

## Article

# A New Kinect V2-Based Method for Visual Recognition and Grasping of a Yarn-Bobbin-Handling Robot

Jinghai Han <sup>1</sup>, Bo Liu <sup>2</sup>, Yongle Jia <sup>2</sup>, Shoufeng Jin <sup>2</sup>, Maciej Sulowicz <sup>3</sup> , Adam Glowacz <sup>3</sup> ,  
Grzegorz Królczyk <sup>4</sup>  and Zhixiong Li <sup>4,5,\*</sup> 

- <sup>1</sup> Institute of Rail Transport, Nanjing Vocational Institute of Transport Technology, Nanjing 211188, China; hanjinghai@njitt.edu.cn  
<sup>2</sup> College of Mechanical and Electrical Engineering, Xi'an Polytechnic University, Xi'an 710600, China; 200221059@stu.xpu.edu.cn (B.L.); jiayle@163.com (Y.J.); jdxysf@126.com (S.J.)  
<sup>3</sup> Department of Electrical Engineering, Cracow University of Technology, 31-155 Cracow, Poland; maciej.sulowicz@pk.edu.pl (M.S.); adglow@agh.edu.pl (A.G.)  
<sup>4</sup> Department of Manufacturing Engineering and Automation Products, Opole University of Technology, 45-758 Opole, Poland; g.krolczyk@po.opole.pl  
<sup>5</sup> Yonsei Frontier Lab, Yonsei University, Seoul 03722, Korea  
\* Correspondence: zhixiong.li@yonsei.ac.kr

**Abstract:** This work proposes a Kinect V2-based visual method to solve the human dependence on the yarn bobbin robot in the grabbing operation. In this new method, a Kinect V2 camera is used to produce three-dimensional (3D) yarn-bobbin point cloud data for the robot in a work scenario. After removing the noise point cloud through a proper filtering process, the M-estimator sample consensus (MSAC) algorithm is employed to find the fitting plane of the 3D cloud data; then, the principal component analysis (PCA) is adopted to roughly register the template point cloud and the yarn-bobbin point cloud to define the initial position of the yarn bobbin. Lastly, the iterative closest point (ICP) algorithm is used to achieve precise registration of the 3D cloud data to determine the precise pose of the yarn bobbin. To evaluate the performance of the proposed method, an experimental platform is developed to validate the grabbing operation of the yarn bobbin robot in different scenarios. The analysis results show that the average working time of the robot system is within 10 s, and the grasping success rate is above 80%, which meets the industrial production requirements.

**Keywords:** machine vision; industrial robots; yarn bobbin identification; robot control



**Citation:** Han, J.; Liu, B.; Jia, Y.; Jin, S.; Sulowicz, M.; Glowacz, A.; Królczyk, G.; Li, Z. A New Kinect V2-Based Method for Visual Recognition and Grasping of a Yarn-Bobbin-Handling Robot. *Micromachines* **2022**, *13*, 886. <https://doi.org/10.3390/mi13060886>

Academic Editor: Aiqun Liu

Received: 2 May 2022

Accepted: 29 May 2022

Published: 31 May 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Textile production, as an important resource for fabrics, clothing, and towels, is a typically labor-intensive industry [1]. Winding is a key process in textile production, which generally warps the yarn in the bobbins [2]. Currently, the yarn bobbins are often made by human workers. In order to reduce inefficient labor demands and costs, the textile industry is embracing AI (artificial intelligence) by using unmanned robots to complete the yarn bobbin production [3].

Current research in industrial robots focuses on robot grabbing with machine vision and graphics processing. Xue et al. [4] proposed a vision-based joint inspection method to improve the tracking accuracy of the weld robots. Ramon-Soria et al. [5] mounted a robotic arm with a camera on top of an unmanned aerial vehicle to achieve the grasping of aerial targets. D'Avella et al. [6] addressed the problem of grasping in a cluttered environment by using vision techniques to extract the edge contours of the target to calculate the grasping point and solve the perception problem caused by the chaotic environment. Jiang et al. [7] used information fusion of color images and depth images to improve the detection success rate of a vision robot. Du et al. [8] proposed a binocular vision-based object recognition and robotic grasping strategy. Yang et al. [9] built a vision recognition system (VRS) based on the

YOLOv3 model to improve the efficiency of picking robots. Matsuo et al. [10] investigated the robots in logistics handling in warehouses and convenience stores, where multi-object grasping was achieved through the dual-arm operation. Xiao et al. [11] designed an automatic sorting robot that used depth information and infrared images to locate objects. Lin et al. [12] proposed a method to solve the occlusion problem by using multiple cameras to capture images and successfully grasp objects in the occluded area. Gao et al. [13] developed a vision-based seal assembly system to improve the efficiency of production lines. Sun et al. [14] proposed an end-to-end deep neural network-based grasping system that successfully grasped targets in the presence of object overlap. Song et al. [15] proposed a tactile-visual fusion-based robot grasping detection method to improve the applicability of manipulators in force-sensitive tasks. Yu et al. [16] proposed a new vision-based grasping method for target objects in occlusion situations. Bergamini et al. [17] proposed a method based on deep learning techniques for detecting the grasp of unknown objects by a robot in an unstructured environment. Hu et al. [18] established a projection mapping relationship between plane and space and proposed a 6D pipeline pose estimation method based on machine vision. Han et al. [19] applied binocular vision to identify and locate targets and improved the accuracy of robotic arm grasping. Lu et al. [20] applied machine vision to a sorting robot and developed a method that combined the YOLOv3 algorithm with manual features to improve sorting efficiency. Lou et al. [21] proposed a Gaussian hybrid robotic arm recognition and grasping method based on machine vision models to improve the grasping accuracy of the robot. According to the existing literature, most robot grabbing technologies address the multidimensional spatial position recognition of a single object and focus on the pose recognition of multiple objects with similar shape parameters; however, very limited work has addressed the challenging task of grabbing multiple objects with overlapping poses. As a result, most existing methods cannot be directly used in yarn bobbin robots; and to our best knowledge, little research has been done on grabbing multiple yarn bobbins with overlapping poses.

To bridge the aforementioned research gap, this work develops a new machine vision method to realize effective robot grabbing of multiple yarn bobbins. This new method is able to effectively recognize the spatial positions and poses of the yarn bobbins in cylindrical/conical shapes. This is the first time that a visual system has been developed for the robots in the practical textile application as a solution for yarn-bobbin grabbing. The visual system can obtain the target imaging and point cloud in specific scenarios with the Kinect V2 camera, derive precise 3D information of the yarn bobbins through target recognition and point cloud processing, and enable the industrial robots to correctly grab the yarn bobbins.

This paper is structured as follows. In Section 2, the proposed machine vision method is introduced in detail. In Section 3, the experimental testing method is described and evaluated. Section 4 presents the main conclusions of this study.

## 2. Method and Materials

### 2.1. Establishment of Experimental Platform

In the textile winding process, the yarn bobbins produced by the winder machines will be collected and sorted into the yarn hopper, as shown in Figure 1, where one human laborer (worker 1) is employed to put the yarn bobbins from the hopper to the conveyor belt, and another human laborer (worker 2) is employed to hang the yarn bobbins onto the rack. As a result, at least two workers are required in the winding process to transfer the yarn bobbins from the hopper to the rack. Several repetitive operations of the workers will significantly increase labor intensity and safety risks. With the rapid development of AI (artificial intelligence) robotics, the labor-to-robot transition is a must in the textile winding process demanded by the textile industry; and at the first step in the roboticization reformation, intelligent robots are expected to handle the yarn bobbin to the rack to minimize manpower, increase efficiency and ensure staff safety. In this study, we aim to develop an industrial robot that can directly grab the yarn bobbins from the hopper and place them onto the

rack. Compared with manual handling, this new robot-based winding system simplifies the operation complexity by removing the conveyor belt part. Figure 2 manifests the new winding system.

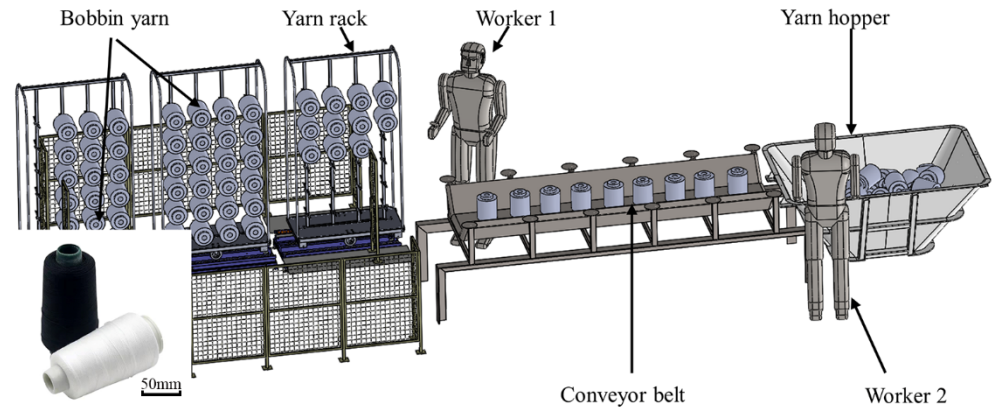


Figure 1. Manual yarn feeding process.

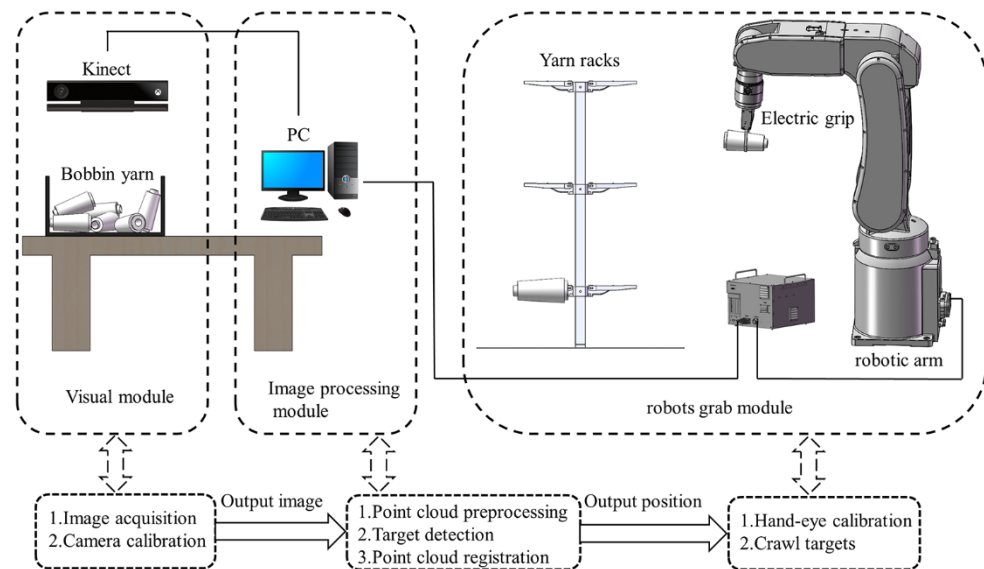
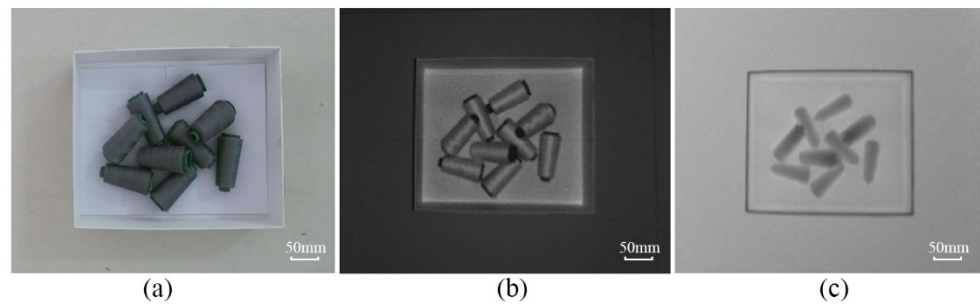


Figure 2. Developed a robot-based winding system.

In Figure 2, the hardware of the new robot-based winding system consists of one Kinect V2 camera, one HIWIN RA605 robotic arm, one HIWIN XEG-32 electric gripper, and one computer. The computer connects with the Kinect V2 camera through a USB 3.0 interface and the robot arm control cabinet through Ethernet; the robot arm control cabinet controls the opening or closing of the electric gripper through I/O signals to realize the yarn-bobbin-grabbing function. The MATLAB programming language is adopted to develop each module in the system to enable the robots with visual recognition ability to effectively complete the grasping operation. According to the functional requirements of each module, the software system is divided into three parts: the vision module, the image processing module, and the robot grab module.

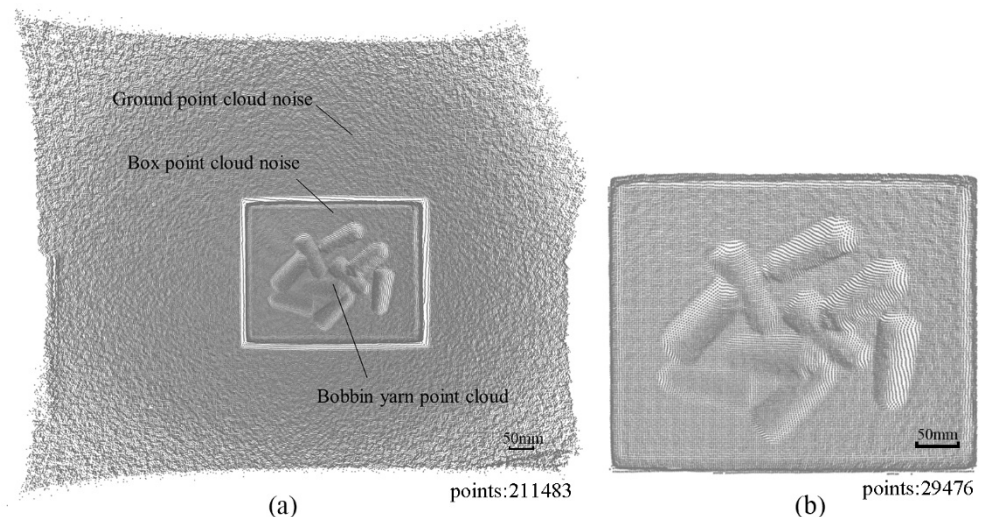
### 2.2. Vision Image Processing

The Kinect V2 camera is used in the vision system to collect the images of the yarn bobbins. This device is based on the Kinect for Windows SDK, a development tool provided by Microsoft, to acquire the images of the yarn bobbins, as shown in Figure 3.



**Figure 3.** Kinect V2 acquired images: (a) RGB image; (b) IR image; (c) Depth image.

In Figure 3c, the depth image is often expressed in grayscale or pseudo-color, which is essentially different from the grayscale and RGB images of traditional cameras. The value corresponding to each pixel in the ordinary color image is the component of the pixel in the three RGB channels, and the value corresponding to each pixel in the depth image represents the spatial distance from the object surface point to the sensor plane. Therefore, the depth image can be converted into 3D point cloud data through coordinate transformation according to the characteristics of the depth image. The transformed point cloud data is shown in Figure 4a.



**Figure 4.** Mapping results: (a) point cloud of yarn bobbin; (b) denoised point cloud.

In Figure 4a, the obtained yarn bobbin point cloud data contains a lot of noise, and the data storage space is huge. Therefore, it is requested to filter the point cloud [22]. Pass-through filtering is employed as the filtering algorithm. By setting the threshold parameter, the points within the specified parameter range can be retained, while the outside points are filtered out. The detailed filtering process is described in the following steps:

- (1) Import the point cloud data into the pass-through filter.
- (2) Set different thresholds to determine the filtering directions.
- (3) Filter the discrete points and save the denoised point cloud data.

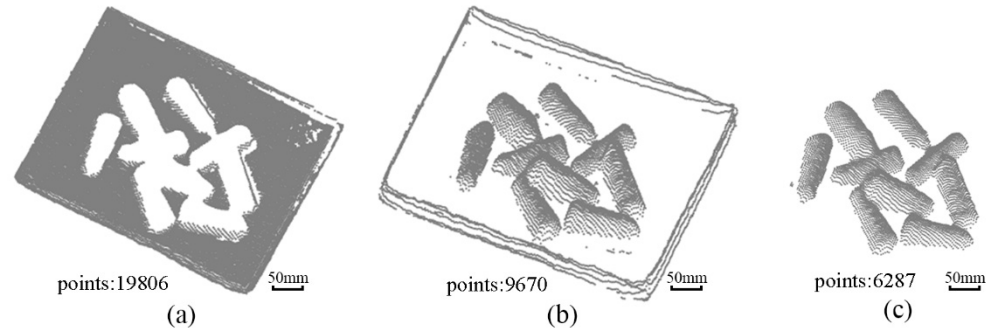
The value ranges of the filter specified in this paper in  $X$ ,  $Y$  and  $Z$  dimensions are set as follows:

$$\begin{cases} -0.3 \leq X < 0.15 \\ -0.15 \leq Y < 0.23 \\ -1 \leq Z < 1 \end{cases} \quad (1)$$

According to the dimensional values set in Equation (1), the original point cloud data shown in Figure 4a is straightly filtered to remove the redundant environmental point cloud, and the result is shown in Figure 4b.



Then, the M-estimated sampling consensus algorithm (MSAC) [23] is adopted to fit the denoised point cloud to a bottom surface box, as shown in Figure 5a. The box side point cloud data is then removed by the direct-pass filtering to separate yarn bobbins from the box, as shown in Figure 5b,c.



**Figure 5.** Extract yarn bobbin images via MSAC: (a) fitting the box bottom point cloud; (b) removing the box bottom point cloud; (c) removing the box side point cloud.

### 2.3. Identification of Yarn Bobbin Targets

In order to identify each yarn bobbin in Figure 5c, the first step is to extract the key points. The key points are those with representative features such as stability and distinctiveness in the image. In this study, the corner points, gradients and grayscale sharply changing points in the point cloud are used as key points. The Internal Shape Descriptor (ISS) [24] is employed to use rich geometric information to extract the key points. Let the point  $P_i(x_i, y_i, z_i)$  be a point in the point cloud of one yarn bobbin; the process of extracting the key points is described as follows.

- (1) Establish a local coordinate system for point  $P_i$  in the point cloud of the yarn bobbin and set the search radius  $r$  for the point.
- (2) Set point  $P_i$  as the center and  $r$  as the neighbourhood radius, traverse all points within the neighbourhood of  $P_i$ , and calculate their weights  $w_{ij}$ .

$$w_{ij} = 1 / \|P_i - P_j\|, \|P_i - P_j\| < r \tag{2}$$

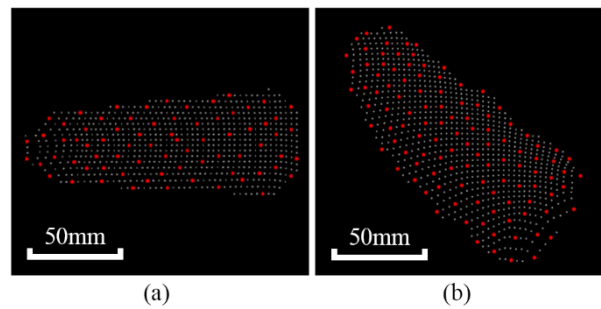
- (3) Calculate the covariance matrix  $cov(P_i)$  for each point  $P_i$ .

$$cov(P_i) = \sum_{|P_i - P_j| < r} w_{ij} (P_i - P_j) (P_i - P_j)^T / \sum_{|P_i - P_j| < r} w_{ij} \tag{3}$$

- (4) Calculate the eigenvalues  $\{\lambda_i^1, \lambda_i^2, \lambda_i^3\}$  of the covariance matrix of each point  $P_i$  and rank them in descending order.
- (5) Set the threshold ( $\epsilon_1 \leq 1$  and  $\epsilon_2 \leq 1$ ) and use the points satisfying Equation (4) as the key points of the yarn target.

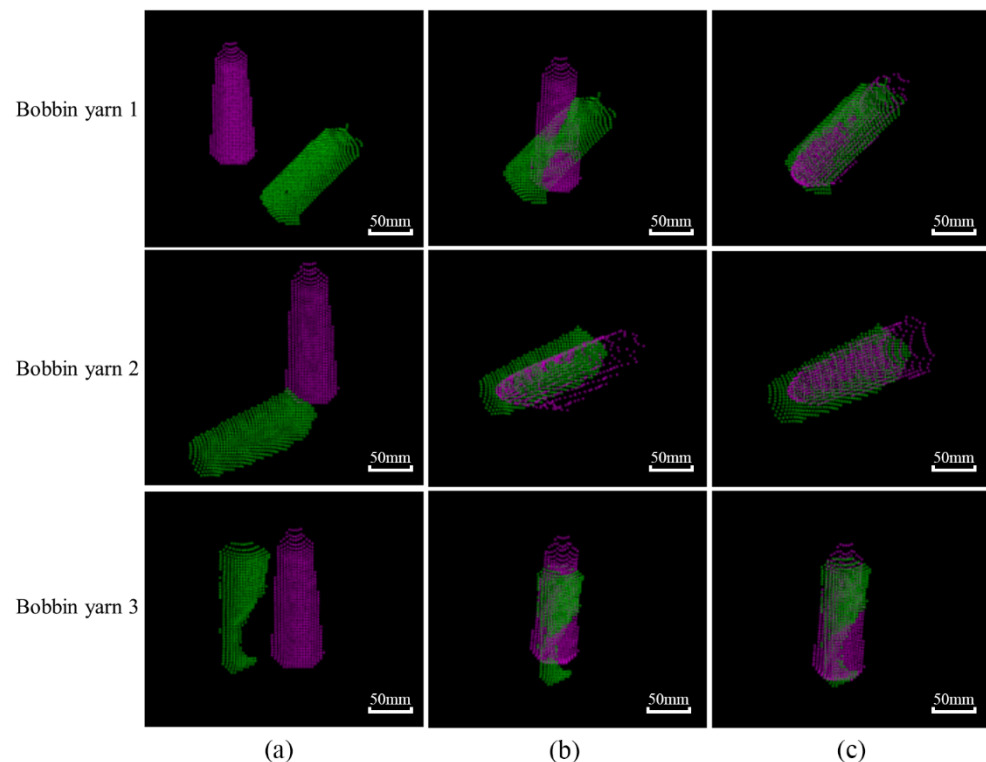
$$\lambda_i^2 / \lambda_i^1 \leq \epsilon_1, \lambda_i^3 / \lambda_i^2 \leq \epsilon_2 \tag{4}$$

- (6) Repeat steps (1) to (5) until all the points in the yarn bobbin point cloud are traversed.
  - (7) Downsample the final extracted target key points.
- Figure 6 provides an example of the key point extraction.



**Figure 6.** Key point extraction: (a) extraction result of the point cloud of the yarn bobbin template; (b) extraction result of the point cloud of the yarn bobbin to be captured.

After generating the key points, the second step is to perform the point cloud alignment between the template and target point cloud. Point cloud registration is to use the yarn bobbin template point cloud to construct point-to-point features to find the yarn bobbin target in the actual scene and determine its 3D pose. The rigid transformation between the template point cloud and the target point cloud is the estimation process of target poses. The registration effects are shown in Figure 7.



**Figure 7.** Point cloud alignment results: (a) initial position; (b) coarse alignment; (c) fine alignment.

The rough registration based on the Principal Component Analysis (PCA) [25] is introduced in the point cloud alignment. The principal axis direction of the point cloud data is calculated for the rough registration, where the rotation matrix and the translation vector in the direction of the principal axis are derived by calculating the offset of the center coordinates of two sets of point clouds. Through the rough registration of point clouds, the target point cloud aligns the template point cloud with relatively low accuracy; it is necessary to minimize the error between the two-point clouds through continuous iteration, i.e., the precise registration. The ICP algorithm, also known as the iterative nearest-point algorithm, is widely used in the fine alignment phase of 3D point clouds because of its fast and simple nature [26]. This paper uses the point-to-point ICP algorithm for fine alignment of point clouds.

The core equation for the point-to-point ICP alignment is given in Equation (5), and the positional matrix for the yarn bobbin 1 in Figure 7 is described in Equation (6).

$$\Delta T = \arg \min_{R,t} \sum_{i=1}^n \|p_{bi} - (Rp_{ai'} + t)\|_2^2 \tag{5}$$

$$\begin{bmatrix} 0.76802416 & -0.62644737 & 0.13305101 & -0.05595922 \\ 0.61897762 & 0.77942243 & 0.09678518 & 0.05432455 \\ -0.16433376 & 0.00802224 & 0.98637217 & 0.01234947 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \tag{6}$$

### 3. Numerical and Experimental Analyses

#### 3.1. Calibration Experiments

The experimental platform is shown in Figure 2. The Kinect V2 camera was mounted on the upper part of the load table through the gantry and camera head device and could be adjusted up and down within a certain range to ensure the quality of the acquired data; the HIWIN six-degrees-of-freedom robot arm was fixed on the load table, and the XEG-32 two-finger electric gripper was mounted at the end and acted as the actuator for the robot grasping. This "eye-to-hand" approach [27] is adopted for calibrating the vision system according to the overall design plan, including the camera calibration and hand-eye calibration, where the specification of the calibration board is an 11 × 8 square checkerboard grid, the size of each square grid is 30 × 30 mm, and the accuracy is ±0.05 mm.

The camera calibration is based on the calibration toolbox of MATLAB 2020a to calibrate the depth camera of the Kinect V2. The corner points of the checkerboard grid can be detected according to the acquired calibration board image and the parameters of the calibration board, and the calibration is completed by returning the pixel point coordinates of the detected corner points. The final Kinect V2 depth camera internal reference is shown in Table 1.

**Table 1.** Kinect V2 depth camera calibration internal reference.

Name of Parameter	Data
Focal length	[359.2873 359.4936]
Point coordinates	[252.6447 204.3739]
Radial distortion	[0.0880 -0.2112]
Error	0.2374

The purpose of hand-eye calibration is to solve the pose transformation matrix from the camera coordinate system to the manipulator coordinate system so as to convert the target yarn-bobbin poses from the camera coordinate system to the manipulator coordinate system. As shown in Figure 8, {B} is the base coordinate system of the manipulator; {E} is the coordinate system of the end of the manipulator; {K} is the calibration board coordinate system; {C} is the Kinect V2 depth camera coordinate system. **A** represents the pose of the end of the manipulator in the base coordinate system of the manipulator, which can be obtained by the positive solution of the robot kinematics; **B** represents the pose of the calibration board in the coordinate system of the end of the manipulator; **C** represents the pose of the calibration board in the camera coordinate system, that is, the external parameters in the camera calibration; **D** is the pose of the camera in the base coordinate system of the manipulator, that is, the final result of the hand-eye calibration.

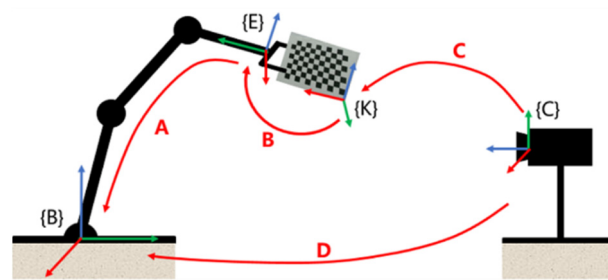


Figure 8. Schematic diagram of hand-eye calibration.

Through the position conversion relationship, the process of the hand-eye calibration can be transformed into the solution  $X$  to  $AX = XB$ , that is, the conversion matrix from the Kinect V2 depth camera coordinate system {C} to the manipulator base coordinate system {B}, and the solution  $X$  is derived as follows, where the calibration accuracy is close to 99%.

$$X = \begin{bmatrix} 0.0022 & 0.9999 & 0.0144 & 57.7571 \\ -0.9999 & 0.0020 & 0.0109 & -24.0240 \\ 0.0109 & -0.0144 & 0.9998 & -147.0713 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix} \quad (7)$$

### 3.2. Grasping Experiments

To evaluate the performance of the proposed robot visual recognition and grabbing method, field tests using two different yarn bobbin products with different specifications were conducted, as shown in Table 2. Two sets of scenarios (i.e., separated and disordered stacking) were designed for comparison and validation, and in each scenario, nine yarn bobbins were placed, as shown in Figure 9, with serial numbers 1 to 9.

Table 2. Yarn bobbin specifications.

Types	Diameter/mm		Height (mm)	Weight (kg)
Tower-shaped	Small 41	Big 52	112	0.11
Cylindrical	47		125	0.12

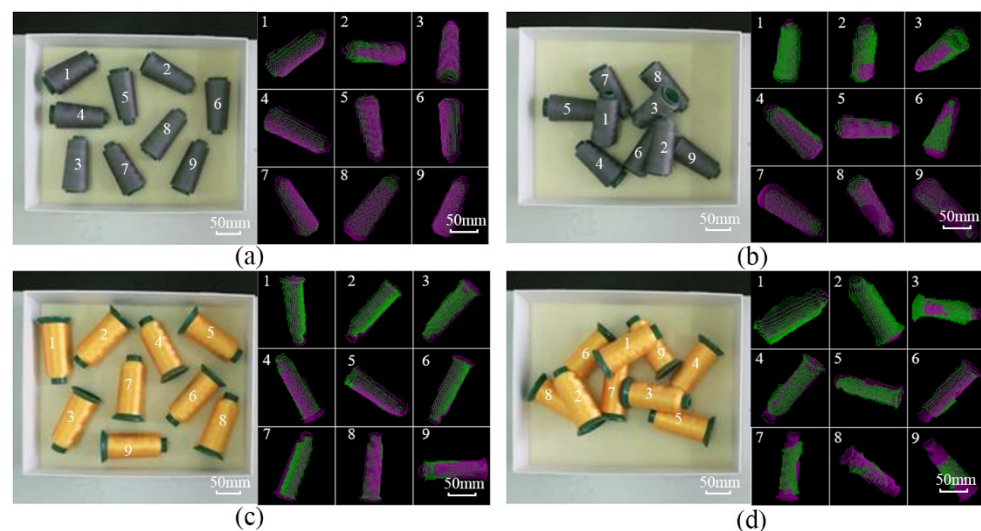
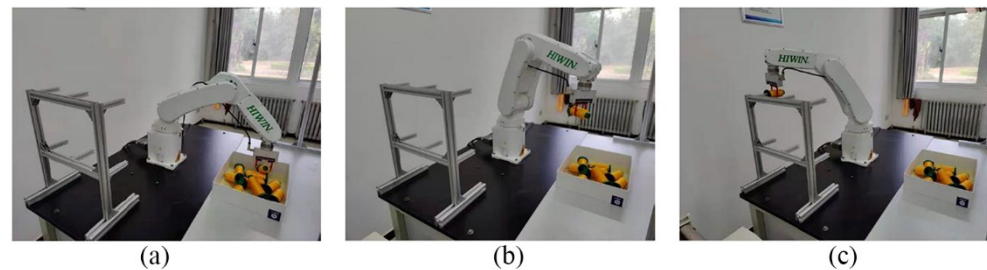


Figure 9. Experimental tests in different scenarios: (a) Tower yarn bobbin without contact scenario; (b) Tower yarn-bobbin stacking scenario; (c) Cylindrical yarn bobbin without contact scenario; (d) Cylindrical yarn-bobbin stacking scenario.

In the two scenarios, each kind of yarn bobbin was tested for 150 grabs, and the box containing the yarn bobbins was placed in the working range of the robot, where the Kinect



V2 camera took the yarn bobbin images and derived the poses of the yarn bobbins through the imaging process to determine the grabbing sequence according to the quantity of measured yarn-bobbin point clouds. Then, the yarn bobbin poses in the camera coordinate system were transferred to the hand-eye calibration; the robot-controlled gripper moved above the yarn bobbin and adjusted the end poses, grabbing the yarn bobbin, then moved the bobbins onto the rack. The grabbing process is shown in Figure 10.



**Figure 10.** Diagram of the gripping process of part of the yarn bobbin. (a) grasping operation, (b) transmitting operation, and (c) loading operation.

The average time from the acquisition of target information by the vision system to the completion of grasping and grasping success rates are provided in Table 3.

**Table 3.** Experimental results.

Experimental Scenarios	Experimental Subjects	Average Time Taken	Number of Experiments	Number of Successes	Success Rate
No contact	Tower-shaped	8.35 s	150	138	92.0%
	Cylindrical	8.07 s	150	139	92.7%
Unordered stacking	Tower-shaped	9.68 s	150	127	84.7%
	Cylindrical	9.82 s	150	129	86.0%

The experimental results show that the proposed vision recognition and grasping system for the yarn bobbin robot can successfully grasp the yarn targets and complete the yarn loading work within the industrial time demand; the grasping success rate in the no-contact scenario is much higher compared with the disordered stacking scenario, and the operation time was shorter in the no-contact scenario because there was interference between the yarn targets in the disordered stacking scenario. In addition, due to incomplete information, the robot failed to grab some bobbins, which led to a relatively low grasping success rate. However, one can note that the bobbin-loading robot is comparable to two workers in the traditional production line, as shown in Figure 1, and has a clear advantage over manual handling when working continuously for a long period of time. As a result, the work efficiency significantly improved, thus reducing the production costs and safety risks.

#### 4. Conclusions

In this research, a new machine vision-based system is proposed for the yarn bobbin robots to complete the yarn-bobbin grabbing operation. An experimental platform was established to evaluate the proposed machine vision-based system. The main conclusions are as follows:

(1) By analyzing the yarn-bobbin loading processing, we develop a Kinect V2-based robot visual recognition and grabbing method, which erects a reality experimental platform for verification. Experimental results show that the robot is able to complete the grabbing operation within 10 seconds with a grabbing success rate of over 80%, which sufficiently meets the industrial requirements.

(2) By analyzing the yarn-bobbin point cloud data acquisition and pre-processing, the yarn-bobbin depth images acquired from the Kinect V2 camera can be converted into 3D

point cloud data, and the noise points in the images are removed by pass-through filtering and MSAC fitting to yield the point cloud data of the yarn bobbins.

(3) By performing the rough registration and precise registration, the exact poses of the identified yarn bobbins can be extracted to help the robot correctly grab the yarn bobbins.

**Author Contributions:** Conceptualization, S.J. and Z.L.; methodology, J.H.; software, B.L.; validation, Y.J., S.J. and G.K.; formal analysis, S.J., B.L. and M.S.; investigation, A.G.; resources, G.K.; data curation, B.L. and Z.L.; writing—original draft preparation, J.H., Y.J. and S.J.; writing—review and editing, M.S. and G.K.; visualization, G.K.; supervision, A.G.; project administration, G.K.; funding acquisition, Z.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** The research leading to these results has received funding from the Norway Grants 2014–2021 operated by National Science Centre under Project Contract No 2020/37/K/ST8/02748.

**Acknowledgments:** The authors gratefully acknowledge the support from Longyan Tobacco company.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Zhang, J.; He, L.; Cheng, L. Is China's Textile Industry Still a Labour-Intensive Industry? *Fibres Text. East. Eur.* **2021**, *29*, 13–16. [[CrossRef](#)]
- Babu, B.S.; Kumar, P.S.; Kumar, M.S. Effect of yarn type on moisture transfer characteristics of double-face knitted fabrics for active sportswear. *J. Ind. Text.* **2018**, *49*, 1078–1099. [[CrossRef](#)]
- Noor, A.; Saeed, M.A.; Ullah, T.; Uddin, Z.; Khan, R.M.W.U. A review of artificial intelligence applications in apparel industry. *J. Text. Inst.* **2021**, *113*, 505–514. [[CrossRef](#)]
- Xue, B.; Chang, B.; Peng, G.; Gao, Y.; Tian, Z.; Du, D.; Wang, G. A Vision Based Detection Method for Narrow Butt Joints and a Robotic Seam Tracking System. *Sensors* **2019**, *19*, 1144. [[CrossRef](#)] [[PubMed](#)]
- Ramon-Soria, P.; Arrue, B.C.; Ollero, A. Grasp planning and visual servoing for an outdoors aerial dual manipulator. *Engineering* **2020**, *6*, 77–88. [[CrossRef](#)]
- D'Avella, S.; Tripicchio, P.; Avizzano, C.A. A study on picking objects in cluttered environments: Exploiting depth features for a custom low-cost universal jamming gripper. *Robot. Comput. Manuf.* **2019**, *63*, 101888. [[CrossRef](#)]
- Jiang, D.; Li, G.; Sun, Y.; Hu, J.; Yun, J.; Liu, Y. Manipulator grabbing position detection with information fusion of color image and depth image using deep learning. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 10809–10822. [[CrossRef](#)]
- Du, Y.C.; Taryudi, T.; Tsai, C.T.; Wang, M.S. Eye-to-hand robotic tracking and grabbing based on binocular vision. *Microsyst. Technol.* **2021**, *27*, 1699–1710. [[CrossRef](#)]
- Yang, H.; Chen, L.; Ma, Z.; Chen, M.; Zhong, Y.; Deng, F.; Li, M. Computer vision-based high-quality tea automatic plucking robot using Delta parallel manipulator. *Comput. Electron. Agric.* **2021**, *181*, 105946. [[CrossRef](#)]
- Matsuo, I.; Shimizu, T.; Nakai, Y.; Kakimoto, M.; Sawasaki, Y.; Mori, Y.; Sugano, T.; Ikemoto, S.; Miyamoto, T. Q-bot: Heavy object carriage robot for in-house logistics based on universal vacuum gripper. *Adv. Robot.* **2020**, *34*, 173–188. [[CrossRef](#)]
- Xiao, W.; Yang, J.; Fang, H.; Zhuang, J.; Ku, Y.; Zhang, X. Development of an automatic sorting robot for construction and demolition waste. *Clean Technol. Environ. Policy* **2020**, *22*, 1829–1841. [[CrossRef](#)]
- Lin, S.; Wang, N. Cloud robotic grasping of Gaussian mixture model based on point cloud projection under occlusion. *Assem. Autom.* **2021**, *41*, 312–323. [[CrossRef](#)]
- Gao, M.; Li, X.; He, Z.; Yang, Y. An Automatic Assembling System for Sealing Rings Based on Machine Vision. *J. Sens.* **2017**, *2017*, 4207432. [[CrossRef](#)]
- Sun, H.; Cui, X.; Song, Z.; Gu, F. Precise grabbing of overlapping objects system based on end-to-end deep neural network. *Comput. Commun.* **2021**, *176*, 138–145. [[CrossRef](#)]
- Song, Y.; Luo, Y.; Yu, C. Tactile–Visual Fusion Based Robotic Grasp Detection Method with a Reproducible Sensor. *Int. J. Comput. Intell. Syst.* **2021**, *14*, 1753–1762. [[CrossRef](#)]
- Yu, Y.; Cao, Z.; Liang, S.; Geng, W.; Yu, J. A Novel Vision-Based Grasping Method Under Occlusion for Manipulating Robotic System. *IEEE Sens. J.* **2020**, *20*, 10996–11006. [[CrossRef](#)]
- Bergamini, L.; Sposato, M.; Pellicciari, M.; Peruzzini, M.; Calderara, S.; Schmidt, J. Deep learning-based method for vision-guided robotic grasping of unknown objects. *Adv. Eng. Inform.* **2020**, *44*, 101052. [[CrossRef](#)]
- Hu, J.; Liu, S.; Liu, J.; Wang, Z.; Huang, H. Pipe pose estimation based on machine vision. *Measurement* **2021**, *182*, 109585. [[CrossRef](#)]
- Han, Y.; Zhao, K.; Chu, Z.; Zhou, Y. Grasping Control Method of Manipulator Based on Binocular Vision Combining Target Detection and Trajectory Planning. *IEEE Access* **2019**, *7*, 167973–167981. [[CrossRef](#)]
- Lu, Z.; Zhao, M.; Luo, J.; Wang, G.; Wang, D. Design of a winter-jujube grading robot based on machine vision. *Comput. Electron. Agric.* **2021**, *186*, 106170. [[CrossRef](#)]

21. Lou, J. Crawling robot manipulator tracking based on gaussian mixture model of machine vision. *Neural Comput. Appl.* **2021**, *34*, 6683–6693. [[CrossRef](#)]
22. Han, X.-F.; Jin, J.S.; Wang, M.-J.; Jiang, W.; Gao, L.; Xiao, L. A review of algorithms for filtering the 3D point cloud. *Signal Process. Image Commun.* **2017**, *57*, 103–112. [[CrossRef](#)]
23. Ebrahimi, A.; Czarnuch, S. Automatic Super-Surface Removal in Complex 3D Indoor Environments Using Iterative Region-Based RANSAC. *Sensors* **2021**, *21*, 3724. [[CrossRef](#)]
24. Zhong, Y. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, Kyoto, Japan, 27 September–4 October 2009; pp. 689–696.
25. Tian, H.; Dang, X.; Wang, J.; Wu, D. Registration method for three-dimensional point cloud in rough and fine registrations based on principal component analysis and iterative closest point algorithm. *Traitement du Signal* **2017**, *34*, 57–75. [[CrossRef](#)]
26. Li, P.; Wang, R.; Wang, Y.; Tao, W. Evaluation of the ICP Algorithm in 3D Point Cloud Registration. *IEEE Access* **2020**, *8*, 68030–68048. [[CrossRef](#)]
27. Cui, H.; Sun, R.; Fang, Z.; Lou, H.; Tian, W.; Liao, W. A novel flexible two-step method for eye-to-hand calibration for robot assembly system. *Meas. Control* **2020**, *53*, 2020–2029. [[CrossRef](#)]