sensors

MDPI

Article

FGFF Descriptor and Modified Hu Moment-Based Hand Gesture Recognition

Beiwei Zhang [1,*], Yudong Zhang [2], Jinliang Liu [1] and Bin Wang [1]

[1] School of Information Engineering, Nanjing University of Finance and Economics, Nanjing 210023, China; liujinliang@vip.163.com (J.L.); wangbin@nufe.edu.cn (B.W.)
[2] School of Computing and Mathematical Sciences, University of Leicester, Leicester LE1 7RH, UK; yudongzhang@ieee.org
* Correspondence: zhangbeiwei@nufe.edu.cn

Abstract: Gesture recognition has been studied for decades and still remains an open problem. One important reason is that the features representing those gestures are not sufficient, which may lead to poor performance and weak robustness. Therefore, this work aims at a comprehensive and discriminative feature for hand gesture recognition. Here, a distinctive Fingertip Gradient orientation with Finger Fourier (FGFF) descriptor and modified Hu moments are suggested on the platform of a Kinect sensor. Firstly, two algorithms are designed to extract the fingertip-emphasized features, including palm center, fingertips, and their gradient orientations, followed by the finger-emphasized Fourier descriptor to construct the FGFF descriptors. Then, the modified Hu moment invariants with much lower exponents are discussed to encode contour-emphasized structure in the hand region. Finally, a weighted AdaBoost classifier is built based on finger-earth mover's distance and SVM models to realize the hand gesture recognition. Extensive experiments on a ten-gesture dataset were carried out and compared the proposed algorithm with three benchmark methods to validate its performance. Encouraging results were obtained considering recognition accuracy and efficiency.

Keywords: FGFF descriptor; Hu moment invariants; finger thickness; hand gesture recognition; weighted AdaBoost classifier

Citation: Zhang, B.; Zhang, Y.; Liu, J.; Wang, B. FGFF Descriptor and Modified Hu Moment-Based Hand Gesture Recognition. *Sensors* **2021**, *21*, 6525. https://doi.org/10.3390/s21196525

Academic Editor: Junseop Lee

Received: 20 August 2021
Accepted: 28 September 2021
Published: 29 September 2021

1. Introduction

Hand gestures carry rich information and provide a natural yet important method for different people to interact in their daily life. They have been used as a friendly interface between humans and computer systems, which enables an intuitive and convenient human–computer interaction, and have found many applications in natural human–computer interaction, such as intelligent robot control, smart homing, virtual reality, computer games, and some quietness-required environments. In [1], the authors explored the recognition application of handwritten Arabic alphabets by tracking and modeling the motion of the hand. To this end, recent years have witnessed an active research interest in the field of hand gesture recognition and human action recognition.

Traditional vision-based recognition algorithms mainly utilize the information of color or texture from 2D RGB camera, which is typically affected by external environments such as illumination, skin color, and cluttered background. Their limitation is the loss of 3D structure information, which obviously decreases their robustness and accuracy. In order to improve the robustness and simplify the hand localization and segmentation, some researchers suggested the use of a colored glove or black belt on the wrist of the gesturing hand [2]. Furthermore, accelerometers, magnetic trackers, and data gloves are involved in obtaining the three-dimensional information of gesture for easy image processing and 3D motion capturing at the granularity of the fingers. However, these strategies are only suitable for handling some simple gestures. When the gesture becomes more complex, it will obviously reduce the recognition accuracy. Furthermore, it impedes the invisibility of

the interface for the users and brings increased inconvenience and can be cumbersome in some cases in which many cables may be involved [3].

Thanks to the development of inexpensive depth cameras, e.g., the Kinect sensor, a new and desirable method is provided to extract motion and visual information for human activities. Instead of wearing data gloves or any other auxiliary equipment, the gesturing hand can be detected and segmented efficiently with the Kinect sensor. Therefore, more and more research has paid attention to this platform in recent years, and the authors can be referred to [4,5] for a comprehensive review work. Basically, all of the existing algorithms can be classified into two categories, i.e., skeleton-based algorithms and depth-based algorithms, depending on the types of input data. The former uses 3D coordinates of the joints to represent the model of full human body. The method proposed by Thanh and Chen [6] falls into this type, which extracted the discriminative patterns as local features to classify skeleton sequences in human action recognition and the key frames were constructed based on skeleton histogram. Many other works tried to study the spatial-temporal descriptions from Kinect skeleton data, e.g., the angular representation [7] and skeletal shape trajectories [8]. As the skeleton information carries little details and is only suitable for human body tracking, it is difficult to detect and segment a small object, such as a human hand, which occupies a very small portion of the image with more complex articulations [9]. In practice, this type of work also suffers from contour distortions since little noise or slight variations in the contour would severely perturb the topology of its skeletal representation.

On the other hand, depth-based algorithms employ depth information for action recognition which shows its advantages in many situations. Joongrock and Sunjin [10] propose an adaptive local binary pattern from depth images for hand tracking. There are some researchers apply the dynamic time warping algorithm for hand gesture recognition with the extracted finger lets, stroke lets, or other characteristics from its depth information [11]. Their work shows that a concise and effective feature descriptor is critical for the recognition performance. Kviatkovsky [12] and Chang [13] suggest the use of covariance descriptors to encode the statistics of temporal shape and motion changes in a low dimensional space with an efficient incremental update mechanism. Zhang and Yang et al. [14] presented a low-cost descriptor via computing 3D histograms of textures from a sequence of depth maps. In their work, the depth sequences were first projected onto three orthogonal Cartesian plane to form three projected maps, then the sign-based, magnitude-based and center-based descriptor salient information were extracted, respectively. Similarly in Reza [15], the weighted depth motion map was proposed to extract the spatiotemporal information by an accumulated weighted absolute difference of consecutive frames and the histogram of gradient and local binary pattern were exploited for the feature descriptor.

Despite many algorithms and solutions in applying the Kinect for hand gesture and action recognition, it still is an open problem in practical applications considering the robustness, accuracy, and computational complexity. As the above-reviewed algorithms cannot process nonlinear and high dimensional data, some researchers tried to solve this problem via the recent advances in convolutional neural network [16–19]. The advantage of the deep neural network lies in that it is able to automatically extract hierarchical features to hold more abstract knowledge from video sequences and thus reduce the need for feature engineering. However, it requires a long time to train and a huge amount of labeled training data, which may not be available in some cases. For small human action recognition datasets, the deep learning methods may not provide satisfactory performance. The extracted features lack of specific physical meaning, thus it is difficult to analyze their characteristics.

It is known that the hand gesture delivers its meaning by the movement of a hand. Different hand gestures are mainly differentiated by the postures of the fingers. When the fingers display different postures, their contour shapes can be differentiated clearly. Therefore, many researchers focus on the extraction of various features [20–27]. Ren et al. [23] employed time series curves to characterize the Euclidean distance between the

hand contour and the palm center, where the starting point of the curve is not easy to track without any auxiliaries. Huang and Yang in [24] suggested a multi-scale descriptor including area of major zone, length of major segment, and central distance. In their method, it is important to choose a proper scale number and a starting point to align all points on the shape contour. Wang [25] constructed features with peak values and valley values from the trend of slope difference distribution of the contour points. The robustness and accuracy for extracting the peak and valley values are prone to be disturbed by various noise. Multiple types of features such as the rotation of joints and fingertip distances were proposed in [26], where the positions of 20 joint points were required to extract from the depth map according to the characteristics of the hand model. Obviously, their computational complexity is high and the extraction accuracy is not easy to control. In practice, it is desired that the feature descriptor possesses the properties of scale, translation, and rotation invariants [28,29]. For example, the contour of the hand region was extracted with Moore neighbor algorithm and the convex hull by Graham scan algorithm, and then the Hu moment invariants for hand gesture recognition were estimated in [28]. However, this algorithm is sensitive to noise and the computational load is heavy.

Basically, the major problem in the surveyed methods lies in that the features representing those gestures are not sufficient, which leads to poor performance and weak robustness. Therefore, this work aims at a comprehensive and discriminative feature for hand gesture recognition. Here, a new framework for hand gesture recognition is proposed by combing Fingertip Gradient orientation and Finger Fourier (FGFF) descriptor together with the modified Hu moments using the depth information collected by a Kinect sensor, where the former concentrates on the details of fingers and the latter encodes the structure of hand contour. According to the characteristics of hand depth image, two efficient procedures are suggested to segment the hand and extract fingers. Taking 10 types of hand gestures representing the digital numbers from zero to nine as an example, a weighted AdaBoost classifier is constructed based on the finger-earth mover's distance (FEMD) method and SVM model. Extensive experiments on a ten-gesture dataset collected in our lab were carried out to validate the proposed algorithm. Compared with three benchmark methods, our work achieves a better performance in terms of recognition accuracy, robustness and computational complexity (a 96.6% mean accuracy on the challenging 10-gesture dataset with average 0.05 s per frame).

The remainder paper is structured as follows. Two algorithms for the hand region segmentation and finger extraction are discussed in Section 2. Section 3 elaborates the FGFF descriptor and modified Hu moments. The weighted AdaBoost classifier for hand gesture recognition is introduced in Section 4. Section 5 presents some experimental results and analysis. Finally, this paper is concluded briefly in Section 6.

2. Hand Segmentation and Finger Extraction

This section firstly elaborates the technique for hand segmentation to obtain the interior points (mHand) and contour points (mContour) of a hand, then suggests two algorithms for extracting the palm center, fingers, and fingertips, denoted as mFingers and mFingertips, respectively. The flowchart of the hand segmentation and finger extraction from its depth image can be summarized in Figure 1.

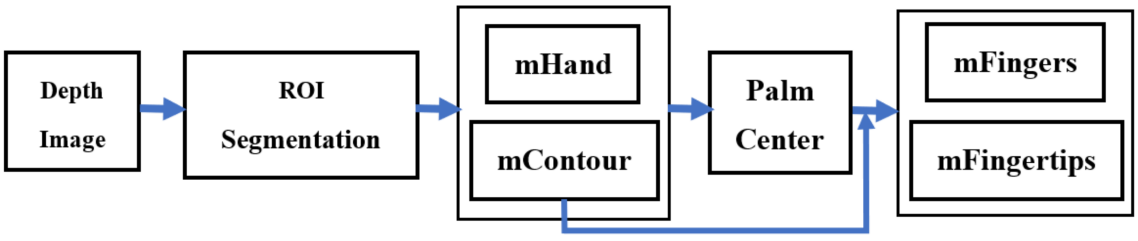


Figure 1. The flowchart of the hand segmentation and finger extraction.

2.1. Hand Region Segmentation

In order to effectively segment the hand region from its depth image, Ren et al. [23] suggested wearing a black belt to highlight the boundaries between the hand and the wrist. It is well known that the gray value of each pixel in the depth image represents the distance between the point and the sensor. The smaller the value is, the closer the distance becomes, and vice versa. Without loss of generality, it can be assumed that the hand is located in the front of the body when performing the gesture and there are no obstacles between the Kinect sensor and the performer. Therefore, the distance between the hand area and the sensor is the closest in this scenario. Considering that there is a certain range of hand size for general adults, this paper proposes a double-threshold-based region growing method to realize the segmentation of hand region and enable a natural mode of interaction without wearing any auxiliary object.

Firstly, the nearest point to the Kinect sensor denoted as M_{min}, is searched in the depth image. With M_{min} as a seeding point, the eight-neighborhood region growing method is iteratively performed. Here, we set three iterative conditions for the growing point as: (1) this point has not been grown before; (2) its difference with the depth value of the preceding point is less than the threshold value of Th1; (3) the difference with the average value of the point set that has been grown is less than Th2. When the iteration process ends, the proper hand region is obtained as

$$H = \left\{ M_{x,y} \middle| d_{M_{min}} < d_{M_{x,y}} < d_{M_{min}} + d_{th} \right\} \; s.t. \; |area(H) - A_0| < A_{th} \tag{1}$$

where $d_{M_{x,y}}$ and $d_{M_{min}}$, respectively, denote the depth value of the point $M_{x,y}$ and M_{min} in the depth image, while d_{th} represents the depth range of the detected hand region considering the general size of human hand. A_0 and A_{th} denote the area of average hand region and its range estimated from the training dataset who helps to remove the contamination regions or fake hand regions from the depth image. The parameter Th1 is used to keep consistency and smoothness in the ROI while Th2 decides whether oversegmentation is involved or not. Their values are set empirically and used to ensure the local and global consistency when growing the hand region. Our experimental results show that some holes may exist when the value of Th1 is set too high or the value of Th2 is too low. Oversegmentation will happen for a larger value of Th2, e.g., part of the wrist may be included as the hand region if a larger Th2 is used. Satisfactory results are obtained when Th1 ranges from 3 to 4 and Th2 ranges from 8 to 10. Figure 2 shows different effects of various threshold values in the hand region segmentation. Here, Th1 and Th2 are, respectively, set 5 and 7 in Figure 2b, while the empirical instructions are followed for the thresholds in Figure 2c,d. Obviously, better results are obtained in the latter two cases. This observation is critical where oversegmentation is needed to obtain part of the wrist as an anchor point to regularize the local features in the next section.

Figure 2. Different effects of the good and bad threshold values in segmentation: (**a**) gives the depth image while (**b**–**d**) show different segmentation results using different threshold values.

Compared with the traditional threshold-based segmentation, the advantage of this mechanism is that the boundary of the hand region is relatively smooth and there are fewer holes, as will be shown later. Therefore, it is easy for subsequent processing. A point is deemed as the interior point if its 3×3 neighborhood is also in the hand region, otherwise

it is the contour point. In this manner, the hand region H can be divided into interior point set and contour point set, respectively, denoted by $mHand$ and $mContour$.

2.2. Extraction of Palm Region

In general, the area of the palm as well as its roundness is larger than those of the fingers for any hand gesture. Based on this observation, the palm region and its center can be found mathematically as the largest inscribed circle in the hand region. Initially, the palm center and the maximum radius of inscribed circle are assumed as M_0 and R_0. The process for the solution can be summarized as Algorithm 1.

Algorithm 1 Calculating the center and maximum radius of the palm region

Input: Interior point set $mHand$ and contour point set $mContour$
Output: The radius R_0 and center of the palm center M_0
Begin
Step1: Set $R_0 = 0$ and $M_0 = []$ initially.
Step2: For one point in $mHand$, compute its distances from all the points in $mContour$.
Step3: Find the minimum distance value, update it as R_0 and the corresponding point as M_0 if it is larger than R_0.
Step4: Go to Step2 and repeat until all the points in $mHand$ are iterated.
Step5: Finally, the R_0 and M_0 are obtained as the radius of the inscribed circle of the palm region and its center.
End

2.3. Fingertip Extraction

When performing a gesture, different meanings are conveyed by different finger shapes and their relative positions, thus representing different digital gestures. It is obvious that the fingertip is the point farthest from the palm center in the contour point set. With this observation, the average distance between the palm center and those points in the contour point set is used to limit the scope of the fingertip and finger extraction to reduce the computational complexity, which means that those points within the average distance will be ignored. Assuming that all the fingers and fingertips are, respectively, denoted as $mFingers$ and $mFingertips$, the proposed solution is briefly summarized as Algorithm 2.

Algorithm 2 Extraction of fingertips and fingers

Input: The palm center M_0 and contour point set $mContour$
Output: The $mFingertips$ and $mFingers$
Begin
Step1: Initialize the sets of $candSet$, $mFingertips$ and $mFingers$;
Step2: Calculate the average distance R_{avg} from M_0 to all the points in $mContour$;
Step3: Add those points to $candSet$, if their distances from M_0 greater than or equal to R_{avg};
Step4: For each of the elements in $candSet$, compute its distance from M_0. Then find the one corresponding to the largest distance and move it to both $mFingertips$ and $mFingers$, and move all the rest points that are connected with it in the $candSet$ to $mFingers$;
Step5: Go to Step4 and repeat until $candSet$ is empty;
Step6: The fingertips and fingers are obtained in $mFingertips$ and $mFingers$.
End

Figure 3 shows the original depth image and its segmented hand region. The palm center and inscribed circle extracted with the first algorithm are referred to the red dot and circle in the right figure. For comparison, the central moment of the hand region is estimated and denoted as black cross. Obviously, the extracted palm center has a higher quality than the central moment. The blue circle shows the average distance between the palm center and the contour point set. From this figure, it is observed that the fingertips, fingers, and wrist of this hand gesture can be easily obtained with the second algorithm.

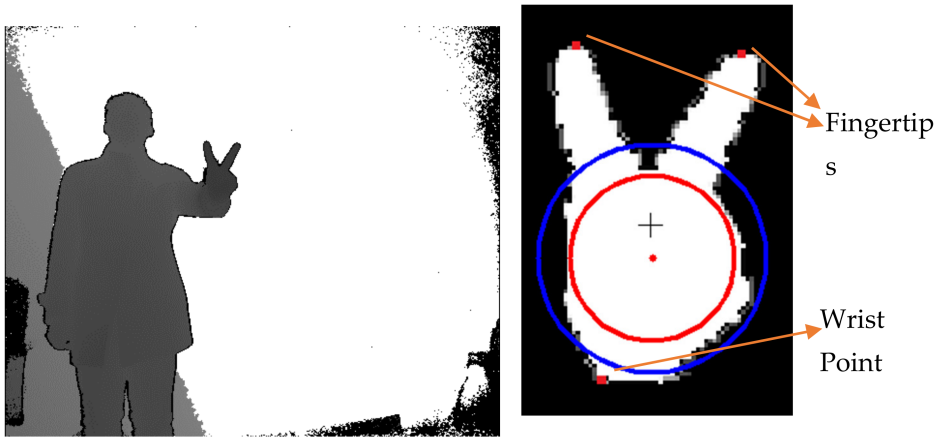


Figure 3. The depth image (**left**) and extracted elements (**right**): palm center with a higher quality denoted by RED point versus central moment of the hand by BLACK cross where the red and blue circles, respectively, represent inscribed and averaging circles.

On the whole, it can be seen that the hand area, contour, palm center, and fingertips are correctly extracted, and their boundaries are fairly smooth and graceful. It is worth to note that the wrist should be extracted as one finger with the above algorithm and its remote point as one fingertip since oversegmentation is involved, as discussed in Section 2.1. We will show how to identify it next.

The wrist can be determined considering that the wrist visually has the largest thickness compared with the fingers. Let the average circle of hand region be denoted as $mAvgcircles$. Mathematically, the remote point in the wrist can be defined as

$$M_{wr} = \{M | M \in mFingertips\} \quad s.t.\ \text{argmax}_{M_1, M_2 \in mFingers \cap mAvgcircles} dist(M_1, M_2) \tag{2}$$

where M_1 and M_2 are two elements in $mFingers$ associated with M and $dist(M_1, M_1)$ represents their Euclidean distance. Once the wrist point is found, it can be used as a benchmark for ordering the point sequences of the fingers since it is stationary for hand gestures. Finally, the fingertips and their related fingers can be easily identified.

3. FGFF Descriptor and Hu Moments

3.1. FGFF Descriptor

The FGFF descriptor consists of fingertip-emphasized and finger-emphasized components, i.e., the fingertip gradient orientations and finger Fourier descriptor. Let the palm center M_0 be $[x_p, y_p, d_p]$ with x_p and y_p as image coordinates and d_p its depth value. Similarly, the i-th fingertip in $mFingertips$ can be expressed as $M_{fi} = \left[x_{fi}, y_{fi}, d_{fi}\right]$. The gradient orientation can be constructed for each fingertip

$$FG_i = (x_i, y_i, d_i)^{\mathrm{T}} \tag{3}$$

where $x_i = x_{fi\_}x_p$, $y_i = y_{fi\_}y_p$, and $d_i = d_{fi\_}d_p$. The relative position is used here to eliminate the differences from different performers and avoid the distortions of some gestures. In Equation (3), the descriptor encodes the position and the depth value, as well as the orientation information for each fingertip, which is invariant to translation, rotation, and scale transformation with normalization.

On the other hand, the finger part associated with each fingertip is obviously connected, which can be represented as a point sequence. Let $\mathrm{s} = \{x(k),\ y(k) | k = 0,\ 1,\ 2, \cdots, L\}$ be the i-th finger. In complex space, it is formulated as 1-D problem

$$\mathrm{s}(k) = x(k) + jy(k)\ s.t.\ j^2 = -1 \tag{4}$$

With 1-D discrete Fourier transform, the spectrum in frequency domain is obtained. The left and right figures in Figure 4, respectively, give the point sequences of fingers and distribution of FF descriptor. It is observed that its energy is mainly concentrated with the lower frequencies and decreases rapidly with the increasing frequency. In this work, we find that more than 80 percent of energy is carried by the first seventeen magnitudes. Therefore, the Fourier descriptor denoted as FF_i, is assigned, considering a small range of lower frequencies. Finally, the FGFF descriptor for the finger can be constructed with FG_i and FF_i.

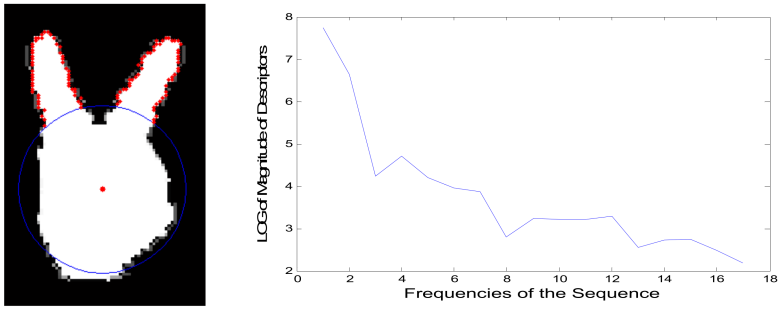


Figure 4. The fingers and distribution of FF descriptor vs. frequencies.

3.2. Modified Hu Moments

The Hu moment is an important feature used to describe image shape based on moment transform. The invariants of translation, rotation, and scale are preserved by the operations of centralization and normalization of the Hu moments for continuous functions. In gesture recognition, there are differences in action amplitude, hand size, and relative positions with respect to the sensor among different performers. Therefore, Hu moment invariants show unique advantages in such cases and can be used to encode the contour of a whole hand. However, different from the continuous function, the hand region is extracted as discrete data. We will first show that the Hu moments do not satisfy the scale invariant and then provide our modification in what follows.

Supposing the performer executes the same hand gesture at two different positions successively, the point coordinates on his hand in depth image will change from (x, y) to (x', y'). Let the scale factor be ρ, then $x' = \rho x$, $y' = \rho y$. According to the definition of moment, we have

$$x' - x'_c = \rho x - \frac{\sum_{x,y} \rho x f(x,y)}{\sum_{x,y} f(x,y)} = \rho(x - x_c) \tag{5}$$

and similarly

$$y' - y'_c = \rho(y - y_c) \tag{6}$$

Then we get

$$\mu'_{pq} = \sum_{x,y} (x' - x'_c)^p (y' - y'_c)^q f'(x' - y') = \rho^{p+q} \mu_{pq} \tag{7}$$

According to the normalization formula of centralized moment, we obtain

$$\varnothing'_{pq} = \frac{\mu'_{pq}}{\mu'^{r+1}_{00}} = \rho^{p+q} \varnothing_{pq} \tag{8}$$

where $r = \frac{p+q}{2}$. It can be seen from the above formula that the normalized central moment of discrete data is a function of the scale factor and the power of the moments. As a result,

the Hu moment is no longer scale invariant. To eliminate the scale factor, a group of new formula of Hu moments is suggested here:

$$a_1 = lg\left|\frac{h_1}{h_0^2}\right| a_2 = lg\left|\frac{h_2}{h_0 h_1}\right| a_3 = lg\left|\frac{h_3}{h_0 h_1}\right| a_4 = lg\left|\frac{h_4}{h_2 h_3}\right| a_5 = lg\left|\frac{h_5}{h_0 h_2}\right| a_6 = lg\left|\frac{h_6}{h_4}\right| \quad (9)$$

where $h_0, h_1, \cdots, h_6$ represent functions of $\varnothing'_{pq}$.

Compared with the forms of Hu moment with sixth power reported in the literature [28], the power degree in Formula (9) is much lower and hence has higher robustness to noise. In this manner, a 6-D Hu moment descriptor denoted as H_u can be formulated for each gesture from its depth image, which is also invariant to translation, rotation, and scale transformation.

4. Weighted AdaBoost Classifier

As shown in the above section, the number of extracted local finger features may be different when performing different types of hand gestures. Therefore, the finger-earth mover's distance (FEMD) method is used to estimate the similarity between two gesture images with their FGFF descriptors. For the modified Hu moment invariant features, this work employs a support vector machine to train and test those gesture images. Finally, the recognition results of these two methods are merged together with the weighted AdaBoost Classifier to perform the gesture recognition algorithm.

4.1. The Finger-Earth Mover's Distance

The finger-earth mover's distance method originated from the classical transportation problem and updated by Ren et al. [23] as FEMD. Let R_c refer to the FGFF descriptor extracted from an arbitrary hand gesture depth image with cth category in the training dataset. Let T represent the testing hand gesture, whose category can be determined by the category of the training sample with the highest similarity, which is defined as:

$$c^* = argmin_c \min\{FEMD(R_c, T) | R_c \text{ is a sample in category } c\} \quad (10)$$

where the parameter c ranges over all categories and c^* denotes the category corresponding the training sample with the highest similarity.

Suppose there are m fingers in the *mFingers* set for a training sample and its local feature descriptor can be represented as $R_c = \{(r_1, w_{r_1}), \cdots, (r_m, w_{r_m})\}$, where r_i and w_{r_i}, respectively, denotes the i-th finger and its weight factor. In the same manner, the feature descriptor for a testing hand gesture with n fingertips is denoted as $T = \{(t_1, w_{t_1}), \cdots, (t_n, w_{t_n})\}$.

Let $D = [d_{ij}]$ be the distance matrix between R_c and T, in which its element can be computed as $d_{ij} = r_i - t_{j2}$. Their FEMD distance is defined as the least work moving the earth piles from R_c to T plus the penalty on the empty hole that is not filled with earth

$$\begin{aligned} FEMD(R_c, T) &= \beta E_{move} + (1-\beta) E_{empty} \\ &= \frac{\beta \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij} + (1-\beta)\left|\sum_{i=1}^{m} w_{r_i} - \sum_{j=1}^{n} w_{t_j}\right|}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}} \end{aligned} \quad (11)$$

where the element f_{ij} in matrix F represents the workload of transporting r_i to t_j and $\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}$ as the normalization factor. The parameter β modulates the importance between E_{move} and E_{empty}. The sensitivity of this parameter to the recognition algorithm had been discussed in [23] and showed that the best results could be obtained when β falls in the range of 0.3 and 0.6. As the $FEMD$ depends on the matrix F, its objective function and constrains is given as what follows

$$F = argminWORK(R, T, F) = argmin \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij} s.t. \begin{cases} f_{ij} \geq 0 \, i = 1, \cdots, m; \; j = 1, \cdots, n \\ \sum_{i=1}^{m} w_{r_i} \leq w_{t_j} \\ \sum_{i=1}^{n} w_{t_j} \leq w_{r_i} \\ \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = min\left(\sum_{i=1}^{m} w_{r_i}, \sum_{i=1}^{n} w_{t_j} \right) \end{cases} \tag{12}$$

In this manner, the finger-earth mover's distances of the testing gesture with all the training samples are obtained. According to Formula (10), the category of the gesture is assigned to that of the training sample corresponding to the minimum finger-earth mover's distance. It should be noted that the proposed FGFF descriptor used here is completed as it encodes the relative position information of one fingertip, palm center, and wrist point, as well as the structure of the finger.

4.2. Support Vector Machine

Support vector machine (SVM) is a generalized linear classifier based on supervised learning for binary classification on the testing data. For multi-class problems, there are various deformations using SVM, e.g., one-to-one method, one-to-remainder method, and binary-tree method. Among those, the one-to-one method has the characteristics of high correct recognition rate, simplicity, and efficiency. Its basic idea is: Given an N-class classification problem, firstly training $\frac{N(N-1)}{2}$ support vector machines, and then the classification result of the testing data can be determined by voting principle with all the SVMs. For the recognition of 10 kinds of digital hand gestures in this paper, the one-to-one method is employed and hence a total of 45 support vector machines are trained, then the classification results are statistically analyzed and fused with those from Section 4.1. Finally, the category of the testing data could be determined.

4.3. Weighted AdaBoost Classifier (WAC)

The FGFF descriptor and modified Hu moment invariants are deemed as local detailed features and global structural features extracted from the hand gesture depth images. The finger-earth mover's distance method and support vector machine model are used as the base classifiers to recognize the hand gesture with those features. During training, the performance of each of the classifiers can be obtained with the training set. Given one test sample, say T_i, its category can be decided with the following weighted AdaBoost classifier

$$Label(T_i) = \sum \alpha_j G_j(T_i) \tag{13}$$

where $G_j \in$ {FEMD method, SVM model} is the j-th basis classifier with the weighted factor

$$\alpha_j = \frac{1}{2} log \frac{P\big(G_j(R_i) = Label(R_i)\big)}{1 - P\big(G_j(R_i) = Label(R_i)\big)} \tag{14}$$

As can be seen from the above equation, the weighted factor α_j is an increasing function of the recognition accuracy of a base classifier. When the recognition accuracy is more than 50%, we have $\alpha_j > 0$. With the increasing accuracy, its relative role in the AdaBoost classifier becomes more and more important. In this manner, we highlight the excellent classifier in our algorithm. Our experiments also show that the accuracy and stability will be strengthened by constructing the combination model with such an addition mechanism.

5. Experiments and Analysis

This section presents our experimental results on the ten-gesture dataset collected in our lab to validate the proposed hand gesture recognition algorithm. Some details and

insights in the algorithm are discussed together with comparison with three benchmark methods to demonstrate the improvement of our algorithm.

5.1. Experimental Dataset

Figure 5 shows the 10 kinds of hand gestures to be recognized in this work, from left to right, respectively, representing the digital numbers from zero to nine. To collect their depth images, ten students are invited to perform those gestures before the Kinect sensor at about three different positions, say 80 cm, 120 cm and 150 cm considering the effective range of the sensor. Their hands are placed in the front of their body for the ease of hand region segmentation. Each kind of hand gestures is repeated 20 times by one person. In this manner, the experimental dataset contains a total of 2000 samples.

Figure 5. The predefined hand gestures for numbers from zero to nine.

5.2. Hand Region Segmentation and Feature Extraction

As the depth value instead of color information is used in the hand region segmentation, it is comparably easy to distinguish the hand gesture from its environment in the depth image. Figure 6 shows the depth images of one group of gestures followed by their hand regions segmented by the method suggested in Section 2.1. It can be observed from these figures that the interior of the regions is relatively uniform, and its boundary is very smooth. Only a very small empty hole is found in the region of gesture seven, and few noisy branches are kept. Therefore, the suggested hand segmentation algorithm works well.

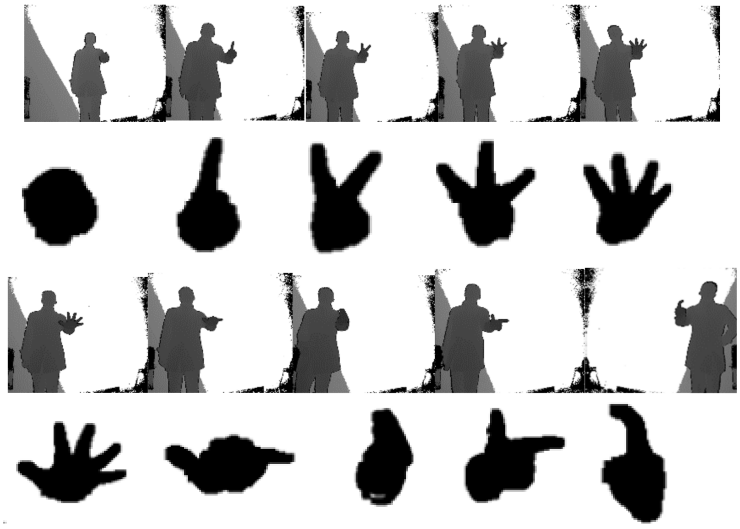


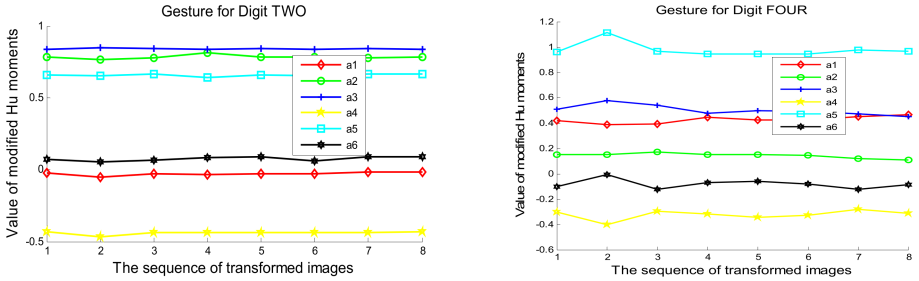
Figure 6. One group of depth images and segmented hand region: the depth images given in the first and third rows with corresponding hand in the second and fourth rows.

5.3. Invariants of Modified Hu Moments

This section validates the rotation and scale invariants of the modified Hu moments as its translation invariant is apparent. The first column in Figure 7 shows the depth images of two different gestures, while the second to fourth columns present their scales by 0.5, 0.75, and 1.5, and the last four columns demonstrate their rotations of 30° and 15° in clockwise as well as anti-clockwise motions. Figure 8 gives the estimations of six elements in the modified Hu moments with Formula (9), where the left and right figures, respectively, mark the results from the first and second gestures. It is observed that these estimations

are fairly steady against those transformations, with low standard deviations of 0.0069 and 0.0278. This validates the rotation and scale invariants of the modified Hu moments both qualitatively and quantitatively. To demonstrate their discrimination ability, Table 1 presents the Euclidean distances of the modified Hu moments from those pairwise images whose order numbers are marked from #1 and #2 to #16. Strong discrimination ability can be observed from this table since the first eight images and the remainders belong to two different categories.

Figure 7. The gestures with their scale and rotation transformations.



(**a**) (**b**)

Figure 8. Distributions of the Hu moments via different transformations, in which the elements of a1–a6 are computed by the Formula (9): (**a**,**b**) respectively present the values of Hu moments for gestures of Two and Four.

Table 1. Distance of modified Hu moments from pairwise images.

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15
#2	0.025														
#3	0.005	0.021													
#4	0.021	0.034	0.023												
#5	0.011	0.028	0.012	0.020											
#6	0.007	0.023	0.008	0.019	0.016										
#7	0.010	0.029	0.012	0.024	0.006	0.016									
#8	0.010	0.031	0.014	0.022	0.007	0.016	0.004								
#9	0.459	0.465	0.460	0.477	0.464	0.462	0.458	0.458							
#10	0.460	0.465	0.460	0.479	0.463	0.464	0.457	0.458	0.110						
#11	0.443	0.449	0.444	0.461	0.448	0.446	0.442	0.442	0.027	0.109					
#12	0.465	0.472	0.466	0.483	0.469	0.468	0.463	0.464	0.028	0.114	0.053				
#13	0.454	0.460	0.455	0.472	0.458	0.457	0.452	0.453	0.031	0.102	0.050	0.021			
#14	0.458	0.464	0.459	0.476	0.463	0.461	0.457	0.457	0.019	0.108	0.041	0.017	0.014		
#15	0.490	0.496	0.491	0.508	0.495	0.493	0.489	0.489	0.033	0.126	0.053	0.040	0.053	0.041	
#16	0.494	0.500	0.495	0.512	0.499	0.497	0.492	0.493	0.043	0.123	0.069	0.030	0.044	0.038	0.028

5.4. Discrimination of Confused Gestures

Generally speaking, the smaller the distance between the gesture performer and the Kinect sensor, the bigger the image size of the hand region and the larger the distance value extracted from the fingertip to the palm center, and vice versa. In order to overcome this influence, the FGFF descriptor for each hand gesture is normalized with its M_{min}. To validate the discriminative ability of the weighted AdaBoost classifier for confused gestures, the feature descriptors for gesture 1, gesture 7, and gesture 9 are firstly studied. Figure 9 shows three different samples for each type of gestures and the results of hand region segmentation. With the proposed fingertip extraction procedure, just one fingertip is extracted from each of those gesture images, corresponding to one finger clustering information. The thickness of gesture 7 is obviously larger than those of the other two. Therefore, its weight factor of is bigger and the finger-earth mover's distance method exhibits a strong distinguishing ability for gesture 7 while it is weak for the remaining two gestures since their weight factors are close to each other. In this case, the SVM model with modified Hu moments shows its advantage. For better visualization, Figure 10 shows the Euclidean distance between the Hu moment features of the nine gesture samples. It can be seen from this figure that the interdistances between different gestures are larger compared with those intradistances. This demonstrates strong recognition ability of the support vector machine and hence the weighted AdaBoost classifier.

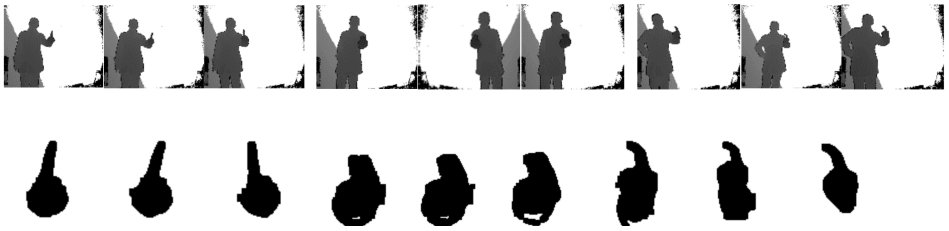


Figure 9. Nine samples from three different types of gestures and extracted hand regions.

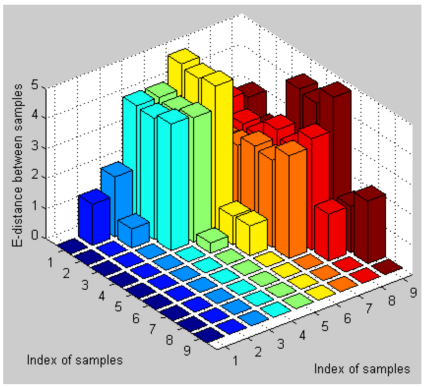


Figure 10. Larger interdistances and smaller intradistances among hand gestures.

5.5. Gesture Recognition and Analysis

We firstly test the effect of different volumes of training dataset on the performance of the proposed algorithm, where its training samples were randomly chosen to keep consistent distribution, varying from 25% to 75% of the whole dataset. The remainders are used as the testing samples. Table 2 gives the accuracy of hand gesture recognition versus volumes of training dataset. From this table, we can see that the accuracy is increased with the increasing volume and an acceptable balance is obtained when a half dataset is used for training the proposed classifier.

Table 2. The recognition accuracy vs. different volumes of training dataset (%).

Volumes	0	1	2	3	4	5	6	7	8	9	AVG acc.
25%	100	89.3	91.3	92.6	92.0	94.0	90.0	94.0	91.3	90.6	92.5
40%	100	93.3	94.1	95.8	96.7	96.7	94.1	96.7	95.8	93.3	95.6
50%	100	96.7	96.0	97.0	98.0	98.0	97.3	96.0	96.0	97.0	97.2
75%	100	96.7	97.3	98.6	98.0	98.0	97.3	96.7	97.0	98.0	97.7

Different distances between the performer and the Kinect sensor will affect the size of hand region and produce a scale in the depth image. Subsequently, the accuracy of image processing and gesture recognition are affected. In order to validate the robustness of the weighted AdaBoost classifier against different scales, we test it under different distances. Here, thirty samples, respectively, from 80 cm distance and 120 cm distance together with forty samples from 150 cm distance were randomly chosen for testing and the rest for training. Table 3 shows the recognition results of those gestures where the last row gives average accuracy.

Table 3. The recognition accuracy vs. different distances from the sensor (%).

Distances	0	1	2	3	4	5	6	7	8	9	AVG acc.
80 cm	100	96.7	96.7	96.7	96.7	96.7	93.3	93.3	96.7	96.7	96.3
120 cm	100	100	96.7	96.7	100	100	96.7	96.7	96.7	96.7	98.0
150 cm	100	95	95	97.5	95	97.5	92.5	97.5	95	92.5	95.7
AVG.acc	100	97	96	97	97	98	94	96	96	95	–

As can be seen from Table 3, the average recognition accuracy for the hand gestures is above 94%. Among them, the accuracy for gesture zero is always the highest, which can be correctly recognized in all tests, since this is the simplest gesture and the performer can make this action more easily and accurately. The procedure for its image segmentation and feature extraction is also the most stable and reliable. The recognition accuracy for the other hand gesture is slightly lower, but more than 92%. Better performance is observed when the distance between the gesture executor and the Kinect is positioned at about 120 cm. With the increasing distance, the recognition accuracy decreases slightly. This is because the hand region in the depth image becomes smaller, which brings the difficulty of image processing and hand region segmentation.

Figure 11 shows the confusion matrix of recognition for these ten hand gestures to go into some details. As can be seen from this figure, the hand gestures for six, seven, and nine are easy to be confused. The recognition accuracy for gesture six is the lowest, which is due to the fact that the little finger in this gesture is easily overlapped in some viewpoints. This may lead to confusion with gesture one or gesture seven since their thicknesses are somewhat near to each other. For gesture seven, its degree of aggregation and curvature of the fingers is different from one performer to another, which makes the feature extraction of fingertips a bit more difficult. In this case, the weighted AdaBoost classifier shows its advantages against any single classifier. The degree of curvature in gesture nine is also performer-dependent and prone to be affected by different viewpoints, which may lead to confused recognition.

5.6. Comparison with Benchmark Algorithms

For comparison of recognition performance, we implement the proposed algorithm and three benchmark methods, including Ren [23], Huang [24], and Pu Xingcheng [28] as they follow a similar mechanism. To be a fair game, all the experiments are carried out on the same dataset collected in our lab. The finger-earth mover's distance method with time-series curve representation of the hand contour was suggested in [23], while similarity measures through the Mahalanobis distance among the Hu moment features

were used in [28]. In [24], different scales of circle regions centered at each of the contour points were employed to extract the area, major segment, and distance information as characteristics of the hand gesture. Basically, it is a multi-resolution analysis along the hand contour. Different ranges of finger motion as well as the noise on the contour have a considerably negative effect on the algorithm. For comparison, the FGFF descriptor and modified Hu moment in the proposed algorithm are independently tested, followed by their combinations with the weighted AdaBoost classifier, as given in the fifth, sixth, and eighth rows of Table 4. It is seen from Table 4 that our algorithm gives the highest accuracy with lower standard deviation since both finger-related and contour-related information is employed, followed by Huang [24] with 96.1% mean accuracy and 1.8 standard deviation. Since detailed shapes of the fingers are ignored in the algorithms from [23,28], their recognition accuracy is lower with an average accuracy of 95% and 95.5%, respectively. On the whole, the proposed method overcomes the shortcomings of the Benchmark methods and can obtain higher and more stable recognition accuracy.

——	0	1	2	3	4	5	6	7	8	9
0	100	0	0	0	0	0	0	0	0	0
1	0	97	0	0	0	0	2	1	0	2
2	0	0	96	0	0	0	0	0	3	0
3	0	0	0	97	2	0	0	0	0	0
4	0	0	0	2	97	2	0	0	0	0
5	0	0	0	0	1	98	0	0	0	0
6	0	1	0	0	0	0	94	2	1	2
7	0	0	1	0	0	0	2	96	0	1
8	0	0	3	1	0	0	1	0	96	0
9	0	2	0	0	0	0	1	1	0	95

Figure 11. Digital gesture recognition confusion matrix.

Table 4. Comparison of the proposed algorithm with Benchmark methods on our dataset (%).

Different Gestures	0	1	2	3	4	5	6	7	8	9	AVG acc.
Ren [23]	100	96	93	95	94	96	93	95	93	95	95.0
Huang [24]	100	96	96	95	96	98	94	96	96	94	96.1
Pu [28]	99	95	96	96	95	97	94	94	95	94	95.5
FGFF+FEMD	100	96	94	96	95	96	94	94	95	94	95.4
Hu+SVM	98	96	96	96	95	97	93	95	96	95	95.7
Proposed Algorithm	100	97	96	97	97	98	94	96	96	95	96.6

6. Conclusions

We have talked about a new hand gesture recognition algorithm based on the Kinect sensor, taking the recognition of ten digital gestures from zero to nine as an example. The region growing method with double thresholds is employed to segment the hand region from its depth image where the influence of different thresholds is discussed and oversegmentation is suggested. Then, fingertip-emphasized features including palm center and fingertips together with their orientations are estimated, followed by the finger-emphasized Fourier descriptor to construct the FGFF descriptors. The modified Hu moment invariants with much lower exponents are discussed to encode the contour structure in the hand region. Finally, a weighted AdaBoost classifier is constructed based on FEMD and SVM model to realize the hand gesture recognition. The characteristics and applicability of the classifier are analyzed and compared with the three Benchmark methods reported in the literature. The results show that the proposed algorithm outperforms those algorithms with better robustness and higher recognition accuracy. Future work includes exploring more representative features and further improving the robustness of the algorithm, developing some interesting HCI applications and deploying it on our mobile robot to perform some routine housework under the instructions of hand gestures.

Author Contributions: Investigation, B.Z. and J.L.; Methodology, B.Z. and J.L.; Software, B.Z. and B.W.; Validation, Y.Z.; Writing—original draft, B.Z.; Writing—review and editing, Y.Z. and B.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by NSFC (Grant No. 61973152 and 61903182), Natural Science Foundation of Jiangsu Province (Grant No. BK20171481 and BK20181414), and Global Challenges Research Fund (GCRF), UK (P202PF11).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Assaleh, K.; Shanableh, T.; Hajjaj, H. Recognition of handwritten Arabic alphabet via hand motion tracking. *J. Frankl. Inst.* **2009**, *346*, 175–189. [CrossRef]
2. Tubaiz, N.; Shanableh, T.; Assaleh, K. Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode. *IEEE Trans. Hum.-Mach.* **2015**, *45*, 526–533. [CrossRef]
3. Cornacchia, M.; Ozcan, K.; Zheng, Y.; Velipasalar, S. A Survey on Activity Detection and Classification using Wearable Sensors. *IEEE Sens. J.* **2016**, *17*, 386–403. [CrossRef]
4. Han, F.; Reily, B.; Hoff, W.; Zhang, H. Space-time representation of people based on 3D skeletal data: A review. *Comput. Vis. Image Underst.* **2017**, *158*, 85–105. [CrossRef]
5. Wang, L.; Huynh, D.Q.; Koniusz, P. A Comparative Review of Recent Kinect-based Action Recognition Algorithms. *IEEE Trans. Image Process.* **2020**, *29*, 15–28. [CrossRef]
6. Thanh, T.T.; Chen, F.; Kotani, K.; Le, B. Extraction of Discriminative Patterns from Skeleton Sequences for Accurate Action Recognition. *Fundam. Inform.* **2014**, *130*, 247–261. [CrossRef]
7. Zhu, H.M.; Pun, C.M. Human action recognition with skeletal information from depth camera. In Proceedings of the IEEE International Conference on Information and Automation, Hailar, China, 28–30 July 2014; pp. 1082–1085.
8. Ben, A.B.; Su, J.; Srivastava, A. Action Recognition Using Rate-Invariant Analysis, of Skeletal Shape Trajectories. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 1–13.
9. Liu, X.; Shi, H.; Hong, X.; Chen, H.; Tao, D.; Zhao, G. 3D Skeletal Gesture Recognition via Hidden States Exploration. *IEEE Trans. Image Process.* **2020**, *29*, 4583–4597. [CrossRef] [PubMed]
10. Kim, J.; Yu, S.; Kim, D.; Toh, K.A.; Lee, S. An adaptive local binary pattern for 3D hand tracking. *Pattern Recognit.* **2017**, *61*, 139–152. [CrossRef]
11. Tang, J.; Cheng, H.; Zhao, Y.; Guo, H. Structured Dynamic Time Warping for Continuous Hand Trajectory Gesture Recognition. *Pattern Recognit.* **2018**, *80*, 21–31. [CrossRef]
12. Kviatkovsky, I.; Rivlin, E.; Shimshoni, I. Online action recognition using covariance of shape and motion. *Comput. Vis. Image Underst.* **2014**, *129*, 15–26. [CrossRef]
13. Tang, C.; Li, W.; Wang, P.; Wang, L. Online human action recognition based on incremental learning of weighted covariance descriptors. *Inf. Sci.* **2018**, *467*, 219–237. [CrossRef]
14. Zhang, B.; Yang, Y.; Chen, C.; Yang, L.; Han, J.; Shao, L. Action Recognition Using 3D Histograms of Texture and A Multi-class Boosting Classifier. *IEEE Trans. Image Process.* **2017**, *26*, 4648–4659. [CrossRef] [PubMed]
15. Azad, R.; Asadi-Aghbolaghi, M.; Kasaei, S.; Escalera, S. Dynamic 3D Hand Gesture Recognition by Learning Weighted Depth Motion Maps. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *26*, 1729–1737. [CrossRef]
16. Cardenas, E.E.; Chavez, G.C. Multimodal Hand Gesture Recognition Combining Temporal and Pose Information Based on CNN Descriptors and Histogram of Cumulative Magnitudes. *J. Vis. Commun. Image Represent.* **2020**, *71*, 102772. [CrossRef]
17. Dong, J.; Xia, Z.; Yan, W.; Zhao, Q. Dynamic gesture recognition by directional pulse coupled neural networks for human-robot interaction in real time. *J. Vis. Commun. Image Represent.* **2019**, *63*, 102583. [CrossRef]
18. Zhang, J.; Li, Y.; Xiao, W.; Zhang, Z. Non-iterative and Fast Deep Learning: Multilayer Extreme Learning Machines. *J. Frankl. Inst.* **2020**, *357*, 8925–8955. [CrossRef]
19. Santos CSamatelo, J.; Vassallo, R. Dynamic gesture recognition by using CNNs and starRGB: A temporal information condensation. *Neurocomputing* **2020**, *400*, 238–254. [CrossRef]
20. Wang, C.; Liu, Z.; Chan, S.C. Superpixel-Based Hand Gesture Recognition with Kinect Depth Camera. *IEEE Trans. Multimed.* **2014**, *17*, 29–39. [CrossRef]
21. He, Y.; Li, G.; Liao, Y.; Sun, Y.; Kong, J.; Jiang, G.; Jiang, D.; Xu, S.; Liu, H. Gesture recognition based on an improved local sparse representation classification algorithm. *Clust. Comput.* **2019**, *22*, 10935–10946. [CrossRef]

22. Lee, D.L.; You, W.S. Recognition of complex static hand gestures by using the wristband-based contour features. *IET Image Process.* **2018**, *12*, 80–87. [CrossRef]
23. Ren, Z.; Yuan, J.; Meng, J. Robust Part-Based Hand Gesture Recognition Using Kinect Sensor. *IEEE Trans. Multimed.* **2013**, *15*, 1110–1120. [CrossRef]
24. Huang, Y.; Yang, J. A multi-scale descriptor for real time RGB-D hand gesture recognition—ScienceDirect. *Pattern Recognit. Lett.* **2021**, *144*, 97–104. [CrossRef]
25. Wang, Z.Z. Gesture recognition by model matching of slope difference distribution features. *Measurement* **2021**, *181*, 109590. [CrossRef]
26. Miao, Y.; Li, J.; Liu, J.; Chen, J.; Sun, S. Gesture recognition based on joint rotation feature and fingertip distance feature. *J. Comput. Sci.* **2020**, *43*, 80–94. (In Chinese)
27. Sun, Y.; Weng, Y.; Luo, B.; Li, G.; Tao, B.; Jiang, D.; Chen, D. Gesture Recognition Algorithm based on Multi-scale Feature Fusion in RGB-D Images. *IET Image Process.* **2020**, *14*, 3662–3668. [CrossRef]
28. Pu, X.; Tao, W.; Yi, Z. Kinect gesture recognition algorithm based on improved Hu moment. *Comput. Eng.* **2016**, *42*, 165–172. (In Chinese)
29. Dhiman, C.; Vishwakarma, D.K. A Robust Framework for Abnormal Human Action Recognition using R-Transform and Zernike Moments in Depth Videos. *IEEE Sens. J.* **2019**, *19*, 5195–5203. [CrossRef]