Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039263
Article

Deep Learning-Based Action Classification Using One-Shot Object Detection

Hyun Yoo[1], Seo-El Lee[2] and Kyungyong Chung[3,*]

[1]Contents Convergence Software Research Institute, Kyonggi University, Suwon-si, 16227, Korea
[2]Department of Public Safety Bigdata, Kyonggi University, Suwon-si, 16227, Korea
[3]Division of Computer Science and Engineering, Kyonggi University, Suwon-si, 16227, Korea
*Corresponding Author: Kyungyong Chung. Email: dragonhci@gmail.com
Received: 18 January 2023; Accepted: 02 June 2023; Published: 30 August 2023

Abstract: Deep learning-based action classification technology has been applied to various fields, such as social safety, medical services, and sports. Analyzing an action on a practical level requires tracking multiple human bodies in an image in real-time and simultaneously classifying their actions. There are various related studies on the real-time classification of actions in an image. However, existing deep learning-based action classification models have prolonged response speeds, so there is a limit to real-time analysis. In addition, it has low accuracy of action of each object if multiple objects appear in the image. Also, it needs to be improved since it has a memory overhead in processing image data. Deep learning-based action classification using one-shot object detection is proposed to overcome the limitations of multi-frame-based analysis technology. The proposed method uses a one-shot object detection model and a multi-object tracking algorithm to detect and track multiple objects in the image. Then, a deep learning-based pattern classification model is used to classify the body action of the object in the image by reducing the data for each object to an action vector. Compared to the existing studies, the constructed model shows higher accuracy of 74.95%, and in terms of speed, it offered better performance than the current studies at 0.234 s per frame. The proposed model makes it possible to classify some actions only through action vector learning without additional image learning because of the vector learning feature of the posterior neural network. Therefore, it is expected to contribute significantly to commercializing realistic streaming data analysis technologies, such as CCTV.

Keywords: Human action classification; artificial intelligence; deep neural network; pattern analysis; video analysis

1 Introduction

Artificial intelligence (AI) systems have been successfully applied to various fields. Deep learning-based image analysis technology shows high accuracy in image recognition and classification and has been used in diverse areas, including medical services, traffic services, and crime prevention.

In particular, accurate and quick image analysis technology is in high demand where objects and actions must be detected in real-time, such as traffic services and crime prevention areas. In the industry, image analysis handles streaming-based massive video data as big data. Likewise, big data is processed in the area of human action classification in an image [1]. A video is not a simple still image but continuous data with time. When such video images are processed, the response speed is slow. Research on recognizing human actions in video data is difficult because finding moving humans and comprehensively recognizing human actions through temporal and spatial considerations of surrounding persons, objects, actions, or situations are necessary [2]. In action detection, it is necessary to analyze human actions, complex movements of human joints, and interactions between various objects, resulting in difficulties during the analysis. OpenPose is a pose estimation algorithm for action recognition [3]. In the past, the top-down method of capturing a human and detecting a person's pose has been used. In a work that adopted the approach of Park et al., the bottom-up method is applied to estimate the joints (feature points) of a person captured in an image, analyze the correlations between these points, and predict the person's pose [4]. Nevertheless, this method has low accuracy for unusual poses and needs to estimate a pose in the overlapped data of multiple persons [5].

The SlowFast network model [6], a typical algorithm of a two-stream network structure, is an action analysis model with excellent performance. It aims to detect various situations and actions using continuous two-dimensional (2D)-image-based data. In work by Shi et al. [7], SlowFast connects and learns multiple frames associated back and forth and uses the two-stream network structure to combine two algorithms: slow and fast pathways. The slow pathway analyzes the overall conditions and situations of the image, whereas the fast pathway captures dynamic actions. The action analysis algorithm based on two-stream networks analyzes continuous image data with one or two separate human objects, presents the effect of the divide-and-conquer algorithm [8], and works relatively fast. However, if multiple humans continue to be extracted from multiple frames, the model exhibits a prolonged response speed. When conventional action detection algorithms are analyzed, the methods proposed by da Silva et al. [9] and Ullah et al. [10] showed significant improvements in response speed but low accuracy. The models presented by Lai et al. [11] and Shou et al. [12] offered high accuracy but low response speed [13].

This work aims to analyze human actions at low operational costs and detect each human action captured in an image. In this work, a one-shot object detection algorithm, which has a high response speed and excellent accuracy, is applied to detect the shape of the human body composed of human body-action pairs all at once in continuous single image frames. Subsequently, the multi-object tracking algorithm continues to track the same person. In this process, an artificial neural network is used to save the human body action pattern by frame and digitalize the pattern change. By classifying these patterns, human actions are analyzed based on the pattern change of the connected actions.

Fig. 1 shows the overall proces of the proposed model. For the performance evaluation of the accuracy and response speed, the human action video dataset of the AI Hub [14] made by the National Information Society Agency is used. Therefore, analyzing images and detecting specific anomalies in real-time in a CCTV-based image-control situation is possible. The video control system based on the designed model is applied to diverse areas and can bring economic and industrial ripple effects in urban safety, policing, national defense, and transportation. It is expected to be critical in the social safety monitoring system. The contribution of the proposed method is as follows.

- The accuracy and speed demonstrate excellent performance, even with limited training data. This suggests that efficient learning is achievable even when datasets are constrained.

- The proposed model achieves practicality and usability by developing a concise and efficient model while excluding complex structures or parameters.
- Our proposed model exhibits a high generalization capability even for untrained poses. The model can effectively recognize and classify new data based on learned patterns. Therefore, it demonstrates stable performance in diverse situations beyond a specific dataset.

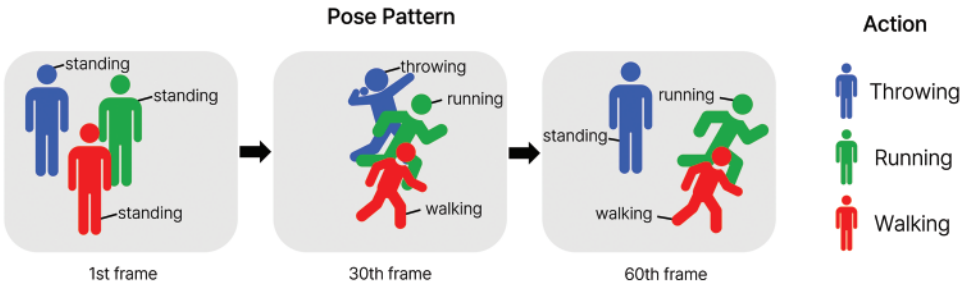

Figure 1: The overall process of the proposed model

2 Related Works

2.1 Technological Trends in the Video Action Classification Model

Human action recognition in videos has been actively investigated. Traditionally, actions have been recognized by extracting feature points from the actions frequently arising in an image, drawing a specific vector, and applying a pattern classifier, such as Support Vector Machine (SVM) [15], AdaBoost [16], or random forest. Obtaining external factors, such as domain knowledge, is necessary to develop an effective model. Considering this, along with technological development, deep learning-based algorithms specialized for video processing are used to recognize actions [17].

An artificial neural network refers to a network of artificial neurons with input and output layers, which simulates the human brain delivering signals through a massive connection of neurons [18]. As a primary classification technique, a decision tree is a tool for classifying data using the influential variables on classification and the reference values of classification [19]. However, the device can generate a significant error for the data value approximant the boundary of a classification reference value, making it unstable. Moreover, it is challenging to classify new data because it is hard to determine the effect of each predictor variable. A multilayer neural network is applied to solve the problem of adding multiple hidden layers to the input and output layers [20]. It solves complex classification, nonlinear, and numerical prediction problems. Using the activation function, the function to execute optimal classification is obtained. In various studies, Convolutional Neural Networks (CNN) models have been applied based on image frames to classify actions [21]. Compared to the techniques of manually extracting actions, this method helps improve accuracy. The 3D CNN improves accuracy by receiving 3D data as input and processing multiple frames simultaneously [22]. However, this method requires considerable memory during the operation process and takes a long time. Therefore, various ways to improve image models' training speed have been investigated. The attention mechanism, which starts with natural language processing, gives weight to the information necessary for image processing and classifies actions [23]. A transformer based on attention collects contextual information from different objects of neighboring images and classifies the object's action to recognize [24]. Despite sufficient training data, this must be appropriately performed for all action classes and accurately acknowledge a small object's action. Currently, a deep learning-based skeleton

analysis algorithm tends to be combined with a Recurrent Neural Network (RNN) and Long-Short Term Memory (LSTM) to consider the temporal factor of actions [25]. Zou et al. [26] proposed the Adaptation-Oriented Feature (AOF) projection for one-shot action recognition, which aims to recognize actions in unseen classes with only one training video. The AOF projection involves pre-training the base network on seen classes and projecting the important and adaptation-sensitive feature dimensions into the adaptation-oriented feature space. This approach achieves both improved adaptation performance for highly variable actions and mitigates the computational complexity associated with deep networks. Zhong et al. [27] proposes a graph complemented latent representation (GCLR) for applying meta-learning. GCLR embeds the representation into a latent space and reconstructs the latent codes using variational information to enhance generalizability. Additionally, it incorporates a graph neural network (GNN) to improve performance by considering the relationships between samples. Peng et al. [28] discuss skeleton-based one-shot action recognition (SOAR), which explicitly addresses occlusions. They generate diverse forms of occlusions using realistic furniture models. Additionally, they introduce a new transformer-based model called Trans4SOAR to mitigate the adverse effects of occlusions and achieve superior performance by leveraging multiple data streams and a mixed-attention fusion mechanism.

Video data are time-series data measured at specific time intervals. Time-series data have patterns in terms of tendency, seasonality, periodicity, autocorrelation, and white-nose [29]. To analyze them, various techniques can be used, such as the Auto Regressive (AR) model [30], Moving Average (MA) model [31], and Auto-Regression Moving Average (ARIMA) model [32]. AR can find past data that influence the current point of time among time-series data, and it explains the value of the current issue with the matter before the specific case. MA presents the data at the current point of time through the linear combination of the finite number of white noises and predicts the current issue using the forecast error. ARIMA signifies the current end of time by converting or differencing time-series data and applying AR and MA models.

2.2 Limitations and Trends in Traditional Action Classification Models

The previous work of Feichtenhofer et al. [6] on slow-fast-based action classification focused on the actions of objects in 3D image data are classified. This slow model simulates the human visual system. Based on a two-stream network, it does not use optical flow but uses only an image as the input. Therefore, it is possible to make end-to-end learning and recognize the action of an object faster. SlowFast comprises the slow pathway to process semantic information and the fast pathway to process information of quickly changing actions. A lateral connection combines the feature maps drawn by the two pathways. The internal structure of each pathway is based on a convolution network. The slow pathway algorithm works more effectively by analyzing continuous image data separated as one object. Using a high frame rate, the fast pathway accounted for up to 20% of the total calculation. This also reduces the calculation cost by reducing the number of image channels. These two pathways are later connected laterally and recognize actions by processing the original image at different speeds [33]. It works more effectively by analyzing continuous image data separated as one object. This method has high accuracy and fast response performance. However, if an object is small or considerably large, it has low recognition accuracy because it sees the entire screen as one input frame.

Carreira et al. [34] proposed an action classification model based on an Inflated 3D ConvNet (I3D) using the kinetics human action video dataset. The model is pre-learned with a large dataset (over 400 data points) beyond a small dataset. The method applies the inflation technique of changing the 2D CNN pre-learned with the ImageNet dataset into a 3D CNN. The 3D filter represents the addition of a time axis to the 2D filter. The weight value of the 2D filter is copied along the time axis of the 3D

filter. After expanding the filter's dimensions, its weight is divided by 1/N. Using an inception module, I3D consists of two models: a model that receives optical flow to predict better action information and a model that gets the RGB image as input for training. Action is expected after calculating the average of the prediction results of these two models. The models using two-stream networks must calculate the optical flow in advance. Thus, they require ample memory space and a long time [35].

Yoo et al. [36] proposed a deep-learning action classification model based on a skeleton analysis algorithm. It makes the skeleton pattern algorithm lightweight, classifies actions, and recognizes human actions in real time. The proposed method uses the CNN-based VGGNet to establish a neural network for human skeleton detection. It predicts the positions of human joints and main action parts in an image using the training and extraction results of the neural network. The skeleton coordinates are extracted utilizing the preference analysis and confidence map. Because the proposed method can generate input data using an image input device without additional costs, it can be carried out with high accessibility and low calculation cost. However, the more objects appear in the image, the more the calculation amount increases. Accordingly, it exhibits low response performance. In addition, in the case of the latest object detection models, relatively large learning data is used, resulting in a limitation in the increase in computational complexity. Also, while using deep-layer neural networks to show high accuracy, there is a limitation in that it is difficult to detect objects in real-time due to slow response speed.

3 Deep Learning-Based Action Classification Using One-Shot Object Detection

The technical principles of the method proposed in this work are to detect the human body-action data at once in a single image frame using one-shot object detection [37] with the best response and accuracy and to classify actions using the time series data deep learning model. The proposed method has a high response speed and can classify actions only by training action patterns without additional image data-based training. Thus, it is necessary to learn based on classified human body-action image data rather than in the unit of objects. A multi-object tracking algorithm tracks the same person and prepares for a situation where multiple human objects are captured. The human body's action patterns by frame are saved in this process, and the changing patterns are digitalized. It is possible to analyze human-connected actions by classifying these patterns.

3.1 Composition and Technical Principles of the Real-Time Action Classification Model

For human action analysis, human actions captured by an image-monitoring device are detected in a three-step process. The first step is object detection, where the coordinates of the human body-action object are obtained from a single image frame. The second step is multi-object tracking, in which the position of the same object is tracked using the obtained coordinates to collect human body-action patterns. The last step is pattern analysis, in which the collected action patterns are analyzed, and actions are classified. Fig. 2 shows the structure of deep learning-based action classification using one-shot object detection.

CNN-based algorithms, such as Single-Shot Multi-box Detector (SSD) [38] and RetinaNet [39], have been used as one-shot object detection for the first step. In this work, the YOLO-based one-shot object detection is expanded for use. You Only Look Once (YOLO) [40] is a considerably fast object detection algorithm and is the object detection model with the highest response speed. It has high accuracy when detecting a new image that does not appear in the training step. The selected object detection model must learn differently based on human actions and detect action objects in an image. It is the most significant difference from general object detection models that extract the number of

objects through simple object detection [41]. Object detection removes only the number of objects by determining a pattern from an image; therefore, it is necessary to use an algorithm to determine whether the object detected in continuous frames is the same human body.

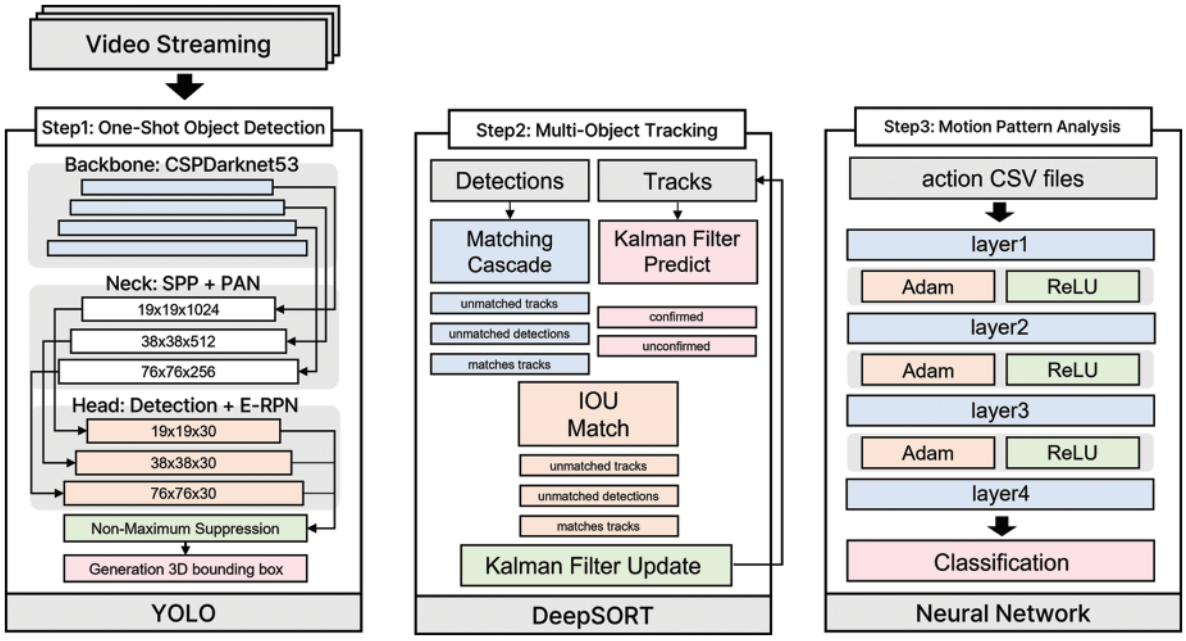

Figure 2: Structure of deep learning-based action classification using one-shot object detection

The multi-object tracking algorithm by DeepSORT [42] et al. is used to determine the linearity and homogeneity of an object in an image and whether the detected object is the same human body in the second step. If the data are used, it is possible to collect pattern changes in actions by the human body of the same person based on the human body-action object [43]. The extraction results of the object detection model are the coordinates of an image box and a human body-action pair. In DeepSORT, tracking the image of an object related to the human body is necessary. Therefore, the object detection model uses only the human body parts of multiple object images as the input data.

Pattern analysis is conducted to analyze the collected patterns. The pattern consists of the class number for each frame extracted using YOLO and DeepSORT when the video is input. In this process, the pattern of actions generated in an accumulated manner is classified, and the action is determined. In this case, the accumulated size changed over the action sampling time. The larger the size, the slower the response speed but the higher the accuracy. Pattern analysis algorithms are applied differently depending on their purpose, from simple similarity evaluation based on Euclidean distance to time-series prediction based on LSTM and RNN [44]. For this reason, it is crucial to select an appropriate object based on the object-action classification size. In this work, an overlapping neural network is applied. In this process, it is possible to analyze the pattern of the accumulated actions of the human body of the same person according to image changes, determining the situation of a person's action. It is necessary to use the human body-action pattern training data by action for a standard action model and to perform training and evaluation processes. The pattern of each action can have a different length. Thus, it is possible to use a time-series prediction algorithm [45]. Models, such as LSTM or RNN, or simple models, such as Deep Neural Network (DNN), are used. This work uses

a relatively deeply overlapped neural network to analyze an object's action pattern. Fig. 3 shows the three continuous frames used to illustrate the general principles.

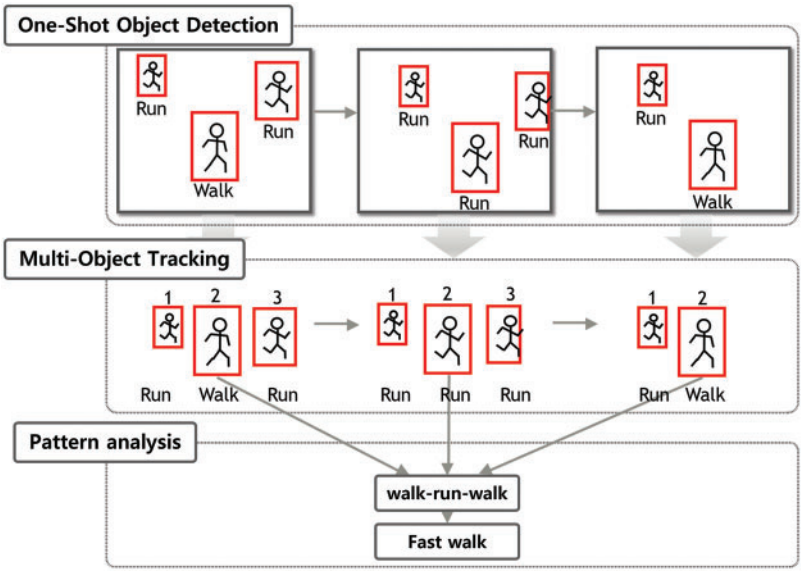

Figure 3: The technical principles of the proposed model

In Fig. 3, object detection captures three humans in frames 1 and 2 and detects two humans in frame 3. The human body image in the center of the first frame indicates that it is seen with a pre-learned walk pose. Object detection enters the human body object image detected as a quadrangle into the multi-object tracking algorithm whenever a frame is input. Multi-object tracking algorithm gives each human object an object number, which is a random number, identifies the identity of the human object entered in the next frame, and notifies the previously assigned object number if it is the same object as before. For example, in Fig. 5, the centralized human body image in frame 1 shows that object 2 is assigned. Pattern analysis combines and compares the poses of the same object based on the progression of successive frames. For example, in the case of a human body with object 2 shown in Fig. 5, the pattern changes to a walk, run, and walk pose as the frame progresses. According to this pattern change, pattern analysis shows that the human body of object number 2 performs a fast walk action.

3.2 Configuration and Training of One-Shot Object Detection

This work uses the One-Shot object detection method to learn models and detect objects. We train with many public datasets for the commonly used Few-Shot object detection method. Moreover, it aims to detect new objects with few annotated instances. On the other hand, One-Shot object detection is the same as Few-Shot object detection in learning with a vast amount of public datasets. However, there is a difference in that annotated instances aim to detect new objects that are only one. In addition, there is an advantage of showing a high real-time performance by simultaneously predicting an object's bounding box and class through a single neural network for an input image.

The YOLOv4 model [40] uses a resolution of 512×512 pixels to increase its ability to detect small and diverse objects. Furthermore, they use more receptive fields to handle a larger resolution. The CSPDarknet53 backbone extracts feature from the images. A Spatial Pyramid Pooling network (SPP)

and Path Aggregation Network (PAN) is used as the neck to summarize the features. A new pooling technique makes it possible to use a specific image size. PAN shortens the information path between the convolutional layers and enhances the information flow of the framework. After that, the algorithm finds a bounding box and classifies the object in the image. Fig. 4 shows the process of using YOLOv4 as a one-shot object detection.

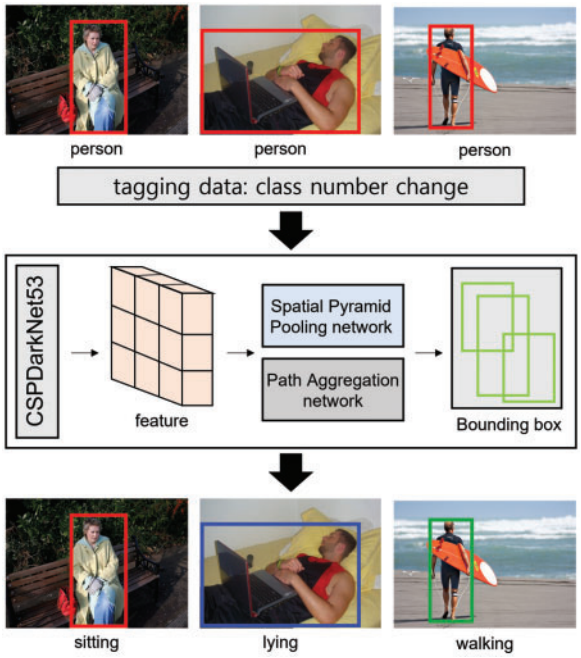

Figure 4: Structure and training of one-shot object detection

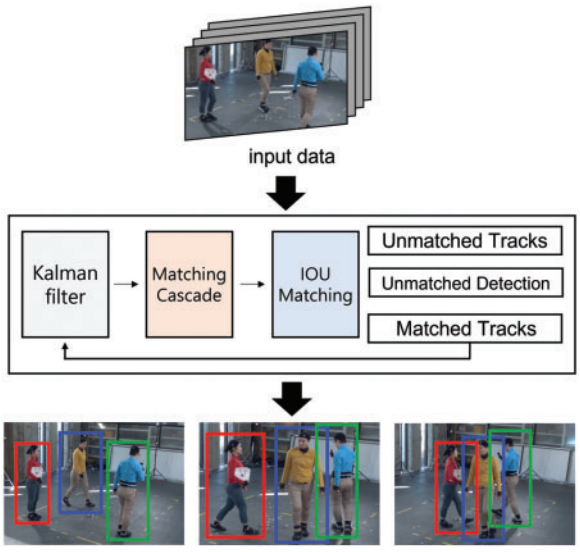

Figure 5: Structure of DeepSORT

The directory of the COCO dataset comprises a dataset for training and a dataset for testing. Each dataset consisted of an image and label. A label is a text file that includes the class number for each image and the x and y coordinates of the center point, width w, and height h. In this work, the algorithm learns new actions based on the criteria and freshly designates the class number for each action. The current COCO dataset saves the class numbers for categories 0–79. Thus, the class numbers and poses after 80 are added. Therefore, it is necessary to change the class number 0 entered as the actual 'person' of the COCO dataset to the class number for a related action. Table 1 lists the final classified actions.

Table 1: Image classification criteria

Class	Pose	Definition of pose	Count
0	unknown	Undefined pose	4,000
1	Throwing pose	A pose of reaching toward the air with an object in one's hand	1,138
2	Lying pose	A pose with the back of the body facing the surface	679
3	Sitting pose	A pose that attaches one's hip to the surface and the upper body, maintaining 90 degrees with the surface	4,887
4	Standing pose	A pose where the person stands upright with both feet on the ground	5,366
5	Walking pose	A pose that stretches one's upper body has a small stride, and the thigh and knee height is significantly lower than running	1,303
6	Running pose	A pose that bends one's upper body has a giant stride, and the thigh and knee heights are higher than walking	1,108
7	Punching pose	Stretching one's arms toward the object or the air with one's fists clenched	354

In Table 1, the class represents the class index of the actual COCO dataset, and pose refers to the name of a classified pose. Posture refers to the detailed definition of a classified pose that expresses the shape of the human body that appears in an image. Mainly, an unknown pose is used to maintain the continuity of the action pattern. A posed image consisting of undefined actions is classified as an Unknown pose.

A total of 18,835 images are used for the training. In the case of ambiguous actions accounting for over 70% of all data, only 4,000 images are used to solve the data imbalance, and 14,835 actions classified based on objective criteria are used. The running and punching pose data that lack training is additionally collected for training through crawling. In a frame with an ambiguous pose, even if an action fails to be recognized, the object of the pose must be recognized as a 'person.' Therefore, an unknown class number, 0, is also used for training. It required 9.418 h to learn, and the epoch size is 200.

3.3 Tracking and Action Vector Extraction of Objects for Continuous Action Detection

In general, CCTV images. Multiple people move with different actions. Tracking each person and analyzing their changes is necessary to classify continuous actions. DeepSORT is used for the object identity of each frame. DeepSORT applies the Re-IDentification (ReID) model to solve the problems of conventional SORT, such as object occlusion and ID switching. In addition, matching cascade logic is added to traditional SORT for more accurate tracking. Fig. 5 shows the basic process of DeepSORT.

The Kalman filter is applied to predict and measure the position of the object to be connected to the next frame. Based on the prediction and measurement results, the object state is extracted by a matching cascade. Matching cascade uses cosine distance to extract a detailed estimate of an object's position. The Hungarian algorithm determines whether the object in the previous and next frames are identical. The Kalman filter is updated if the object tracked in the process is detected. A new object is provisionally classified and tracked if it appears thrice. If a tracked object is not detected for a certain period, it is excluded from the tracking procedure using the time_since_update variable. If the tracked object is not found again, it is set to the tentative state and recorded as not found in the variable. It is excluded from tracking if it exceeds a certain number of times. If the object is found again, the variable is initialized to zero, and the object is tracked likewise. DeepSORT tracks the human body image and extracts a pose via one-shot object detection, saving the pose of each human body object over frames by AI.

The video dataset for pattern analysis and performance evaluation is the human action video dataset of the AI Hub created by the National Information Society Agency [14]. Because each operation has more than 1,500 videos, 500 randomly extracted functions are used for pattern analysis. Another 500 actions are used for the performance evaluation. The data are classified into 50 action types. This dataset is not labeled. Thus, the action name of the video is used as a label. The dataset includes complex actions such as hugging and crossing the arm. Therefore, we use six clearly defined actions: walking, running, sitting, falling, clapping, and going up the stairs. Three thousand video files are used to extract the CSV file for the class numbers of the action data. The saved data are presented in Fig. 6.

	0 frame	1 frame	2 frame	3 frame	...	62 frame	63 frame
Video 1	4	5	4	4	...	5	5
Video 2	5	5	5	5	...	5	0
Video 3	3	3	3	3	...	3	3
Video 4	2	2	2	2	...	5	5
Video 500	4	3	4	3	...	0	3

Figure 6: Structure of generated pattern CSV data file

For the action patterns in the image data, the constant vector value of the softmax value in the output layer of the neural network in one-shot object detection is used. The action of an object in a video can be captured in a short time. Therefore, 64 frames within 2–4 s after the video start is used. Therefore, the data vector is designed using 64 repeated time-series datasets. These are used by normalizing to a 0 to 1 for pattern classification. The first row shows the frame index.

3.4 Action Classification Using Artificial Neural Network

An artificial neural network is used for training and classification to determine the pattern of an object's actions. A deep neural network is applied to the classification of action patterns. As artificial neural networks, deep neural networks have hidden layers between the input and output layers. It is necessary to prevent the network from being too deep to solve problems with artificial neural networks, such as the reduction in operation, memory overhead, and overfitting. To solve the problem of gradient vanishing in general artificial neural networks, the ReLU function is used. In addition, the optimizer Stochastic Gradient Descent (SGD) is used to determine the optimal weight value and the minimum point of the loss function value. Fig. 7 shows the structure of the neural network used in this work.

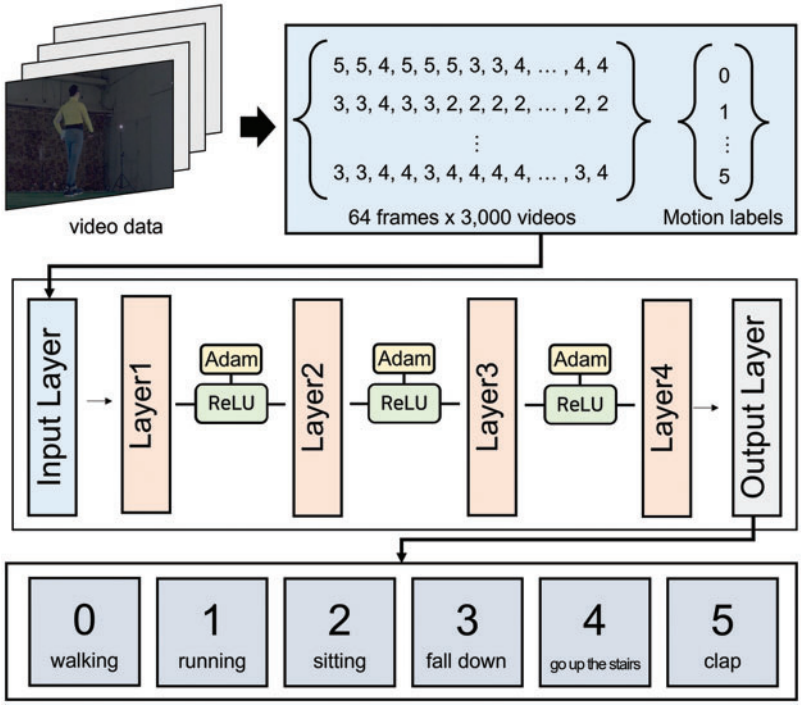

Figure 7: Structure of action classification using neural network

For the neural network training, the class number of the action image by the object extracted with one-shot object detection and DeepSORT is normalized in the CSV file. Then, action pattern files are generated for training. For the actions used for classification, 3,000 data points created in the two steps mentioned above are divided at a ratio of 7:3. Thus, 2,100 and 900 data points are used for training and testing, respectively. They are categorized as walking, running, sitting, falling, clapping, and climbing stairs. The neural network consisted of four layers. As the hyperparameters for training, the batch size, learning rate, and epoch are set to 32, 0.02, and 180, respectively.

4 Result and Performance Evaluation

The action detection results obtained using YOLO are shown in Fig. 8. When the walking validation video data are entered, the object is not detected by the human class number 0 but by the action class number corresponding to the action.

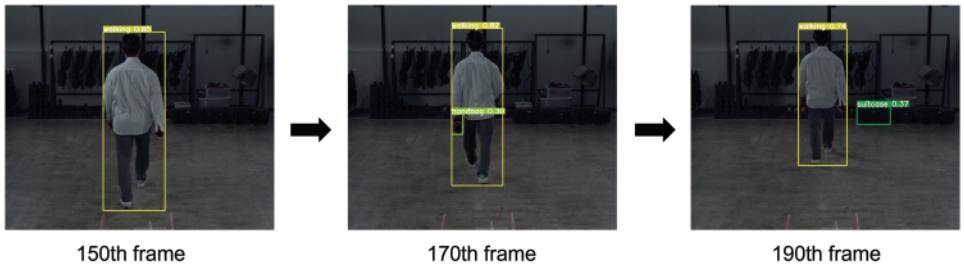

Figure 8: Action detection result using trained YOLO

For the evaluation of the model, its accuracy, the number of parameters, and response speed are evaluated. It is compared with the Slowfast model [6] and the Openpose model [3] for relative evaluation. The SlowFast and OpenPose models have widely used action detection and classification techniques, making them suitable for proposing and comparing performance. As previously mentioned, the evaluation data used part of the dataset for pattern analysis. For evaluation, it was cut into 64 frames, corresponding to the actual action of the entire video. Therefore, in evaluating the designed model, it is possible to assess it relatively using the same image data. The hardware system used for the neural network was composed of Intel® i9-9900K, 16 GB memory, and NVIDIA GeForce RTX 3090, which consisted of Python (Ver 3.10) and PyTorch (Ver 1.12.1+cu11.3).

Actions with the highest accuracy are evaluated based on the action classification data extracted from the video images. Accuracy is based on the confusion matrix. The confusion matrix represents the accuracy rate of the total evaluation results. The accuracy is intuitive that it is generally used.

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

In Eq. (1), True Positive (TP) means that the actual value and the extraction result are positive, True Negative (TN) means that the actual value and the extraction result are negative, False Positive (FP) means that the actual value is negative, but the extraction result is positive. False Negative (FN) means that the actual value is positive, but the extraction result is negative. In this equation, the more accurate the model, the closer its accuracy value is to 1. Fig. 9 shows the average accuracy results.

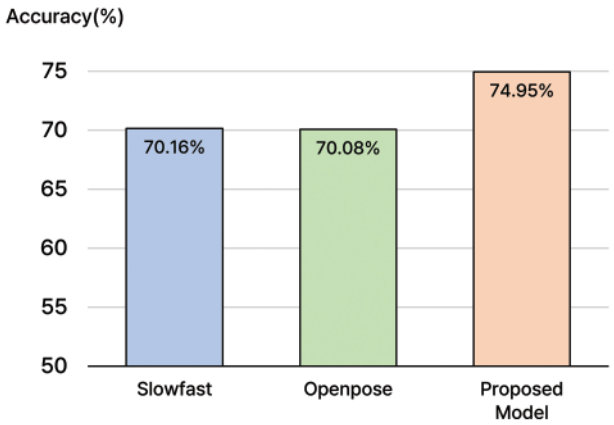

Figure 9: Action classification average accuracy result

Fig. 11 shows the accuracy results and the training of the time-series data of the deep learning model. The average accuracy results show that all existing action classification models offer more than 70% accuracy. The Slowfast model shows 70.16% accuracy, and the Openpose model shows 70.08% accuracy. The accuracy of the proposed model is 74.95%, more than 4% higher than the existing models. This demonstrates superior classification accuracy compared to existing models.

Fig. 10 shows the classification accuracy of each action dataset. Directly associated actions mean that the learned posed image data includes actions equal to those in the training images. The average classification accuracy of directly associated actions was approximately 76.31%. Regarding accuracy, sitting, running, and walking were 79.11%, 77.00%, and 74.33%, respectively. The high accuracy is because the action data for the one-shot object detection include actions equal to those in the training images.

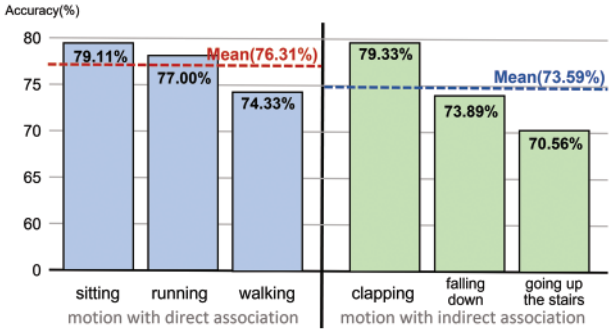

Figure 10: Each action classification accuracy result

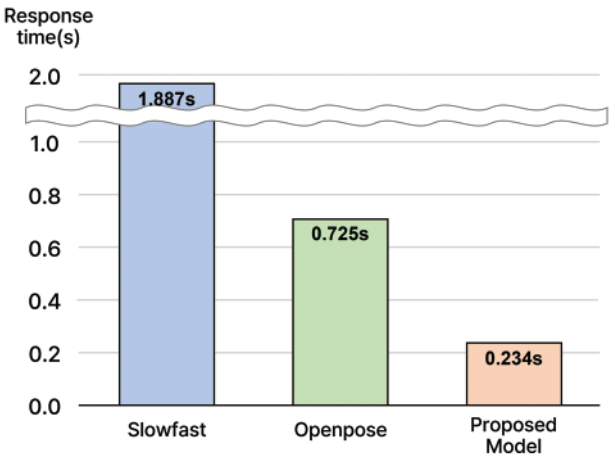

Figure 11: Comparison of response time with the existing models

Indirectly associated actions imply that the learned-to-posed image data do not include actions equal to those in the training images. The average classification accuracy of indirectly associated actions was 73.59%. In terms of accuracy, the accuracy of clapping was 79.33%. The model focuses on the standing pose rather than the hands since the clapping action appears in an image of a standing pose. It recognizes the action as standing in the way it learns as the standing action of the training data image. For this reason, it has high accuracy. The action of falling, which was not included in the trained action data, had an accuracy of 73.89%. The action of going upstairs had an accuracy of

70.56%. Compared to the actions of the trained data images, they have low accuracy. Given that all indirectly associated actions have a high classification accuracy of over 70%, it is possible to classify a specific action pattern without separate training.

To verify the performance evaluation for memory overhead, the proposed model is compared the number of parameters with the existing models. Because the number of parameters and the memory overhead are proportional, the smaller the number of parameters, the smaller the memory. All models used for performance evaluation do the COCO dataset and train models. Table 2 shows the comparison of the number of parameters with existing models. The Openpose model uses the most memory with 65.7 M parameters, followed by the Slowfast R101-FPN model with 42.8 M. The proposed model uses the least memory using 12.1 M parameters, which shows a lower overhead than existing action classification models.

Table 2: Comparison of the number of parameters with existing models

Model	Number of parameters
Slowfast R101-FPN	42.8 M
Openpose	65.7 M
Proposed model	12.1 M

Similar to the accuracy evaluation, in the speed evaluation, the human action video data of the AI Hub were used [14]. The proposed model is compared to existing models under the same conditions. The existing models are the Slowfast model [6], the Openpose model [3], and the response speed used. Fig. 11 shows the proposed model's response speed compared to the existing models.

According to the comparative measurement, the average response speed per frame in the Slowfast model [6] is 1.887 s. And the average response speed per frame in the Openpose model [3] is 0.725 s. In the model proposed in this work, the response speed per frame was approximately 0.234 s and 0.491 s higher than the Openpose model [3]. In addition, compared to the Slowfast model [6], a much higher response speed can be confirmed. These results made it possible to design a model with almost real-time classification. Real-time action classification technology can aid the development of various fields that require observing people, such as public safety or CCTV monitoring. Regarding performance comparison based on actual practical images, it is necessary to consider that it is a more general computing situation and the point at which Full HD resolution is used. Therefore, there was a difference from the available test results.

5 Conclusion

A model was proposed to quickly detect human actions in a streaming-based video, such as a CCTV image. The proposed model performs in three steps: it detects pattern changes in an object's actions based on the extracted object and action vector generated by the one-shot object detection and object tracking, then classifies the action. According to the performance evaluation, the proposed model has equal or better accuracy and a higher response speed than the most typical action classification model, the Openpose model. Because the neural network, after its end, can train an action vector only separately, it is highly scalable. Even if there are no direct image training data, it is possible to classify a specific faction accurately only by combining the action vector. Also, Nevertheless, there are limited actions for classification because the image data for training in object

detection needs more diversity. In future research, it will be possible to solve this problem by securing more extensive data. With the work results, it is possible to analyze human action patterns rapidly. In particular, it is possible to analyze images in real-time and detect specific abnormal actions or actions in industrial site environments with continuous streaming data input or in image control situations based on CCTV. Also, it is expected to bring economic and industrial ripple effects.

Funding Statement: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2022R1I1A1A01069526).

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

References

[1] E. Vahdani and Y. Tian, "Deep learning-based action detection in untrimmed videos: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4302–4320, 2022.
[2] Z. Tu, H. Li, D. Zhang, J. Dauwels, B. Li *et al.,* "Action-stage emphasized spatiotemporal VLAD for video action recognition," *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 2799–2812, 2019.
[3] H. Yan, B. Hu, G. Chen and E. Zhengyuan, "Real-time continuous human rehabilitation action recognition using OpenPose and FCN," in *AEMCSE*, Shenzhen, China, pp. 239–242, 2020.
[4] H. J. Park, J. -W. Baek and J. -H. Kim, "Imagery based parametric classification of correct and incorrect motion for push-up counter using OpenPose," in *CASE*, Hong Kong, China, pp. 1389–1394, 2020.
[5] C. H. Chen, A. Tyagi, A. Agrawal, D. Drover and R. MV *et al.*, "Unsupervised 3D pose estimation with geometric self-supervision," in *CVPR*, Long Beach, USA, pp. 5714–5724, 2019.
[6] 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.
[7] S. Shi and C. Jung, "Deep metric learning for human action recognition with SlowFast networks," in *VCIP*, Munich, Germany, pp. 1–5, 2021.
[8] J. C. Stroud, D. A. Ross, C. Sun, J. Deng and R. Sukthankar, "D3D: Distilled 3D networks for video action recognition," in *WACV*, Snowmass, CO, USA, pp. 614–623, 2020.
[9] R. E. da Silva, J. Ondrej and A. Smolic, "Using LSTM for automatic classification of human motion capture data," in *GRAPP*, Prague, Czech Republic, vol. 1, pp. 236–243, 2019.
[10] H. Ullah, S. D. Khan, M. Ullah, M. Uzair and F. A. Cheikh, "Two stream model for crowd video classification," in *EUVIP 2019*, Roma, Italy, pp. 93–98, 2019.
[11] S. C. Lai, H. K. Tan and P. Y. Lau, "3D deformable convolution for action classification in videos," *IWAIT 2021*, vol. 11766, pp. 149–154, 2021.
[12] Z. Shou, X. Lin, Y. Kalantidis, L. Sevilla-Lara, M. Rohrbach *et al.,* "DMC-Net: Generating discriminative motion cues for fast compressed video action recognition," in *CVPR*, Long Beach, USA, pp. 1268–1277, 2019.
[13] S. A. Bhat, A. Mehbodniya, A. E. Alwakeel, J. Webber and K. Al-Begain, "Human motion patterns recognition based on RSS and support vector machines," in *WCNC*, Seoul, Korea, pp. 1–6, 2020.
[14] A. I. hub, 2022. [Online]. Available: https://aihub.or.kr/
[15] F. Zang, T. Y. Wu, J. S. Pan, G. Ding and Z. Li, "Human motion recognition based on SVM in VR art media interaction environment," *Human-Centric Computing and Information Sciences*, vol. 9, no. 1, pp. 1–15, 2019.
[16] H. Sun, W. Tao, R. Wang, C. Ren and Z. Zhao, "Research on image classification method based on Adaboost-DBN," in *Int. Conf. on Wireless and Satellite Systems*, Barcelona, Spain, pp. 220–228, 2019.

[17] R. S. Sandhya, N. G. Apparao and S. V. Usha, "Kinematic joint descriptor and depth motion descriptor with convolutional neural networks for human action recognition," *Materials Today: Proceedings*, vol. 37, pp. 3164–3173, 2021.
[18] H. Yang, C. Yuan, B. Li, Y. Du, J. Xing *et al.,* "Asymmetric 3D convolutional neural networks for action recognition," *Pattern Recognition*, vol. 85, pp. 1–12, 2019.
[19] B. Zhang, J. Ren, Y. Cheng, B. Wang and Z. Wei, "Health data driven on continuous blood pressure prediction based on gradient boosting decision tree algorithm," *IEEE Access*, vol. 7, pp. 32423–32433, 2019.
[20] M. A. Khan, M. Sharif, T. Akram, M. Raza, T. Saba *et al.,* "Hand-crafted and deep convolutional neural network features fusion and selection strategy: An application to intelligent human action recognition," *Applied Soft Computing*, vol. 87, pp. 105986–105999, 2020.
[21] C. Cheng and K. K. Parhi, "Fast 2D convolution algorithms for convolutional neural networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 5, pp. 1678–1691, 2020.
[22] M. Kalfaoglu, S. Kalkan and A. Alatan, "Late temporal modeling in 3D CNN architectures with BERT for action recognition," in *European Conf. on Computer Vision*, Springer, Cham, pp. 731–747, 2020.
[23] Z. Zheng, G. An, D. Wu and Q. Ruan, "Global and local knowledge-aware attention network for action recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 334–347, 2021.
[24] R. Girdhar, J. João Carreira, C. Doersch and A. Zisserman, "Video action transformer network," in *CVPR*, Long Beach, USA, pp. 244–253, 2019.
[25] S. W. Pienaar and R. Malekian, "Human activity recognition using LSTM-RNN deep neural network architecture," in *WAC*, Pretoria, South Africa, pp. 1–5, 2019.
[26] Y. Zou, Y. Shi, D. Shi, Y. Wang, Y. Liang *et al.,* "Adaptation-oriented feature projection for one-shot action recognition," *IEEE Transactions on Multimedia*, vol. 22, no. 12, pp. 3166–3179, 2020.
[27] X. Zhong, C. Gu, M. Ye, W. Huang and C. Lin, "Graph complemented latent representation for few-shot image classification," *IEEE Transactions on Multimedia*, vol. 25, pp. 1979–1990, 2022.
[28] K. Peng, A. Roitberg, K. Yang, J. Zhang and R. Stiefelhagen, "Delving deep into one-shot skeleton-based action recognition with diverse occlusions," *IEEE Transactions on Multimedia*, vol. 25, pp. 1489–1504, 2023.
[29] R. Chandra, S. Goyal and R. Gupta, "Evaluation of deep learning models for multi-step ahead time series prediction," *IEEE Access*, vol. 9, pp. 83105–83123, 2021.
[30] A. Katharopoulos, A. Vyas, N. Pappas and F. Fleuret, "Transformers are RNNs: Fast autoregressive transformers with linear attention," *International Conference on Machine Learning. PMLR*, vol. 119, pp. 5156–5165, 2020.
[31] J. Guo, K. Tian, K. Ye and C. -Z. Xu, "MA-LSTM: A multi-attention based LSTM for complex pattern extraction," in *ICPR*, Milan, Italy, pp. 3605–3611, 2021.
[32] S. I. Alzahrani, I. A. Aljamaan and E. A. AI-Fakih, "Forecasting the spread of the COVID-19 pandemic in Saudi Arabia using ARIMA prediction model under current public health interventions," *Journal of Infection and Public Health*, vol. 13, no. 7, pp. 914–919, 2020.
[33] D. Wei, Y. Tian, L. Wei, H. Zhong, S. Chen *et al.,* "Efficient dual attention SlowFast networks for video action recognition," *Computer Vision and Image Understanding*, vol. 222, pp. 103484–103490, 2022.
[34] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, USA, pp. 4724–4733, 2017.
[35] L. Wang, P. Koniusz and D. Huynh, "Hallucinating IDT descriptors and I3D optical flow features for action recognition with CNNs," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Long Beach, USA, pp. 8698–8708, 2019.
[36] H. Yoo and K. Chung, "Classification of multi-frame human motion using CNN-based skeleton extraction," *Intelligent Automation & Soft Computing*, vol. 34, no. 1, pp. 1–13, 2022.
[37] T. I. Hsieh, Y. C. Lo and H. T. Chen, "One-shot object detection with co-attention and co-excitation," in *33rd Conf. on Neural Information Processing System (NeurIPS 2019)*, Vancouver, Canada, vol. 32, 2019.

[38] S. Zhai, D. Shang, S. Wang and S. Dong, "DF-SSD: An improved SSD object detection algorithm based on DenseNet and feature fusion," *IEEE Access*, vol. 8, pp. 24344–24357, 2020.
[39] Y. Wang, C. Wang, H. Zhang, Y. Dong and S. Wei, "Automatic ship detection based on RetinaNet using multi-resolution Gaofen-3 imagery," *Remote Sensing*, vol. 11, no. 5, pp. 531–544, 2019.
[40] A. Bochkovskiy, C. Y. Wang and H. Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," 2020. [Online]. Available: https://arxiv.org/abs/2004.10934
[41] X. Wu, D. Sahoo and S. C. H. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, 2020.
[42] S. Kapania, D. Saini, S. Goyal, N. Thakur, R. Jain *et al.,* "Multi object tracking with UAVs using deep SORT and YOLOv3 RetinaNet detection framework," in *Proc. of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems*, Ontario, Canada, pp. 11–16, 2020.
[43] T. Meinhardt, A. Kirillov, L. Leal-Taixé and C. Feichtenhofer, "TrackFormer: Multi-object tracking with transformers," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Arima, USA, pp. 8844–8854, 2022.
[44] A. Sherstinsky, "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network," *Physica D: Nonlinear Phenomena*, vol. 404, pp. 132306–132333, 2020.
[45] C. Y. Lin, Y. M. Hsieh, F. T. Cheng, H. C. Huang and M. Adnan, "Time series prediction algorithm for intelligent predictive maintenance," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2807–2814, 2019.