

A Comparative Study of Temporal Models and Skeleton Representations for Temporal Action Segmentation in Dance

Jian-Xu Wang¹ and Hendri Sutrisno^{2,*}

¹ Department of Information Management, National Dong Hwa University, Taiwan; 611335110@gms.ndhu.edu.tw

² Department of Industrial Engineering and Management, National Taipei University of Technology, Taiwan; hendri@gmail.ntut.edu.tw

* Correspondence

<https://doi.org/10.5392/IJoC.2026.22.2.001>

Manuscript Received 19 March 2026; Received 27 May 2026; Accepted 7 June 2026

Abstract: Temporal Action Segmentation (TAS) assigns an action label to each frame in a sequence. While TAS has been extensively studied in contexts such as daily activities, sports, and surgical videos, its application to continuous dance motion derived from video-based skeleton data is still limited. This paper evaluates vision-based TAS for continuous dance motion using video-derived skeleton sequences, eliminating the need for wearable sensors. A labeled dance dataset was created from video clips, including multi-person scenes where each performer was extracted and treated as an individual sequence. The dataset also incorporates duet performances, which introduce greater variation in execution and recording conditions. To examine how pose representation and temporal modeling affect segmentation quality, we evaluated two skeleton settings (BODY-25 and COCO-18) and two representative Temporal Action Segmentation (TAS) models [Multi-Stage Temporal Convolutional Network++ (MS-TCN++) and Frame-Action Cross-Attention Temporal Modeling (FACT)] under a unified protocol. Both models effectively segmented dance actions from skeleton data. FACT consistently achieved higher segment-level performance than MS-TCN++, particularly in F1 and Edit scores, while BODY-25 generally outperformed COCO-18 across both models. Using BODY-25, FACT achieved F1@10 and F1@25 scores of 94.6 and an F1@50 score of 90.6, compared to MS-TCN++'s scores of 89.7, 89.7, and 89.1, respectively. These results demonstrate that both pose representation and temporal modeling materially affect segmentation quality in continuous dance motion.

Keywords: Temporal Action Segmentation; Skeleton-based Action Recognition; Dance Action Analysis; Pose Estimation; Computer Vision; Deep Learning

1. Introduction

Temporal Action Segmentation (TAS) divides an untrimmed sequence into labeled action segments [1]. It is a fundamental problem in computer vision because many human activities unfold as continuous streams containing multiple sequential actions. TAS has been applied to daily activities [2], sports [3], and surgical procedures [4], where both action identity and temporal boundaries are important. However, segmentation remains difficult for fine-grained actions that are temporally connected and visually similar [5, 6].

Human action analysis is important in domains such as sports training, rehabilitation, and dance performance analysis, where body motion must be interpreted over time. This analysis can be performed using wearable sensors or visual data. Sensor-based approaches capture body movement effectively but depend on body-worn devices and are therefore less practical in natural settings [7, 8]. In contrast, video-based approaches are non-intrusive and easier to deploy. With recent advances in pose estimation, human motion can be represented as video-extracted skeleton sequences [2], [9], making video-based skeleton analysis a promising direction for TAS because it emphasizes body movement while reducing appearance and background variation [10, 11].

Despite this promise, video-based TAS from skeleton data remains challenging in continuous motion scenarios where action boundaries are ambiguous and execution varies across performers. Dance is a representative case: actions are linked by smooth transitions, and the same movement may differ in timing, posture, body control, experience, viewpoint, and recording conditions. These properties make dance a demanding benchmark for fine-grained temporal segmentation and motion understanding [12, 13].

However, existing work still provides limited evidence on how well video-based skeleton TAS handles such continuous motion, especially under realistic multi-person settings and variations in execution [14]. In particular, it remains unclear how pose representation and temporal modeling jointly affect segmentation quality for continuous dance motion represented by video-derived skeleton sequences.

To address this gap, we used dance as a case study for vision-based TAS from skeleton sequences. In this work, we constructed a labeled dance dataset from video clips for frame-level temporal segmentation. Because many of the source videos contained multiple dancers, each performer was cropped and treated as an individual sequence. The dataset also included duet performances in which two students performed the same choreography simultaneously, introducing less controlled conditions and greater variation in execution. On this basis, we investigated two technical questions. First, we compared BODY-25 [15] and COCO-18 [16] to examine whether a denser keypoint representation improves temporal segmentation. Second, we benchmarked two representative TAS architectures, Multi-Stage Temporal Convolutional Network++ (MS-TCN++) [17] and Frame-Action Cross-Attention Temporal Modeling (FACT) [18], to assess how temporal convolution and transformer-based modeling behave under a unified protocol in continuous motion. These models were selected because they represent two contrasting temporal modeling strategies for long untrimmed sequences. MS-TCN++ captures temporal context through multi-stage dilated temporal convolutions and iterative refinement, whereas FACT combines frame-level features with action-level representations through cross-attention. This comparison is particularly relevant for dance segmentation, where smooth transitions and weak action boundaries require both temporal continuity and segment-level coherence.

The contributions of this work are threefold. First, we investigate the feasibility of TAS from video-based skeletal sequences without the use of wearable body sensors. Second, we propose a practical data preparation pipeline for multi-person dance videos, encompassing performer-level skeleton extraction and frame-level action annotation. Third, we conduct a structured comparison of two skeleton representations, BODY-25 and COCO-18, and two TAS models, MS-TCN++ and FACT, under a shared experimental protocol. Beyond dance analysis, the study is also relevant to other fine-grained motion analysis problems that require segmentation from ordinary video, such as sports training and movement-based skill assessment. Collectively, these contributions elucidate how pose representation and temporal modeling choices affect TAS performance in a challenging, fine-grained motion analysis domain.

2. Materials and Methods

2.1 Frame-wise TAS

TAS is commonly formulated as a dense sequence labeling task in which each frame in a sequence is assigned an action label. In this setting, the model must recognize the category of actions while maintaining the temporal boundaries between adjacent actions [19]. Accordingly, many TAS methods are designed to generate frame-wise predictions while modeling temporal context across long sequences.

A major line of work in this setting is represented by the Multi-Stage Temporal Convolutional Network (MS-TCN) and its variants. Farha and Gall proposed MS-TCN as a strong frame-wise baseline built around temporal convolution and multi-stage refinement [19]. Its importance lies in demonstrating that iterative refinement can substantially improve temporal segmentation quality while maintaining efficient frame-level prediction. Li et al. later introduced MS-TCN++ [17], which strengthened this framework by improving the quality of early-stage predictions and, in turn, the effectiveness of subsequent refinements. As a result, MS-TCN++ has become a widely used benchmark for frame-wise TAS and a representative model for studying how refinement-based temporal modeling performs. This model family is particularly relevant to the present study because it emphasizes frame-wise prediction refinement, which is important when action boundaries must be localized precisely.

The influence of this model family is also evident in applied domains. MS-TCN-based approaches have been adopted in surgical workflow analysis [20], used as key baselines in few-shot skeleton-based TAS [21],

and included in structured sports benchmarks such as figure skating [22]. These studies indicate that frame-wise refinement remains a central paradigm for TAS, particularly in settings where temporal precision is essential.

2.2 Higher-level temporal modeling and skeleton-based motion analysis

Although frame-wise refinement is effective, it may be less sufficient when actions are highly similar, extended in duration, or connected by smooth transitions. This limitation has motivated work on higher-level temporal modeling, where the goal is not only to refine frame predictions but also to capture a broader action structure. Hierarchical Action Segmentation Refiner (HASR) is an example of this direction [23]. As a plug-in refinement module, it shows that segment- and video-level context can further improve temporal consistency, supporting the view that TAS benefits from representations beyond the frame level.

A more explicit step in this direction is FACT, proposed by Lu et al. [18]. FACT combines frame-level representation with action-level temporal modeling, allowing the model to capture both local motion cues and longer-range dependencies among actions. This design is particularly relevant when temporal meaning depends not only on short-term motion patterns but also on the ordering and interaction of action phases across a sequence. FACT has achieved strong results on standard TAS benchmarks [18] and has also been applied to 3D pose-based figure-skating segmentation [24], suggesting its suitability for continuous and fine-grained motion domains. In contrast to MS-TCN++, which relies on iterative temporal convolutional refinement, FACT explicitly models interactions between frame features and action-level representations. This architectural difference makes the comparison between the two models informative for continuous dance motion, where both local transitions and segment-level coherence are important.

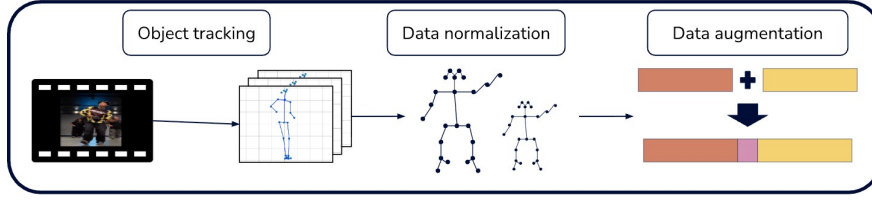
At the same time, skeleton-based motion representation has become increasingly important in human action understanding. Compared with RGB-based features, skeleton data focus more directly on body movement and are less affected by appearance and background variation. This makes them especially relevant for motion-centered tasks in which pose dynamics carry the main discriminative information. Previous studies have also explored skeleton-based TAS in low-data or structured-motion settings [21], but this line of work remains more limited than RGB-based TAS, particularly for continuous movement scenarios with substantial performer variation. This representation is especially suitable for dance analysis, where fine-grained motion patterns are expressed through posture, limb coordination, and movement transitions rather than scene context. Based on the literature above, two research gaps remain. First, although TAS has progressed rapidly, much of the literature still focuses on standard activity benchmarks, with less attention to continuous human movement settings characterized by smooth transitions and performance-dependent variation. Second, while MS-TCN++ and FACT represent two important temporal modeling strategies, there is still limited evidence on how they compare under a skeleton-based setting for fine-grained motion analysis. These gaps motivate the present study, which evaluates MS-TCN++ and FACT on video-derived dance skeleton sequences and examines whether the denser BODY-25 representation provides more reliable segmentation cues than COCO-18.

3. Methodology

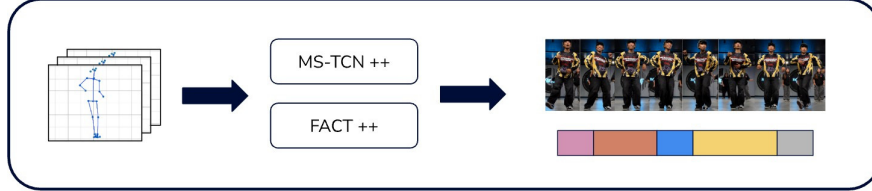
3.1 Processing pipeline

We evaluate two temporal action segmentation (TAS) models, Multi-Stage Temporal Convolutional Network++ (MS-TCN++) and Frame-Action Cross-Attention Temporal Modeling (FACT), on a video-based skeleton dataset. The processing pipeline consists of five stages: video collection, pose extraction, performer tracking, skeleton pre-processing, and frame-wise TAS. Given a video containing one or more performers, 2D skeletons are first extracted with OpenPose [25]. The detected skeletons are then associated across frames to form one temporally consistent sequence for each performer. Next, normalization, missing-keypoint completion, and sequence augmentation are applied. The resulting sequence is used as input to a TAS model that predicts one action label for each frame. The overall workflow is shown in Figure 1.

1. Pre-Processing



2. Temporal Action Segmentation

**Figure 1.** Overview of the proposed pipeline

Let $V = \{I_t\}_{t=1}^T$ denote an input video of length T , where I_t is the frame at time t . After pose estimation and tracking, each performer is represented by a skeleton sequence $K = \{K_t\}_{t=1}^T$, Where

$$K_t = [k_{t,1}, k_{t,2}, \dots, k_{t,J}]^T \in \mathbb{R}^{J \times 2}$$

And J denotes the number of body joints. Here, $k_{t,j} \in \mathbb{R}^2$ is the 2D coordinate of joint j at frame t . After pre-processing, the normalized sequence is denoted by $\tilde{K} = \{\tilde{K}_t\}_{t=1}^T$. Each $\tilde{K}_t$ is reshaped into a frame feature vector $\mathbf{x}_t \in \mathbb{R}^{2J}$, and the full sequence is written as

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T]^T \in \mathbb{R}^{T \times 2J}$$

The corresponding frame-level ground-truth labels are

$$\mathbf{y} = [y_1, y_2, \dots, y_T], \quad y_t \in \{1, \dots, A\}$$

where A denotes the number of action classes. A TAS model outputs a frame-wise prediction matrix $\mathbf{P} \in \mathbb{R}^{T \times A}$, and the predicted label at frame t is given by

$$\hat{y}_t = \arg \max_{a \in \{1, \dots, A\}} P_{t,a}$$

3.2 Skeleton pre-processing

3.2.1 Pose extraction and performer tracking

The source videos contain multiple performers in the same scene. OpenPose [25] is applied to each frame I_t to obtain a set of detected skeletons $D_t = \{K_t^{(1)}, K_t^{(2)}, \dots, K_t^{(N_t)}\}$, where N_t denotes the number of detected people at frame t . Because these detections do not preserve a stable person index across frames, temporal association is required before action segmentation.

Performer-specific sequences are constructed using SORT-based tracking [26]. For each detected skeleton $K_t^{(n)}$ a bounding box $B_t^{(n)} = [x_{min}, y_{min}, x_{max}, y_{max}]$ is computed from the joint coordinates and enlarged by a factor of 1.2. SORT then predicts track boxes $\hat{B}_t^{(m)}$ and matches them to detections using the Intersection-over-Union (IoU) criterion in Eq. (1), followed by mini minimum-cost assignment.

$$\text{IoU}(\hat{B}_t^{(m)}, B_t^{(n)}) = \frac{|\hat{B}_t^{(m)} \cap B_t^{(n)}|}{|\hat{B}_t^{(m)} \cup B_t^{(n)}|} \quad (1)$$

After association, each track yields one temporally consistent skeleton sequence $K^{(m)} = \{K_t^{(m)}\}_{t=1}^{T_m}$, where T_m denotes the sequence length of performer m . In the following sections, the performer index is omitted for simplicity, and the tracked sequence is written as $K = \{K_t\}_{t=1}^T$.

3.2.2 Skeleton normalization and missing-keypoint completion

The skeletons extracted from OpenPose vary across performers in position and body scale. Each frame is therefore normalized by centering at the neck and scaling by shoulder width. Let $\mathbf{k}_{t,nc}$ denote the neck joint, and let $\mathbf{k}_{t,l}$ and $\mathbf{k}_{t,r}$ denote the left and right shoulder joints. The normalized coordinate of joint j at frame t is defined in Eq. (2).

$$\tilde{\mathbf{k}}_{t,j} = \frac{\mathbf{k}_{t,j} - \mathbf{k}_{t,nc}}{\|\mathbf{k}_{t,l} - \mathbf{k}_{t,r}\|_2} \quad (2)$$

Equation (2) centers each skeleton at the neck and rescales it by shoulder width, reducing variation caused by body position and scale.

Some frames contain missing keypoints due to occlusion, truncation, or pose estimation failure. Instead of zero filling, missing joints are completed using a mean relative-vector strategy. Let r denote a reference joint, with the neck used whenever available. For each frame t and joint j , the relative vector is defined in Eq. (3).

$$\mathbf{r}_{t,j} = \mathbf{k}_{t,j} - \mathbf{k}_{t,r} \quad (3)$$

Using the relative vectors in Eq. (3), the mean relative vector of joint j at frame index t is computed over all available samples, as defined in Eq. (4).

$$\bar{\mathbf{r}}_{t,j} = \frac{1}{|\Omega_{t,j}|} \sum_{n \in \Omega_{t,j}} \mathbf{r}_{t,j}^{(n)} \quad (4)$$

In Eq. (4), $\Omega_{t,j}$ denotes the set of samples in which joint j is available at frame index t . If joint j is missing at frame t , it is restored by Eq. (5).

$$\hat{\mathbf{k}}_{t,j} = \mathbf{k}_{t,r} + \bar{\mathbf{r}}_{t,j} \quad (5)$$

Equation (5) restores the missing joint by combining the reference-joint coordinate with the mean relative offset.

3.2.3 Sequence augmentation

To increase data diversity, a motion interpolation strategy inspired by Xu et al. [21] is applied. Transitions between adjacent clips are smoothed by linear interpolation at the clip boundary.

Let $k_j^{(1)}$ and $k_j^{(2)}$ denote the coordinates of joint j in the last frame of the first clip and the first frame of the second clip, respectively. For an interpolation window of length S , the interpolated joint coordinate at step i is defined in Eq. (6)

$$k_{i,j}^{int} = (1 - \alpha_i)k_j^{(1)} + \alpha_i k_j^{(2)}, \quad \alpha_i = \frac{i}{S}, \quad i = 0, 1, \dots, S \quad (6)$$

Equation (6) linearly interpolates the coordinate of joint j between the two clip boundaries, where α_i denotes the interpolation ratio at step i .

To filter unstable interpolated poses, a torso-angle outlier check is applied. Let $k_{t,hip}$ denote the center hip joint. The torso vector is defined as $\mathbf{v}_t^{torso} = \mathbf{k}_{t,hip} - \mathbf{k}_{t,neck}$, and the corresponding torso angle is $\theta_t = \text{atan2}(\mathbf{v}_t^{(y)}, \mathbf{v}_t^{(x)})$. The interquartile range (IQR) of the torso-angle sequence is then computed, and frame t is marked as abnormal if it satisfies Eq. (7)

$$\theta_t < Q_1 - 1.5 \text{ IQR} \quad \text{or} \quad \theta_t > Q_3 + 1.5 \text{ IQR} \quad (7)$$

In Eq. (7), Q_1 and Q_3 denote the first and third quartiles of the torso-angle distribution. Frames identified as abnormal are replaced by forward fill.

After pre-processing, each frame $\tilde{\mathbf{K}}_t \in \mathbb{R}^{J \times 2}$ is reshaped into a vector $\mathbf{x}_t \in \mathbb{R}^{2J}$. The resulting sequence is written as $\mathbf{X} \in \mathbb{R}^{T \times 2J}$, which is used as input to the TAS model.

3.3 Prediction models

Given the pre-processed input sequence $\mathbf{X} \in \mathbb{R}^{T \times 2J}$, TAS predicts one action label for each frame. Two models are evaluated in the same skeleton-based setting: MS-TCN++ and FACT. MS-TCN++ represents refinement-based temporal convolutional modeling, whereas FACT combines frame-level temporal features

with action-level representations through cross-attention. This contrast makes the comparison informative for continuous dance motion, where both local transitions and segment-level coherence are important.

MS-TCN++ produces frame-wise predictions through multi-stage temporal refinement [17]. Let S denote the number of stages and L the number of temporal convolution layers in each stage. The first stage takes $\mathbf{X}$ as input, and each later stage takes the prediction of the previous stage. The output of stage s is $\mathbf{P}^{(s)} \in \mathbb{R}^{T \times A}$.

In the first stage, the hidden representation $\mathbf{H}_l^{(1)}$ is updated with a Dual Dilated Layer:

$$\mathbf{Z}_{l,d_1} = W_{d_1} * \mathbf{H}_{l-1}^{(1)} + b_{d_1} \quad (8)$$

$$\mathbf{Z}_{l,d_2} = W_{d_2} * \mathbf{H}_{l-1}^{(1)} + b_{d_2} \quad (9)$$

$$\hat{\mathbf{H}}_l^{(1)} = \text{ReLU}([\mathbf{Z}_{l,d_1}, \mathbf{Z}_{l,d_2}]) \quad (10)$$

$$\mathbf{H}_l^{(1)} = \mathbf{H}_{l-1}^{(1)} + W_{1 \times 1} \hat{\mathbf{H}}_l^{(1)} + b_{1 \times 1} \quad (11)$$

In Eqs. (8)–(11), W_{d_1} and W_{d_2} denote temporal convolution kernels with different dilation factors, and $*$ denotes 1D temporal convolution.

For stages $s \geq 2$, the hidden representation is refined by standard dilated residual layers:

$$\hat{\mathbf{H}}_l^{(s)} = \text{ReLU}(W_d * \mathbf{H}_{l-1}^{(s)} + b_d) \quad (12)$$

$$\mathbf{H}_l^{(s)} = \mathbf{H}_{l-1}^{(s)} + W_{1 \times 1} \hat{\mathbf{H}}_l^{(s)} + b_{1 \times 1} \quad (13)$$

After layer L , each stage outputs frame-wise class probabilities:

$$\mathbf{P}^{(s)} = \text{softmax}(W_{\text{cls}} \mathbf{H}_L^{(s)} + b_{\text{cls}}) \in \mathbb{R}^{T \times A} \quad (14)$$

The final prediction of MS-TCN++ is $\mathbf{P}_{\text{MS}} = \mathbf{P}^{(S)}$.

FACT maps the same input sequence $\mathbf{X} \in \mathbb{R}^{T \times 2J}$ to frame-level features and action-level tokens [18]. The input is first projected to a latent frame representation:

$$\mathbf{F}_0 = \phi(\mathbf{X}) \in \mathbb{R}^{T \times D} \quad (15)$$

In Eq. (15), $\phi(\cdot)$ denotes a learnable projection and D denotes the hidden dimension. In parallel, the model initializes M learnable action tokens $\mathbf{A}_0 \in \mathbb{R}^{M \times D}$.

For update block s , the action tokens are first updated from the frame features:

$$(\mathbf{A}'_s, \Lambda_s^a) = \text{CrossAttn}_{f \rightarrow a}(\mathbf{A}_{s-1}, \mathbf{F}_{s-1}) \quad (16)$$

$$\mathbf{A}_s = \text{Trans}(\mathbf{A}'_s) \quad (17)$$

The updated action tokens are then used to refine the frame features:

$$(\mathbf{F}'_s, \Lambda_s^f) = \text{CrossAttn}_{a \rightarrow f}(\mathbf{F}_{s-1}, \mathbf{A}_s) \quad (18)$$

$$\mathbf{F}_s = \text{Trans}(\mathbf{F}'_s) \quad (19)$$

After the last block, the action and frame branches produce $\mathbf{P}^a = FC(\mathbf{A}) \in \mathbb{R}^{M \times (A+1)}$ and $\mathbf{P}^f = FC(\mathbf{F}) \in \mathbb{R}^{T \times A}$, respectively. After removing the null-label component from $\mathbf{P}^a$, the remaining token predictions are denoted by $\mathbf{P}_{\setminus \emptyset}^a \in \mathbb{R}^{M \times A}$. These predictions are aligned to the frame axis using the final attention map:

$$\mathbf{P}^{a \rightarrow f} = \Lambda_{\text{last}}^f \mathbf{P}_{\setminus \emptyset}^a \in \mathbb{R}^{T \times A} \quad (20)$$

The final frame-wise prediction is obtained by fusing the two branches:

$$\mathbf{P}_{\text{FACT}} = w\mathbf{P}^{a \rightarrow f} + (1 - w)\mathbf{P}^f \quad (21)$$

In Eq. (21), $w \in [0,1]$ notes the fusion weight. Both models output a class probability matrix $\mathbf{P} \in \mathbb{R}^{T \times A}$, and the predicted label at frame t defined as

$$\hat{y}_t = \arg \max_{a \in \{1, \dots, A\}} \mathbf{P}_{t,a} \quad (22)$$

The complete predicted sequence is $\hat{y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T]$.

4. Results

4.1 Dataset

The source material was collected from a dance tutorial on YouTube and includes one instructor and four students. Two additional participants recorded supplementary action clips based on the same tutorial. Each performer was treated as an independent sample for temporal action segmentation. The dataset contains 7 action classes, and all action segments were manually annotated.

All recordings were captured from a frontal view with no obvious body occlusion. Since the analysis focuses on body movement rather than facial or identity-related cues, all sequences were checked to ensure clear visibility of the body region from the neck downward. An example frame is shown in Figure 2.



Figure 2. Example frame from the dance recordings

The dataset is publicly available at [27]. After annotation, the sequences were processed using the workflow in Figure 1, including pose extraction with OpenPose, skeleton pre-processing, and data augmentation. All sequences were resampled to 30 fps, and BODY-25 and COCO-18 keypoints were extracted for comparison.

After preprocessing and augmentation, the dataset contains 761 training sequences and 57 test sequences. During training, the training set was divided into 10 folds, with one fold used for validation in each run.

4.2 Setup

Two models were evaluated: MS-TCN++ [17] and FACT [18]. All experiments were conducted on Ubuntu LTS 22.04 with a single NVIDIA RTX 4070 SUPER. Both models were trained for 150 epochs using Adam with an initial learning rate of 10^{-4} . The batch size was set to 1, and the dropout rate was set to 0.3.

For MS-TCN++, a 4-stage architecture was used. Each stage contains 10 dilated temporal convolution layers with a kernel size $k = 3$, a hidden channel width of 64, and 1×1 convolutions for input projection and frame-wise classification. For FACT, the model consists of one input block and three update blocks. In each block, the frame branch contains 10 dilated temporal convolution layers with a kernel size $k = 3$. The action branch contains 6 multi-head attention layers in the input block and 1 multi-head attention layer in each update block. The number of attention heads was set to 8, the number of action tokens to 20, and the channel width to 64.

The models were evaluated using frame-wise accuracy (Acc), Edit score (Edit), and segmental F1 score at IoU thresholds of 10%, 25%, and 50%, denoted as $F1@10$, $F1@25$, and $F1@50$. Acc measures the percentage

of correctly classified frames. However, frame-wise accuracy alone does not fully reflect segmentation quality, because a prediction may classify many individual frames correctly while still producing fragmented segments or inaccurate boundaries. Edit measures the normalized Levenshtein distance between the predicted and ground-truth segment sequences and therefore reflects ordering errors and segmentation consistency. F1@10, F1@25, and F1@50 measure segment-level overlap after matching predicted and ground-truth segments at the corresponding IoU threshold. These segment-level metrics are more appropriate for TAS because they evaluate both action identity and temporal boundary quality.

4.3 Training dynamics

Figures 3 and 4 show the training and validation loss curves for MS-TCN++ and FACT under BODY-25 and COCO-18. In both skeleton settings, MS-TCN++ shows a sharper loss decrease during the early epochs, whereas FACT decreases more gradually and with less smooth trajectories.

These curves indicate different optimization behaviors between the two models. MS-TCN++ reaches a low-loss regime earlier, while FACT converges more progressively over training. The gap between BODY-25 and COCO-18 is visible in both models, but the loss curves alone do not determine the final segmentation quality. A plausible explanation is that FACT uses frame-action attention-based modeling, which introduces a more complex optimization process and may therefore require more gradual convergence. By contrast, MS-TCN++ relies on dilated temporal convolutional layers and iterative refinement, which may allow more direct optimization in the early training stage. However, faster convergence does not necessarily imply better segment-level performance. As shown later in the quantitative and qualitative results, FACT produces more stable segment predictions than MS-TCN++, especially under more challenging recording conditions.

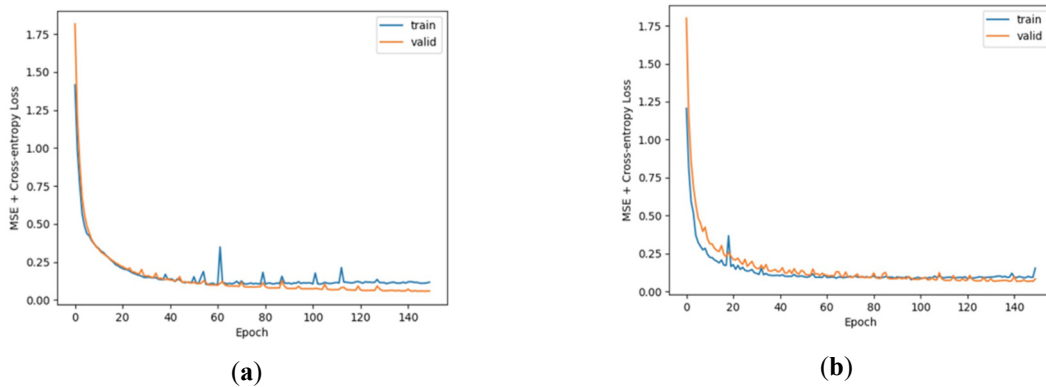


Figure 3. MS-TCN++ training and validation loss curves under BODY-25 (a) and COCO-18 (b)

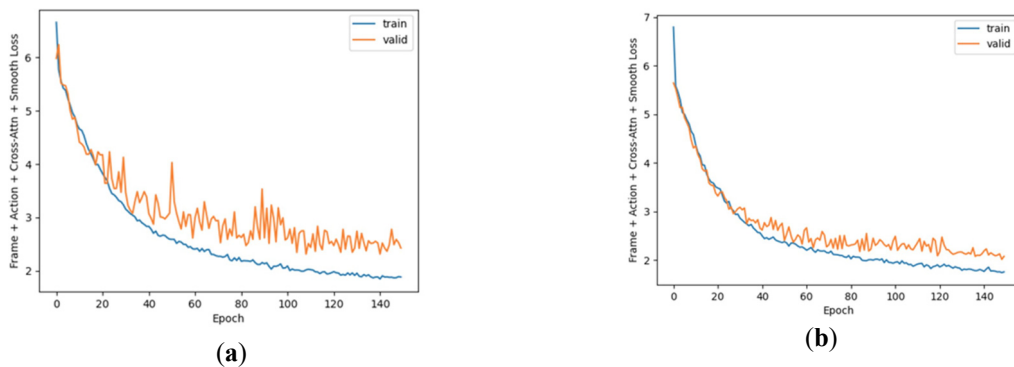


Figure 4. FACT training and validation loss curves under BODY-25 (a) and COCO-18 (b)

4.4 Quantitative results

Table 1 shows two consistent patterns. First, FACT achieves better segment-level performance than MS-TCN++ under both skeleton settings. Second, BODY-25 yields better segmentation results than COCO-18 for both models.

Table 1. Quantitative comparison of MS-TCN++ and FACT under BODY-25 and COCO-18

	BODY-25					COCO-18				
	F1@{10,25,50}			Edit	Acc	F1@{10,25,50}			Edit	Acc
MS-TCN++ [17]	89.7	89.7	89.1	90.9	95.1	74.0	74.0	73.7	86.5	96.1
FACT [18]	94.6	94.6	90.6	89.8	92.6	88.6	88.2	82.9	91.7	92.2

Comparing the two models, FACT outperforms MS-TCN++ on most segment-level metrics, with the largest margin under COCO-18. Under BODY-25, FACT improves F1@10 and F1@25 from 89.7 to 94.6 and F1@50 from 89.1 to 90.6, although its Edit score is slightly lower than that of MS-TCN++ (89.8 vs. 90.9). Under COCO-18, FACT improves F1@10 from 74.0 to 88.6, F1@25 from 74.0 to 88.2, F1@50 from 73.7 to 82.9, and Edit from 86.5 to 91.7. These results indicate that FACT produces more accurate action segments and more consistent temporal structure, especially when the skeleton representation is reduced from BODY-25 to COCO-18.

Comparing the two skeleton representations, BODY-25 consistently outperforms COCO-18 in F1 and Edit. For MS-TCN++, the drop from BODY-25 to COCO-18 is substantial across all segment-level metrics. For FACT, the same pattern is observed, although the degradation is smaller. This result suggests that the denser BODY-25 representation provides more reliable motion cues for temporal segmentation.

Frame-wise accuracy does not follow the same trend. For example, MS-TCN++ achieves slightly higher Acc under COCO-18 than under BODY-25, despite markedly worse F1 and Edit. This difference shows that accuracy alone is not sufficient for evaluating TAS. A model may classify many individual frames correctly while still producing fragmented segments or inaccurate boundaries, as illustrated in Figure 5. For this reason, segment-level metrics such as F1 and Edit provide a more reliable basis for comparison in the present task.



Figure 5. Example of fragmented action segmentation. The predicted labels match many individual frames, but the segment boundaries remain inconsistent with the ground truth.

5. Discussion

5.1 Analysis on real performers

To examine model behavior on real performance sequences, Figure 6 compares the ground-truth labels with the predictions from MS-TCN++ and FACT under the two skeleton settings. In the figure, each color segment corresponds to one action label.

In the YouTube performer examples on the left of Figure 6, MS-TCN++ exhibits several clear error patterns, including merged adjacent segments, short-term jitter near action boundaries, and unstable transitions between neighboring actions. These errors are especially visible near the beginning of the sequence, where MS-TCN++ under COCO-18 merges multiple intervals into a single twist segment. By contrast, FACT aligns more closely with the ground truth, producing cleaner segment overlaps and fewer spurious cuts, although minor boundary offsets remain.

The participant-recorded examples on the right of Figure 6 show greater variation in execution and recording conditions. Under this more difficult setting, errors increase for both models, but the degradation is more pronounced for MS-TCN++, which shows more frequent mislabeling and less stable temporal segmentation. In the second half of the sequence, COCO-18 produces clearer prediction errors, while BODY-25 still shows some local mislabeling. FACT remains comparatively cleaner, especially under BODY-25, although some over-segmentation is still visible under COCO-18.

Overall, the qualitative comparison indicates that MS-TCN++ is more prone to over-segmentation and unstable boundary prediction, whereas FACT produces cleaner and more temporally coherent action segments. This difference is most evident when FACT is combined with BODY-25. These observations are consistent

with the quantitative results in Table 1 and suggest that frame-action modeling, together with a richer keypoint representation, helps preserve segment consistency in continuous dance motion.

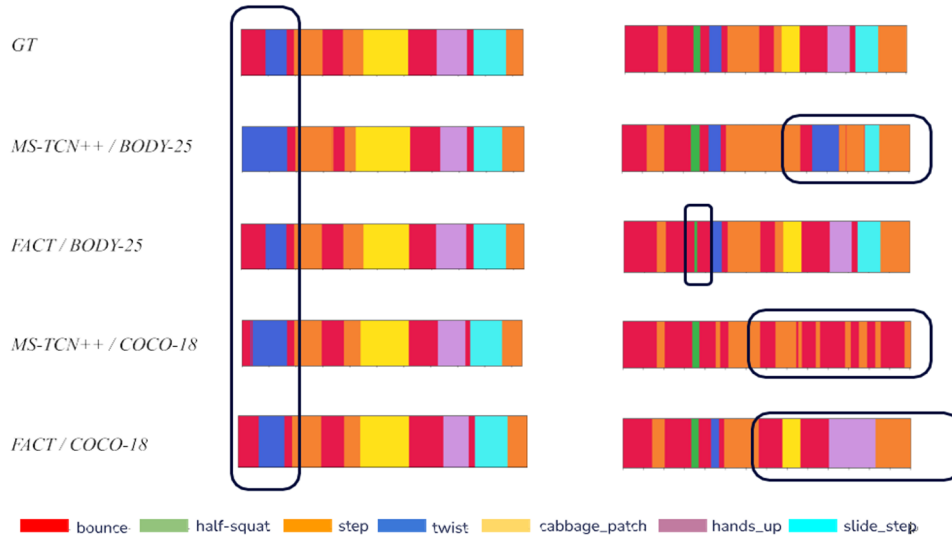


Figure 6. Qualitative comparison of ground-truth and predicted action segments on YouTube performer videos (left) and participant-recorded videos (right). The YouTube examples show minor boundary inconsistencies near the beginning of the sequence, whereas the participant-recorded examples show more pronounced errors near the end, including over-segmentation and unstable transitions in MS-TCN++ and mild over-segmentation in FACT under COCO-18. FACT with BODY-25 produces the most coherent segment structure.

5.2 Effect of skeleton representation

Skeleton representation is especially relevant in dance action segmentation because subtle differences in posture, foot placement, and torso coordination can affect both action recognition and boundary localization. In this task, the level of pose detail influences how well the model captures the fine-grained motion cues required for segmentation.

Figure 7 shows the per-action F1@50 scores under the BODY-25 and COCO-18 settings. Over-all, BODY-25 performs better than COCO-18 for most action classes, and FACT performs better than MS-TCN++ across the same action set. This pattern is consistent with the overall results in Table 1, where BODY-25 achieves higher F1 and Edit scores than COCO-18 for both models, suggesting that the denser representation provides more informative cues for segment-level dance segmentation. The advantage is especially visible for actions such as bounce, cabbage-patch, twist, and step, where more detailed lower-body and full-body motion cues may help the model distinguish segment boundaries more clearly.

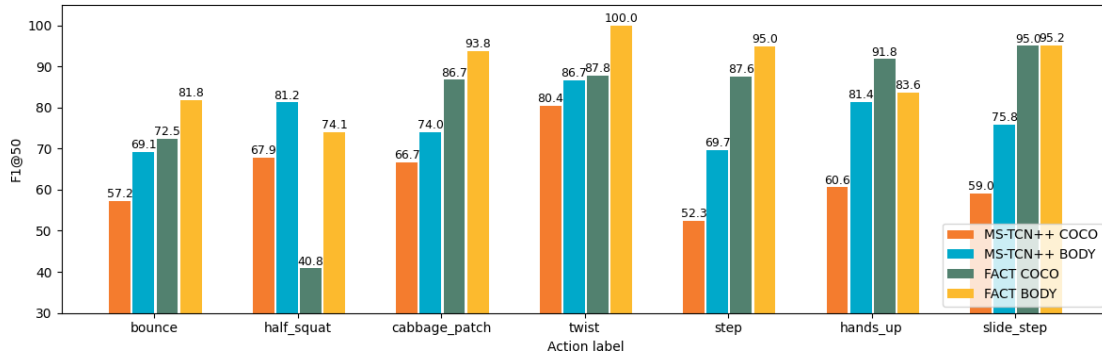


Figure 7. Per-action F1@50 comparison between BODY-25 and COCO-18

Two exceptions are observed for half-squat and hands-up. For half-squat, the movement mainly consists of a downward squat followed by an upward rise, so the dominant cue may come from overall vertical body motion rather than finer joint detail. In this case, the additional joints in BODY-25 appear to provide limited

extra benefit, especially for FACT. For hands-up, the hands move around the waist area, and the torso may also move during the action. Here, the additional torso-related joints in BODY-25 may not always provide stable discriminative information, which may reduce their benefit for segmentation.

These observations suggest that the value of a richer skeleton representation is action-dependent. When the additional joints capture motion detail that is relevant to the action, BODY-25 tends to improve segmentation performance. When the added joints provide less stable or less relevant cues, the gain becomes smaller. Even so, the overall pattern in Figure 7 still favors BODY-25 over COCO-18 for this task.

6. Conclusion

This work addressed the limited evaluation of vision-based temporal action segmentation (TAS) from skeleton sequences in continuous human motion by using dance as a case study. The results show that video-derived skeleton data can support frame-level dance segmentation without wearable sensors and that multi-person dance videos can be converted into usable performer-level sequences through the proposed data preparation pipeline.

The experiments also answer the two technical questions posed in the introduction. First, the comparison between BODY-25 and COCO-18 shows that a denser keypoint representation generally improves segment-level performance. Second, the comparison between MS-TCN++ and FACT shows that the choice of temporal modeling materially affects segmentation quality in continuous motion. Under the evaluated conditions, FACT achieved better segment-level results than MS-TCN++ in most cases, while BODY-25 generally outperformed COCO-18 across both models. These findings indicate that both pose representation and temporal modeling contribute to the quality of temporal segmentation in continuous dance motion.

Taken together, the study makes three contributions: it demonstrates the feasibility of skeleton-based TAS without wearable sensors, establishes a practical pipeline for performer-level extraction and annotation in multi-person dance videos, and provides a structured comparison of skeleton representation and TAS architecture under a shared protocol. These results also suggest that fine-grained motion segmentation can be performed from ordinary video, with potential relevance to movement-based analysis tasks beyond dance.

At the same time, the findings should be interpreted within the scope of the current dataset. The videos were mainly captured from a frontal view, which provides a clear observation of lateral body motion but may limit the representation of front-to-back movement. The current action set also emphasizes extended limb motion more than subtle spine-related movement, and the use of 2D skeletons further limits the analysis of torso articulation. In addition, the dataset is small and still requires manual frame-level labeling, which constrains scalability.

Future work should therefore extend the evaluation to more diverse dance settings with additional performers, camera viewpoints, recording conditions, and movement types. It would also be useful to explore 3D pose estimation, richer skeleton designs, and more scalable labeling strategies such as semi-automatic annotation or weakly supervised segmentation. Beyond dance, the pro-posed pipeline may also be adapted to other continuous-motion domains, including sports training, rehabilitation, and movement-based skill assessment.

Acknowledgments: The work was supported by the National Science and Technology Council of Taiwan (NSTC) with grant number 112-2222-E-027-006-MY2 (H.S.). The authors would like to thank the reviewers and editors for their constructive comments, which helped improve the quality of this manuscript.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Author Contributions: Jian-Xu Wang contributed to manuscript preparation, experimental work, and analysis. Hendri Sutrisno contributed to conceptualization, supervision, methodology, manuscript revision, and final approval.

References

- [1] G. Ding, F. Sener, and A. Yao, "Temporal Action Segmentation: An Analysis of Modern Techniques," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 1011–1030, 2022, doi: <https://doi.org/10.1109/TPAMI.2023.3327284>.
- [2] D. Gong, G. Medioni, and X. Zhao, "Structured time series analysis for human action segmentation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1414–1427, 2013, doi: <https://doi.org/10.1109/TPAMI.2013.244>.

- [3] C. Direkoğlu and N. E. O'Connor, "Temporal segmentation and recognition of team activities in sports," *Mach. Vis. Appl.*, vol. 29, no. 5, pp. 891–913, 2018, doi: <https://doi.org/10.1007/s00138-018-0944-9>.
- [4] G. Quellec, M. Lamard, B. Cochener, and G. Cazuguel, "Real-time segmentation and recognition of surgical tasks in cataract surgery videos," *IEEE Trans. Med. Imaging*, vol. 33, no. 12, pp. 2352–2360, 2014, doi: <https://doi.org/10.1109/TMI.2014.2340473>.
- [5] C. Lea, A. Reiter, R. Vidal, and G. D. Hager, "Segmental spatiotemporal CNNs for fine-grained action segmentation," in *European conference on computer vision*, pp. 36–52, 2016, doi: https://doi.org/10.1007/978-3-319-46487-9_3.
- [6] Y. H. Li, K. Y. Liu, S. L. Liu, L. Feng, and H. Qiao, "Involving distinguished temporal graph convolutional networks for skeleton-based temporal action segmentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 1, pp. 647–660, 2023, doi: <https://doi.org/10.1109/TCSVT.2023.3285416>.
- [7] B. M. Booth et al., "Multimodal human and environmental sensing for longitudinal behavioral studies in naturalistic settings: Framework for sensor selection, deployment, and management," *J. Med. Internet Res.*, vol. 21, no. 8, p. e12832, 2019, doi: <https://doi.org/10.2196/12832>.
- [8] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE communications surveys & tutorials*, vol. 15, no. 3, pp. 1192–1209, 2012, doi: <https://doi.org/10.1109/SURV.2012.110112.00192>.
- [9] M. Shi et al., "Motionet: 3d human motion reconstruction from monocular video with skeleton consistency," *Acm Transactions on Graphics (tog)*, vol. 40, no. 1, pp. 1–15, 2020, doi: <https://doi.org/10.1145/3407659>.
- [10] H. Liu, Y. Niu, K. Zeng, C. Liu, M. Hu, and Q. Song, "Beyond-Skeleton: Zero-shot Skeleton Action Recognition enhanced by supplementary RGB visual information," *Expert Syst. Appl.*, vol. 273, p. 126814, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.126814>.
- [11] H. Duan et al., "Skeletr: Towards skeleton-based action recognition in the wild," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13634–13644, 2023, doi: <https://doi.org/10.1109/ICCV51070.2023.01254>.
- [12] K. Endo, S. Tsuchida, T. Fukusato, and T. Igarashi, "Automatic Dance Video Segmentation for Understanding Choreography," in *Proceedings of the 9th International Conference on Movement and Computing*, pp. 1–9, 2024, doi: <https://doi.org/10.1145/3658852.3659076>.
- [13] Y. Wang and J. Gong, "Research on Dance-Specific Action Recognition Method Based on Spatio-Temporal Skeleton Graph," in *Proceedings of the 2024 International Conference on Sports Technology and Performance Analysis*, pp. 98–104, 2024, doi: <https://doi.org/10.1145/3723936.3723952>.
- [14] H. Sutrisno, "Video-Based Assessment of Apprentice and Expert Performances in Dance," 2025 *The International Conference on Convergence Content (ICCC)*, pp. 249–250, 2025.
- [15] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7291–7299, 2017, doi: <https://doi.org/10.1109/CVPR.2017.143>.
- [16] Z. Cao, G. Hidalgo, T. Simon, S. E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, 2019, doi: <https://doi.org/10.1109/TPAMI.2019.2929257>.
- [17] S. Li, Y. A. Farha, Y. Liu, M. M. Cheng, and J. Gall, "MS-TCN++: Multi-Stage Temporal Convolutional Network for Action Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 6647–6658, 2020, doi: <https://doi.org/10.1109/TPAMI.2020.3021756>.
- [18] Z. Lu and E. Elhamifar, "FACT: Frame-Action Cross-Attention Temporal Modeling for Efficient Action Segmentation," 2024 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18175–18185, 2024, doi: <https://doi.org/10.1109/CVPR52733.2024.01721>.
- [19] Y. A. Farha and J. Gall, "MS-TCN: Multi-Stage Temporal Convolutional Network for Action Segmentation," 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3570–3579, 2019, doi: <https://doi.org/10.1109/CVPR.2019.00369>.
- [20] B. Zhang, A. Ghanem, A. Simes, H. Choi, A. Yoo, and A. Min, "SWNet: Surgical Workflow Recognition with Deep Convolutional Network," in *International Conference on Medical Imaging with Deep Learning*, 2021. [Online] Available: <https://proceedings.mlr.press/v143/zhang21b/zhang21b.pdf>
- [21] L. Xu, Q. Wang, X. Lin, and L. Yuan, "An Efficient Framework for Few-shot Skeleton-based Temporal Action Segmentation," *Computer Vision and Image Understanding*, vol. 232, 2023, doi: <https://doi.org/10.1016/j.cviu.2023.103707>.

- [22] Z. Gan et al., “SkatingVerse: A large-scale benchmark for comprehensive evaluation on human action understanding,” *IET Comput. Vis.*, vol. 18, no. 7, pp. 888–906, 2024, doi: <https://doi.org/10.1049/cvi2.12287>.
- [23] H. Ahn and D. Lee, “Refining Action Segmentation with Hierarchical Video Representations,” 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 16282–16290, 2021, doi: <https://doi.org/10.1109/ICCV48922.2021.01599>.
- [24] R. Tanaka, T. Suzuki, and K. Fujii, “3D Pose-Based Temporal Action Segmentation for Figure Skating: A Fine-Grained and Jump Procedure-Aware Annotation Approach,” *Proceedings of the 7th ACM International Workshop on Multimedia Content Analysis in Sports*, 2024, doi: <https://doi.org/10.1145/3689061.3689077>.
- [25] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, “Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1302–1310, 2016, doi: <https://doi.org/10.1109/CVPR.2017.143>.
- [26] A. Bewley, Z. Ge, L. Ott, F. T. Ramos, and B. Upcroft, “Simple online and realtime tracking,” 2016 IEEE International Conference on Image Processing (ICIP), pp. 3464–3468, 2016, doi: <https://doi.org/10.1109/ICIP.2016.7533003>.
- [27] H. Sutrisno, *Skeleton-Dance-TAS: Dataset and annotations for skeleton-based temporal action segmentation in dance*, GitHub Repository, 2026. [Online] Available: <https://github.com/hendrisutrisno/skeleton-dance-tas>



© 2026 The Author(s). Copyright remains with the author(s). This article is published by The Korea Contents Association under the Creative Commons Attribution 4.0 International License (CC BY 4.0): <https://creativecommons.org/licenses/by/4.0/>