LD-ConGR: A Large RGB-D Video Dataset for Long-Distance Continuous Gesture Recognition

Dan Liu[1] Libo Zhang[1,2*] Yanjun Wu[1,2]

[1]Institute of Software Chinese Academy of Sciences, Beijing, China
[2]Hangzhou Institute for Advanced Study, UCAS, Hangzhou, China

{liudan, libo, yanjun}@iscas.ac.cn

Abstract

Gesture recognition plays an important role in natural human-computer interaction and sign language recognition. Existing research on gesture recognition is limited to close-range interaction such as vehicle gesture control and face-to-face communication. To apply gesture recognition to long-distance interactive scenes such as meetings and smart homes, a large RGB-D video dataset LD-ConGR is established in this paper. LD-ConGR is distinguished from existing gesture datasets by its long-distance gesture collection, fine-grained annotations, and high video quality. Specifically, 1) the farthest gesture provided by the LD-ConGR is captured $4m$ *away from the camera while existing gesture datasets collect gestures within* $1m$ *from the camera; 2) besides the gesture category, the temporal segmentation of gestures and hand location are also annotated in LD-ConGR; 3) videos are captured at high resolution (*1280×720 *for color streams and* 640×576 *for depth streams) and high frame rate (*30 *fps). On top of the LD-ConGR, a series of experimental and studies are conducted, and the proposed gesture region estimation and key frame sampling strategies are demonstrated to be effective in dealing with long-distance gesture recognition and the uncertainty of gesture duration. The dataset and experimental results presented in this paper are expected to boost the research of long-distance gesture recognition. The dataset is available at https://github.com/Diananini/LD-ConGR-CVPR2022.*

1. Introduction

Gesture is an important way of information transmission. We use gestures to assist our language expression, communicate with the deaf, direct traffic, and so on. Furthermore, gestures help us interact with machines more naturally and conveniently: 1) A simple gesture can replace multiple mouse and keyboard operations. 2) In scenes such as hospitals, conferences, and smart homes, people prefer touchless interaction methods such as gestures and voice. 3) Gesture interaction is more user-friendly and lowers the barrier to using smart devices. There are many datasets [1, 12, 15, 21, 27] and related research [2, 3, 10, 16, 18, 28] for gesture recognition. These datasets focus on interaction with wearable devices (*e.g.*, EgoGesture [27]), interaction with vehicles (*e.g.*, NVIDIA Gesture [15]), sign language and symbolic gestures (*e.g.*, ChaLearn ConGD [21]), or interaction with computers (*e.g.*, Jester [12] and IPN Hand

Figure 1. Example frames from gesture datasets. The upper left corner of each frame marks the dataset from which it is sampled. The gestures in our dataset LD-ConGR are collected at long distances and are more challenging to recognize.

*Corresponding author (libo@iscas.ac.cn). This work was supported by the Key Research Program of Frontier Sciences, CAS, Grant No. ZDBS-LY-JSC038. Libo Zhang was supported CAAI-Huawei MindSpore Open Fund and Youth Innovation Promotion Association, CAS (2020111).

[1]). As we can see, the existing datasets are all oriented towards close-range gesture interaction and collect gestures at a very close distance from the subjects. However, in many scenarios such as the meeting and home automation, users are far away from the machines to be controlled. Moreover, limited by the early data acquisition sensors, the existing datasets can not meet the high demand of long-distance gesture recognition for video quality.

In this paper, a large high-quality RGB-D video dataset LD-ConGR is established for long-distance continuous gesture recognition. Firstly, LD-ConGR draws attention to long-distance gesture interaction. Unlike the existing datasets that record gestures within $1m$ from the camera, we capture gestures at long distances (between $1m$ and $4m$). Fig. 1 shows example frames sampled from different gesture datasets. It can be seen that the gestures in LD-ConGR are captured with a large field of view, and the hands are small and difficult to recognize, which poses a new challenge for gesture recognition. Secondly, LD-ConGR provides fine-grained annotations for continuous gesture recognition. Continuous gesture recognition requires not only to classify gestures but also to detect the specific duration of gestures in the video. In LD-ConGR, each video contains multiple gestures, and all gestures are manually labeled with the categories and the start and end frames in the video. It should be noted that we also annotate the position of the hand in each frame, which provides researchers with more detailed information and brings more possibilities for accurate gesture recognition. Lastly, videos collected in LD-ConGR are of high quality. The Kinect V4[1], equipped with an advanced depth sensor, is used to collect high-quality RGB-D video data. The color and depth streams are captured synchronously at 30 fps with resolutions of 1280×720 and 640×570 respectively.

Based on the proposed LD-ConGR dataset, we conducted a series of experimental explorations. A baseline model based on 3D ResNeXt [23] is implemented and achieves 85.33% accuracy on RGB data. To make good use of the depth information, we learn from the ideas of [9] and build a multimodal gesture recognition model ResNeXt-MMTM. It achieves an accuracy of 89.66% on LD-ConGR. To deal with long-distance gesture recognition, we estimate the possible appearing area of the gesture based on hand location and conduct recognition on the estimated gesture region. This strategy increases the accuracy by 9.33% and 7.67% on RGB data and RGB-D data, respectively. Moreover, we observe the large difference in gesture duration brings great difficulties to recognition. In view of this, we try to extract key frames of the video based on inter-frame difference to remove redundant frames for long-duration gestures. Results show that the key frame sampling strategy reduces the impact of gesture speed and duration, and realizes high-speed and accurate recognition with fewer frames.

In summary, the main contributions of this paper are as follows: 1) We release a new large-scale RGB-D video dataset LD-ConGR. To the best of our knowledge, this is the first dataset for long-distance continuous gesture recognition. LD-ConGR is finely annotated with gesture category, temporal segmentation (the start and end frames of the gesture in the video), and hand position. The dataset will be available to the public. 2) The results of baseline methods and state-of-the-art gesture recognition methods on LD-ConGR are reported to provide references for subsequent research. 3) For the two main challenges raised by LD-ConGR: the long-distance recognition and the uncertainty of gesture duration, we explore possible solutions and provide more research directions.

2. Related Work

2.1. Gesture Recognition Datasets

Existing gesture recognition datasets are collected close to the subject making gestures, as they are established for close-range human-computer interaction or sign language understanding. EgoGesture [27] focuses on gesture interaction with wearable devices. It is collected with Intel RealSense SR300 RGB-D camera mounted on the head of the subject. In Jester [12] and IPN Hand [1] datasets, the subjects are asked to record gestures using their own personal computer or laptop. They sit in front of the computer camera and simulate using gestures to operate the computer. NVIDIA Gesture [15] aims to make it possible to manipulate cars through gestures. The gestures in NVIDIA Gesture [15] are recorded inside a car simulator. The gesturing hand is directly in front of the collector, SoftKinetic DS325 sensor, and is very close to the sensor. As for ChaLearn ConGD [21], the subjects perform gestures standing within $1m$ from the Kinect V1 camera. It can be seen that the gesturing hand in these datasets is very close to the camera, which means that the gesture is salient and easy to recognize (See Fig. 1). However, in many application scenarios, it is necessary to interact with the machine from a long distance. For example, in conferences, participants hope to remotely control the interactive meeting board to play slides and turn pages. In home automation, we are pleased to use gestures to adjust the lights, TV volume, and movie playback progress. To fill the vacancy of long-distance gesture interaction data, we release a large RGB-D video dataset LD-ConGR in this paper. Subjects are asked to perform gestures at 6 recording spots in each scene, which are evenly distributed within a range of $1m$ to $4m$ from the camera. This makes our dataset contain various gesture distances and complex backgrounds. It provides more realistic and comprehensive data for gesture interaction.

Another issue is the granularity of annotations. For ges-

[1]https://azure.microsoft.com/en-us/services/kinect-dk

ture recognition datasets, the category of gestures is the coarsest-grained annotation, and the finer annotation needs to mark the start and end of the gesture in video. Except for the EgoGesture dataset [27], all other datasets mentioned above provide precise temporal segments of gestures. It is worth noting that our proposed dataset LD-ConGR further annotates the location of the hand in each frame, which can help to quickly locate the key area of the gesture in long-distance gesture recognition.

2.2. Gesture Recognition Methods

According to whether the temporal boundary of the gesture is pointed out, gesture recognition can be divided into isolated gesture recognition and continuous gesture recognition. Isolated gesture recognition refers to the classification of a given sequence that contains a single gesture. Continuous gesture recognition refers to detecting the beginning and end of each gesture instance and identifying its category for a given video sequence, which may contain more than one gesture.

The general method of continuous gesture recognition. Sliding window is a common strategy to deal with continuous gesture recognition [2, 10, 15]. The basic idea of this method is to slide on the video sequence with a certain step size and window size and conduct gesture classification on these window clips. The prediction results of the windows are fused through the cumulative average or other strategies to generate the final gesture detection and recognition results. An inevitable problem in continuous gesture recognition is the detection and processing of non-gesture segments. The existing methods can be roughly divided into two categories. One is to deal with non-gesture clips separately [1, 10]. This kind of methods first trains a lightweight binary classifier to detect whether gestures appear. If there is a gesture in the video clip, then perform multi-class classification on it. The other is to add an extra *no gesture* class and process it together with the gesture classes [3, 15]. In other words, it directly predicts the probabilities that the video clip belongs to all the gesture classes and *no gesture* class. Compared with the first class of methods, the second one can be optimized end-to-end, so we deal with non-gesture fragments according to the second method.

Feature extraction for dynamic gestures. For the feature extraction of dynamic gestures, both spatial and temporal dimensions need to be considered. Most of the existing methods use Convolutional Neural Networks (CNNs) to extract spatial features. For the representation of temporal features, there are three main methods: The first is based on optical flow [4, 16, 18], motion vectors [25], *etc*., the second is to learn temporal features using Recurrent Neural Networks (RNNs) [3, 8, 15, 17, 26], and the last is based on 3D convolutions [4, 10, 14, 20, 28], which perform convolution in three dimensions (two spatial dimensions and one temporal dimension) to extract spatial features and temporal features simultaneously. In this paper, we adopt 3D ResNeXt [23], a 3D CNN architecture, as a baseline model.

Multimodal learning in gesture recognition. Multimodal data can reflect different aspects of the context. Learning and fusing relevant features from multimodal data will greatly benefit gesture recognition. Multimodal data fusion can be achieved at the data level, feature level, or decision level. The data-level fusion tries to fuse multimodal data before feeding it to the recognition model. [11] fuses the optical flow and color modalities by appending the optical flow maps calculating from the previous frames to the RGB frame as extra channels. The feature-level fusion first extracts the features of different modal data and then designs algorithms to fuse these features for prediction. [13] analyzes the pair-wise correlation between features from different modalities to fuse the features. [9] proposes a Multimodal Transfer Module (MMTM), which can be applied to any level of the feature hierarchy. It enables the fusion of modalities with different spatial dimensions. The hierarchical progressive fusion by adding MMTM to multiple layers of the network improves the performance significantly. [6] fuses features from different modalities based on attention mechanism, and the temporal order of the data is considered during the fusion. This ensures that the multi-modal features are aligned in the temporal dimension. As for the decision-level fusion, it designs the network structure separately for each data modality and then averages [4] or weighted averages [16] the predicted scores obtained from different data modalities as the final result. The available information for the decision-level fusion is limited to the top-level output of the network, which is very abstract and compact, so the performance improvement it brings is often not as much as the feature-level fusion. In view of the above analysis, we integrate the information of RGB and depth modalities at the feature level based on MMTM.

3. The LD-ConGR Dataset

The LD-ConGR dataset is developed for the long-distance gesture recognition task. It contains 10 gesture classes, of which three are static gestures and seven are dynamic gestures. The standard practice for these gestures is shown in Fig. 2. It can be seen that a variety of hand shapes and movements are involved in the design of gestures. A total of 542 videos and 44,887 gesture instances are collected in LD-ConGR. The videos are collected from 30 subjects in 5 different scenes and captured in a third perspective with Kinect V4. Each video contains a color stream and a depth stream. The two streams are recorded synchronously at 30 fps. The resolutions of the color stream and the depth stream are 1280×720 and 640×576, respectively. The distance between the subject and the camera ranges from $1m$ to $4m$, which ensures the long-distance characteristic

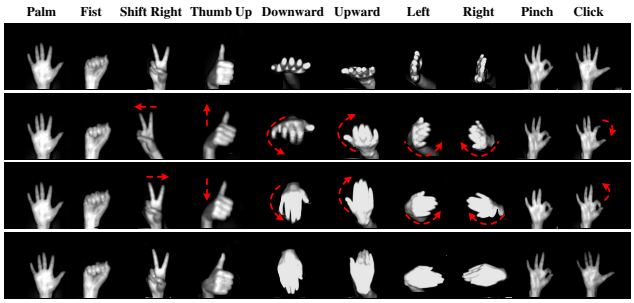

Figure 2. Ten gesture classes of LD-ConGR dataset, including three static gestures ('palm', 'fist', 'pinch') and seven dynamic gestures ('shift right', 'thumb up', 'downward', 'upward', 'left', 'right', 'click'). Each column shows the standard practice of the gesture noted above. The red arrows indicate the direction of the hand movement.

Classes	Instances			Duration			
	Total	Train	Test	Avg.	Std.	Max.	Min.
Palm	15,315	11,672	3,643	10.96	4.20	54	4
Fist	2,689	2,059	630	16.38	4.60	37	5
Thumb Up	2,689	2,059	630	36.56	7.60	72	10
Shift Right	2,689	2,062	627	38.52	8.29	78	8
Downward	2,686	2,055	631	27.32	7.09	62	7
Upward	2,679	2,049	630	27.39	7.13	85	11
Left	2,684	2,053	631	26.27	6.40	92	10
Right	2,690	2,060	630	25.58	6.33	54	8
Pinch	2,686	2,056	630	16.28	4.75	60	5
Click	8,080	6,190	1,890	11.88	3.48	33	5
Total	44,887	34,315	10,572	18.70	10.80	92	4

Table 1. Statistics of the proposed LD-ConGR dataset. Gesture duration is measured in frames.

of the gestures in the dataset. We label the category and the start and end frames for each gesture instance in video. In addition, the locations of the hands in each frame are marked with bounding boxes, which benefits long-distance gesture recognition. In the following subsections, we will introduce the collection and annotation of the dataset, report data statistics, and make a comparative analysis of the LD-ConGR dataset and other gesture recognition datasets.

3.1. Data Collection and Annotation

Collection. We use Kinect V4 to collect RGB-D video data. Kinect V4, known as Kinect for Azure and released in 2019, is equipped with a 12-megapixel RGB camera and a 1-megapixel depth camera, ensuring the quality of the captured videos. We synchronously record color and depth streams with resolutions of 1280×720 and 640×576 respectively and a frame rate of 30 fps. The video recording is arranged in 5 meeting rooms with different designs and furnishings. Six recording spots are set in each scene, and the distance from the recording spot to the camera is between $1m$ and $4m$ (See supplementary material for more details).

A total of 30 subjects participate in data collection and are randomly assigned to 5 scenes to record gestures. All subjects are shown the standard gestures before recording, and the recorded video will be further checked whether the gestures are correct. The subjects are asked to perform gestures continuously, and a short break is allowed between two gesture instances. The data is only allowed for academic research and we will provide strict access for applicants who sign data use agreements. The subjects were informed of the uses of the data and signed informed consent.

Annotation. We label the category and the start and end frames for each gesture instance in the video. Frames are extracted from the video at 30 fps for gesture annotation. As the color and depth video streams are synchronized, only the color streams need to be labeled, and the annotations of depth videos can be obtained accordingly. In addition, the positions of the hands in each frame are annotated with bounding boxes. In long-distance gesture recognition, the hand area accounts for a small proportion of the frame. Therefore, the localization of hands can help to get rid of the interference of redundant background information and focus on the gesture itself. Different from the gesture labeling, hand position annotation is carried out at 15 fps. Since the hand position between adjacent frames changes little at a high frame rate, marking the hand position at a low frame rate can save a lot of time and manpower.

3.2. Data Statistics

The LD-ConGR dataset contains $44,887$ gesture instances of 10 different hand gesture classes. The dataset is randomly divided into training set and testing set by subjects. The training set includes a total of $34,315$ gestures collected from 23 subjects. The gestures performed by the other 7 subjects, $10,572$ gestures in total, are collected as the testing set. The number of instances of each gesture class is shown in Tab. 1. There are more 'Palm' instances as the 'Click' starts and ends with 'Palm' (See Fig. 2) and these palms are also counted. As for 'Click', it is collected in two forms: one is an independent click, and the other is two consecutive clicks (simulating a double-click), so the amount of 'Click' is about three times that of other classes (except for 'Palm'). The gesture duration, measured in frames, is analyzed and detailed statistics, including the average, standard deviation, maximum, and minimum, are reported in Tab. 1. It can be seen that the duration of gestures fluctuates greatly both within the same gesture class and between different gesture classes. In the whole data set, the duration difference between the longest gesture and the shortest gesture can reach 88 frames (92 *vs.* 4). Even for instances of the same gesture class, the maximum duration difference is 82

Dataset	Classes	Videos	Instances	Distance	Label			Resolution		Frame Rate/fps
					Cat.	Seg.	Loc.	RGB	Depth	
Jester [12]	27	148,092	148,092	$< 1m$	✓			$* \times 100$	-	12
NVIDIA Gesture [15]	25	1,532	1,532	$< 1m$	✓	✓		320×240	320×240	30
EgoGesture [27]	83	2,081	24,161	$< 1m$	✓	✓		640×480	640×480	30
ChaLearn ConGD [21]	249	22,535	47,933	$< 1m$	✓	✓		320×240	320×240	10
IPN Hand [1]	13	200	4,218	$< 1m$	✓	✓		640×480	-	30
LD-ConGR (Ours)	10	542	44,887	$1m \sim 4m$	✓	✓	✓	1280×720	640×576	30

Table 2. Comparison of our dataset LD-ConGR and popular gesture recognition datasets. 'Distance' means subject distance, *i.e.*, the distance between the camera and the subject. The label here includes gesture category ('Cat.'), temporal segmentation ('Seg.'), and hand location ('Loc.').

frames ('Left' class, 92 *vs*. 10). There are two main reasons for the large difference in gesture duration. One is that individuals make gestures at different speeds, and the other is that different classes of gestures take different amounts of time. The huge difference and uncertainty of gesture duration also bring great challenges to gesture recognition. We will analyze and explore possible solutions in Sec. 4.3.

3.3. Comparative Analysis

In Tab. 2, we compare our dataset LD-ConGR with the publicly available gesture recognition datasets, including Jester [12], NVIDIA Gesture [15], EgoGesture [27], ChaLearn ConGD [21], and IPN Hand [1]. Below we will make a detailed comparison and explain the advantages of our dataset from three aspects: subject distance, label , and video quality.

Subject distance. In all these datasets, only our dataset LD-ConGR is established for long-distance gesture recognition. As shown in Fig. 1, in the previously published datasets, the subjects are very close to the camera during recording (within $1m$). Therefore, the hands in the video are salient, and the details of the gestures are clear and distinct. It is easy to recognize such gestures correctly. However, in many scenes that require long-distance gesture interaction, such as meetings and movie watching, it is necessary to accurately recognize gestures even when the subject is far away from the camera. Our data set is constructed to solve this problem. In our setting, the subject is $1m \sim 4m$ away from the camera when performing gestures. To the best of our knowledge, the LD-ConGR dataset is the first dataset targeted at long-distance gesture recognition.

Label. The Jester dataset [12] is collected for gesture classification and provides only gesture category annotations. Other than the category information, the NVIDIA Gesture [15], EgoGesture [27], ChaLearn ConGD [21], IPN Hand [21], and our dataset LD-ConGR also provide specific temporal segmentation for each gesture, *i.e.*, the start and end frames of the gesture in the video. This is very important for continuous gesture detection, which needs not only to classify gestures but also to determine the beginning and end of gestures. Moreover, we annotate the location of hands in each video frame. The LD-ConGR is the first gesture recognition dataset to provide such fine-grained annotations. We hope that the precise annotation of hand position can bring more help to long-distance gesture recognition.

Quality. The videos collected in our dataset are of high quality. As we can see in Tab. 2, our dataset provides high-definition RGB video data (1280×720), while the highest resolution of other gesture datasets is only 640×480. In addition, the depth streams (captured synchronously with color streams) are available in our dataset and have a higher resolution (640×576) compared to NVIDIA Gesture (320×240), EgoGesture (640×480), and ChaLearn ConGD (320×240). Moreover, the color and depth streams are captured at a high frame rate (30 fps).

4. Experimental Studies

In this section, we will first introduce a baseline method for the LD-ConGR dataset, and then discuss two important issues raised by the dataset: long-distance gesture recognition and great uncertainty of gesture duration. Finally, we evaluate the state-of-the-art methods in the field of gesture and action recognition on the LD-ConGR dataset.

4.1. A Baseline Method

We build a baseline model based on ResNeXt-101 [23] and conduct experiments to explore the recognition performance of different input modalities. In our experiments, the ResNeXt-101 network is used to do gesture recognition on a single modality (RGB or depth). For multi-modal gesture recognition, we design a multimodal fusion model ResNeXt-MMTM learning from the idea of [9]. The architecture of ResNeXt-MMTM is shown in Fig. 3. ResNeXt-MMTM maintains a ResNeXt-101 network for each modality and fuses the features of different modalities at multiple layers through Multimodal Transfer Modules (MMTMs) [9]. The MMTM learns a multimodal embedding and uses

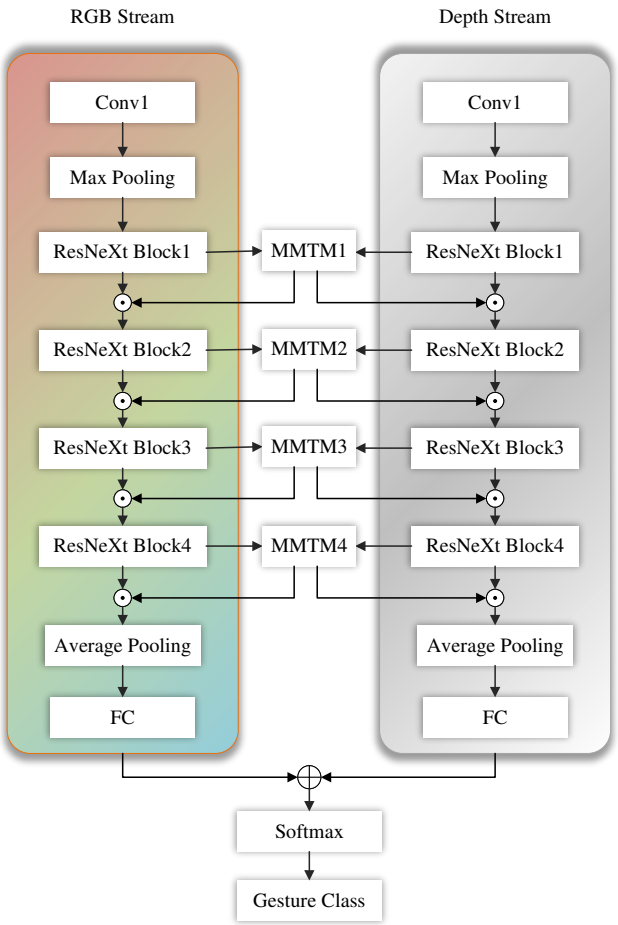

Figure 3. The architecture of the RGB-D baseline model ResNeXt-MMTM. 3D ResNeXt-101 [23] is adopted to extract features from RGB and depth streams, and the features of the two modalities are fused at four levels through MMTMs [9].

it to recalibrate the features of each modality. In our design, the features extracted from RGB and depth streams are blended by the MMTM after each ResNeXt block. The feature vectors output from the fully connected layers are integrated by element-wise addition and then passed to the softmax layer to obtain the final prediction results.

Evaluation metrics. Both isolated gesture recognition and continuous gesture recognition are studied on the LD-ConGR dataset. The accuracy is adopted to evaluate the model for isolated gesture recognition task, while the mean Jaccard Index [21] for continuous gesture recognition task. The Jaccard Index is the ratio of the overlapping frames of the ground-truth gesture segment and the predicted gesture segment to the total frames. For video v and gesture class c, the Jaccard Index can be calculated as:

$$J_{v,c} = \frac{G_{v,c} \cap P_{v,c}}{G_{v,c} \cup P_{v,c}}, \tag{1}$$

where $G_{v,c}$ and $P_{v,c}$ are the sets of all frames belonging to gesture c in the ground truth and prediction of video v, respectively. The $J_{v,c}$ is set to 0 when $G_{v,c} = \emptyset$ or $P_{v,c} = \emptyset$. The Jaccard Index for the video v is defined as the average of the Jaccard Index on all the ground-truth gesture categories C:

$$J_v = \frac{1}{|C|} \sum_{c \in C} J_{v,c}. \tag{2}$$

The model performance is evaluated by the mean Jaccard Index of all the test videos:

$$\overline{J} = \frac{1}{|V|} \sum_{v \in V} J_v, \tag{3}$$

where V is the test video set and $|V|$ refers to the number of videos in the set. $\overline{J}$ cannot accurately reflect the model performance when the classes are imbalanced. In this case, the mean Jaccard Index of each class can be used to further evaluate the model:

$$\overline{J_c} = \frac{1}{|V|} \sum_{v \in V} J_{v,c}. \tag{4}$$

Train and test. In the training phase, the gesture clip is first randomly cropped or cyclically filled to a fixed length of 32 frames, and then multi-scale random clipping is applied to all frames in the spatial dimension to obtain 112×112 regions. The elastic distortion is used for further data enhancement in our experiments. In the testing phase of isolated gesture recognition, central clipping or cyclic filling is employed to generate gesture clips with a length of 32 frames and the frames are scaled to 112×112 in the spatial dimension before being fed into the model. The beginning and end of the gesture are known in isolated gesture recognition, and there is only the need to classify the gesture segments. For continuous gesture recognition, the model is not told when the gesture appears in the video. Here the sliding window method is adopted to deal with this issue. We use a 32-frame window to slide over the video sequence in a certain stride (2 frames in our experiments). Each window clip is preprocessed the same as the isolated gesture recognition and then sent to the model to determine the gesture class (including *no gesture* class). The initial gesture is set to *no gesture*. When the predicted gesture is inconsistent with the previous gesture for two consecutive windows, a new gesture is considered to appear and the indices of the first and last frames of the window are recorded as the start and end of the new gesture, respectively. When the predicted gesture is consistent with the previous window, the end of the gesture is updated to the index of the last frame of the current window.

Results. Results of the baseline method are reported in Tab. 3. The basic ResNeXt-101 model trained on the RGB modality achieves 85.33% accuracy and 0.31 mean

Classes	Accuracy/%		Mean Jaccard Index	
	RGB	RGB-D	RGB	RGB-D
Palm	84.90	90.94	0.02	0.01
Fist	76.35	81.90	0.11	0.11
Thumb Up	95.08	99.21	0.33	0.57
Shift Right	98.41	99.36	0.61	0.65
Downward	98.89	98.89	0.54	0.54
Upward	93.97	96.03	0.34	0.38
Left	91.28	97.62	0.52	0.51
Right	91.90	97.78	0.50	0.50
Pinch	38.89	46.19	0.02	0.01
Click	85.45	87.30	0.11	0.09
Total	85.33	89.66	0.31	0.34

Table 3. Results of the baseline method.

Jaccard Index. It should be noted that when combining color and depth information for recognition, the accuracy and the mean Jaccard Index increase by 4.33% (85.33% *vs*. 89.66%) and 0.03 (0.31 *vs*. 0.34) respectively compared with using only RGB modality. The results indicate that the hand details contained in RGB streams are essential for long-distance gesture recognition, and the depth modality can provide extra information to assist the recognition.

4.2. Long-Distance Gesture Recognition

In long-distance gesture recognition, the hand area occupies a very low proportion of the picture, and the features that the model can use to capture and recognize gestures are very limited. We try to use the position of the hand to estimate the region where the gesture may occur, and then perform gesture detection and recognition in the estimated region. The hand location annotations provided by the LD-ConGR dataset can assist the training process and can be exploited to train a hand detector to estimate the gesture region in the test stage. The gesture region is predicted using the first tracked hand location $R_{hand} = (x, y, w, h)$, where x, y are center coordinates of the hand bounding box R_{hand}, and w, h are the width and height of R_{hand}. The gesture region R_{ges} is the extended rectangular area centered on (x, y):

$$R_{ges} = (x, y, r_w \times w, r_h \times h)\,. \tag{5}$$

$r_w > 1$ and $r_h > 1$ are the extension ratios in width and height. As the 10 gesture classes in LD-ConGR have large horizontal movements and small vertical movements (See Fig. 2), r_w and r_h are set to 5 and 4 respectively in our experiments. The specific processes of training and predicting with gesture region estimation are illustrated in the supplementary material.

The gesture region estimation strategy removes most of the redundant information in the spatial dimension and magnifies the gesture details, which can help the model learn gesture features faster. Moreover, it can locate and recognize gestures in situations where multiple gestures occur at the same time. To verify the effect of this strategy, we have carried out experiments on different input lengths and data modalities. Test results without and with gesture region estimation are shown in the 'Raw' and 'Region' columns of Tab. 4 respectively. With 16-frame length input, the accuracy is improved by 9.86% (92.02% *vs*. 82.16%) on RGB data and 7.75% (93.75% *vs*. 86.00%) on RGB-D data. With 32-frame length input, the accuracy is improved by 9.33% (94.66% *vs*. 85.33%) on RGB data and 7.67% (97.33% *vs*. 89.66%) on RGB-D data. The significant performance improvement proves that gesture region estimation is a good strategy for long-distance gesture recognition.

Input	Modality	Strategy		
		Raw	Region	Region&Key
16-frame	RGB	82.16	92.02	**93.26**
	RGB-D	86.00	93.75	**94.68**
32-frame	RGB	85.33	**94.66**	94.62
	RGB-D	89.66	97.33	**97.45**

Table 4. Accuracy of with and without gesture region estimation and key frame sampling strategies.

4.3. Uncertainty of Gesture Duration

The uncertainty of gesture duration brings great difficulties to gesture recognition, as mentioned in Sec. 3.2. For long-duration gestures, a large window needs to be used to capture long-term temporal dependencies, while for short-duration gestures, a small window is adequate. Moreover, in real-time continuous gesture detection, it is unknowable how long the gesture will last, and it is tricky to set an appropriate prediction window size. A large window will increase the computational cost and slow down the inference speed. On the other hand, too many interference factors may be involved, such as adjacent gestures and non-gesture fragments. A small window may not be able to capture the key information of the gesture, leading to wrong judgments. To solve this problem, we try to extract key frames of the video and perform gesture recognition on the key frames. The key frames function in three aspects: 1) As the video is recorded at a high frame rate, there is a lot of similar information between adjacent frames. Sampling the key frames can remove redundant frames and reduce the computational burden. 2) Different individuals make gestures at different speeds, which makes the temporal features of gestures changeable and hard to learn. Using key frames reduces the impact of different gesture speeds. 3) Key frames reduce the number of frames required to recognize a gesture. In

Classes	Avg. duration		Std. duration		Max. duration	
	Raw	Key	Raw	Key	Raw	Key
Palm	10.96	6.85	4.20	2.74	54	24
Fist	16.38	8.86	4.60	3.03	37	19
Thumb Up	36.56	16.59	7.60	5.06	72	37
Shift Right	38.52	15.82	8.29	5.15	78	41
Downward	27.32	13.02	7.09	4.76	62	31
Upward	27.39	12.92	7.13	4.63	85	32
Left	26.27	12.50	6.40	4.59	92	35
Right	25.58	10.78	6.33	4.47	54	29
Pinch	16.28	8.90	4.75	3.30	60	23
Click	11.88	6.78	3.48	2.23	33	20
Total	18.70	9.60	10.80	4.95	92	41

Table 5. Statistics of gesture duration before and after key frame sampling.

other words, just a small window can achieve high recognition accuracy. In addition, the small window ensures fast predicting speed.

We extract the key frames of the gesture according to the inter-frame difference. The frame difference calculation is limited to the rectangular area centered on the hand. The width is five times the hand width and the height is four times the hand height, consistent with the size of 'gesture region' (See Eq. (5)). Statistics of raw gesture frames and key frames are shown in Tab. 5. It can be seen that key frames sampling removes about half the number of frames (18.70 *vs*. 9.60 in average) and lowers the difference in gesture duration (10.80 *vs*. 4.95 in standard deviation). The longest gesture is reduced from 92 frames to 41 frames. We add the key frame sampling strategy to the model introduced in Sec. 4.2. The test results are listed in the 'Region&Key' column of Tab. 4. It can be seen that with 16-frame input, the accuracy is improved by 1.24% and 0.93% on RGB data and RGB-D data respectively. With 32-frame input, there is no significant performance improvement, as the large window already provides enough long-term information for recognition. On the raw data, a small window has an advantage in speed compared to a large window, but not in accuracy. By contrast, with the key frames, the small input window can store long-term information to obtain high recognition accuracy while maintaining a speed advantage.

4.4. State-of-the-art Evaluation

We evaluate the state-of-the-art gesture and action recognition methods on the proposed LD-ConGR dataset. The results are shown in Tab. 6. Publicly available pretrained models are used considering pretraining these models from scratch may result in suboptimal performance. To avoid the influence of different multimodal data processing methods, the comparison is based on the RGB modality. C3D [19], I3D [4], SlowFast [5], and our method are based on 3D CNNs, and the input sizes of the temporal dimension in the experiments are listed in the 'Input' column. Different from 3D CNN-based methods, which extract spatial and temporal features simultaneously via 3D convolutions, the TSN [22] and TPN-TSM [24] model the spatial and temporal information separately. They segment the video and sample one frame from each segment. 2D CNNs are then used to extract spatial features from the sampled frames, and the temporal features are represented by the optical flow as in TSN [22] or learned from the temporal context as in TPN-TSM [24]. The results show the performance of TSN [22] and TPN-TSM [24] is not as good as that of 3D CNN-based methods on the LD-ConGR dataset. This is mainly because gestures with long duration may lose key frames in the segmentation and random sampling. The 3D CNNs show strong ability in extracting spatiotemporal features. It is worth noting that our model achieves 94.66% accuracy (higher than all other methods), which proves its superiority.

Model	Input	Pretrain	Acc./%
C3D [19]	32-frame	-	88.32
I3D [4]	32-frame	Kinetics400 [4]	90.11
SlowFast [5]	64-frame	Kinetics400 [4]	93.51
TSN [22]	8-seg	Kinetics400 [4]	86.80
TPN-TSM [24]	8-seg	Sth-V1 [7]	87.45
Ours	32-frame	Jester [12]	**94.66**

Table 6. Results of representative methods on the RGB modality of LD-ConGR. 'Sth-V1' means Something-Something V1 dataset.

5. Conclusion

In this paper, we present a large RGB-D video dataset LD-ConGR. It is the first dataset targeted at long-distance continuous gesture recognition. LD-ConGR contains high-quality video data and fine-grained annotations, including the gesture category, temporal segmentation, and hand location. In contrast to the existing gesture dataset, the LD-ConGR captures gestures at long distances ($1m \sim 4m$), and the gesture duration varies in a wide range (from 4 frames to 92 frames). Two strategies, gesture region estimation and key frame sampling, are proposed to deal with long-distance gesture recognition and the uncertainty of gesture duration. Moreover, representative methods of gesture and action recognition are evaluated and discussed on the LD-ConGR. We believe that our dataset and experimental studies can inspire research in many fields, including but not limited to gesture recognition, action recognition, and human-computer interaction.

References

[1] Gibran Benitez-Garcia, Jesus Olivares-Mercado, Gabriel Sanchez-Perez, and Keiji Yanai. Ipn hand: A video dataset and benchmark for real-time continuous hand gesture recognition. In *ICPR*, pages 4340–4347. IEEE, 2021. 1, 2, 3, 5

[2] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Using convolutional 3d neural networks for user-independent continuous gesture recognition. In *ICPR*, pages 49–54. IEEE, 2016. 1, 3

[3] Congqi Cao, Yifan Zhang, Yi Wu, Hanqing Lu, and Jian Cheng. Egocentric gesture recognition using recurrent 3d convolutional neural networks with spatiotemporal transformer modules. In *ICCV*, pages 3783–3791, 2017. 1, 3

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 3, 8

[5] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 8

[6] Harshala Gammulle, Simon Denman, Sridha Sridharan, and Clinton Fookes. TMMF: temporal multi-modal fusion for single-stage continuous gesture recognition. *IEEE Trans. Image Process.*, 30:7689–7701, 2021. 3

[7] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *ICCV*, pages 5842–5850, 2017. 8

[8] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. Video-based sign language recognition without temporal segmentation. In *AAAI*, 2018. 3

[9] Hamid Reza Vaezi Joze, Amirreza Shaban, Michael L Iuzzolino, and Kazuhito Koishida. Mmtm: Multimodal transfer module for cnn fusion. In *CVPR*, pages 13289–13299, 2020. 2, 3, 5, 6

[10] Okan Köpüklü, Ahmet Gunduz, Neslihan Kose, and Gerhard Rigoll. Real-time hand gesture detection and classification using convolutional neural networks. In *IEEE Int. Conf. Automatic Face & Gesture Recog.*, pages 1–8. IEEE, 2019. 1, 3

[11] Okan Kopuklu, Neslihan Kose, and Gerhard Rigoll. Motion fused frames: Data level fusion strategy for hand gesture recognition. In *CVPRW*, pages 2103–2111, 2018. 3

[12] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The jester dataset: A large-scale video dataset of human gestures. In *ICCVW*, pages 0–0, 2019. 1, 2, 5, 8

[13] Qiguang Miao, Yunan Li, Wanli Ouyang, Zhenxin Ma, Xin Xu, Weikang Shi, and Xiaochun Cao. Multimodal gesture recognition based on the resc3d network. In *ICCVW*, pages 3047–3055, 2017. 3

[14] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz. Hand gesture recognition with 3d convolutional neural networks. In *CVPRW*, pages 1–7, 2015. 3

[15] Pavlo Molchanov, Xiaodong Yang, Shalini Gupta, Kihwan Kim, Stephen Tyree, and Jan Kautz. Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network. In *CVPR*, pages 4207–4215, 2016. 1, 2, 3, 5

[16] Pradyumna Narayana, Ross Beveridge, and Bruce A Draper. Gesture recognition: Focus on the hands. In *CVPR*, pages 5235–5244, 2018. 1, 3

[17] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. Fingerspelling recognition in the wild with iterative visual attention. In *ICCV*, pages 5400–5409, 2019. 3

[18] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014. 1, 3

[19] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 8

[20] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pages 6450–6459, 2018. 3

[21] Jun Wan, Yibing Zhao, Shuai Zhou, Isabelle Guyon, Sergio Escalera, and Stan Z Li. Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition. In *CVPRW*, pages 56–64, 2016. 1, 2, 5, 6

[22] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, pages 20–36. Springer, 2016. 8

[23] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, pages 1492–1500, 2017. 2, 3, 5, 6

[24] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal pyramid network for action recognition. In *CVPR*, 2020. 8

[25] Bowen Zhang, Limin Wang, Zhe Wang, Yu Qiao, and Hanli Wang. Real-time action recognition with enhanced motion vector cnns. In *CVPR*, pages 2718–2726, 2016. 3

[26] Liang Zhang, Guangming Zhu, Lin Mei, Peiyi Shen, Syed Afaq Ali Shah, and Mohammed Bennamoun. Attention in convolutional lstm for gesture recognition. In *NeurIPS*, pages 1957–1966, 2018. 3

[27] Yifan Zhang, Congqi Cao, Jian Cheng, and Hanqing Lu. Egogesture: a new dataset and benchmark for egocentric hand gesture recognition. *IEEE TMM*, 20(5):1038–1050, 2018. 1, 2, 3, 5

[28] Yifan Zhang, Lei Shi, Yi Wu, Ke Cheng, Jian Cheng, and Hanqing Lu. Gesture recognition based on deep deformable 3d convolutional neural networks. *PR*, 107:107416, 2020. 1, 3