TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 08/12 pp255–269
DOI: 10.26599/TST.2019.9010009
Volume 25, Number 2, April 2020

Machine Learning-Based Multi-Modal Information Perception for Soft Robotic Hands

Haiming Huang*, Junhao Lin, Linyuan Wu, Bin Fang, Zhenkun Wen, and Fuchun Sun*

Abstract: This paper focuses on multi-modal Information Perception (IP) for Soft Robotic Hands (SRHs) using Machine Learning (ML) algorithms. A flexible Optical Fiber-based Curvature Sensor (OFCS) is fabricated, consisting of a Light-Emitting Diode (LED), photosensitive detector, and optical fiber. Bending the roughened optical fiber generates lower light intensity, which reflecting the curvature of the soft finger. Together with the curvature and pressure information, multi-modal IP is performed to improve the recognition accuracy. Recognitions of gesture, object shape, size, and weight are implemented with multiple ML approaches, including the Supervised Learning Algorithms (SLAs) of K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (LR), and the unSupervised Learning Algorithm (un-SLA) of K-Means Clustering (KMC). Moreover, Optical Sensor Information (OSI), Pressure Sensor Information (PSI), and Double-Sensor Information (DSI) are adopted to compare the recognition accuracies. The experiment results demonstrate that the proposed sensors and recognition approaches are feasible and effective. The recognition accuracies obtained using the above ML algorithms and three modes of sensor information are higer than 85 percent for almost all combinations. Moreover, DSI is more accurate when compared to single modal sensor information and the KNN algorithm with a DSI outperforms the other combinations in recognition accuracy.

Key words: multi-modal sensors; optical fiber; gesture recognition; object recognition; Soft Robotic Hands (SRHs); Machine Learning (ML)

1 Introduction

Soft Robotic Hands (SRHs) are presently of interest to many scientists, as they have a number of advantages over traditional hard hands[1,2]. Deimel et al.[3,4] designed a compliant, underactuated pneumatic robotic hand for dexterous manipulation and robust grasping. Gupta et al.[5] trained an SRH to perform dexterous manipulation tasks learning from human demonstrations. Hao et al.[6] designed a programmable mechanical freedom and variable stiffness soft actuator using low melting point alloy.

Although various SRHs have been developed for safe, adaptable grasping, most of them still lack sensory feedback. Flexible tactile sensors enable an SRH to interact in a more friendly way with humans and the environment, and to perform more dexterous manipulations[7]. Most of these sensors are based on capacitive[8], resistive, and piezoresistive[9], or optical principles[10]. The capacitance of capacitive sensors varies with the change of the gap between

• Haiming Huang, Junhao Lin, Linyuan Wu, and Fuchun Sun are with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China. E-mail: haimhuang@163.com; fcsun@tsinghua.edu.cn.
• Fuchun Sun and Bin Fang are with Department of Computer Science and Technology, Tsinghua University, Tsinghua University, Beijing 100084, China.
• Zhenkun Wen is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China.
∗ To whom correspondence should be addressed.
Manuscript received: 2018-12-03; revised: 2019-02-14; accepted: 2019-03-11

the conductive plates on the imposition of external force[11]. Resistive sensors produce a change in electrical resistance when mechanical deformation occurs[9]. Optical fiber sensors are immune to magnetic fields and are inherently safe for interacting with human beings[12]. Zhao et al.[13] performed curvature control of a soft orthotic using low cost solid-state optical fiber, and then completed a five-finger soft orthosis integrating these optical strain sensors and EMG control[14]. Most recently, they developed an optoelectronically innervated soft prosthetic hand via stretchable optical waveguides. The optical waveguide was fabricated with two soft silicone composites with different refractive index. The soft prosthetic featured active haptic sensing to detect shape, texture, and softness, and was able to select the ripest among a group of tomatoes[15].

However, soft prosthetics still fell short in two ways: the use of a single modal sensor and the lack of Machine Learning (ML) in Information Perception (IP). To address the first of these shortcomings, Kampmann and Kirchner[16] integrated fiber-optic sensor arrays into a multi-modal tactile sensor processing system. To address the other, ML algorithms have been adopted to improve the accuracy of recognition[17–19]. Naya et al.[20] adopted a K-Nearest Neighbor (KNN) algorithm to support a haptic interface for a pet-like robot, while Hu et al.[21] utilized Principal Component Analysis (PCA) to compress the attribute data and extract feature information, and used the Support Vector Machine (SVM) algorithm to classify low-dimensional features. These Supervised Learning Algorithms (SLAs) require the trained dataset to be labeled, whereas, unSupervised Learning Algorithms (un-SLAs) do not require parameter tuning or data labeling[22]. Chan-Maestas and Sofge[23] performed tactile perception based on K-Means Clustering (KMC); and Shimoe et al.[24] developed artificial haptic models with Neural Network (NN) technology.

From the prior research presented above, we can establish that the tactile sensor and IP for an SRH should include the following features:

(1) The sensors need to be flexible for compatibility with SRHs;

(2) The sensors need to be multi-modal to describe different aspects of tactile information; and

(3) The IP should have various recognition abilities with the help of ML.

Therefore, this paper focuses on ML-based multi-modal IP for Soft Robotic Hands (SRHs). The paper is organized as follows. Section 2 introduces two modal sensors: the Optical Fiber-based Curvature Sensor (OFCS) and the pressure sensor, and the double-modal IP system is constructed. Section 3 describes the information pre-processing of the double modal sensors, explains the datasets for ML algorithms, and then provides the pattern recognition for IP. In Section 4, the ML algorithms are implemented, and the recognition results of gesture, object shape, size, and weight are shown. Finally, Section 5 presents the conclusion of this paper.

2 Double-Modal Sensors and IP System

2.1 OFCS

2.1.1 Fabrication

Figure 1 illustrates the fabrication of OFCS. The top raw figures describe the fabrication of a 3D model, whereas the bottom raw figures depict the fabrication of a real object. The curvature sensor is made of Light-Emitt Diode (LED), optical fiber, a photosensitive sensor, and two pieces of thermo-plastic tube (see Fig. 1a). The optical fiber is bent to a U shape (see Fig. 1b), which is able to be conveniently embedded into the soft finger. The surface of the optical fiber is roughened by engraving it with a laser cutting machine (see Fig. 1c). The LED is then attached to one end of the optical fiber with the help of a thermo-plastic tube, and the photosensitive sensor is attached to the other end. The OFCS is then fabricated (see Fig. 1d).

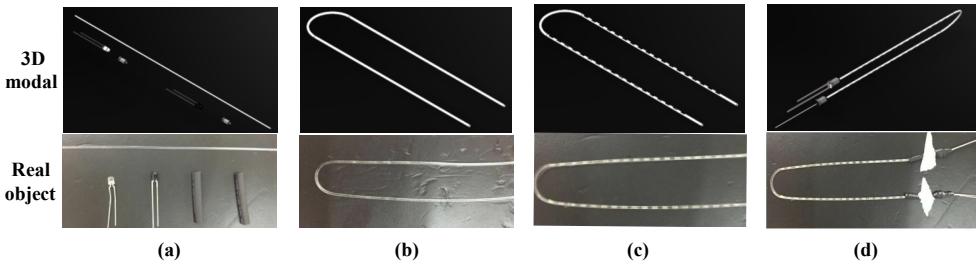


Fig. 1 Fabrication of OFCS.

2.1.2 Roughening process

The process of roughening optical fiber is shown in Fig. 2. The U-shaped optical fibers are affixed to a hardboard, and placed on the working plane of laser cutting machine (see Fig. 2a). The laser head cuts one side of the optical fiber when it is operating along the motion trajectory (see Fig. 2b). The roughened optical fiber features equal intervals between adjacent roughened segments, as shown in Fig. 2c. Figure 2d shows the motion trajectory of the laser head for the roughening of 15 pieces of optical fiber, while the detailed trajectory for every piece of optical fiber is shown in Fig. 2e. There are 19 segments of roughened area for every piece of optical fiber, and 25 straight lines for every segment of roughened area. The lengths of each roughened and un-roughened area are equal at 2 mm.

2.1.3 Integrated in SRH

Figure 3 shows the SRH with integrated OFCSs. Each OFCS is embedded into a soft finger, such that the curvature sensor bends in unison with the soft finger. The five soft fingers with embedded curvature sensors are integrated on a palm support. The effective length of the middle finger is 80 mm, while the other fingers are 70 mm long.

2.1.4 Measurement principle

The measurement principle of the OFCS is shown in Fig. 4. The OFCS and soft finger are combined, and are therefore bent and unbent synchronously. The light is emitted from an LED, transmitted through optical fiber, and received by the photosensitive sensor. In Fig. 4a, the light transmits through an unprocessed optical fiber (not roughened or bent), such that there is no loss of light, with all of the light received by the photosensitive sensor. In Fig. 4b, the optical fiber is roughened, so that some light is lost by being shut out. In Fig. 4c, the finger is bent, and the lost light increases. As the degree of bending increases, the photosensitive sensor receives less reserved light.

Assuming that the roughened optical fiber is in an initial state (Fig. 4b) without bending, stretching, or pressing, the baseline light energy as measured by the photosensitive sensor is expressed as I_0. When the roughened optical fiber is bent (Fig. 4c), some light

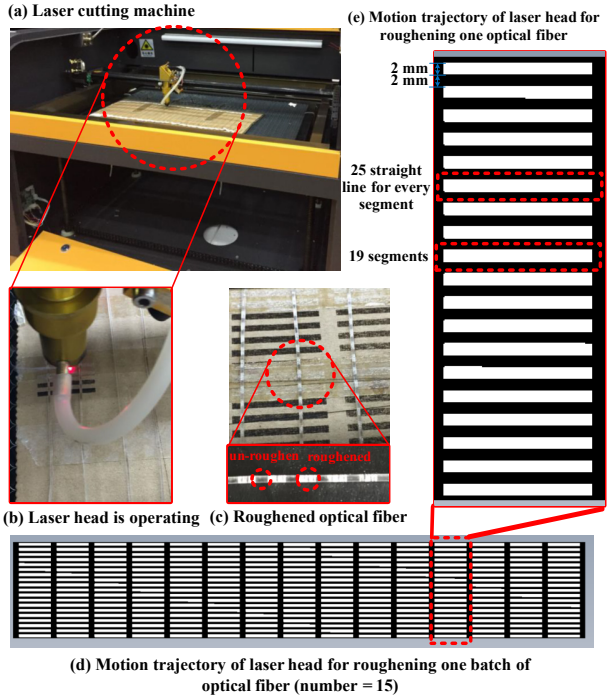


Fig. 2 Process of roughening optical fiber.

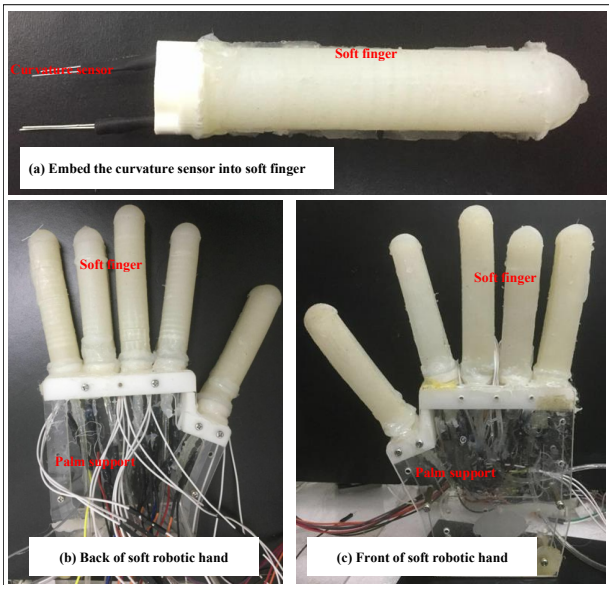


Fig. 3 SRH with OFCSs.

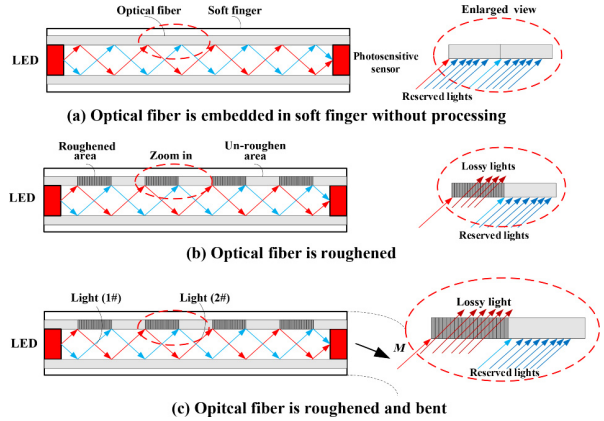


Fig. 4 Measurement principle of OFCS.

energy is lost, and the light energy as measured by the photosensitive sensor is expressed as I. The lossy of light energy can then be expressed in decibels as

$$a = 10\log_{10}^{I_0/I} \tag{1}$$

When the optical fiber is in the initial state, $a = 0$; when the light energy is increasing, $a < 0$; and when the light energy is reducing, $a > 0$.

2.2 Pressure sensor

The pressure sensor is an intelligent digital display pressure transmitter, as shown in Fig. 5a. In every gas pressure channel, the pressure transmitter is installed in parallel with the soft finger. The electrical connections are given in Fig. 5b. The current acquisition module converts the current signal to a voltage signal which is acquired by an Analog-Digital Converter (ADC) on the control board.

2.3 Double-modal IP system

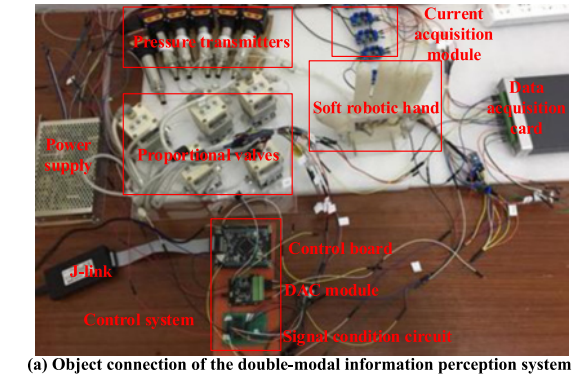
The double-modal IP system is shown in Fig. 6. The object connection is shown in Fig. 6a, while the control principle is shown in Fig. 6b. The CPU of the control board is an STM32 (STM32F407, STMicroelectronics Inc.). The proportional valves are controlled with the voltage output from the DAC module, and output controllable pressure to the soft fingers. The pressure transmitters output a current signal which is processed by the current acquisition module and acquired by the data acquisition card. The OFCSs are embedded in soft fingers and also output voltage type signals acquired by the data acquisition card. A PC receives the sensor data from the data acquisition card via Universal Serial Bus (USB).

3 Data Processing for ML Algorithms

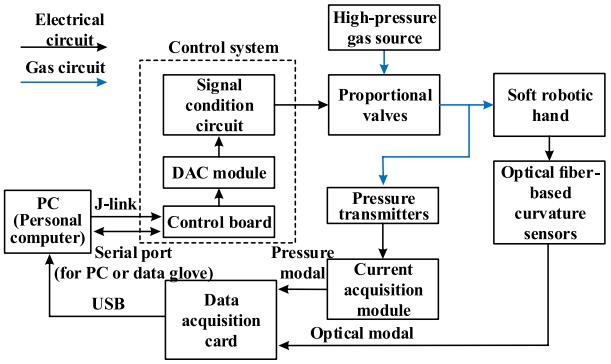
3.1 Information pre-processing

3.1.1 Information expression

The voltage is used to express the Pressure Sensor Information (PSI) and Optical Sensor Information



(a) Object connection of the double-modal information perception system



(b) Control principle of the double-modal information perception system

Fig. 6 Double-modal IP system.

(OSI). When the soft fingers are unbent, the PSI is expressed as $Vp_{i,0}$ (here and in what follows, $i = 0, 1, 2, 3, 4$ represent the thumb, index, middle, ring, and little fingers, respectively), and the OSI is expressed as $Vo_{i,0}$. When the soft fingers are bent to their maximum state, the PSI is expressed as $Vp_{i,\max}$, and the OSI is expressed as $Vo_{i,\max}$. Therefore, when the soft fingers are in states of bending between the initial and maximum, the PSI Vp_i and OSI Vo_i are expressed, respectively, as

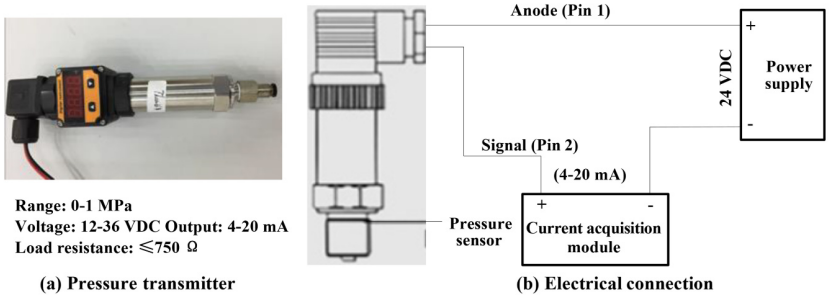
$$Vp_i \in [Vp_{i,0}, Vp_{i,\max}] \tag{2}$$

and

$$Vo_i \in [Vo_{i,0}, Vo_{i,\max}] \tag{3}$$

3.1.2 Normalization processing

The necessary to execute normalization processing is



(a) Pressure transmitter (b) Electrical connection

Fig. 5 Intelligent digital display pressure transmitter and its connection circuit

based on several considerations: (1) The output voltages of PSI and OSI are different for different soft fingers in their initial or maximum state; (2) The voltage ranges between PSI and OSI are not at the same level; and (3) The ML algorithms applied in the IP system require normalized data.

For PSI, the normalization process is expressed as

$$\mathrm{NOR}p_i = \frac{Vp_i - Vp_{i,0}}{Vp_{i,\max} - Vp_{i,0}} \tag{4}$$

While for OSI, the normalization process is expressed as

$$\mathrm{NOR}o_i = \frac{Vo_i - Vo_{i,0}}{Vo_{i,\max} - Vo_{i,0}} \tag{5}$$

3.2 Datasets for ML algorithms

3.2.1 Dataset pattern

The dataset pattern arising from N-dimension samples and M-dimension features is shown in Table 1. X_n $(n = 1, 2, \ldots, N)$ is the feature vector of the n-th sample, while x_{nm} $(m = 1, 2, \ldots, M)$ is the feature value corresponding to the m-th feature of the n-th sample. y_n is the class label corresponding to the n-th sample. $y_n \in c_m$ $(m = 1, 2, \ldots, M)$, where m is the object class, and the total number of classes is M.

3.2.2 Datasets for different ML algorithms

The related datasets for use with ML algorithms are shown in Fig. 7. These datasets are set up according to the dataset pattern given in Table 1.

The dataset with labels for use with SLAs is expressed as

$$DS\_super = \{(X_1, y_1), (X_2, y_2), \ldots, (X_N, y_N)\}^{\mathrm{T}} \tag{6}$$

Table 1 Dataset pattern.

Sample	Feature					
	Dim.1	Dim.2	Dim.3	$\cdots$	Dim.M	Label
X_1	x_{11}	x_{12}	x_{13}	$\cdots$	x_{1M}	y_1
X_2	x_{21}	x_{22}	x_{23}	$\cdots$	x_{2M}	y_2
X_3	x_{31}	x_{32}	x_{33}	$\cdots$	x_{3M}	y_3
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
X_N	x_{N1}	x_{N2}	x_{N3}	$\cdots$	x_{NM}	y_N

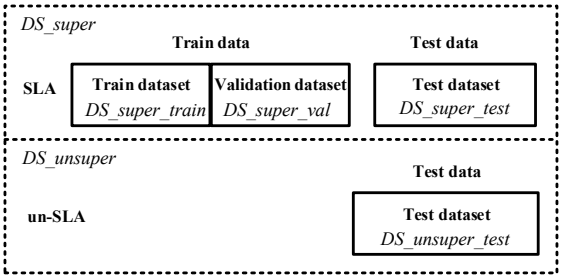


Fig. 7 Datasets for different ML algorithms.

The dataset without labels for use with the un-SLA is expressed as

$$DS\_unsuper = \{X_1, X_2, \ldots, X_N\}^{\mathrm{T}} \tag{7}$$

For SLAs, there are two kinds of data. One kind is the training data for training model parameters, which includes a training dataset and a validation dataset for cross-validation of the model parameters. The other kind is the test data, which includes a test dataset for predicting outcomes and evaluating the accuracy of the algorithms.

Training dataset for SLAs:

$$DS\_super\_train = \{(X_1, y_1), (X_2, y_2), \ldots, (X_{N1}, y_{N1})\}^{\mathrm{T}} \tag{8}$$

Validation dataset for SLAs:

$$DS\_super\_val = \{(X_1, y_1), (X_2, y_2), \ldots, (X_{N2}, y_{N2})\}^{\mathrm{T}} \tag{9}$$

Test dataset for SLAs:

$$DS\_super\_test = \{(X_1, y_1), (X_2, y_2), \ldots, (X_{N3}, y_{N3})\}^{\mathrm{T}} \tag{10}$$

where $N1$, $N2$, and $N3$ are the total number of samples in the datasets *DS_super_train*, *DS_super_val*, and *DS_super_test*, respectively.

For the un-SLA, no training data are not required as no model parameters needed to be trained, but a test dataset is used to predict outcomes and evaluate the accuracy of the algorithm.

Test dataset for un-SLA:

$$DS\_unsuper\_test = \{X_1, X_2, \ldots, X_{N4}\}^{\mathrm{T}} \tag{11}$$

where $N4$ is the total number of samples in the dataset *DS_unsuper_test*.

3.2.3 Pattern recognition for IP

As shown in Fig. 8, the IP includes the recognitions of gesture, object shape, size, and weight. The pattern recognition methods include SLAs and an un-SLA. The

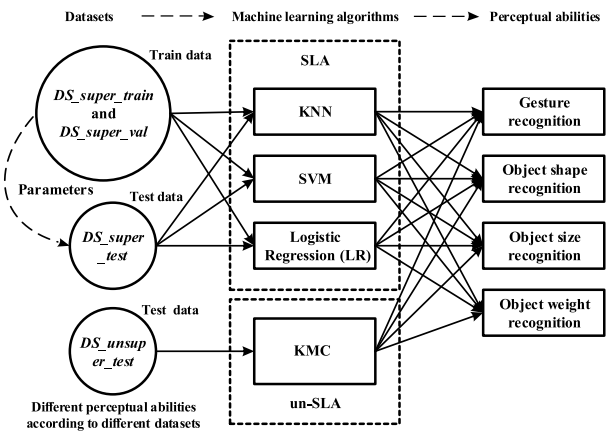


Fig. 8 Pattern recognition for IP.

adopted SLAs are KNN, SVM, and Logistic Regression (LR), whereas the adopted un-SLA is KMC. For the SLAs, the training data are used to derive optimal model parameters, then the test dataset is used to predict outcomes and evaluate the recognition accuracy according to these trained model parameters. For the un-SLA, the test data are used to perform clustering and evaluate the recognition accuracy.

In order to compare the recognition accuracies with information from different sensors, PSI, OSI, and Double-Sensor Information (DSI) are employed. There are five channels of PSI, and five channels of OSI. Therefore, the dimension of features is 5 ($M = 5$) in *DS_super_train*, *DS_super_test*, and *DS_unsuper_test* when either PSI or OSI is used, while the dimension of features is 10 ($M = 10$) when DSI is used.

4 IP-Based ML Algorithms

4.1 Implementation of the ML algorithms

4.1.1 KNN

The Euclidian distance is employed to perform the KNN algorithm, which is expressed as

$$DistE(X_t, X_v) = \sqrt{\sum_{j=0}^{M}(X_{tj} - X_{vj})^2} \qquad (12)$$

where $X_v = \{x_{v1}, x_{v2}, \ldots, x_{vM}, y_v\}$ is one validating vector in *DS_super_val*; $v = 1, 2, \ldots, N2$; $X_t = \{x_{t1}, x_{t2}, \ldots, x_{tM}, y_t\}$ is one training vector in *DS_super_train*; and $t = 1, 2, \ldots, N1$.

In the KNN algorithm, the number of nearest neighbors (k) is the unknown parameter to be obtained.

4.1.2 SVM

For the SVM algorithm, the software development kit named *Libsvm* is used to solve multi-classification problems. The Gaussian-based Radial Basis Function (RBF) as a kernel function is expressed as

$$W(X_v, X_t) = \exp\left(\frac{||(X_v - X_t)||^2}{2\sigma^2}\right) \qquad (13)$$

where σ is variance which is an adjustable parameter.

The function *SVMcgForClass* is used to find the optimal penalty parameter C and kernel parameter g. The function *libsvmtrain* is used to train the model for the SVM algorithm, and the function *libsvmpredict* is used to test the accuracy of the model.

In the SVM algorithm, the penalty parameter C and kernel parameter g need to be obtained.

4.1.3 LR

For the LR algorithm, the function *OneVsAll* is used to train the model, and the function *predictOneVsAll* is used to test the accuracy of the model. In the LR algorithm, the regularization parameter λ is needed, which is obtained by searching from 0.001 to 1 at an interval of 0.001 while repeating the functions *OneVsAll* and *predictOneVsAll*.

4.1.4 KMC

For the KMC algorithm, the function *kmeans* in Matlab is used to implement clustering, and the function *silhouette* in Matlab is used to plot the cluster silhouettes. The function *princomp* in Matlab is used to perform dimensionality reduction for intuitively displaying the cluster result.

4.2 Gesture recognition

4.2.1 Gesture exhibition

Figure 9 shows that the SRH performs twelve gestures following the movement of data glove for gesture recognition. The gestures are one, two, three, four, five, six, big crawl (B. C.), good, middle crawl (M. C.), ok, rock, and small crawl (S. C.).

4.2.2 Recognition results

(1) SLAs

The training data for gesture recognition is given in Fig. 10. The parameters for the SLAs are shown in Table 2.

After training the model parameters, the test experiments were performed. In the test experiments, every gesture was repeated three times, and 200 samples are recorded each time, such that 7200 ($N3 = 3 \times 200 \times$

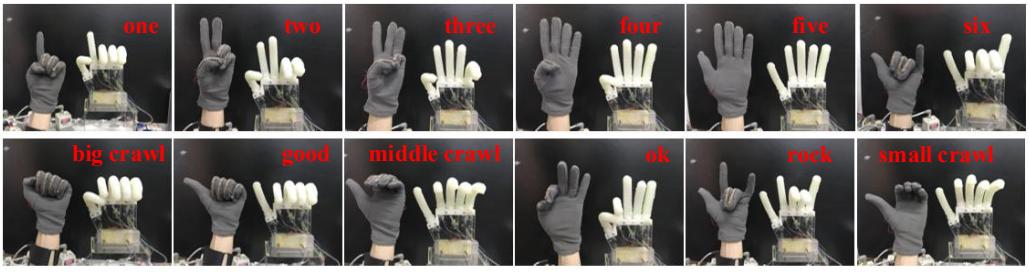


Fig. 9 Twelve gestures for gesture recognition.

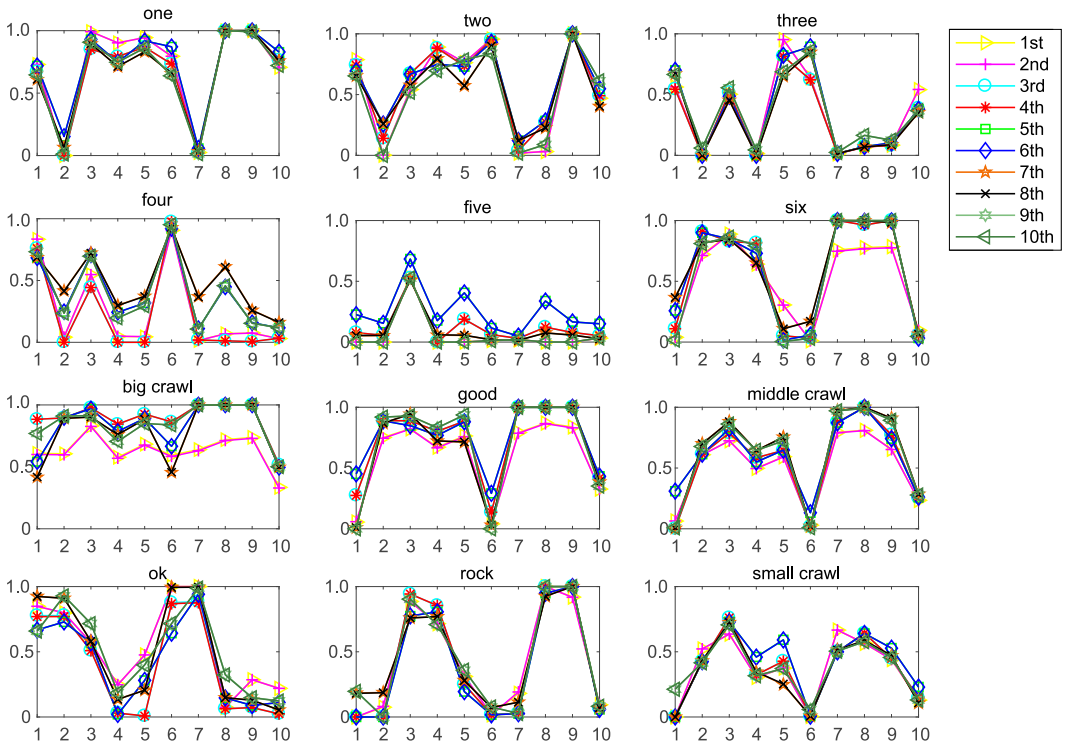


Fig. 10 Training data for gesture recognition (ten times), where 1–5 on the x-axis are corresponding to the PSI for thumb, index finger, middle finger, ring finger, and little finger, and 6–10 are corresponding to the OSI for five fingers respectively, the y-axis is the normalized values of PSI and OSI for every finger.

Table 2 Model parameters for SLAs for different pattern recognition.

Parameter		Gesture	Shape	Size	Weigh
k		9	9	9	9
DSI	C	5.278	0.003 903 6	0.009 765 6	0.003 903 6
	g	3.0314	0.003 903 6	0.003 903 6	0.003 903 6
PSI	C	0.003 906 3	0.003 906 3	0.003 906 3	0.003 906 3
	g	3.0314	0.003 906 3	0.003 906 3	0.003 906 3
OSI	C	5.278	0.003 906 3	0.003 906 3	0.003 906 3
	g	3.0314	0.003 906 3	0.003 906 3	0.003 906 3
λ		0.03	0.03	0.03	0.03

$12 = 7200$) samples are used to predict and evaluate the models of the SLAs.

Figure 11 shows the confusion matrixes of the test results for gesture recognition. The results indicate that most of the gestures are well recognized with each different combination of ML algorithm and sensor information. The gesture recognition accuracies using DSI are all greater than 50 percent and most of them are greater than 90 percent. The worst accuracy is 59.22 percent when the LR algorithm is used to recognize middle crawl. In this case, "M. C." is incorrectly recognized as "good" (21.88 percent) because the thumb is not clearly identified, and as small crawl (18.9 percent) because all the fingers are very close to the critical position. Further inaccuracies are (i) The subplot (1, 2) is the test result for the KNN algorithm with PSI, where subplot (x, y) means that the subploy at row x and column y, and "ok" is recognized as "four" (44.08 percent) because the index finger is not clearly identification. (ii) The subplot (1, 3) gives the test results for the KNN algorithm with OSI, where "six" is recognized as "M. C." (88.2 percent) because the thumb and little fingers are not sufficiently unbent. (iii) The subplot (3, 3) displays the test results for the LR algorithm with OSI, where "M. C." is recognized as "good" (76.05 percent) because the thumb is not sufficiently bent.

Figure 12 summarizes the average accuracies of gesture recognition. Figure 12a shows that the highest average accuracy is 97.96 percent, achieved using the KNN algorithm with Double-Sensor Information (DSI). Figure 12b shows that the average accuracy of DSI (96.70 percent) is the highest, followed by PSI at 92.59 percent, with OSI the least accurate at 88.12 percent. Figure 12c shows that the average accuracy of the KNN algorithms using DSI is the highest at 97.96 percent, followed by the SVM algorithm at 96.55 percent, with the LR algorithm the least accurate at 95.60 percent. The results for gesture recognition using SLAs can

KNN	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0.4
six	0	0	0	0	0	79	0	0	21	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0	0	0	100	0	0	0
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0	0	0	0	0	0	100	0
S.C.	0	0	0	0	3.1	0	0	0	0	0	0	97

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	62	0	0	0	0	0	0	38
six	0	0	0	0	0	89	0	4.5	6.4	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0	7.8	0	92	0	0	0
ok	0	0	0	44	0	0	0	0	0	56	0	0
rock	0	0	0	0	0	0	0	0	0	0	100	0
S.C.	0	0	0	4	0	0	0	0	34	0	0	62

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	67	33	0	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0
six	0	0	0	0	0	12	0	0	88	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	97	2.8	0	0	0
M.C	0	0	0	0	0	0	0	0	100	0	0	0.4
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0	0	0	0	0	0	84	16
S.C.	0	0	0	0	10	0	0	0	0	0	0	90

SVM	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0
six	0	0	0	0	0	100	0	0	0	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0.3	0	33	67	0	0	0
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0.7	0	0	0	0	0	99	0
S.C.	0	0	0	0	7.9	0	0	0	0	0	0	92

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	62	0	0	0	0	0	0	38
six	0	0	0	0	0	93	0	0.4	6.1	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0	0	0	100	0	0	0
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0	0	0	0	0	0	100	0
S.C.	0	0	0	1.8	0	0	0	0	34	1.8	0	62

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0
six	0	0	0	0	0	71	0	0	29	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	97	2.8	0	0	0
M.C	0	0	0	0	0	0.3	0	0	100	0	0	0
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	14	0	0	0	0	0	85	1.7
S.C.	0	0	0	0	17	0	0	0	0	0	0	83

LR	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0
six	0	0	0	0	0	100	0	0	0	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0	0	22	59	0	0	19
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0	0	0	0	0	0	100	0
S.C.	0	0	0	0	12	0	0	0	0	0	0	88

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	100	0	0	0	0	0	0	0	0	0	0	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	100	0	0	0	0	0	0	0	0	0
four	0	0	0	100	0	0	0	0	0	0	0	0
five	0	0	0	0	64	0	0	0	0	0	0	36
six	0	0	0	0	0	100	0	0	0	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	0	0	100	0	0	0	0
M.C	0	0	0	0	0	0	0	0	99	0	0	1.2
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	0	0	0	0	0	0	100	0
S.C.	0	0	0	0	0	0	0	0	5.3	3.3	0	91

	one	two	three	four	five	six	B.C.	good	M.C	ok	rock	S.C.
one	97	0	0	0	0	0	0	0	0	0	3.1	0
two	0	100	0	0	0	0	0	0	0	0	0	0
three	0	0	99	0	0.5	0	0	0	0	0	0	0
four	0	0	0	69	31	0	0	0	0	0	0	0
five	0	0	0	0	100	0	0	0	0	0	0	0
six	0	0	0	0	0	67	0	0	33	0	0	0
B.C.	0	0	0	0	0	0	100	0	0	0	0	0
good	0	0	0	0	0	1.1	0	99	0	0	0	0
M.C	0	0	0	0	0	0	0	76	15	0	0	9.4
ok	0	0	0	0	0	0	0	0	0	100	0	0
rock	0	0	0	0	1.7	0	0	0	0	0	98	0
S.C.	0	0	0	0	22	0	0	0	0	0	0	78

Fig. 11 Confusion matrixes of the test results for gesture recognition (%): KNN in Row 1, SVM in Row 2, and LR in Row 3.

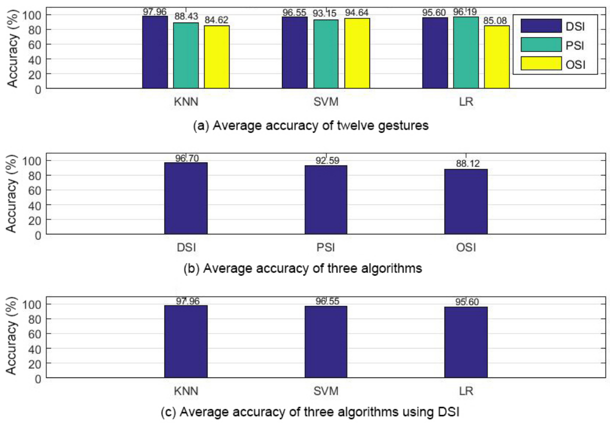


Fig. 12 Average accuracies of gesture recognition.

be summarized as follows: (1) Accuracy is highest using DSI, which indicates that increasing dimensions of features makes it easier to distinguish the gestures; (2) Different sensor information has little effect with the SVM algorithm; and (3) The KNN algorithm combined with DSI is the optimal method for gesture recognition.

(2) un-SLA

Figure 13 shows the clustering results for gesture recognition using the KMC algorithm. The clustering data is 7200 ($N4 = 7200$) samples (equal to the number of test samples used for SLAs). Looking at the subplots on the left of Fig. 13, for the DSI all the silhouette coefficients are positive and most are greater than 0.7, whereas for PSI and OSI a few silhouette coefficients are negative, although most are greater than 0.6. Looking at the cluster scatter map on the right of the figure, the clustering effect is shown visually by reducing dimensions with the PCA method. The number of errors for DSI, PSI, and OSI are 228, 548, and 1027, respectively, out of the total of 7200 samples, corresponding to accuracies of 96.83%, 92.38%, and 85.73%. This indicates that DSI is superior to both PSI and OSI.

4.3 Object recognition: Shape, size, and weight

4.3.1 Grab objects

Figure 14 shows how the SRH grabs objects with different shapes, sizes, and weights. In Row 1, there are four different shapes all with the same size of 40 mm: sphere, regular tetrahedron, cuboid, and cylinder. In Row 2, there are four spheres of different diameters: 60 mm, 50 mm, 40 mm, and 30 mm. In Row 3, there are three cylindrically shaped cups: the first cup is empty with a weight of 38 g, the second cup is filled with green beans with a weight of 138 g, and the last cup is filled with metal components with a weight of 238 g. The SRH adopts the same gesture of a big crawl to grab every one of the objects.

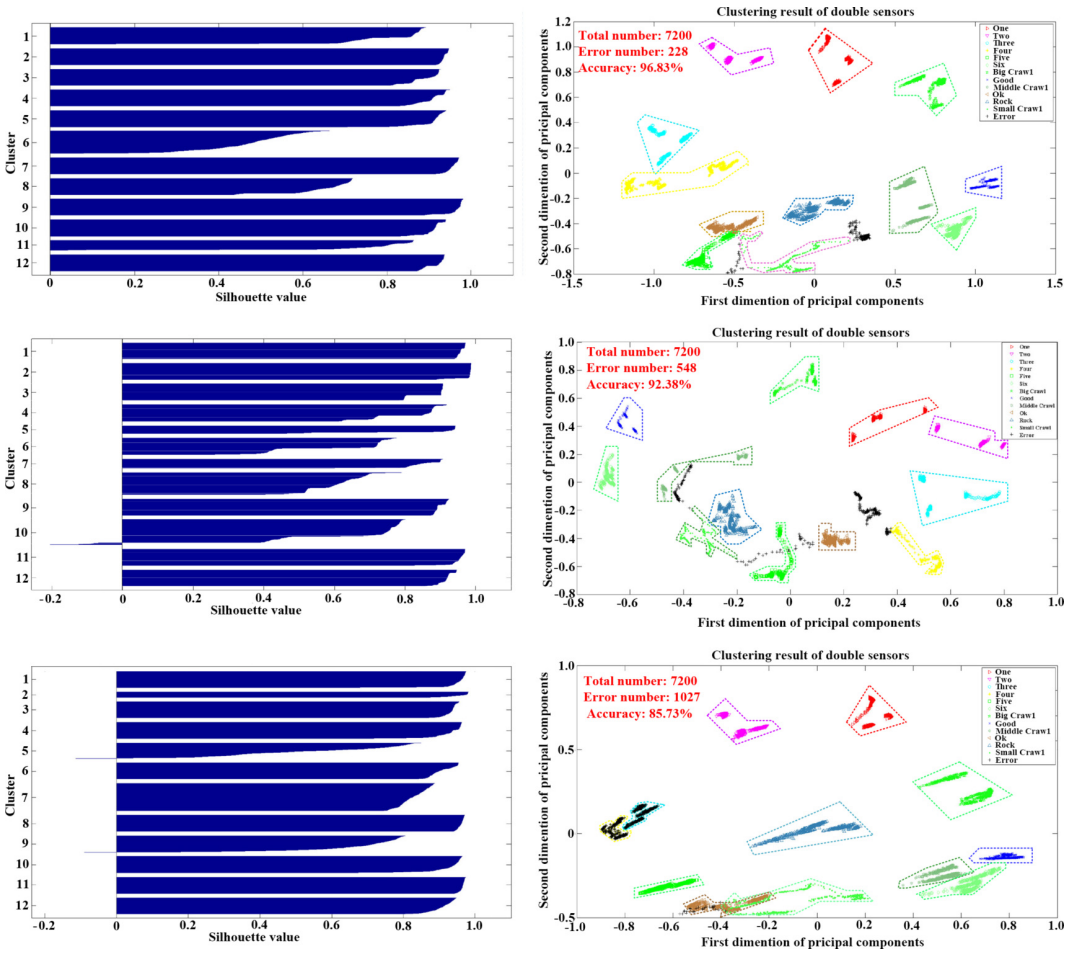


Fig. 13 Clustering results for gesture recognition by using KMC algorithm: DSI in Row 1, PSI in Row 2, and OSI in Row 3, and the left figures represent silhouette plots and the right figures represent cluster scattep maps.

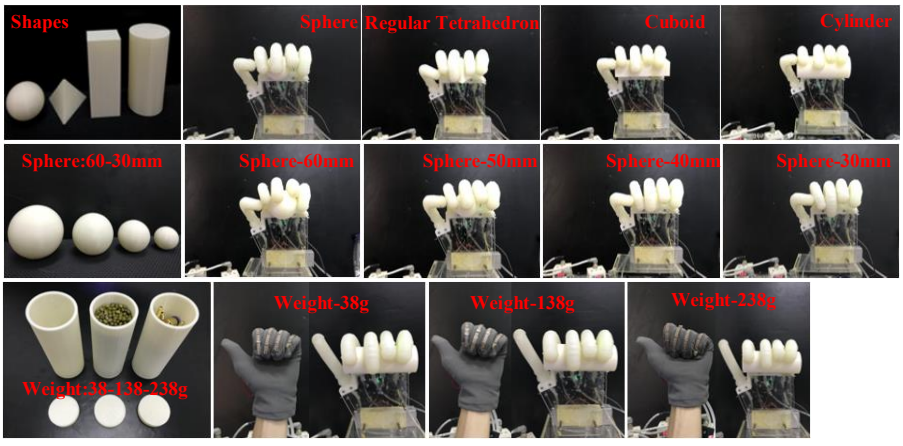


Fig. 14 SRH grabs different objects: different shapes in Row 1, different sizes in Row 2, and different weights in Row 3.

4.3.2 Recognition results

(1) SLAs

(i) Shape. The training data for object shape recognition is given in Fig. 15. The parameters for SLAs are shown in Table 2.

The test experiments for every shape are repeated three times, 200 samples are recorded each time, and the best group data is selected to predict and evaluate the models of the SLAs, so that the total number of test samples is 800 ($N3 = 200 \times 4 = 800$).

Figure 16 shows the confusion matrixes of the

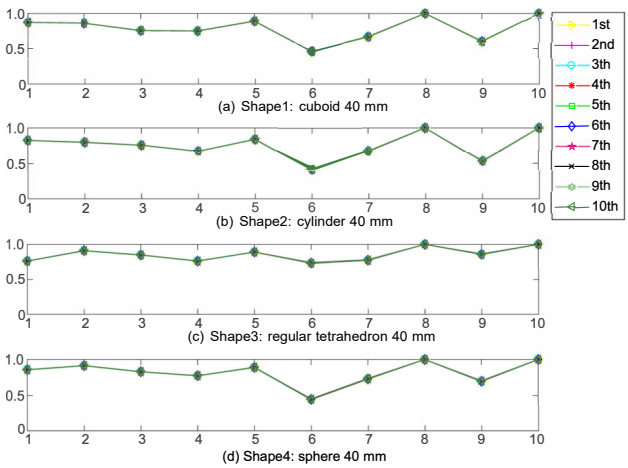


Fig. 15 Training data for object shape recognition (ten times), where 1–5 on the *x*-axis are corresponding to the PSI for thumb, index finger, middle finger, ring finger, and little finger, and 6–10 are corresponding to the OSI for five fingers respectively, the *y*-axis is the normalized values of PSI and OSI for every finger.

test results for object shape recognition. The results reveal that the regular tetrahedron (R.T.) and sphere are perfectly recognized with accuracies of 100 percent. The reason for this is that the shapes of regular tetrahedron and sphere differ greatly from each other. Conversely, the cuboid shape is similar to a cylinder, so these are confused frequently. The accuracy is 2.75 percent for cuboid shape recognition using the LR algorithm with PSI. The accuracy is 46.25 percent for cylinder shape recognition using the KNN algorithm with OSI, and 45.75 percent for cylinder shape recognition using the SVM algorithm with OSI. Considering the sensor information, the best recognition accuracy is achieved with DSI; considering the ML algorithms, the KNN algorithm is the most accurate.

The highest average accuracy is 98.06 percent using the KNN algorithm with DSI. The average accuracy for DSI is 97.40 percent, for OSI is 87.21 percent, and for PSI is 83.71 percent. The average accuracy for the KNN algorithm is 90.81, for the SVM algorithm is 90.56 percent, and for the LR algorithm is 86.94 percent.

(ii) Size. The training data for object size recognition is given in Fig. 17. The parameters for the SLAs are shown in Table 2.

The test experiments are repeated three times for every size, 200 samples are recorded each time, and the best group data is selected to predict and evaluate the models of the SLAs, such that the total number of application samples is 800 ($N3 = 200 \times 4 = 800$).

Figure 18 shows the confusion matrixes of the test

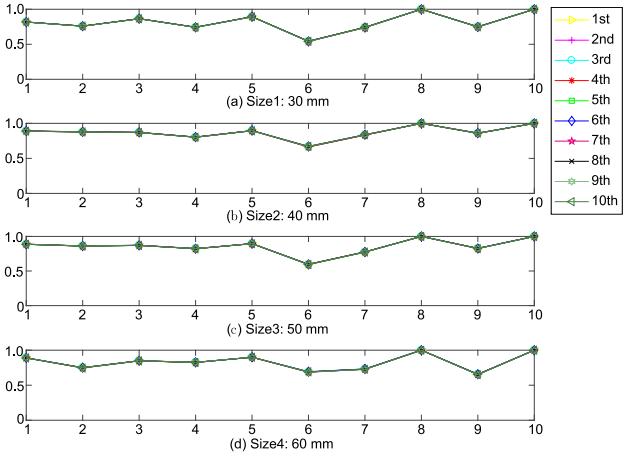


Fig. 17 Training data for object size recognition (ten times), where 1–5 on the *x*-axis are corresponding to the PSI for thumb, index finger, middle finger, ring finger, and little finger, and 6–10 are corresponding to the OSI for five fingers respectively, the *y*-axis is the normalized values of PSI and OSI for every finger.

		DSI				PSI				OSI			
		cuboid	cylinder	R.T.	sphere	cuboid	cylinder	R.T.	sphere	cuboid	cylinder	R.T.	sphere
KNN	cuboid	99.75	0.25	0	0	54.5	45.5	0	0	100	0	0	0
	cylinder	7.5	92.5	0	0	1.25	98.75	0	0	53.75	46.25	0	0
	R.T.	0	0	100	0	0	0	98	2	0	0	100	0
	sphere	0	0	0	100	0	0	0	100	0	0	0	100
		cuboid	cylinder	R.T.	sphere	cuboid	cylinder	R.T.	sphere	cuboid	cylinde	R.T.	sphere
SVM	cuboid	100	0	0	0	54.75	45.25	0	0	100	0	0	0
	cylinder	10.5	89.5	0	0	1.25	98.75	0	0	54.25	45.75	0	0
	R.T.	0	0	100	0	0	0	98	2	0	0	100	0
	sphere	0	0	0	100	0	0	0	100	0	0	0	100
		cuboid	cylinder	R.T.	sphere	cuboid	cylinde	R.T.	sphere	cuboid	cylinder	R.T.	sphere
LR	cuboid	87.5	12.5	0	0	2.75	97.25	0	0	94.75	5.25	0	0
	cylinder	0.5	99.5	0	0	0.25	99.75	0	0	40.25	59.75	0	0
	R.T.	0	0	100	0	0	0.75	99.3	0	0	0	100	0
	sphere	0	0	0	100	0	0	0	100	0	0	0	100

Fig. 16 Confusion matrixes of the test results of object shape recognition (%): KNN in Row 1, SVM in Row 2, and LR in Row 3.

		DSI				PSI				OSI			
		30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm
KNN	30 mm	87.5	0	12.5	0	87.25	0	12.75	0	90	0	10	0
	40 mm	0	99.5	0.5	0	0	100	0	0	0	94	0	6
	50 mm	2.75	3.5	93.75	0	0	20.5	79.5	0	35.75	2.75	58	3.5
	60 mm	0	0	0	100	0	0	0	100	0	0	0	100
		30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm
SVM	30 mm	87.25	0	12.75	0	87.25	0	12.75	0	90	0	10	0
	40 mm	0	99.5	0.5	0	0	100	0	0	0	94.25	0	5.75
	50 mm	2.75	3.5	93.75	0	0	20.25	79.75	0	37	3.5	59	0.5
	60 mm	0	0	0	100	0	0	0	100	0	0	0	100
		30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm	30 mm	40 mm	50 mm	60 mm
LR	30 mm	85	0	15	0	93.75	0	6.25	0	80.25	0	19.75	0
	40 mm	0	100	0	0	0	100	0	0	0	83.25	0	16.75
	50 mm	0.25	27.5	49.25	23	0	35.25	64.75	0	19.75	32	0.5	47.75
	60 mm	0	0	0	100	0	0	0	100	0	0	0	100

Fig. 18 Confusion matrixes of the test results of object size recognition (%): KNN in Row 1, SVM in Row 2, and LR in Row 3.

results for object size recognition. Using DSI achieves superior object size recognition than PSI or OSI. Considering the different ML algorithms, the KNN algorithm is superior to the SVM and LR algorithms. The accuracies of recognizing a sphere size of 50 mm are generally lower than for the other sizes, perhaps because that this size is in the middle, and easy to confuse with the neighboring 40 mm and 60 mm sizes.

The highest average accuracy is 95.19 percent achieved using the KNN algorithm with DSI, while the worst average accuracy is 66.0 percent using the LR algorithm with OSI. The average accuracy for DSI is the highest at 91.29 percent, the OSI is the worst at 79.10 percent. The average accuracy for the SVM algorithm is 90.90 percent, for the KNN is 90.79 percent, and for the LR algorithm is lowest at 79.73 percent.

(iii) Weight. The training data for object weight recognition is given in Fig. 19. The parameters for the SLAs are shown in Table 2.

The test experiments for each weight are repeated three times, 200 samples are recorded each time, and the best group data is selected to predict and evaluate the models of the SLAs, such that the total number of test samples is 600 ($N3 = 200 \times 3 = 600$).

Figure 20 shows the confusion matrixes of the test results of object weight recognition. All the object weights are identified clearly regardless of which SLAs and sensor information are used. The average accuracies are therefore 100 percent.

(2) un-SLA

(i) Shape. Figure 21 shows the clustering results for object shape recognition using the KMC algorithm with DSI. The silhouette plot indicates that all the silhouette coefficients are positive and that almost of them are greater than 0.7. The cluster scatter map shows that the object shapes are clearly identified with no errors.

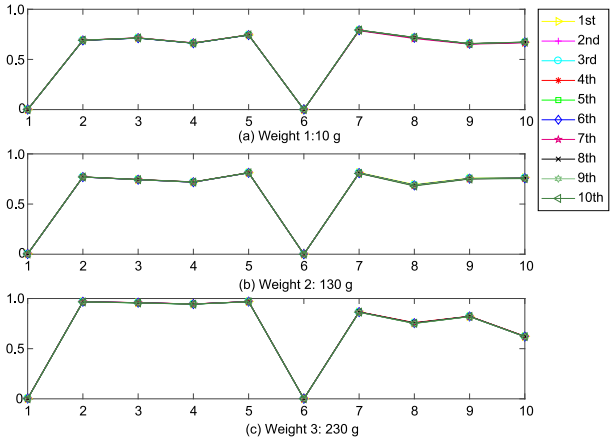
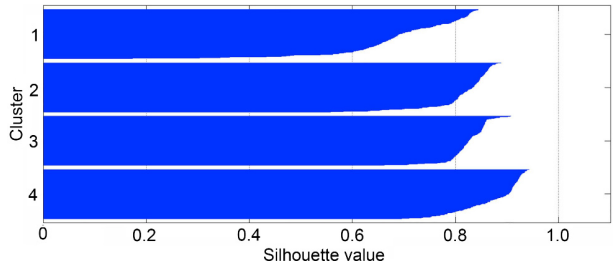


Fig. 19 Training data for object weight recognition (ten times), where 1–5 on the *x*-axis are corresponding to the PSI for thumb, index finger, middle finger, ring finger, and little finger, and 6–10 are corresponding to the OSI for five fingers respectively, the *y*-axis is the normalized values of PSI and OSI for every finger.

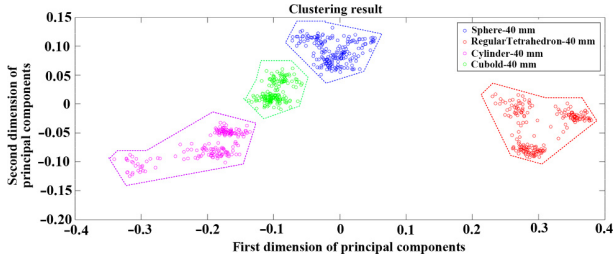
		DSI			PSI			OSI		
		38 g	138 g	238 g	38 g	138 g	238 g	38 g	138 g	238 g
KNN	38 g	100	0	0	100	0	0	100	0	0
	138 g	0	100	0	0	100	0	0	100	0
	238 g	0	0	100	0	0	100	0	0	100
		38 g	138 g	238 g	38 g	138 g	238 g	38 g	138 g	238 g
SVM	38 g	100	0	0	100	0	0	100	0	0
	138 g	0	100	0	0	100	0	0	100	0
	238 g	0	0	100	0	0	100	0	0	100
		38 g	138 g	238 g	38 g	138 g	238 g	38 g	138 g	238 g
LR	38 g	100	0	0	100	0	0	100	0	0
	138 g	0	100	0	0	100	0	0	100	0
	238 g	0	0	100	0	0	100	0	0	100

Fig. 20 Confusion matrixes of the test results of object weight recognition (%): KNN in Row 1, SVM in Row 2, and LR in Row 3.

(ii) Size. Figure 22 shows that the clustering results for object size recognition. The silhouette plot indicates

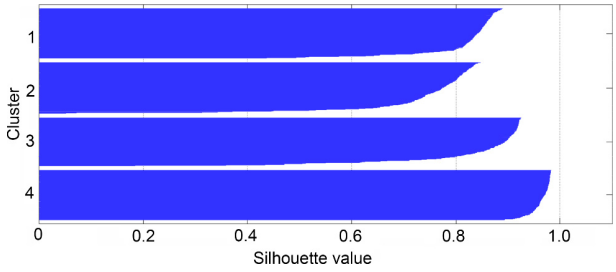


(a) Silhouette plot

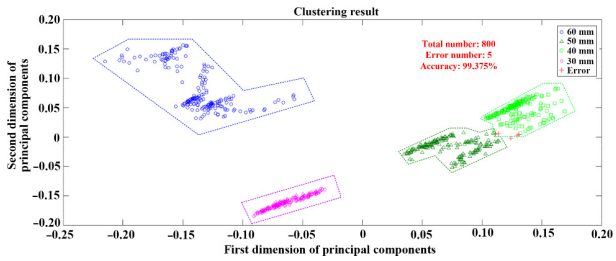


(b) Cluster scattep map

Fig. 21 Clustering results for object shape recognition.



(a) Silhouette plot

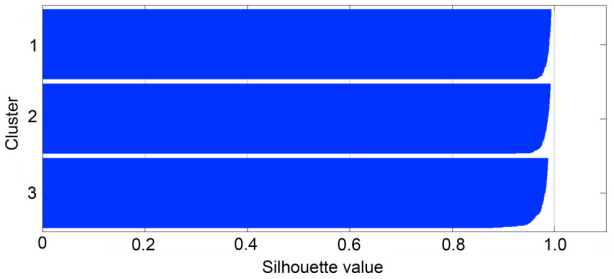


(b) Cluster scattep map

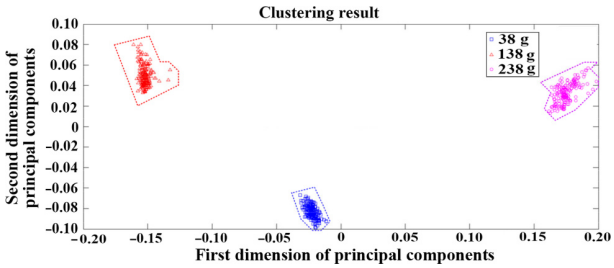
Fig. 22 Clustering results for object size recognition.

that all the silhouette coefficients are positive and that most of them are greater than 0.8. The cluster scatter map shows that the object shapes are able to be identified correctly in almost all cases, and the accuracy reaches 99.375 percent with five errors out of the total of 800 recognitions.

(iii) Weight. Figure 23 shows the clustering results of weight recognition. The silhouette coefficients are all near to 1, meaning that the effect of weight recognition is excellent. The cluster scatter map also shows that the different weights are identified clearly with no errors.



(a) Silhouette plot



(b) Cluster scattep map

Fig. 23 Clustering results for object weight recognition.

4.4 Summary

Comparing the experimental results of recognizing gesture, object shape, size, and weight using KNN, SVM, LR, and KCM with DSI, PSI, and OSI, several conclusions can be drawn: (1) The IP of the SRH is able to perform gesture recognition, and object shape, size, and weight recognitions using PSI and OSI, but the use of DSI results in superior recognition accuracy; (2) The SLAs and KMC algorithms are able to perform IP, but the KNN algorithm is superior to the others in most cases, and (3) The ideal combination for information perception is the KNN algorithm with DSI.

5 Conclusion

The OFCS is designed and embedded in the soft finger to provide curvature sensing. Combining OSI with PSI can provide dual-modal sensor information to perform information perception for an SRH. Three SLAs, namely the KNN, SVM, and LR algorithms, and one un-SLA, namely KMC, were excuted with dual-modal sensor information to recognize gestures made by an SRH and identify object shape, size, and weight. The experimental results demonstrate that the KNN algorithm with DSI is the best combination for information perception.

Current SRHs often have a single modal sensor, and lack of learning ability. Therefore, this research into ML-based multi-modal information perception

for SRHs presents a novel idea for SRHs and soft robots. Although the sensors are somewhat crude, and classical ML algorithms are adopted, the recognition effect is clear. In the future, to study in more depth the perception of SRHs for performing dexterous manipulation, we will conduct the following exploratory researches:

(1) Design a nanometer material-based pressure contact sensor for detecting positive pressure in the palm and three-dimensional force in the tip of a soft finger, which can increase multi-modal sensor information for tactile perception;

(2) Introduce a neural network-based ML algorithm to perform pattern recognition for tactile perception; and

(3) Develop a real-time tactile perception system in the FPGA-based system on chip for practical applications.

Appendix

A Training Data for Pattern Recognition

A.1 Gesture recognition

As shown in Fig. 10, the training data has 5-d PSI (1–5 on the x-axis) and 5-d OSI (6–10 on the x-axis), thus the number of dimensions of features is $M = 10$. The data are normalized data, with the value ranging from 0 to 1 on the y-axis.

The experiments for every gesture are repeated five times and more than 200 sampling values are recorded each time. The average values of the odd number series and even number series are calculated as the samples of the dataset. Therefore, there are $5 \times 2 \times 12 = 120$ samples. The number of samples in the training dataset is $N1 = 120 \times 75\% = 90$; in the validation dataset it is $N2 = 120 \times 25\% = 30$.

A.2 Object shape recognition

There are $5 \times 2 \times 4 = 40$ samples in the training data for object shape recognition ($N1 = 40 \times 75\% = 30$, $N2 = 40 \times 25\% = 10$). Figure 15 shows the training data for object shape recognition.

A.3 Object size recognition

There are 40 samples in the training data for object size recognition ($N1 = 30$, $N2 = 10$). Figure 17 shows the training data for object size recognition.

A.4 Object weight recognition

There are 30 samples in the training data for object weight recognition ($N1 = 20$, $N2 = 10$). Figure 19 shows the training data for object weight recognition.

B Model Parameters for SLAs

The model parameters for the SLAs for different pattern recognitions are given in Table 2.

Acknowledgment

The authors are grateful for the support provided by the National Natural Science Foundation of China (Nos. 61803267 and 61572328), the China Postdoctoral Science Foundation (No. 2017M622757), the Beijing Science and Technology program (No. Z171100000817007), and the National Science Foundation of China (NSFC) and the German Re- search Foundation (DFG) in the project Cross Modal Learning, NSFC 61621136008/DFG TRR-169. The authors are grateful for the support of Science and Technology Commissioned Project scheme of Shenzhen University.

References

[1] G. Ponraj, S. K. Kirthika, N. V. Thakor, C. H. Yeow, S. L. Kukreja, and H. L. Ren, Development of flexible fabric based tactile sensor for closed loop control of soft robotic actuator, in *Proc. 13th IEEE Conf. on Automation Science and Engineering*, Xi'an, China, 2017, pp. 1451–1456.

[2] J. H. Low, W. W. Lee, P. M. Khin, N. V. Thakor, S. L. Kukreja, H. L. Ren, and C. H. Yeow, Hybrid tele-manipulation system using a sensorized 3-D-printed soft robotic gripper and a soft fabric-based haptic glove, *IEEE Rob. Autom. Lett.*, vol. 2, no. 2, pp. 880–887, 2017.

[3] R. Deimel and O. Brock, A novel type of compliant and underactuated robotic hand for dexterous grasping, *Int. J. Rob. Res.*, vol. 35, nos. 1–3, pp. 161–185, 2016.

[4] R. Deimel, M. Radke, and O. Brock, Mass control of pneumatic soft continuum actuators with commodity components, in *Proc. 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Daejeon, South Korea, 2016, pp. 774–779.

[5] A. Gupta, C. Eppner, S. Levine, and P. Abbeel, Learning dexterous manipulation for a soft robotic hand from human demonstrations, in *Proc. 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Daejeon, South Korea, 2016, pp. 3786–3793.

[6] Y. F. Hao, T. M. Wang, and L. Wen, A programmable mechanical freedom and variable stiffness soft actuator with low melting point alloy, in *Proc. 10th Int. Conf. on Intelligent Robotics and Applications*, Wuhan, China, 2017, pp. 151–161.

[7] G. De Boer, N. Raske, H. B. Wang, J. Kow, A. Alazmani, M. Ghajari, P. Culmer, and R. Hewson, Design optimisation of a magnetic field based soft tactile sensor, *Sensors*, vol. 17, no. 11, p. 2539, 2017.

[8] Z. P. Ji, H. Zhu, H. C. Liu, N. Liu, T. Chen, Z. Yang, and

L. N. Sun, The design and characterization of a flexible tactile sensing array for robot skin, *Sensors*, vol. 16, no. 12, p. 2001, 2016.
[9] S. Seo, S. Kim, J. Jung, R. J. Ma, S. Baik, and H. Moon, Flexible touch sensors made of two layers of printed conductive flexible adhesives, *Sensors*, vol. 16, no. 9, p. 1515, 2016.
[10] W. Z. Yuan, S. Y. Dong, and E. H. Adelson, Gelsight: High-resolution robot tactile sensors for estimating geometry and force, *Sensors*, vol. 17, no. 12, p. 2762, 2017.
[11] Z. Kappassov, J. A. Corrales, and V. Perdereau, Tactile sensing in dexterous robot hands — Review, *Rob. Auton. Syst.*, vol. 74, pp. 195–220, 2015.
[12] L. Zou, C. Ge, Z. J. Wang, E. Cretu, and X. O. Li, Novel tactile sensor technology and smart tactile sensing systems: A review, *Sensors*, vol. 17, no. 11, p. 2653, 2017.
[13] H. C. Zhao, R. K. Huang, and R. F. Shepherd, Curvature control of soft orthotics via low cost solid-state optics, in *Proc. 2016 IEEE Int. Conf. on Robotics and Automation*, Stockholm, Sweden, 2016, pp. 4008–4013.
[14] H. C. Zhao, J. Jalving, R. K. Huang, R. Knepper, A. Ruina, and R. Shepherd, A helping hand: Soft orthosis with integrated optical strain sensors and EMG control, *IEEE Rob. Autom. Mag.*, vol. 23, no. 3, pp. 55–64, 2016.
[15] H. C. Zhao, K. O'Brien, S. Li, and R. F. Shepherd, Optoelectronically innervated soft prosthetic hand via stretchable optical waveguides, *Sci. Rob.*, vol. 1, no. 1, p. eaai7529, 2016.
[16] P. Kampmann and F. Kirchner, Integration of fiber-optic sensor arrays into a multi-modal tactile sensor processing system for robotic end-effectors, *Sensors*, vol. 14, no. 4, pp. 6854–6876, 2014.
[17] R. F. Xu, J. N. Hu, Q. Lu, D. Y. Wu, and L. Gui, An ensemble approach for emotion cause detection with event extraction and multi-kernel SVMs, *Tsinghua Sci. Technol.*, vol. 22, no. 6, pp. 646–659, 2017.
[18] Z. L. Yuan, Y. Q. Lu, and Y. B. Xue, Droiddetector: Android malware characterization and detection using deep learning, *Tsinghua Sci. Technol.*, vol. 21, no. 1, pp. 114–123, 2016.
[19] F. Wang, H. P. Liu, F. C. Sun, and H. H. Pan, Fabric recognition using zero-shot learning, *Tsinghua Sci. Technol.*, vol. 24, no. 6, pp. 645–653, 2019.
[20] F. Naya, J. Yamato, and K. Shinozawa, Recognizing human touching behaviors using a haptic interface for a pet-robot, in *Proc. 1999 IEEE Int. Conf. on Systems, Man, and Cybernetics*, Tokyo, Japan, 1999, pp. 1030–1034.
[21] H. H. Hu, Y. Z. Han, A. G. Song, S. G. Chen, C. H. Wang, and Z. Wang, A finger-shaped tactile sensor for fabric surfaces evaluation by 2-dimensional active sliding touch, *Sensors*, vol. 14, no. 3, pp. 4899–4913, 2014.
[22] X. Li and H. F. Liu, Greedy optimization for K-means-based consensus clustering, *Tsinghua Sci. Technol.*, vol. 23, no. 2, pp. 184–194, 2018.
[23] H. Chan-Maestas and D. A. Sofge, Tactile sensor system processing based on K-means clustering, in *Proc. 2011 10th Int. Conf. on Machine Learning and Applications and Workshops*, Honolulu, HI, USA, 2011, pp. 287–292.
[24] H. Shimoe, K. Matsumura, H. Noma, M. Sohgawa, and M. Okuyama, Development of artificial haptic model for human tactile sense using machine learning, in *Proc. 2011 IEEE SENSORS*, Glasgow, UK, 2017, pp. 1–3.

Haiming Huang received the PhD degree from Beihang University in 2016. He is currently an assistant professor in the College of Electronics and Information Engineering, Shenzhen University. His research interests include soft robotics, flexible sensor, embedded mechatronics control, and robotics.

Junhao Lin received the BS degree from Dongguan University of Technology in 2016. He is currently pursuing the master degree in the College of Information Engineering, Shenzhen University. His research interests include soft robotics and machine learning.

Linyun Wu received the BS degree from Shenzhen University in 2017. He is currently pursuing the MS degree at Shenzhen University. His research interests include soft robotics and embedded system.

Bin Fang received the PhD degree from Beihang University in 2014, and now he is the research assistant in the Department of Computer Science and Technology at Tsinghua University. His research interests include sensor fusion, and robotics and human-robot interaction.

Fuchun Sun received the PhD degree from Tsinghua University in 1997. He is a full professor with the Department of Computer Science and Technology, Tsinghua University, China. His current research interest includes robotic perception and cognition. He was a recipient of the National Science Fund for Distinguished Young Scholars. He serves as an associate editor of a series of international journals including *IEEE Transactions on Systems, Man and Cybernetics: Systems, IEEE Transactions on Fuzzy Systems, Mechatronics,* and *Robotics and Autonomous Systems*.

Zhenkun Wen received the MS degree from Tsinghua University in 1999. He is a full professor with the College of Computer Science and Software Engineering, Shenzhen University, China. His current research interest includes video information processing, intelligent analysis of medical big data, and large network information system.