

## Research Article

# Vision Positioning-Based Estimation Method and Its Simulation Studies on State of Underwater Manipulator

Junli Wang <sup>1,2</sup>, Shitong Wang,<sup>1</sup> and Wenhao Leng<sup>3</sup>

<sup>1</sup>School of AI & Computer Science, Jiangnan University, Wuxi 214122, China

<sup>2</sup>Wuxi Vocational Institute of Commerce, Wuxi 214153, China

<sup>3</sup>China Ship Scientific Research Center, Wuxi 214082, China

Correspondence should be addressed to Junli Wang; wangjunli@wxic.edu.cn

Received 15 November 2020; Revised 27 December 2020; Accepted 9 February 2021; Published 25 February 2021

Academic Editor: M. Syed Ali

Copyright © 2021 Junli Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Work class remote operated vehicles (ROVs) are generally equipped with underwater manipulators and are widely used in underwater intervention and maintenance tasks. As the load of underwater operation is relatively heavy, most commercial underwater manipulators are hydraulically actuated and are not equipped with any sensor for joint angles to keep their architectures compact. Therefore, the automatic control methods widely used in industrial robots cannot be simply applied to underwater manipulators. In this paper, an estimation method on joint angles of manipulator is presented, in which several markers are arranged on the arm links and positioned from the corresponding cameras; consequently, the joint angles of the manipulator are estimated. The simulation results show that under typical optical vision positioning error (RMS: 5 mm), the positioning error of the end effector can be estimated as about 10 mm (RMS), which means that the proposed estimation method is feasible for the state estimation for automatic control of underwater manipulators.

## 1. Overview

In underwater investigation, construction, and maintenance, underwater manipulators are considered to be the most suitable tools. Work class ROVs are usually equipped with underwater manipulators, which are remotely controlled by experienced operators [1, 2]. Generally, when working under water, at least two skilled operators are required: one pilots the ROV, so as to keep it as stable as possible by compensating for external motion disturbances, while the other operator performs the actual teleoperated manipulation task [3]. Compared with industrial manipulators, underwater manipulators have higher requirements on actuating load, and most commercial underwater manipulators are actuated hydraulically [1, 4]. Hydraulic systems have higher power to weight ratio, much larger than electric equipments. Hydraulic manipulators are almost controlled in two main kinds of control methods: rate control and position control. In rate control, directional control valves switch the hydraulic oil passing through the valve and then control the

joint turning speed, which actually forms the open-loop control. Whilst in position control, the joint angles are measured by position sensors and then are used to control servo valves for realization of closed loop joint position control for the manipulators. In recent years, aiming at some special fields such as archaeology, underwater manipulators with force feedback control have been specially developed to realize some fine underwater operations [5, 6]. However, in the mainstream commercial products, the most popular is the compact underwater manipulator which can only be operated in joint rate control mode without joint angle sensors.

Since the operations of ROVs and underwater manipulators require strong professional skills and take up a lot of labor intensity, in order to better complete the underwater investigation tasks, a lot of attention has been paid on the study on the automatic control of underwater manipulator. Many scholars have proposed the motion control algorithms by using adaptive control, neural network, fuzzy logic, and so on [7–9]. What is more, motion control has been well

implemented in industrial manipulators (robots) and has been widely used in industrial applications. However, these methods are not suitable for underwater manipulators, because sensing of joint states of underwater manipulators is poor than those for industrial manipulators.

As for an underwater manipulator system, a more important work to realize the automatic control is to realize its position state measurement of the manipulator. It is the most direct method to arrange the angle sensor at the arm joint. Due to the compact structure of the underwater manipulator, it is difficult and infeasible for its engineering implementation. In order to avoid the sensing problem, some scholars selected the electric manipulator [10], or just assumed that the joint angles of the manipulator have been measured. However, electric underwater manipulators are often made as prototypes for research purposes but are rarely used in commercial applications [11, 12]. Therefore, such a problem has not been solved yet in engineering application.

As a result, it is a popular choice to solve relative position between ROV and manipulators or manipulated targets indirectly by locating the feature markers in optical images [13]. For example, Kawamura et al. [14] presented a more ingenious method, in which several marker balls were arranged on the manipulator. The relative position between the marker balls and the target was taken as the control object; thus, the calibrating procedure of the optical servo system was omitted. With this method, the displacement between the manipulator and the target could be measured accurately in the coordinate system fixed to ROV body, where the camera is fixed. When the manipulator has more joint DOFs (degrees of freedom), only with the displacement between the end effector and the target, joint angles of the manipulator could not be determined, and therefore the control value of joint angles could not be determined consequently. Peñalver et al. [15] arranged cameras respectively on the last arm link of the manipulator and the main body of ROV, and the marker object was taken as photograph by the two cameras. After recognizing the markers in images, the coordinate transformation matrix from the marker objects to the arm link of the manipulator and ROV could be determined respectively, and consequently, the transformation from the arm link to ROV could be solved by simple matrix operation. Similar optical positioning methods are also used in automatic control during ROV docking [16]. Mangipudi and Li [17] focused on the accuracy of multipoint perspective problem and proposed a gradient descent least square algorithm to estimate pose accurately and efficiently. After optimization, the calculation amount of this algorithm was greatly reduced and had been implemented on a low-cost single-board computer. Positional accuracy of the order of 1–2.5 mm while the landmark is approximately 1 m away has been demonstrated. If the general solution of the multipoint perspective problem (PNP) is directly applied to the manipulator positioning, at least 4 key points need to be arranged on one arm link, and no occlusion is required during the movement. This becomes very difficult in practice. Therefore, an effective optical positioning method should be developed according to the arm links of the

manipulator, such as the motion constraints of the multiarm joint.

In this paper, a commercial manipulator (model HLK-HD6W) equipped on an ROV is studied. After locating the marker points of the manipulator based on the optical image, the joint angles of each arm link of the manipulator is estimated, and therefore, the foundation of automatic control underwater hydraulic manipulator in Cartesian space is established. In Section 2, manipulator model HLK-HD6W is introduced and the forward kinematics is reviewed. In Section 3, the scheme on vision positioning is presented, and consequently, the method is presented to estimate the joint angles of the arm links with the positions of the markers on the arm links in images as inputs. In Section 4, simulation is carried out to verify the effectiveness of the algorithm presented in Section 3. Finally, Section 5 concludes the paper.

## 2. On Motion of Underwater Manipulator

The manipulator HLK-HD6W consists of 5 rotational joints without any translational joint, which is illustrated in Figure 1. Its end effector has 5 DOFs relative to the base of the manipulator, which meets the motion requirements of 5 DOFs. Hydraulic power makes this manipulator very compact with large load capacity. Meanwhile, there is no any sensor installed on the manipulator to measure the joint angle of each arm link. At present, the robot arm is mainly used in small and medium ROVs. The operator remote operates the manipulator by observing the television video which takes the operated object and manipulator in viewpoint. Consequently, it also greatly limits the efficiency of underwater operations. In order to implement the automatic control of the manipulator, the first task is to realize the sensing of the arm motion states. The motion parameters of each arm are estimated indirectly, and then the position of the end effector in space is determined.

As there are only 5 DOFs in the movement of the end effector of the manipulator, the operation of the end effector is limited. Without moving the installation foundation of the manipulator by moving ROV, it is impossible for the end effector to grasp the target in any posture (or direction). Therefore, after the manipulator motion controller receives the moving target of the end effector, it is necessary to analyze whether the position is reachable and how to move from the current position to the target position. According to the analysis of DOFs, the end effector can realize 5-DOF motion, which just corresponds to rod-shaped objects such as cables, ignoring the DOF of rotation around the axis of the rod. Therefore, the end effector can grasp rod-shaped object in a special direction. This study focuses on how to obtain the joint state of the manipulator by indirect measurement and how to determine the position of the end effector, which is the foundation for planning and controlling the spatial motion and operation of the end effector of the manipulator.

In order to describe the motion of the manipulator conveniently, a local coordinate system is established on

each joint link. Based on cascaded coordinate systems, the position of the end effector in global coordinate system can be easily expressed. Using Denavit-Hartenberg (D-H) parameter and its convention for local coordinate system [18], the coordinate system of each joint arm link is described relative to its upper coordinate system. For the  $i$ -th arm link, the local coordinate system  $Ox_i y_i z_i$  is established and fixed on it, by the following steps: firstly, taking the rotation axis of the next link as  $z_i$  axis; secondly, taking the coordinate system  $Ox_{i-1} y_{i-1} z_{i-1}$  as the original frame, rotating the frame  $\theta$  about  $z_{i-1}$  axis, called as joint angle, so that  $x_{i-1}$  axis reaches a new position and intersect  $z_i$  axis, notated as  $x'_{i-1}$ ; thirdly, translating the frame  $d_i$  through  $z_{i-1}$ , called as link offset, so that  $x'_i$  reaches the origin point of new coordinate system, taken as final  $x_i$  axis; fourthly, translating the frame  $a_i$  through  $x_i$ , called as link length, so that  $z_{i-1}$  reaches the origin point of new coordinate system, notated as  $z'_i$ ; and finally, rotating the frame  $\alpha_i$  about  $x_i$  axis, called as twist angle, so that  $z'_i$  reaches the final  $z_i$  axis. In a rotating

manipulator, the joint angles  $\theta_i$  denote the variables in motion, while in a translational manipulator, the link offsets  $d_i$  are the variables.

In this way, local coordinate system on every arm link of the manipulator HLK-HD6W can be established, which is shown in Figure 2, where frame No. 0 is the global coordinate system, which is fixed on the mounting base of the manipulator, and frame No. 1 will rotate around the  $z$  axis of this frame. The frame No. 5 is fixed on the end effector of the manipulator, and the origin of frame No. 5 is located at the center of the gripper, which clamps the rod object in the  $x$  axis. D-H parameters of the manipulator HLK-HD6W are measured and shown in Table 1, where the joint angles  $\theta_i$  are variables during motion, and their values shown in Table 1 are the ones when the manipulator is in the extended state.

According to the definition of D-H parameters, the transformation matrix of the  $i$ -th local coordinate system can be described as

$$\mathbf{R}_i^{i-1}(\theta_i) = \mathbf{R}_z(\theta_i) \mathbf{T}_z(d_i) \mathbf{T}_x(a_i) \mathbf{R}_x(\alpha_i), \quad (1)$$

$$\begin{bmatrix} c_\theta & -s_\theta & 0 & 0 \\ s_\theta & c_\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\alpha & -s_\alpha & 0 \\ 0 & s_\alpha & c_\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_\theta & -s_\theta c_\alpha & s_\theta s_\alpha & a c_\theta \\ s_\theta & c_\theta c_\alpha & -c_\theta s_\alpha & a s_\theta \\ 0 & s_\alpha & c_\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where the symbols  $s_x, c_x$  denote the sine and cosine of the subscript quantities.

Cascaded transformation matrix from the end effector ( $n$ -th level) to the base of manipulator (0-th level) can be described as

$$\mathbf{r}_0 = \mathbf{R}_n^0(\theta_1, \theta_2, \dots, \theta_n) \mathbf{r}_n = \mathbf{R}_1^0(\theta_1) \cdot \mathbf{R}_2^1(\theta_2) \dots \mathbf{R}_n^{n-1}(\theta_n) \mathbf{r}_n, \quad (3)$$

where  $r_0, r_n$  denote the vector values of the same position vector in the global coordinate system and the local coordinate system of the  $n$ -th link.

When the joint angles (motion state variables)  $\theta_i$  are determined, the positions of the arm links are determined consequently. The motion control of the manipulator is carried out on the joint angle of each arm link; that is, the solution of each joint angle should be given according to the given position or motion track of the end effector, and the arm links should be driven to the corresponding joint angles by hydraulic actuator. However, for the commercial underwater manipulator such as HLK-HD6W, the position sensors actually conflict with the size of the manipulator, and there is no position sensor equipped to measure the joint angles so as to keep its size compact. Therefore, in order to realize the automatic control of the underwater manipulator, a method to estimate joint angles should be explored under the premise of keeping the compactness of the manipulator.

### 3. The Proposed Estimation Method

When ROV works under water, several cameras are usually arranged on ROV to observe underwater objects and environments. At the same time, the manipulator operator also observes the manipulator and its position relative to the object through video and then teleoperates the manipulator. Therefore, it is natural to arrange markers on the manipulator, and then, take photograph of these markers from different angles, recognize and locate the markers from the video images, and finally, solve the joint angles state of the manipulator in order to realize the indirect measurement of the manipulator states.

**3.1. Scheme on Vision Positioning.** As the DOF of the end effector (the 5th arm link) of the manipulator HLK-HD6W rotates around its longitudinal axis, the displacement of the arm joint caused by the motion of this DOF is very small. The structure of the end effector is also relatively small, which tends to be covered by either itself or the operated object if the arranged marker is small or tends to interfere with the operated object if the marker is big. In order to avoid such case, other methods should be expected to measure the motion of the DOF of the end effector. In this study, the estimation of motion state of the first four arm links is considered; that is said, 4 DOFs are investigated.

When a single marker is imaged in the camera, there are 2 constraints to DOFs of this marker. So, ideally, if 2 marker

points on different arm links are observed by a single camera, there are enough constraints to solve the joint angles of the manipulator. If one more camera is used, a single marker is perhaps imaged in multiple cameras. Although each image provides two constraints to the DOFs, there are up to three constraints to DOFs for the whole system, since the marker point has only three DOFs in space. In the same way, the first arm link has only one rotational DOF, and imaging the marker point on this arm link can only provide at most one effective constraint to the DOF. Considering that the first and second arm links of the manipulator are relatively short and small while the third and fourth ones are relatively long such that there is enough space to arrange the markers, it is very suitable to set two markers on the third and fourth arm links to measure the joint angles of the manipulator.

In order to suit the more complex underwater manipulation scenes, it is necessary to ensure that sufficient number of markers should be observed simultaneously even when some markers are covered by the operated objects or the manipulator itself in some directions. Several cameras will be arranged in the front of ROV to observe the whole working space of the manipulator in several directions. It should be pointed out that although a single marker point appears in multiple cameras, it can only provide at most 3 effective DOF constraints at most, and all the remaining constraints become redundant. Even so, these redundant constraints can effectively be used to reduce observation errors.

Referencing to the basic concepts of the manipulator, the 3rd and 4th arm links have 3 and 4 DOFs, respectively, and the DOFs of the 4th arm links contain all the ones of the 3rd arm links. Therefore, as for the four markers, even if only two markers can be observed effectively, which are not all on the third arm link, at least four DOFs can be determined, which is enough for us to effectively determine the positions of the first 4 arm links of the manipulator.

Table 2 lists the location parameters of markers in local coordinate systems, which is illustrated in Figure 2.

In order to observe all these marks, two cameras, of which the lateral viewing angle is about  $63^\circ$ , are arranged on the left and right above the mounting base of the manipulator. The location and azimuth parameters of the cameras in the global coordinate system are given in Table 3. Initially, the optical axis of the cameras is located along the  $x$  axis of the coordinate system, and then the cameras are rotated in the yaw and pitch angles shown in Table 3, such that the cameras point towards the center of working area of the manipulator and cover the working range of the manipulator as much as possible.

With the known joint angles of every arm link of the manipulator, by using the forward kinematics of manipulator, it is very convenient to determine the position of the marker located on the arm links in the global coordinate system. And then, using the layout parameters of the cameras in the optical imaging system, the positions of the markers in images by each camera can be easily obtained. Therefore, as an inverse problem, the nonlinear least squared method can be taken to estimate the parameters. That is to

say, firstly, the values of the joint angles of the manipulator are initialized, and then the position of each marker in camera is calculated using the forward kinematics. After the differences between the estimated positions and the observed positions in images are calculated, the nonlinear least squared algorithm is used to solve the corresponding problem, and the appropriate values of the joint angles can be accordingly obtained after some iterations.

As described in the previous section, after establishing the local coordinate system on the arm link, the  $j$ -th marker is arranged at the  $k_j$ -th arm link, whose position is denoted as the vector  $r_{k_j,j}$  in the corresponding local coordinate system and as the vector  $r_{0,j}$  in the global coordinate system. Denoting the joint angles of the first four arm links of the manipulator as  $\theta_i$ ,  $i = 1, 2, 3, 4$ , and the image position of the  $j$ -th marker in the  $c$ -th camera as  $p_{c,j}$ ,  $p_{c,j}$  can be determined by adopting the classical image processing techniques with the images from the cameras; meanwhile,  $p_{c,j}$  may also be estimated by  $\theta_i$  and  $r_{k_j,j}$  with forward kinematics, denoted as  $\hat{p}_{c,j}$ , i.e.,

$$\theta_{i|i=1,2,\dots,n} = \arg \min \Psi(\theta_1, \dots, \theta_n) = \arg \min \sum_{c,j} \frac{1}{2} (\hat{p}_{c,j} - p_{c,j})^2. \quad (4)$$

Please note that  $p_{c,j}$  in equation (4) denotes the position of the point in the image, which is a 2-dimensional vector, and the distance between any two points can be expressed as the squared sum of the differences of the corresponding components of the vectors of two points.

Nonlinear least squared method is a very popular method and becomes available on hand from the library [19]. In generally, in order to use the on-hand nonlinear squared method, the objective function and its partial derivatives (i.e., Jacobian matrix) should be provided.

*3.2. Forward Kinematics for the Positions of the Markers in Images.* The position of the markers on arm links in the corresponding local coordinate system is denoted as  $r_{k_j,j}$  ( $j = 1, 2, \dots, m$ ), where  $k_j$  is the sequence number of the arm link in which the marker is located. Therefore, according to equation (3), the transformation from the local coordinate system to the global coordinate system may be expressed as

$$\hat{\mathbf{r}}_{0,j} = \left[ \prod_{i=1}^{k_j} \mathbf{R}_i^{i-1}(\theta_i) \right] \mathbf{r}_{k_j,j}, \quad (5)$$

where the symbol  $\hat{\phantom{x}}$  means that the quantity is derived from the joint angles of the arm links but not measured directly by the visual images or other measurements.

In a visual imaging system, a local coordinate system is established for every camera, where the origin of the coordinate system is put in the optical center of camera, and the  $z$ -axis meets the optical axis and points to observed objects and  $x$ -axis in right. Now, the transformation matrix and the offset vector from the local coordinate system of the camera to the global coordinate system are denoted,

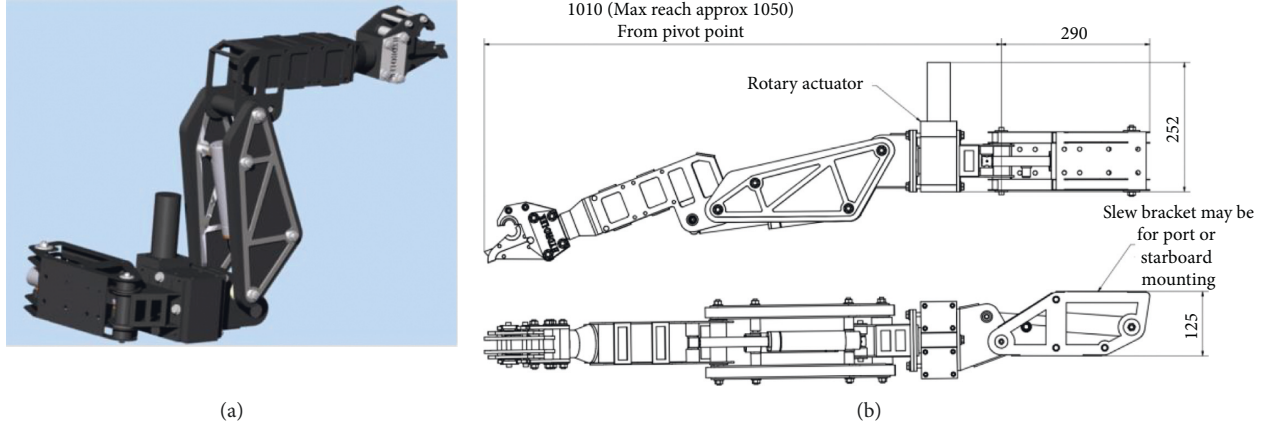


FIGURE 1: Manipulator model HLK-HD6W.

respectively, as  $R_{nh,c}, t_c$  ( $c = 1, 2, \dots, C$ ). Therefore, the homogeneous transformation matrix from the local coordinate system to the global coordinate system may be expressed as

$$\mathbf{R}_c = \begin{bmatrix} \mathbf{R}_{nh,c} & \mathbf{t}_c \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}. \quad (6)$$

Based on equation (5), the position of the  $j$ -th marker in image from the  $c$ -th camera can be determined:

$$\mathbf{r}_{(c,j)} = \mathbf{R}_c^{-1} \mathbf{r}_{0,j} = \begin{bmatrix} \mathbf{R}_{nh,c}^T & -\mathbf{R}_{nh,c}^T \mathbf{t}_c \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \mathbf{r}_{0,j}, \quad (7)$$

where  $r_{0,j}$  is the position of the  $j$ -th marker in the global coordinate system.

With the pinhole camera model, the position of the 3-D point in camera local coordinate system,  $r_{(c,j)} = \{x_{(c,j)}, y_{(c,j)}, z_{(c,j)}, 1\}^T$ , may be mapped into  $p_{(c,j)} = \{\bar{x}_{(c,j)}, \bar{y}_{(c,j)}, 1\}^T$  in the image plane of camera, where the first 2 components are the coordinate values in the image coordinate system, and

$$\begin{aligned} \hat{\bar{x}}_{(c,j)} &= \frac{x_{(c,j)}}{z_{(c,j)}}, \\ \hat{\bar{y}}_{(c,j)} &= \frac{y_{(c,j)}}{z_{(c,j)}}. \end{aligned} \quad (8)$$

Therefore, after the joint angles of the arm links are determined, the positions of every marker in the image planes of the cameras can be estimated. If there are enough images of the markers in the visual imaging system, the differences of the estimated positions and the measured positions of the markers in the image planes can be calculated and can be accordingly used to get the optimal joint angles of the arm links of the manipulator. That is to say,

$$\begin{aligned} \Psi(\theta_1, \dots, \theta_n) &= \frac{1}{2} \sum_{(c,j)} \left[ \hat{\mathbf{p}}_{(c,j)}(\theta_1, \dots, \theta_n) - \mathbf{p}_{(c,j)} \right]^2 \\ &= \frac{1}{2} \sum_{(c,j)} \left[ (\hat{\bar{x}}_{(c,j)} - x_{(c,j)})^2 + (\hat{\bar{y}}_{(c,j)} - y_{(c,j)})^2 \right], \end{aligned} \quad (9)$$

where the subscript  $\cdot_{(c,j)}$  means the  $j$ -th marker in the  $c$ -th camera. In a multieye visual imaging system, due to possible occlusions among objects, some markers are occasionally invisible in some cameras. In other words, the subscript  $\cdot_{(c,j)}$  may not traverse all combinations.

According to the nonlinear least squared method, besides the object function  $\hat{p}_{(c,j)}$ , its partial derivatives with respect to the independent variable  $\theta_i$  should be derived. From equation (8), it can be found that the partial derivatives of the positions  $\hat{p}_{(c,j)}$  in the image are determined by  $\partial r_{(c,j)} / \partial \theta_i$  depending on the partial derivatives of the positions of the markers in the camera local coordinate system with respect to independent variable.

$$\frac{\partial \mathbf{r}_{(c,j)}}{\partial \theta_i} = \left( \frac{\partial x_{(c,j)}}{\partial \theta_i}, \frac{\partial y_{(c,j)}}{\partial \theta_i}, \frac{\partial z_{(c,j)}}{\partial \theta_i}, 0 \right)^T. \quad (10)$$

Furthermore, the Jacobian matrix can be expressed by the partial derivatives with respect to the positions:

$$\begin{aligned} J_{(c,j,x),i} &= \frac{\partial \hat{\bar{x}}_{(c,j)}}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \left( \frac{x_{(c,j)}}{z_{(c,j)}} \right) = \frac{1}{z_{(c,j)}} \frac{\partial x_{(c,j)}}{\partial \theta_i} - \frac{x_{(c,j)}}{z_{(c,j)}^2} \frac{\partial z_{(c,j)}}{\partial \theta_i}, \\ J_{(c,j,y),i} &= \frac{\partial \hat{\bar{y}}_{(c,j)}}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} \left( \frac{y_{(c,j)}}{z_{(c,j)}} \right) = \frac{1}{z_{(c,j)}} \frac{\partial y_{(c,j)}}{\partial \theta_i} - \frac{y_{(c,j)}}{z_{(c,j)}^2} \frac{\partial z_{(c,j)}}{\partial \theta_i}. \end{aligned} \quad (11)$$

According to equation (7), the partial derivatives of the positions of markers in a camera local coordinate system will be transformed into the partial derivatives of the positions in the global coordinate system. In the physical sense, the derivative with respect to the independent variable  $\theta_i$  means

$$\mathbf{R}_z(\theta_i) + d\mathbf{R}_z(\theta_i) = \mathbf{R}_z(\theta_i + d\theta_i) = \begin{bmatrix} \cos d\theta_i & -\sin d\theta_i & 0 & 0 \\ \sin d\theta_i & \cos d\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & -s_\theta & 0 & 0 \\ s_\theta & c_\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (12)$$

As for  $d\theta_i$ , we have  $\lim_{d\theta_i \rightarrow 0} \sin d\theta_i = d\theta_i$ ,  $\lim_{d\theta_i \rightarrow 0} \cos d\theta_i = 1$ . Therefore, we have

$$d\mathbf{R}_z(\theta_i) = \mathbf{R}_z(\theta_i + d\theta_i) - \mathbf{R}_z(\theta_i) = d\theta_i \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_\theta & -s_\theta & 0 & 0 \\ s_\theta & c_\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (13)$$

For the  $i$ -th arm link, all the other components of the transformation matrix are independent on the joint angle. Therefore, the full partial derivative of the transformation matrix to the joint angle may be expressed as

$$\frac{\partial \mathbf{R}_i^{i-1}(\theta_i)}{\partial \theta_i} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{R}(\theta_i) = \mathbf{D}_z \mathbf{R}(\theta_i). \quad (14)$$

Substituting the former equation into equation (5), we will get the full partial derivatives of the positions of the markers in the global coordinate system to the joint angles of the manipulator:

$$\frac{\partial \mathbf{r}_{0,j}}{\partial \theta_i} = \left[ \prod_{p=1}^{i-1} \mathbf{R}_p^{p-1}(\theta_p) \right] \mathbf{D}_z \mathbf{R}_i^{i-1}(\theta_i) \left[ \prod_{p=i+1}^{k_j} \mathbf{R}_p^{p-1}(\theta_p) \right] \mathbf{r}_{k_j,j}. \quad (15)$$

In equation (15), the lower limit of the cumulative multiplication operator is larger than the upper limit; that is, the cumulative multiplication does not exist. And when  $k_j < i$ , equation (15) is invalid, and all the components of the partial derivatives is null, i.e.,  $\partial \mathbf{r}_{0,j} / \partial \theta_i = \mathbf{0}_{3 \times 1}$ .

During the motion of the manipulator, the installation parameters of the cameras are kept unchangeable, and therefore, according to equation (7) and the chain rule for the derivatives, the partial derivatives of the positions in the local coordinate system with respect to joint angles may be expressed as

the variation of the transformation matrix of the coordinate systems derived from the small change of the joint angle of the  $i$ -th arm link. That is to say, in equation (2), the variation of the component  $R_z(\theta_i)$  derived by a small change of  $\theta_i$  can be expressed as

$$\frac{\partial \mathbf{r}_{(c,j)}}{\partial \theta_i} = \mathbf{R}_c^{-1} \left[ \prod_{p=1}^{i-1} \mathbf{R}_p^{p-1}(\theta_p) \right] \mathbf{D}_z \cdot \mathbf{R}_i^{i-1}(\theta_i) \left[ \prod_{p=i+1}^{k_j} \mathbf{R}_p^{p-1}(\theta_p) \right] \mathbf{r}_{k_j,j}. \quad (16)$$

Together with equation (11), the partial derivatives of the object function  $p_{(c,j)}$  with respect to the joint angles of the arm links, i.e., the called Jacobian matrix, are determined, and can accordingly be taken to estimate the optimal joint angles.

**3.3. The Proposed Algorithm.** Base on the basic ideas of the above algorithm, we can establish the flow chart to solve the joint angles of the arm links shown in Figure 3. Firstly, initialize the parameters of the proposed algorithm, including the parameters of the manipulator, camera (including the layout and internal parameters), and arrangement parameters of the markers on the arm links. Secondly, run a server loop to provide the joint angles to the consequent control procedure in the following steps: acquire the images from the cameras and get the positions of the markers from images, setup initial values for the joint angles and solve the optimal joint angles in iteration with nonlinear least-squares fitting, and output the joint angles to the consequent control procedure. During iterations, the nonlinear least-squares fitting algorithm refers to the subprogram provided by GSL (GNU Scientific Library) library [19], where the objective function and Jacobian matrix calculation subroutine provided by user are called during the iterative solution.

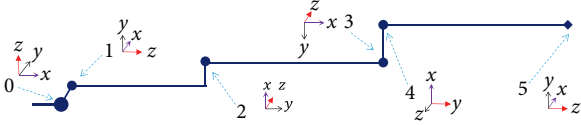


FIGURE 2: Local coordinate systems on every arm link of the manipulator HLK-HD6W.

TABLE 1: D-H parameters of the manipulator HLK-HD6W.

$i$	$\theta_i$	$d_i$ (mm)	$a_i$ (mm)	$\alpha_i$
1	$\pi/2$	0	8	$\pi/2$
2	$\pi/2$	232	34	$\pi/2$
3	$\pi/2$	0	385	0
4	$-\pi/2$	0	82	$-\pi/2$
5	$-\pi/2$	367	0	0

TABLE 2: Locations of the markers on the arm links of the manipulator.

Point no.	Link no.	$x$ (mm)	$y$ (mm)	$z$ (mm)
1	3	-300	-100	0
2	3	0	0	-80
3	4	100	0	0
4	4	0	-100	300

TABLE 3: Location parameters of the cameras.

Cam. No.	$x$ (mm)	$y$ (mm)	$z$ (mm)	Roll ( $^\circ$ )	Pitch ( $^\circ$ )	Yaw ( $^\circ$ )
1: Left	0	500	1200	-45	65	-35
2: Right	0	0	1200	0	65	0

## 4. Simulation Results

In order to verify the proposed method on estimating the joint angles of the arm links by visual imaging, the simulation study is carried out. Firstly, a motion process of the manipulator is preset, in which some joint angles change in turn, the trajectory of the marker on the arm links in space is synthesized with the forward kinematics method of the manipulator, and the trajectory of the marker in images is determined by being transformed to camera local coordinate system and projected to the image plane. Secondly, using this trajectory in images as the input, the joint angles are estimated using the method described in the previous section. Compared with the preset joint angles, the effectiveness of the joint angles estimation algorithm is analyzed. At the same time, noise is added to the synthetic trajectory to represent the positioning error of the camera images, and the joint angles are estimated by this trajectory with errors as the input and are compared with the preset joint angles. As a result, the influence of the camera image positioning error on the joint angles estimation is analyzed.

The simulation study is carried out on the platform Ubuntu 1804. All the experiments are realized with CPython 3.6, and the Numpy library is called for numerical calculation, and the GSL library is used for nonlinear least-squares

fitting. GSL is a scientific computing library developed by C language. In our experiments, the python-cffi library is used to call subroutines in the GSL library from Python language and pass subroutine in Python language as the callback function to library in C language.

**4.1. Simulation Setups and Results.** The manipulator is set to move according to the following procedure. It starts with the shrinking state  $\theta = \{(\pi/2), 0, -(\pi/2), (3\pi/4), 0\}^T$ , extends through  $\theta_1: (\pi/2) \rightarrow 0$  first, it extends through  $\theta_3: -(\pi/2) \rightarrow 0, \theta_4: (3\pi/4) \rightarrow 0$  simultaneously to the max extension, and then turns through  $\theta_2: 0 \rightarrow (\pi/2)$ , finally shrinks through  $(\theta_3, \theta_4)$  simultaneously, and shrinks through  $\theta_2: (\pi/2) \rightarrow (\pi/6), \theta_1: 0 \rightarrow (\pi/2), \theta_2: (\pi/6) \rightarrow 0$  in order to go back the original state. Every joint angle changes with time shown in Figure 4.

After the joint angles of the arm links are determined, the positions of the marker in images from cameras can be calculated. The positions of the marker No. 4 in images from the left and right cameras varying with time during motions are shown in Figure 5, where the position is described in pixel coordinate. The trajectory of all the markers in images from the left camera during motions is shown in Figure 6, where it can be found that without regard to the shielding effect between the manipulator or operated object and the markers, all marker points fall within the effective observation range of the cameras.

With the solution progress by the forward kinematics, the trajectories of the makers in images have been determined and shown in Figures 5 and 6, and there are 8 trajectories in images for 4 markers in 2 cameras. In these image point trajectories, any number of them is picked as the input, and the proposed algorithm on estimating joint angles in Figure 3 will be used to solve the joint angles of the arm links of the manipulator. It can be found from the results that compared with the preset joint angles in Figure 4, the deviations of the estimated joint angles are very small. When the constraint condition on DOFs is satisfied (at least 2 marker points in images are needed to constrain 4 DOFs, and if only 2 marker points in images are visible, these 2 image points must be that of different markers; besides, there must be at least a marker point on the 4th arm link visible in images.), it does not affect the accuracy of the joint angles that how many and which marker points have been picked to run the proposed algorithm on estimating joint angles.

**4.2. Influence Analysis of Vision Positioning Errors.** The simulation results in the previous subsection shows that after obtaining the accurate positions of the markers on the manipulator in the images from cameras, it is very convenient to estimate the joint angles of the manipulator. While the positions of the markers are actually obtained by image processing, the influence of both imaging factors and the adopted image processing on the errors of the positions of the marker points should be analyzed. Moreover, these errors will also be transmitted to the estimated values of the joint angles of the manipulator. In this paper, it is assumed that the position errors of the marker points are all random

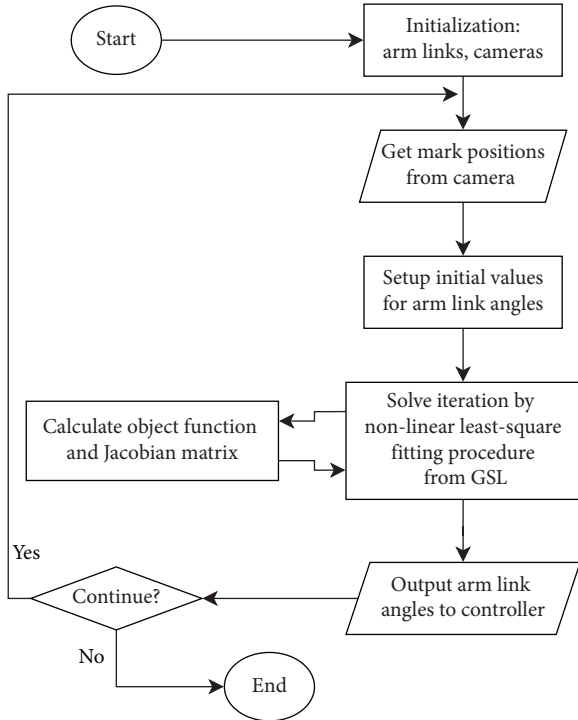


FIGURE 3: Flow chart to solve joint angles of arm links.

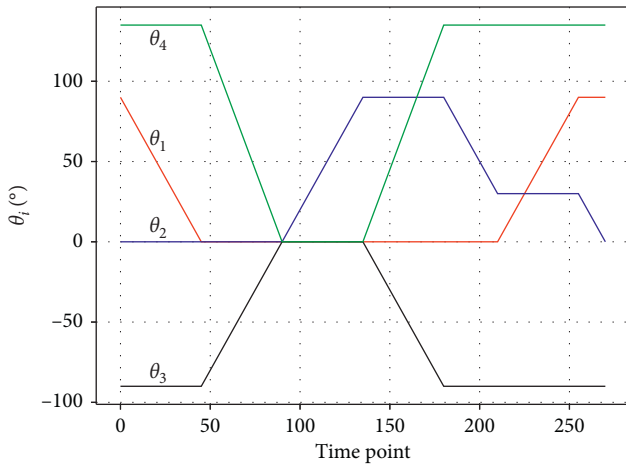


FIGURE 4: Preset joint angles  $\theta_i$  during the manipulator motion.

errors, since the system errors caused by camera installation errors and imaging errors can be eliminated by the whole system calibration. The main sources of these random errors are that each marker point receives different illuminations and other conditions at different times, which bring the difference of imaging effect and lead to the final positioning errors of the markers in images.

In this study, we do not pay attention on the accuracy of vision positioning of the markers on the manipulator, and we assume that the positioning error is no more than  $\pm 5$  mm in the range of 1 meter just likewise in reference [17]. And now, simulated errors are added to the position values of the markers in images, which are calculated in the previous section, and then the influence on estimating joint angles is

analyzed. It is assumed that the errors of the positions of different markers even in the same camera are independent and those of the same marker but in different cameras are also independent. With such assumptions, normal noises are taken as errors and added to the results of the forward kinematics' solution, and the standard deviation of noises is 11 pixels, which is equivalent to a distance of 5.0 mm in a camera with a resolution of  $1920 \times 1080$  for the range 1 meter.

Joint angles estimated with positions of 3 markers in image from a single camera as the input is shown in Figure 7. With the positions of markers 2, 3, and 4 in image from the left camera, errors become larger and larger in joint angles in Figure 7(a), [fig:theta-inv-resolve-n3]a; whilst with the positions of markers 1, 2, and 4 in image from right camera, the errors of joint angles are not so large except at some states, shown in Figure 7(b).

If more markers are visible in more cameras simultaneously, more positions of the markers in images may be taken as the input to estimate joint angles, and their result is shown in Figure 8. When there are 5 markers visible in 2 cameras, the estimated joint angles are very closed to the values preset in simulation, which are shown in Figure 8(a). Furthermore, when the number of positions grows up to 7, the errors of joint angles further decrease. However, when positions of all the 8 markers in images are taken as the input, the errors of the joint angles do not decrease anymore, as shown in Figure 8(b)). The difference between estimated joint angles and the preset ones are calculated, and the standard deviations are calculated, which are shown in Figure 9. It is found that the standard deviation decreases with the increase of the number of the markers in images as the input, especially in the case with less markers in images.

For the final purpose of estimating joint angles of the manipulator is to evaluate the position of the end effector (gripper) of the manipulator, we need to further analyze the influence of the errors of estimated joint angles on the accuracy of the position of the end effector of the manipulator in the forward kinematics. In other words, the accuracy of the position of the end effector or of the relative position between the end effector and the operated objects will directly determine the effect of the automatic control of the underwater manipulator. Therefore, the position of the center of the end effector will be evaluated by the estimated joint angles and compared with that evaluated by the preset joint angles.

The position value of the center of the end effector in the global coordinate system, which is evaluated by the forward kinematics with the joint angles estimated with 5 markers in images as the input, is shown in Figure 10. Because there are errors in the estimated joint angles, the evaluated positions of the end effector contain errors and deviate from the preset ones. During all the trajectories, the maximal position deviation is 55.6 mm, and the mean value is 9.8 mm approximately.

During underwater manipulation, except some small objects, the accuracy meets the requirements of automatic control of underwater manipulator. Even for small objects, it is feasible



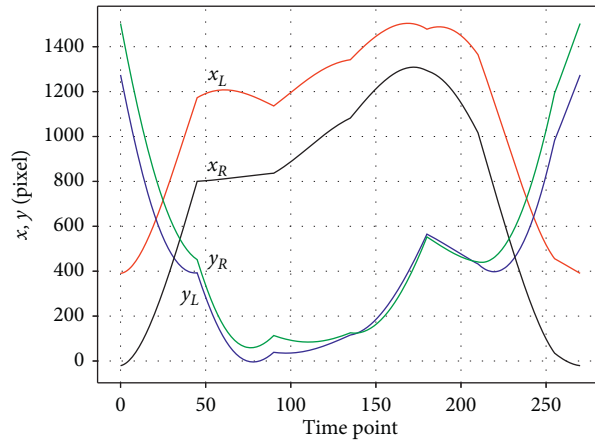


FIGURE 5: Positions of the maker No. 4 in images during the motions by the forward kinematics.

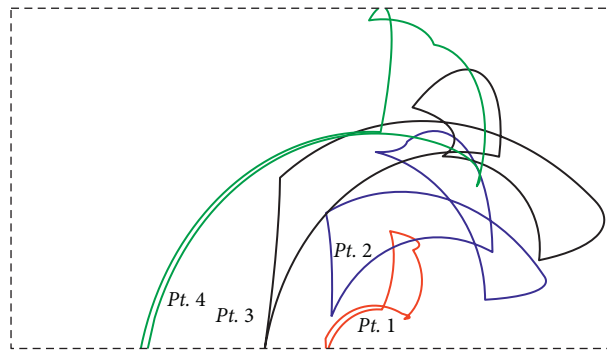


FIGURE 6: Trajectories of the markers in images from left camera.

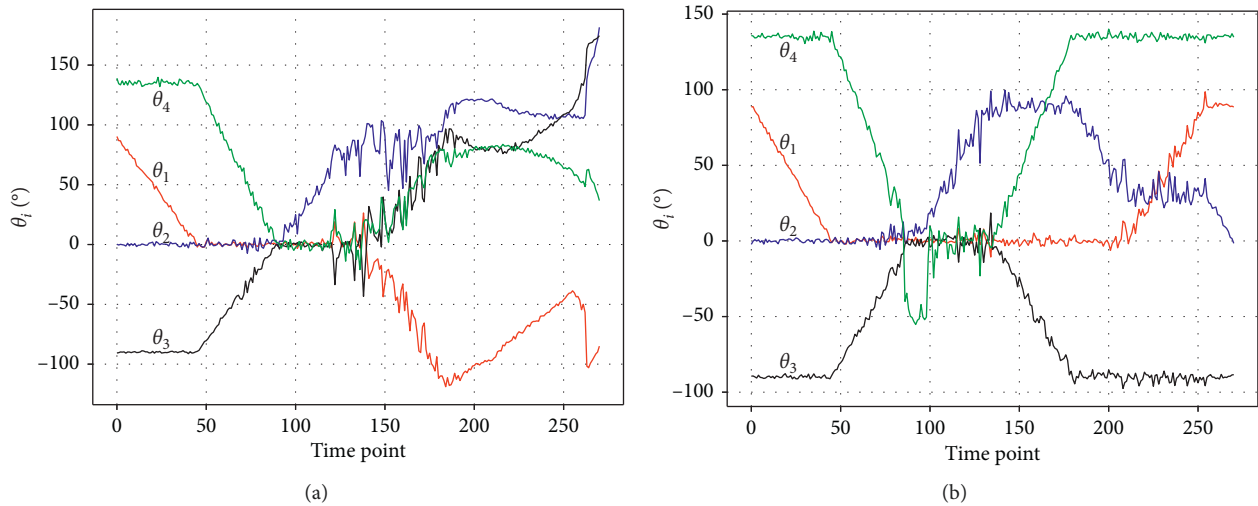


FIGURE 7: Joint angles estimated with 3 markers in image as the input: (a) with image points 2, 3, and 4 in the left camera; (b) with image points 1, 2, and 4 in the right camera.

to observe the operated object by cameras on hand to realize more accurate positioning relatively between the end effector and the operated object, and with estimated joint angles of the

manipulator, the manipulator can be controlled in the joint angles individually more finely, which leads to that the manipulator operates more accurately in underwater manipulation.

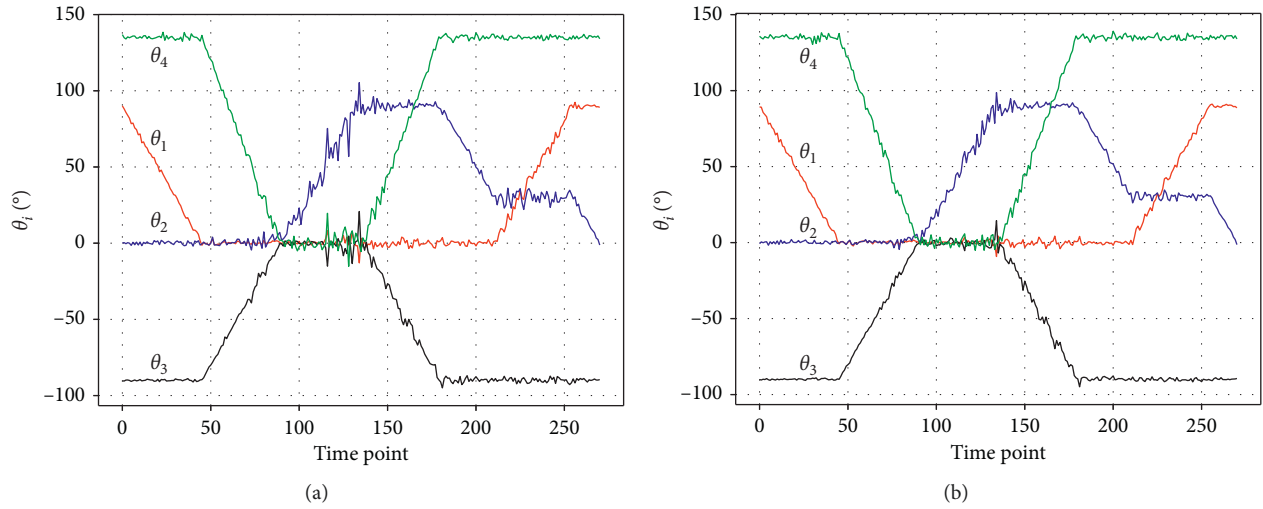


FIGURE 8: Joint angles estimated with more markers in images as the input: (a) with 5 markers in images from both cameras; (b) with 7 markers in images from both cameras.

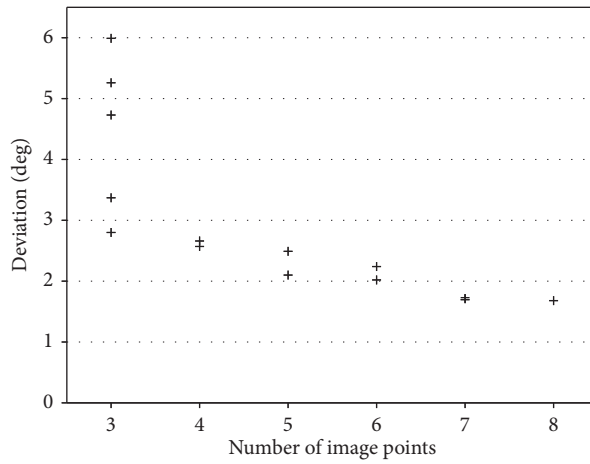


FIGURE 9: Standard deviation of the joint angles with different number of markers as input.

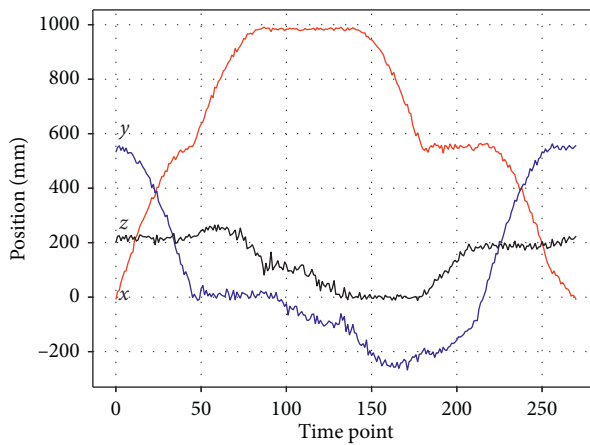


FIGURE 10: Positions of the end effector with estimated errors.

## 5. Conclusions

Due to the large load of underwater manipulation, the underwater manipulator is almost driven by the hydraulic actuator. In addition, these manipulators are not equipped with sensors for the joint angles usually. Therefore, the automatic control method which has been widely used in industrial robots cannot be used for underwater remotely controlled by skilled operators. For the purpose of automatic control of underwater manipulator, this work proposes a schema to sense the joint angles of manipulator, where some markers are arranged on the arm links of manipulator, cameras are arranged to observe the markers and the positions of markers in images are located, and then the joint angles of the arm links are estimated with the positions of the markers in the images as the input. The estimated joint angles may be taken as the input to control the manipulator automatically, which lets the operator operate the manipulator in the Cartesian coordinate system.

In this study, the motion process of the manipulator extending from the shrinking state to extreme extension state and shrinking back to shrinking state is simulated. During the process, the variations of the joint angles with time are preset, and the trajectories of the markers in images from cameras on the arm links are evaluated with the forward kinematics method. According to the typical error of vision positioning, random error noises are added to the trajectories of the markers. Then, the joint angles are estimated by the positions of the markers in images with noises using the method proposed in this paper. Finally, the spatial positions of the end effector are synthesized by the estimated joint angles, and then are compared with the preset ones. With the positioning error in visual image point (RMS: 5 mm), the final error of the end effector is no more than 10 mm (RMS) mostly. This accuracy can almost meet the positioning requirements for manipulators in underwater manipulation. In addition, in aspect of real-time performance, for a typical video with frame rate 30 fps, the delay of each frame is about 33 ms, whilst the delay for image processing and recognition is about 30–40 ms; consequently, the total delay is about 60–70 ms. Such a delay is far greater than that of electromechanical sensors that have an impact on the method and performance of automatic control. Compared with the time required in the process of human observation, cognition, decision-making, and operation, the manipulator automatic control based on visual positioning can still achieve satisfactory performance. Therefore, the method proposed in this paper, which estimates the joint angles of the manipulator by the positions of the markers in image from cameras, establishes the foundation for automatic control of the manipulator, which needs joint angles as the input and feedback.

## Data Availability

The labelled dataset used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

The project was supported by the National Natural Science Foundation of China (grant nos.61572236 and 61300151).

## References

- [1] S. Sivčev, J. Coleman, E. Omerdić et al., "Underwater manipulators: a review," *Ocean Engineering*, vol. 163, pp. 431–450, 2018.
- [2] I. Schjølberg and I. B. Utne, "Towards autonomy in ROV operations," *IFAC-Papers OnLine*, vol. 48, no. 2, pp. 183–188, 2015.
- [3] N. J. Cooke, "Human factors of remotely operated vehicles," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 1, Article ID 166169, 2006.
- [4] R. B. Walters, *Hydraulic and Electric-hydraulic Control Systems*, Springer Science and Business Media, Berlin, Germany, 2013.
- [5] M. W. Dunnigan, D. M. Lane, A. C. Clegg et al., "Hybrid position/force control of a hydraulic underwater manipulator," *IEEE Proceedings—Control Theory and Applications*, vol. 143, no. 2, Article ID 145151, 1996.
- [6] D. Scaradozzi, L. Sorbi, F. Zoppini et al., "Tools and techniques for underwater archaeological sites documentation," in *Proceedings of the 2013 OCEANS (MTS/IEEE)*, pp. 1–6, San Diego, CA, USA, 2013.
- [7] S. Sivčev, J. Coleman, D. Adley, G. Dooly, E. Omerdić, and D. Toal, "Closing the gap between industrial robots and underwater manipulators," in *Proceedings of the OCEANS 2015 (MTS/IEEE)*, Washington, DC, USA, October 2015.
- [8] S. R. Pandian and N. Sakagami, "A neuro-fuzzy controller for underwater robot manipulators," in *Proceedings of the 2010 11th International Conference on Control Automation Robotics Vision*, Article ID 21352140, Singapore, December 2010.
- [9] L. Wang, C. Wang, W. Wang et al., "A novel hybrid control method for the underwater manipulator," in *Proceedings of the 2008 International Workshop on Education Technology and Training*, vol. 1, pp. 790–794, Shanghai, China, December 2008.
- [10] J. J. Fernandez, M. Prats, P. J. Sanz et al., "Grasping for the seabed: developing a new underwater robot arm for shallow-water intervention," *IEEE Robotics & Automation Magazine*, vol. 20, no. 4, pp. 121–130, 2013.
- [11] M. Ishitsuka and K. Ishii, "Modularity development and control of an underwater manipulator for AUV," in *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Article ID 36483653, San Diego, CA, USA, October 2007.
- [12] M. Hildebrandt, J. Kerdels, J. Albiez, and F. Kirchner, "A multilayered controller approach for high precision end-effector control of hydraulic underwater manipulator systems," in *Proceedings of the OCEANS 2009*, pp. 1–5, Bremen, Germany, May 2009.
- [13] D. De Palma and G. Indiveri, "Underwater vehicle guidance control design within the DexROV project: preliminary results," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 265–272, 2016.
- [14] A. Kawamura, M. Kubo, K. Yokoi, N. Sak-Agami, and S. Kawamura, "Motion control of underwater robotic arm using calibration-free visual servoing system," in *Proceedings of the Oceans IEEE*, Washington, DC, USA, October 2015.
- [15] A. Peñalver, J. Pérez, J. J. Fernández et al., "Autonomous intervention on an underwater panel mockup by using

- visually-guided manipulation techniques”” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 5151–5156, 2014.
- [16] P. Trslic, M. Rossi, L. Robinson et al., “Vision based autonomous docking for work class ROVs,” *Ocean Engineering*, vol. 196, Article ID 106840, 2020.
- [17] C. P. Mangipudi and P. Y. Li, “Vision based passive arm localization approach for underwater ROVs using a least squares on  $SO(3)$  gradient algorithm,” in *Proceedings of the 2019 American Control Conference (ACC) Philadelphia*, pp. 5798–5803, Philadelphia, PA, USA, July 2019.
- [18] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Pearson Education, Inc., Cranbury, NY, USA, 3rd edition, 2005.
- [19] The GSL Team, “Nonlinear least-squares fitting,” 2019, <https://www.gnu.org/software/gsl/doc/html/nls.html>.