Hand Gesture Recognition From Wrist-Worn Camera for Human–Machine Interaction

Hong-Quan Nguyen, Trung-Hieu Le, Trung-Kien Tran, Hoang-Nhat Tran, Thanh-Hai Tran, Thi-Lan Le, Hai Vu, Cuong Pham, Thanh Phuong Nguyen, Huu Thanh Nguyen

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 29 April 2023, accepted 16 May 2023, date of publication 25 May 2023, date of current version 6 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3279845

RESEARCH ARTICLE

Hand Gesture Recognition From Wrist-Worn Camera for Human–Machine Interaction

HONG-QUAN NGUYEN[1,2], TRUNG-HIEU LE[1,3], TRUNG-KIEN TRAN[4], HOANG-NHAT TRAN[1], THANH-HAI TRAN[1], THI-LAN LE[1], HAI VU[1], CUONG PHAM[5], THANH PHUONG NGUYEN[6], AND HUU THANH NGUYEN[1]

[1]School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi 100000, Vietnam
[2]Faculty of Information Technology, Viet-Hung Industrial University, Hanoi 12712, Vietnam
[3]Faculty of Information Technology, Dainam University, Hanoi 152440, Vietnam
[4]Military Information Technology Institute, Hanoi 11353, Vietnam
[5]Posts and Telecommunications Institute of Technology, Hanoi 100000, Vietnam
[6]LIS UMR 7020, Université de Toulon, Aix Marseille Université, CNRS, 83041 Toulon, France

Corresponding author: Thanh-Hai Tran (hai.tranthithanh1@hust.edu.vn)

This work was supported by the Air Force Office of Scientific Research under Award FA2386-20-1-4053.

ABSTRACT In this work, we study the ability to use hand gestures for human-machine interaction from wrist-worn sensors. Towards this goal, we design a wrist-worn prototype to capture RGB video stream of hand gestures. Then we built a new wrist-worn gesture dataset (named WiGes) with various subjects in interaction with home appliances in different environments. To the best of our knowledge, this is the first benchmark released for studying hand gestures from a wrist-worn camera. We then evaluate various CNN models for vision-based recognition. Furthermore, we deeply analyze the models that produce the best trade-off between accuracy, memory requirement, and computational cost. We point out that among studied architectures, MoviNet produces the highest accuracy. Then, we introduce a new MoviNet-based two-stream architecture that takes both RGB and optical flow into account. Our proposed architecture increases the Top-1 accuracy by 1.36% and 3.67% according to two evaluation protocols. Our dataset, baselines, and proposed model analysis give instructive recommendations for human-machine interaction using hand-held devices.

INDEX TERMS Convolutional neural network, hand gesture recognition, human-machine interaction, wearable sensors.

I. INTRODUCTION

The use of hand gestures provides an attractive alternative to cumbersome interface devices in human-computer interaction (HCI). In the literature, the number of studies on this topic has significantly increased in recent years due to their wide range of applications [1], [2], [3], [4] in virtual reality, games, and healthcare. In such applications, gestures are acquired by sensors and then automatically inferred and mapped to a finite set of predefined control commands. Common sensors used for this task are ambient cameras or wearable Inertial Measurement Units (IMUs). Ambient cameras give rich information including the human body, human hand in action, and background. The trajectory of hands observed by an ambient camera provides the main characteristic to distinguish between gestures. However, static camera-based systems are limited by the camera field of view. If we want to control home appliances in a large space, it may require a large number of installed cameras to cover the whole area [5], [6]. Regarding wearable motion sensors, thanks to their mobility and low cost, they can overcome the above issues of ambient cameras [7], [8]. Nevertheless, motion data are very sensitive to noise and lack the capability of explanation. Using wearable cameras is an alternative solution. Some works proposed using a camera mounted on the head or the chest of the subject [9], [10]. We argue that such mounting positions are not comfortable for daily use.

This paper proposes a solution for hand gesture recognition using a wrist-worn camera. First, we design a smart-watch-like prototype with an integrated camera. Such a mounting position is more suitable and comfortable for users to wear than head or chest mounting. Some similar wrist-worn

The associate editor coordinating the review of this manuscript and approving it for publication was Shovan Barma.

designs have been introduced in [11], [12], and [13]. However, they usually convey a number of limitations, i.e., types and mounting positions of sensors being inconsistent, activities being collected in different contexts (daily life activities or finger taps only), or not publically available datasets. With the designed prototype, we collect a set of dynamic gestures. Concerning hand gesture recognition methodology, we investigate and produce baseline results with some state-of-the-art CNN models. We finally propose a novel two-stream MoviNet architecture that processes RGB and optical flow simultaneously to improve the overall accuracy. In summary, the main contributions of this paper are as follows:

i We design a system that consists of a wrist-worn device connected to an embedded computer for capturing visual data of human hand gestures. This system can be utilized in various practical applications, such as human-machine interaction or healthcare.
ii We collect and publish a new dataset of twelve dynamic hand gestures using the wrist-worn device with 50 subjects in the context of home-appliance control. To the best of our knowledge, this is the first dataset of hand gestures captured by a wrist-worn camera publicly available to the research community.
iii We investigate various CNN architectures and their performance in terms of accuracy, memory footprint, and computational time. They serve as baseline approaches for further improvements as well as for finding suitable architectures for the deployment of practical applications.
iv We propose a two-stream MoviNet architecture that takes both RGB and optical flow into account. It improves the Top-1 accuracy by 1.36% (from 94.87% to 96.23%) and 3.67% (from 94.81% to 98.48%) compared to single RGB stream MoviNet according to two evaluation protocols on our collected dataset.

Figure 1 illustrates the main components of our proposed framework. It contains six parts: designing the wearable device for collecting data, designing the hand gesture set for human-machine interaction, conducting data collection and annotation, implementing and evaluating the baseline methods as well as proposing a new method, analyzing and selecting the best model to be deployed on edge device for application of human-machine interaction.

The remaining sections of this paper are organized as follows: Section II briefly reviews related works on existing wearable devices, gesture datasets, and hand gesture recognition methods. The new prototype of the wrist-worn device and the collected dataset are presented in Section III. The proposed framework for dynamic hand gestures recognition is described in Sections IV. Section V reports experimental results on our dataset using some baseline CNNs. Finally, further discussions and conclusions are presented in Section VI.

II. RELATED WORKS

Human action and dynamic hand gesture recognition, in particular, have been highly active research topics recently. A myriad of methods has been proposed to tackle this task [14], [15]. The methods differ in the type of sensors employed to capture hand gestures (e.g., RGB, RGB-D, Gyro/IMU/Cap), as well as the recognition algorithms used to deal with video sequences [13], skeletal sequences [16], or time-series data [17]. Generally, any human action recognition method could be repurposed for hand gesture recognition. In the following, to condense the paper, we review the works directly relevant to ours.

A. EXISTING WEARABLE DEVICES, DATASETS, AND APPLICATIONS

The number of wrist-worn prototypes, as well as the number of egocentric datasets of hand gestures compared to those captured by ambient cameras, are still very limited. The type, the characteristics, and the mounting position of each integrated sensor vary from prototype to prototype. The activities are dependent on applications. Table 1 summarizes the attributes of existing datasets of hand gestures collected by wrist-worn devices and their availability.

In [18], Maekawa et al. designed a wrist-worn device integrating a camera, a microphone, an accelerometer, and a compass. Their objective was to collect data using heterogeneous sensors for object-based action recognition. The dataset contains 15 daily life activities with a duration from 0.67 to 3.65 minutes. Unfortunately, the dataset is not available for evaluation. Ohnishi et al. proposed a network to capture and recognize activities of daily living using a wrist-mounted and a head-mounted camera [13]. The wrist-mounted camera also gazes at the hand, allowing to facilitate the observation of the hand when interacting with objects. Twenty-three ADL classes were collected and annotated using both cameras. Yeo et al. [19] introduced a prototype of a wearable camera with a view of the opisthenar (back of the hand) area. The device used a single infrared camera (Leap Motion device) with an active infrared light source for easy removal of the background. They collected ten static hand postures and five individual finger tapping actions for dynamic gestures. In [20], [21], and [11], the authors deployed an RGB camera worn on the backside of user's right to capture hand gestures for human-machine interaction application. In [21], only static hand postures were collected. Chen et al. introduced another work [11] which collected dynamic hand gestures for human-robot interaction applications. Ten gestures were collected by a wrist-worn camera from 15 subjects. In [22], Wu et al. designed a wrist-worn camera to capture the fingers for hand pose estimation task. Ten static gestures of ASL digits and six dynamic gestures of finger tapping were collected with this device.

B. HAND GESTURE RECOGNITION FROM WRIST-WORN CAMERAS

The number of works on hand gesture recognition from wrist-worn devices, especially wrist-worn cameras, is finger countable. Basically, most researchers tried to demonstrate the possibility to recognize gestures from a wrist-worn camera with existing techniques without considering practical

aspects of deployment on edge or low-resource devices. Maekawa et al. [18] converted RGB images to HSV (Hue Saturation Value) color space, then utilized K-means clustering to group pixels into K classes and ranked them in terms of information gain. Then a histogram of color was computed from top-m candidate colors as a feature vector for each sequence of frames of one gesture. Finally, a combination of Adaboost with HMM (Hidden Markov Model) was deployed to recognize gestures collected from wrist-worn cameras. Chen et al. [21] first cropped the hand region and converted the image into Lab color space. Then Lazy Snapping algorithm is adopted for segmentation of the hand that served for temporal hand segmentation. For gesture recognition, they extracted SURF (Speeded up robust features) features and tracked them during the time. Finally, the dynamic time warping (DTW) algorithm was deployed for the classification of gestures. In [13], the authors extracted features from each video frame using a VGG-16 model. Then, a weighted Vector of Locally Aggregated Descriptors (VLAD) was applied for Video Pooling on Convolutional Neural Network (CNN) descriptors. The authors also combined deep features with hand-crafted features (i.e., improved Dense Trajectory - iDT) to enhance the classification performance of Support Vector Machine (SVM). Wu et al. proposed a CNN model, namely DorsalNet, that takes RGB and motion history as inputs to generate 3D hand pose [22]. DorsalNet consists of 3 parts: the pre-processing stage with the encoder-decoder network for hand masking and the motion image computation; the two-stream Long-Short Term Memory (LSTM) CNN with Kalman filter as feature extractor; and the hand simulator which reconstructs the finger angles. For gesture recognition, they appended a Multilayer Perceptron (MLP) just after the DorsalNet.

III. PROTOTYPE DESIGN AND DATA COLLECTION

A. PROTOTYPE DESIGN

We use an ordinary low-cost wide-angle RGB camera. The camera model is IMX219-160, which gives the highest resolution of 3280 × 2464 at 15 fps, and a side field of view of 160^o. At 1280 × 720 resolution, the acquisition rate may reach 90 fps. A device that integrates the camera is worn on the backside of user's wrist. As a result, the camera will capture images from the back of the hand. The camera is connected to an embedded computer (Jetson Nano) to transfer data through a CSI port. To accommodate greater flexibility for the subject, we developed a compact pack to put the embedded computer inside and power the computer for up to four hours using a 5v 10000mA battery. The solution for wireless connection between the sensors and the embedded computer is under consideration. Figure 2 illustrates the final design of our prototype.

B. DATASET COLLECTION AND ANNOTATION

1) DESIGN OF HAND GESTURE SET

As mentioned in the related works section, there are some existing works on human hand gesture recognition using wearable sensors. However, each work has designed and built a proper dataset for a specific purpose, which makes it impossible to generalize or re-use them in different applications. Moreover, most of them are not publicly available. Our work aims to develop a system for controlling home appliances (e.g., fans, air conditioners, television) through hand gestures captured by wrist-worn devices. As a consequence, a new set of hand gestures is designed. The main required characteristic for any new gesture set in human-machine interaction is that they have to be intuitive, distinguishable, easy to memorize, and performed by users.

After a careful design process, a set of twelve dynamic hand gestures named from G_1 to G_{12} has been designed. The trajectory of each hand gesture is shown in Figure 3. Each hand gesture can be mapped to one command to control in-home appliances, such as turning on/off the light switch or increasing/decreasing the temperature of air-conditioners. Intuitively, according to the subjects participated in data collection, these gestures are easy to perform and memorize. The shapes of gesture trajectories are distinctive in appearance. We will validate their discriminance by experiments in later sections.

We recruited 12 individuals aged between 23 to 50 years to conduct a survey and evaluate our gesture set. Based on the easy-to-implement criteria, 50% of the participants found the gestures to be very easy to implement, 25% found them easy, and 25% found them normal. Regarding memorability, 8.3% of participants could memorize the gestures after the first time, 50% could do so after the second time, and 41.73% after the third time. The majority of participants consent to the use of hand gestures for controlling home appliances.

2) DATA COLLECTION AND ANNOTATION

To ease data collection by a large number of subjects in different places, we have produced four identical kits, each of which includes a wrist-worn camera, a Jetson Nano, and a power supply. In all data acquisition sessions, image data was captured at 30fps with a resolution of 1280 × 720 pixels.

We invite 50 volunteers (33 men and 17 women, aged from 10 to 65 years old) to perform twelve designed gestures while standing or sitting in different environments, such as at offices, lab rooms, or at home. In each collection session, volunteers are informed consent to provide data for research purposes and explained how to wear the device and implement the gestures correctly. Each subject performs in his natural manner 12 gestures; each gesture is repeated from 2 to 12 times. All visual frames from the camera are stored in the memory of the embedded device. Furthermore, each gesture's starting and ending times are marked during data acquisition via a keypad or a remote control device to facilitate the labeling process. Therefore, all gesture instances can be automatically segmented. In total, we conducted 50 sessions of collection for 50 subjects to obtain a dataset of 5408 samples. We obtain a multimodal dataset of 5408 gesture samples.

Figure 4 shows an example of gesture G_3. The top part of the figure shows some frames extracted from a sequence

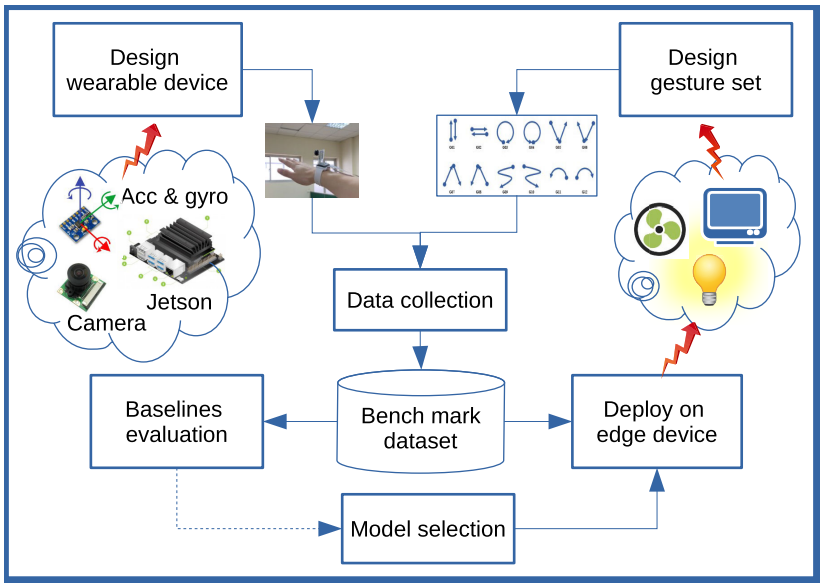


FIGURE 1. Schema of our proposed study.

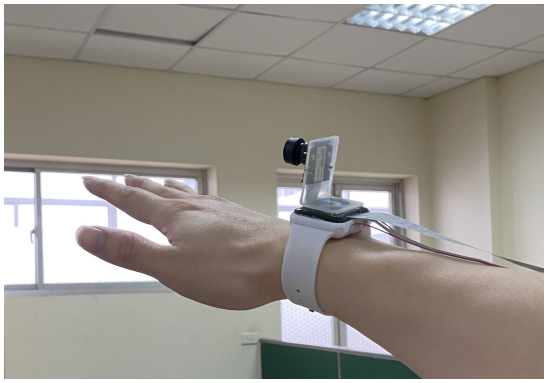


FIGURE 2. Illustration of our designed prototype.

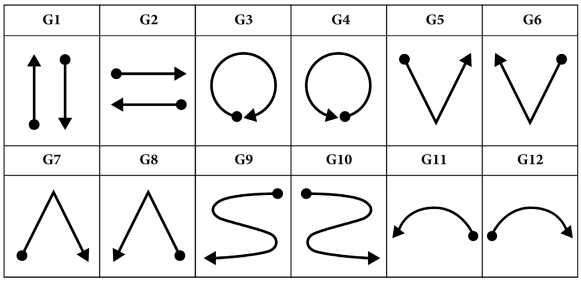


FIGURE 3. Illustration of 12 hand gestures designed in our work.

captured by a third-person view camera that observes the subject in action. The bottom part of the figure shows the corresponding frames extracted from the wrist-worn camera. We denote extraction time above each frame. Figure 5 shows the mean and deviation of gesture lengths. We can observe that the length of instances of the same gesture class may vary depending on the way that volunteers perform the gesture. However, most gesture instances have a length in the range of 60 and 80 frames.

Depending on the subjects, hand pose can be changed or not during gesture implementation. We ask the subjects to respect the pre-defined trajectories of gestures, but they can relax their hand posture in their natural manner. This naturalness represents the first challenge of our dataset. The second challenge comes from characteristics of gestures captured by a wrist-worn camera as the camera moves according to hand movement. It is noticed that while a third-person view camera can observe the whole human in action and can easily identify the trajectory of the hand (in case of occlusion free), the wrist-worn camera focuses only on a small part of the hand pose. As a result, when a subject implements a dynamic hand gesture, his/her hand's position and orientation are static relative to the camera while the background will change. The last challenge of our dataset is the poor quality of images due to the use of the low-cost camera. We compare our dataset with some relevant existing ones in Table 1. Our dataset is significant in terms of the number of subjects, number of gesture instances, diversity of background environments, number of modalities, and accessibility for the research community. The dataset and our pre-trained models are available at https://www.mica.edu.vn/perso/Tran-Thi-Thanh-Hai/MuWiGes.html.

3) EVALUATION PROTOCOL

The collected dataset is divided distinctly into training and testing sets based on two evaluation protocols we defined for the comparison of recognition algorithms, i.e., cross-subject evaluation and cross-scene-subject evaluation:

- **Cross-subject evaluation**: This evaluation protocol aims to evaluate the robustness of investigated models

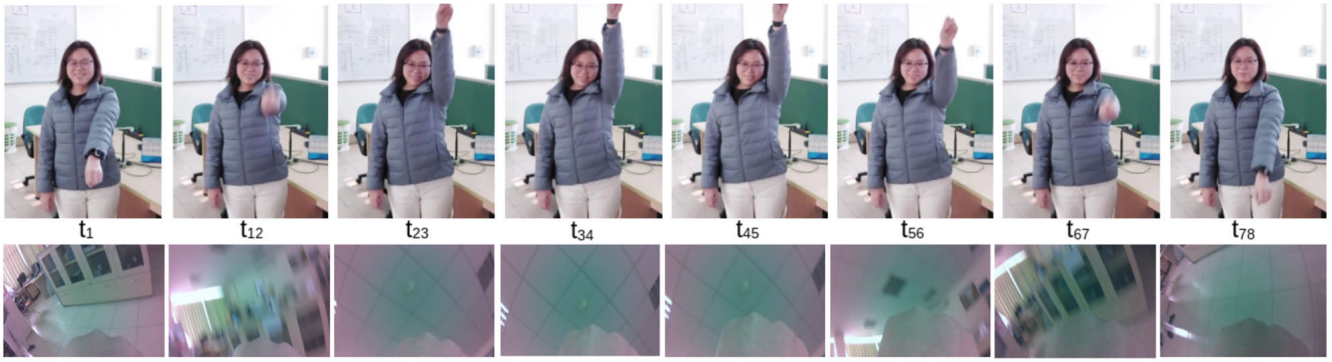


FIGURE 4. An example of gesture G_3: 8 frames uniformly extracted from the original sequence from the third person view are shown in the top row while the eight corresponding frames captured by the prototype are in the bottom row.

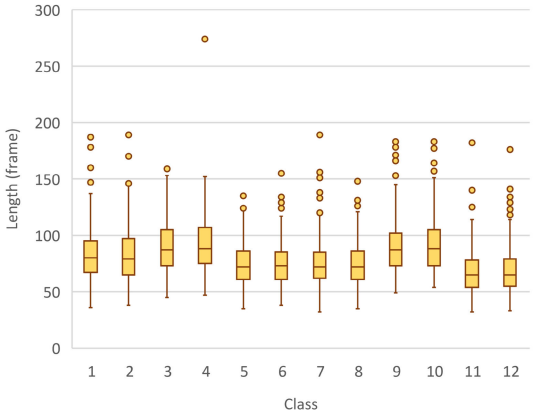


FIGURE 5. Distribution of video lengths (i.e., number of frames) in each gesture class of the dataset.

agnostic to the subjects. Data from 35 subjects are used for training, and the remaining 15 subjects are used for testing. Accordingly, the training set has 3636 video clips (i.e., gesture instances), while the test set contains 1772 instances of 12 gestures.

- **Cross-scene-subject evaluation**: This evaluation protocol is double strict in the sense that the investigated models are tested on videos of subjects and scenes both unseen during the training process. We also select data from 35 subjects performing the gestures in various scenes for training and 15 remaining subjects in other scenes for testing. Accordingly, the training set has 3633 video clips (i.e., gesture instances), while the test set contains 1775 instances of 12 gestures.

IV. BENCHMARK EVALUATION AND PROPOSED METHOD

A. GENERAL EVALUATION FRAMEWORK

In this work, we propose a framework for investigation and evaluation of various CNN models for gesture recognition from video data. We first deploy some existing baseline models from video understanding task, then we propose a new architecture that takes both RGB and optical flow to improve over the baselines. Our framework for hand gesture recognition consists of two main phases:

1) Training phase: We pre-process video data and feed them into various CNN models for training. Our objective is to find out the best CNN model for the given task in terms of accuracy, memory requirement, and GFLOPs so that it can be best suited for deployment on edge devices.
2) Testing phase: Given a new video, we uniformly resample it to produce a fixed 16-frame clip, pre-process then input it into the trained models for inference. It outputs the label of a recognized gesture.

Figure 6 shows the main steps of our framework. In the following, we will detail each step.

B. DATA PRE-PROCESSING

1) RESAMPLING DATA

The implementation of gestures varies from gesture to gesture and from subject to subject. Consequently, the length of videos is highly diverse as analyzed in the previous section. To deal with the variation in video length, we adopt a technique introduced in [23]. In the training phase, each input video is divided into K segments (K = 16 in our experiments). From each segment, we randomly select one frame. Finally, we get a K-frame clip as input of CNN models for each video sample. Due to the random frame selection, we may produce one different clip from the same gesture instance in each training batch. This technique can be considered as data augmentation in the training phase. However, in the testing phase, for a fair comparison of CNN models, we utilize one fixed clip extracted from each video by uniformly taking K frames to remove the effect of randomness.

2) OPTICAL FLOW

Optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of objects or of the camera. It is a 2D vector field in which each vector is a displacement vector representing the movement of points from the first frame to the second frame. For action recognition, optical flow has been used

TABLE 1. Comparison with existing dynamic gesture datasets captured from wrist camera (na. stands for not available).

Dataset	Instances	Activities	#Classes	#Subjects	Scenes	Modalities	Availability
Maekawa et al. 2010 [18]	na.	ADL	15	10	Home-like, Lab.	RGB, Acc., Sound	No
Ohnishi et al. 2016 [13]	628	ADL	23	20	na.	RGB	Yes[1]
Jiang et al. 2017 [20]	na.	Air gesture	3	10	na.	sEMG & IMU	No
Chen et al. 2018 [11]	1350	HCI	10	15	Room, Outdoor	RGB	No
Yeo et al. 2019 [19]	na.	Finger Tapping	5	10	na.	IR	Yes[2]
Wu et al. 2020 [22]	na.	Finger Tapping	5	6	Indoor, Outdoor	RGB	No[3]
WiGes (our dataset)	**5408**	HCI	12	**50**	Home/Office	RGB	**Yes**

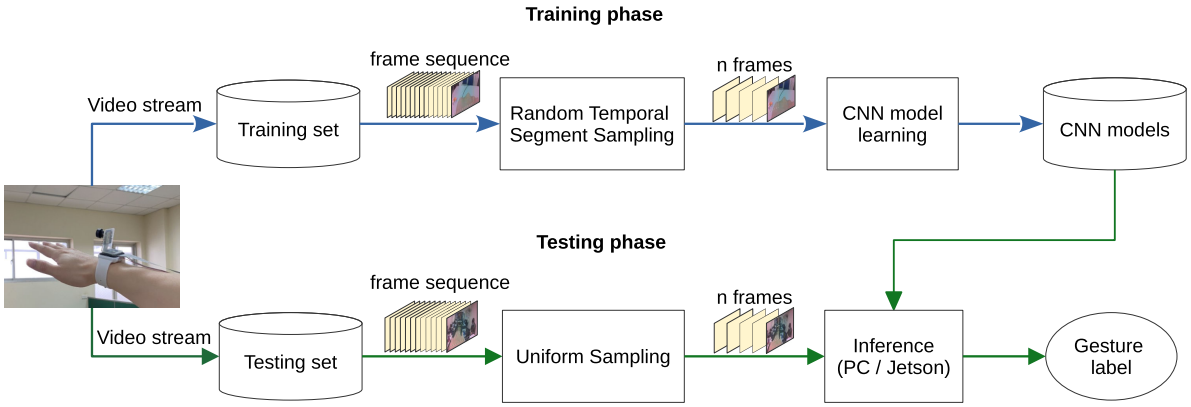


FIGURE 6. The proposed framework for hand gesture recognition.

as a robust feature for action representation. It sometimes produces higher performance than RGB stream as it focuses on characterizing transitions of objects in the scene. In this paper, we compute the dense optical flow using the Gunnar-Farneback algorithm [24]. Figure 7 illustrates two consecutive RGB images and their corresponding dense optical flow.

C. BASELINES FOR HAND GESTURE RECOGNITION

In the following, we will briefly describe state-of-the-art 2D architecture ResNet and some 3D known architectures such as C3D, R(2+1)D, R3D, MobileNet, MoviNet, and EfficientNet. We then finally describe in detail our proposed two-stream model.

1) C3D

The C3D model (3D deep convolution neural network) was first introduced in [25]. This model has shown to be very efficient for action recognition tasks and widely utilized as a baseline. The C3D network contains eight convolutional, five max-pooling, and two fully connected layers. The number of filters of convolution layers from Conv1 to Conv5 is 64, 128, 256, 512, and 512 respectively. All 3D convolution kernels are of size ($3 \times 3 \times 3$) with stride ($1 \times 1 \times 1$). The C3D takes input as an image sequence (normally a 16-frame clip) and computes the 3D convolution on each 3D cube.

2) R3D

R3D or 3D ResNet [26] is an extension of the famous residual network ResNet [27]. R3D is developed by applying convolution with 3D kernels based on ResNet network architecture. Similar to ResNet, R3D also has many variants such as ResNet18 (R3D-18), ResNet34 (R3D-34), Resnet50 (R3D-50). These networks varies in term of number of convolution layers in each elementary block, consequently increasing the network size. Similar to the C3D model, R3D allows one to take a sequence of images and return a feature vector for that sequence before passing through a classifier.

3) R(2+1)D

The R2plus1D (ResNets (2+1)D deep neural network) model was introduced in [28] and has shown relatively good performance for extracting features for action recognition task. Being almost similar to R3D, the only difference is that R(2+1)D decomposes the spatial and temporal model into two separate steps. It involves replacing 3D convolutional filters of size ($t \times d \times d$) by a (2+1)D block consisting of a 2D spatial convolution filter of size ($1 \times d \times d$) and a temporal convolution filter of size ($t \times 1 \times 1$).

4) MobileNet3D

MobileNet3D introduced in [29] is an efficient model for mobile and embedded vision applications. The MobileNet model is based on depth-separable convolutions. The depth-wise separable convolution splits it into two layers, a separate layer for filtering and a separate layer for combining. This greatly reduces computation time and model size, making it possible to build lightweight deep neural networks. MobileNet3D has 28 layers with the condition that the depth-

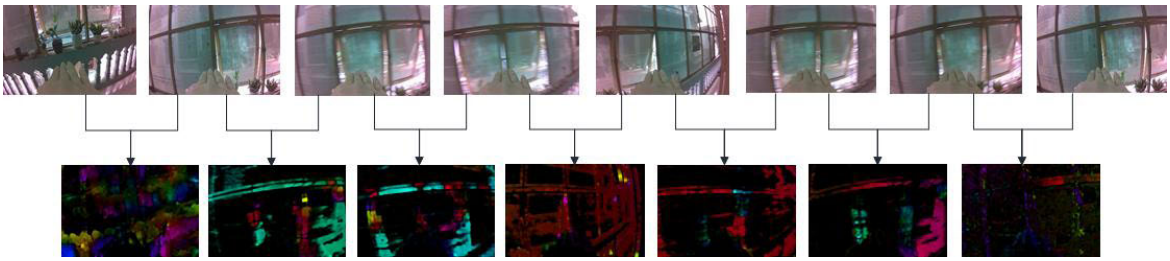


FIGURE 7. Illustration of two consecutive frames and the computed optical flow.

wise and pointwise convolutions in each MobileNet block are counted as separate layers.

5) EfficientNet3D

While C3D and R3D, R(2+1)D are heavy models, EfficientNet3D has been introduced in [30] as an additional resource-efficient architecture to be deployed on mobile devices. It has been shown that proportional feature extraction for video analysis gives relatively good results. The network takes input images with dimensions of $N \times C \times T \times H \times W$, where N and C are the batch size and channels, respectively, H and W are the height and width of the frames, T is the duration of video of the dataset. Each video is sampled to 32 frames, both of which are resized to 224×224.

6) MoviNets

Mobile Video Networks (MoViNets), which are recent works of Kondratyuk et al. [31], belong to a family of computation- and memory-efficient 3D CNNs to cope with streaming video. It includes six sub-models (i.e., MoviNet a0, a1,..a5) and one assembled model (a6) of a4 and a5 that build on image-based MobileNet [32] search space while also including the expansion parameters of X3D. It has demonstrated outstanding performance in terms of processing time and accuracy in a recently developed 3D-CNN. The three first models (MoViNets a0, a1, and a2) are lightweight versions that can be used on mobile devices such as wrist-worn devices, hence satisfying the need in our work. Kondratyuk et al. proposed three progressive steps to design efficient video models: i) Build a search space (MoViNet Search Space) on MobileNetV3 and scale it to find the best architectures according to image resolution and FPS values, ranging from a0 (the smallest model) to a5 (the largest one); ii) Introduce Stream Buffer for MoviNets as a mechanism to cache feature activations on the boundaries of sub-clips, allowing the temporal receptive field to cover the whole video and requiring no recomputation; and iii) Create Temporal Ensembles of streaming MoViNet to recover loss of accuracy from stream buffer.

7) OUR PROPOSED TWO-STREAM MoviNet

In this work, we propose to combine the RGB stream with the optical flow stream to boost the recognition rate. The combination of RGB and optical flow has shown to be more efficient than single-stream because optical flow captures better the motion information while RGB extracts the appearance features of the scene [6], [33], [34]. The work in [33] proposed a two-stream network for video-based action recognition where the spatial stream works with a still frame while the optical stream works with a stack of optical flow. Both streams utilize a 2D CNN. The authors in [34] and [6] input RGB and optical streams into two 3D CNNs. In this work, as we explored that MoviNet outperforms all other models, we then propose to use MoviNet as the building block for the two-stream framework.

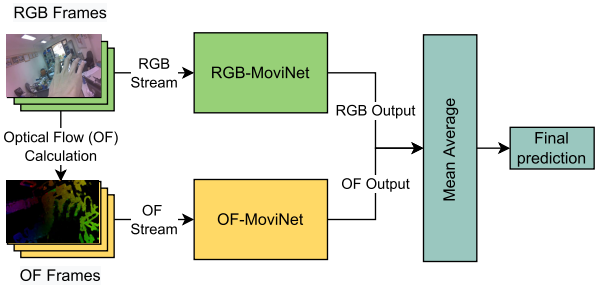


FIGURE 8. The proposed vision-based framework for hand gesture recognition.

We aggregate information from two CNN streams in a simple average-of-logits late fusion method. This means we compute the average of two probability score vectors output by two streams, and the gesture label will be decided by the argmax of the final vector. Without deep modifications, any pair of deep models can be easily plugged into this framework. In this study, we use the best models, MoviNet-a2* and MoviNet-a0 for each evaluation protocol, Cross-subject, and Cross-scene-subject, respectively (see in the experimental section). We use a * symbol to distinguish model MoviNet-a2 with input using a 224×224 image from the one with an input resolution of 172×172. The framework is depicted in Figure 8.

8) IMPLEMENTATION AND TRAINING DETAILS

In our experiments, we run the Movinet family with the Tensorflow environment and the others with the Pytorch environment. The Movinet-a0 is also implemented on both these environments, and the obtained results are equivalent. For the Pytorch environment, we use version 1.10 with

Cuda 11.3. The training of all models is implemented with 30 epochs, CrossEntropyLoss loss function, and SGD optimizer with momentum = 0.9. The learning rate is initialized at 0.0003. The a0, a2, and a5 variants of the MoviNet model are deployed on the Tensorflow version 2.7 environment using their official pre-trained models. This training process uses the RMSProp algorithm to learn the network parameters. We employed a data-parallel strategy with 4 GeForce GTX 1080 Ti GPUs to speed up training. The learning rate is set as 0.01 and changed using a cosine decay schedule during the training process. We experienced that these models also reach a stable state after around 30 epochs.

V. EXPERIMENTS

A. COMPLEXITY COMPARISON OF EXPERIMENTED MODELS

In this section, we elaborate on our findings conducted for different network architectures on our testing dataset. Various experiments have been carried out with various variations of hyperparameters, such as the number of sampling frames, temporal sampling strategies, learning rate, input resolution, and input transformation, to optimize our proposed two-stream frameworks. The reported results in this work are of models with the best hyperparameters for conducting experiments regarding system efficiency (i.e., suitable for low-performance devices and processing time, etc.).

- C3D and R3D are common architectures for action recognition problems but are not suitable for real-time and mobile applications due to their high complexity. They require the most memory as their number of parameters is very high (63.37M for C3D and 46.2M for R3D-50). They also have the highest GFLOPs (77.3 for C3D and 80.1 for R3D-50). EfficientNet3D, MobileNet3D and two lightweight implementations of R3D and R(2+1)D with only 18 layers are resource-efficient architectures. R3D-18 and R(2+1)D-18 have similar total parameters (33.2M vs 31.3M), which reduce nearly half of the memory requirement compared to the conventional version R3D-50 or C3D while keeping the same GFLOPS. EfficientNet3D and MobileNet3D are lightweight models with only 4.72M and 2.4M of parameters with GFLOPs of 0.06 and 1.1, respectively. The family of three various MoviNet-based architectures: MoviNet-a0, MoviNet-a2, and MoviNet-a5 sorted in the complexity order. MoviNet-a0 has only 1.9M of total parameters with GFLOPs of 1.8; MoviNet-a2 requires double the number of parameters (4M)) and triple GFLOPs (4.9). The most complicated MoviNet-a5 requires 17.5M and GFLOPs of 23.9, which is still much lower than R3D-18 or R(2+1)D-18.
- The last model (our proposed model) takes two streams of RGB and optical flow to two MoviNets. As a result, the number of parameters and GFLOPs double those of the original MoviNet.

B. RECOGNITION RESULTS

1) CROSS-SUBJECT EVALUATION PROTOCOL

Table 2 shows the results according to the first evaluation protocol. Two complex CNN models such as C3D or R3D-50 fail to provide accordingly higher accuracy than some lightweight models. C3D achieved Top-1 accuracy of 70.88% which is lower than that of R3D-50 (88.54%) by 17.66%. This significant improvement is raised by the residual block in R3D compared to the conventional convolutional layers in C3D.

Resource-efficient 3D models such as EfficientNet3D and MobileNet3D achieved relatively low accuracy (52.94% and 67.42% respectively). The Top-1 accuracy by R3D-18 (76.85%), the lightweight version of R3D with 18 layers, reduces dramatically by 11.69% compared to R3D-50 (88.54%). However, R(2+1)D-18 increases impressive Top-1 accuracy of 90.46% compared to his 3D version R3D-18 (76.85%) by 13.46%. It is even slightly better than R3D-50 by 1.92%. It could be explained by the fact that the decomposition of 3D convolution into two separate convolutions (2D spatial convolution and 1D temporal convolution) is advantageous, as indicated in [28].

All experimented MoviNet models output very high accuracy ranging from 92.44% (MoviNet-a0) to 94.81% (MoviNet-a2⋆), exceeding all CNN models mentioned above. The best model is MoviNet-a2⋆ with the input resolution of 224 × 224, which achieves Top-1 accuracy of 94.81% - 1.53% higher than its variant with the resolution of 172 × 172. This can be explained by the fact that the bigger input size would better capture the hand and background movement than the small size. The more complex MoviNet-a5 model fails to achieve better results than MoviNet-a2 and MoviNet-a2⋆ on this dataset as it uses only input with the resolution of 172 × 172 in our experiments due to its memory requirement as well as our system's capacity. In general, all MoviNet models get perfect Top-5 accuracy at above 99%.

2) CROSS-SCENE-SUBJECT EVALUATION PROTOCOL

To evaluate the robustness of models to environmental change, we conduct the second experiment according to the second splitting data. Table 3 shows the performance of the studied models where the subjects and the environment in the testing set are unacquainted with the training set. We observe that the MoviNet-a0 achieved the highest Top-1 accuracy (94.87%). It gradually decreases with the later models MoviNet-a2 (89.41%), MoviNet-a2⋆ (92.73%), and MoviNet-a5 (90.06%). The worst model is MobiNet3D (59.90%).

Comparing to the first evaluation protocol (Figure 9), performance (Top-1 accuracy) of almost models reduces from 2.08% (MoviNet-a2⋆) to 15.08% (C3D). However, the two models get higher Top-1 accuracy (EfficientNet3D by 15.73% and MoviNet-a0 by 2.43%). In general, we found that MoviNet is quite robust to subjects and scenes.

TABLE 2. Cross-subject evaluation: Comparison of experimental results of experimented CNN models for hand gesture recognition.

Model	Modality	Pre-trained	Top-1 Acc (%)	Top-5 Acc (%)	Frame Resolution	Params	GFLOPs
C3D	RGB	Kinetics 700	70.88	96.33	112x112	63.37M	77.3
R3D-50	RGB	Kinetics 700	88.54	99.21	224x224	46.2M	80.1
EfficientNet3D-b0	RGB	Kinetics 600	52.94	89.79	224x224	4.72M	0.06
MobileNet3D_v2_1.0x	RGB	Kinetics 600	67.42	96.26	112x112	2.4M	1.1
R3D-18	RGB	Kinetics 700	76.85	95.35	224x224	33.2M	65.9
R(2+1)D-18	RGB	Kinetics 400	90.46	99.32	112x112	31.3M	81.4
MoviNet-a0	RGB	Kinetics 600	92.44	99.38	172x172	1.9M	1.8
MoviNet-a2	RGB	Kinetics 600	93.28	99.66	172x172	4M	4.9
MoviNet-a2*	RGB	Kinetics 600	94.81	99.66	224x224	4M	4.9
MoviNet-a5	RGB	Kinetics 600	92.78	99.55	172x172	17.5M	23.9
MoviNet-a2*	OF	Kinetics 600	95.59	99.43	224x224	4M	4.9
Two-stream MoviNet-a2* (Ours)	RGB+OF	Kinetics 600	**98.48**	**99.83**	224x224	8M	9.8

TABLE 3. Cross-scene-subject evaluation: Comparison of experimental results of experimented CNN models for hand gesture recognition.

Model	Modality	Pre-trained	Top-1 Acc (%)	Top-5 Acc (%)	Frame Resolution	Params	GFLOPs
C3D	RGB	Kinetics 700	64.08	94.74	112x112	63.37M	77.3
R3D-50	RGB	Kinetics 700	81.05	97.23	224x224	46.2M	80.1
EfficientNet3D-b0	RGB	Kinetics 600	68.67	95.14	224x224	4.72M	0.06
MobileNet3D_v2_1.0x	RGB	Kinetics 600	59.90	94.51	112x112	2.4M	1.1
R3D-18	RGB	Kinetics 700	73.30	95.31	224x224	33.2M	65.9
R(2+1)D-18	RGB	Kinetics 400	83.04	98.08	112x112	31.3M	81.4
MoviNet-a0	RGB	Kinetics 600	94.87	98.82	172x172	1.9M	1.8
MoviNet-a2	RGB	Kinetics 600	89.41	98.93	172x172	4M	4.9
MoviNet-a2*	RGB	Kinetics 600	92.73	98.65	224x224	4M	4.9
MoviNet-a5	RGB	Kinetics 600	90.06	97.52	172x172	17.5M	23.9
MoviNet-a0	OF	Kinetics 600	91.27	99.04	172x172	1.9M	1.8
Two-stream MoviNet-a0 (Ours)	RGB+OF	Kinetics 600	**96.23**	**99.38**	172x172	3.8M	3.6

TABLE 4. Comparison of experimental results of experimented CNN models on EgoGesture dataset.

Model	Modality	Pre-trained	Top-1 Acc (%)	Top-5 Acc (%)	Frame Resolution	Frames	Params	GFLOPs
C3D softmax [9]	RGB	-	85.1	-	-	16	-	-
C3D fc6 [9]	RGB	-	86.4	-	-	16	-	-
C3D+LSTM+RSTTM [9]	RGB	-	89.3	-	-	16	-	-
ResNeXt-101 [35]	RGB	Jester	90.94	-	112x112	16	89M	16
C3D [35]	RGB	Jester	86.88	-	112x112	16	63.37M	77.3
MoviNet-a0	RGB	Kinetics 600	90.26	97.11	172x172	16	1.9M	1.8
MoviNet-a0	OF	Kinetics 600	80.63	94.92	172x172	16	1.9M	1.8
MoviNet-a2*	RGB	Kinetics 600	88.57	97.17	224x224	16	4M	4.9
MoviNet-a2*	OF	Kinetics 600	83.24	95.26	224x224	16	4M	4.9
Two-stream MoviNet-a0 (Ours)	RGB+OF	Kinetics 600	**91.22**	97.47	172x172	16	3.8M	3.6
Two-stream MoviNet-a2* (Ours)	RGB+OF	Kinetics 600	89.97	**97.69**	224x224	16	8M	9.8
ResNeXt-101 [35]	RGB	Jester	**93.75**	-	112x112	32	89M	16
C3D [35]	RGB	Jester	90.57	-	112x112	32	63.37M	77.3
MoviNet-a0	RGB	Kinetics 600	91.66	97.71	172x172	32	1.9M	1.8
MoviNet-a0	OF	Kinetics 600	83.81	95.5	172x172	32	1.9M	1.8
MoviNet-a2*	RGB	Kinetics 600	86.92	96.89	224x224	32	4M	4.9
MoviNet-a2*	OF	Kinetics 600	71.93	92.93	224x224	32	4M	4.9
Two-stream MoviNet-a0 (Ours)	RGB+OF	Kinetics 600	**92.14**	**97.93**	172x172	32	3.8M	3.6
Two-stream MoviNet-a2* (Ours)	RGB+OF	Kinetics 600	88.03	97.43	224x224	32	8M	9.8

Optical flow shows its great advantage when there are continual movements between consecutive frames. We take the best model on RGB (i.e., MoviNet-a2*) to train on optical flow data, the Top-1 accuracy is 95.59% which is 0.78% higher than MoviNet-a2* on RGB stream (94.81%).

3) EVALUATION OF OUR PROPOSED MODEL

We combine the two streams, RGB and optical flow by late fusion; each stream utilizes a MoviNet-a2*. With the first evaluation protocol, the highest Top-1 accuracy is obtained (98.48%), which is 2.89% and 3.67% higher than MoviNet-a2* on single optical flow and single RGB stream, respectively (Table 2). This proves that our two-stream method is capable of exploring both temporal movement and appearance change well across time.

With the second evaluation protocol, MoviNet-a0 performed worse on the optical flow stream with only 91.27% of Top-1 accuracy compared to the RGB stream with 94.87% of Top-1 accuracy. However, when we combine RGB and optical flow streams, the Top-1 accuracy improved to 96.23%, which is 1.36% higher than MoviNet-a0 with RGB and 3.5% higher vs. MoviNet-a0 with optical flow stream (Table 3).

Figure 9 compares the Top-1 accuracy of different models on RGB or derived RGB with two evaluation protocols.

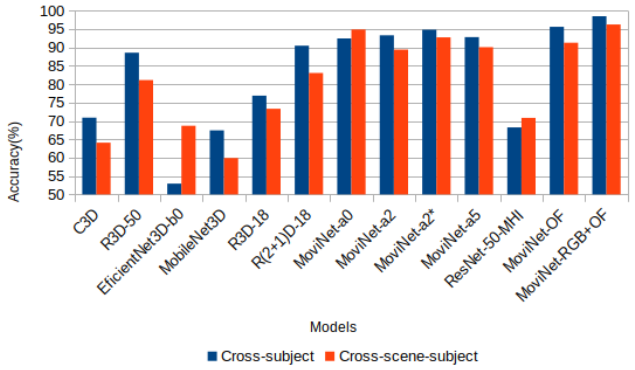


FIGURE 9. Comparison of vision-based models' Top-1 accuracy in two evaluation protocols. For the models using Optical Flow (OF) and RGB+OF combination, the model MoviNet used is the best one corresponding to the evaluation protocols (MoviNet-a2* and MoviNet-a0 with cross-subject and cross-scene-subject evaluation protocol, respectively).

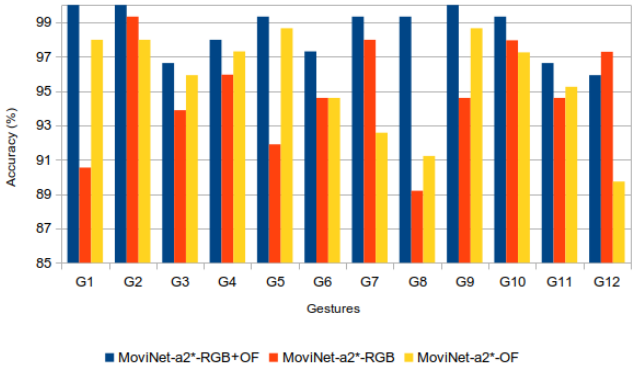


FIGURE 10. Comparison of the performance of vision-based models for each gesture in the cross-subject evaluation protocol.

In most cases, the Top-1 accuracy obtained by models is reduced from 2.08% to 7.52% in the cross-scene-subject evaluation protocol than in the cross-subject evaluation protocol. It shows that background change has an impact on the recognition rate. However, in both protocols, our proposed two-stream model achieved the highest Top-1 accuracy of 98.48% and 96.23%.

Figure 10 and Figure 11 show the Top-1 accuracy of each gesture class using RGB, optical flow, or combined RGB and optical flow using MoviNet-a2⋆ for the first evaluation protocol and using MoviNet-a0 for the second evaluation protocol. We found that the combination improves the accuracy for many gesture classes.

Figure 12 depicts an example of misclassification. It can be seen that the mistaken G_1 (the two middle rows) and the proper G_8 (the last two rows) share the same trend of g_z (angular velocity along the z-axis). This example shows that our wrist-worn prototype could achieve better performance by combining both images from camera and motion data.

4) EVALUATION OF THE PROPOSED TWO-STREAM MoviNet ON EgoGesture DATASET

To confirm the efficiency of our proposed two-stream MoviNet, we conduct experiments on another benchmark Egogesture [9]. The Egogesture dataset was collected by

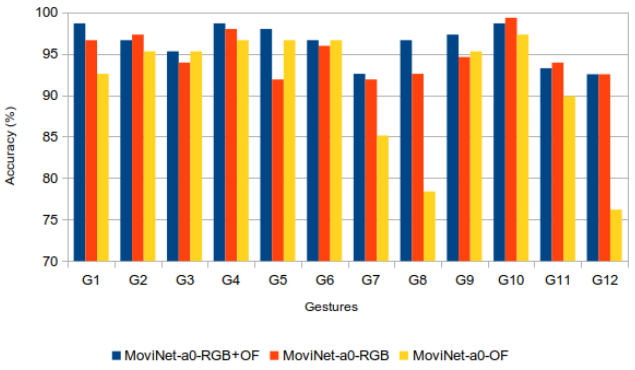


FIGURE 11. Comparison of the performance of vision-based models for each gesture in the cross-scene-subject evaluation protocol.

an egocentric camera Intel RealSense SR300 that provides both RGB and depth modalities with a resolution of 640 × 480 with a frame rate of 30 fps in four indoor and two outdoor scenes. 50 distinct subjects were invited to perform 83 classes of gestures. Totally, 24,161 video gesture samples and 2,953,224 frames are collected in RGB and depth modality, respectively. We follow the same data-splitting strategy for training and testing our framework as in the original paper. Specifically, we trained MoviNet family models with both training and validation sets (i.e., a total of 1450 videos, of which 1239 videos belong to the training set and 411 belong to the validation set) and test the model with the testing set (i.e., 431 videos). To speed up the training process, the labeled actions in each RGB video are split into smaller clips but keep their original video's parameters such as resolution, frame rate, and so on. These clips are then converted to the corresponding OF clips for OF branch of our proposed framework.

We conduct two types of experiments where the frames per clip are 16 and 32 respectively. Table 4 shows the experimental results on the Egogesture dataset. We compare our performance with existing results using C3D and ResNeXt-101 reported in [35]. It is noticed that on 16-frame clip configuration, our models achieved the highest performance (Top-1 accuracy of 91.22%) whereas the number of parameters and GLOPS is significantly reduced, even with the use of RGB and Optical Flow. The two-stream model helps to increase Top-1 accuracy from 90.26% (MoviNet-a0) to 91.22% or 88.57% (MoviNet-a2⋆) to 89.97% in case of traditional RGB modality. When testing with 32-frame clips, the performance of our two-stream MoviNet-a0 and MoviNet-a2⋆ have shown improvement over the previous experiment with 16 frames, and remains still very high (Top-1 accuracy of 92.14% and 88.03% respectively). It is not much higher than the case of 16-frame clips due to the low memory capacity of our training server. We have to set the batch size quite small (i.e., 4 and 2 with Movinet-a0 and Movinet-a2⋆, respectively) while ResNeXt-101 in [35] utilized batch size of 8. However, the difference in accuracy (e.g., Top-1 accuracy) between the baseline (ResNeXt-101) and our proposal is little while

FIGURE 12. An example of a misclassification of the model MoviNet-a2* of the same subject. The first two rows are captured images of a typical gesture G_1 and its corresponding gyroscope data. The following two rows are data of another instance of G_1, which is confused with gesture G_8, and the last two rows are collected data of the proper gesture G_8.

ensuring the necessary conditions to be deployable on edge devices (i.e., 93% on ResNeXt-101 compared to 92.14% on our two-stream Movinet-a0). We believe that these results could be improved further with proper settings. In summary, we confirm that our proposed model attains higher accuracy than other existing models. The two-stream settings also improved the accuracy of the single models of both modalities (OF and RGB).

5) DISCUSSIONS

Although most of the studied methods achieved promising results, our current method has some following limitations:

- The current method is only executed with segmented videos. In a practical application where the frames come continuously, it requires a temporal segmentation of gestures, which in turn is a more challenging task. In our previous work [36], we proposed a method to deal with continuous recognition on other datasets using other methods of recognition. We may apply the same idea to this problem but with an adaptation for dealing with continuous gesture recognition.
- The combined methods of OF with RGB are more time-consuming than using a single stream which may cause some issues when deploying it on an embedded device. In the future, we will optimize the architecture, (e.g. using quantization techniques) to reduce the memory footprint and computational requirement.
- The computation of OF step still relies on traditional methods. Currently, there are several techniques to compute OF using deep neural networks. We may exploit them in an end-to-end system.
- The recognition modules are currently executed on a separate server. We are going to deploy them on a Jetson Xavier computer for practical application of human-machine interaction.

VI. CONCLUSION

In this paper, we designed a sensory smart-watch-like device and introduced a new benchmark for hand gesture recognition from a wrist-worn camera. The dataset is challenging because it covers many real-world issues (different in-home and office environments, large intra-class variability of gestures, large extra-class similarity, and a high number of participants). It could be considered the first dataset for training and evaluating deep learning models. We also evaluated different vision-based CNN models for hand gesture recognition, including the most conventional 3D models such as C3D and R3D, as well as lightweight models (R3D-18, R(2+1)D-18), resource-efficient models such as MobileNet3D, EfficientNet3D, and MoviNets. Our experiment shows that the MoviNet variant gets high accuracy. Our proposed method that combines optical flow with RGB in a two-stream MoviNet improved the Top-1 accuracy from 1.36% to 3.37%. Consequently, this approach could be applied to enhance the efficiency of different SOTA works in video action classification fields. This finding can help the designers toward practical deployment of the most suitable model on edge devices. In the future, we will develop an application and evaluate it with subjects in different lighting, environments. We also study methods to detect quickly the gestures before calling the recognizer.

In addition, in this work, as we want to focus on the methodology to analyze actions based on hand moving from these wrist-watch-like devices, our model could easily adapt to any device with such a configuration. However, the components make the device look bulkier and ''fragile''. To deploy our device effectively in practice, we consider improving the design to comfort the wearer better and suit actual implementation.

REFERENCES

[1] S. S. Rautaray and A. Agrawal, ''Vision based hand gesture recognition for human computer interaction: A survey,'' *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, Jan. 2015.

[2] H. Doan, H. Vu, and T. Tran, ''Dynamic hand gesture recognition from cyclical hand pattern,'' in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 97–100.

[3] T. Vuletic, A. Duffy, L. Hay, C. McTeague, G. Campbell, and M. Grealy, ''Systematic literature review of hand gestures used in human computer interaction interfaces,'' *Int. J. Hum.-Comput. Stud.*, vol. 129, pp. 74–94, Sep. 2019.

[4] T. Tran, T. H. Nguyen, and V. Sang Dinh, ''Significant trajectories and locality constrained linear coding for hand gesture representation,'' in *Proc. IEEE 8th Int. Conf. Commun. Electron. (ICCE)*, Jan. 2021, pp. 359–364.

[5] D.-M. Truong, H.-G. Doan, T.-H. Tran, H. Vu, and T.-L. Le, "Robustness analysis of 3D convolutional neural network for human hand gesture recognition," *Int. J. Mach. Learn. Comput.*, vol. 9, no. 2, pp. 135–142, Apr. 2019.

[6] T.-H. Tran, H.-N. Tran, and H.-G. Doan, "Dynamic hand gesture recognition from multi-modal streams using deep neural network," in *Proc. Int. Conf. Multi-Disciplinary Trends Artif. Intell.* Cham, Switzerland: Springer, 2019, pp. 156–167.

[7] T.-H. Le, T.-H. Tran, and C. Pham, "The Internet-of-Things based hand gestures using wearable sensors for human machine interaction," in *Proc. MAPR*, 2019, pp. 1–6.

[8] E. Valarezo Añazco, S. J. Han, K. Kim, P. R. Lopez, T.-S. Kim, and S. Lee, "Hand gesture recognition using single patchable six-axis inertial measurement unit via recurrent neural networks," *Sensors*, vol. 21, no. 4, p. 1404, Feb. 2021.

[9] Y. Zhang, C. Cao, J. Cheng, and H. Lu, "EgoGesture: A new dataset and benchmark for egocentric hand gesture recognition," *IEEE Trans. Multimedia*, vol. 20, no. 5, pp. 1038–1050, May 2018.

[10] V. Pham, T. Tran, and H. Vu, "Detection and tracking hand from FPV: Benchmarks and challenges on rehabilitation exercises dataset," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, Aug. 2021, pp. 1–6.

[11] F. Chen, H. Lv, Z. Pang, J. Zhang, Y. Hou, Y. Gu, H. Yang, and G. Yang, "WristCam: A wearable sensor for hand trajectory gesture recognition and intelligent human–robot interaction," *IEEE Sensors J.*, vol. 19, no. 19, pp. 8441–8451, Oct. 2019.

[12] T. Maekawa, Y. Kishino, Y. Yanagisawa, and Y. Sakurai, "WristSense: Wrist-worn sensor device with camera for daily activity recognition," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, Mar. 2012, pp. 510–512.

[13] K. Ohnishi, A. Kanehira, A. Kanezaki, and T. Harada, "Recognizing activities of daily living with a wrist-mounted camera," in *Proc. CVPR*, 2016, pp. 3103–3111.

[14] R. M. Al-Eidan, H. Al-Khalifa, and A. M. Al-Salman, "A review of wrist-worn wearable: Sensors, models, and challenges," *J. Sensors*, vol. 2018, pp. 1–20, Dec. 2018.

[15] M. Oudah, A. Al-Naji, and J. Chahl, "Hand gesture recognition based on computer vision: A review of techniques," *J. Imag.*, vol. 6, no. 8, p. 73, Jul. 2020.

[16] S. B. Abdullahi and K. Chamnongthai, "American sign language words recognition using spatio-temporal prosodic and angle features: A sequential learning approach," *IEEE Access*, vol. 10, pp. 15911–15923, 2022.

[17] X. Zhang, Z. Yang, T. Chen, D. Chen, and M. Huang, "Cooperative sensing and wearable computing for sequential hand gesture recognition," *IEEE Sensors J.*, vol. 19, no. 14, pp. 5775–5783, Jul. 2019.

[18] T. Maekawa, Y. Yanagisawa, Y. Kishino, K. Ishiguro, K. Kamei, Y. Sakurai, and T. Okadome, "Object-based activity recognition with heterogeneous sensors on wrist," in *Proc. Int. Conf. Pervasive Comput.* Cham, Switzerland: Springer, 2010, pp. 246–264.

[19] H.-S. Yeo, E. Wu, J. Lee, A. Quigley, and H. Koike, "Opisthenar: Hand poses and finger tapping recognition by observing back of hand using embedded wrist camera," in *Proc. 32nd Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2019, pp. 963–971.

[20] S. Jiang, B. Lv, W. Guo, C. Zhang, H. Wang, X. Sheng, and P. B. Shull, "Feasibility of wrist-worn, real-time hand, and surface gesture recognition via sEMG and IMU sensing," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3376–3385, Aug. 2018.

[21] F. Chen, J. Deng, Z. Pang, M. Baghaei Nejad, H. Yang, and G. Yang, "Finger angle-based hand gesture recognition for smart infrastructure using wearable wrist-worn camera," *Appl. Sci.*, vol. 8, no. 3, p. 369, 2018.

[22] E. Wu, Y. Yuan, H.-S. Yeo, A. Quigley, H. Koike, and K. M. Kitani, "Back-hand-pose: 3D hand pose estimation for a wrist-worn camera via dorsum deformation network," in *Proc. 33rd Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2020, pp. 1147–1160.

[23] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks for action recognition in videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2740–2755, Nov. 2019.

[24] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Proc. Scandin. Conf. Image Anal.* Cham, Switzerland: Springer, 2003, pp. 363–370.

[25] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. ICCV*, 2015, pp. 4489–4497.

[26] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6546–6555.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[28] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. CVPR*, 2018, pp. 6450–6459.

[29] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[30] A. Noor, B. Benjdira, A. Ammar, and A. Koubaa, "DriftNet: Aggressive driving behavior classification using 3D EfficientNet architecture," 2020, *arXiv:2004.11970*.

[31] D. Kondratyuk, L. Yuan, Y. Li, L. Zhang, M. Tan, M. Brown, and B. Gong, "MoViNets: Mobile video networks for efficient video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2021, pp. 16015–16025, doi: 10.1109/CVPR46437.2021.01576.

[32] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324, doi: 10.1109/ICCV.2019.00140.

[33] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[34] V. Khong and T. Tran, "Improving human action recognition with two-stream 3D convolutional neural network," in *Proc. 1st Int. Conf. Multimedia Anal. Pattern Recognit. (MAPR)*, Apr. 2018, pp. 1–6.

[35] O. Köpüklü, A. Gunduz, N. Kose, and G. Rigoll, "Real-time hand gesture detection and classification using convolutional neural networks," in *Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2019, pp. 1–8.

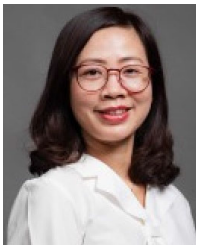
[36] T.-H. Tran and V.-H. Do, "Improving continuous hand gesture detection and recognition from depth using convolutional neural networks," in *Proc. Int. Conf. Intell. Syst. Netw.* Cham, Switzerland: Springer, 2021, pp. 80–86.

HONG-QUAN NGUYEN received the M.A. degree in computer science from Le Quy Don University, in 2010. Since 2004, he has been a Lecturer with Vietnam–Hungary Industrial University. From 2003 to 2011, he was a Software Developer with Hanoi Informatics and Telecommunication joint stock company. He is currently pursuing the Ph.D. degree with the School of Electronics and Telecommunications, Hanoi University of Science and Technology, Vietnam. His research interests include computer vision and machine learning with an emphasis on deep learning.

TRUNG-HIEU LE received the master's degree in information systems from the Academy of Posts and Telecommunications, Hanoi, Vietnam. He is currently pursuing the Ph.D. degree in computer science with the Hanoi University of Science and Technology, Hanoi. His research interests include sensor signal processing, human action recognition, machine learning, and deep learning.

TRUNG-KIEN TRAN received the degree in electronic and telecommunication from the Hanoi University of Science and Technology, in 2004, and the master's and Ph.D. degrees in machine learning from the Artificial Intelligence Laboratory, National Defense Academy of Japan, in 2014 and 2020, respectively. He was with the Department of Telecommunication Engineering and Cyber Warfare, Military Information Technology Institute, AMST, from 2004 to 2014. He is currently an AI Scientist with the Military Information Technology Institute, AMST. His research interests include meta learning for few-shot learning, machine learning on edge devices, machine learning for cyber security, computer vision, and robotics.

HOANG-NHAT TRAN received the B.E. and M.S. degrees in control engineering and automation both from the Hanoi University of Science and Technology, in 2019 and 2020, respectively. His research interests include computer vision and action recognition.



THANH-HAI TRAN received the degree in information technology engineering from the Hanoi University of Science and Technology (HUST), in 2001, and the M.S. and Ph.D. degrees in imagery vision robotics from Grenoble INP, France, in 2002 and 2006, respectively. She is currently a Lecturer/Researcher with the Department of Computer Vision, School of Electronics and Telecommunications, International Research Institute in Multimedia, Information, Communication and Application, HUST. Her research interests include visual object recognition, video understanding, human–robot interaction, and text detection for applications in computer vision.

THI-LAN LE received the degree in information technology and the M.S. degree in signal processing and communication from the Hanoi University of Science and Technology (HUST), Vietnam, and the Ph.D. degree in video retrieval from INRIA Sophia Antipolis, France, in 2009. She is currently a Lecturer/Researcher with the Department of Computer Vision, HUST. Her research interests include computer vision, content-based indexing and retrieval, video understanding, and human–robot interaction.

HAI VU received the B.E. degree in electronic and telecommunications and the M.E. degree in information processing and communication from the Hanoi University of Science and Technology (HUST), in 1999 and 2002, respectively, and the Ph.D. degree in computer science from Osaka University, Japan, in 2009. He has been a Lecturer and a Researcher with the Department of Computer Vision, MICA International Research Institute, HUST–Grenoble INP, since 2012. His current research interests include computer vision and pattern recognition, with an emphasis on applying these techniques in agricultural engineering, medical imaging, and human–computer interactions.

CUONG PHAM received the B.S. degree in computer science from Vietnam National University, in 1998, the M.S. degree in computer science from New Mexico State University, USA, in 2005, and the Ph.D. degree in computer science from Newcastle University, in 2012. He was a Visiting Professor with the University of Palermo, Italy, and a Marie Curie Research Fellow with Philips Research, Eindhoven, The Netherlands. He is currently an Associate Professor of computer science with the Posts and Telecommunications Institute of Technology (PTIT) and a Visiting Research Scientist with VinAI Research. His research interests include deep learning, computer vision, ubiquitous computing, wearable computing, human activity recognition, human–computer interaction, and pervasive healthcare.

THANH PHUONG NGUYEN received the Ph.D. degree in computer science from Lorraine University, France, in 2010, and the Habilitation degree (HDR) in 2021. He was a Teaching Assistant with Lorrain University from 2009 to 2011. He was a Post-Doctoral Fellow with CMM, Mines Paristech, in 2012, and the Robotics and Computer Vision Laboratory, ENSTA Paristech, from 2013 to 2015. Since 2015, he has been an Associate Professor with the University of Toulon, France, and a member of the SIIM Team, LIS laboratory. His research interests include image analysis, visual descriptor, action recognition, texture analysis, shape representation, and discrete geometry.

HUU THANH NGUYEN received the B.S. and M.Sc. degrees in electrical engineering from the Hanoi University of Science and Technology, Vietnam, in 1993 and 1995, respectively, and the Ph.D. degree (summa cum laude) in computer science from the University of Federal Armed Forces Munich, Germany, in 2002. From 2002 to 2004, he was with the Fraunhofer Institute for Open Communication Systems (FOKUS), Berlin, Germany. Since 2004, he has been an Associate Professor in electrical and electronic engineering with the Hanoi University of Science and Technology. He is the leader and a member of several national and international research projects on network security and the Future Internet. His research interests include edge-cloud computing, network security, the Future Internet, energy-efficient networking, and QoS/QoE. He was the Chair of the IEEE Vietnam Section from 2013 to 2018. He is currently the Vice President of the Radio Electronics Association of Vietnam (REV).