\dot{\boldsymbol{e}}_{\mathrm{i}}=\boldsymbol{L}_{\mathrm{i}}\boldsymbol{V}_{\mathrm{co}}, \tag{28}$$

where $\boldsymbol{e}_{\mathrm{i}}=[u_1-u_{10},v_1-v_{10},...,u_4-u_{40},v_4-v_{40}]^{\mathrm{T}}$ is the displacement between the current and target pixel positions of the four feature points in the pixel coordinate system and λ_0 is a positive constant adjusting the convergence speed. $\boldsymbol{L}_{\mathrm{i}}$ is the Jacobian matrix of the combined velocity of the four feature points on the marker in the pixel coordinate system with respect to the camera's generalized velocity as shown in Eq. (29). The pixels of the four feature points in the camera are $[u_1,v_1]$, $[u_2,v_2]$, $[u_3,v_3]$, and $[u_4,v_4]$, the focal length of the camera is F, and z_i is the z-axis distance between the i^{th} feature point and the camera's optical center in the camera coordinate system.

$$\boldsymbol{L}_{\mathrm{i}}=\begin{bmatrix}-\frac{F}{z_1} & 0 & \frac{u_1}{z_1} & \frac{u_1v_1}{F} & -\frac{F^2+u_1^2}{F} & v_1\\ 0 & -\frac{F}{z_1} & \frac{v_1}{z_1} & \frac{F^2+u_1^2}{F} & -\frac{u_1v_1}{F} & -u_1\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ -\frac{F}{z_4} & 0 & \frac{u_4}{z_4} & \frac{u_4v_4}{F} & -\frac{F^2+u_4^2}{F} & v_4\\ 0 & -\frac{F}{z_4} & \frac{v_4}{z_4} & \frac{F^2+u_4^2}{F} & -\frac{u_4v_4}{F} & -u_4\end{bmatrix}. \tag{29}$$

Then Eq. (28) can be rewritten as follows:

$$\boldsymbol{V}_{\mathrm{co}}=\boldsymbol{L}_{\mathrm{i}}^{+}\dot{\boldsymbol{e}}_{\mathrm{i}}=-\lambda_0\boldsymbol{L}_{\mathrm{i}}^{+}\boldsymbol{e}_{\mathrm{i}}. \tag{30}$$

Combining Eqs. (19) and (30), we have

$$\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{i})}=\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\dot{\boldsymbol{e}}_{\mathrm{i}}=-\lambda_0\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\boldsymbol{e}_{\mathrm{i}}. \tag{31}$$

If only PBVS is considered in the control rule, a good servo result cannot be obtained. Because the target marker is likely to move out of the camera's field of view during the movement, and if the camera fails to track the target marker, the subsequent motion process will not continue. The positional relationship of the target marker's feature points in the pixel coordinate system must be taken into account in the whole process. If only IBVS is considered in the control rule, the movement of the robot may be very zigzag. So, both of them should be used. Now a new control rule is proposed:

$$\boldsymbol{V}_{\mathrm{g}}=\kappa_1\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{p})}+\lambda_1\boldsymbol{V}_{\mathrm{g}}^{(\mathrm{i})}=\kappa_1\boldsymbol{L}_{\mathrm{p}}^{-1}\dot{\boldsymbol{e}}_{\mathrm{p}}+\lambda_1\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\dot{\boldsymbol{e}}_{\mathrm{i}}, \tag{32}$$

where κ_1 and λ_1 are positive constants. Then

$$\boldsymbol{V}_{\mathrm{g}}=-\kappa\boldsymbol{L}_{\mathrm{p}}^{-1}\boldsymbol{e}_{\mathrm{p}}-\lambda\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\boldsymbol{e}_{\mathrm{i}}, \tag{33}$$

where $\kappa=\kappa_0\kappa_1$ and $\lambda=\lambda_0\lambda_1$. Combined with Eq. (17), the last form of the HVS control rule is

$$\begin{aligned}&\begin{bmatrix}\frac{\mathrm{d}\theta_1}{\mathrm{d}t} & \frac{\mathrm{d}\theta_2}{\mathrm{d}t} & \frac{\mathrm{d}\theta_3}{\mathrm{d}t} & \frac{\mathrm{d}\theta_4}{\mathrm{d}t} & \frac{\mathrm{d}\theta_5}{\mathrm{d}t} & \frac{\mathrm{d}\theta_6}{\mathrm{d}t} & \omega & v\end{bmatrix}^{\mathrm{T}}\\ &=-\boldsymbol{J}^{+}(\kappa\boldsymbol{L}_{\mathrm{p}}^{-1}\boldsymbol{e}_{\mathrm{p}}+\lambda\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\boldsymbol{e}_{\mathrm{i}}).\end{aligned} \tag{34}$$

3.3 Stability analysis

We investigate the stability of our HVS control rule. We can prove the stability with a quadratic candidate Lyapunov function. From the preceding part of the text, we can obtain

$$\begin{aligned}\boldsymbol{V}_{\mathrm{g}}&=\kappa_1\boldsymbol{L}_{\mathrm{p}}^{-1}\dot{\boldsymbol{e}}_{\mathrm{p}}+\lambda_1\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\dot{\boldsymbol{e}}_{\mathrm{i}}\\ &=\begin{bmatrix}\kappa_1\boldsymbol{L}_{\mathrm{p}}^{-1} & \boldsymbol{0}\\ \boldsymbol{0} & \lambda_1\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{e}}_{\mathrm{p}}\\ \dot{\boldsymbol{e}}_{\mathrm{i}}\end{bmatrix}.\end{aligned} \tag{35}$$

We use $\boldsymbol{S}$ to represent the matrix of Eq. (35). Then we can construct a Lyapunov function as

$$V=\frac{1}{2}[\boldsymbol{e}_{\mathrm{p}}^{\mathrm{T}}\ \boldsymbol{e}_{\mathrm{i}}^{\mathrm{T}}]\begin{bmatrix}\boldsymbol{e}_{\mathrm{p}}\\ \boldsymbol{e}_{\mathrm{i}}\end{bmatrix}. \tag{36}$$

It is positive at all times. Then differentiating the equation with respect to time yields

$$\dot{V}=[\boldsymbol{e}_{\mathrm{p}}^{\mathrm{T}}\boldsymbol{e}_{\mathrm{i}}^{\mathrm{T}}]\begin{bmatrix}\dot{\boldsymbol{e}}_{\mathrm{p}}\\ \dot{\boldsymbol{e}}_{\mathrm{i}}\end{bmatrix}=[\boldsymbol{e}_{\mathrm{p}}^{\mathrm{T}}\ \boldsymbol{e}_{\mathrm{i}}^{\mathrm{T}}]\boldsymbol{S}^{+}\boldsymbol{V}_{\mathrm{g}}, \tag{37}$$

where $\boldsymbol{S}^{+}$ is the pseudo inverse of matrix $\boldsymbol{S}$.

We can obtain another expression of $\boldsymbol{V}_{\mathrm{g}}$:

$$\begin{aligned}\boldsymbol{V}_{\mathrm{g}}&=-\kappa\boldsymbol{L}_{\mathrm{p}}^{-1}\boldsymbol{e}_{\mathrm{p}}-\lambda\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\boldsymbol{e}_{\mathrm{i}}\\ &=-\begin{bmatrix}\kappa_1\boldsymbol{L}_{\mathrm{p}}^{-1} & \boldsymbol{0}\\ \boldsymbol{0} & \lambda_1\boldsymbol{H}\boldsymbol{L}_{\mathrm{i}}^{+}\end{bmatrix}\begin{bmatrix}\boldsymbol{e}_{\mathrm{p}}\\ \boldsymbol{e}_{\mathrm{i}}\end{bmatrix}.\end{aligned} \tag{38}$$

Combining Eq. (37) with Eq. (38) and taking $\boldsymbol{Q}$ to represent the matrix of Eq. (38), we have

$$\dot{V}=-[\boldsymbol{e}_{\mathrm{p}}^{\mathrm{T}}\boldsymbol{e}_{\mathrm{i}}^{\mathrm{T}}](\boldsymbol{S}^{+}\boldsymbol{Q})\begin{bmatrix}\boldsymbol{e}_{\mathrm{p}}\\ \boldsymbol{e}_{\mathrm{i}}\end{bmatrix}. \tag{39}$$

We can use $\boldsymbol{W}$ to represent $(\boldsymbol{S}^{+}\boldsymbol{Q})$:

$$\boldsymbol{W}=\begin{bmatrix}\kappa_0\boldsymbol{I}_6 & \boldsymbol{0}\\ \boldsymbol{0} & \lambda_0\boldsymbol{L}_{\mathrm{i}}\boldsymbol{L}_{\mathrm{i}}^{+}\end{bmatrix}. \tag{40}$$

Because $\boldsymbol{L}_{\mathrm{i}}\boldsymbol{L}_{\mathrm{i}}^{+}$ can be proved to be positive semidefinite (Chaumette and Hutchinson, 2006), matrix $\boldsymbol{W}$ is also positive semidefinite. Then the local asymptotic stability of our control rule can be ensured.

3.4 Using Kalman filter to correct the real-time position and orientation of the robot end

In the actual movement, the position and orientation of the gripper calculated using the data observed by the camera may be inaccurate. At this time, Kalman filter can be combined with the odometer itself provided by the robot to obtain a more accurate location:

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \boldsymbol{J}_k \cdot \left[\frac{\mathrm{d}\theta_1}{\mathrm{d}t} \ \frac{\mathrm{d}\theta_2}{\mathrm{d}t} \ \frac{\mathrm{d}\theta_3}{\mathrm{d}t} \ \frac{\mathrm{d}\theta_4}{\mathrm{d}t} \ \frac{\mathrm{d}\theta_5}{\mathrm{d}t} \ \frac{\mathrm{d}\theta_6}{\mathrm{d}t} \ \omega \ v\right]_k^{\mathrm{T}} \delta t + \boldsymbol{\omega}_k, \tag{41}$$

$$\boldsymbol{z}_k = \boldsymbol{H}_k \cdot \boldsymbol{x}_k + \boldsymbol{v}_k, \tag{42}$$

where $\boldsymbol{x}_k = (\delta x, \delta y, \delta z, \delta\phi_x, \delta\phi_y, \delta\phi_z)_k^{\mathrm{T}}$ is the position and orientation of the gripper in the world coordinate system. The above equations can be written as a standard Kalman equation:

$$\boldsymbol{x}_k = \boldsymbol{A}_k\boldsymbol{x}_{k-1} + \boldsymbol{B}_k\boldsymbol{u}_k + \boldsymbol{\omega}_k, \tag{43}$$

$$\boldsymbol{z}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k, \tag{44}$$

where $\boldsymbol{A}_k = \boldsymbol{I}_{6\times6}$ and $\boldsymbol{B}_k = \boldsymbol{J}_k\delta t$. In this way, we can make predictions:

$$\check{\boldsymbol{x}}_k = \boldsymbol{A}_k\hat{\boldsymbol{x}}_{k-1} + \boldsymbol{B}_k\boldsymbol{u}_k, \tag{45}$$

$$\check{\boldsymbol{P}}_k = \boldsymbol{A}_k\hat{\boldsymbol{P}}_{k-1}\boldsymbol{A}_k^{\mathrm{T}} + \boldsymbol{Q}_k. \tag{46}$$

Then perform position and orientation correction based on the data observed by the camera:

$$\boldsymbol{K}_k = \check{\boldsymbol{P}}_k\boldsymbol{H}_k^{\mathrm{T}}(\boldsymbol{H}_k\check{\boldsymbol{P}}_k\boldsymbol{H}_k^{\mathrm{T}} + \boldsymbol{R})^{-1}, \tag{47}$$

$$\hat{\boldsymbol{x}}_k = \check{\boldsymbol{x}}_k + \boldsymbol{K}_k(\boldsymbol{z}_k - \boldsymbol{H}_k\check{\boldsymbol{x}}_k), \tag{48}$$

$$\hat{\boldsymbol{P}}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\check{\boldsymbol{P}}_k. \tag{49}$$

The data obtained are smooth and continuous, and the visual servo control equation will be calculated to obtain a very smooth and stable velocity control rule.

The overall process of the algorithm in this study is shown in Fig. 3.

4 Experimental results

The mobile manipulator consists of a Bulldog chassis, a UR5 robotic arm, a Robotiq-85 gripper, and a ZED camera. The Bulldog chassis is a differential chassis, so its motion can be controlled by two physical quantities, velocity and angular velocity. The UR5 arm has six degrees of freedom. The Robotiq-85 holder has only one degree of freedom to open and close the gripper. The marker for visual recognition and positioning shown here is a whiteboard consisting of four square points located above the blue table. The location of the object to be grabbed is about 30 cm in front of the marker, to ensure that the camera can see the marker while the gripper is grasping the object.

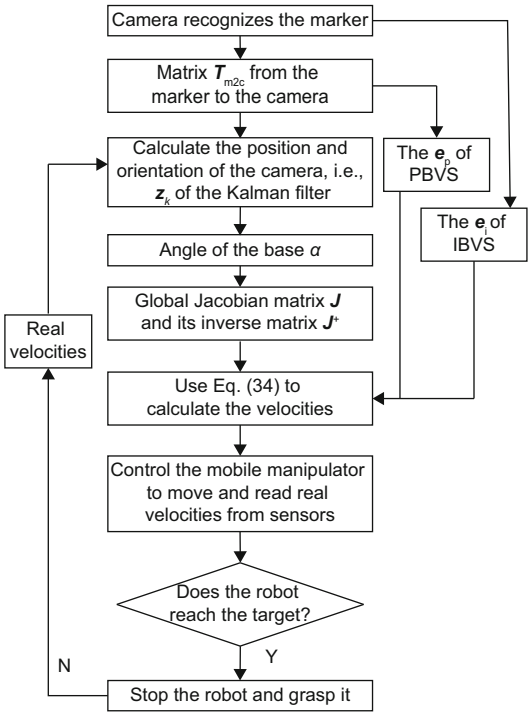

Fig. 3 The algorithm flowchart showing the process of hybrid visual servo (HVS) for mobile manipulation used in this study

4.1 Calculating positions and orientations of the camera and robot base

According to the visual servo control rule of the mobile manipulator (Eq. (34)), the robot needs only to observe the four feature points of the marker through the ZED camera, obtain their positions in the image coordinate system, and substitute them into the SolveP4P module to calculate the relative position and orientation of the camera to the target marker. The classical study has described the way of solving the perspective four-point problem in detail (Horaud et al., 1989). The positional and orientational relationship from the marker to the camera, which can be converted into a 4×4 homogeneous

matrix $\boldsymbol{T}_{\mathrm{m2c}}$, can be calculated by this method. $\boldsymbol{T}_{\mathrm{m2c}}$ belongs to Lie group SE(3). The so-called SE(3) is the group which contains all rotations and translations of the rigid body. The 3×3 matrix block at the top left of $\boldsymbol{T}_{\mathrm{m2c}}$ is the rotation matrix. The 3×1 matrix block at the top right of $\boldsymbol{T}_{\mathrm{m2c}}$ is the translation vector.

The homogeneous coordinate transformation matrix from the world coordinate system to the marker is known as a $\boldsymbol{T}_{\mathrm{w2m}}$ matrix (world to marker). The positional and orientational relationship between the camera and the gripper is obtained by hand-eye calibration to obtain a 4×4 hand-eye relationship matrix, the $\boldsymbol{T}_{\mathrm{c2e}}$ matrix (camera to end). Then, by the inverse kinematics of the manipulator, a homogeneous transformation matrix $\boldsymbol{T}_{\mathrm{e2b}}$ (end to base) from the gripper to the base of the arm is obtained. Finally, there is a 4×4 homogeneous transformation matrix $\boldsymbol{T}_{\mathrm{b2r}}$ (arm base to robot) from the robotic arm base to the robot base coordinate system. By multiplying the above five homogeneous coordinate transformation matrices, the coordinate transformation matrix $\boldsymbol{T}_{\mathrm{w2r}}$ (world to robot) from the world coordinate system to the local coordinate system of the robot can be obtained. That is, the positions and orientations of the robot and the gripper are obtained in the world coordinate system:

$$\begin{aligned}\boldsymbol{T}_{\mathrm{w2r}} =&\boldsymbol{T}_{\mathrm{w2m}}\cdot\boldsymbol{T}_{\mathrm{m2c}}\cdot\boldsymbol{T}_{\mathrm{c2e}}\cdot\boldsymbol{T}_{\mathrm{e2b}}\cdot\boldsymbol{T}_{\mathrm{b2r}}\\ =&\begin{bmatrix}\boldsymbol{R}_{\mathrm{w2m}} & \boldsymbol{t}_{\mathrm{w2m}}\\ \boldsymbol{0}_{1\times3} & \boldsymbol{1}\end{bmatrix}\cdots\begin{bmatrix}\boldsymbol{R}_{\mathrm{b2r}} & \boldsymbol{t}_{\mathrm{b2r}}\\ \boldsymbol{0}_{1\times3} & \boldsymbol{1}\end{bmatrix}\\ =&\begin{bmatrix}\boldsymbol{R}_{\mathrm{w2r}} & \boldsymbol{t}_{\mathrm{w2r}}\\ \boldsymbol{0}_{1\times3} & \boldsymbol{1}\end{bmatrix},\end{aligned}\tag{50}$$

$$\boldsymbol{T}_{\mathrm{w2e}}=\boldsymbol{T}_{\mathrm{w2m}}\cdot\boldsymbol{T}_{\mathrm{m2c}}\cdot\boldsymbol{T}_{\mathrm{c2e}}=\begin{bmatrix}\boldsymbol{R}_{\theta} & \boldsymbol{t}\\ \boldsymbol{0}_{1\times3} & \boldsymbol{1}\end{bmatrix},\tag{51}$$

where $\boldsymbol{R}_{\theta}$ is the rotation matrix of the gripper and $\boldsymbol{t}$ is the current position vector of the gripper which was described in Section 3.2. Assume that $\boldsymbol{R}_0$ is the target rotation matrix of the gripper. The rotation matrix between the current and target orientations can be represented as follows:

$$\delta\boldsymbol{R}=\boldsymbol{R}_{\theta}\cdot\boldsymbol{R}_0^{-1}.\tag{52}$$

Then, the angular vector of the rotation between the current and target orientations of the gripper $\theta\boldsymbol{u}$ can be calculated as follows:

$$\theta\boldsymbol{u}=\begin{bmatrix}\mathrm{atan}\left(\dfrac{\delta R(3,2)}{\delta R(3,3)}\right)\\ \mathrm{atan}\left(-\dfrac{\delta R(3,1)}{\sqrt{\delta R(3,2)^2+\delta R(3,3)^2}}\right)\\ \mathrm{atan}\left(\dfrac{\delta R(2,1)}{\delta R(1,1)}\right)\end{bmatrix}.\tag{53}$$

In this study, the homogeneous coordinate transformation obeys the principle of right multiplication. That is, the rotation part of the transformation is written in the form of the x-, y-, and z-axis unit column vectors of the transformed coordinate system with respect to the old coordinate system. The purpose of obtaining the $\boldsymbol{T}_{\mathrm{w2r}}$ matrix is to obtain not only the position of the robot chassis, but also the angle α of the robot in the world coordinate system. Under ideal conditions, $\boldsymbol{R}_{\mathrm{w2r}}$ should be $\boldsymbol{R}$ in Eq. (14), and $\boldsymbol{t}_{\mathrm{w2r}}$ can be written in terms of $[x_0,y_0,0]^{\mathrm{T}}$. So, α can be deduced as follows:

$$\alpha=\mathrm{atan}\left(\frac{R_{\mathrm{w2r}}(2,1)}{R_{\mathrm{w2r}}(1,1)}\right).\tag{54}$$

After obtaining the $\boldsymbol{T}_{\mathrm{w2e}}$ matrix, we can find the position and orientation of the gripper in the world coordinate system at every moment, and then find the difference between it and the target pose of grasping $[\delta x,\delta y,\delta z,\delta\phi_x,\delta\phi_y,\delta\phi_z]^{\mathrm{T}}$.

4.2 Comparison with image- and position-based visual servo algorithms

All three methods, HVS, PBVS, and IBVS, have been verified on our mobile manipulation platform. However, their performances are different. In the experiment, we set κ=λ=0.1 for the HVS method, λ=0.2 for the IBVS method, and κ=0.2 for the PBVS method. The three methods calculate the servo velocity to control the motion of the mobile manipulator in real time. The distances and displacements of the base and the gripper are slightly different, and the differences are not large. As shown in Table 1, the ratio of distance to displacement of IBVS, whether it is of the base or of the gripper, is the largest, the ratio of PBVS is the smallest, and the ratio of HVS is somewhere in between. The smaller the ratio of the distance to the displacement is, the smoother the path is. These indicate that the IBVS path is the most tortuous during the movement, the PBVS path

is the smoothest, and the HVS path is somewhere in between. This can also be seen very intuitively from Fig. 4. In Fig. 4, the green lines are the paths of PBVS, where the paths of the base and the gripper are very smooth, the blue lines show the paths of IBVS, where the path of the base is a broken line and the path of the gripper is relatively smooth, and the red lines are the paths of HVS with smoothness ranging between those of the previous two methods.

Fig. 5 shows the plots of velocities with respect to time obtained by the three visual servo control methods. Fig. 5a shows that the changes of velocities obtained by the IBVS method are relatively steep. At the beginning, the velocity of the base is 1.39 m/s. After a short time period, it is reduced to a small value, and the subsequent motion is slow. The whole process involves 310 iterations, and each iteration is equal to 0.1 s, so the total time of IBVS is 31 s. The changes of velocities of the PBVS method obtained in Fig. 5b are relatively smooth, slower than those of IBVS, and the velocity of the base undergoes a process of increasing first and then decreasing, reaching a peak at 0.7 s, and the peak is only 0.35 m/s, but it maintains a certain speed movement in the later stage. The entire process lasts 41 s (410 iterations). The changes of the velocities obtained by the HVS method with time are shown in Fig. 5c. The advantages of the IBVS and PBVS methods are combined in the HVS method. At the beginning, the base velocity of the HVS method reaches a peak value of 0.80 m/s, which is between the two. In the meantime, a large displacement is quickly moved, and then the base velocity of the HVS method maintains a small but a certain size of speed to converge toward

Table 1 Comparison of parameters of the three algorithms

Item	IBVS	PBVS	HVS
Displacement of the base (m)	1.8348	2.0020	1.7567
Distance of the base (m)	2.3442	2.3524	2.1396
Ratio of the actual path distance to the displacement of the base	1.2776	1.1750	1.2180
Displacement of the gripper (m)	1.9954	2.2932	1.8825
Distance of the gripper (m)	2.2313	2.4408	2.0566
Ratio of the actual path distance to the displacement of the gripper	1.1182	1.0644	1.0925
Time spent (s)	31	41	17.5
Largest velocity (m/s)	1.39	0.35	0.8

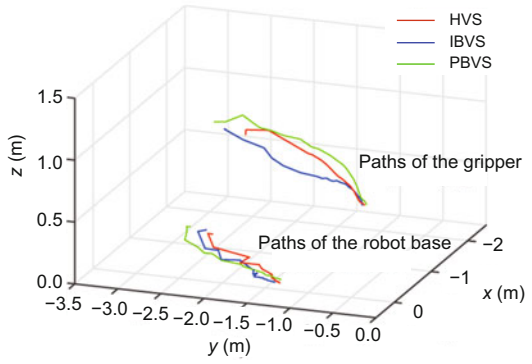

Fig. 4 Motion paths of the robot chassis and gripper in the real space of the three algorithms. References to color refer to the online version of this figure

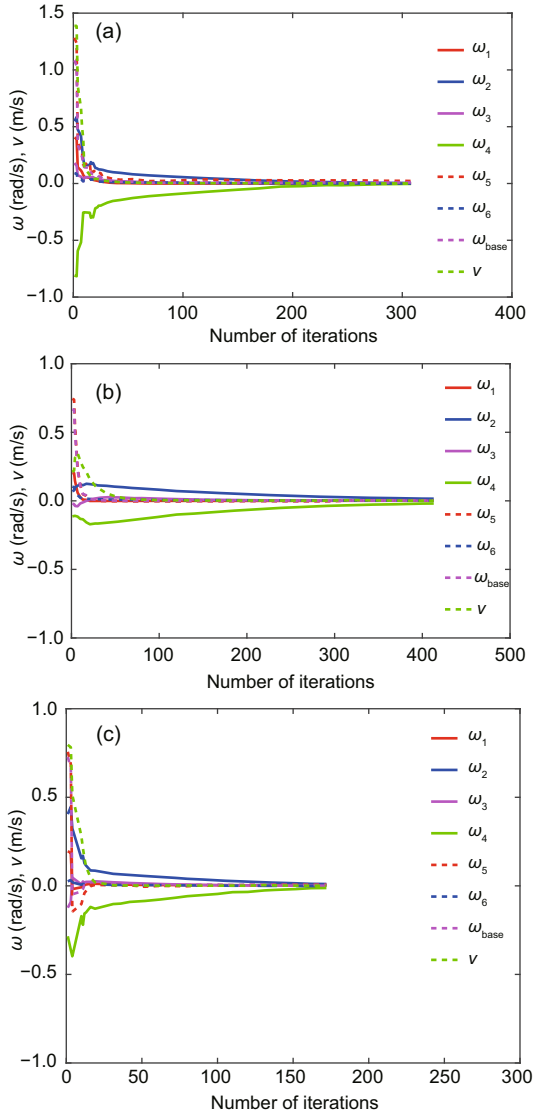

Fig. 5 Angular velocity or velocity of each degree of freedom of the robot as a function of time: planning results of the IBVS (a), PBVS (b), and HVS (c) algorithms

the target position. So, as shown in Fig. 5, the HVS method has a relatively short move time, lasting only 17.5 s (175 iterations), and quickly reaches the target position to grasp the object.

Fig. 6 is the graph showing the change of the distance between the grippers and the targets obtained by the three servo programming methods over time. The comparison shows that the gripper's position and orientation of the IBVS method converge more sharply than those of the PBVS method, and the fluctuation of the position change of the IBVS method is larger than that of the PBVS method. This is because the IBVS method aims only to make the positions of the feature points converge in the image coordinate system. So, it inevitably leads to the nonlinearity of the movement of the gripper in the real space. The HVS method combines the characteristics of these two methods, so its trajectory will also show some compromise. The convergence speed in the early stage will be higher than that of PBVS, but will be lower than that of IBVS.

Although the path of the robot base is tortuous, the motion of the IBVS algorithm is still very advantageous. As shown in Fig. 7a, the paths of four feature points in the image coordinate system obtained by the IBVS method are more convergent and smoother than the paths depicted by the PBVS method in Fig. 7b. This ensures that when the robot is moving, the marker's feature points are always in the field of view of the camera and are always in the center. The marker moves toward the target position, and there is not a large deviation from the line between the current and target positions. The

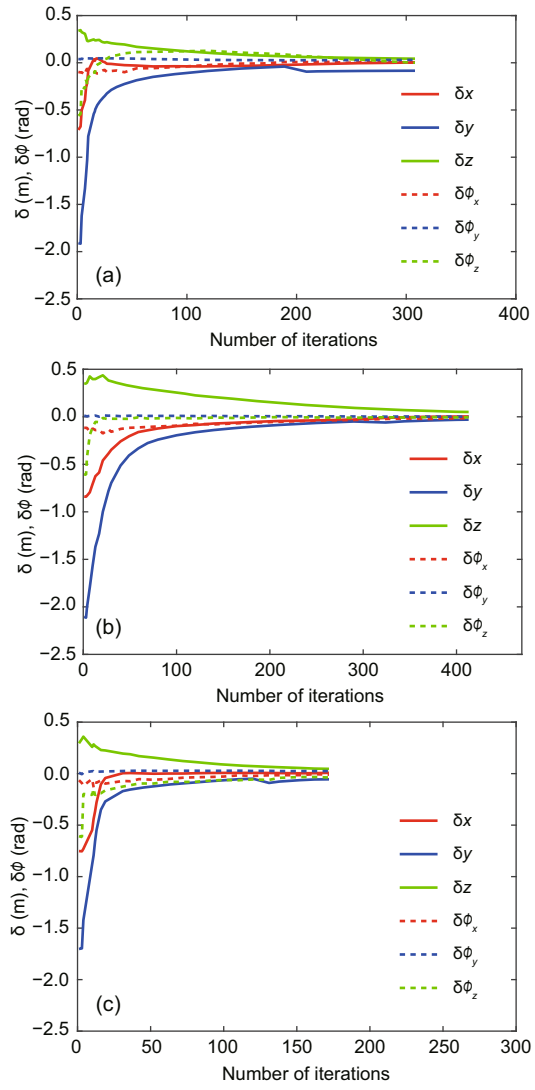

Fig. 6 Distance between the gripper and the target positions as a function of time: planning results of the IBVS (a), PBVS (b), and HVS (c) algorithms

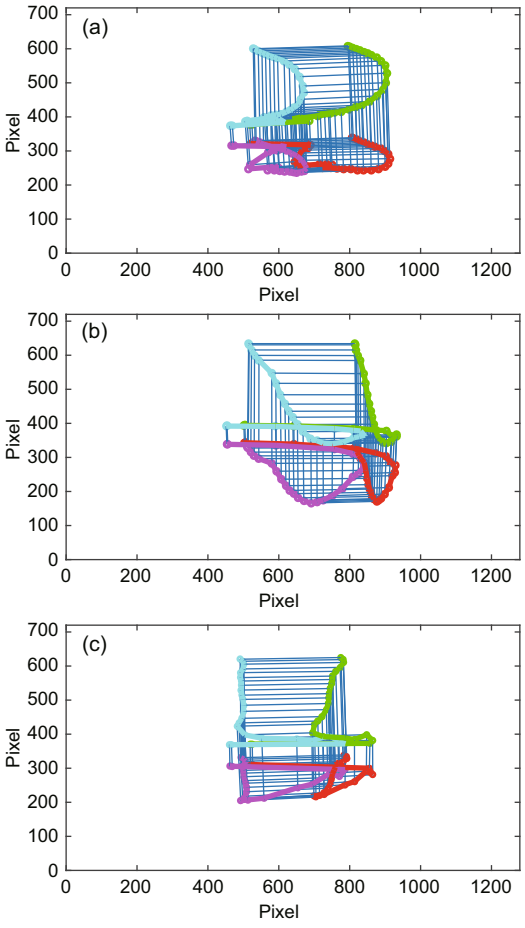

Fig. 7 Changes of positions of the four feature points on the camera image with time when the IBVS (a), PBVS (b), or HVS (c) algorithm is running

PBVS method will cause the paths of the marker's feature points to fluctuate greatly in the camera's field of view, especially at the beginning, because it considers only the smoothness of the robot path in the world coordinate system, and the paths of the feature points in the image coordinate system are not taken into account. It is not conducive to visual tracking of the target. The HVS method combines the advantages of the IBVS method well and effectively circumvents the shortcomings of the PBVS algorithm. As shown in Fig. 7c, the paths of the marker's feature points in the camera's field of view given by the HVS method show a little volatility at the beginning, and become smooth quickly. Some fluctuations are quickly corrected back to the position of the target within a straight-line advance toward the target position.

In summary, HVS combines the advantages of IBVS and PBVS, and the base path and gripper path of HVS are relatively stable and smooth. The paths of the marker's feature points moving in the camera's field of view are relatively gentle during the movement, always in the center of the field of view. This ensures that visual location information would not be lost during the exercise. Also, the HVS method captures the entire process and takes the shortest time, while PBVS takes the longest time.

4.3 Effect of Kalman filter and results of physical experimental verification

Since Kalman filter is used to optimize the estimated position of the robot, the angular velocity read from the robot odometer is combined with the position obtained by the ZED camera at the end of the robot arm. After using the Kalman filter, the position and orientation estimation of the robot gripper is precise, which can be used to calculate the distance from the target object and determine if the object can be grabbed. The effect of Kalman filter is shown in Fig. 8. After using the Kalman filter, the paths are less volatile and the curves are smoother. This is particularly evident in the Euler angles ϕ_x, ϕ_y, and ϕ_z.

Finally, we implement the HVS method on the physical platform, and the effect is shown in Fig. 9. The marker is placed on the chair and the mineral water bottle is placed about 30 cm in front of the marker. At the beginning, the mobile manipulator stops at about 3 m in right front of the marker, and the configuration of the robotic arm enables the marker to be in the field of view of the camera. When the program is running, the base and the robotic arm move toward the marker at the same time. At the beginning, the base moves very fast

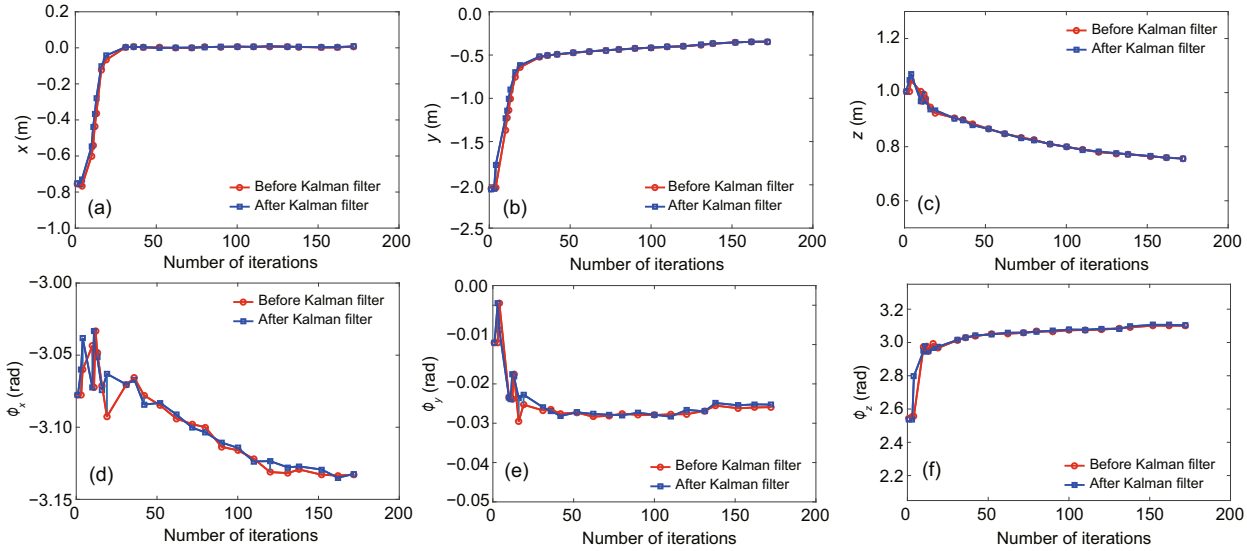

Fig. 8 Time-varying graph of pose data of the robot end gripper in the world coordinate system before and after using the Kalman filter: (a) end clamp data before and after filtering at the x-axis position; (b) end clamp data before and after filtering at the y-axis position; (c) end clamp data before and after filtering at the z-axis position; (d) end clamp data of the rotation angle before and after filtering in the x-axis direction; (e) end clamp data of the rotation angle before and after filtering in the y-axis direction; (f) end clamp data of the rotation angle before and after filtering in the z-axis direction

(about 0.8 m/s) and the robotic arm moves very slowly. However, when the gripper is close to the target bottle (<30 cm), the movement of the base becomes very slow (<1 cm/s), and the movement of the gripper toward the target is caused mainly by the robotic arm.

Fig. 9 Verification of the HVS method using a real robot

5 Conclusions

In this study we have proposed an HVS control method for the overall planning of the mobile manipulator while moving and grasping. This method uses the positions of four feature points recognized by the camera, and derives the rough positions and orientations of the robot gripper and the base. Combined with Kalman filter, the spatial positions and orientations of the gripper and the base are obtained more accurately. Using the global Jacobian matrix of the mobile manipulator and the HVS control rule derived in this study, the executive velocities of all degrees of freedom of the robot from the base to the arm at each moment are calculated in real time, to control the movement of the whole robot to the target object and complete the grasp action.

The method is different from the current visual servo methods that plan the base and the arm of a mobile manipulation separately, and obtains a good grasping effect. The method of this study is flexible. If the marker and the target object move, they will be tracked in real time, and this will not lead to the failure of the planning and will provide convenience for future industrial and domestic applications.

Contributors

Wei LI designed the research, conducted the experiments, and drafted the manuscript. Rong XIONG helped improve the idea and organize the manuscript. Wei LI and Rong XIONG revised and finalized the paper.

Compliance with ethics guidelines

Wei LI and Rong XIONG declare that they have no conflict of interest.

References

Agravante DJ, Chaumette F, 2017. Active vision for pose estimation applied to singularity avoidance in visual servoing. IEEE/RSJ Int Conf on Intelligent Robots and Systems, p.2947-2952. https://doi.org/10.1109/IROS.2017.8206129

Bateux Q, Marchand E, Leitner J, et al., 2017. Visual servoing from deep neural networks. https://arxiv.org/abs/170508940

Benhimane S, Malis E, 2007. Homography-based 2D visual tracking and servoing. *Int J Robot Res*, 26(7):661-676. https://doi.org/10.1177/0278364907080252

Chaumette F, Hutchinson S, 2006. Visual servo control. I. Basic approaches. *IEEE Robot Autom Mag*, 13(4):82-90. https://doi.org/10.1109/MRA.2006.250573

Chaumette F, Hutchinson S, 2007. Visual servo control. II. Advanced approaches. *IEEE Robot Autom Mag*, 14(1):109-118. https://doi.org/10.1109/MRA.2007.339609

Dong GQ, Zhu ZH, 2015. Position-based visual servo control of autonomous robotic manipulators. *Acta Astronaut*, 115:291-302. https://doi.org/10.1016/j.actaastro.2015.05.036

Fantacci C, Vezzani G, Pattacini U, et al., 2018. Markerless visual servoing on unknown objects for humanoid robot platforms. IEEE Int Conf on Robotics and Automation, p.3099-3106. https://doi.org/10.1109/icra.2018.8462914

Hafez AHA, Cervera E, Jawahar C, 2008. Hybrid visual servoing by boosting IBVS and PBVS. 3rd Int Conf on Information and Communication Technologies: from Theory to Applications, p.1-6. https://doi.org/10.1109/ICTTA.2008.4530116

Horaud R, Conio B, Leboulleux O, et al., 1989. An analytic solution for the perspective 4-point problem. *Comput Vis Graph Image Process*, 47(1):33-44. https://doi.org/10.1016/0734-189X(89)90052-2

Hu G, MacKunis W, Gans N, et al., 2008. Homography-based visual servo control via an uncalibrated camera. American Control Conf, p.4791-4796. https://doi.org/10.1109/ACC.2008.4587252

Kermorgant O, Chaumette F, 2011. Combining IBVS and PBVS to ensure the visibility constraint. IEEE/RSJ Int Conf on Intelligent Robots and Systems, p.2849-2854. https://doi.org/10.1109/IROS.2011.6094589

Lang HX, Khan MT, Tan KK, et al., 2016. Application of visual servo control in autonomous mobile rescue robots. *Int J Comput Commun Contr*, 11(5):685-696. https://doi.org/10.15837/ijccc.2016.5.2680

Levine S, Pastor P, Krizhevsky A, et al., 2016. Learning hand-eye coordination for robotic grasping with large-scale data collection. Int Symp on Experimental Robotics, p.173-184. https://doi.org/10.1007/978-3-319-50115-4_16

Sandy T, Buchli J, 2017. Dynamically decoupling base and end-effector motion for mobile manipulation using visual-inertial sensing. IEEE/RSJ Int Conf on Intelligent Robots and Systems, p.6299-6306. https://doi.org/10.1109/IROS.2017.8206533

Tsakiris DP, Rives P, Samson C, 1998. Extending visual servoing techniques to nonholonomic mobile robots. In: Kriegman DJ, Hager GD, Morse AS (Eds.), The Confluence of Vision and Control. Springer, London, p.106-117. https://doi.org/10.1007/BFb0109666

Wang Y, Zhang GL, Lang HX, et al., 2014. A modified image-based visual servo controller with hybrid camera configuration for robust robotic grasping. *Robot Auton Syst*, 62(10):1398-1407. https://doi.org/10.1016/j.robot.2014.06.003

Zhong XG, Zhong XY, Peng XF, 2015. Robots visual servo control with features constraint employing Kalman-neural-network filtering scheme. *Neurocomputing*, 151:268-277. https://doi.org/10.1016/j.neucom.2014.09.043