Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2022.024890
Article

Tech Science Press

Classification of Multi-Frame Human Motion Using CNN-based Skeleton Extraction

Hyun Yoo[1] and Kyungyong Chung[2,*]

[1]Contents Convergence Software Research Institute, Kyonggi University, Suwon-si, 16227, Korea
[2]Division of AI Computer Science and Engineering, Kyonggi University, Suwon-si, 16227, Korea
*Corresponding Author: Kyungyong Chung. Email: dragonhci@gmail.com
Received: 03 November 2021; Accepted: 23 December 2021

Abstract: Human pose estimation has been a major concern in the field of computer vision. The existing method for recognizing human motion based on two-dimensional (2D) images showed a low recognition rate owing to motion depth, interference between objects, and overlapping problems. A convolutional neural network (CNN) based algorithm recently showed improved results in the field of human skeleton detection. In this study, we have combined human skeleton detection and deep neural network (DNN) to classify the motion of the human body. We used the visual geometry group network (VGGNet) CNN for human skeleton detection, and the generated skeleton coordinates were composed of three-dimensional (3D) vectors according to time changes. Based on these data, we used a DNN to identify and classify human motions that were most similar to the existing learned motion data. We applied the generated model to the data set that could occur in general closed circuit television (CCTV) to check the accuracy. The configured learning model showed effective results even with two-dimensional continuous image data composed of red, green, blue (RGB).

Keywords: Artificial intelligence; artificial neural networks; video analysis; human pose estimation; skeleton extraction

1 Introduction

Advances in artificial intelligence, when combined with the field of image analysis, have been used in various ways, such as object classification, text reading, and disease detection, which could not be done in previous studies [1,2]. In the field of video recognition, 2D continuous image data is used as the basic data. These image data are generally acquired from image sensors, such as CCTVs and webcams [3]. The field of video recognition includes detecting, classifying, and extracting human skeletons from the acquired image data, which is useful for understanding the behavior of objects and people to analyze the relationship between objects. Video recognition technology can be applied in various fields, such as city safety, police, national defense, and transportation. Recent studies applied it to practical fields such as traffic flow analysis and vehicle license plate character recognition [4,5]. 2D human pose estimation is a part of the field of video recognition, and studies concerning it involve classifying human shapes and motions

from 2D images composed of RGB and categorizing motion behavior. Therefore, we can understand the image by classifying the characteristics of human behavior. The analysis method determines feature points related to the human body in the image, sets the surrounding environment, and traces the trajectory. Recent studies conducted research on image analysis using deep learning, some of which implemented using equipment such as intelligent CCTV [6]. The implemented system detected abnormal behavior such as beatings and falls from images, and various methods, including intrusion detection and cluster analysis, were being tried. However, human pose estimation is difficult to interpret because it requires identifying complex human joint objects and analyzing their movement over time. In particular, continuous 2D images show a low recognition rate owing to various problems such as search failure of some body objects, interference between objects, overlapping, and a relatively slow response speed. Therefore, it is sometimes implemented by using equipment such as motion sensors or 3D multiview [7]. Such a search method has problems because expensive or complex equipment configuration is required. Thus, its practical application is limited owing to its low accuracy, large-scale operation, and high cost [8].

In this study, the human skeleton was extracted from 2D continuous image data composed of RGB using CNN-based VGGNet. The generated skeleton coordinates were evaluated for changes over time, and the most similar human behavior patterns were classified. In this process, only continuous 2D RGB images were used without additional equipment, so it was easy to apply to general internet of things (IoT) equipment such as CCTVs and webcams. We propose a method for classifying risk behavior by lightening skeleton's pattern comparison algorithm and recognizing human behavior in real time. To this end, CNN-based VGGNet was used first to construct a human skeleton detection neural network [9,10]. The positions of the body joints and major motion parts within the image were predicted through the learning and extraction results of the neural network. Then, the skeleton coordinates were extracted by connecting each motion part through the confidence map and preference analysis of the body part position. A simple DNN was constructed for motion classification, and a 3D vector called a continuous motion pattern dataset was constructed for DNN learning [11]. This 3D vector is a form in which the body skeleton coordinates are extracted from the 2D video images and continuously connected over time. Therefore, the constructed model can compare the similarity between the new skeleton extraction data and the DNN trained on the existing learning dataset. The constructed DNN model can evaluate the similarity between the new skeleton extraction data and the existing training dataset by using a 3D vector. Based on the evaluation, it was classified as motion within the image data. The built learning model showed effective results in real-time situations with only 3D vector information.

In this study, video recognition is described in Section 2. The proposed classification of multi-frame human motion using CNN-based skeleton extraction is described in Section 3. The results and performance evaluation are described in Section 4, and the conclusions are presented in Section 5.

2 Related Works

2.1 Video Recognition

The goal of video recognition is to identify whether an object exists in a specified category and indicate the spatial location and extent of each object. Existing image analysis studies have been conducted using image analysis algorithms such as the Lucas-Kanade algorithm [12] and the Gunnar Farneback algorithm [13], and information such as the size, movement, and direction of the object is extracted through this method. However, these studies are vulnerable to noise such as body movements, rain, or wind. Above all, it is difficult to capture and classify the positions of various objects. However, the field of video recognition has greatly advanced through algorithms such as CNN based on deep learning. Convolutional neural networks (CNNs), the most representative model of deep learning, classify and recognize images through a convolution layer for feature extraction, and a pooling layer for classification [14]. This method

works effectively for visual data analysis of images because it can directly learn the unique feature pattern of an object from an RGB image composed of 2D. Since then, AlexNet [15] has been improved to a method for constructing the interior in parallel based on CNN, which won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC) and gained attention as it surpassed the existing method [16]. R-CNN (regions with CNN features) further improved the existing CNN [17]. This method used the region proposal algorithm to set a candidate region where an object might exist and determined the location of the object to recognize the object through CNN. Although R-CNN showed a high recognition rate, it has a disadvantage in that it is slow because all region proposals must go through CNN. After compensating for this disadvantage of R-CNN, Fast R-CNN had a fast processing speed by recognizing objects through the feature map output through the CNN for the entire image [18]. Unlike the existing R-CNN and Fast R-CNN, which generated region proposals through an external algorithm called selective search, Faster R-CNN further improved the processing speed by using the region proposal network (RPN) inside the CNN [19]. The R-CNN series model had a slow operation speed because it predicted the area where the object may exist and went through two steps of object detection and classification for the area. Residual network (ResNet) surpassed human accuracy capability in 2015 [20], while squeeze and excitation network (SENet) achieved a figure well below the human recognition error rate in 2017 [21]. In 2018, a study used ResNet's deep learning method to extract and separate a new spatial feature called Deep-Crowd and applied it to security-related work [22].

As the inclusion of the temporal dimension is required, in recent studies, research using not only stereotypical videos but also unstructured videos capable of temporal motion extraction has been actively conducted. The study extracted various visual features from RGB video data, and improved the performance of temporal motion analysis by fusion of the RGB frame, which is spatial information, and optical flow, which is temporal information [23]. To improve accuracy, the study combined RGB and depth-based approaches to obtain higher recognition results compared to the single detection method. You only look once (YOLO) is a single-step algorithm that sets a bounding box and detects objects in one network [24]. YOLO divides the image into grids of the same size and creates two bounding boxes around each grid cell. Subsequently, it calculates the probability that the bounding box might contain the object and classifies the objects. This process was carried out in one network. Therefore, it has the advantage of faster execution speed, making real-time prediction possible compared to the existing algorithm. However, there is a possibility of detection errors, such as the estimation of a small object as a background [24]. In 2019, a lighter and more effective SlowFast Network was developed, which does not use the existing two-stream network that uses RGB and optical flow as inputs [25]. The SlowFast Network separates spatial structures and temporal events. Two streams are used in this process: the slow pathway, which captures the spatial meaning, and the fast pathway, which captures behavioral information such as rapidly changing movements. The SlowFast Network took 1st place with its high performance in the field of AVA Challenge Action of the CVPR2019 workshop [26], and it still occupies the top rank in the active field. However, these various algorithms have different purposes and operational characteristics. Therefore, it is difficult to compare the accuracy between each other. For example, SlowFast's Slow algorithm uses a low frame rate and analyzes the spatial situation according to the overall contents of the image. Therefore, due to the characteristics of the slow model, it has a different purpose from the basic behavior pattern analysis. On the other hand, the existing pose estimation using human skeleton detection shows high accuracy in the still image. However, since still image data is basically used, motions occurring in continuous images cannot be classified. In recent studies, studies to detect continuous motion are being conducted. However, it is centered on the detection of one action. In this way, since the purpose of operation classification is diverse, the results of performance comparison between each other are not accurately expressed.

Separately, there is a study on designing small and lightweight networks to reduce or eliminate network redundancy using existing algorithms [27], and interest in research on network acceleration is increasing [28]. Research is also being conducted to balance the depth, width, and resolution of the network as well as accuracy so that it can be used in various devices and improve accuracy and efficiency. Models that consider the size and calculation process of the model are also being studied [29].

3 Classification of Multi-Frame Human Motion Using CNN-Based Skeleton Extraction

3.1 Pose Data Collection and Data Pre-Processing

To configure the classification of multi-frame human motion using CNN-based skeleton extraction, a neural network and dataset for extracting 2D skeleton shapes are required. The neural network configuration uses a CNN-based OpenPose [30,31]. Skeleton extraction using OpenPose uses a 2D RGB image as the input. In this method, each joint and major part of the human body are estimated using a bottom-up method for more than one person in the image, and the relationship between the parts is used to extract skeleton data composed of 2D coordinates for each person. Internally, CNN-based VGG-19 [32] is used to simultaneously generate confidence maps and affinity fields indicating the location of body parts, direction information, and the relationship between each body part is compared [30,31]. Fig. 1 illustrates the skeleton extraction structure using OpenPose.

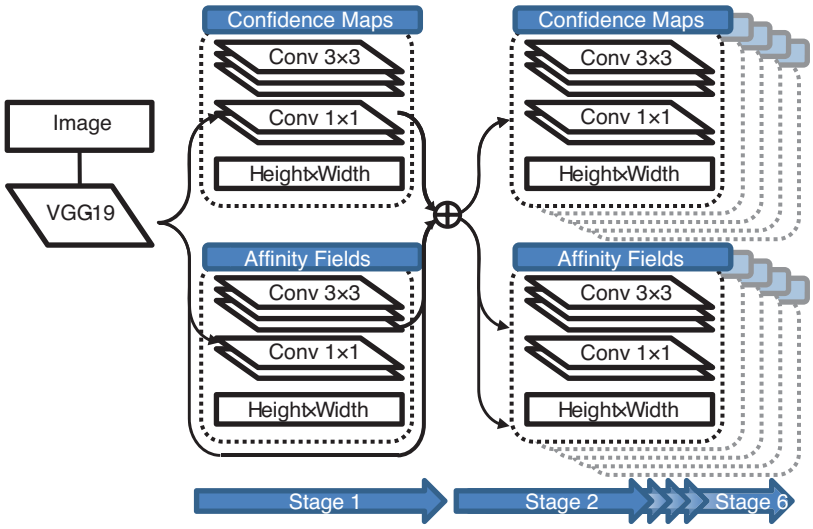
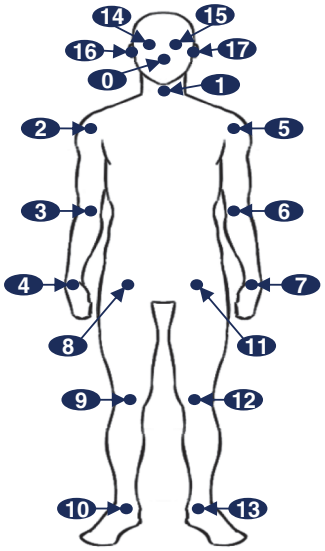


Figure 1: Skeleton extraction structure using OpenPose

In Fig. 1, the upper part is a configuration that enters an image as VGG-19 and converts it into a vector, and the left part is a neural network for confidence maps [30,31]. The right side shows the configuration for the affinity fields. The internal structure of both neural networks consists of three 3 × 3 conv layers. Three 3 × 3 layers and two 1 × 1 conv layers. The two neural networks have a cascaded stage structure and are connected in the form of extending the stage. The expanding stage connects the confidence map and affinity field information via vector addition. Therefore, the linked structure refers to all the contents of the confidence map and affinity field of the previous stage. In the internal structure of OpenPose, the blocks connected through this method show more accurate results as the stage progresses, and it is repeated six times. The operation procedure of this method consists of three steps: first, an image is entered, and the position of each object is predicted. These predictions appear as confidence maps in which the position of the object is presented as a probability value. Next, vector fields connecting objects

related to each other are allocated and expressed as affinity fields. Finally, the affinity fields that estimate whether two objects connected to each other belong to the same person are used to perform the bipartite matching operation.

The common objects in context (COCO) data set [33] was used as data for extracting the learning and skeleton shape. As large-scale data for object detection, segmentation, and captioning, the COCO dataset was presented with more than 100,000 human body parts labeled. This dataset consisted of data representing various difficulties, such as clustering and contact from general problems such as size change and occlusion in image analysis. Fig. 2 illustrates the position of human body objects based on COCO dataset. The learning and motion analysis results using the COCO dataset consisted of 18 objects and background coordinates of the human body, and the details of their location are shown in Fig. 2.



Index	Object	Side
0	Nose	
1	Neck	
2	Shoulder	Right
3	Elbow	Right
4	Wrist	Right
5	Shoulder	Left
6	Elbow	Left
7	Wrist	Left
8	Hip	Right

Index	Object	Side
9	Knee	Right
10	Ankle	Right
11	Hip	Left
12	Knee	Left
13	Ankle	Left
14	Eye	Right
15	Eye	Left
16	Ear	Right
17	Ear	Left
18	Background	

Figure 2: Position of human body objects based on COCO data set

The detailed object of the human body was created as a two-dimensional tensor composed of the x and y coordinates. In the generated data, various errors occurred during the extraction process. Skeleton errors were mainly divided into a detection error in which some coordinates were detected in the wrong place, and a missing error caused due to the occlusion of specific objects such as hands and feet. In addition, there was a bias error in which the center position of the coordinates and the size of the human body were different depending on the position of the object in the image. Because omission errors and bias errors are relatively frequent, a response was necessary. Therefore, correction and normalization were required according to the occurrence of errors. The work is mainly composed of corrections according to object loss and normalization according to the coordinate bias. It was not used when omissions occurred in Neck (1), Hip-Right (8), and Hip-Left (11), which were important components of compensation work according to object loss. In addition, it was not used if more than four objects were missing from other objects. If some objects are missing, they are assumed to be obscured by the body and initialized to the body center coordinate BC (Body Center). BC is the center coordinate of the skeleton structure; specifically, it is the mean of the coordinate values of Neck (1), Hip-Right (8), and Hip-Left (11). Next, in the generated coordinates, the position was biased, or the shapes with different human body sizes were normalized. Normalization ensures that the positions of all objects have the same center and size.

The skeleton structure coordinates have different sizes and positions depending on the characteristics of the human body in the image. Therefore, the minimum value among the coordinates of all objects was used for each frame to ensure that all shapes started from the same coordinates. Next, the maximum value was used to zoom in/out to ensure that the positions of all objects were expressed as percentages. Thus, the skeleton structure coordinates were normalized to the same central coordinates and size. Fig. 3 shows the shape of the result of transforming the real image into coordinates and normalizing it.

Index	0	1	2	3	4
x	37	62	75	75	12
y	13	15	39	60	13
Index	**5**	**6**	**7**	**8**	**9**
x	0	12	50	100	62
y	39	60	60	73	100
Index	**10**	**11**	**12**	**13**	**14**
x	12	37	37	33	33
y	60	66	93	44	44
Index	**15**	**16**	**17**	**18**	
x	50	25	33	37	
y	10	5	44	13	

Figure 3: Coordinate transformation and normalization of images

The image on the left in Fig. 3 shows the shape of the actually detected object of the human body object. The image on the right is the result of transforming and normalizing this result to each coordinate. This result changed over time in the video. Therefore, the coordinates extracted from the image were accumulated and used according to the progress of the frame.

3.2 Motion Data Collection and Data Pre-processing

Prior to motion recognition, data collection and pre-processing were performed first. The basic data used the human motion video of the AI Hub produced by the National Information Society Agency. This video data was used to estimate the posture of the human body image inside the video entered through the CCTV. These data included two-dimensional images and 3D joint coordinates of human motions in various postures and consisted of 200,000 clip images composed of a total of 50 types of human motion images. This study focuses on public environmental facilities; therefore, sports scenarios such as soccer and basketball and indoor scenarios such as housework and home training are excluded. In addition, the minimum unit of the motion recognition algorithm is a single object, so data such as handshakes and hugs in which many people overlap are excluded. This part can be expanded and used for the recognition of multiple object motions in combination with an object recognition algorithm in the future. In addition, data such as Taekwondo and national military gymnastics using a combination of various movements rather than simple movements were excluded. Tab. 1 shows the basic motion data.

Table 1: Basic motion data

No	Style	Motion	Count	Use
1	basic	walking	4,128	O
2	basic	running	2,016	O
3	basic	sit down	2,112	O
4	basic	greetings (worship + bow)	3,840	O
5	basic	hug	2,112	O
6	basic	fall down	3,360	O
7	basic	cross arms	2,304	O
8	basic	eating action	2,304	O
9	basic	jump	2,688	O
10	basic	climbing stairs	3,072	O
11	basic	push up	2,304	O
12	basic	direction indication	2,304	O
13	basic	clap	1,920	O
14	basic	sit-up	2,304	O
15	basic	crawl	5,184	O
16	basic	taekwondo	2,304	X
17	basic	national military gymnastics	1,440	X
18	basic	greeting (waving hand)	3,840	X
19	basic	greeting (handshake)	1,440	X
20	complex	yoga	9,600	X
21	complex	dance (idol) - multiple people	14,400	X
22	complex	dance (outside the club) - multiple people	14,400	X
:	:	:	:	X
48	complex	weightlifting	1,728	X
49	complex	fencing	1,728	X
50	complex	housework	9,216	X
Total		199,896		42,048

Each image of the configured dataset was composed of image data with various frame sizes of 1920 × 1072. Therefore, the skeleton object was extracted after reducing the size to 960 × 536 for computational efficiency in the skeleton data extraction process. The extracted data form was created by connecting the array of each coordinate to the size of the learning frame. A two-dimensional tensor shape was superimposed over time to form a three-dimensional structure, through which changes in motion over time were stored. It was composed of the same form as the queue. The frame size consisted of a minimum of 16 and a maximum of 48 frames, and this size meant an operation of 0.53 to 1.40 s. Data that could not be formatted as data owing to the temporal size of the image, out-of-screen object, omission caused by false detection, etc., were removed. As a result, under-sampling was used because the

learning data size, depending on the motion of each data, had an unbalanced distribution. Hence, 1,000 learning datasets and 360 evaluation datasets were constructed for each motion.

3.3 Motion Classification Using DNN

To classify the difference between each motion using the generated learning data, it is necessary to construct an artificial neural network. The preprocessed motion data are composed of a one dimensional array created by alternately connecting the X and Y coordinates of each object and stacking this configuration horizontally for each frame. Each image is stacked vertically. Therefore, the final data structure has a form in which the object and coordinates of the human body are continuous in a frame structure. When this dataset is called a continuous motion pattern dataset (CMPD), the size of the CMPD is objects in the human body × coordinates × frames.

An effective algorithm is required to classify motion data using CMPD. Because the two-dimensional motion pattern recognition is relatively clear, a simple DNN structure was used. The basic structure of a DNN consists of input, hidden, and output layers. First, a basic unit of motion was entered from the input layer. Then, nodes were configured according to the size of the motion pattern dataset, and batch normalization was performed. The output layer was composed of a total of 15 according to the classification type of the motion, and the final output value was extracted through average pooling. Fig. 4 shows the constructed structure of the motion-classification DNN.

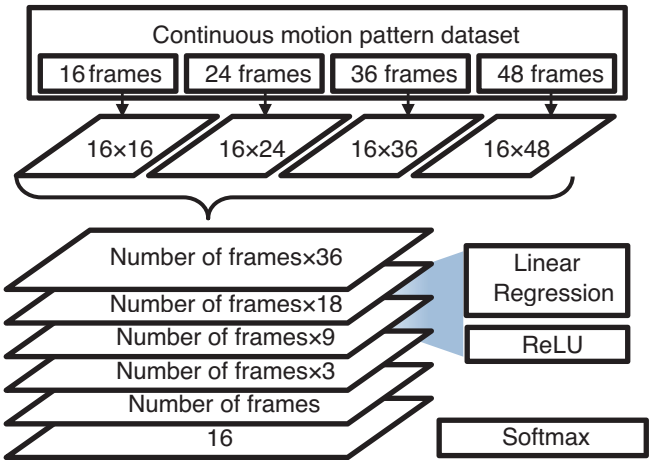


Figure 4: Structure of motion-classification DNN

In Fig. 4, the upper part shows the CMPD classified into four types. The left side shows a case where the frame size is small, while the right side shows the structure of the input layer that receives a large input. Each DNN was constructed according to the input, and the mutual accuracy was compared. The hidden and output layers were configured identically. The preprocessed motion dataset was applied to train the configured neural network, and the weights were calculated until they passed through all layers of the artificial neural network through feedforwarding to calculate the weights between nodes. Next, an optimizer was required for error correction through backpropagation. Stochastic gradient descent (SGD), a representative optimizer, was effective for general neural network operations because it was the simplest and most intuitive [34]. Adaptive moment estimation (Adam), a more improved algorithm, combines momentum optimization [35] and the root mean square propagation (RMSProp) algorithm [36], and shows faster and higher accuracy [37]. However, in some special cases, the SGD shows better results. Therefore, both SGD and Adam were used to compare the results. After the completion of learning, a configuration for

comparing the real-time response performance was required. The response performance mainly depended on the part that composed the real-time image frame into the motion pattern dataset and the performance of the DNN that classified the motion. The part for composing the motion pattern data set consisted of a part for extracting skeleton data for each frame and a part for accumulating data in first in first out (FIFO) format over time. This result was transmitted directly to the input layer of the DNN and configured to extract the result. In addition, each part was configured separately to compare response performance.

4 Result and Performance Evaluation

To evaluate the model, the accuracy and response speed are evaluated. For accuracy, the accuracy improvement depending on the change in frame scale is compared, while for response speed, the average extraction time of the neural network is compared after the initial frame accumulation time. For this, a personal computer with an Intel® i7-10700F and NVIDIA GeForce RTX 3070 and 64 GB of memory was used. For model implementation, Python (Ver 3.8.8) [38] and PyTorch (Ver 1.7.1+cu110) were used [39].

Accuracy refers to the percentage of correct answers for all evaluation results and is the most commonly used method because it is intuitive. Accuracy is calculated using Eq. (1) [40,41].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

In Eq. (1), true positive (TP) is the case where both the actual value and the extraction result are true, and true negative (TN) is the case where both the actual value and the extraction result are negative. False positive (FP) is the case where the actual value is negative, but the result is positive, and false negative (FN) is the case where the actual value is positive, but the result is negative [41]. Accuracy evaluates the change in accuracy according to the comparison frame size. Figs. 5 and 6 show the evaluation results using SGD and Adam at frames.

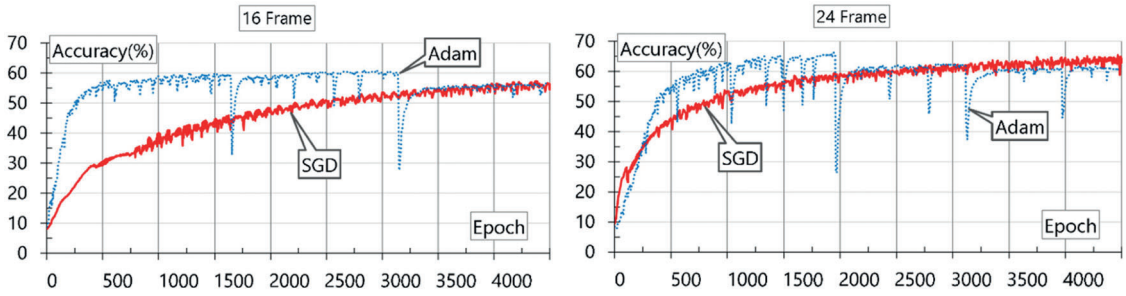


Figure 5: Evaluation results using SGD and Adam at frames 16 and 24

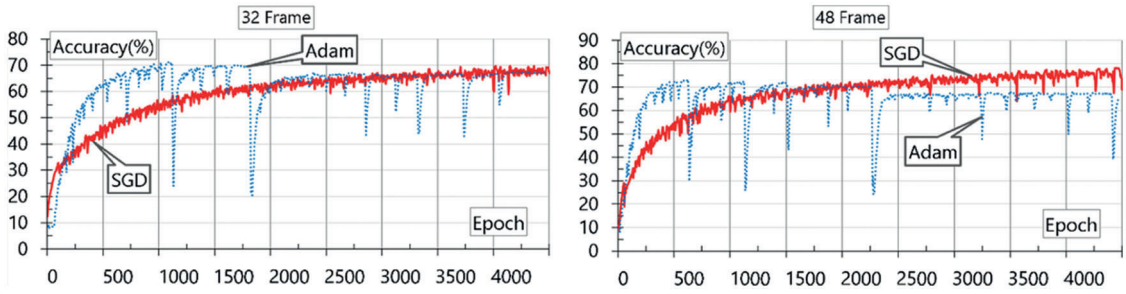


Figure 6: Evaluation results using SGD and Adam at frames 32 and 48

As a result of the evaluation, it can be seen that the maximum accuracy appears when SGD is used, and the frame size is 48. Adam learns fast, but shows an unstable appearance owing to the problem of data size. On the other hand, the larger the frame size, the higher the accuracy, but a frame that is too large has a disadvantage in that it takes a long time to collect image data. Therefore, it is necessary to make appropriate choices for this purpose. In addition, this accuracy shows that the classification of multi-frame human motion using CNN-based skeleton extraction can be utilized in a practical environment. This study aims to operate in practical areas such as city safety and crime prevention using video equipment such as actual CCTV. Therefore, real-time response performance must be guaranteed at a general computing level and high resolution. Because the performance of OpenPose was the same as that of the existing library, it was not evaluated separately, and only the performance of the added DNN was evaluated separately. The configured model was tested in a PC-level performance equipped with one graphics processing unit (GPU), and the test data were evaluated using the same full high definition (FHD) level resolution (1920 × 1080) used for learning. Fig. 7 shows the results of responsive performance evaluation.

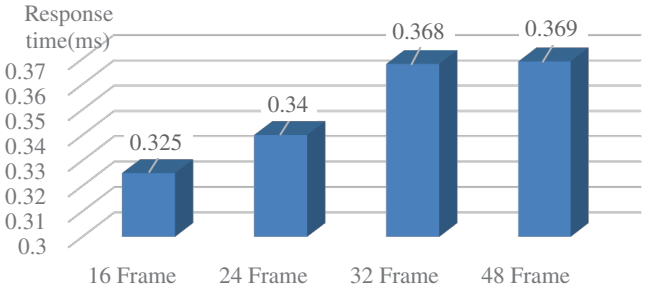


Figure 7: Results of responsive performance evaluation

The average response performance of the constructed model was within 0.37 ms. This result appears to be a level available in general practice areas. However, this result seems to be caused by the use of the evaluation data composed of one person due to the characteristics of the evaluation data. The form of the constructed model is a form in which the amount of computation rapidly increases according to the complexity of the image. Therefore, it should be considered that the performance may be drastically reduced in an image with many real people.

5 Conclusion

The field of human pose estimation can be used for various purposes, from social safety fields such as national defense and police to fields related to physical health, such as medical care and physical education. In particular, the data classified by interpreting various situations have value as information. This study extracts the meaning from continuous video images and detects their behavior. To detect behavior, a method of extracting skeleton data from a 2D image based on CNN was used, based on which the process of classifying behavior into DNN that has learned the basic motion patterns stored in advance is combined. Accuracy was used to evaluate the performance of the constructed model, and the result of the accuracy comparison, the SGD-based 42 frame model, showed more than 70% accuracy. Through this process, it is possible to present information on the operation of the image extraction results. This type of prediction method can be implemented with high accessibility and low cost because it can be used through an image input device such as a basic CCTV without any extra cost. In addition, the constructed model can distinguish various motions according to the expansion of the learning data. Because the internal structure of the model is a skeleton analysis-based method, it can be visually and clearly analyzed compared to an end-to-end method such as SlowFast Network. Because the configured model

can be extracted in real time, information can be transmitted immediately to experts in security and medical care-related fields, which enables effective feedback. However, the constructed model sharply increases the amount of computation according to the group of people on the screen. Thus, the response performance may be lower in a practical environment. Therefore, to realize real-time response performance in the future, it is necessary to study better model configurations. In the future, such a problem could be overcome through continuous research in the field of human pose estimation, and it is expected to be an essential element combined with various social infrastructures.

Funding Statement: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (Grant No.: NRF-2020R1A6A1A03040583).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz *et al.,* "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, pp. 1–20, 2021.

[2] H. Yoo, R. C. Park and K. Chung, "IoT-Based health Big-data process technologies: A survey," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 15, no. 3, pp. 974–992, 2021.

[3] K. Morimoto, A. Ardelean, M. Wu, A. C. Ulku, I. M. Antolovic *et al.,* "Megapixel time-gated SPAD image sensor for 2D and 3D imaging applications," *Optica*, vol. 7, no. 4, pp. 346–354, 2020.

[4] D. Tabernik and D. Skočaj, "Deep learning for large-scale traffic-sign detection and recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1427–1440, 2019.

[5] J. -C. Kim and K. Chung, "Prediction model of user physical activity using data characteristics-based long short-term memory recurrent neural networks," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 13, no. 4, pp. 2060–2077, 2019.

[6] K. Singh and P. C. Jain, "Traffic control enhancement with video camera images using AI," in *Optical and Wireless Technologies*, Springer, pp. 137–145, 2020.

[7] E. Valero, A. Sivanathan, F. Bosché and M. Abdel-Wahab, "Analysis of construction trade worker body motions using a wearable and wireless motion sensor network," *Automation in Construction*, vol. 83, pp. 48–55, 2017.

[8] S. Díaz, J. B. Stephenson and M. A. Labrador, "Use of wearable sensor technology in gait, balance, and range of motion analysis," *Applied Sciences*, vol. 10, no. 1, pp. 234, 2020.

[9] U. Muhammad, W. Wang, S. P. Chattha, and S. Ali, "Pre-trained VGGNet architecture for remote-sensing image scene classification," in *2018 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, pp. 1622–1627, 2018.

[10] V. Rathod, R. Katragadda, S. Ghanekar, S. Raj and P. KolliparaI *et al.,* "Smart surveillance and real-time human action recognition using OpenPose," in *ICDSMLA 2019*, Springer, pp. 504–509, 2020.

[11] G. Li, S. K. S. Hari, M. Sullivan, T. Tsai, K. Pattabiraman *et al.,* "Understanding error propagation in deep learning neural network (DNN) accelerators and applications," in *Proc. of the Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, Denver, Colorado, USA, pp. 1–12, 2017.

[12] C. -H. Chang, C. -N. Chou and E. Y. Chang, "Clkn: Cascaded lucas-kanade networks for image alignment," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, Hawaii, USA, pp. 2213–2221, 2017.

[13] J. Tanaś and A. Kotyra, "Comparison of optical flow algorithms performance on flame image sequences," in *Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2017*, Bellingham, Washington, USA, vol. 10445, pp. 104450 V, 2017.

[14] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Int. Conf. on Machine Learning*, Long Beach, California, USA, pp. 6105–6114, 2019.

[15] Z. W. Yuan and J. Zhang, "Feature extraction and image retrieval based on AlexNet," in *Eighth International Conference on Digital Image Processing (ICDIP 2016)*, Chengu, China, vol. 10033, pp. 100330E, 2018.

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.,* "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[17] C. Chen, M. -Y. Liu, O. Tuzel and J. Xiao, "R-CNN for small object detection," in *Asian Conf. on Computer Vision*, Taipei, Taiwan, pp. 214–230, 2016.

[18] R. Girshick, "Fast r-cnn," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Santiago, Chile, pp. 1440–1448, 2015.

[19] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[20] H. Lin and S. Jegelka, "Resnet with one-neuron hidden layers is a universal approximator," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, Montréal Canada, pp. 6172–6181, 2018.

[21] F. Sultana, A. Sufian and P. Dutta, "Advancements in image classification using convolutional neural network," in *2018 Fourth Int. Conf. on Research in Computational Intelligence and Communication Networks (ICRCICN)*, Kolkata, India, pp. 122–129, 2018.

[22] G. R. Kotapalle and S. Kotni, "Security using image processing and deep convolutional neural networks," in *2018 IEEE Int. Conf. on Innovative Research and Development (ICIRD)*, Bangkok, Thailand, pp. 1–6, 2018.

[23] A. de Souza Brito, M. B. Vieira, S. M. Villela, H. Tacon, H. de Lima Chaves *et al.,* "Weighted voting of multi-stream convolutional neural networks for video-based action recognition using optical flow rhythms," *Journal of Visual Communication and Image Representation*, vol. 77, pp. 103112, 2021.

[24] W. Fang, L. Wang and P. Ren, "Tinier-YOLO: A real-time object detection method for constrained environments," *IEEE Access*, vol. 8, pp. 1935–1944, 2019.

[25] C. Feichtenhofer, H. Fan, J. Malik and K. He, "Slowfast networks for video recognition," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Seoul, Korea, pp. 6202–6211, 2019.

[26] L. Davis, P. Torr, S. C. Zhu, https://cvpr2019.thecvf.com, 2019.

[27] G. Huang, S. Liu, L. Van der Maaten and K. Q. Weinberger, "Condensenet: An efficient densenet using learned group convolutions," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, USA, pp. 2752–2761, 2018.

[28] Y. Cheng, D. Wang, P. Zhou and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[29] M. Tan, R. Pang and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, Washington, USA, pp. 10781–10790, 2020.

[30] D. Osokin, "Real-time 2d multi-person pose estimation on CPU: Lightweight OpenPose," in *ICPRAM 2019-Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*, Prague, Czech Republic, pp. 744–748, 2019.

[31] 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, 2019.

[32] M. Mateen, J. Wen, S. Song and Z. Huang, "Fundus image classification using VGG-19 architecture with PCA and SVD," *Symmetry (Basel)*, vol. 11, no. 1, pp. 1, 2019.

[33] T. -Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona *et al.,* "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*, Zurich, Switzerland, pp. 740–755, 2014.

[34] N. Ketkar, "Stochastic gradient descent," in *Deep Learning with Python*, Springer, Apress, Berkeley, California, USA, pp. 113–132, 2017.

[35] R. Kidambi, P. Netrapalli, P. Jain and S. Kakade, "On the insufficiency of existing momentum schemes for stochastic optimization," *in*, *2018 Information Theory and Applications Workshop (ITA)*, San Diego, California, USA, pp. 1–9, 2018.

[36] F. Zou, L. Shen, Z. Jie, W. Zhang and W. Liu, "A sufficient condition for convergences of adam and rmsprop," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Long Beach, California, USA, pp. 11127–11135, 2019.

[37] I. K. M. Jais, A. R. Ismail and S. Q. Nisa, "Adam optimization algorithm for wide and deep neural network," *Knowledge Engineering and Data Science*, vol. 2, no. 1, pp. 41–46, 2019.

[38] Python, https://www.python.org, 2021.

[39] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury *et al.,* "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8026–8037, 2019.

[40] H. Jung and K. Chung,, "Social Mining based Clustering Process for Big-data Integration," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no.1, pp. 589–-600, 2021.

[41] H. Yoo and K. Chung, "Deep learning-based evolutionary recommendation model for heterogeneous big data integration," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 14, no. 9, pp. 3730–3744, 2020.