applied sciences

MDPI

Article

3D Skeletal Joints-Based Hand Gesture Spotting and Classification

Ngoc-Hoang Nguyen, Tran-Dac-Thinh Phan, Soo-Hyung Kim, Hyung-Jeong Yang and Guee-Sang Lee *

Department of Artificial Intelligence Convergence, Chonnam National University, 77 Yongbong-ro, Gwangju 500-757, Korea; hoangnguyenkcv@gmail.com (N.-H.N.); phantrandacthinh2382@gmail.com (T.-D.-T.P.); shkim@jnu.ac.kr (S.-H.K.); hjyang@jnu.ac.kr (H.-J.Y.)

* Correspondence: gslee@jnu.ac.kr

Abstract: This paper presents a novel approach to continuous dynamic hand gesture recognition. Our approach contains two main modules: gesture spotting and gesture classification. Firstly, the gesture spotting module pre-segments the video sequence with continuous gestures into isolated gestures. Secondly, the gesture classification module identifies the segmented gestures. In the gesture spotting module, the motion of the hand palm and fingers are fed into the Bidirectional Long Short-Term Memory (Bi-LSTM) network for gesture spotting. In the gesture classification module, three residual 3D Convolution Neural Networks based on ResNet architectures (3D_ResNet) and one Long Short-Term Memory (LSTM) network are combined to efficiently utilize the multiple data channels such as RGB, Optical Flow, Depth, and 3D positions of key joints. The promising performance of our approach is obtained through experiments conducted on three public datasets—Chalearn LAP ConGD dataset, 20BN-Jester, and NVIDIA Dynamic Hand gesture Dataset. Our approach outperforms the state-of-the-art methods on the Chalearn LAP ConGD dataset.

Keywords: continuous hand gesture recognition; gesture spotting; gesture classification; multi-modal features; 3D skeletal; CNN

check for updates

Citation: Nguyen, N.-H.; Phan, T.-D.-T.; Kim, S.-H.; Yang, H.-J.; Lee, G.-S. 3D Skeletal Joints-Based Hand Gesture Spotting and Classification. *Appl. Sci.* **2021**, *11*, 4689. https://doi.org/10.3390/app11104689

Academic Editor: Hyo-Jong Lee

Received: 14 April 2021
Accepted: 18 May 2021
Published: 20 May 2021

1. Introduction

Nowadays, the role of dynamic hand gesture recognition has become crucial in vision-based applications for human-computer interaction, telecommunications, and robotics, due to its convenience and genuineness. There are many successful approaches to isolated hand gesture recognition with the recent development of neural networks, but in real-world systems, the continuous dynamic hand gesture recognition remains a challenge due to the diversity and complexity of the sequence of gestures.

Initially, most continuous hand gesture recognition approaches were based on traditional methods such as Conditional Random Fields (CRF) [1], Hidden Markov Model (HMM), Dynamic Time Warping (DTW), and Bézier curve [2]. Recently, deep learning methods based on convolution neural networks (CNN) and recurrent neural networks (RNN) [3–7] have gained popularity.

The majority of continuous dynamic hand-gesture recognition methods [3–6] include two separate procedures: gesture spotting and gesture classification. They utilized the spatial and temporal features to improve the performance mainly in gesture classification.

However, there are limitations in the performance of gesture spotting due to its inherent variability in the duration of the gesture. In existing methods, gestures are usually spotted by detecting transitional frames between two gestures. Recently, an approach [7] simultaneously performed the task of gesture spotting and gestures classification, but it turned out to be suitable only for feebly segmented videos.

Most of the recent researches [8–11] intently focus on improving the performance of the gesture classification phase, while the gesture spotting phase is often neglected on the assumption that the isolated pre-segmented gesture sequences are available for input to the gesture classification.

However, in real-world systems, spotting of the gesture segmentation plays a crucial role in the whole process of gesture recognition, hence, it greatly affects the final recognition performance. In paper [3], they segmented the videos into sets of images and used them to predict the fusion score, which means they simultaneously did the gesture spotting and gesture classification. The authors in [5] utilized the Connectionist temporal classification to detect the nucleus of the gesture and the no-gesture class to assist the gesture classification without requiring explicit pre-segmentation. In [4,6], the continuous gestures are often spotted into isolation based on the assumption that hands will always be put down at the end of each gesture which turned out to be inconvenient. It does not work well for all situations, such as in "zoom in", "zoom out" gestures, i.e., when only the fingers move while the hand stands still.

In this paper, we propose a spotting-classification algorithm for continuous dynamic hand gestures which we separate the two tasks like [4,6] but we avoid the existing problems of those methods. In the spotting module, as shown in Figure 1, the continuous gestures from the unsegmented and unbounded input stream are firstly segmented into individually isolated gestures based on 3D key joints extracted from each frame by 3D human pose and hand pose extraction algorithm. The time series of 3D key poses are fed into the Bidirectional Long Short-Term Memory (Bi-LSTM) network with connectionist temporal classification (CTC) [12] for gesture spotting.

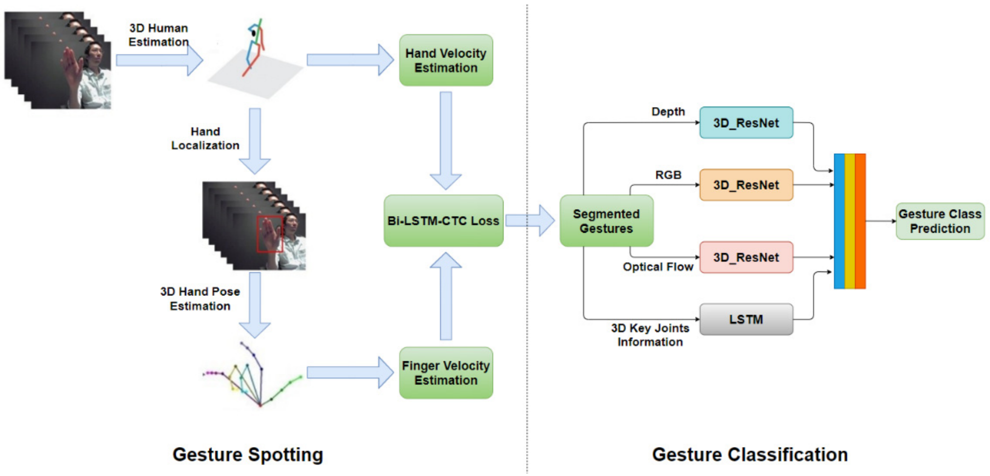


Figure 1. Gesture Spotting-Classification Module.

The isolated gestures segmented using the gesture spotting module are classified in the gesture classification module with a multi-modal M-3D network. As indicated in Figure 1, in the gesture classification module, the M-3D network is built by combining multi-modal data inputs which comprise RGB, Optical Flow, Depth, and 3D pose information data channels. Three residual 3D Convolution Neural Network based on ResNet architectures (3D_ResNet) [13] stream networks of RGB, Optical Flow and Depth channel along with an LSTM network of 3D pose channel are effectively combined using a fusion layer for gesture classification.

The preliminary version of this paper has appeared in [14]. In this paper, depth information has been considered together with 3D skeleton joints information with extensive experiments, resulting in upgraded performance.

The remainder of this paper is organized as follows. In Section 2, we review the related works. The proposed continuous dynamic hand gesture recognition algorithm is intently discussed in Section 3. In Section 4, the experiments with proposed algorithms conducted on three published datasets—Chalearn LAP ConGD dataset, 20BN-Jester, and NVIDIA Dynamic Hand Gesture Dataset are presented with discussions. Finally, we conclude the paper in Section 5.

2. Related Works

In general, the continuous dynamic gesture recognition task is more complicated than the isolated gesture recognition task, where the sequence of gestures from an unsegmented and unbounded input stream are separated into complete individual gestures, called gesture spotting or gesture segmentation before classification. The majority of recent researchers solve the continuous dynamic gesture recognition task using two separate processes—gesture spotting and gesture recognition [1,4–6].

In the early years, the approaches for gesture spotting were commonly based on traditional machine learning techniques for the time series problems such as Conditional Random Fields (CRF) [1], Hidden Markov Model (HMM) [2], and Dynamic Time Warping (DTW) [3]. Yang et al. [1] presented a CRF threshold model that recognized gestures based on system vocabulary for labeling sequence data. Similar to the method introduced by Yang, Lee et al. [2] proposed the HMM-based method, which recognized gestures by the likelihood threshold estimation of the input pattern. Celebi et al. [3] proposed a template matching algorithm, i.e., the weighted DTW method, which used the time sequence of the weighted joint positions obtained from a Kinect sensor to compute the similarity of the two sequences. Krishnan et al. [15] presented a method using the Adaptive Boosting algorithm based on the threshold model for gesture spotting using continuous accelerometer data and the HMM model for gesture classification. The limitations of these methods are the parameter of the model has been decided through experience and the algorithm is sensitive to noise. In the recent past, with the success of deep learning applications in computer vision, deep learning approaches have been utilized for hand gesture recognition to achieve impressive performance compared to traditional methods.

The majority of the methods using recurrent neural networks (RNN) [16–18] or CNN [8,10,19–21] focus only on isolated gesture recognition, which ignores the gesture spotting phase. After the dataset for continuous gesture spotting-Chalearn LAP ConGD dataset was provided, a number of methods have been proposed to solve both phases of gesture spotting and gestures recognition [3,4,6]. Naguri et al. [6] applied 3D motion data input from infrared sensors into an algorithm based on CNN and LSTM to distinguish gestures. In this method, they segmented gestures by detecting transition frames between two isolated gestures. Similarly, Wang et al. [3] utilized transition frame detection using two streams CNN to spot gestures. In another approach proposed by Chai et al. [4], continuous gestures were spotted based on the hand position detected by Faster R-CNN and isolated gesture was classified by two parallel recurrent neural network SRNN with RGB_D data input. The multi-modal network, which combines a Gaussian-Bernoulli Deep Belief Network (DBN) with skeleton data input and a 3DCNN model with RGB_D data, was effectively utilized for gesture classification by Di et al. [7]. Tran et al. [22] presented CNN based method using a Kinect Camera for spotting and classification of hand gestures. However, the gesture spotting was done manually from a pre-specified hand shape or finger-tip pattern. And classification of hand gestures used only fundamental 3DCNN networks without employing the LSTM network. The system is based on the Kinect system and the comparison using a commonly used public dataset is almost impossible.

Recently, Molchanov et al. [5] proposed a method for joint gesture spotting and gesture recognition using a zero or negative lag procedure through a recurrent three-dimensional convolution neural network (R3DCNN). This network is highly effective in recognizing weakly segmented gestures from multi-modal data.

In this paper, we propose an effective algorithm for both spotting and classification tasks by utilizing extracted 3D human and hand skeletal features.

3. Proposed Algorithm

In this section, we intently focus on the proposed method using two main modules: gesture spotting and gesture classification. For entire frames of continuous gesture video, the speed of hand and finger estimated from the extracted 3D human pose and 3D hand pose are utilized to segment continuous gesture. The isolated gesture segmented by gesture spotting module is classified using the proposed M-3D network with RGB, Optical flow, Depth, and 3D key joints information.

3.1. Gesture Spotting

The gesture spotting module is shown on the left of Figure 1. All frames of continuous gesture sequence are utilized to extract 3D human pose using the algorithm proposed in [23]. Through RGB hand ROI localized from 3D hand palm position $J_h(x,y,z)$ when the hand palm stands still and over spine base joint, we use a 3D hand pose estimation algorithm to effectively extract the 3D position of the finger joints. From the extracted 3D human pose, the hand speed v_{hand} is estimated using the movement distance of the hand joint between two consecutive frames.

- **3D human pose extraction:** From each RGB frame, we obtain a 3D human pose by using one of the state-of-the-art methods for 2D/3D human pose estimation in the wild-pose-hgreg-3d network. This network has been proposed by Zhou et al. [23] which provides the pre-trained model on the Human3.6M dataset [24]. This is the largest dataset providing both 2D, 3D annotations of human poses in 3.6 million RGB images. This network is a fast, simple, and accurate neural network based on 3D geometric constraints for weakly-supervised learning of 3D pose with 2D joint annotations extracted through the state-of-the-art of 2D pose estimation method, i.e., stacked hourglass network of Newell et al. [25]. In our proposed approach, we use this 3D human pose estimation network to extract the exact 3D hand joint information, which is effectively utilized for both gesture spotting and gesture recognition task.

Let $J_h(x_{hk,}\ y_{hk,}\ z_{hk})$, $J_h(x_{hk-1},\ y_{hk-1},\ z_{hk-1})$ be the 3D position of the hand joint at the kth frame, and $(k-1)$th frame, respectively. The hand speed is estimated as

$$v_{hand} = \alpha\cdot\sqrt{(x_{hk}-x_{hk-1})^2+(y_{hk}-y_{hk-1})^2+(z_{hk}-z_{hk-1})^2} \tag{1}$$

where α is the frame rate.

The finger speed is estimated by the change in distance between the 3D position of fingertips of the thumb and the index finger in sequence frames. Let denote $J_{ft}(x_{ftk},\ y_{ftk,}\ z_{ftk})$, $J_{fi}(x_{ink},\ y_{ink},\ z_{ink})$ the 3D position fingertips of the thumb and the index finger at the kth frame, respectively. The distance between the two fingertips at the kth frame is given as

$$d_{fk} = \sqrt{\left(x_{ftk}-x_{ink}\right)^2+\left(y_{ftk}-y_{ink}\right)^2+\left(z_{ftk}-z_{ink}\right)^2} \tag{2}$$

where d_{fk} and d_{fk-1} represent the distances of the kth frame and previous frame, respectively, the finger speed v_{finger} is estimated as

$$v_{finger} = \alpha\cdot\left(d_{fk}-d_{fk-1}\right) \tag{3}$$

The function utilizes v_{hand} and v_{finger} extracted from each frame:

$$v_k = v_{hand} + v_{finger} \tag{4}$$

and is used as the input of the Bi-LSTM network to spot gestures from video streams, as shown in Figure 2. In our network, the Connectionist temporal classification [12] CTC loss is used to identify whether the sequence frames are in gesture frames or transition frames.

- **3D hand pose extraction:** Using the hand palm location detected by the 3D human pose estimation network, we extract hand ROI and use it for 3D hand pose extraction when the hand palm stands still over the spine base joint. We also estimate the 3D hand pose by using the real-time 3D hand joints tracking network of OccludedHands proposed by Mueller et al. [26] and further additionally fine-tuned it with the hand pose dataset of Stereo Hand Pose Tracking Benchmark [27]. In this method, they utilized both RGB and Depth information to robustly and accurately localize the hand center position and regress the 3D joint from the 2D hand position heat-map. Firstly, they used a CNN network called HALNet to estimate the heat-map of the hand center and then crop the hand region. Secondly, they applied another CNN network called JORNet for a hand cropped frame to generate a heat-map of 2D hand joints and regress 3D hand joint positions from it. The Stereo Hand Pose Tracking Benchmark is a large dataset for 2D and 3D hand pose estimation with 21 joint points for 18,000 images. Due to the robustness and accuracy of its performance, the 3D position of the thumb and index fingertips detected by the network are used for finger speed calculation and other recognition features. In the case where the predicted joint becomes invisible with very low confidence, we estimate this joint position based on its last known position.

- **LSTM:** An LSTM network is a recurrent neural network of a special kind, in which current network output is influenced by previously memorized inputs. The network can learn the contextual information of a temporal sequence. In an LSTM network, the gates and memory cells at time t are given as follows:

$$\begin{cases} i_t = \sigma(W_i[x_t,\ h_{t-1}] +\ b_i \\ f_t = \sigma\Big(W_f[x_t,\ h_{t-1}] +\ b_f \\ o_t = \sigma(W_o[x_t, h_{t-1}] +\ b_o \\ \widetilde{c_t} = \tanh(W_c[x_t, h_{t-1}] +\ b_c), \\ c_t =\ f_t * c_{t-1} +\ i_t * \widetilde{c_t}, \\ h_t = \tanh(c_t) * o_t \end{cases} \tag{5}$$

where i, f, and o are the vectors of input, forget and output gate, respectively. $\widetilde{c_t}$ and c_t are called the "candidate" hidden state and internal memory of the unit. h_t represents the output hidden state. $\sigma(.)$ is a sigmoid function while W and b are connected weights matrix and bias vectors, respectively.

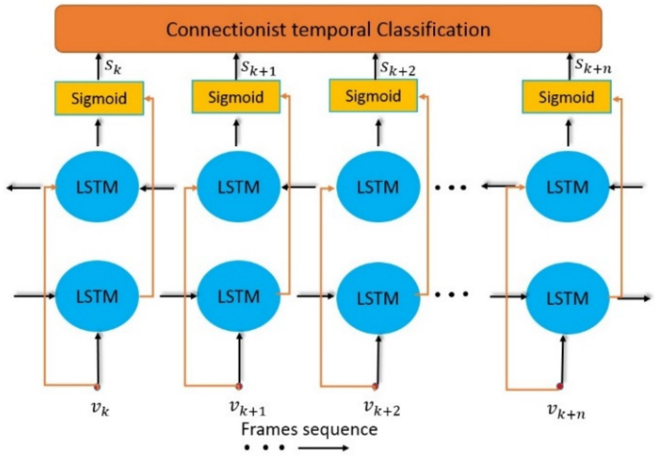


Figure 2. Gesture segmentation with Bi_LSTM and CTC loss.

- **Bi-LSTM network**: While the output of a single forward LSTM network depends only on previous input features, the Bi-LSTM network is known as an effective method for sequence labeling tasks, which is beneficial to both previous and future input features. Bi-LSTM can be considered as a stack of two LSTM layers, in which, a forward LSTM layer utilizes the previous input features while the backward LSTM layer captures the future input features. The benefit of the fact that the Bi-LSTM network considers both previous and future input features is its effectiveness to classify the frame in sequence frame, gesture frame, or transition frame. The prediction error can be reduced by using Bi-LSTM instead of LSTM.

- **Connectionist temporal classification:** The Connectionist temporal classification CTC is known as the loss function which is highly effective in sequential label prediction problems. The proposed algorithm utilizes CTC to detect whether the sequence frames are in gesture frames or transition frames with input from a sequence of Soft-Max layer outputs.

3.2. Gesture Classification

The isolated gestures segmented by the present gesture spotting module are classified into individual gesture classes in the gesture recognition module. The proposed gesture recognition module is a multi-model network called the M-3D network. This model is based on a multi-channel network with three different data modalities, as shown on the right of Figure 1.

In our approach, from each frame of a video, we extract optical flow, 3D pose (hand joint, thumb tip, and index fingertip joint) information of multi-channel features input to the model. Optical flow is determined by two adjacent frames. There are some existing methods of optical flow extraction such as Farneback [28], MPEG flow [29], and Brox flow [30]. The quality motion information of optical flow clearly affects the performance of the gesture recognition model. Therefore, the Brox flow technique is applied to our approach as it has better quality performance compared to other optical flow extraction techniques.

While the key hand and finger joints positions are extracted by the 3D human pose and 3D hand pose extraction network presented in Section 3.1, we only focus on the two most important joints of thumb tip and index fingertip which can describe all gesture types. Our gesture classification algorithm is based on the combination of three 3D_ResNet stream networks of RGB, Optical Flow, Depth channels with an LSTM network of 3D key joint features.

- **Three stream RGB, Optical Flow, and Depth 3D_ResNet networks:** The 3D_CNN framework is regarded as one of the best frameworks for spatiotemporal feature learning. The 3D_ResNet network is an improved version of the residual 3D_CNN framework based on ResNet [31] architecture. The effectiveness of 3D_ResNet has been proved by remarkable performance in action video classification.

The single 3D_ResNet is described in Figure 3. The 3D_ResNet consists of a 3D convolutional layer and is followed by a batch normalization layer and rectified-linear unit layer. Each RGB and Optical Flow stream model is pre-trained on the largest action video classification dataset of the Sports-1M dataset [32]. Input videos are resampled into 16 frames-clips before being fed into the network. Let a resampled sequence of 16 frames RGB frames be $V_c = \{x_{c1}, x_{c2}, \ldots, x_{c16}\}$, Optical Flow frames be $V_{of} = \{x_{of1}, x_{of2}, \ldots, x_{of16}\}$ and Depth frames be $V_d = \{x_{d1}, x_{d2}, \ldots, x_{d16}\}$ and operation function 3D_ResNet network of RGB, Optical Flow and Depth modalities be $\Theta_{\mathrm{c}}(.)$, $\Theta_{\mathrm{of}}(.)$ and $\Theta_{\mathrm{d}}(.)$, respectively. Hence, the prediction probability of two single networks for i classes is

$$P_c\{p_1,\ p_2,\ \ldots,\ p_{16}|V_c\} = \Theta_c(V_c) \tag{6}$$

$$P_{of}\left\{p_1,\ p_2,\ \ldots,\ p_{16}\middle|V_{of}\right\} = \Theta_{of}\left(V_{of}\right) \tag{7}$$

$$P_D\{p_1,\ p_2,\ \ldots,\ p_{16}|V_D\} = \Theta_D(V_D) \tag{8}$$

where p_i is the prediction probability of video belonging to the ith class.

- **LSTM network with 3D pose information:** In dynamic gesture recognition, temporal information learning plays a critical role in the performance of the model. In our approach, we utilize the temporal features by tracking the trajectory of the hand palm together with the specific thumb tip and index fingertip joint. LSTM framework is suitably proposed to learn the features for the gesture classification task. The parameters of our LSTM refer to the approach [33]. Input vectors from a sequence of the LSTM network frames is defined as: $V_j = \{v_{j1}, v_{j2}, \ldots, v_{j16}\}$ where: $v_{jk} = \{J_h(x_{hk,}\ y_{hk,}\ z_{hk})$,

$J_{ft}(x_{ftk}, y_{ftk}, z_{ftk}), J_{fi}(x_{ink}, y_{ink}, z_{ink})\}$ is a 9 × 1 vector which contains 3D position information of key joints at kth frame. The input of the LSTM network corresponds to the dimension of a single frame of sequences of 16 sampled frames in a gesture video that is a tensor for 1 × 9 numbers. The prediction probability output using LSTM with input V_j is

$$P_L\{p_1, p_2, \ldots, p_{16}|V_j\} = \Theta_L(V_j) \tag{9}$$

where $\Theta_L(.)$ denotes the operation function of the LSTM network.

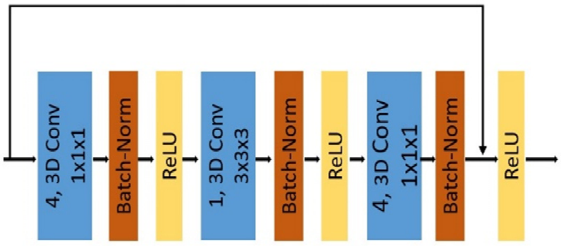


Figure 3. The overview of 3D_ResNet architecture. This figure showed the number of feature map, kernel size of the 3D convolutional layer (3D Conv), batch normalization layer (Batch-Norm), and Rectified-Linear unit layer (ReLU).

- **Multi-modality fusion:** The results of the multiple different channel networks are fused in the final fusion layer to predict a gesture class. It is a fully connected layer where the number of output units is equal to the number of classes on the dataset. The output probability of each class is estimated by pre-trained last fusion layer with $\Theta_{fusion}(.)$ operation function:

$$P\{p_1, p_2, \ldots, p_{16}|V_c\} = \Theta_{fusion}\left\{\begin{array}{ll} P_c\{p_1, p_2, \ldots, p_{16}|V_c\}, & P_{of}\left\{p_1, p_2, \ldots, p_{16}\middle|V_{of}\right\}, \\ P_D\{p_1, p_2, \ldots, p_{16}|V_D\}, & P_L\{p_1, p_2, \ldots, p_{16}|V_j\} \end{array}\right\} \tag{10}$$

The performance of the gesture recognition task is improved by combining the temporal information learning by LSTM network with spatiotemporal features learning by 3D_ResNet that is proved through experimental results.

4. Experiments and Results

In this section, we describe the experiments that evaluate the performance of the proposed approach on three public datasets: 20BN_Jester dataset [34], NVIDIA Dynamic Hand Gesture dataset [5], and Chalearn LAP ConGD dataset [35].

4.1. Datasets

- **20BN_Jester dataset:** is a large dataset collected from 148,092 densely-labeled RGB video clips for hand gesture recognition tasks from 27 gestures classes. The dataset is divided into three subsets: the training set having 118,562 videos, 14,787 videos for the validation set, and 14,743 videos (without labels) for the test set. This dataset has only been used for the gesture classification module.
- **NVIDIA Dynamic Hand Gesture dataset** is a collection of 1532 feebly segmented dynamic hand gesture RGB-Depth videos captured using SoftKinetic DS325 sensor with a frame rate of 30 fps of 20 subjects for 25 gesture classes. The continuous data streams are captured in an indoor car with both dim and bright lighting conditions. This weakly segmented gesture video includes the preparation, nucleus, and transition frames of gesture.
- **Chalearn LAP ConGD dataset:** is a large dataset containing 47,933 gesture instances with 22,535 RGB-Depth videos for both continuous gesture spotting and gesture recognition task. The dataset includes 249 gestures performed by 21 different individ-

uals. This dataset is further divided into three subsets: training set (14,314 videos), validation set (4179 videos), and test set (4042 videos).

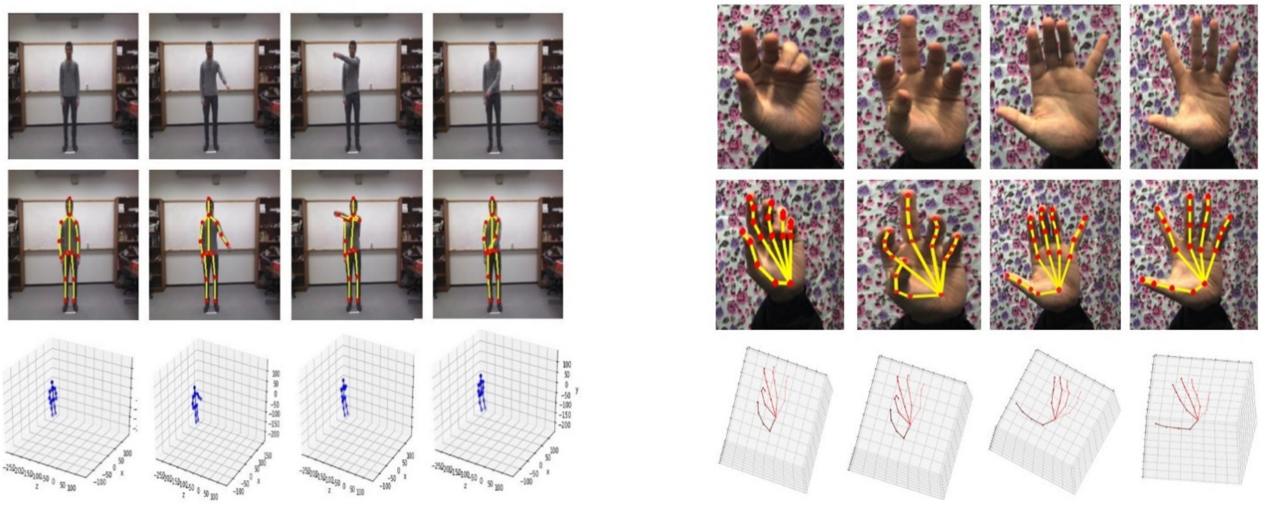
The summary of the three datasets is shown in Table 1.

Table 1. Ablation studies on the ISBI 2016 and ISBI 2017 datasets.

Dataset	Number of Classes	Number of Videos	umber of Videos for Train, Validation, Test Set	Gesture Segmentation Task Provided
20BN_Jester	27	148,092	118,562 | 14,787 | 14,743	No
NVIDIA Hand Gesture	25	1532	1050 | – | 428	Yes
Chalearn LAP ConGD	249	22,535 (47,933 instances)	14,314 | 4179 | 4042	Yes

4.2. Training Process

- **Network training for hand gesture spotting:** To train the Bi-LSTM network for segmentation of continuous gestures, we firstly use a pre-trained 3D human pose extraction network (on Human3.6M dataset) and a pre-trained 3D hand pose extraction network (on Stereo Hand Pose Tracking dataset) to extract the 3D position of key poses. The quality between human and hand pose extraction algorithms are demonstrated in Figure 4. Using those extracted input features for the network, we train the Bi_LSTM network with the provided gesture segmentation labels by a training set of Chalearn LAP ConGD dataset.



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

Figure 4. (**a**) The 2D and 3D human pose estimation examples and (**b**) The 2D and 3D hand pose estimation examples.

Bi-LSTM network is trained with CTC loss for predicting the sequence of binary output values to classify whether the frame belongs to gesture frame or transition frame. In Bi_LSTM, the input layer has 20 time-steps, the hidden layer has 50 memory units, and the last fully connected layer output has one binary value per time-step with a sigmoid active function. The efficient ADAM optimization algorithm [36] is applied to find the optimal weight of the network. The spotting output of the Bi-LSTM network by a given speed input is displayed as in Figure 5.

- **Network training for hand gesture classification:** The single-stream network (pre-trained on Sports-1M dataset) is separately fine-tuned on the huge dataset Chalearn LAP ConGD dataset. Each fine-tuned stream 3D_CNN network weights is learned using ADAM optimization, learning rate with an initial value of 0.0001 reducing by half for every 10 epochs on 200 epochs. Ensemble modeling with 5 3D_ResNet models is applied to increase the classification accuracy. The LSTM network parameters are selected through the observations of experimental results. The optimal LSTM model parameters are 3 memory blocks, and 256 LSTM Cells per memory block. The pre-trained LSTM network is trained with a learning rate of 0.0001 on 1000 epochs. After pre-training of each streaming network, we retrain these networks with a specific dataset. Finally, we concatenate the prediction probability outputs of these trained models to train the weights of the last fusion fully connected layer for gesture classification. Besides training with the 3D_ResNet framework, we also train with the 3D_CNN framework to prove the effectiveness of the proposed algorithm.

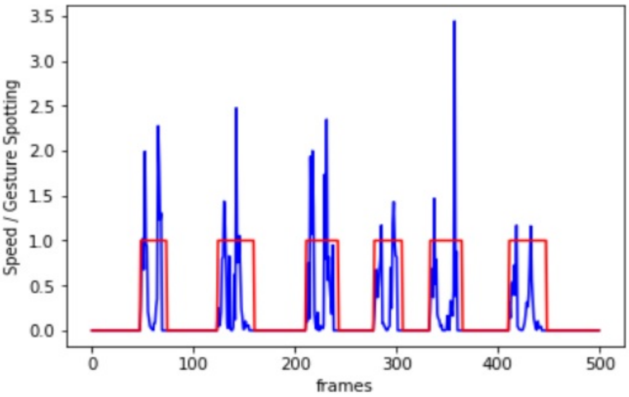


Figure 5. Example of sequence frames segmentation by the Bi_LSTM network. The blue line is the given speed input, and the red line is gesture spotting output (a value of 1.0 indicates the gesture frames).

4.3. Results and Analysis

- **Hand gesture spotting:** To prove the performance of the proposed hand gesture spotting module, we evaluated the model on two datasets—the NVIDIA Dynamic Hand Gesture dataset and Chalearn LAP ConGD dataset. The frame-wise accuracy metric and edit distance score [37] are used to measure the gesture segmentation performance. The results and comparison with other methods are shown in Tables 2 and 3. From the results shown in these tables, our proposed approach achieved the best performance as compared to other methods in both datasets. Our approach gets higher frame-wise accuracy and edits distance score on NVIDIA Dynamic Hand Gesture dataset and Chalearn LAP ConGD dataset than existing works. The significantly improved experimental results proved the effectiveness of the proposed approach.

- **Hand gesture classification:** The performance of our gesture classification module is evaluated by experiments conducted on the 20BN_Jester dataset (without Depth modality) and NVIDIA Dynamic Hand Gesture dataset. The accuracy comparison with other approaches for isolated dynamic hand gesture classification is shown in Table 3.

Table 3 shows that our gesture recognition module obtained a positive result. The recognition performance is improved by using 3D data information of key joints. Moreover, the recognition performance of our method is among the top performers of existing approaches, with an accuracy of 95.6% on the 20BN-Jester Dataset and an accuracy of 82.4% on the NVIDIA Hand Gesture dataset.

- **Continuous hand gesture spotting classification:** To entirely evaluate our approach on continuous dynamic hand gesture spotting recognition, we apply the Jaccard index [3] for measuring the performance. For a given gesture video, the Jaccard index estimates the average relative overlap between the ground truth and the predicted sequences of frames. A sequence S is given by *i*th class gesture label and binary vector ground truth $G_{s,i}$, while the binary vector prediction for the ith class is denoted as $P_{s,i}$. The binary vector $G_{s,i}$ and $P_{s,i}$ are vectors with 1-values indicating the corresponding frames in which the ith gesture class is being performed. So, the Jaccard Index for the given sequence S is computed by the following formula of

$$J_{s,i} = \frac{G_{s,i} \bigcap \ P_{s,i}}{G_{s,i} \bigcup \ P_{s,i}} \tag{11}$$

When $G_{s,i}$ and $P_{s,i}$ are empty vectors, the Jaccard Index $J_{s,i}$ is set as 0. For a given sequence S containing L number of true class labels l_s, the Jaccard Index is estimated by the function:

$$J_s = \frac{1}{l_s} \sum\nolimits_{i=1}^{l} J_{s,i} \tag{12}$$

For all testing sequence of n gestures: $s = \{s_1, s_2, \ldots, s_n\}$ the mean Jaccard Index $\overline{J_s}$ $J \ \rightarrow_s \ J \ \rightarrow_s$ is applied to evaluation as follows:

$$\overline{J_s} = \frac{1}{n} \sum\nolimits_{j=1}^{n} J_{s,j} \tag{13}$$

The spotting-recognition performance comparison of our proposed approach to the existing methods by evaluation experiment on the test set of Chalearn LAP ConGD dataset is shown in Table 4.

Table 2. Gestures spotting performance comparison with different methods on NVIDIA Hand Gesture dataset. Bold values are highest indices.

Method	NVIDIA Hand Gesture		Chalearn LAP ConGD	
	Frame-Wise Accuracy	Edit Distance Score	Frame-Wise Accuracy	Edit Distance Score
2S-RNN [4]	80.3	74.8	87.5	86.3
Proposed in [3]	84.6	79.6	90.8	88.4
R-3DCNN [5]	90.1	88.4	90.4	90.1
Our proposed	**91.2**	**89.6**	**93.1**	**93.8**

Table 3. Gesture classification performance comparison of different methods on the 20BN_Jester dataset and NVIDIA Hand Gesture dataset. Bold values are highest indices.

Method	Accuracy on 20BN-Jester Dataset	Accuracy on NVIDIA Hand Dataset
iDT-HOG [2]	-	59.1
C3D [8]	91.6	69.3
R-3DCNN [5]	95.0	79.3
MFFs [9]	**96.2**	**84.7**
3D_ResNet (RGB)	92.8	75.5
3D_ResNet (RGB + Optical flow)	93.3	77.8
Ours M3D	95.6	82.4

Table 4. The spotting-recognition performance comparison of our proposed approach to existing methods on the test set of the Chalearn LAP ConGD dataset. Bold values are highest indices.

Method	Mean Jaccard Index
2S-RNN [4]	0.5162
Proposed in [3] (RGB + Depth)	0.5950
R-3DCNN [5]	0.5154
Our proposed (3D_CNN)	0.5982
Our proposed	**0.6159**

From the results illustrated in Table 4, the mean Jaccard Index on the test set of the Chalearn LAP ConGD dataset shows that the proposed method achieves satisfactory performance on the dataset. By using 3D key joint features and multiples, the recognition performance is significantly enhanced.

5. Discussions

In Section 4.3, we have shown the effectiveness of our method on the three datasets. In terms of hand gesture spotting, we get the best results of both indexes on the NVIDIA Dynamic Hand Gesture dataset and Chalearn LAP ConGD dataset. The extraction of human pose and hand pose helps us track the hand movement more accurately and detect the beginning and the end of the sequence, avoiding the minor motion that could contaminate the following classification task. In the task of hand gesture classification, Table 3 presents the efficiency of the addition of modalities into our model on both the 20BN_Jester dataset and the NVIDIA Dynamic Hand Gesture dataset. Different views of data are crucial to the performance of the hand gesture classification. Continuous gesture classification is more difficult when there are several kinds of gestures in one video, which means the capability of gesture spotting greatly influences the performance of gesture classification. In Table 4, we get the best results when doing both tasks on the Chalearn LAP ConGD dataset.

6. Conclusions

In this paper, we presented an effective approach for continuous dynamic hand gesture spotting recognition for RGB input data. The continuous gesture sequences are firstly segmented into separate gestures by utilizing the motion speed of key 3D poses as the input of the Bi-LSTM network. After that, each segmented gesture is defined in the gesture classification module using a multi-modal M-3D network. In this network, three 3D_ResNet stream networks of RGB, Optical Flow, Depth data channel, and LSTM networks of 3D key pose features channel are effectively combined for gesture classification purposes. The results of the experiments conducted on the ChaLearn LAP ConGD Dataset, NVIDIA Hand Gesture dataset, and 20_BN Jester dataset proved the effectiveness of our proposed method. In the future, we will try to include other different modalities to improve the performance. The tasks of gesture spotting and classification in this paper are performed separately into 2 steps. The upcoming plan is to do both tasks by one end-to-end model so that it is more practical in real-world problems.

Author Contributions: Conceptualization, G.-S.L. and N.-H.N.; methodology, N.-H.N.; writing—review and editing, N.-H.N., T.-D.-T.P., and G.-S.L.; supervision, G.-S.L., S.-H.K., and H.-J.Y.; project administration, G.-S.L., S.-H.K., and H.-J.Y.; funding acquisition, G.-S.L., S.-H.K., and H.-J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT). (NRF-2020R1A4A1019191) and also by the Bio & Medical Technology Development Program of the National Research Foundation (NRF)& funded by the Korean government (MSIT) (NRF-2019M3E5D1A02067961).

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: Not Applicable.

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

References

1. Yang, H.-D.; Sclaroff, S.; Lee, S.-W. Sign Language Spotting with a Threshold Model Based on Conditional Random Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *31*, 1264–1277. [CrossRef] [PubMed]
2. Słapek, M.; Paszkiel, S. Detection of gestures without begin and end markers by fitting into Bézier curves with least squares method. *Pattern Recognit. Lett.* **2017**, *100*, 83–88. [CrossRef]

3. Wang, H.; Wang, P.; Song, Z.; Li, W. Large-scale multimodal gesture recognition using heterogeneous networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 3129–3137.
4. Chai, X.; Liu, Z.; Yin, F.; Liu, Z.; Chen, X. two streams recurrent neural networks for large-scale continuous gesture recognition. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 31–36.
5. Molchanov, P.; Yang, X.; Gupta, S.; Kim, K.; Tyree, S.; Kautz, J. Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4207–4215.
6. Naguri, C.R.; Bunescu, R.C. Recognition of Dynamic Hand Gestures From 3D Motion Data Using LSTM and CNN Architectures. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; pp. 1130–1133.
7. Wu, D.; Pigou, L.; Kindermans, P.-J.; Le, N.D.-H.; Shao, L.; Dambre, J.; Odobez, J.-M. Deep Dynamic Neural Networks for Multimodal Gesture Segmentation and Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 1583–1597. [CrossRef] [PubMed]
8. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
9. Kopuklu, O.; Kose, N.; Rigoll, G. Motion Fused Frames: Data Level Fusion Strategy for Hand Gesture Recognition. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 2184–21848.
10. Narayana, P.; Beveridge, J.R.; Draper, B.A. Gesture Recognition: Focus on the Hands. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5235–5244.
11. Zhu, G.; Zhang, L.; Mei, L.; Shao, J.; Song, J.; Shen, P. Large-scale Isolated Gesture Recognition Using Pyramidal 3D Convolutional Networks. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 19–24.
12. Graves, A.; Fernández, S.; Gomez, F.; Schmidhuber, J. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In Proceedings of the 23rd International Conference on Machine Learning, Pittsburgh, PA, USA, 25–29 June 2006; pp. 369–376.
13. Hara, K.; Kataoka, H.; Satoh, Y. Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet? In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6546–6555.
14. Hoang, N.N.; Lee, G.-S.; Kim, S.-H.; Yang, H.-J. Continuous Hand Gesture Spotting and Classification Using 3D Finger Joints Information. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 539–543.
15. Krishnan, N.C.; Lade, P.; Panchanathan, S. Activity gesture spotting using a threshold model based on Adaptive Boosting. In Proceedings of the 2010 IEEE International Conference on Multimedia and Expo, Singapore, 19–23 July 2010; pp. 155–160.
16. Ullah, A.; Ahmad, J.; Muhammad, K.; Sajjad, M.; Baik, S.W. Action Recognition in Video Sequences using Deep Bi-Directional LSTM With CNN Features. *IEEE Access* **2018**, *6*, 1155–1166. [CrossRef]
17. Donahue, J.; Hendricks, L.A.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.
18. Du, Y.; Wang, W.; Wang, L. Hierarchical recurrent neural network for skeleton based action recognition. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1110–1118.
19. Zhu, T.; Zhou, Y.; Xia, Z.; Dong, J.; Zhao, Q. Progressive Filtering Approach for Early Human Action Recognition. *Int. J. Control Autom. Syst.* **2018**, *16*, 2393–2404. [CrossRef]
20. Ding, Z.; Chen, Y.; Chen, Y.L.; Wu, X. Similar Hand Gesture Recognition by Automatically Extracting Distinctive Features. *Int. J. Control Autom. Syst.* **2017**, *15*, 1770–1778. [CrossRef]
21. Zhu, T.; Xia, Z.; Dong, J.; Zhao, Q. A Sociable Human-robot Interaction Scheme Based on Body Emotion Analysis. *Int. J. Control Autom. Syst.* **2019**, *17*, 474–485. [CrossRef]
22. Tran, D.-S.; Ho, N.-H.; Yang, H.-J.; Baek, E.-T.; Kim, S.-H.; Lee, G. Real-Time Hand Gesture Spotting and Recognition Using RGB-D Camera and 3D Convolutional Neural Network. *Appl. Sci.* **2020**, *10*, 722. [CrossRef]
23. Zhou, X.; Huang, Q.; Sun, X.; Xue, X.; Wei, Y. Towards 3D Human Pose Estimation in The Wild: A Weakly-Supervised Approach. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 398–407.
24. Ionescu, C.; Papava, D.; Olaru, V.; Sminchisescu, C. Human3. 6m: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *36*, 1325–1339. [CrossRef] [PubMed]
25. Newell, A.; Yang, K.; Deng, J. Stacked Hourglass Networks for Human Pose Estimation. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 483–499.

26. Müeller, F.; Mehta, D.; Sotnychenko, O.; Sridhar, S.; Casas, D.; Theobalt, C. Real-Time Hand Tracking Under Occlusion from an Egocentric RGB-D Sensor. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 1284–1293.
27. Zhang, J.; Jiao, J.; Chen, M.; Qu, L.; Xu, X.; Yang, Q. 3D Hand Pose Tracking and Estimation Using Stereo Matching. *arXiv* **2016**, arXiv:1610.07214.
28. Farnebäck, G. Two-Frame Motion Estimation Based on Polynomial Expansion. In Proceedings of the Scandinavian Conference on Image Analysis, Halmstad, Sweden, 29 June–2 July 2003; Springer: Berlin/Heidelberg, Germany, 2003; pp. 363–370.
29. Kantorov, V.; Laptev, I. Efficient Feature Extraction, Encoding and Classification for Action Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2593–2600.
30. Brox, T.; Bruhn, A.; Papenberg, N.; Weickert, J. High Accuracy Optical Flow Estimation Based on a Theory for Warping. In Proceedings of the European Conference on Computer Vision, Prague, Czech Republic, 11–14 May 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 25–36.
31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
32. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-Scale Video Classification with Convolutional Neural Networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1725–1732.
33. Sarkar, A.; Gepperth, A.; Handmann, U.; Kopinski, T. Dynamic Hand Gesture Recognition for Mobile Systems Using Deep LSTM. In Proceedings of the 9th International Conference on Intelligent Human Computer Interaction, Evry, France, 11–13 December 2017; pp. 19–31.
34. Materzynska, J.; Berger, G.; Bax, I.; Memisevic, R. The Jester Dataset: A Large-Scale Video Dataset of Human Gestures. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 28–29 October 2019.
35. Wan, J.; Zhao, Y.; Zhou, S.; Guyon, I.; Escalera, S.; Li, S.Z. ChaLearn Looking at People RGB-D Isolated and Continuous Datasets for Gesture Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 56–64.
36. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference for Learning Representation (ICLR), San Diego, CA, USA, 5–8 May 2015.
37. Lea, C.; Vidal, R.; Hager, G.D. Learning convolutional action primitives for fine-grained action recognition. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1642–1649.