IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 27 June 2024, accepted 1 August 2024, date of publication 5 August 2024, date of current version 13 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3438546

RESEARCH ARTICLE

Interpretable Information Visualization for Enhanced Temporal Action Detection in Videos

DASOM AHN[1], JONG-HA LEE[2], (Member, IEEE), AND BYOUNG CHUL KO[1], (Senior Member, IEEE)

[1]Department of Computer Engineering, Keimyung University, Daegu 1095, South Korea
[2]Department of Biomedical Engineering, Keimyung University, Daegu 1095, South Korea

Corresponding author: Byoung Chul Ko (niceko@kmu.ac.kr)

This work was supported by Korea Health Industry Development Institute (Contact-Free Intelligent Monitoring Devices for Infectious Diseases) under Grant HI23C0942.

ABSTRACT Temporal action detection (TAD) is one of the most active research areas in computer vision. TAD is the task of detecting actions in untrimmed videos and predicting the start and end times of the actions. TAD is a challenging task and requires a variety of temporal cues. In this paper, we present a one-stage transformer-based temporal action detection model using enhanced long- and short-term attention. Recognizing multiple actions in a video sequence requires an understanding of various temporal continuities. These temporal continuities encompass both long- and short-term temporal dependencies. To learn these long- and short-term temporal dependencies, our model leverages long- and short-term temporal attention based on transformers. In short-term temporal attention, we consider long-term memory to learn short-term temporal features and use compact long-term memory to efficiently learn long-term memory. Long-term temporal attention uses deformable attention to dynamically select the required features from long-term memory and efficiently learn the long-term features. Furthermore, our model offers interpretability for TAD by providing visualizations of class-specific probability changes for temporal action variations. This allows for a deeper understanding of the model's decision-making process and facilitates further analysis of TAD. Based on the results of experiments conducted on the THUMOS14 and ActivityNet-1.3 datasets, our proposed model achieves an improved performance compared to previous state-of-the-art models. Our code is available at https://github.com/tommy-ahn/LSTA.

INDEX TERMS Temporal action detection, transformer, cross attention, video understanding.

I. INTRODUCTION

Temporal action detection (TAD) aims to predict human actions and localize the start and end frames of video sequences. TAD is a challenging but essential algorithm in the field of video understanding and is widely used in areas such as autonomous driving and video analysis. Understanding the temporal characteristics of a video is particularly important for artificial intelligence (AI) models when predicting human actions and their boundaries. To achieve such understanding, various approaches such as anchor and proposal-based [1], [2], [3], recurrent [4], and convolution [5], [6] models have been studied regarding the learning and prediction of temporal features. Transformer-based studies [7], [8] incorporating multi-head self-attention (MHSA) [9] in the learning of long-range dependencies have recently been conducted.

The recognition of multiple actions in a video sequence requires a deep understanding of the various temporal continuities that can occur. These temporal continuities are determined or predicted based on long-term temporal features maintained for lengthy periods and short-term features maintained for relatively shorter periods. Long-term temporal features are important when events in the distant past affect current events. For example, the previous action of *"cooking"* is an important clue for predicting the action of *"eating food"*.

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

Short-term temporal features are also important for TAD. For instance, the specific movements that occur in each frame or clip are crucial for action recognition in video sequences. Therefore, it is necessary to develop a new method to enhance the TAD features by focusing on both long- and short-term temporal features.

Transformer [9] has achieved state-of-the-art (SoTA) results in natural language processing (NLP) tasks. It uses the attention mechanism to learn long-range dependencies and relationships between words. Although TAD is a challenging task as it must consider various temporal cues such as object motion, object shape, and video context, the transformer is well-suited for TAD because it can learn long-range temporal relationships between frames. Therefore, we propose a novel long and short-term attention mechanism for transformer-based TAD models. Given the critical importance of the relationships between long- and short-term features in TAD, we draw inspiration from Long Short-Term Transformer (LSTA) [20] and design our algorithm to learn both long and short-term temporal features effectively.

Existing TAD models have traditionally focused primarily on multi-scale features for action detection, which may impose performance limitations. However, TAD requires intricate and detailed learning of temporal features. Therefore, by effectively integrating long- and short-term temporal features, we can capture finer and comprehensive temporal characteristics. This approach offers opportunities for achieving superior performance.

Our proposed mechanism is called long- and short-term temporal attention (LSTA), for exploiting the importance of interactions between temporal long- and short-term features in sequence analysis tasks. These mechanisms enhance the temporal features by incorporating both long- and short-term representations at each time step and conducting interactions, thereby enabling the TAD model to achieve better performance. Our model constructs video clip features from untrimmed videos by utilizing a 3D CNN backbone [10], [11], [12], following the approach of traditional models. Afterward, the video clip features are downsampled to create a pyramid structure, facilitating the computation of multi-temporal scale features. To enhance the learning of long- and short-term temporal features even further, we integrate the short-term temporal attention (STA) block and long-term temporal attention (LTA) block into this process. STA block first divides the features extracted by the backbone network into arbitrarily sized blocks to create short-term features. In the STA block, we leverage short-term features and additional compact long-term memory (CLM) to effectively learn the relationships between long-term and short-term features. Through this process, the STA strengthens the relationships between long- and short-term temporal features and enhances the short-term features. LTA block efficiently learns and enhances deformable features that are important for long-term relationships. This enables the LTA to relearn the important features of the outputs. In addition, LSTA's visualizations of class-specific probability changes provide insights into the model's decision-making process, facilitating further analysis of temporal action detection.

Our main contributions are summarized as follows:

- To learn and enhance each feature, we introduced a novel TAD model based on LSTA that exploits the interactions between long- and short-term temporal features, which are important in sequence analysis tasks.
- LSTA includes STA and LTA, where STA learns and enhances short-term temporal features by taking into account long-term memory, and LTA dynamically learns and emphasizes the necessary parts from the entire long-term temporal feature.
- LSTA capability to visualize class-specific probabilities for temporal action variations offers an interpretative perspective on how the model identifies specific actions across various video sequences.
- We demonstrate that our proposed LSTA model performs better than the existing SoTA models of THUMOS14 and ActivityNet-1.3, which are known to be challenging for TAD.

II. RELATED WORK

A. TEMPORAL ACTION DETECTION

TAD resembles object detection models and is broadly categorized into one- [4], [7], [8], [13], [14] and two-stage methods [1], [2], [3], [15], [16], [17]. In two-stage methods, such as anchor windows or proposals approach, motion boundary candidates are first generated, and motion prediction is then conducted. In terms of speed and model size, two-stage methods are inefficient owing to the requirement of additional submodels. Therefore, recent studies have used one-stage methods that predict both actions and their boundaries directly, in contrast to a two-stage approach. One-stage methods predict the actions for all frames separately when estimating the action boundaries.

B. TRANSFORMER-BASED TEMPORAL ACTION DETECTION

Transformer [9] has achieved significant success in natural language processing (NLP) tasks. It uses the attention mechanism to learn long-range dependencies and relationships between words. Transformer has been extended to various attention techniques [18], [19], [20], such as MHSA and cross-attention, which have been shown to improve the performance of many AI networks. The transformer has been successfully applied to vision tasks, such as image classification [19], [21], [22], [23], [24], [25], object detection [25], [26], [27], [49], [50], action recognition [18], [28], [29], [30], [31], and explainable AI [32], [33], [34], [35] through vision transformer (ViT) [22].

Transformer is highly suitable for temporal learning because they can learn long-range dependencies between frames and attention mechanism allows the transformer to consider different temporal cues. This makes transformer well-suited for tasks such as TAD, where it is important to

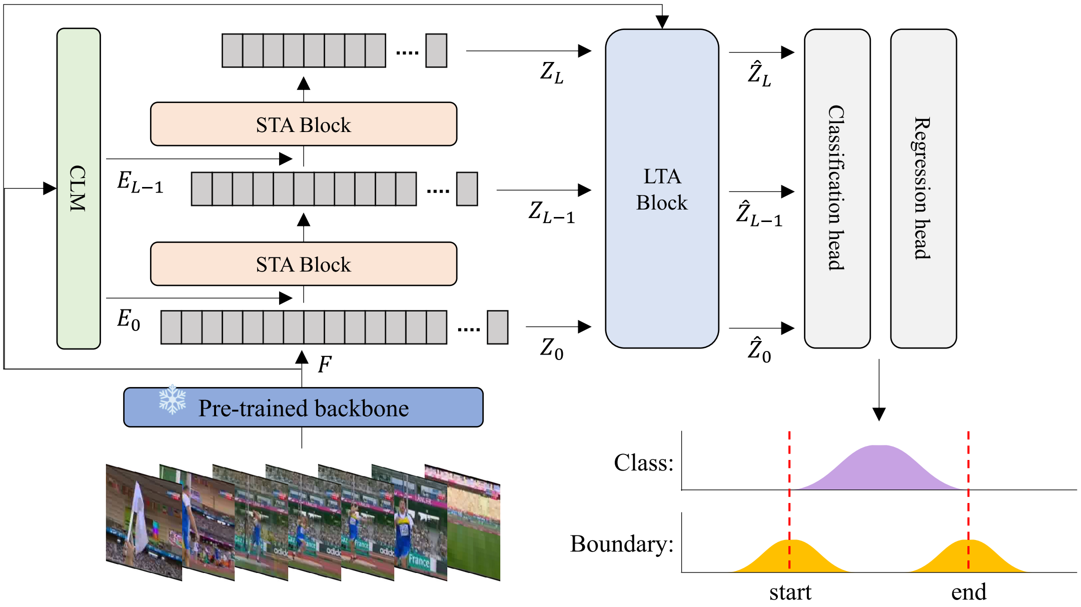


FIGURE 1. The overall architecture of LSTA. Our model learns long- and short-term temporal features to predict actions and estimate action boundaries. First, a set of features is extracted from the video clip. The included features consider various temporal scales using a feature pyramid of L levels which is created using STA blocks to enhance short-term features. The feature pyramid is then enhanced with LTA for long-term features, and the final output is generated. Finally, a shared classification and regression head is used to generate task candidates at every step.

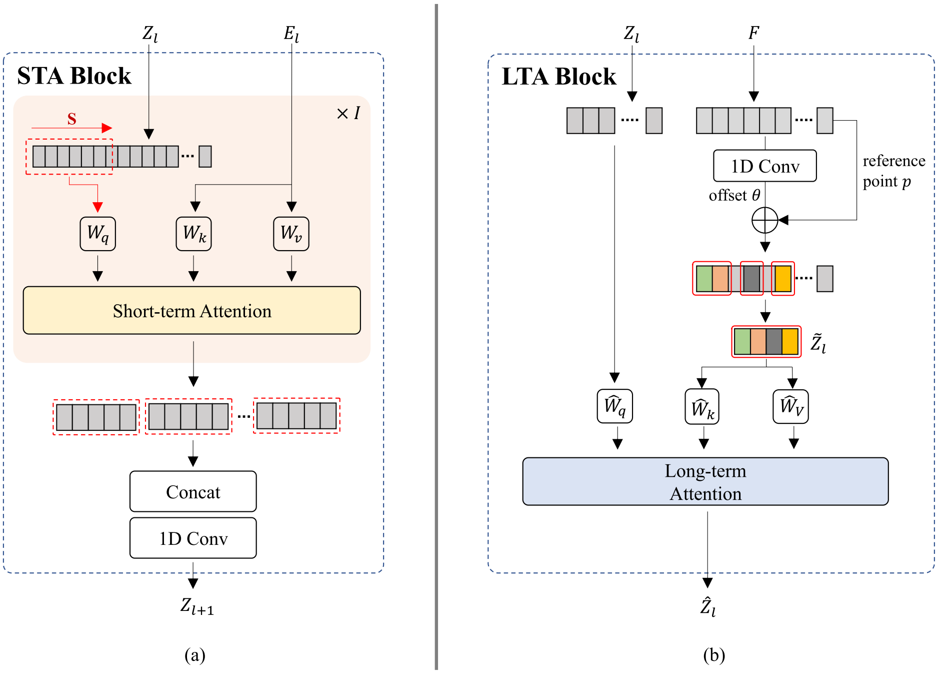


FIGURE 2. Proposed LSTA block mechanisms: (a) STA and (b) LTA blocks.

consider the temporal order of events. A few transformer-based TAD models [7], [8] have been proposed to exploit such benefits. ActionFormer [7] used a feature pyramid structure and a lightweight convolutional decoder to achieve improved performance. It shows the effectiveness in a variety of TAD tasks, including action detection and localization. TriDet [8] inspired from ActionFormer added relative boundary modeling techniques to achieve superior performance. TriDet has a good performance in action detection tasks, especially for actions with complex boundaries. The existing approaches still exhibit deficiencies in the detailed comparison and integrated learning of temporal features. Given the criticality of temporal characteristics in TAD, the effective integration of both long- and short-term temporal features is paramount. Hence, we propose a novel paradigm aimed at comparing and flexibly integrating these temporal features.

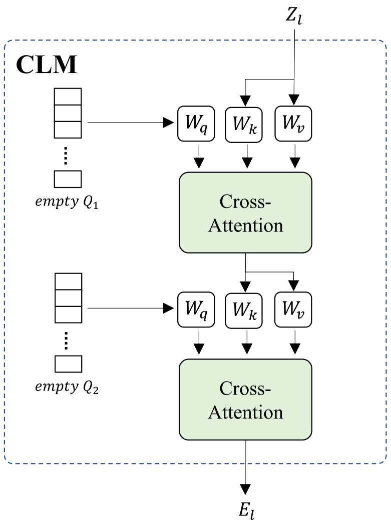


FIGURE 3. **Proposed CLM mechanism.**

III. METHOD

Our proposed LSTA model (Fig. 1) uses a pre-trained backbone network to extract features $F \in \mathbb{R}^{T\times C}$ for input videos, where T represents the temporal dimension of the extracted features, and C represents the channel dimension. Through feature embedding, F is tokenized into $Z \in \mathbb{R}^{T\times D}$ and is used to create a feature pyramid structure. LSTA uses a feature pyramid structure to incorporate multi-scale temporal features and employs a novel STA mechanism to learn and enhance short-term temporal features. Using long-term memory E, which can be constructed using compact long-term memory (CLM), the STA learns and enhances short-term temporal features. These features are then processed by the LTA to dynamically learn and enhance the long-term temporal features in producing the output $\hat{Z}$. The prediction head consists of classification and regression heads. The classification head predicts the action label, and the regression head predicts the action boundaries using $\hat{Z}$. The final result is represented as $Y = (y_t, y_t^s, y_t^e)$, where y_t is the predicted probability of an action at time t, and y_t^s and y_t^e represent the predicted distances to the start and end of the action from time t, respectively.

A. SHORT-TERM TEMPORAL ATTENTION BLOCK

To learn and enhance short-term temporal features, STA is trained as shown in Fig. Fig. 2(a). Although a naive approach, such as the use of an arbitrarily sized time window, can be used for training, we apply a long-term memory query to learn the interactions among the short- and long-term temporal information.

First, input Z_l is divided into short-term temporal memory features $Z_{short} = \{Z_i\}_{i=1}^{I}$ using a predefined short-term window of size S, where $I = T/S$. Z_{short} is used as query Q in the STA. To learn the relationship between short- and long-term memories, we use feature embedding F to create long-term memory E (Fig. 2 (a)). Using F directly results in a time complexity of T^2, leading to a significant increase in the computation time. Therefore, inspired by a long short-term transformer (LSTR), we construct a compressed long-term memory E using learnable empty queries Q_1 and Q_2 ($Q_2 < Q_1 < T$), which represent the entire temporal features.

$$E = \text{Decoder}(Q_2, \text{Decoder}(Q_1, F)) \tag{1}$$

The Decoder in Equation (1) refers to the decoder of the transformer [9] model following the LSTR. Cross-attention, where E is used as the key K value of the V pair and Z_{short} is used for query Q, is applied to learn the relationship between short- and long-term memory while the short-term memory features are enhanced. This operation is formalized as follows.

$$Q = Z_{short} \cdot W_q, K = E \cdot W_k, V = E \cdot W_v \tag{2}$$

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{Q \cdot K^{\text{T}}}{\sqrt{d_h}}) \cdot V \tag{3}$$

where K, and V are computed from E using linear projection layers with trainable parameters W_k and W_v. In addition, Z_{short} is used as Q, which is applied through a linear projection layer with the trainable parameter W_q. Moreover, Q, K, and V undergo cross-attention operations, resulting in the production of short-term temporal features that encode the long-term temporal dependencies. After repeating this process I times, we concatenate all resulting features to obtain the final feature Z_{l+1}.

B. LONG-TERM TEMPORAL ATTENTION BLOCK

The LTA block (Fig. 2 (b)) leverages the strengthened short-term temporal features from the STA block to enhance the long-term temporal features for F. As mentioned previously, in STA, applying attention over the entire time T for feature F is computationally expensive. Therefore, we propose an LTA with deformable attention that allows the essential tokens that affect feature F to be flexibly selected. We first generate a reference point p for the entire feature F, and the offset θ, learned from the convolution layers, is then used to compute K and V by generating the necessary token coordinate values. We then apply to cross attention with K, V, and Q generated from Z, enhancing the long-term temporal features. LTA can be defined through the following equations:

$$\tilde{Z} = \phi(F; p + \Delta p), \Delta p = \theta(F), \tag{4}$$

$$Q = Z \cdot \hat{W}_q, K = \tilde{Z} \cdot \hat{W}_k, V = \tilde{Z} \cdot \hat{W}_v, \tag{5}$$

$$\text{LTA}(Z, F) = \text{Attention}(Q, K, V), \tag{6}$$

$$\hat{Z} = \text{LTA}(Z, F) \tag{7}$$

Here, ϕ is a function that uses reference points p and offsets θ to learn and generate the token coordinate values. $\tilde{Z}$ is a token chosen to be deformable using ϕ and p. Here, $\hat{W}_q$, $\hat{W}_k$, and $\hat{W}_v$ are trainable parameters used to generate Q, K, and V. In this manner, through LTA, we can fuse the enhanced short-term temporal feature values Z_l outputted from STA with the entire input feature F to generate the final output

TABLE 1. **mAP values according to the variation in IoU threshold in comparison with the SoTA methods on the THUMOS14 dataset. (Avg = average mAP). The best results and our results are in bold.**

Method	IoU threshold					Avg.
	0.3	0.4	0.5	0.6	0.7	
TCANet [41]	60.6	53.2	44.6	36.8	26.7	44.3
RTD-Net [16]	68.3	62.3	51.9	38.8	23.7	49.0
VSGN [42]	66.7	60.4	52.4	41.0	30.4	50.2
ContextLoc [43]	68.3	63.8	54.3	41.8	26.2	50.9
AFSD [14]	67.3	62.4	55.5	43.7	31.1	52.0
ReAct [44]	69.2	65.0	57.1	47.8	35.6	55.0
TadTR [45]	74.8	69.1	60.1	46.6	32.8	56.7
ActionFormer [7]	82.1	77.8	71.0	59.4	43.9	66.8
DiffTAD [46]	74.9	72.8	71.2	62.9	**58.5**	68.0
TriDet [8]	**83.6**	**80.1**	72.9	62.4	47.4	**69.3**
LSTA	**81.4**	**78.5**	**73.1**	**63.3**	**48.3**	**68.9**

$\hat{Z}$. Two heads are used to predict the action class label and the action boundary from the output of LTA, *i.e.*, $\hat{Z}$.

C. HEADS AND LOSSES

The proposed LSTA model consists of two heads as shown in Fig. 1: a classification head that predicts the action labels using feature $\hat{Z}$ learned through STA and LTA, and a regression head that predicts the action boundaries.

The classification head consists of three convolutions, two-layer norms, and ReLU activation functions. It predicts C action classes for each frame. We train this head for binary classification with a focal loss [36] $\mathcal{L}_{class}$, which is robust against a data imbalance. The regression head has a structure similar to that of the classification head and predicts the distances to the action boundaries from each frame. The regression head is regularized with $\mathcal{L}_{reg}$ using the DIoU loss [37]. The formula for the total training loss $\mathcal{L}_{total}$ is defined as follows:

$$\mathcal{L}_{total} = \frac{(\mathcal{L}_{class} + \mathbb{C}_t \mathcal{L}_{reg})}{T_{pos}} \tag{8}$$

where T_{pos} denotes the number of positive samples, and $\mathbb{C}_t$ is an indicator function that distinguishes between the foreground and background at time step t.

IV. EXPERIMENTS

In this section, we introduce the implementation details such as the datasets used in the experiments and the hyperparameters of the proposed model. We conducted comparative experiments using existing SoTA models to demonstrate the competitive performance of the proposed model. Furthermore, we conducted an ablation study to analyze each module of the proposed model.

A. EXPERIMENTAL SETUP

1) DATASETS

In this study, experiments were conducted using the THUMOS14 [38] and ActivityNet-1.3 [39] datasets. The THUMOS14 dataset consists of 20 action classes and 413 video sequences. The dataset is divided into 200 video sequences for the validation set and 213 video sequences for the test set. Our model was trained using a validation set and evaluated using a test set. The ActivityNet-1.3 dataset is a large-scale dataset containing 20,000 video sequences and 200 action classes. The dataset was divided into 10,024 videos for training, 4,926 videos for validation, and 5,044 videos for testing.

TABLE 2. **mAP values according to the variations in IoU thresholds in comparison with the SoTA methods on the ActivityNet-1.3 dataset. (Avg = average mAP). The best results and our results are in bold.**

Method	Backbone	IoU threshold			Avg.
		0.50	0.75	0.95	
ReAct [44]	TSN	49.6	33.0	8.6	32.6
G-TAD [17]	TSN	50.4	34.6	9.0	34.1
AFSD [14]	I3D	52.4	35.2	6.5	34.3
TadTR [45]	TSN	51.3	35.0	9.5	34.6
TadTR [45]	R(2+1)D	53.6	37.5	**10.5**	36.8
VSGN [42]	I3D	52.3	35.2	8.3	34.7
TCANet+BMN [41]	TSN	52.3	36.7	6.9	35.5
TCANet+BMN [41]	SlowFast	54.3	**39.1**	8.4	**37.6**
DiffTAD [46]	I3D	56.1	36.9	9.0	36.1
ActionFormer [7]	R(2+1)D	54.7	37.8	8.4	36.6
TriDet [8]	R(2+1)D	54.7	38.0	8.4	36.8
LSTA	R(2+1)D	**55.2**	**38.1**	**9.3**	**37.1**

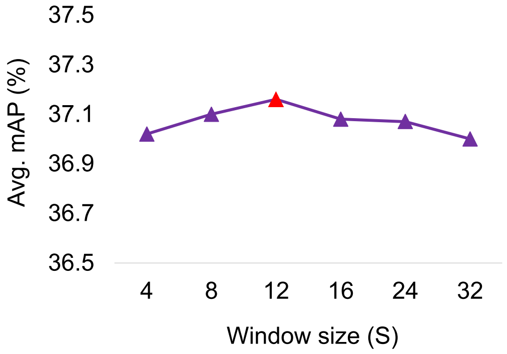


FIGURE 4. **Performance evaluation according to short-term window sizes.**

2) IMPLEMENTATION DETAILS

In this experiment, the proposed model was implemented using PyTorch software. Similar to ActionFormer [7], we used pre-extracted features from two-stream I3D [10] and R(2+1)D [11] pre-trained on Kinetics. For the THUMOS14 dataset experiment, we chose AdamW as the optimizer and trained the model with a batch size of 8, learning rate of 0.0001, and weight decay of 0.05 for 30 epochs. For the ActivityNet1.3 dataset experiment, we used a batch size of 32, a learning rate of 0.001, a weight decay of 0.05, and training for 10 epochs. Soft-NMS [40] was used for postprocessing, and the experiments were conducted using a system with a single GeForce RTX 3090 GPU.

3) EVALUATION METRIC

We evaluated the performance using intersection over union (IoU) thresholding and the mean average precision (mAP). For the THUMOS14 dataset, we report the IoU thresholds at

[0.3:0.1:0.7]. For the ActivityNet-1.3 dataset, we report the results at the IoU threshold [0.5,0.75,0.95] and the average mAP computed at [0.5:0.05:0.95]. The evaluation metric was applied identically to ActionFormer and TriDet.

B. COMPARISON WITH STATE-OF-THE-ART METHODS

1) THUMOS14 DATASET

Table 1 presents the results of comparison experiments with other SoTA TAD methods for the THUMOS14 dataset. For our experiments on the THUMOS14 dataset, we used I3D [10] as the backbone network. As shown in Tab. 1, our proposed LSTA model exhibits a competitive performance in comparison to existing SoTA models. Notably, when compared to ActionFormer, a baseline model for TAD based on transformers, LSTA exhibited the second-highest performance, with an average mAP of 68.9%, which is a difference of 2.1%. Our proposed model showed a 0.4% lower performance than the best-performing TriDet model. This is because the LTA block of the proposed LSTA was not sufficiently learned because the THUMOS14 dataset was composed of a small amount of data; however, there was no significant difference, and outperform was observed when the IoU thresholds were set to 0.5, 0.6 or 0.7. Our proposed LSTA model stands out for its ability to effectively learn and integrate long-term and short-term representations, which significantly enhances its overall performance.

2) ACTIVITYNET-1.3 DATASET

Table 2 shows the results of the comparison experiments with other SoTA TAD models for the ActivityN-et-1.3 dataset. For the experiments conducted on the ActivityNet-1.3 dataset, we used R(2+1)D [47] as the backbone network. Overall, we achieved a competitive performance in comparison with existing SoTA models. The unique strength of our proposed LSTA model lies in its effective learning and integration of long-term and short-term representations. This capability is crucial for capturing complex temporal patterns inherent in the ActivityNet-1.3 dataset. Our model outperformed TriDet with a 0.3% higher mean average precision (mAP) on ActivityNet-1.3. It is thought that this result is due to differences in the dataset characteristics between THUMOS14 and ActivityNet-1.3. When learning long- and short-term temporal features, more complex temporal patterns must be learned. With more data, the model can learn more accurate temporal features. However, for small datasets, it is difficult for the model to generalize, resulting in lower performance. Therefore, experiments conducted on the ActivityNet-1.3 dataset instead of the THUMOS14 dataset will result in better performance, as shown through the experiment results. TCANet, a two-stage model that uses SlowFast [48] as the backbone network for training, exhibited the best performance. Performance improvements can be expected when using a better backbone network. We expect that the proposed LSTA model trained with the SlowFast backbone will achieve a better performance in future studies.

TABLE 3. Effectiveness evaluation of the proposed LSTA's modules.

Module	CLM	STA	LTA	Avg.
Ours			✓	36.7
Ours		✓	✓	36.9
Ours	✓	✓		36.9
Ours	✓	✓	✓	**37.1**

C. ABLATION EXPERIMENTS

In this section, we validate the performance of LSTA based on the results of several ablation experiments using two datasets.

1) EFFECTIVENESS OF EACH MODULE

Table 3 shows the results of our experiments, indicating how each module of the proposed model affects its overall performance. As shown in Tab. 3, the best average mAP performance of 37.1% was achieved on the ActivityNet-1.3 when all proposed modules were used together. In addition, the performance was slightly improved when the CLM was applied. Compared to when STA and LTA were utilized separately, the performance was better when both modules were used together. It can therefore be concluded that STA and LTA can be flexibly combined to effectively learn long-term temporal features.

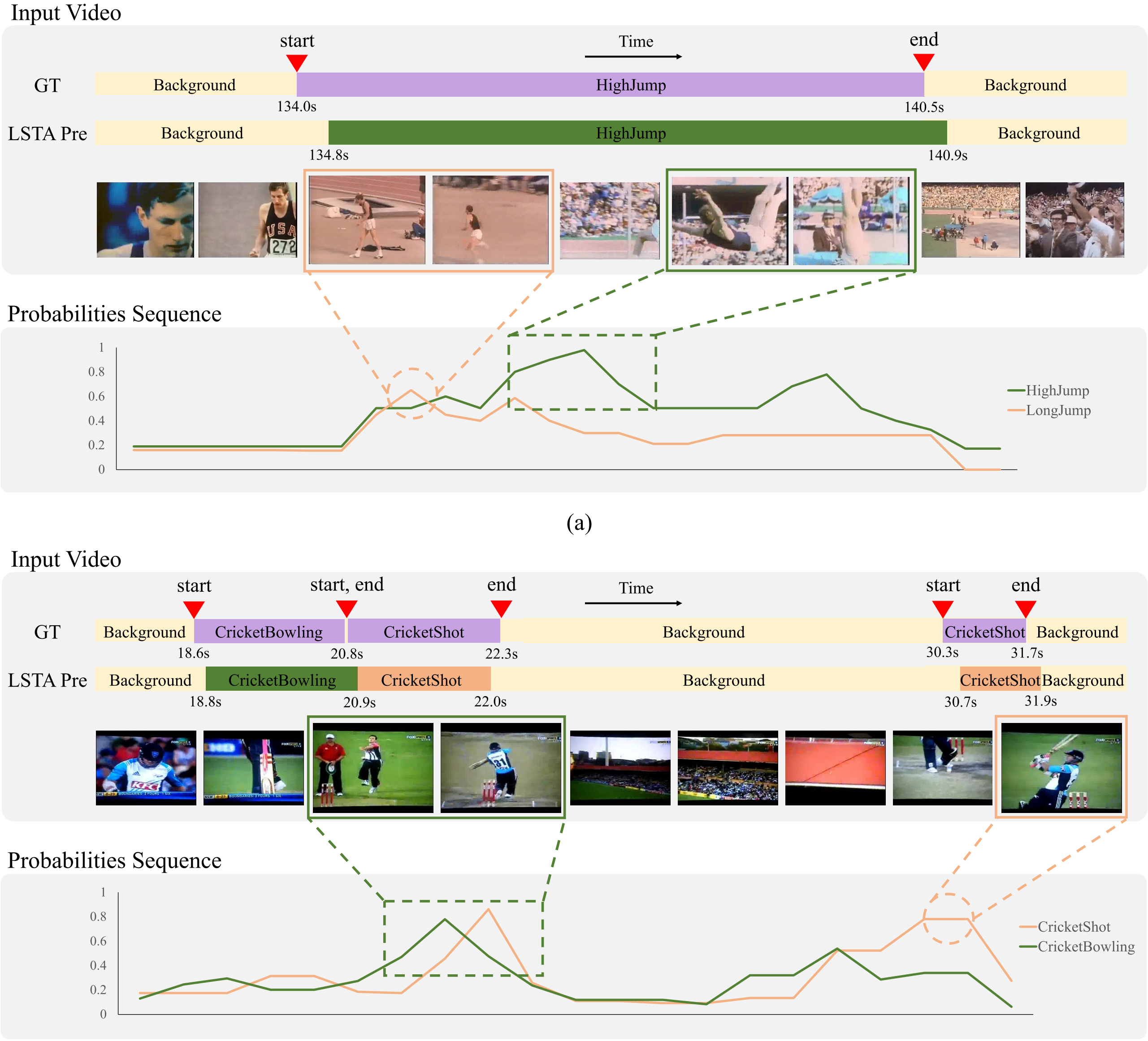
2) EFFECTIVENESS OF THE NUMBER OF SHORT-TERM WINDOW SIZES

Figure 4 shows the changes in performance as the short-term window size varies using the ActivityNet-1.3 dataset. The experiment results indicate that the model performs best when the window size is 12, and the performance gradually decreases as the window size deviates. Therefore, in this study, we conducted experiments using a fixed window size of 12. Because the length of the actions in a video sequence may vary across datasets, different window sizes can affect the performance depending on the characteristics of each dataset.

3) INFORMATION VISUALIZATION

Proposed LSTA provides interpretability of action recognition results by visualizing the frame-wise action class probabilities for each video sequence. Figure 5(a) visualizes the relative action probabilities over time for the THUMOS14 test set, only showing the top two classes with the highest probabilities. As shown in Fig. 5(a), the video is correctly classified as *'High Jump'*, which is the same as the ground truth. The probability of *'Long Jump'* is high for a short period from 132 seconds to 136 seconds, which is interpreted as the scene where a person is jumping before performing *'High Jump'*. However, the probability of *'High Jump'* is high overall in the remaining frames, so the video is correctly judged as *'High Jump'* in the end.

Additionally, LSTA produces fine-grained predictions of action boundaries and shows accurate predictions for both



(a)

(b)

FIGURE 5. Visualization results on the THUMOS14 test set. (a) Frame-by-frame single action prediction of a person in a video with moving cameras and various scene transitions, (b) Frame-by-frame complex action prediction of multiple people in videos with moving cameras and various scene transitions.

actions and action boundaries. Figure 5(b) is a complex action video containing two actions. In the video sequence, the probability of *'Cricket Bowling'* is high for about 20.9 seconds, and then at 20.9 seconds, the video scene changes and the probability of *'Cricket Shot'* becomes high. Then, at 23 seconds, where the scene changes again, the probabilities of both classes are low, so it is predicted as *'Background'*. At 30.7 seconds, it is accurately predicted as *'Cricket Shot'*, which has the highest probability.

In summary, the proposed LSTA accurately predicts the action over time in both examples, and also predicts the start and end times of the action with a very small difference of about 0.1 seconds to 0.4 seconds. This means that the proposed LSTA model has improved performance by properly learning temporal long- and short-term features. Additionally, by visualizing the frame-wise action prediction results in a way that is easy to understand, the proposed LSTA improves the interpretability of action recognition.

V. DISCUSSION AND CONCLUSION

In this study, we proposed a novel LSTA algorithm consisting of classification and regression heads to enhance the long- and short-term temporal features of TAD. A classification head that predicts the action labels through STA and LTA blocks and a regression head that predicts the action boundaries were applied. By learning long- and short-term relationships in untrimmed videos through transformer-based STA and LTA blocks, the LSTA algorithm showed excellent performance in predicting actions and visualization for action interpretation.

Through various experiments on the most representative datasets of TAD, the THUMOS14 and AcitvityNet-1.3 datasets, the proposed model was able to achieve similar performance on THUMOS14 and outperformed the previous SoTA model on ActivityNet-1.3. In addition, ablation studies confirmed that each proposed module has a positive impact on overall performance. We believe that LSTA is a promising approach to TAD because it can learn long-range temporal dependencies and consider a variety of temporal cues. In future research, our proposed model is expected to contribute to future TAD research and can potentially be applied in applications such as video retrieval. However, handling video datasets remains a challenging issue, leading many TAD models to rely on extracted features. To address these limitations, future research will focus on developing efficient video feature extractors and exploring methods to enable end-to-end learning for performance enhancement. Additionally, we plan to continue researching to enable real-time action detection in video streams.

REFERENCES

[1] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "BMN: Boundary-matching network for temporal action proposal generation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3888–3897.

[2] F. C. Heilbron, J. C. Niebles, and B. Ghanem, "Fast temporal activity proposals for efficient detection of human actions in untrimmed videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1914–1923.

[3] V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem, "DAPs: Deep action proposals for action understanding," in *Proc. 14th Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 768–784.

[4] S. Buch, V. Escorcia, B. Ghanem, and J. C. Niebles, "End-to-end, single-stream temporal action detection in untrimmed videos," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 1–20.

[5] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2933–2942.

[6] Z. Shou, J. Chan, A. Zareian, K. Miyazawa, and S.-F. Chang, "CDC: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1417–1426.

[7] C.-L. Zhang, J. Z. Wu, and Y. Li, "Actionformer: Localizing moments of actions with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 492–510.

[8] D. Shi, Y. Zhong, Q. Cao, L. Ma, J. Lit, and D. Tao, "TriDet: Temporal action detection with relative boundary modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 18857–18866.

[9] A. Vaswani, "Attention is all you need," in *Proc. 31st Conf. Neural Inf. Process. Syst.*, 2017, pp. 1–11.

[10] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4724–4733.

[11] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6450–6459.

[12] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. ECCV*, 2016, pp. 1–36.

[13] T. Lin, X. Zhao, and Z. Shou, "Single shot temporal action detection," in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 98–996.

[14] C. Lin, C. Xu, D. Luo, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu, "Learning salient boundary feature for anchor-free temporal action localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3319–3328.

[15] S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. C. Niebles, "SST: Single-stream temporal action proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6373–6382.

[16] J. Tan, J. Tang, L. Wang, and G. Wu, "Relaxed transformer decoders for direct action proposal generation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13526–13535.

[17] M. Xu, C. Zhao, D. S. Rojas, A. Thabet, and B. Ghanem, "G-TAD: Sub-graph localization for temporal action detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10153–10162.

[18] D. Ahn, S. Kim, H. Hong, and B. Chul Ko, "STAR-transformer: A spatio-temporal cross attention transformer for human action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 3330–3339.

[19] C. R. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-attention multi-scale vision transformer for image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 347–356.

[20] M. Xu, Y. Xiong, H. Chen, X. Li, W. Xia, Z. Tu, and S. Soatto, "Long short-term transformer for online action detection," in *Proc. NeurIPS*, 2021, pp. 1086–1099.

[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16 × 16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2020, pp. 1–21.

[23] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jgou, "Training data-efficient image transformers & distillation through attention," in *Proc. ICML*, 2021, pp. 10347–10357.

[24] Z. Xia, X. Pan, S. Song, L. E. Li, and G. Huang, "Vision transformer with deformable attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4784–4793.

[25] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 548–558.

[26] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–16.

[27] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.

[28] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lucic, and C. Schmid, "ViViT: A video vision transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 6816–6826.

[29] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" in *Proc. ICML*, 2021, pp. 1–3.

[30] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video Swin transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 3192–3201.

[31] J. Wang and L. Torresani, "Deformable video transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14033–14042.

[32] S. Kim, J. Nam, and B. C. Ko, "Vit-Net: Interpretable vision transformers with neural tree decoder," in *Proc. ICML*, 2022, pp. 11162–11172.

[33] H. Itaya, T. Hirakawa, T. Yamashita, H. Fujiyoshi, and K. Sugiura, "Visual explanation using attention mechanism in actor-critic-based deep reinforcement learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–10.

[34] S. Abnar and W. Zuidema, "Annual meeting of the association for computational linguistics," in *Proc. ACL*, 2021, pp. 4190–4197.

[35] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 782–791.

[36] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Italy, Oct. 2017, pp. 2999–3007.

[37] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. CVPR*, 2019, pp. 658–666.

[38] H. Idrees, A. R. Zamir, Y. G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah, "The THUMOS challenge on action recognition for videos in the wild," in *Proc. CVPR*, vol. 155, Feb. 2017, pp. 1–23.

[39] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS—Improving object detection with one line of code," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Italy, Oct. 2017, pp. 5561–5569.

[40] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 961–970.

[41] Z. Qing, H. Su, W. Gan, D. Wang, W. Wu, X. Wang, Y. Qiao, J. Yan, C. Gao, and N. Sang, "Temporal context aggregation network for temporal action proposal refinement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 485–494.

[42] C. Zhao, A. Thabet, and B. Ghanem, "Video self-stitching graph network for temporal action localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13638–13647.

[43] Z. Zhu, W. Tang, L. Wang, N. Zheng, and G. Hua, "Enriching local and global contexts for temporal action localization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13496–13505.

[44] D. Shi, Y. Zhong, Q. Cao, J. Zhang, L. Ma, J. Li, and D. Tao, "React: Temporal action detection with relational queries," in *Proc. ECCV*, 2022, pp. 105–121.

[45] X. Liu, Q. Wang, Y. Hu, X. Tang, S. Zhang, S. Bai, and X. Bai, "End-to-end temporal action detection with transformer," *IEEE Trans. Image Process.*, vol. 31, pp. 5427–5441, 2022.

[46] S. Nag, X. Zhu, J. Deng, Y.-Z. Song, and T. Xiang, "DiffTAD: Temporal action detection with proposal denoising diffusion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, France, Oct. 2023, pp. 10362–10374.

[47] H. Alwassel, S. Giancola, and B. Ghanem, "TSP: Temporally-sensitive pretraining of video encoders for localization tasks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 3166–3176.

[48] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "SlowFast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6201–6210.

[49] A. Gomaa, T. Minematsu, M. M. Abdelwahab, M. Abo-Zahhad, and R.-I. Taniguchi, "Faster CNN-based vehicle detection and counting strategy for fixed camera scenes," *Multimedia Tools Appl.*, vol. 81, no. 18, pp. 25443–25471, Mar. 2022.

[50] A. Gomaa, M. M. Abdelwahab, and M. Abo-Zahhad, "Efficient vehicle detection and tracking strategy in aerial videos by employing morphological operations and feature points motion analysis," in *Multimedia Tools Application*. Cham, Switzerland: Springer, Jul. 2020, pp. 26023–26043.

DASOM AHN received the B.S. and M.S. degrees in computer engineering from Keimyung University, Daegu, South Korea, in 2018 and 2023, respectively, where she is currently pursuing the Ph.D. degree with the Computer Vision and Pattern Recognition Laboratory. Her current research interests include video action recognition, temporal action detection, and explainable AI (XAI).

JONG-HA LEE (Member, IEEE) received the B.S. degree in electronics engineering from Inha University, South Korea, in 2000, M.S. degree in electrical engineering from New York University, Brooklyn, NY, USA, in 2005, and the Ph.D. degree in electrical engineering from Temple University, Philadelphia, PA, USA. He worked as a Research Staff Member at the Samsung Advanced Institute of Technology. He is currently an Associate Professor with the Department of Biomedical Engineering, Keimyung University, Daegu, South Korea. His research interests include intelligence systems, spectral analysis, and medical image registration.

BYOUNG CHUL KO (Senior Member, IEEE) received the B.S. degree from Kyonggi University, Suwon, South Korea, in 1998, and the M.S. and Ph.D. degrees in computer science from Yonsei University, Seoul, South Korea, in 2000 and 2004, respectively. From 2004 to 2005, he was a Senior Researcher at Samsung Electronics, Suwon, where he worked on the Ubiquitous Robot Companion (URC) Project on the subject of robot event detection and face recognition. He is currently a Professor with the Department of Computer Engineering and the Dean of the College of Engineering, Keimyung University (KMU), Daegu, South Korea, where he is also the Chief of the AI Fusion Research Center. His current research interests include interpretable machine learning, deep model compression, explainable AI, and graph neural networks. He has received excellent paper awards from several conferences. Furthermore, he was selected as the Best Achieved Researcher with Keimyung University, in 2023. He is active in various international journals and conferences and serves as president of the IEIE AI and Signal Processing Society.