Pose-based Human Behavior Detection for Real-time Security Surveillance

1st Jua Park*
Dongduk Women's University
Dept. Statistics and Information Science
Seoul, Republic of Korea
20201054@dongduk.ac.kr

1st Jeongin Cho*
Pukyong National University
Dept. Electronic Engineering
Busan, Republic of Korea
202112541@pukyong.ac.kr

2nd Soonchan Park
Jeonbuk National University
Dept. Computer Science and AI
Jeonju, Republic of Korea
soonchan.park@jbnu.ac.kr

3rd Jun Seong Lee
ETRI
Content Research Division
Daejeon, Republic of Korea
jslee0708@etri.re.kr

4th Moonwook Ryu
ETRI
Content Research Division
Daejeon, Republic of Korea
moonwook@etri.re.kr

***Abstract*—Security is a critical function for protecting target individuals from potential threats. In particular, safeguarding officials such as politicians requires significant resources to ensure their safety. In this paper, we investigate a real-time video surveillance system for safeguarding and develop an accurate and efficient human behavior detection method. After defining four target behaviors associated with potential threats, we construct a video dataset for these behaviors. Using sequences of estimated human poses, we then implement a lightweight human behavior detection method. Specifically, our approach combines convolutional layers with a Transformer encoder to capture both local and global features of human behavior. Experimental results demonstrate that our network achieves an average accuracy of 96.84% with an inference time of 2.1 milliseconds. We expect that the proposed method will significantly reduce the operational cost of video surveillance while maintaining effective detection of potential security threats.**

***Index Terms*—Human behavior detection, video surveillance, security, human pose**

I. Introduction

Understanding human behavior from video has made remarkable progress in computer vision technologies. Automatic detection of human behavior in video surveillance not only reduces operational costs but also provides faster and more accurate performance compared to traditional human-based monitoring [1]–[3]. Such intelligent surveillance systems can also be applied to security scenarios to continuously monitor the surroundings of a target individual to protect and identify persons who may have potential threats to the protected target.

In this paper, we investigate the requirements of a real-time video surveillance system for security, construct the corresponding dataset, and develop a human behavior detection method to rapidly recognize potential threats in video. We specifically defined four target human behaviors and collected a corresponding video dataset. We extracted human-related information such as bounding boxes and human poses and utilized this information to implement an efficient behavior detection algorithm with a simple yet effective neural network architecture and a data augmentation strategy. Our proposed network combines convolutional layers with a Transformer encoder to capture both local and global features from sequences of human poses. Additionally, we apply a data augmentation technique to enhance robustness against human pose estimation errors. As a result, the proposed system achieves approximately 96.7% mAP with an inference latency of 2 milliseconds, making it suitable for real-time surveillance applications in security scenarios.

* Both authors contributed equally.

II. Environment Settings

We begin by analyzing the requirements of a video surveillance system for security applications. In Section II-A, we investigate the system and performance specifications necessary to achieve this goal. Based on these requirements, we define the target human behaviors and collect data to develop a behavior detection network. Section II-B describes the details of the data collection process.

A. Video Surveillance System for Security

In this study, we develop a human behavior detection method and integrate it into a video surveillance system for security applications. The system is designed for location-independent deployment and enables the monitoring of multiple individuals by analyzing video streams transmitted to a central command center. Considering various practical requirements, the proposed system satisfies the following specifications:

- The video surveillance system is designed for flexible deployment, utilizing camera sensors from mobile devices rather than relying on fixed CCTV installations. It supports mobile setup and wireless video transmission.
- The system assumes a monitoring scenario where a security operator observes multiple screens. Accordingly,

TABLE I
SPECIFICATIONS OF THE DATA COLLECTION ENVIRONMENT

Item	Details
Camera	Galaxy S24
Camera Angles	15° (indoor), 45° (indoor), 85° (outdoor)
Shooting Locations	8 locations including indoor and outdoor
Body Types	Female / Male
Clothing	With / Without padded jacket
Number of People	1–4
Orientation	0° to 360°
Number of Participants	9 in total

it aims to detect behaviors unrelated to the main event or indicative of potential pre-incident activities.

- To minimize occlusion among individuals, cameras are installed at least two meters above the ground level.
- The system includes not only the proposed behavior detection algorithm but also a comprehensive analysis framework. Therefore, person detection, tracking, and pose estimation modules may be executed as preliminary steps. Given the computational load of these components, the behavior detection algorithm is required to operate within five milliseconds per frame to ensure real-time performance.
- The system is designed to be robust against inaccuracies in human pose estimation that may occur due to occlusions between individuals in a crowd.

B. Dataset Construction

1) Target behavior for security scenario: When we consider existing datasets such as *NTU RGB + D* [4] and *RWF-2000* [5], they are not suitable for detecting potential threats in our defined security applications. Therefore, we constructed a new dataset to train and evaluate the human behavior detection algorithm. Based on our preliminary studies investigating the requirements in real-world security settings, we defined four target human behaviors to identify potential security threats in video surveillance. The four defined behaviors are as follows: *Throw*, *Protest*, *Run*, and *Look around*. *Throw* behavior is a direct threat to security, while the others are considered precursor behaviors that may indicate a potential threat. To ensure the stability of human behavior detection, all other unrelated behaviors are categorized as *Idle* behaviors. Therefore, we defined five behavior categories in this study.

2) Data collection environment: We defined our data collection environment as summarized in Table I. To ensure the diversity of the dataset, seven key attributes were considered. The dataset includes variations in the following aspects: three camera angles, eight different locations encompassing both indoor and outdoor environments, gender (male/female), presence or absence of heavy winter clothing (to represent season-independent detection), number of people (ranging from 1 to 4), and orientation of individuals. Heavy winter clothing was considered because it can influence the structure of the estimated human pose. In addition, as shown in Table II, various sub-scenarios were defined within each behavior class to enable more accurate learning of diverse patterns of target behaviors that may occur in real-world surveillance environments.

TABLE II
SUB-SCENARIOS DEFINED FOR EACH BEHAVIOR CLASS

Behavior Class	Sub-scenarios
Throw	• Throwing while standing • Throwing while sitting • Throwing while taking out an object
Protest	• Protesting with both arms raised aggressively • Protesting with one arm holding a picket
Run	• Sprinting at full speed
Look around	• Looking around while standing • Looking around while sitting • Looking around while talking on the phone • Looking around while writing something • Looking around while taking out an object

TABLE III
QUANTITY OF VIDEO SAMPLES FOR EACH BEHAVIOR CLASS IN THE COLLECTED DATASET

Class Label	Behavior Name	Count
0	Idle	1023
1	Throw	401
2	Protest	117
3	Run	296
4	Look around	457

3) Raw data capturing: To maximize efficiency, we developed detailed scripts for each class based on various environmental factors such as gender, number of people, and camera angles. Each behavior was performed repeatedly under diverse settings and conditions to ensure the model's ability to generalize across a wide range of real-world situations.

The raw data was categorized into five classes, from Class 0 to Class 4, corresponding to *Idle*, *Throw*, *Protest*, *Run*, and *Look around*, respectively, as shown in Table III. A total of 2,294 video samples were collected, comprising 1,023 samples for *Idle*, 401 for *Throw*, 117 for *Protest*, 296 for *Run*, and 457 for *Look around*. The relatively small number of samples in the *Protest* class was due to preliminary experiments showing that this behavior has distinctive characteristics that make it clearly distinguishable from other classes. Specifically, in contrast to the rapid motion of the *Throw*, *Protest* behavior is more static and typically performed in place, which contributes to more stable human detection and reduced motion blur in video frames. The *Idle* class includes all behaviors that cannot be categorized into the four predefined actions. For example, standing, walking, and hand-waving are considered part of the *Idle* class. Fig. 1 illustrates a sample of our dataset. The test set was constructed by additionally recording 38 videos featuring individuals not included in the training dataset. Furthermore, to evaluate the generalization performance of the model in more diverse environments, an additional 60 videos were collected from YouTube using keywords such as *protest*, *crowd*, *baseball*, and *running*, which are related to the

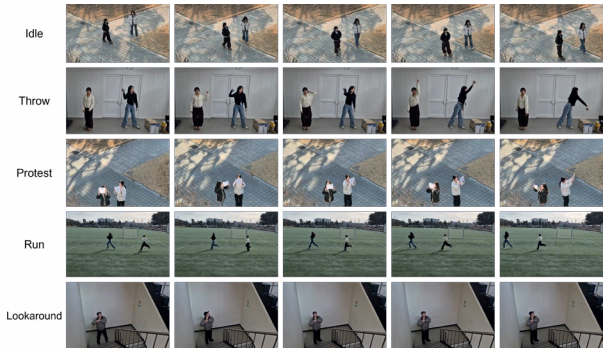


Fig. 1. Samples of collected dataset

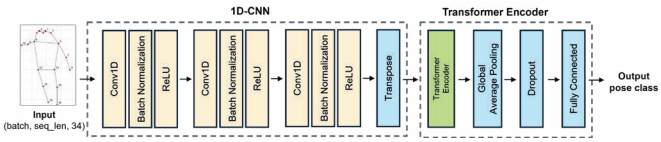


Fig. 2. Structure of the 1D-CNN + Transformer encoder model

experimental scenarios.

4) Dataset labeling: The recorded videos were manually segmented to isolate the portion where the defined behaviors occurred. Temporal segmentation was performed for the four target classes, while segments that did not correspond to any of these behaviors were labeled as *Idle*. This process ensured a clear distinction between *Idle* and the other target behaviors. To enhance the generalization capability of the dataset, recordings were repeated under various conditions as illustrated in Table II. Based on the raw videos, short clips ranging from 2 to 7 seconds were extracted, centered around the onset of each behavior, resulting in a total of 2,294 samples.

To track the pose of each individual and detect their behaviors, we utilized off-the-shelf algorithms for bounding box and pose estimation. Specifically, we applied a bounding box detector and tracker with [6] (i.e., ByteTrack) and a top-down human pose estimator with [7] (i.e., PCT) to the curated video clips in our dataset. Using estimated bounding boxes, identity indices, and human poses, our system is able to continuously monitor each person in the scene and analyze their behavior over time.

III. Human Behavior Detection Network

Based on the dataset described in Section II-B, we developed a neural network that recognizes human behaviors using sequences of human poses (i.e., a set of 2D coordinates of human keypoints). Considering the requirements for real-time operation and accurate behavior detection, we designed a lightweight neural network architecture that leverages both local and global features from pose sequences. We also applied a data augmentation technique to ensure robust performance under the instability of preceding human pose estimations.

A. Network Architecture

The human pose sequence is one of the primary sources for understanding human behaviors. For fast estimation, short-term changes in human pose can provide important cues. However, relying solely on short-term features may lead to unstable predictions, because they reflect only partial pose changes rather than a holistic analysis. Therefore, in this study, we design a human behavior detection network that considers not only short-term changes but also long-term changes in human pose.

The structure of the behavior detection model based on 1D-CNN and Transformer encoder, proposed in this study, is illustrated in Fig. 2. Specifically, with 2D coordinates of 17 keypoints in n frames, $H = \{h_0, h_1, ..., h_n\} \in \mathbb{R}^{n\times 34}$, 1D-convolutional Neural Network (CNN) layer $\mathbf{E_{local}}$ extracts local features from the H by sliding convolution kernels along the temporal axis. Unlike traditional time-series models such as Recurrent Neural Networks (RNN) or Long Short-Term Memory networks (LSTM), the 1D-CNN enables parallel processing of the entire input sequence, leading to faster training — an important advantage for real-time video surveillance scenarios. Then, the estimated local features $Z_0 \in \mathbb{R}^{seq\_length\times 128}$ are input to the following The Transformer encoder $\mathbf{E_{global}}$ to consider global information to understand patterns of given behavior [8]. In $\mathbf{E_{global}}$, self-attention mechanism is utilized by using the input feature itself as its query, key, and value and refining the input latent Z_0 to $Z \in \mathbb{R}^{seq\_length\times 128}$. This makes it effective in modeling dependencies across the entire sequence, which is particularly useful for recognizing behaviors that involve changes in posture over multiple frames. The self-attention mechanism in this block can be illustrated as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{1}$$

where $Q, K, V = Z_0$ and d_k is the dimension of key and query.

Finally, as a classification head, a fully-connected layer $\mathbf{C_{cls}}$ uses the refined latent feature to estimate probabilities of each behavior category. In conclusion, the entire process of our proposed network can be illustrated as:

$$Z_0 = \mathbf{E_{local}}(H) \tag{2}$$

$$Z = \mathbf{E_{global}}(Z_0) \tag{3}$$

$$\hat{R} = \mathbf{C_{cls}}(\mathbf{P}(Z)) \tag{4}$$

where $\mathbf{P}$ is global pooling and $\hat{R} \in \mathbb{R}^5$ presents probabilities of the four target behavior and *Idle*.

B. Data Augmentation for Robust Detection

According to the environment configured in this study, the estimation of human pose becomes unstable due to partial occlusions caused by other subjects or limited field of view of the camera, as well as rapid movements [10], [11]. Some human keypoints may be missing, which causes a

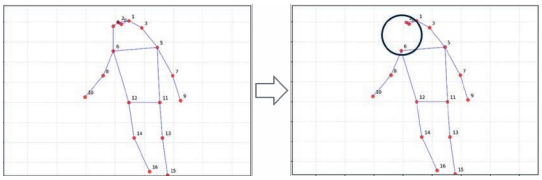


Fig. 3. Examples of keypoint drop for data augmentation

drop in performance in subsequent human behavior detection algorithms. To build a robust situational awareness network capable of operating reliably under degraded pose estimation conditions, we developed a data augmentation technique that simulates such challenging environments. For each frame, we dropped the keypoints with probability P_{drop}. When we set $P_{drop} = 0.3$, one to five keypoints were randomly selected and their coordinates were set to (0, 0), indicating missing keypoints. To preserve the structural information of human pose, no more than five keypoints were dropped in each case. Fig. 3 illustrates an example of data augmentation, in which the right ear in the facial region has been removed. This may occur depending on the orientation of the human head.

C. Training

Using the class label of the target behaviors and the one-hot encoding result of our behavior detection network, we employ a cross-entropy loss to train the entire network. This can be formally expressed as :

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{C} y_i \log(\hat{R}_i) \tag{5}$$

where C is the number of classes, y_i is the one-hot encoding of ground-truth's class label, and $\hat{R}$ is the detection result.

IV. Experiments

In this section, we provide details of the experiments to verify the effectiveness of our proposed methods. Section IV-A describes the experimental setting for evaluation. Section IV-C and Section IV-D present the experimental results in a quantitative and a qualitative manner, respectively.

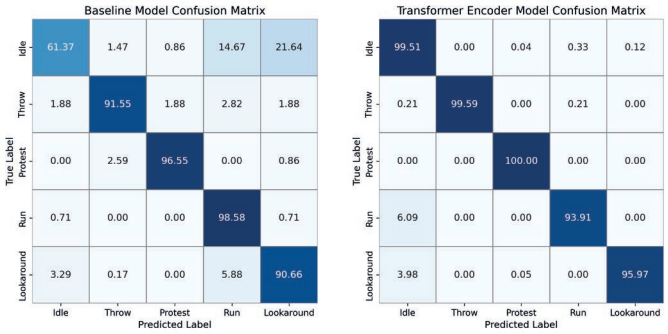
A. Experimental Setting

As described in Section II-B, the dataset collected in this study consists of five behavior classes: *Idle*, *Throw*, *Protest*, *Run*, and *Look around*, comprising 1,023, 401, 117, 296, and 457 samples, respectively, for a total of 2,294 sequences. To more accurately evaluate the generalization performance of the model, we applied a 5-fold cross-validation. In each iteration, one fold was used for validation while the remaining four were used for training. As a result, the model was trained and evaluated five times with different training and validation splits, and the final performance was calculated as the average across all folds. For each fold, approximately 1,836 samples were used for training and 458 samples for validation.

For performance comparison, we defined a baseline model with a simple architecture consisting of a 1D convolutional layer followed by a gated recurrent unit (GRU), and compared its detection performance with that of our proposed architecture. As the evaluation metric, we measured the accuracy of each network by using the predicted labels with the highest probabilities at the final layer.

TABLE IV
Evaluation results of baseline and our proposed method

Metric (unit)	Baseline	Ours
Averaged Precision (%)	72.46	96.84
Inference time (ms)	0.838	2.114



(a) Baseline (b) Ours

Fig. 4. Confusion matrices of evaluation result

B. Implementation Details

The entire framework was developed using the PyTorch. As we discussed in Section II-B4, a 34-dimensional vector estimated by [7] was used as input and sequence length of behavior is 16. The 1D-CNN consists of three convolutional layers with a kernel size of 3 and padding of 1. Each layer is followed by batch normalization and a ReLU activation function. The output channels of the convolutional layers are set to 64, 128, and 128, respectively. For the Transformer Encoder, we used 8 heads and 128 latent channels. During training, we adopted the Adam optimizer [12] with a fixed learning rate of 3e-4. The model was trained for 100 epochs on one NVIDIA RTX 3090 with a batch size of 8. To prevent overfitting, we applied a dropout rate of 0.3 to both the Transformer Encoder and the final features before the last classification layer. The entire training time was approximately 2.5 hours for each validation.

C. Quantitative Evaluation

As Table IV illustrates, the mean accuracies of the baseline and our proposed network were 72.46% and 96.84%, respectively, in our experiments. Our proposed method outperformed the baseline by using local and global information from pose sequences. The inference time of the detection network increased, but it still met the requirements discussed in Sec. II-A. For in-depth analysis, we visualized a confusion matrix as illustrated in Fig. 4. The baseline model had a particularly low performance on the *Idle* class, which recorded only 61.37%. This was likely due to the high similarity between *Idle* and

(a) Running phase

(b) Walking phase

Fig. 5. Behavior transition from running to walking

(a) Protesting phase

(b) Throwing phase

Fig. 6. Behavior transition from protest to throw

other behaviors, making them difficult to distinguish, as well as the limitation of the GRU-based architecture that focuses solely on local information. The *Idle* class was frequently misclassified as *Run* (14.67%) and *Look around* (21.64%). In contrast, the accuracy of the proposed Transformer encoder model on the *Idle* class increased to 99.51%, and the *Protest* and *Throw* classes also achieved high accuracies of 100% and 99.59%, respectively. These results indicate that using the self-attention mechanism of the Transformer encoder with local features estimated by 1D convolution effectively captures local and global contextual information simultaneously, enabling the model to distinguish fine-grained differences, thereby achieving superior performance in behavior detection tasks.

D. Qualitative Evaluation

To evaluate the model's ability to detect behavior transitions, we tested it on video sequences containing continuous shifts between multiple target behaviors, such as *running to walking* and *protesting followed by throwing*. These composite sequences were not explicitly included during training, yet our proposed model accurately segmented and classified each behavior within the sequence.

As shown in Fig. 5 and Fig. 6, the model successfully identified the transition to *Walking* immediately following a brief *Running* phase, and similarly detected a shift from *Protest* to *Throw* within a single 7-second clip. In particular, in the *running to walking* video, the subject is partially occluded by a tree during the transition. Despite this occlusion and the reduced apparent size of the subject in the frame, the model maintained accurate detection performance, demonstrating robustness to partial visual obstruction and distant-object scenarios.

In addition, the model showed resilience to keypoint noise caused by scale reduction and motion blur, which often occur when the subject moves rapidly or appears smaller in the frame. These results suggest that the model has learned to recognize transitions between temporally adjacent behaviors by focusing on consistent pose dynamics, rather than relying on fixed-duration patterns or isolated key-frame cues. This ability to detect behavior transitions, even under occlusion and pose instability, is crucial for real-world surveillance scenarios, where human behaviors often unfold as continuous, context-dependent sequences in visually challenging environments.

V. Conclusion

In this paper, we address the problem of human behavior detection in video surveillance scenarios for security. We analyzed the requirements of real-world security contexts and constructed a dedicated dataset to support effective threat monitoring. The proposed detection method demonstrates high accuracy and stability by leveraging both local and global information from human pose sequences. With an inference time of around 2 milliseconds, the proposed network can reliably detect target behaviors that may serve as cues for potential threats, making it well-suited for real-time surveillance applications.

Acknowledgment

This work was supported by National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No.RS-2024-00462874, Development of Sentinel AI system technology for predicting and preemptively responding to threats in open crowded environments).

References

[1] T. Liu and H. Wang, "Low-resolution activity recognition using super-resolution and model ensemble networks," ETRI Journal, vol. 46, no. 1, Article no. e12708, 2024.

[2] A. A. U. Rakhmonov, M. Kim, J. Choi, and Y. M. Ro, "AONet: Attention network with optional activation for unsupervised video anomaly detection," ETRI Journal, vol. 46, no. 5, pp. 890–903, 2024.

[3] M. Cormier, A. Clepe, A. Specker, and J. Beyerer, "Where are we with human pose estimation in real-world surveillance?," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2022, pp. 591–601.

[4] A. Shahroudy, J. Liu, T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1010–1019.

[5] M. Cheng, K. Cai, and M. Li, "RWF-2000: An open large-scale video database for violence detection," in Proceedings of the 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 4185–4192.

[6] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, C. Xie, W. Zhang, and L. Wang, "ByteTrack: Multi-object tracking by associating every detection box," in Proceedings of the European Conference on Computer Vision (ECCV), 2022, pp. 144–160.

[7] Z. Geng, Y. Zhang, C. Xie, and W. Zhang, "Human pose as compositional tokens," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 12345–12354.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[9] OpenMMLab, "MMPose: OpenMMLab pose estimation toolbox and benchmark," GitHub, [Online]. Available: https://github.com/open-mmlab/mmpose, Accessed: May 19, 2025.

[10] S. Park, S. Lee, and J. Park, "Data augmentation method for improving the accuracy of human pose estimation with cropped images," Pattern Recognition Letters, vol. 136, pp. 244–250, 2020.

[11] S. Park and J. Park, "Position puzzle network and augmentation: Localizing human keypoints beyond the bounding box," Machine Vision and Applications, vol. 34, no. 6, p. 129, 2023.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in Proceedings of the International Conference on Learning Representations (ICLR), 2015.