Journal of Information Hiding and Multimedia Signal Processing ©2022 ISSN 2073-4212
Ubiquitous International Volume **13**, Number **1**, March **2022**

Action recognition application using artificial intelligence for smart social surveillance system

Phat Nguyen Huu*, Dung Nguyen Tien, and Kien Nguyen Manh

School of Electrical and Electronic Engineering
Hanoi University of Science and Technology, Hanoi, Vietnam
Email: phat.nguyenhuu@hust.edu.vn; {dung.nt160685;kien.nm162234}@sis.hust.edu.vn

Received July 2021; revised December 2021

Abstract. *Computer vision is an important area of artificial intelligence that aims to help it gain the ability to activate similar to humans. In the past, we often classify fruit by hand. Today, it is performed by the development of image-processing technology. When the quantity of fruits is huge, we need machine learning for classifying them. Therefore, we propose the fruit classification using the Tensorflow and Keras model (high-level framework of Tensorflow) in the paper. This is a simple problem of computer vision since it solves the basis problems such as object detection or face recognition. In the paper, we focus on modifying the network architecture of the Tensorflow model. As a result, the accuracy of the proposed model achieves 99% with only five epochs.*

Keywords: Smart surveillance, action recognition, deep learning; human tracking; COVID19.

1. Introduction.

Nowadays, monitoring systems are more and more widely applied for human life with the strong development of science and technology [1–7]. Accordingly, monitoring systems are also increasingly diverse. Smart surveillance systems apply artificial intelligence (AI) technology that is capable of automatically identifying human actions and gestures. They are attracting more attention from computer vision researchers. Currently, using an intelligent surveillance system that can rely on gestures to identify an object with symptoms of COVID19 is necessary due to its danger and rapid spread. Action is a combination of movements of body parts since we can classify one action with another. Therefore, we will research and develop an action and gesture recognition system to identify subjects with disease symptoms and close contacts at the same time that supports COVID 19 tracing system in the paper. This is the next development of the paper [1].

In this paper, we aim to identify typical actions of subjects with COVID symptoms such as coughing, sneezing, or headache, and also identify the actions of a normal subject such as walking, running, or answering the phone. Besides, we also identify subjects that are closing contact with each other. The system can work together with another warning device to support the spread of COVID 19 in public places such as airports and train stations.

The rest of the paper is presented as follows. In Section II, we will present related work. In Section III and IV, we present and evaluate the effectiveness of the proposed model, respectively. Finally, we give a conclusion in Section V.

[*] P. N. Huu is the corresponding author at the School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Hanoi, Vietnam (email: phat.nguyenhuu@hust.edu.vn).

2. **Related work.** Due to the great demand for computer vision technology in general and in the issue of intelligent monitoring devices in particular, more and more researchers are focusing on exploiting the problem of human action recognition for surveillance systems. intelligent monitoring. Therefore, the problem of identifying human actions can be approached and handled by many different methods. There are many studies towards action identification in society [2–7]. Based on the input data, the existing literature on action recognition is divided into two categories, namely skeleton and image-based methods. Regarding the method of using 3D skeletons [2], 3D and 2D frameworks are created by Microsoft Kinect and OpenPose [3]. The methods of using images include single frame [4], multi-frame [5] and optical flow [6]. We found that the size of the skeleton dataset is much smaller than the size of the image dataset. For example, the size of the skeleton is 5.8 GB while image data is 136 GB in the case of the NTU RGB + D dataset. Therefore, the process of training the model with skeleton data will be faster than using image data. However, image data will contain other important information such as age, gender, background, etc., as well as being easier to collect than skeleton data. Therefore, the research and development of action recognition systems based on image datasets become more popular. The authors [7] a real-time action detection system to find hands-up actions in surveillance videos based on background subtraction. The results show that the proposal improves a good performance of up to 90% recall and 89% precision rates. However, the disadvantage of this method is that it is only good when the background does not change much.

Machine learning algorithms include locally-oriented histograms and support vector machines (SVMs) [8]. In the problem of action recognition based on image data, CNN is a powerful tool. The method of applying CNN to solve the problem can be divided into 2D and 3D convolutional networks. Although 2D offers attractive performance in body gesture recognition, it cannot effectively handle the recognition of continuous action streams. This is due to the temporary lack of information leading to the limitation of continuous action modeling with 2D convolution. On the other hand, 3D agglomeration contains an additional time that can correct the deficiency. Therefore, 3D convolution has been widely used in recent data-driven action recognition architectures [9]. The accuracy of I3D methods is superior to that of traditional methods. The approaches in the action recognition dataset are well known, such as UCF101 [10] and Kinetic [11].

Although using I3D networks for action recognition is highly efficient, the original version of I3D cannot recognize the actions of many people at the same time. Based on I3D, we propose a system that can identify the actions of many people. Specifically, through a model that can identify where many people appear in the video and an algorithm to track each object separately to make predictions about the actions of each of those objects. For application in an intelligent surveillance system during the spread of COVID-19 disease, we apply the Euclidean algorithm to identify close contacts.

Object detection models in general and human detection models, in particular, can be separated into two main categories, namely two-stage and one-stage methods. In the past, the authors [12] detect the positions of different objects of an image at first and then recognize RCNN, Fast RCNN, Faster RCNN, and Mask RCNN. Later models (YOLO [13] and SSD [14, 15]) combine both tasks through a neural network. YOLOV3 [13] is a network-based object detection algorithm neuron implemented in the Darknet framework. It can obtain the corresponding layer and bounding box for every object in the image and video. To compare with previous methods, YOLOV3 has the advantages of fast recognition speed, high accuracy, and the ability to detect multiple objects at the same time [13].

Online and real-time tracking (SORT) is a method of real-time multi-object tracking proposed by Bewley [16]. It combines the Kalman filter and Hungarian algorithm to predict the object of the next frame by measuring detecting speed. It tracks based on the result of detecting objects per frame. Is an updated version of SORT, the online and real-time monitoring with deep association index (Deep SORT) proposed by Wojke [17], containing a neural network additional complex for feature extraction supplement that is pre-trained by the large-scale video dataset, the motion and refinement analyzer (MARS). Therefore, Deep SORT can reduce tracking errors by more than 45% compared to SORT [17]. In the proposal system, the goal of using Deep SORT is to perform multi-person tracking where the ID number corresponding to each individual is created into a database based on the detection results of YOLO v3. This means that each person appearing in the scene will be linked by an individual bounding box and an ID number.

However, there are some challenges for system development as follows:

- Development of training templates:
  Recognition using machine learning requires a suitable sample dataset since it takes a long time to collect data to generate standard patterns.
- Processing time:
  We need to process a large amount of data. Therefore, with a network that has to handle too many parameters with weakly configured analyzers, the processing will be slow affecting the results in real-time.
- Method accuracy:
  For conventional cameras (webcams), accuracy is affected by other conditions such as lighting, background, and hand movement speed as we have to make several assumptions for the application.

Based on the above analysis results, we propose a system that combines the YOLO V3 network to determine the position of a person in the frame, Deep SORT algorithm to track the movement of a person, and I3D network for recognizing actions of a subject at risk of infection, and finally image processing algorithm to identify closing contacts.

3. **Proposal system.**

3.1. **Overview of system.**

Figure 1 shows a complete diagram of an intelligent monitoring system that records the actions of many people and identifies closing contacts. We first used YOLO v3 to determine the location of the person appearing in the scene. We then use the Deep SORT algorithm to track people and give each of them an ID. The sliding window is preprocessed and resized to include action-aware I3D.

3.2. **Implementation steps.**

According to Fig. 1, we have the following system implementation steps:

- Step 1: Preparing data

  In the paper, the training is performed on the NTU RGB + D dataset [18] with 60 action classes. Each sample scene was captured by three Microsoft Kinect V2 sensors placed in different positions. Each scene includes RGB video, depth map sequence, and 3D and infrared (IR) video frame data. The resolution of the RGB video is 1920 × 1080 pixels while the respective depth map and IR video are both 512 × 424 pixels. Skeleton data contains 3D coordinates of 25 body joints for each frame. The system will record the recommended action for a real situation. Applications for long-term medical monitoring are selected in the paper. In the training, the layers contain 800 videos excepting the background layer that has 1187 videos. The

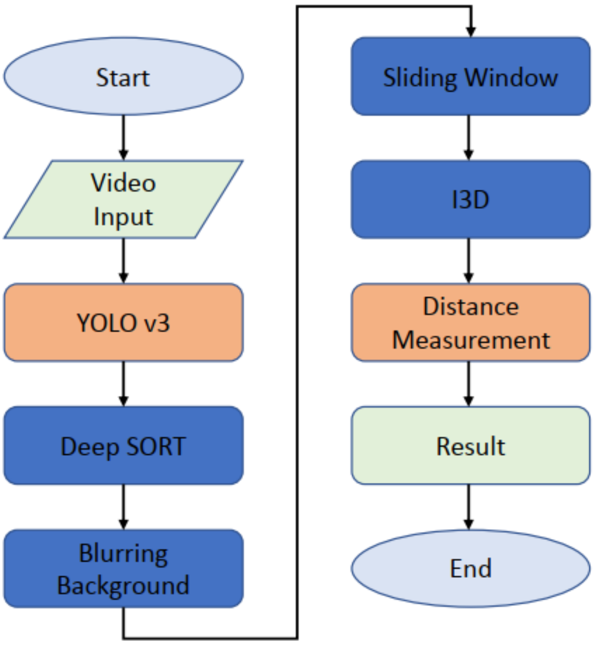


FIGURE 1. Model overview of the implementation system.

TABLE 1. Number of images and labels preparing for execution.

Number	Action	Videos
1	Run	800
2	Walking	800
3	Sneeze/cough	800
4	Headache	800
5	Phone call	800
6	Stand	800

experiment dataset uses 1841 videos. We select 6 classes of actions that are likely to be acceptable in this environment as shown in Tab. 1.

- Step 2: Using YOLO v3

This paper uses YOLO v3 in the first stage of action recognition for two reasons. The first is because the system has to determine the location of each person appearing in the scene in real-time. The other reason is that the information of the bounding box corresponding to each person in the field is very important for preprocessing in the proposed system. We convert the input video from the camera into frames. The frames are then resized from 1440 × 1080 to 640 × 480 so that YOLO V3 represents the result obtained using the coordinates of the bounding boxes. The purpose of the frame resizing process is to improve the speed and accuracy of object detection.

- Step 3: Performing Deep SORT

In Multiple Object Tracking, especially for tracking by detection, after determining the position of the object in the current frame, Deep SORT uses Kalman Filter to predict new track states based on the tracks in the current frame. in the past as shown in Fig. 2 [2]. These states are initially assigned a tentative value (tentative). This value, if it is still guaranteed to be maintained for the next 3 frames, the state

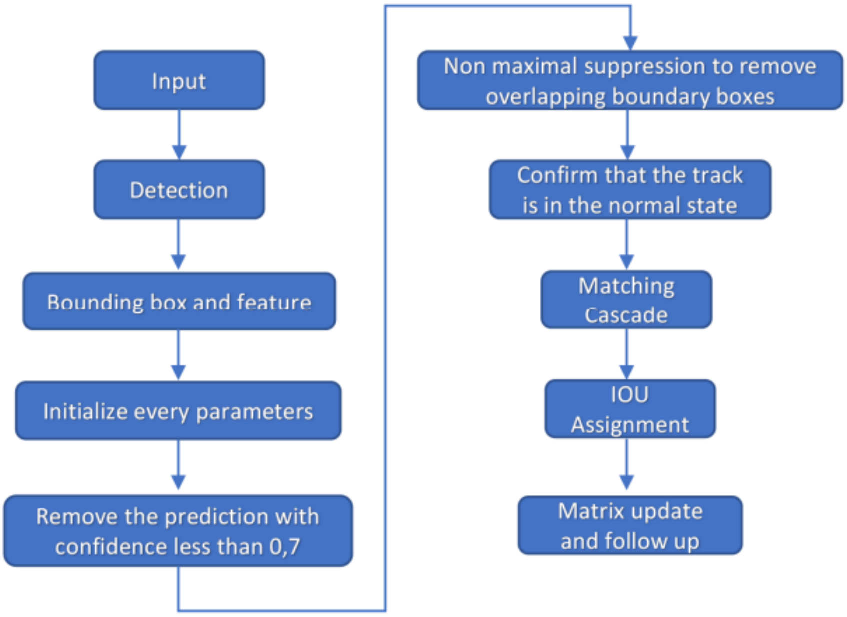

Input
Detection
Bounding box and feature
Initialize every parameters
Remove the prediction with confidence less than 0,7
Non maximal suppression to remove overlapping boundary boxes
Confirm that the track is in the normal state
Matching Cascade
IOU Assignment
Matrix update and follow up

Figure 2. Overview of Deep SORT model.

will change from poll to confirmed (confirmed) and try to stay tracked for the next 30 frames. Conversely, if the track is lost when less than 3 frames are reached, the status will be removed from the tracker. Next, the model uses validated tracks and feeds them into a matching cascade to associate detections based on distance and feature metrics. Unassociated tracks and detections are passed to the next filter layer. Using the Hungarian algorithm, solve the assignment problem with the IOU cost matrix for the second association. After that, the model processes and classifies the detections and tracks. Finally, use the Kalman filter to recalibrate the values of the tracks from the detections associated with the track and create a new track.

- Step 4: Pre-processing to remove Blurring background

  In this step, we aim to process the image to enhance quality with the I3D network for action recognition. Specifically, we blur the entire 640 × 480 frame excepting the image in the object of a frame with a Gaussian Kernel after tracking each person based on the Deep SORT algorithm. All images associated with each individual are then reduced to smaller 224 × 224 images for collection into individual datasets. The process retains important information of the image and minimizes background area to improve the accuracy of subsequent action recognition.
- Step 5: Slide Window
  As mentioned above, most human action detection work assumes video clips are segmented first and then resolved with task attribution. However, the start and end time information of an observed action is very important in processing action streams. Here, we apply sliding windows [19] to split the input video into a sequence of short overlapping videos as shown in Fig. 3. We sample the video with a blurred background after five frames. frames to create a sequence of frames to a process by sliding the window 16 frames as time goes on. The 16 frames in the sliding window for each detected person are then fed into I3D to recognize the action taken by that person. Specifically, each F video consisting of 16 sliding window frames can be built as follows:

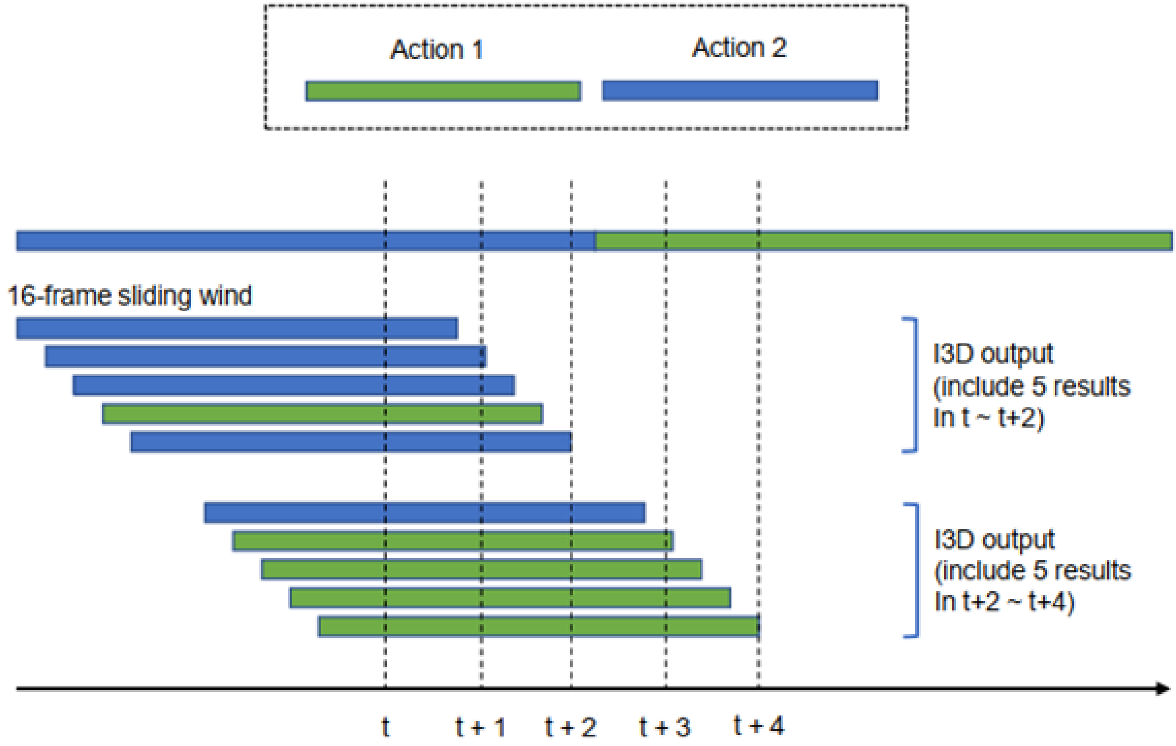


FIGURE 3. Tensorflow lite model.

$$F \;=\; \{f_c - 5n \;\; ||| \; 0 \leq n < 16\}\,, \tag{1}$$

where f_c is the frame taking at the current time. Therefore, each video clip representing 80 frames from the camera will be fed into I3D for sequence action recognition. In this paper, we use five consecutive frames of sliding windows and each window consists of 16 frames and their corresponding recognition layer by I3D grouped as a set of inputs for processing by NMS, as shown in Fig. 3 based on [2].

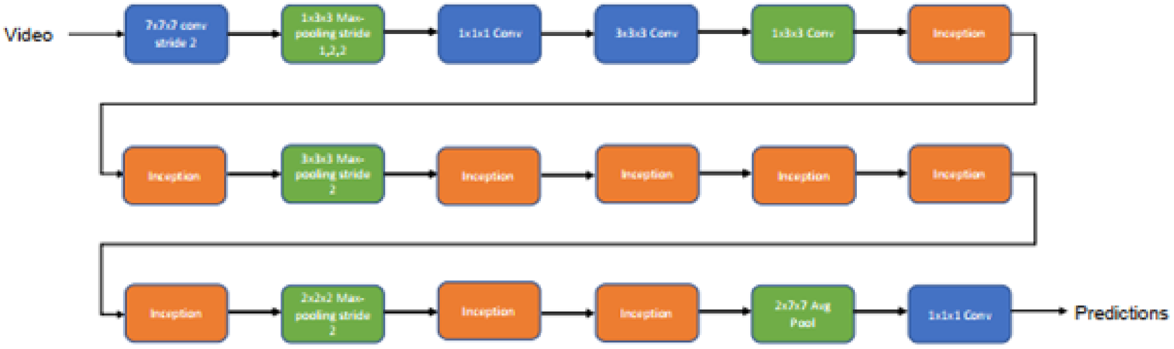


FIGURE 4. Inflated 3D (I3D) network architecture.

- Step 6: Performing I3D
I3D, a 3D convolution-based neural network architecture proposed by Carreira [10] as shown in Fig. 4 , is adopted for action recognition in a system that takes only RGB images as input data. The optical stream input in the original approach is eliminated in the design to improve recognition speed. Besides, I3D contains several starter modules containing convolution units with 1 × 1 × 1 filter, as shown in

Fig. 5 based on [12]. The design allows the size of the input data to be adjusted with different parameters by varying the number of those convolution units. In the proposal method, the input data for I3D are 16-frame window-sized videos from the previous stage. Therefore, each video segment is used to generate a corresponding recognition layer and confidence score via I3D based on the input.

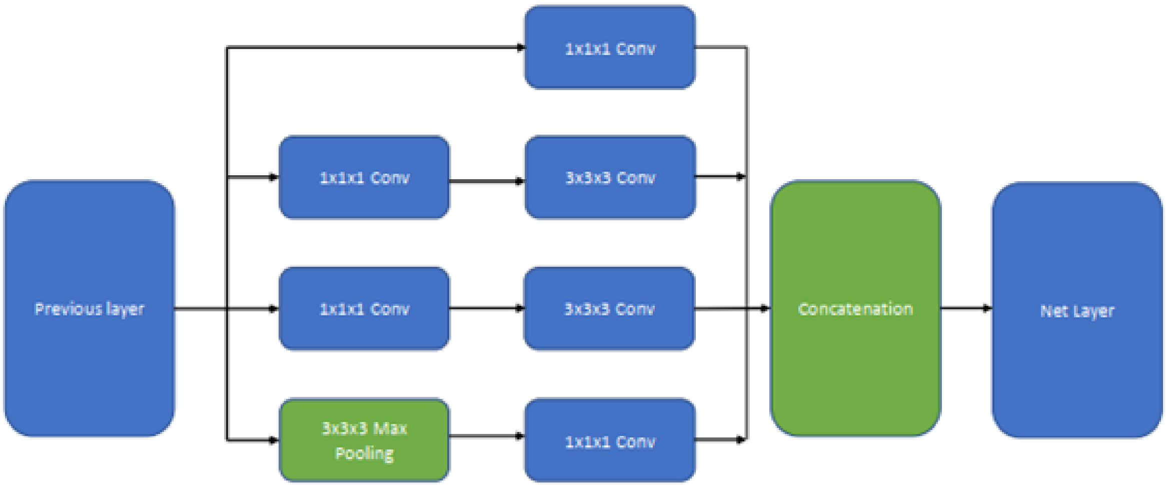


FIGURE 5. Inception module network architecture.

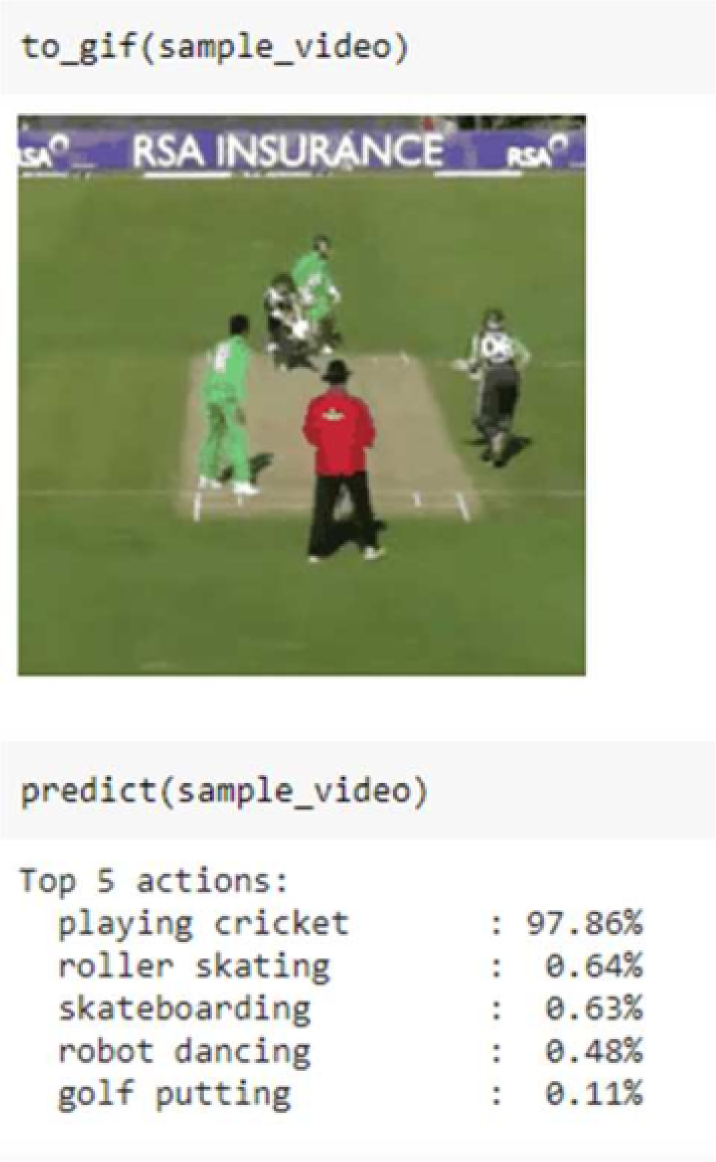
- Step 7: Calculating the distance
  In this paper, we use the Euclidean algorithm to determine the distance between objects. Specifically, we calculate the distance between the two people based on the ratio of received video pixels and the actual size of the known space after determining the position of the person in the frame.

  The intelligent monitoring system will save the information of those individuals and can make appropriate notifications for people who have been in close contact. This is important for identifying and tracking infections.

4. **Simulation and Result.** In the article, we just stopped at determining the action of an object based on the I3C model built on the Tensor Flow framework. The result of action recognition is shown in Fig. 6.

Perform action recognition based on 3D skeleton data as shown in Fig. 7. In the paper, we just stopped at determining the posture of each individual in an adult population-based on OpenPOSE [20, 21]. Besides, the data about the 3D skeleton is smaller and more convenient for the training process. However, the system also confused the parts of individuals leading to incorrect identification of 3D skeletons under conditions of a high density of people in the frame. We think that it is possible to develop an action recognition system combined with the identification of close contact objects based on the proposed method for ideal conditions.

To perform distance determination of objects in the frame, we define close contacts marked with a red Bounding Box, and objects that do not violate close contact will remain marked with a green Bounding Box. The result of the operation is as shown in Fig. 8. From the above results, we see that the system achieves the set requirements with an accuracy of over 90%. However, the system is not efficient to perform in real-time and in a large space with a high density of people. Therefore, the system development will be further improved.



```
to_gif(sample_video)
```

```
predict(sample_video)
```

```
Top 5 actions:
  playing cricket       : 97.86%
  roller skating        :  0.64%
  skateboarding         :  0.63%
  robot dancing         :  0.48%
  golf putting          :  0.11%
```

Figure 6. Action prediction results in video.

Table 2. Comparing accuracy of proposal with other methods.

Method	Accuracy using UCF101 (%)	Accuracy using HMDB51 (%)	Multi-person action	Distance measurement
[1]	98.66	95.04	YES	NO
[22]	98.40	84.20	NO	NO
[23]	93.60	66.20	NO	NO
[24]	N/A	93.70	NO	NO
[25]	96.20	71.10	NO	NO
[26]	78.43	NO	NO	NO
[27]	94.50	69.80	NO	NO
[28]	89.70	61.30	NO	NO
[29]	91.50	65.90	NO	NO
Proposal	**98**	**98**	**YES**	**YES**

Besides, we compare the accuracy of a proposal with other methods. The results are shown in Tab. 2. The results show that the proposed method has equal accuracy up to 95% while recognizing multi-person action and determining the distance among objects.

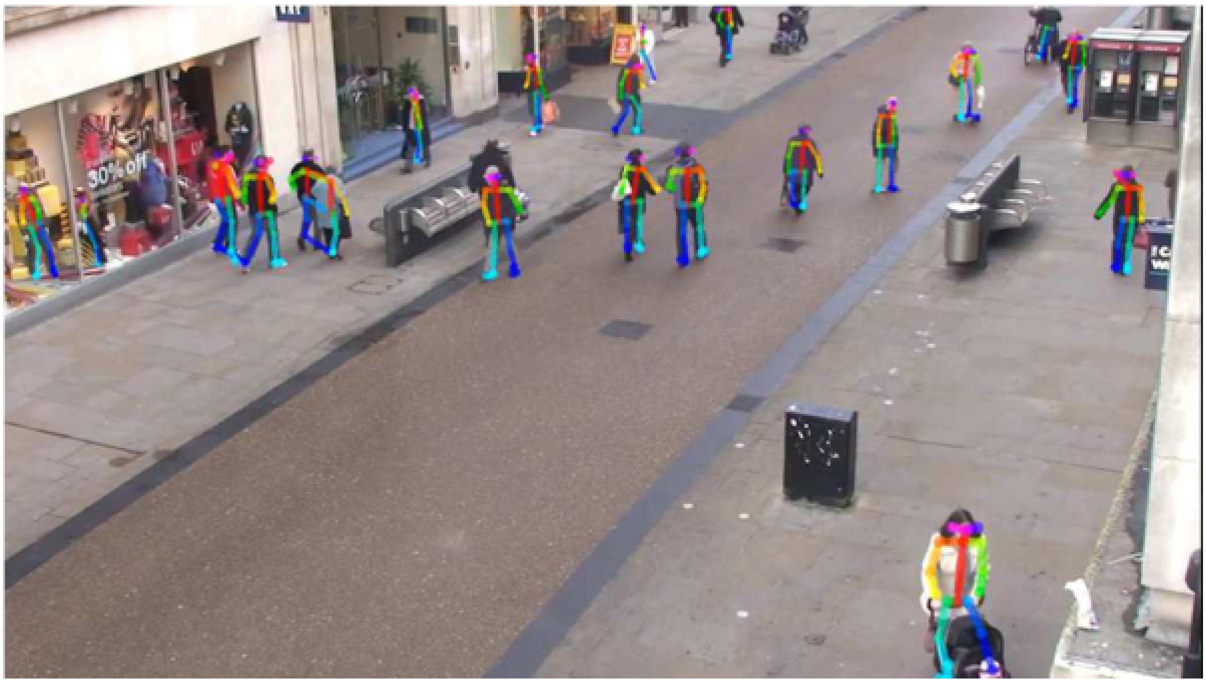


Figure 7. Posture recognition results of many objects.

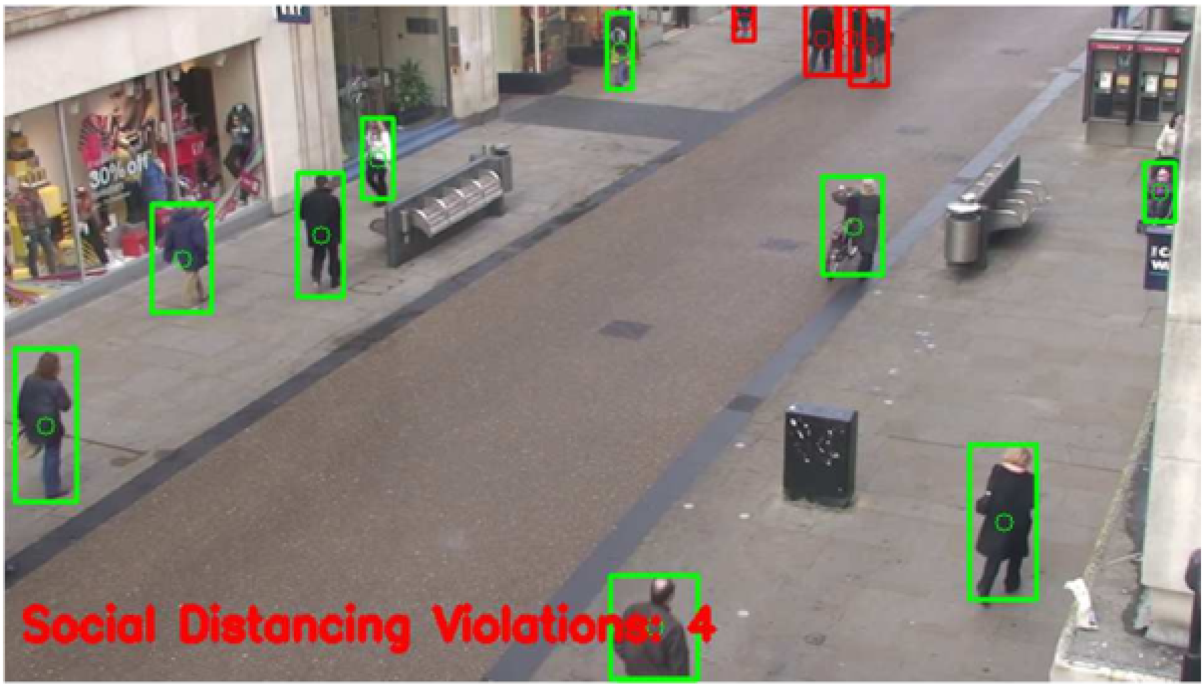


Figure 8. Posture recognition results of many objects.

To evaluate the performance of the proposed algorithm on real hardware, we compare the execution time for 5 videos. The resulting video processing time is reduced by 50% that shows the ability to perform for real applications.

5. **Conclusion.** The article focuses on researching the use of neural networks in recognizing human actions. In this article, we have identified actions with over 90 percent

Table 3. Evaluating the processing time of proposal for each UCF101 video with different length.

GPU	Tensorflow backend	Input shape	Video length (second)	Processing time (second)
Tesla T4	2.5.0	(187, 224, 224, 3)	7	3
Tesla T4	2.5.0	(153, 224, 224, 3)	6	3
Tesla T4	2.5.0	(89, 224, 224, 3)	3	2
Tesla T4	2.5.0	(168, 224, 224, 3)	6	3
Tesla T4	2.5.0	(129, 224, 224, 3)	5	2

accuracy. However, the system still has disadvantages such as low action detection results and low frame rate per second. Therefore, we will take steps such as increasing the frame rate per second, improving the accuracy by increasing the resolution of the input image, or using the preprocessing method implemented in as well as combining neural networks with other networks to increase computational efficiency and performance with any object.

Acknowledgments. This research is carried out in the framework of the project funded by the Ministry of Education and Training (MOET), Vietnam under the grant B2020-BKA-06. The authors would like to thank the MOET for their financial support.

REFERENCES

[1] N. Almaadeed, O. El Harrouss, S. Al-ma'adeed, A. Bouridane, and A. Beghdadi, "A novel approach for robust multi human action detection and recognition based on 3-dimentional convolutional neural networks," pp. 1–19, July 2019.

[2] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal lstm with trust gates for 3d human action recognition," *Computer Vision – ECCV 2016. ECCV*, pp. 816–833, 2016.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1302–1310.

[4] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, T. Darrell, and K. Saenko, "Long-term recurrent convolutional networks for visual recognition and description," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2625–2634.

[5] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[6] P. N. Huu, T. L. Ngoc, and Q. T. Minh, "Proposing gesture recognition algorithm using two-stream convolutional network and lstm," in *2020 IEEE Eighth International Conference on Communications and Electronics (ICCE)*, 2021, pp. 427–432.

[7] M.-H. Hung and J.-S. Pan, "A real-time action detection system for surveillance videos using template matching," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, pp. 1088–1099, Jan. 2015.

[8] C.-C. Hsieh and D.-H. Liou, "Novel haar features for real-time hand gesture recognition using svm," *Journal of Real-Time Image Processing*, vol. 10, pp. 357–370, June 2012.

[9] P.-J. Hwang, C.-C. Hsu, and W.-Y. Wang, "Development of a mimic robot—learning from demonstration incorporating object detection and multiaction recognition," *IEEE Consumer Electronics Magazine*, vol. 9, no. 3, pp. 79–87, 2020.

[10] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4724–4733.

[11] K. Soomro, A. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, pp. 1–7, Dec. 2012.

[12] J. Carreira, E. Noland, C. Hillier, and A. Zisserman, "A short note on the kinetics-700 human action dataset," *ArXiv*, vol. abs/1907.06987, 2019.

[13] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," April 2018.

[14] P. Nguyen, H. Thu, and M. Q. Tran, "Proposing a recognition system of gestures using mobilenetv2 combining single shot detector network for smart-home applications," *Journal of Electrical and Computer Engineering*, vol. 2021, pp. 1–18, Feb. 2021.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[16] Y.-T. Wu, Y.-H. Chien, W.-Y. Wang, C.-C. Hsu, and C.-K. Lu, "A yolo-based method on the segmentation and recognition of chinese words," *Proceedings of the International Conference on System Science and Engineering*, June 2018.

[17] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3645–3649.

[18] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+d: A large scale dataset for 3d human activity analysis," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1010–1019.

[19] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," June 2016, pp. 1049–1058.

[20] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2021.

[21] S. Qiao, Y. Wang, and J. Li, "Real-time human gesture grading based on openpose," in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017, pp. 1–6.

[22] S. Asghari-Esfeden, M. Sznaier, and O. Camps, "Dynamic motion representation for human action recognition," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020, pp. 546–555.

[23] M. Majd and R. Safabakhsh, "Correlational convolutional lstm for human action recognition," *Neurocomputing*, vol. 396, pp. 224–229, 2020.

[24] M. Khan, K. Javed, T. Saba, and U. Habib, "Human action recognition using fusion of multiview and deep features: An application to video surveillance," *Multimedia Tools and Applications*, March 2020.

[25] D. Avola, M. Cascio, L. Cinque, G. L. Foresti, C. Massaroni, and E. Rodolà, "2-d skeleton-based action recognition via two-branch stacked lstm-rnns," *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2481–2496, 2020.

[26] H. Rashwan, M. García, S. Abdulwahab, and D. Puig, "Action representation and recognition through temporal co-occurrence of flow fields and convolutional neural networks," *Multimedia Tools and Applications*, vol. 79, Dec. 2020.

[27] Z. Tu, W. Xie, Q. Qin, R. Poppe, R. C. Veltkamp, B. Li, and J. Yuan, "Multi-stream cnn: Learning representations based on human-related regions for action recognition," *Pattern Recognition*, vol. 79, pp. 32–43, 2018.

[28] L. Wang, L. Ge, R. Li, and Y. Fang, "Three-stream cnns for action recognition," *Pattern Recognition Letters*, vol. 92, pp. 33–40, 2017.

[29] L. Wang, Y. Qiao, and X. Tang, "Action recognition with trajectory-pooled deep-convolutional descriptors," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4305–4314.