IPN Hand: A Video Dataset and Benchmark for Real-Time Continuous Hand Gesture Recognition

Gibran Benitez-Garcia [*], Jesus Olivares-Mercado[†], Gabriel Sanchez-Perez [†] and Keiji Yanai[*]
[*]Department of Informatics, The University of Electro-Communications, Tokyo, Japan
Email: gibran@ieee.org, yanai@cs.uec.ac.jp
[†]Instituto Politecnico Nacional, ESIME Culhuacan, Mexico City, Mexico
Email: jolivares@ipn.mx, gasanchezp@ipn.mx

Abstract*—Continuous hand gesture recognition (HGR) is an essential part of human-computer interaction with a wide range of applications in the automotive sector, consumer electronics, home automation, and others. In recent years, accurate and efficient deep learning models have been proposed for HGR. However, in the research community, the current publicly available datasets lack real-world elements needed to build responsive and efficient HGR systems. In this paper, we introduce a new benchmark dataset named IPN Hand with sufficient size, variety, and real-world elements able to train and evaluate deep neural networks. This dataset contains more than 4,000 gesture samples and 800,000 RGB frames from 50 distinct subjects. We design 13 different static and dynamic gestures focused on interaction with touchless screens. We especially consider the scenario when continuous gestures are performed without transition states, and when subjects perform natural movements with their hands as non-gesture actions. Gestures were collected from about 30 diverse scenes, with real-world variation in background and illumination. With our dataset, the performance of three 3D-CNN models is evaluated on the tasks of isolated and continuous real-time HGR. Furthermore, we analyze the possibility of increasing the recognition accuracy by adding multiple modalities derived from RGB frames, i.e., optical flow and semantic segmentation, while keeping the real-time performance of the 3D-CNN model. Our empirical study also provides a comparison with the publicly available nvGesture (NVIDIA) dataset. The experimental results show that the state-of-the-art ResNext-101 model decreases about 30% accuracy when using our real-world dataset, demonstrating that the IPN Hand dataset can be used as a benchmark, and may help the community to step forward in the continuous HGR. Our dataset and pre-trained models used in the evaluation are publicly available at github.com/GibranBenitez/IPN-hand.

I. Introduction

Gestures are a natural form of human communication [1]. Hand gesture recognition (HGR) is an essential part of human-computer interaction. Systems using vision-based interaction and control are more common nowadays [2]–[4]. Compared to traditional inputs such as mouse and keyboard, vision-based interfaces can be more practical and natural based on the intuitiveness of designed gestures. HGR has a wide range of applications in the automotive sector, consumer electronics, public transit, gaming, home automation, and others [4]–[6]. For these applications, HGR systems must be designed to work online and deal with continuous gestures that users may input. There are mainly four issues that systems must deal with for continuous HGR applications: (i) continuous gestures without transition states, (ii) natural behaviors of users' hands similar to target gestures, (iii) intra-class variability of gestures' duration, and (iv) lag between performing a gesture and its classification. It is worth mentioning that (i) is particularly important for some real-world applications such as interacting with touchless screens. For example, someone can open a picture and zoom in, by using two continuous gestures (double-click + zoom-in) without a transition state that forces the hand returning to a neutral position before the second gesture. This example is shown in Figure 1(a).

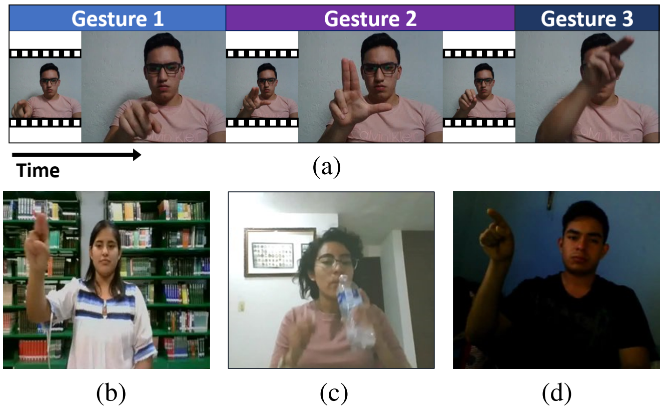


Fig. 1. Some examples that show the challenges of our dataset. (a) Continuous gestures without transition states. (b) Clutter backgrounds. (c) Natural interaction with objects. (d) Weak illumination conditions.

Thanks to the advances on deep neural networks, in recent years, accurate and efficient deep learning models have been proposed to overcome the challenges on continuous HGR [7]–[11]. However, these methods were evaluated using datasets that do not cover the main real-world issues. Currently, it is not easy to find a benchmark dataset able to evaluate the four main issues of continuous HGR. Most of the hand gesture dataset used for continuous HGR, like ChaLearn ConGD [12], nvGesture [7] and EgoGesture [13], do not include continuous gestures without transition states (i), nor natural hand movements as non-gesture actions (ii). To the best of our knowledge, there a no publicly available hand gesture datasets that cover these two issues of continuous HGR. Note that, some works have designed specific datasets for controlling automotive interfaces [14], [15], that partially include these issues. However, the datasets are not publicly available.

TABLE I
COMPARISON OF THE PUBLIC CONTINUOUS GESTURE DATASETS

Dataset	Instances	Videos	Instance/video	Classes	Subjects	Scenes	View	Modalities
ChaLearn ConGD, 2016 [12]	47,933	22,535	2.1	249	21	15	3rd	RGB-D
nvGesture, 2016 [7]	1,532	1,532	1.0	25	20	1	3rd	RGB-D
EgoGesture, 2018 [13]	24,161	2,081	11.6	83	50	6	1st	RGB-D
IPN Hand (ours)	4,218	200	21.1	13	50	28	3rd	RGB

In this paper, we introduce a new dataset called IPN Hand for the task of continuous hand gesture recognition. The dataset contains more than four thousand RGB gesture samples and 800 thousand frames from 50 distinct subjects. We design 13 classes of static and dynamic gestures for interaction with touchless screens. Our dataset has the most realistic scenario for continuous HGR than other hand gesture datasets. IPN Hand includes the largest number of continuous gestures per video and the largest speed of intra-class variation for different subjects when they were performing the same gesture. Besides, our dataset is more complex as it was collected from about thirty representative scenes with considerable variation, including clutter backgrounds, strong and weak illumination conditions, static and dynamic background environments. We specially design two scenarios that reflects the real-world issues of continuous HGR: when continuous gestures are performed without transition states, and when subjects perform natural movements with their hands as non-gesture actions. Some examples of the main challenges of our dataset are shown in Figure 1.

Given our dataset, we evaluate three state-of-the-art (SOTA) methods based on 3D-CNN models for the tasks of isolated and continuous real-time HGR. Furthermore, we analyze the possibility of increasing the recognition accuracy by adding multiple modalities derived from RGB frames. Specifically, we evaluate the data level fusion of RGB with semantic segmentation of target hands as an alternative of the RGB+Optical Flow or RGB+Depth modalities for real-time HGR. Our empirical study also provides a comparison with the publicly available nvGesture (NVIDIA) dataset. The experimental results on the IPN Hand dataset demonstrate that it can be used as a benchmark for the data-hungry 3D-CNN methods, which may help the community to step forward in the continuous HGR.

II. RELATED WORKS

A. Datasets for continuous HGR

Existing continuous HGR datasets differ by factors such as scale, the number of classes, sensors used, and the domain of gestures. However, all of them must include non-gesture frames to emulate the online behavior of real applications. These frames are important to define the realism of the designed dataset. Therefore, we analyze the commonly used datasets based on this element.

The ChaLearn LAP ConGD dataset [12] is derived from 9 different gesture domains, from Italian sign language, activities to pantomime. It contains 249 classes and more than 40 thousand instances from 21 subjects, which makes ChaLearn ConGD the largest dataset for continuous HGR. This dataset contains videos with one to five continuous gestures with transition states. The nvGesture dataset [7] is designed to control in-car automotive devices, and it includes 25 gesture types from 20 subjects, consisting of 1 532 instances. Note that this dataset is commonly used for online HGR evaluation, even though it only contains videos with isolated gestures. The EgoGesture dataset [13] is a benchmark dataset for egocentric (first-person) view, which consists of 83 classes with more than 20 thousand instances from 50 subjects. This dataset includes at max 12 instances per video. To the best of our knowledge, our proposed IPN Hand dataset contains the largest number of instances per video (21), and different scenes (28), making it suitable for evaluating continuous HGR systems. Detailed comparison between our IPN Hand and related gesture datasets can be found in Table I.

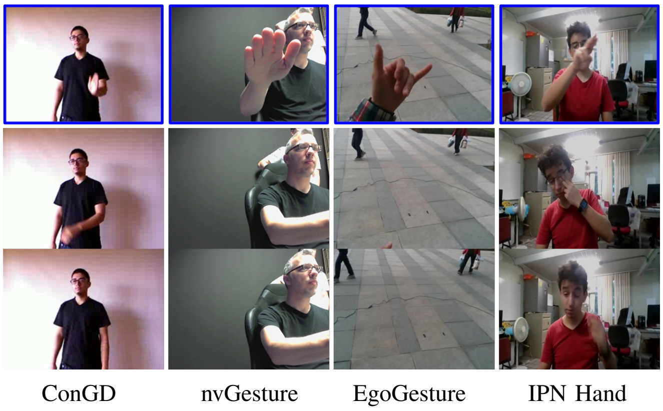


Fig. 2. Comparison of gesture vs. non-gesture frames from the public continuous gesture datasets. Note that only the first row (blue) shows examples of gesture frames.

Figure 2 shows the comparison of the non-gesture frames from the public continuous gesture datasets. Most of the non-gesture frames from the ChaLearn ConGD dataset consist of transition states between gestures, usually showing the subject in a neutral position as in the first column of the figure. On the other hand, the nvGesture shows drivers with their hands at the steering wheel, while EgoGesture shows only backgrounds with the hands out of view as non-gesture frames. It is clear that our dataset presents more challenge on distinguishing gesture vs. non-gesture frames, since we include subjects performing natural movements with their hands.

B. Continuous HGR

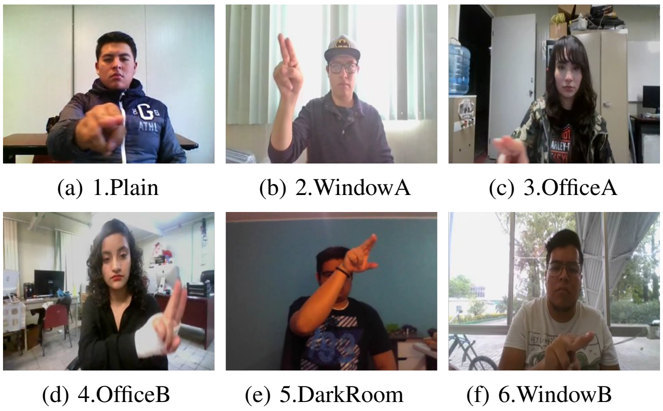
Continuous HGR can be divided into two stages: gesture spotting and gesture classification. Gesture spotting aims to detect temporal video segments which contain gesture instances, while the classification stage aims to classify the gesture of each spotted segment. Multi-stream architectures have been widely employed for both tasks. Simonyan et al. [16] were the pioneers of fusing features from two modalities, using one stream with RGB images and the other with flow fields for isolated HGR. On This multi-modality approach is prevalent as well for continuous HGR, as shown in the 2017 ChaLearn Look At People (LAP) continuous gesture detection (ConGD) challenge [17], where all entries used multi-stream architectures of at least RGB and depth. For instance, the winners [8] introduced a two-stream 3D-CNN combining hand-location features of RGB and depth modalities by explicitly overlaying a black mask on the input frames. They firstly spotted the gestures based on a dataset-specific observation: subjects raise their hands at the beginning of gestures and put them down again at the end. One year after ConGD challenge, Zhu et al. [9] overcame the results of the winners by proposing a temporal dilated 3D-CNN architecture to binary classify gesture/non-gesture frames, and 3D-CNN+LSTM for gesture classification. The current SOTA method of ConGD uses a 12-channels architecture with extra RNN layers to simultaneously spot and classify continuous gestures [10]. Recently, Kopuklu et al. [11] proposed a hierarchical structure of 3D-CNN architectures to detect and classify continuous hand gestures. Their spotting method consists of detecting gesture frames using a shallow 3D-CNN model on eight consecutive frames, while a SOTA model is used for classification only if a gesture is detected.

To evaluate our dataset for real-world applications of continuous HGR, detection and classification must work online or even with a negative lag between performing a gesture and its classification feedback. Therefore, we follow the hierarchical structure of 3D-CNNs [11], since it can detect a gesture when a confidence measure reaches a threshold level before the gesture ends (early-detection). Furthermore, we evaluate the multi-modality accuracy derived from RGB frames, by keeping in mind the real-time performance. We employ a data level fusion to avoid the significant increase in the computational cost of the 3D-CNN models. Besides, we propose to use semantic segmentation results as an alternative to the absent depth modality and the computational expensive optical flow.

III. The IPN Hand Dataset

A. Data collection

To collect the dataset, the RGB videos were recorded in the resolution of 640 × 480 with the frame rate of 30 fps. The participants were asked to record the gestures using their own PC or laptop by keeping the defined resolution and frame rate. Thus, the distance between the camera and each subject varies, since we instructed all participants to be located in a comfortable position to manipulate the screen of their PC with hand gestures. In this way, the videos were generated with 28 different scenes in total. For some participants that do not have access to a camera that can cover the recording specifications, we prepared three scenes (scene 2-4, shown in Figure 3(b)-(d)), including clutter and variable illumination backgrounds. When collecting data, we first teach the subjects how to perform each gesture and tell them the gesture names (short descriptions). Then we generate four gesture name lists with random order for recording four videos per subject. Thus, the subject was told the gesture name and performed the gesture accordingly. They were asked to continuously perform 21 gestures with 3 random breaks as a single session which was recorded as a video.


(a) 1.Plain (b) 2.WindowA (c) 3.OfficeA

(d) 4.OfficeB (e) 5.DarkRoom (f) 6.WindowB

Fig. 3. Some examples of the scenes with more videos in the dataset. Note that all subjects are developing the same gesture, ”pointing with two fingers”.

B. Dataset characteristics

1) Gesture Classes: We design the gestures in our dataset focused on interaction with touchless screens. We include gestures able to control the location of the pointer on the screen (pointer), and to manipulate interfaces (actions). We take some gestures used in common smartphones, as they are natural and easy to remember by the users. Thus, we designed 13 gestures (shown in Figure 4) defined to control the punter and actions.

For the punter location, we defined two static gestures of pointing to the screen with one and two fingers, respectively. Note that these gestures are static due to the gesture itself does not need temporal information to be detected. However, the application of this gesture includes the hand movement to control the pointer. For the action gestures, we defined 11 gestures, including click with one and two fingers, throw to four positions (up, down, left & right), double click with one and two fingers, zoom-in, and zoom-out. Table II provides the name and descriptions of each gesture in our dataset. Note that we also include segments were natural hand movements are performed as non-gestures states.

2) Subjects: A small number of subjects could make the intra-class variation very limited. Hence, we asked 50 subjects for our data collection. In the 50 subjects, there are 16 females and 34 males. The average age of the subjects is 21.2, where the minimum age is 18, and the maximum age is 25. All of the subjects are currently students at Instituto Politecnico from Mexico.

3) Extreme illumination and clutter backgrounds: To evaluate the robustness to the illumination change of the baseline

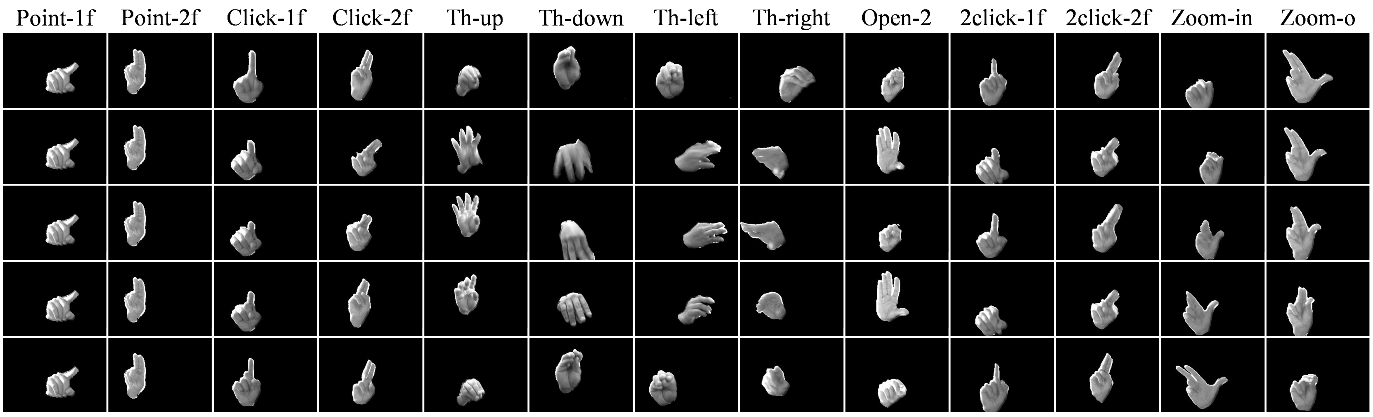


Fig. 4. Examples of the 13 gesture classes included in our dataset. For visualization purposes, semantic segmentation masks were blended to the RGB images.

TABLE II
STATICS PER GESTURE OF OUR IPN HAND DATASET

id	Gesture	Name	Instances	Mean duration (std)	Duration $\overline{nstd_k}$
0	No gesture	No-gest	1431	147 (133)	0.904
1	Pointing with one finger	Point-1f	1010	219 (67)	0.308
2	Pointing with two fingers	Point-2f	1007	224 (69)	0.309
3	Click with one finger	Click-1f	200	56 (29)	0.517
4	Click with two fingers	Click-2f	200	60 (43)	0.718
5	Throw up	Th-up	200	62 (25)	0.400
6	Throw down	Th-down	201	65 (28)	0.424
7	Throw left	Th-left	200	66 (27)	0.400
8	Throw right	Th-right	200	64 (28)	0.439
9	Open twice	Open-2	200	76 (31)	0.410
10	Double click with one finger	2click-1f	200	68 (28)	0.412
11	Double click with two fingers	2click-2f	200	70 (30)	0.435
12	Zoom in	Zoom-in	200	65 (29)	0.440
13	Zoom out	Zoom-o	200	64 (28)	0.432

methods, we have data collected under extreme conditions such as facing a window with strong sunlight (Figure 3(b)), or in a room with almost null artificial light (Figure 1(d)). Some scenes with static background placed with student-life stuff (Figure 1(b)), and dynamic background with walking people appearing in the camera view (Figure 3(d)-(f)) were also included.

C. Dataset statics

Fifty distinct subjects participated in performing 13 classes of gestures in 28 diverse scenes. Totally 4,218 gesture instances and 800,491 frames were collected in RGB. Figure 5 illustrates the sample distribution on each video among the 28 scenes. In the figure, the horizontal axis and the vertical axis indicate the ID of the scenes and the number of the videos, respectively. In addition, Figure 3 shows examples of the six scenes with more videos on our dataset.

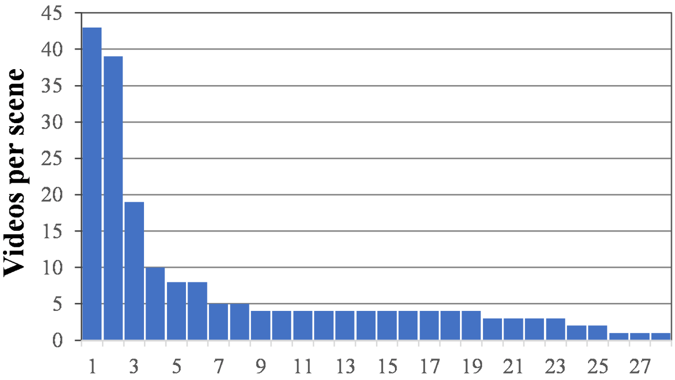


Fig. 5. Distribution of the 28 different scenes included in the dataset.

When data collection, 21 gestures are considered as a session and recorded in a single video. Thus, we get 200 RGB videos in total. Note that the order of the gestures performed is semi-randomly generated, trying to include one dynamic after one static gesture, which resembles the realistic interaction with a touchless screen. The start and end frame index of each gesture instance in the video were manually labeled, which provides enough information to train and evaluate continuous HGR approaches in a fully supervised manner. In the dataset, the minimum length of a gesture is 9 frames. The maximum length of a gesture is 650 frames.

In Table III, we show the data statistics of our dataset compared with the continuous HGR datasets that are currently publicly available. The dataset statistics include the number of total frames, the mean video durations, the mean of the gesture sample durations, the standard deviation of gesture durations, the percentage of the training data. The mean and standard deviation of the gesture sample durations is calculated over the samples from all gesture classes in the dataset. Following [13], we use the normalized standard deviation

TABLE III
STATICS OF THE PUBLIC CONTINUOUS GESTURE DATASETS

Dataset	Frames	Mean video duration	Mean gesture duration	Duration std	Duration $\overline{nstd_k}$	% of train
ChaLearn ConGD, 2016 [12]	1,714,629	76.1	41	18.5	0.37	0.635
nvGesture, 2016 [7]	122,560	80.0	71	32.3	0.3	0.685
EgoGesture, 2018 [13]	2,953,224	1,419.1	38	13.9	0.33	0.595
IPN Hand (ours)	800,491	4,002.5	140	93.9	0.43	0.739

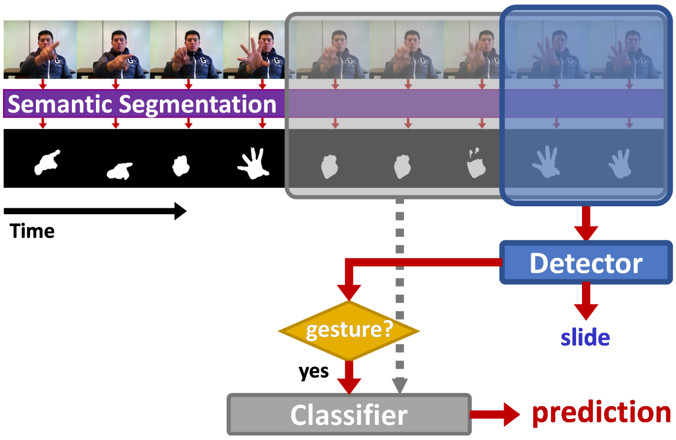


Fig. 6. The flowchart of the continuous HGR approach based on two hierarchical 3D-CNNs models.

for gesture duration in each gesture class to describe the speed variation of different subjects when performing the same gesture. The normalized standard deviation of durations in a gesture class k is calculated as follows:

$$nstd_k = \frac{1}{\bar{l^k}}\sqrt{\frac{\sum_i^N (l_i^k - \bar{l^k})^2}{N}} \quad (1)$$

where in gesture class k, l_i^k represents the duration of the ith sample, $\bar{l^k}$ is the average duration of samples, and N is the number of samples. We get the average $\overline{nstd_k}$ over all gesture classes, for the whole dataset.

From Table III, we can see that our IPN Hand datset has the largest duration $\overline{nstd_k}$ (0.43), significantly larger than EgoGesture, which is the second largest. This demonstrates that our dataset has a large speed variation for different subjects when performing the same gesture. In resume, our proposed dataset covers all the real-world issues for continuous HGR, as described in the introduction: (i) continuous gestures without transition states, (ii) natural behaviors of users' hands similar to target gestures, (iii) intra-class variability of gestures' duration ($\overline{nstd_k}$).

IV. Benchmark Evaluation

We evaluate three SOTA 3D-CNN models as baselines for the tasks of isolated and continuous HGR, with our new IPN Hand dataset. For continuous HGR, we adopt a two-model hierarchical architecture to detect and classify the continuous input stream of frames.

A. *Hierarchical 3D-CNNs for continuous HGR*

As mentioned before, we use the framework proposed by Kopuklu et al. [11] to detect and classify continuous hand gestures. The flowchart of the two-model hierarchical architecture is shown in Figure 6. Sliding windows with a constant stride run through incoming video frames where the detector queue is placed at the very beginning of the classifier queue. If the detector recognizes a gesture, then the classifier is activated. The detector's output is post-processed for more robust performance, and the final decision is made using a single-time activation block where only one activation occurs per performed gesture. Note that the sliding window of the detector is the only who is working every single stride. In contrast, the classification window is activated based on detectors output, so the inference time is not always the sum of both models.

Since we also evaluate multi-modal 3D-CNNs models, we first calculate the semantic segmentation masks of each streaming frame. For this task, we use the efficient semantic segmentation approach of HarDNet (Harmonic Dense Net) [18] which achieves SOTA results with a network inspired by DenseNet [19]. Its core component, the *HarDBlock* (Harmonic Dense Block), is specifically designed to address the problem of the GPU memory traffic. Therefore, we can achieve real-time performance on each frame before applying 3D-CNNs models.

B. *Experimental setup*

We randomly split the data by subject into training (74%), and testing (26%) sets, resulting in 148 training, and 52 testing videos. The numbers of gesture instances in training, and testing splits are 3 117, and 1 101, respectively. Thus, 37 and 13 subjects were designed for training and testing, respectively.

1) Implementation details: We compare the 3D-CNN models of C3D [20], and the 3D versions [21] of Resnet-50 [22] and ResNeXt-101 [23], as deep models for classification. On the other hand, for the shallow 3D-CNN detector we use the ResLight (Resnet-10) as proposed in [11]. All 3D-CNN models were trained using a fully supervised strategy using the manually segmented gestures.

For the multi-modality tests, we train a HarDNet [18] model, and use the SPyNet [24] approach for calculation of semantic segmentation and optical flow, respectively. We trained the HarDNet model with a synthetic hand pose estimation dataset [25] which contains more than 40 thousand images with hands fully annotated at pixel-level[1]. For the SPyNet, we

[1] https://lmb.informatik.uni-freiburg.de/projects/hand3d/

TABLE IV
RESULTS OF ISOLATED HGR TASK USING OUR IPN HAND DATASET

Model	Input sequence	Modality	Results	Parameters	Model Size	Inference time
C3D	32-frames	RGB	77.75	50.75 M	387 MB	76.2 ms
ResNeXt-101	32-frames	RGB	83.59	47.51 M	363 MB	27.7 ms
ResNeXt-101	32-frames	RGB-Flow	**86.32**	47.56 M	363 MB	28.9 ms
ResNeXt-101	32-frames	RGB-Seg	84.77	47.56 M	363 MB	28.9 ms
ResNet-50	32-frames	RGB	73.1	**46.25 M**	**353 MB**	**17.8 ms**
ResNet-50	32-frames	RGB-Flow	74.65	46.27 M	**353 MB**	18.2 ms
ResNet-50	32-frames	RGB-Seg	75.11	46.27 M	**353 MB**	18.2 ms

TABLE V
COMPARISON OF THE EXTRA PROCESSES FOR MULTI-MODALITY MODELS.

Process	Model	Params.	Model size	Inference time
Segmentation	HarDNet	4.114 M	15.8 MB	**8.1 ms**
Optical Flow	SPyNet	**1.440 M**	**5.50 MB**	21.9 ms

used the open-source implementation and pre-trained model of [26] to calculate real-time densely optical flow from each input frame.

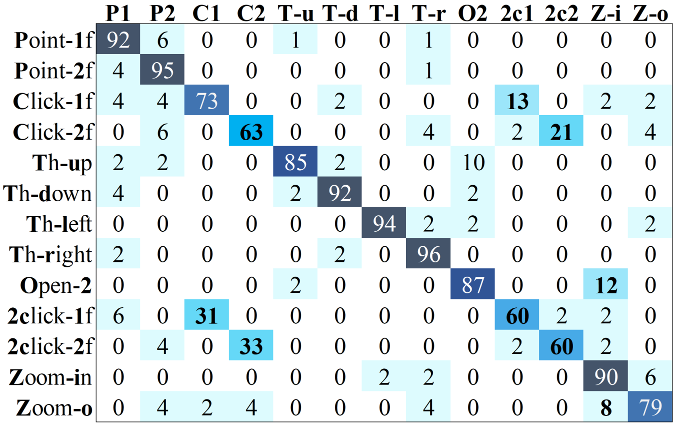
All 3D-CNN models were pre-trained on the Jester dataset [27], while the HarDNet on the ImageNet dataset [28]. Contrary, since ResLight is a compact model, we trained it from scratch using all non-gesture and gesture instances of the IPN Hand training set. The inference time (FPS) was measured on an Intel Core i7-9700K desktop with a single NVIDIA GTX 1080ti GPU. More implementation details, as well as training and evaluation codes can be found in our open-source repository[2]

C. Isolated HGR task

We evaluate this task with the conventional classification metric. We segment the video sequences into isolated gesture samples based on the beginning and ending frames manually annotated. The learning task is to predict class labels for each gesture sample. We use classification accuracy, which is the percent of correctly labeled examples, and the confusion matrix of the predictions, as evaluation metrics for this learning task.

1) Experimental results using the IPN Hand dataset: Table IV presents the results of evaluated models with different modalities using our IPN Hand dataset. As expected, the best results were obtained by the ResNeXt-101 model, which can barely achieve real-time performance with the different modalities. However, it is clearly faster and more accurate than the robust C3D model. On the other hand, ResNet-50 is the most efficient model, but present the lowest accuracy results among the evaluated approaches.

It is worth noting that our RGB-seg achieves competitive results compared to the RGB-flow, which is significant since the optical flow process is more computationally expensive. To evaluate the efficiency of the multi-modal approaches, Table V shows the inference time and the model size of extra-processes related to semantic segmentation and optical flow estimation.



	P1	P2	C1	C2	T-u	T-d	T-l	T-r	O2	2c1	2c2	Z-i	Z-o
Point-1f	92	6	0	0	1	0	0	1	0	0	0	0	0
Point-2f	4	95	0	0	0	0	0	1	0	0	0	0	0
Click-1f	4	4	73	0	0	2	0	0	0	**13**	0	2	2
Click-2f	0	6	0	**63**	0	0	0	4	0	2	**21**	0	4
Th-up	2	2	0	0	85	2	0	0	10	0	0	0	0
Th-down	4	0	0	0	2	92	0	0	2	0	0	0	0
Th-left	0	0	0	0	0	0	94	2	2	0	0	0	2
Th-right	2	0	0	0	0	2	0	96	0	0	0	0	0
Open-2	0	0	0	0	2	0	0	0	87	0	0	**12**	0
2click-1f	6	0	**31**	0	0	0	0	0	0	**60**	2	2	0
2click-2f	0	4	0	**33**	0	0	0	0	0	2	**60**	2	0
Zoom-in	0	0	0	0	0	0	2	2	0	0	0	90	6
Zoom-o	0	4	2	4	0	0	0	4	0	0	0	**8**	79

Fig. 7. Confusion matrix of the best result obtained from ResNext-101 model with RGB-flow.

TABLE VI
RESULTS OF ISOLATED HGR TASK USING THE NVGESTURE DATASET

Model	Input sequence	Modality	Results
ResNeXt-101	32-frames	RGB	79.46
ResNeXt-101	**32-frames**	**RGB-Flow**	**82.36**
ResNeXt-101	32-frames	RGB-Seg	82.15

From this table, we can see that the semantic segmentation is more than two times faster than the optical flow, making the RGB-seg alternative feasible for real-time applications.

In addition, in the Figure 7, we show the confusion matrix of the best result obtained for isolated HGR, ResNext-101 with RGB-flow. As expected, the problems are related to the gestures that are closer such as clicks with double clicks.

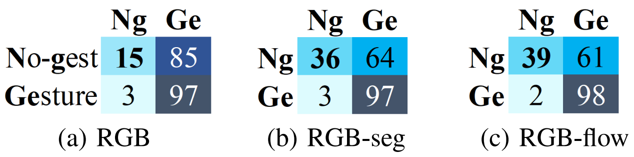
2) Experimental results using the nvGesture dataset: As mentioned before, we also evaluate the performance of our RGB-seg alternative with a common online HGR dataset. Table VI presents the results of the evaluated models with different modalities using the nvGesture dataset. From these results, we can notice that the trend of the multi-modal results from IPN Hand is still valid (RGB-flow $>$ RGB-seg $>$ RGB).

In addition, Table VII shows the result of our real-time RGB-seg alternative compared to SOTA methods of nvGesture dataset. From this table, we validate the use of semantic segmentation as an important source of valuable features for HGR. The results of ResNeXt-101 with RGB-seg, are competitive to SOTA methods that even employ different modalities, such as depth. These findings are significant due to the semantic segmentation process only takes 8 ms with an

[2] https://github.com/GibranBenitez/IPN-hand

TABLE VII
COMPARISION WITH SOTA METHODS OF NVGESTURE DATASET.

Model	Modality	Results
C3D [20]	RGB	73.8
R3DCNN [7]	RGB	74.1
MTUT [29]	RGB*	81.3
R3DCNN [7]	RGB+Flow	79.3
MFF [30]	**RGB+Flow**	**84.7**
R3DCNN [7]	Depth+Flow	82.4
ResNeXt-101	RGB+Seg	82.2



	Ng	Ge
No-gest	15	85
Gesture	3	97

(a) RGB

	Ng	Ge
Ng	36	64
Ge	3	97

(b) RGB-seg

	Ng	Ge
Ng	39	61
Ge	2	98

(c) RGB-flow

Fig. 8. Confusion matrices of gesture detection on the IPN hand dataset using different modalities for the Resnet-Light model.

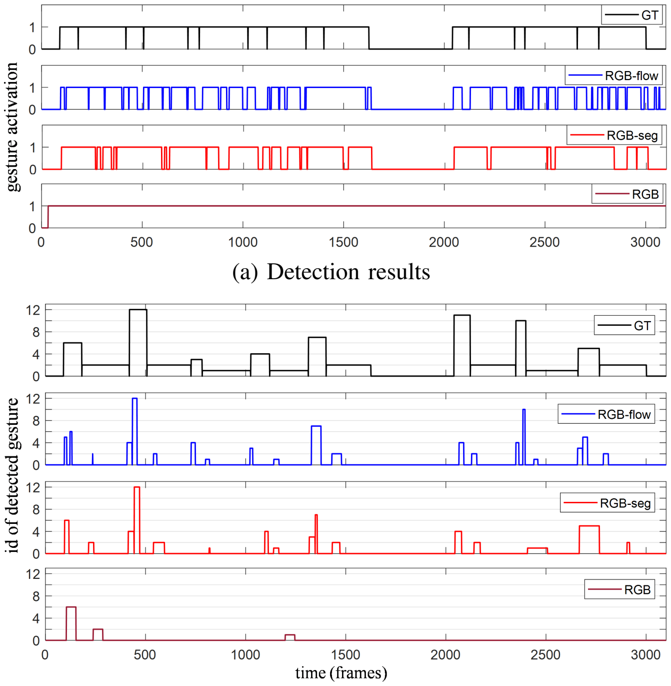
input image of 360 × 240 pixels.

D. Continuous HGR task

We use the Levenshtein accuracy [11] as evaluation metric for the continuous HGR taks. This metric employs the Levenshtein distance to measure the distance between sequences by counting the number of item-level changes (insertion, deletion, or substitutions) to transform one sequence into the other. In the case of continuous hand gestures, the difference between the sequence of predicted and ground truth gestures is measured. For example, if a ground truth sequence is $[1, 2, 3, 4, 5, 6, 7, 8, 9]$ and predicted gestures of a video is $[1, 2, 7, 4, 5, 6, 6, 7, 8, 9]$, the Levenshtein distance is 2. Defined by the deletion of one of the ”6” which is detected two times, and the substitution of ”7” with ”3”. Thus, the Levenshtein accuracy is obtained by averaging this distance over the number of true target classes. In our example, the accuracy is $1 - (2/9) \times 100 = 77.78\%$.

We obtain the average Levenshtein accuracy over the 52 testing videos to asses the continuous HGR performance. Besides, we also evaluate the detection accuracy of the 3D-CNN detectors with different multi-modalities. Note that detectors are trained and evaluated using isolated gesture vs. non-gesture samples. So that we use a binary classification accuracy, and the confusion matrix of the predictions, as evaluation metrics for these models.

1) Experimental results: First, we evaluate the detector model with the different modalities in Table VIII. As expected, the same trend is maintained, but since ResLight-10 is a much more compact model, the inference time is not affected. It is also important to analyze the misclassification of the detector, therefore Figure 8 shows the confusion matrices from each modality. It is clear that the benefits from multi-modal approaches are reflected in the detection of non-gesture frames, since the RGB approach misrecognized 85% of these frames.



(a) Detection results

(b) Classification results

Fig. 9. Qualitative results of the proposed networks. The most significant improvements occur on pixels belonging to large objects.

We evaluate the complete process of the hierarchical two-model approach using Resnet-50 and ResNeXt-101 as classifiers, and ResLight-10 as a detector. In addition, we also analyze the multi-modal alternatives covering real-time performance. Table IX shows the Levenshtein accuracy, as well as the model size and inference time of each evaluated model and modality. The results of this test make more evident the advantages of using the RGB-seg alternative for real-time continuous HGR. For instance, the inference time results of ResNeXt-101 RGB are comparable with those of ResNet-50 RGB-seg. However, the Levenshtein accuracy of the latter is significantly better (around 10% of improvement). Finally, we show the temporal visualization of continuous predictions from one testing video in Figure 9.

V. CONCLUSION

In this paper, we introduced a new benchmark dataset for continuous HGR that includes real-world issues, such as continuous gestures without transition states, natural behaviors of users’ hands, and large intra-class variability of gestures’ duration. Besides, we evaluate the data level fusion of RGB with semantic segmentation as an alternative of the RGB+Optical Flow or RGB+Depth modalities for real-time HGR. From the experimental results, we conclude that RGB-seg is a suitable multi-modal alternative for real-time continuous hand gesture recognition. Furthermore, we believe that the proposed dataset can be used as a benchmark and help the community to move steps forward on the continuous HGR.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number 15H05915, 17H01745, 17H06100 and 19H04929.

TABLE VIII
RESULTS OF THE DETECTOR MODEL USING OUR IPN HAND DATASET

Model	Input sequence	Modality	Results	Parameters	Model Size	Inference time
ResLight-10	8-frames	RGB	75.4	**0.895 M**	**6.83 MB**	**2.4 ms**
ResLight-10	8-frames	RGB-Flow	**82.43**	0.908 M	6.94 MB	2.9 ms
ResLight-10	8-frames	RGB-Seg	80.06	0.908 M	6.94 MB	2.9 ms

TABLE IX
RESULTS OF THE HIERARCHICAL TWO-MODEL APPROACH FOR CONTINUOUS HGR USING OUR IPN HAND DATASET

			Model size			Inference time		
Model	Modality	Results	detector	classifier	total	detector	classifier	total
ResNeXt-101	RGB	25.34	**6.83 MB**	**363 MB**	**370 MB**	**2.9 ms**	**27.7 ms**	**30.1 ms**
ResNeXt-101	RGB-Flow	**42.47**	12.4 MB	363 MB	375 MB	11.1 ms	28.9 ms	53.7 ms
ResNeXt-101	RGB-seg	39.01	22.7 MB	363 MB	386 MB	24.8 ms	28.9 ms	39.9 ms
Resnet-50	RGB	19.78	**6.83 MB**	**353 MB**	**360 MB**	**2.9 ms**	**17.8 ms**	**20.4 ms**
Resnet-50	RGB-Flow	**39.47**	12.4 MB	353 MB	365 MB	11.1 ms	18.2 ms	43.1 ms
Resnet-50	RGB-seg	33.27	22.7 MB	353 MB	376 MB	24.8 ms	18.2 ms	29.2 ms

REFERENCES

[1] A. Kendon, “Gesticulation and speech: Two aspects of the process of utterance,” *The relationship of verbal and nonverbal communication*, no. 25, p. 207, 1980.

[2] M. Leo, G. Medioni, M. Trivedi, T. Kanade, and G. M. Farinella, “Computer vision for assistive technologies,” *Computer Vision and Image Understanding*, vol. 154, pp. 1–15, 2017.

[3] L. P. Berg and J. M. Vance, “Industry use of virtual reality in product design and manufacturing: a survey,” *Virtual reality*, vol. 21, no. 1, pp. 1–17, 2017.

[4] S. S. Rautaray and A. Agrawal, “Vision based hand gesture recognition for human computer interaction: a survey,” *Artificial intelligence review*, vol. 43, no. 1, pp. 1–54, 2015.

[5] C. A. Pickering, K. J. Burnham, and M. J. Richardson, “A research study of hand gesture recognition technologies and applications for human vehicle interaction,” in *2007 3rd Institution of Engineering and Technology conference on automotive electronics*. IET, 2007, pp. 1–15.

[6] F. Parada-Loira, E. González-Agulla, and J. L. Alba-Castro, “Hand gestures to control infotainment equipment in cars,” in *IEEE Intelligent Vehicles Symposium*, 2014, pp. 1–6.

[7] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, “Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4207–4215.

[8] Z. Liu, X. Chai, Z. Liu, and X. Chen, “Continuous gesture recognition with hand-oriented spatiotemporal feature,” in *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2017, pp. 3056–3064.

[9] G. Zhu, L. Zhang, P. Shen, J. Song, S. Shah, and M. Bennamoun, “Continuous gesture segmentation and recognition using 3dcnn and convolutional lstm,” *IEEE Transactions on Multimedia*, 2018.

[10] P. Narayana, J. R. Beveridge, and B. Draper, “Continuous gesture recognition through selective temporal fusion,” in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[11] O. Köpüklü, A. Gunduz, N. Köse, and G. Rigoll, “Real-time hand gesture detection and classification using convolutional neural networks,” in *14th IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, 2019, pp. 1–8.

[12] J. Wan, Y. Zhao, S. Zhou, I. Guyon, S. Escalera, and S. Z. Li, “Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2016, pp. 761–769.

[13] Y. Zhang, C. Cao, J. Cheng, and H. Lu, “Egogesture: A new dataset and benchmark for egocentric hand gesture recognition,” *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1038–1050, 2018.

[14] G. Benitez-Garcia, M. Haris, Y. Tsuda, and N. Ukita, “Similar finger gesture recognition using triplet-loss networks,” in *Sixteenth IAPR International Conference on Machine Vision Applications (MVA)*. IEEE, 2019, pp. 1–6.

[15] ——, “Finger gesture spotting from long sequences based on multi-stream recurrent neural networks,” *Sensors*, vol. 20, no. 2, p. 528, 2020.

[16] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014, pp. 568–576.

[17] J. Wan, S. Escalera, G. Anbarjafari, H. J. Escalante, X. Baró, I. Guyon, M. Madadi, J. Allik, J. Gorbova, C. Lin *et al.*, “Results and analysis of chalearn lap multi-modal isolated and continuous gesture recognition, and real versus fake expressed emotions challenges,” in *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2017, pp. 3189–3197.

[18] P. Chao, C.-Y. Kao, Y.-S. Ruan, C.-H. Huang, and Y.-L. Lin, “HarDNet: A Low Memory Traffic Network,” in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[19] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[20] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning spatiotemporal features with 3d convolutional networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4489–4497.

[21] K. Hara, H. Kataoka, and Y. Satoh, “Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 6546–6555.

[22] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[23] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492–1500.

[24] A. Ranjan and M. J. Black, “Optical flow estimation using a spatial pyramid network,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4161–4170.

[25] C. Zimmermann and T. Brox, “Learning to estimate 3d hand pose from single rgb images,” in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4903–4911.

[26] S. Niklaus, “A reimplementation of SPyNet using PyTorch,” https://github.com/sniklaus/pytorch-spynet, 2018.

[27] J. Materzynska, G. Berger, I. Bax, and R. Memisevic, “The jester dataset: A large-scale video dataset of human gestures,” in *IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 0–0.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[29] M. Abavisani, H. R. V. Joze, and V. M. Patel, “Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1165–1174.

[30] O. Kopuklu, N. Kose, and G. Rigoll, “Motion fused frames: Data level fusion strategy for hand gesture recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 2184–21 848.