

Locating X-ray coronary angiogram keyframes via long short-term spatiotemporal attention with image-to-patch contrastive learning

Ruipeng Zhang¹, Binjie Qin¹, *Member, IEEE*, Jun Zhao¹, Yueqi Zhu, Yisong Lv¹, Song Ding¹

Abstract—Locating the start, apex and end keyframes of moving contrast agents for keyframe counting in X-ray coronary angiography (XCA) is very important for the diagnosis and treatment of cardiovascular diseases. To locate these keyframes from the class-imbalanced and boundary-agnostic foreground vessel actions that overlap complex backgrounds, we propose long short-term spatiotemporal attention by integrating a convolutional long short-term memory (CLSTM) network into a multiscale Transformer to learn the segment- and sequence-level dependencies in the consecutive-frame-based deep features. Image-to-patch contrastive learning is further embedded between the CLSTM-based long-term spatiotemporal attention and Transformer-based short-term attention modules. The imagewise contrastive module reuses the long-term attention to contrast image-level foreground/background of XCA sequence, while patchwise contrastive projection selects the random patches of backgrounds as convolution kernels to project foreground/background frames into different latent spaces. A new XCA video dataset is collected to evaluate the proposed method. The experimental results show that the proposed method achieves a mAP (mean average precision) of 72.45% and a F-score of 0.8296, considerably outperforming the state-of-the-art methods. The source code is available at <https://github.com/Binjie-Qin/STA-IPCn>.

Index Terms—X-ray coronary angiography, temporal action localization, spatiotemporal attention, Transformer, contrastive learning

I. INTRODUCTION

IN X-ray coronary angiography (XCA, all acronyms in this paper are listed in Table I) for the diagnosis and treatment of cardiovascular diseases, measuring the contrast-diffusion time span from the two phases of filling and disappearing of contrast agents in myocardial perfusion can

The manuscript was received on April 14, 2022, and accepted on June 16, 2023. This work was partially supported by the Science and Technology Commission of Shanghai Municipality (19dz1200500, 19411951507), National Natural Science Foundation of China (61271320, 82070477), Shanghai Science and Technology Committee (22XD1421800), Shanghai “Rising Stars of Medical Talent” Youth Development Program “Outstanding Youth Medical Talents” (SHWSRS (2021) 099), and Shanghai Jiao Tong University Medical-Engineering Cross Research Fund (ZH2018ZDA19, YG2021QN122, YG2021QN99). (Corresponding authors: Binjie Qin and Song Ding)

Ruipeng Zhang, Binjie Qin and Jun Zhao are with School of Biomedical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China. E-mail: bj Qin@sjtu.edu.cn

Yueqi Zhu is with the Department of Radiology, Shanghai Jiao Tong University Affiliated Sixth People’s Hospital, Shanghai Jiao Tong University, 600 Yi Shan Road, Shanghai 200233, China.

Yisong Lv is with School of Continuing Education, Shanghai Jiao Tong University, Shanghai 200240, China.

Song Ding is with Department of Cardiology, Ren Ji Hospital, School of Medicine, Shanghai Jiao Tong University, Shanghai 200127, China. Email: dingsong@renji.com

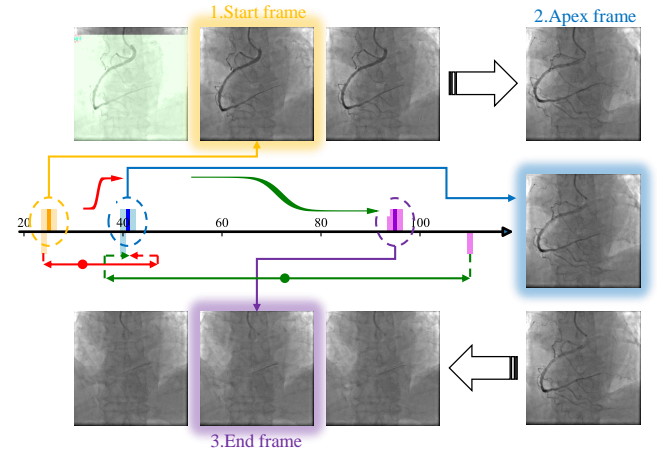


Fig. 1. Start-to-apex-to-end XCA keyframe localization in the two actions (red and green) of contrast filling/disappearing phases. We predict the two mid-points of actions and regress the lengths of action phases to determine the start and end frames, averaging these two frames to obtain the apex keyframe. The three images of each frame are the previous frame of the keyframe, the keyframe, and the next frame of the keyframe.

be directly used to evaluate coronary microvascular function [1], [2]. Locating the start, apex and end keyframes (see Fig. 1) of moving contrast agents for keyframe counting in the two phases revives it as the main mode of decision-making but suffers from the challenging problems: extreme foreground–background imbalance with a very small number of low-contrast foreground vessels that overlap with complex and dynamic backgrounds, subtle changes in foreground action volume and limited inter-keyframe variation, and the missing boundary between the keyframes and surrounding frames (see Fig. 1). Without separating the small number of vessels from the complex and dynamic backgrounds [3], [4], we hardly identified the boundary-agnostic keyframes from the imbalanced and overlapping XCA sequence and rarely classified and then localized these keyframes.

We assume that learning the vessel’s evolving trend (see Fig. 1) by aggregating long short-term spatiotemporal features for segment- and sequence-level dependency modeling is the key solution to the challenging keyframe localization. Specifically, we treat the keyframe extraction as temporal action localization (TAL). As one of the most challenging problems in computer vision, TAL has been studied [5]–[7] for general video sequences but not for the challenging XCA sequences. Recently, Actionformer [6] achieved the best

TAL performance [5], [6]. However, most TAL methods refine discriminative action boundaries from segment-level semantics [7]–[10] and model inter-frame relationships directly based on Transformer architecture, hardly focusing on image-to-patch spatiotemporal features to model the gradually changing small features in video sequence. Besides, Transformer usually divides video sequences into small segments (or snippets) and model temporal relationships in each segment with local-window attention [11], [12], leading to a loss of the long-range inter-segment information exchange. Multiscale Transformer [6] used temporal downsampling to shorten the time length and increase the receptive field of sequence for establishing inter-segment dependencies. However, the loss of long-range inter-segment information is still unavoidable because the downsampling will lead to the loss of temporal information. Therefore, existing methods will ignore a gradually evolving trend of blood vessels in XCA sequence and lead to a loss of long-term dependencies in TAL. To solve these problems, we propose a long short-term spatiotemporal attention network with image-to-patch contrastive learning to refine segment- and sequence-level spatiotemporal attention modeling, increasing the contrastive learning performance for boundary-agnostic XCA keyframe localization. The main contribution of this work is threefold:

1) An effective XCA keyframe localization is proposed to build upon the convolutional long short-term memory (CLSTM) network for learning segment- and sequence-level long short-term dependencies and the Actionformer [6] for modeling short-term attention in sequential XCA segments.

2) A low-rank background patch is selected randomly as a convolutional kernel in patchwise convolutional projection in each frame, effectively projecting foreground/background patches to different latent spaces simultaneously with contrasting image-level foreground/background features via reuse of long short-term spatiotemporal attention.

3) To the best of our knowledge, this is the first study about XCA keyframe localization by exploiting the class-imbalanced small foreground features that are sparsely distributed and overlapped with complex backgrounds. The proposed model obviously outperforms state-of-the-art (SOTA) methods on the collected dataset.

II. RELATED WORKS

A. XCA Sequence Recognition

The XCA sequence provides consecutive frames containing heterogeneous blood vessels that overlap with various interferences, such as anatomical structures, mixed Poisson-Gaussian noises [13], [14], respiratory and cardiac motions. Vessel segmentation [15]–[17] and vessel extraction [3], [18] are the main topics on XCA sequences. Most vessel segmentation methods based on deep learning use an encoder-decoder architecture for single image segmentation and use multidimensional convolution or long short-term memory (LSTM) for sequence processing. For vessel extraction methods, traditional algorithms are mainly built upon grey value or tubular feature representation, simultaneously enhancing the background structures with similar tubular feature artifacts to

TABLE I
ACRONYMS

Acronym	Definition
XCA	X-ray coronary angiography
LSTM	long short-term memory
CLSTM	convolutional long short-term memory
TAL	temporal action localization
SOTA	state-of-the-art
TA	temporal attention
SA	spatial attention
STA	spatiotemporal attention
S-T	short-term attention
L-T	long-term attention
ICon	imagewise contrastive
PCon	patchwise contrastive
IPCon	image-to-patch contrastive
MBR	multiple boundary regression
IoU	intersection over union
TP	true positives
FP	false positives
FN	false negatives
AP	average precision
mAP	mean average precision
P	precision
R	recall
F	F-score
AD	average deviation
DL	deviation list
T-v	T-value
P-v	P-value
CI	confidence interval

introduce more difficulty in subsequent vessel classification or tracking. Recently, by decomposing video sequences into low-rank backgrounds and sparsely distributed foreground objects, robust principal component analysis (RPCA) [19]–[21] has proven to successfully separate moving contrast-filled vessels from complex and dynamic backgrounds in XCA sequences. To address the computational costs and noisy remnants, RPCA-UNet [3] has greatly improved computational efficiency in the excellent restoration of heterogeneous vessel profiles by exploiting patchwise feature selection in an RPCA unrolling network [22]. We refer interested readers to recent comprehensive reviews on XCA vessel extraction [3], [4].

B. Keyframe Extraction for Video Summarization

Keyframe extraction finds a small subset of frames that represent the most representative frames from a video sequence for static video summarization [23], [24], which traditionally includes three main categories: 1) Frame clustering [25], [26] clusters similar frames by feature representation and similarity metrics and then extracts the frame closest to the cluster center as a keyframe. 2) Shot segmentation [27], [28] first detects shots by representing low- and mid-level features of video content and identifying shot boundaries in the original

video and then extracts one or more keyframes from each shot. Both methods lack effective feature representation to distinguish subtle changes between consecutive unstructured frames within poor quality sequence [9], [29]. 3) Sparse coding methods [30], [31] extract a few (sparse) frames while preserving the essential video content, which is best reconstructed as a linear combination of a few selected keyframes. Keyframe dictionary selection [31] and RPCA-based methods [30], [32] used $L_{2,1}$ -norm [32], L_1 -norm [30], or $L_{2,p}$ -norm [31] sparsity constraints to ensure the sparsity of reconstruction coefficients, selecting keyframes as local/global maximums of the norm-regularized reconstruction optimization function. Patch-based sparse representation [33] has been proven to outperform frame-level sparse representation due to its balancing the representativeness of global features and local details.

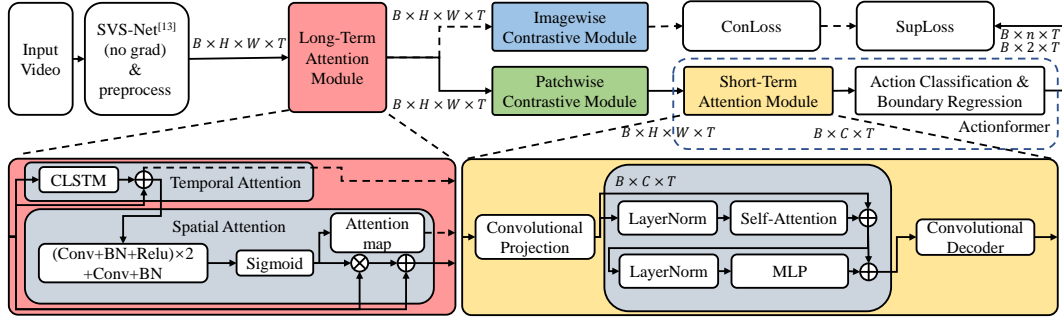


Fig. 2. The proposed network has long-term attention, short-term attention, patchwise and imagewise contrastive modules. The modules in the dotted box are from Actionformer [6]. The solid arrows represent the data streams and the dashed arrows represent data streams that can be chosen to activate or not.

imagewise contrastive module can be activated and deactivated alternately during the first ten epochs of training to accelerate its convergence. These modules can be skipped and then degenerate into the original Actionformer [6].

A. Problem Definition

We define TAL [5], [6] for input XCA sequence $X = \{x_1, x_2, \dots, x_T\}$ that considers x_i the i th frame and T the sequence length. What we want is an action list $\hat{Y} = \{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T\}$, where $\hat{y}_i = (n_i, start_i, end_i)$ is responsible for predicting the action category $n_i \in \{0, 1, \dots, G\}$, start frame number $start_i$ and end frame number end_i . When n_i is 0, \hat{y}_i represents the filling action of contrast agents, otherwise it represents the disappearing action of contrast agents (see Fig. 1). Specifically, the proposed TAL method predicts two mid-points of actions and regresses the lengths of action phases to determine the start and end frames of the filling/disappearing actions. The apex frame is determined by the average of the end frame of the filling action and the start frame of the disappearing action.

B. Preprocessing

We use SVS-Net [15] to extract 3D spatiotemporal features from consecutive frames in sequential segments at the beginning of training and inference. We choose 64×64 size deep spatiotemporal features as the processed high-level features per segment. Since each segment contains four consecutive frames, the sequential temporal information is compressed into a visual tube to enrich 3D spatiotemporal information and reduce long-term memory loss in subsequent CLSTM-based spatiotemporal attention modeling (Section III-C). This is important to take full advantage of CLSTM's capabilities in modeling long short-term spatiotemporal attention. These preprocessed deep features of segments are called original input features, which have dimensions of $B \times H \times W \times T$ with B , H , W and T representing the batch size, the height and width of the image, and the time, respectively.

C. Long-Term Attention Module

To highlight long-term spatiotemporal features for modeling up and down evolution trend of vessel changes, a long-term spatiotemporal attention module is built upon CLSTM with two parts, i.e., temporal and spatial attentions (see Fig. 2). For

the temporal attention, the convolutional-recurrent learning of CLSTM has proven to capture the evolution trend of temporal changes [60]. CLSTM changes the fully connected layer of LSTM into a convolutional layer when calculating the gates by input X_t and hidden state h_{t-1} so that CLSTM handles spatial data better. Each $CLSTM_{cell}(X_t, c_{t-1}, h_{t-1})$ [61] at time t in the CLSTM has formulation:

$$\begin{aligned}
 i_t &= \sigma(W_{xi} X_t + W_{hi} h_{t-1} + W_{ci} c_{t-1} + b_i) \\
 f_t &= \sigma(W_{xf} X_t + W_{hf} h_{t-1} + W_{cf} c_{t-1} + b_f) \\
 c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc} X_t + W_{hc} h_{t-1} + b_c) \\
 o_t &= \sigma(W_{xo} X_t + W_{ho} h_{t-1} + W_{co} c_{t-1} + b_o) \\
 h_t &= o_t \odot \tanh(c_t) \oplus 3050 \mathbb{W}_{t-1} s01167(h)d
 \end{aligned} \quad (1)$$

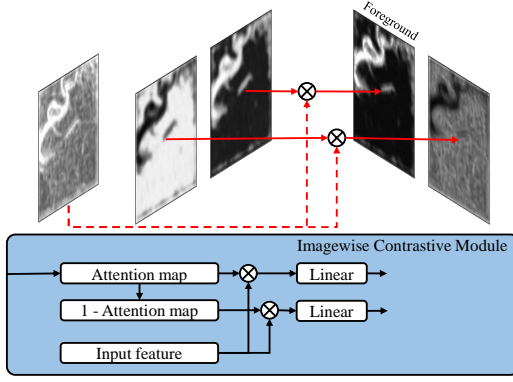


Fig. 3. The architectures of the imagewise contrastive module and patchwise contrastive module. (a) Imagewise contrastive module. It uses the attention map generated by the long-term attention module to separate the foreground vessels from the background. (b) Patchwise contrastive module. A random patch from contrast-free background images is used as a convolutional kernel to project foreground and background patches into different spaces for enhancing the contrast between foreground and background.

where $Conv(\cdot)$ is the convolution operation, $BN(\cdot)$ is the batch normalization, $Relu(\cdot)$ is the activation function and $(\cdot)^2$ means two repeated operations. The spatiotemporal attention (STA) module is then defined as follows:

$$STA(X) = SA(X + TA(X)) \quad (5)$$

Thus, when $TA(\cdot)$, $SA(\cdot)$ and $STA(\cdot)$ are used respectively, this module has the following output:

$$\begin{aligned} Out(X) &= X + Sigmoid(TA(X)) \\ Out(X) &= X + X \cdot Sigmoid(SA(X)) \\ Out(X) &= X + X \cdot Sigmoid(STA(X)) \end{aligned} \quad (6)$$

The corresponding attention map is shown in Fig. 5(b)-(d). The long-term attention module receives the original input features processed by SVS-Net with $B \times H \times W \times T$ and does not change the feature size, so that the attention map can be used to calculate the Hadamard product with the input features for contrasting the foreground/background as described in Section III-E.

D. Short-Term Attention Module

The short-term attention module receives the features processed by the long-term attention module and patchwise contrastive module with $B \times H \times W \times T$. Each frame $x_i \in \mathbb{R}^{H \times W=4096}$ of the input sequence $X \in \mathbb{R}^{H \times W \times T}$ is flattened and projected into $C = 512$ dimensions using convolution $E(\cdot)$ to form $\hat{X} = [E(x_1), E(x_2), \dots, E(x_T)]g$ with $\hat{X} \in \mathbb{R}^{T \times C}$. A Transformer encoder (see the yellow block of Fig. 2) is then used for encoding via layer normalization, self-attention, MLP and a residual structure. Here, self-attention mechanism [52] is implemented by projecting \hat{X} to three different subspaces Q, K, V as formulated:

$$Q = \hat{X}W_Q, K = \hat{X}W_K, V = \hat{X}W_V \quad (7)$$

where $W_Q, W_K \in \mathbb{R}^{C \times \hat{C}}$ and $W_V \in \mathbb{R}^{C \times \hat{C}}$ are the projection matrices, \hat{C} and \check{C} are the hidden dimension and output dimension, $Q, K \in \mathbb{R}^{T \times \hat{C}}$ and $V \in \mathbb{R}^{T \times \hat{C}}$ are the projection

results. In our practice, both \hat{C} and \check{C} are equal to 128. Generally, self-attention is calculated as:

$$SelfAtten(X) = softmax(QK^T / \sqrt{d_k})V \quad (8)$$

where K^T denotes the transpose of K , and $QK^T \in \mathbb{R}^{T \times T}$ denotes the correlation matrix between frames. Then, the $softmax$ activation function is used to normalize the correlation coefficient and multiplied by V for weighting. d_k is the dimension of the key [63] in Transformer, which is equal to \hat{C} . $\sqrt{d_k}$ is used to avoid a large value appearing in the correlation matrix and causing a small activation function gradient. Multi-head self-attention (MSA) mechanism and multiscale Transformer [6] are also used in our practice and are ignored in the equations for simplicity.

Due to the high complexity of Transformer described in Section II-C, all of the self-attention methods use windows to reduce computational consumption and lead to the lack of long-term dependency. To alleviate this problem, a multiscale Transformer [6] is used to increase the receptive field by downsampling on the temporal domain, which may lose information. Thus, we solely use short-term attention to process the features extracted by the long-term attention module. After encoding, the obtained features are decoded by convolution as [6]. The dimensions of the output features are $B \times C \times T$.

E. Imagewise Contrastive Module

To learn the subtle and contrastive differences between foreground and background for identifying the start and end frames, we introduce an image-to-patch contrastive learning [59] module (see Fig. 3) to enable the network to better distinguish the foreground from background in the absence of pixel-level labels. Contrastive learning [59] forces the same class (foreground/background) close by and different classes far apart, which is often used for self-supervised learning or semisupervised learning by constructing positive and negative pairs for unlabeled data. Similar to [64] that used image-level attention map for contrastive learning, we calculate the Hadamard product of the long-term attention map and the

feature processed by TA to obtain the vessel features and background features as follows:

$$\begin{aligned} \text{Foreground} &= X \text{ AttentionMap} \\ \text{Background} &= (1 - X) \text{ AttentionMap} \end{aligned} \quad (9)$$

where X is the features processed by TA and AttentionMap is the attention map generated by STA in Section III-C. According to the annotations labeled by clinicians, we select a time t that must contain the foreground vessels. Then, Foreground_t can be regarded as a positive case, and Background_t can be regarded as a negative case (see Fig. 3(a)). Foreground_t and Background_t are processed by the fully connected layer to generate vectors representing their features before the usage of contrastive learning. A positive and negative pair has thus been successfully constructed.

A specific setting for batch size of two is designed for generating more pairs for contrastive learning such that each sequence in this setting solely generates one positive and negative pair. After obtaining two positive and negative pairs from different sequences, a soft-nearest-neighbors contrastive loss is employed to increase the similarity between same category cases (i.e., same foreground or background) and reduce the similarity between different category cases:

$$\text{ConLoss} = \sum_{i \in F} \log \frac{\sum_{j \in 2F; i \neq j} e^{\cos(f_i; f_j)} = \tau}{\sum_{j \in 2F; i \neq j} e^{\cos(f_i; f_j)}} \quad (10)$$

where \cos denotes the cosine similarity function, F is the number of sampled foreground/background, f_i denotes the i th of F , F_i is the number of sampled foregrounds/background that is similar to f_i , and τ denotes the temperature parameter. This loss function minimizes the feature gap of the same categories (foreground/background), maximizes the feature gap between foreground and background, and forces the attention map to distinguish the foreground/background with the largest difference in the original input features. It can be used in the first five odd epochs of training (epoch = 1, 3, 5, 7, 9).

The reason why ConLoss is solely activated in the first five odd epochs is based on the following two observations from experiments: 1) ConLoss can converge quickly, so adding it in the beginning of training is sufficient. Activating ConLoss during the whole training can affect the optimization of the main loss that is defined in Section III-H; 2) Activating and deactivating ConLoss alternately rather than activating ConLoss continuously can help the network explore more potential optimum solutions instead of limited suboptimum solutions in optimization space.

F. Patchwise Contrastive Module

To solve class-imbalance and imperceptible differences between foreground vessels and vessel-like background disturbances, we further compare and contrast the foreground and background samples at the patchwise scale. Patchwise contrastive learning has recently been studied in a few works [65], [66]. Unfortunately, there is still no feasible method to construct positive and negative pairs of XCA sequences at the patchwise scale because it is not known whether the patch contains the small number of foreground vessels, though there

is a contrast-free XCA frames that solely contains background images during the initial phase of XCA imaging.

We exploit random patch projection [67], [68] to design a patchwise contrastive module (see Fig. 3(b)), where the features processed by the long-term attention module with $B \times H \times W \times T$ are inputted. A 5×5 background patch is selected randomly from the input contrast-free features at $t = 0$ to act as a convolution kernel for projecting all input features. This patchwise contrastive learning is built upon the fact that the convolution of two patches is equivalent to the dot-product of two vectors, reflecting their similarity by calculating the length of projection of one vector on another vector's space in terms of L_2 -norm of vectors. When a contrast-free background patch is used as the convolution kernel, a larger convolution result is obtained if the kernel convolves the foreground patches with a large amount of vessels due to the L_2 -norm of foreground patches being large, which is shown in the green block of Fig. 3(b). On the contrary, if the kernel convolves the background patches, which is shown in the blue block of Fig. 3(b), the convolution result is obviously small. This patchwise contrastive learning can automatically distinguish the foreground and background by projecting foreground/background patches into different spaces.

G. Action Classification and Boundary Regression

After the short-term attention module, a 1D convolution layer with a convolution and activation function are used to map the high-dimensional temporal features to the n dimension to generate the action classification probability $p_t \in \mathbb{R}^{T \times n}$ ($n = 2$), and another layer is used to generate the boundary regression distance $d_t = ($

I. Postprocessing and Inference

What we want in TAL is action list $\hat{Y} = \hat{r}\hat{y}_1, \hat{y}_2g$ where $\hat{y}_i = (n_i, start_i, end_i)$ as described in Section III-A. However, the direct output of the proposed model is $Y = \hat{r}\hat{y}_1, y_2, \dots, y_Tg \in \mathbb{R}^{T \times 4}$ where $y_t = (p_t, d_t)$ is for action classification probability and boundary regression distance that are described in Section III-G. Therefore, we can convert Y into action list \hat{Y} during inference as follows:

$$n_t = \operatorname{argmax}(p_t), start_t = t - d_t^{start}, end_t = t + d_t^{end} \quad (13)$$

This operation can generate an action with the highest probability for each time stamp t . Then, Soft-NMS [71] is used to decrease overlapping background actions. In addition, each category of the two actions (filling/disappearing action) selects an action instance with the highest probability as the final prediction result of the model during the inference. Then, we can obtain two action localization results $\hat{Y} = \hat{r}\hat{y}_1, \hat{y}_2g$ of a sequence. In detail, \hat{y}_1 can be parsed as the mid-point (red solid point) of filling action and the two corresponding red rays in Fig. 1, and \hat{y}_2 can be parsed as mid-point of disappearing action and its two corresponding rays in green color. Then, the three keyframes can be calculated as:

$$\begin{aligned} StartFrame &= start_1 \\ ApexFrame &= (end_1 + start_2)/2 \\ EndFrame &= end_2 \end{aligned} \quad (14)$$

IV. EXPERIMENTAL RESULTS

A. Experimental Materials

Two hundred and sixty clinical XCA sequences were collected from Renji Hospital of Shanghai Jiao Tong University. The length of each sequence ranges from 31 to 379 frames. Because of following the setting of [6], the model can process sequences of different lengths by padding 0. The original dataset has different resolutions, including 512 × 512 and 800 × 800 pixels. Each frame is reshaped to 512 × 512 and processed by SVS-Net [15]. The final resolution of features is 64 × 64 pixels. Each sequence is annotated by two clinicians to obtain the three keyframe locations and frames per second (FPS), which means that the clinicians only need to simply label each sequence. We calculate the average of two clinician annotations as the final annotation. The dataset is converted to the ActivityNet-1.3 dataset [72] format, which contains the action category, start time and end time of actions. During training, the three keyframes are converted to temporal labels by center sampling as [6]. To facilitate comparison with other advanced methods, the dataset is divided randomly into three subsets for training, validation and test, at a ratio of 136:60:4.

B. Evaluation Metrics

Average precision (AP) and mean average precision (mAP) are widely used in TAL [6], [7], being calculated to evaluate the sequence-level performance of the proposed method. This means that a whole action is taken to be compared with the

true action. We define Precision (P), Recall (R) and F-score (F) as

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}, F = \frac{2 \cdot P \cdot R}{P + R} \quad (15)$$

where TP (true positives) is the total number of detected actions whose IoU with the ground truth is higher than the IoU threshold, FP (false positives) indicates the total number of detected actions whose IoU is lower than the threshold, and FN (false negatives) indicates the total number of undetected actions but the ground truth shows that there is an action. The IoU threshold is predesigned. When the IoU between the result and ground truth exceeds the threshold, the TAL result will be considered correct. Therefore, it can evaluate the sequence-level performance. P represents the ratio of the TP among all results, which is used to evaluate the prediction accuracy. R represents the proportion between the correctly detected actions and total actions in the ground truth. F comprehensively considers both the P and R metrics and indicates the overall performance [6], [7], [50]. These metrics range from 0 to 1, and the higher values indicate the better performances. We rank the results according to the confidence score and calculate P and R one by one according to the IoU. Then, a series of P and R values can be obtained and a P-R or P(R) curve can be drawn in the Cartesian coordinate system. The area under the P-R curve has become a general metric to measure the performance of various detection tasks, which is called AP and formulated as:

$$AP = \int_0^1 P(R)dR \quad (16)$$

The average area under the P-R curves with different IoU thresholds is the mAP. AP and mAP mainly evaluate the performance of sequence-level detection.

We also check the frame-level performance from two aspects. First, P, R and F are used to evaluate whether a frame is detected as the correct category. Specifically, we judge whether each frame is the same as the true value. Furthermore, to evaluate the keyframe localization ability, we define the average deviation (AD) as

$$AD = \sum_{i \geq 1} \frac{(jP_s^i - L_s^i + jP_a^i - L_a^i + jP_e^i - L_e^i)}{jIj \cdot 3} \quad (17)$$

where P_s^i is the prediction of the start frame number of the i th sample. P_a^i and P_e^i are the predictions of the apex and end frame number, respectively. L_s^i , L_a^i and L_e^i are the target

fsei

TABLE II
PERFORMANCE OF DIFFERENT SOTA TAL METHODS IN TERMS OF AP AND MAP VALUES

Method	AP@0.3''	AP@0.4''	AP@0.5''	AP@0.6''	AP@0.7''	mAP''
AFSD [7]	73.87	56.75	35.40	14.53	2.93	36.70
TALLFormer [50]	70.75	69.47	57.81	38.04	17.56	50.73
E2E-TAD [51]	83.27	74.13	57.94	42.51	19.22	55.41
Actionformer [6]	90.85	85.25	70.45	52.56	32.62	66.35
Ours	98.44	92.93	80.91	53.85	36.10	72.45

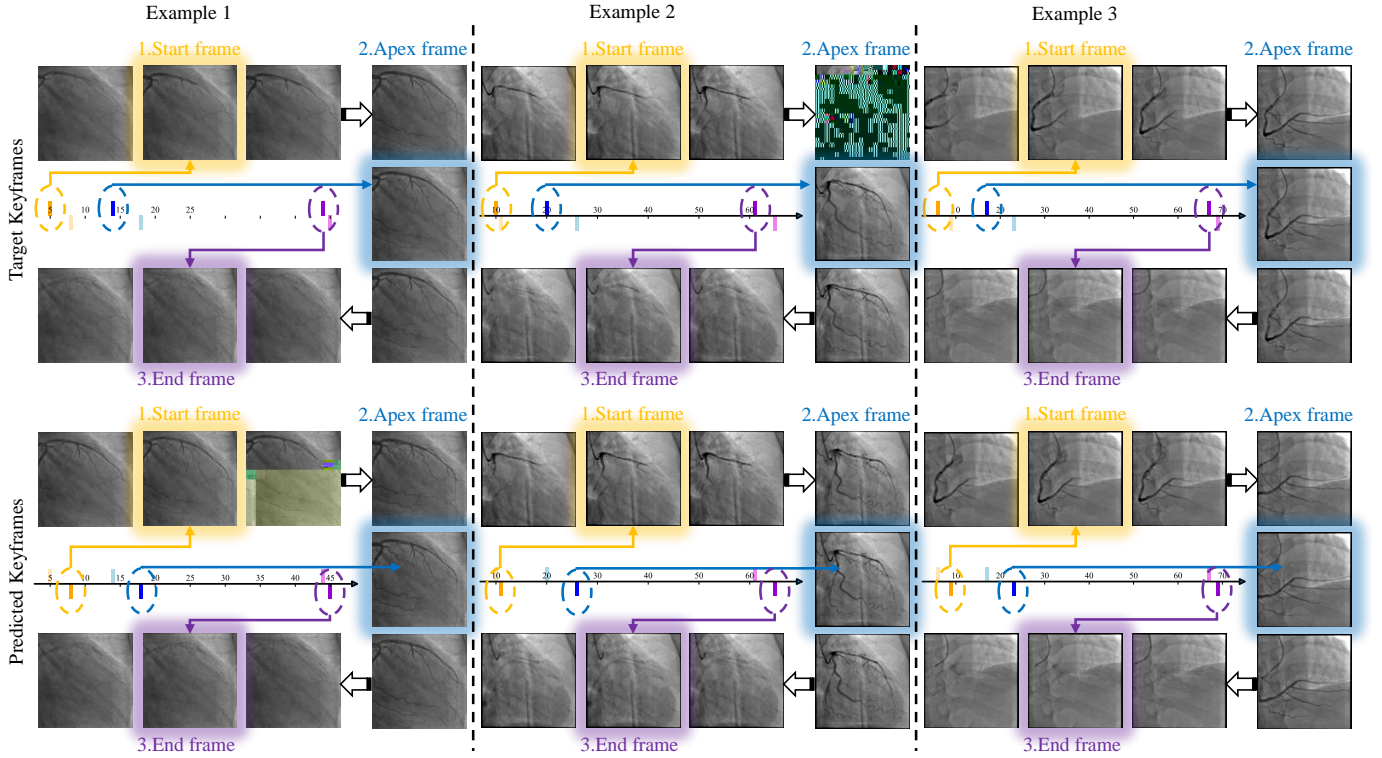


Fig. 4. Comparison of targets and keyframe localization results predicted by the proposed model. The first row shows the target frames, and the second row shows the predicted frames.

long-term spatiotemporal attention module uses convolutional kernels of size = 3 with stride = 1 and padding = 1 for the first two convolutional layers in spatial attention, and utilizes convolutional kernels of size = 1 with stride = 1 and padding = 0 for the last layer. The temporal attention uses a standard CLSTM with three hidden layers that have 8, 8 and 1 output dimensions, and the kernel size is set to 3. Moreover, the other setting follows [6]. In summary, the initial learning rate is $1e-4$, and a cosine learning rate decay is used. The batch size is 2, and a weight decay of $1e-4$ is used. The model is evaluated after 50 epochs of training. $AP@[0.3:0.1:0.7]$ is used to evaluate the mAP of our model. The code is implemented by PyTorch and trained on a NVIDIA GeForce RTX 3090.

D. Comparison Methods

To evaluate the performance of our algorithm, we select several SOTA Transformer-based TAL methods for compari-

son, including AFSD [7], TALLFormer [50], E2E-TAD [51] and Actionformer [6]. The parameters of these algorithms are trained with their default settings and our dataset. Due to the different data formats used by the open-source code, we

TABLE III
STATISTICAL ANALYSIS OF ONE-SAMPLE T-TEST.

Popmean	5.2		5.4		5.6		5.8		6.0		6.2		CI
	T-v	P-v	T-v	P-v	T-v	P-v	T-v	P-v	T-v	P-v	T-v	P-v	
Baseline [6]	0.65	0.74	0.16	0.56	-0.34	0.37	-0.84	0.20	-1.33	0.09	-1.83	0.03	0-6.13
Ours	-1.36	0.09	-1.92	0.03	-2.48	0.01	-3.04	0.00	-3.60	0.00	-4.16	0.00	0-5.30

TABLE IV
PERFORMANCE OF ABLATION STUDY ON SPATIOTEMPORAL ATTENTION MODULE.

Method	L-T		S-T	AP@0.3 "	AP@0.4 "	AP@0.5 "	AP@0.6 "	AP@0.7 "	mAP "	P "	R "	F "	AD #
	TA	SA											
Baseline w/o S-T				82.35	60.79	37.00	17.14	6.95	40.84	0.7189	0.7387	0.6893	11.14
TA w/o S-T	×			79.15	62.45	39.11	17.94	7.33	41.20	0.7336	0.7508	0.7016	10.35
SA w/o S-T		×		79.36	60.78	37.20	20.82	8.88	41.41	0.7491	0.7707	0.7167	9.83
STA w/o S-T	×	×		77.39	65.81	42.18	22.22	8.11	43.14	0.7218	0.7523	0.6936	10.54
Baseline [6]			×	90.85	85.25	70.45	52.56	32.62	66.35	0.8096	0.8356	0.8013	5.46
TA	×		×	90.48	87.43	70.17	50.74	37.35	67.23	0.7974	0.8279	0.7951	5.81
SA		×	×	<u>91.64</u>	90.76	<u>74.35</u>	<u>54.21</u>	35.62	<u>69.31</u>	<u>0.8136</u>	<u>0.8393</u>	<u>0.8092</u>	<u>5.30</u>

TABLE V
PERFORMANCE OF ABLATION STUDY ON IMAGE-TO-PATCH CONTRASTIVE MODULE.

Method	ICon	PCon	$AP@0.3$	$AP@0.4$	$AP@0.5$	$AP@0.6$	$AP@0.7$	mAP	P	R	F	$AD \#$
STA			92.60	89.42	75.23	57.94	35.86	70.21	0.8177	0.8422	0.8115	5.16
STA+ICon	×		94.27	91.41	75.16	<u>56.67</u>	35.26	70.56	0.8195	0.8509	0.8161	<u>5.07</u>
STA+PCon		×	<u>97.34</u>	<u>92.45</u>	<u>80.07</u>	<u>53.20</u>	36.40	<u>71.90</u>	<u>0.8294</u>	<u>0.8575</u>	<u>0.8258</u>	5.14
STA+IPCon	×	×	98.44	92.93	80.91	53.85	<u>36.10</u>	72.45	0.8342	0.8612	0.8296	4.71

TABLE VI
PERFORMANCE OF STUDY ON DIFFERENT FEATURE EXTRACTION STRATEGIES.

Method	Feature	$AP@0.3$	$AP@0.4$	$AP@0.5$	$AP@0.6$	$AP@0.7$	mAP	P	R	F	$AD \#$
Baseline	SVS-Net	90.85	85.25	70.45	52.56	32.62	66.35	0.8096	0.8356	0.8013	5.46
Baseline	I3D	88.79	77.02	62.61	48.05	26.27	60.55	0.6767	0.6867	0.6660	9.26
Ours (stride = 1)	SVS-Net	<u>96.90</u>	89.27	71.91	51.41	37.41	69.38	0.8216	0.8515	0.8167	5.13
Ours (stride = 2)	SVS-Net	96.63	<u>91.25</u>	<u>72.90</u>	54.78	35.94	<u>70.30</u>	<u>0.8222</u>	<u>0.8564</u>	<u>0.8195</u>	<u>5.10</u>
Ours (stride = 4)	SVS-Net	98.44	92.93	80.91	<u>53.85</u>	<u>36.10</u>	72.45	0.8342	0.8612	0.8296	4.71

TABLE VII
PERFORMANCE OF STUDY ON DIFFERENT RANDOM PATCHES.

Method	$AP@0.3$	$AP@0.4$	$AP@0.5$	$AP@0.6$	$AP@0.7$	mAP	P	R	F	$AD \#$
Ours	98.44	92.93	<u>80.91</u>	53.85	36.10	72.45	0.8342	0.8612	0.8296	4.71
Ours (other patch 1)	98.44	92.93	80.64	<u>53.71</u>	35.10	72.16	<u>0.8349</u>	<u>0.8619</u>	<u>0.8303</u>	<u>4.71</u>
Ours (other patch 2)	98.44	<u>91.67</u>	81.95	53.25	<u>35.97</u>	<u>72.25</u>	0.8352	0.8629	0.8313	4.70

achieved the best performances in terms of most metrics. In particular, it achieved the highest values in terms of P , R and F metrics and the lowest 4.71 in AD , which means that the proposed method has a 4.71-frame distance between the prediction and the target on average, being shorter than average 5.46-frame distance of the most advanced method [6]. Note that due to the existence of boundary-agnostics in XCA sequence, it is more difficult to optimize this metric with a smaller standard deviation.

H. Experiments with Different Feature Extraction Strategies

SVS-Net [15] is used for feature extraction in this work instead of I3D [73] that is used in Actionformer [6]. To prove the rationality, we used these two methods to extract features and conducted experiments. The results are shown in TABLE VI. Baseline model is selected for the experiment because the I3D method will destroy the spatial dimensions of features. The results show that SVS-Net achieves better feature extraction performance than I3D in our scene.

We use SVS-Net [15] with stride = 4 for feature extraction in this paper, which means that there are not overlapping between neighborhood features and the length of sequence will decrease. To investigate the influence of feature stride, we decrease the feature stride and make some overlapping between neighborhood features. The results in TABLE VI show that overlapping strategy does not perform better because it can lead to a large amount of redundant calculations and

overlapping interference between neighborhood features. In addition, the amount of training time has been further extended when the smaller strides are used.

I. Experiments with Different Random Patches

A low-rank background patch is selected randomly as a convolutional kernel in patchwise contrastive module. We only randomly select the convolutional kernel in the frame of $t = 0$ and use this kernel to all frames. Therefore, it solely contains background features due to no use of contrast agents in this stage, so different selections do not influence performance significantly. To show this issue, we have added an experiment which is reported in TABLE VII. Specifically, for the same trained model, we select different patches when $t = 0$ as the convolution kernel and test the performance. The results show that different selections hardly influence the results.

J. Experiments with Different Problem Settings

We define the TAL as the filling/disappearing localization. To prove the rationality of this setting, it is also compared with other two problem settings for locating whole stage with filling and disappearing stages, respectively. The first setting called setting-1 is to localize the filling and whole actions, while

TABLE VIII
PERFORMANCE OF STUDY ON DIFFERENT PROBLEM SETTINGS.

Method	MBR	P "	R "	F "	AD #
Ours	0.8342	0.8612	0.8296	4.71	
Ours (setting-1)	0.8104	0.8419	0.8061	5.59	
Ours (setting-2)	0.5808	0.5551	0.5563	10.31	
Ours (setting-1) ×	0.8051	0.8391	0.8027	6.15	
Ours (setting-2) ×	<u>0.8232</u>	<u>0.8402</u>	<u>0.8151</u>	<u>5.19</u>	

setting-2 is to localize the disappearing and whole actions. The postprocessing of setting-1 can be calculated as:

$$\begin{aligned}
 StartFrame &= (start_1 + start_2)/2 \\
 ApexFrame &= end_1 \\
 EndFrame &= end_2
 \end{aligned} \tag{19}$$

The postprocessing of setting-2 can be calculated as:

$$\begin{aligned}
 StartFrame &= start_1 \\
 ApexFrame &= start_2 \\
 EndFrame &= (end_1 + end_2)/2
 \end{aligned} \tag{20}$$

We conducted experiments under these two settings, as shown in TABLE VIII. The results show that our original setting is optimal. The worse results of setting-1 and setting-2 could be related to the problem of overlapping between the whole stage and the filling/disappearing stage. We find that handling overlapping actions is not good enough for Actionformer based method, because Actionformer based method is to regress boundary with

indicate that our method can learn the vessel structures clearly.

V. CONCLUSIONS

We proposed a novel long short-term spatiotemporal attention network with image-to-patch contrastive modules for locating keyframes in the challenging XCA sequence. An XCA sequence dataset was collected. SOTA experiments and ablation experiments have proved the strong outperformance of the proposed method over SOTA methods. The proposed method can be applied to any flow-like scenarios in monitoring spatiotemporal networks. For example, [74] built a model to estimate the crowd traffic in public places. Overcrowding and stampedes may occur in public places with the gathering of crowds. The action of crowd-gathering can be monitored by the strategies of our method to mitigate and prevent risk without estimating the crowd traffic directly. Another example is traffic inflow and outflow prediction as [75] did. The proposed method has potential applications to locate the moments when inflows and outflows significantly increase or decrease to reduce traffic congestion and accidents.

ACKNOWLEDGEMENTS

The authors thank all cited authors for providing the source code used in this work and the anonymous reviewers for their valuable comments on the manuscript.

REFERENCES

- [1] S. Ding, J. Pu, Z.-q. Qiao, P. Shan, W. Song, Y. Du, J.-Y. Shen, S.-x. Jin, Y. Sun, L. Shen, Y.-l. Lim, and B. He, "TIMI myocardial perfusion frame count: A new method to assess myocardial perfusion and its predictive value for short-term prognosis," *Catheterization and Cardiovascular Interventions*, vol. 75, no. 5, pp. 722–732, 2010.
- [2] H. Ge, S. Ding, D. aolei An, Z. Li, H. Ding, F. Yang, L.-C. Kong, J. Xu, J. Pu, and B. He, "Frame counting improves the assessment of post-reperfusion microvascular patency by TIMI myocardial perfusion grade: Evidence from cardiac magnetic resonance imaging.," *International Journal of Cardiology*, vol. 203, pp. 360–366, 2016.
- [3] B. Qin, H. Mao, Y. Liu, J. Zhao, Y. Lv, Y. Zhu, S. Ding, and X. Chen, "Robust PCA unrolling network for super-resolution vessel extraction in X-ray coronary angiography," *IEEE Transactions on Medical Imaging*, vol. 41, no. 11, pp. 3087–3098, 2022.
- [4] B. Qin, M. Jin, and S. Ding, "Extracting heterogeneous vessels in X-ray coronary angiography via machine learning," in *Cardiovascular and Coronary Artery Imaging*, pp. 89–127, Elsevier, 2022.
- [5] E. Vahdani and Y. Tian, "Deep learning-based action detection in untrimmed videos: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4302–4320, 2023.
- [6] C.-L. Zhang, J. Wu, and Y. Li, "Actionformer: Localizing moments of actions with transformers," in *Computer Vision – ECCV 2022* (S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, eds.), (Cham), pp. 492–510, Springer Nature Switzerland, 2022.
- [7] 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 *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3319–3328, 2021.
- [8] W. Dong, Z. Zhang, C. Song, and T. Tan, "Identifying the key frames: An attention-aware sampling method for action recognition," *Pattern Recognition*, vol. 130, p. 108797, 2022.
- [9] R. Huang, Q. Ying, Z. Lin, Z. Zheng, L. Tan, G. Tang, Q. Zhang, M. Luo, X. Yi, P. Liu, W. Pan, J. Wu, B. Luo, and D. Ni, "Extracting keyframes of breast ultrasound video using deep reinforcement learning," *Medical Image Analysis*, vol. 80, p. 102490, 2022.
- [10] C. Sun, H. Song, X. Wu, Y. Jia, and J. Luo, "Exploiting informative video segments for temporal action localization," *IEEE Transactions on Multimedia*, vol. 24, pp. 274–287, 2022.
- [11] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, "A survey on vision transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 87–110, 2023.
- [12] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, vol. 8, no. 3, pp. 331–368, 2022.
- [13] W. Zhao, Q. Liu, Y. Lv, and B. Qin, "Texture variation adaptive image denoising with nonlocal PCA," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5537–5551, 2019.
- [14] P. Irrera, I. Bloch, and M. Delplanque, "A flexible patch based approach for combined denoising and contrast enhancement of digital X-ray images," *Medical Image Analysis*, vol. 28, pp. 33–45, 2016.
- [15] D. Hao, S. Ding, L. Qiu, Y. Lv, B. Fei, Y. Zhu, and B. Qin, "Sequential vessel segmentation via deep channel attention network," *Neural Networks*, vol. 128, pp. 172–187, 2020.
- [16] X. Zhu, Z. Cheng, S. Wang, X. Chen, and G. Lu, "Coronary angiography image segmentation based on pspnet," *Computer Methods and Programs in Biomedicine*, vol. 200, p. 105897, 2021.
- [17] T. J. Jun, J. Kweon, Y.-H. Kim, and D. Kim, "T-net: Nested encoder-decoder architecture for the main vessel segmentation in coronary angiography," *Neural Networks*, vol. 128, pp. 216–233, 2020.
- [18] B. Qin, M. Jin, D. Hao, Y. Lv, Q. Liu, Y. Zhu, S. Ding, J. Zhao, and B. Fei, "Accurate vessel extraction via tensor completion of background layer in X-ray coronary angiograms," *Pattern Recognition*, vol. 87, pp. 38–54, 2019.
- [19] M. Jin, R. Li, J. Jiang, and B. Qin, "Extracting contrast-filled vessels in X-ray angiography by graduated RPCA with motion coherency constraint," *Pattern Recognition*, vol. 63, pp. 653–666, 2017.
- [20] H. Ma, A. Hoogendoorn, E. Regar, W. J. Niessen, and T. van Walsum, "Automatic online layer separation for vessel enhancement in X-ray angiograms for percutaneous coronary interventions," *Medical Image Analysis*, vol. 39, pp. 145–161, 2017.
- [21] M. Jin, D. Hao, S. Ding, and B. Qin, "Low-rank and sparse decomposition with spatially adaptive filtering for sequential segmentation of 2d+t vessels," *Physics in Medicine & Biology*, vol. 63, no. 17, p. 17LT01, 2018.
- [22] O. Solomon, R. Cohen, Y. Zhang, Y. Yang, Q. He, J. Luo, R. J. van Sloun, and Y. C. Eldar, "Deep unfolded robust PCA with application to clutter suppression in ultrasound," *IEEE Transactions on Medical Imaging*, vol. 39, no. 4, pp. 1051–1063, 2019.
- [23] X. Li, B. Zhao, and X. Lu, "Key frame extraction in the summary space," *IEEE Transactions on Cybernetics*, vol. 48, no. 6, pp. 1923–1934, 2018.
- [24] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, "Video summarization using deep neural networks: A survey," *Proceedings of the IEEE*, vol. 109, no. 11, pp. 1838–1863, 2021.
- [25] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 296–305, 2005.
- [26] C. Choudary and T. Liu, "Summarization of visual content in instructional videos," *IEEE Transactions on Multimedia*, vol. 9, no. 7, pp. 1443–1455, 2007.
- [27] G. Guan, Z. Wang, S. Lu, J. D. Deng, and D. D. Feng, "Keypoint-based keyframe selection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 729–734, 2013.
- [28] Y. Zhang, R. Tao, and Y. Wang, "Motion-state-adaptive video summarization via spatiotemporal analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 6, pp. 1340–1352, 2017.
- [29] T. Liu, Q. Meng, J.-J. Huang, A. Vlontzos, D. Rueckert, and B. Kainz, "Video summarization through reinforcement learning with a 3d spatiotemporal u-net," *IEEE Transactions on Image Processing*, vol. 31, pp. 1573–1586, 2022.
- [30] C. Dang and H. Radha, "RPCA-KFE: Key frame extraction for video using robust principal component analysis," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3742–3753, 2015.
- [31] M. Ma, S. Mei, S. Wan, Z. Wang, X.-S. Hua, and D. D. Feng, "Graph convolutional dictionary selection with $L_{2,p}$ norm for video summarization," *IEEE Transactions on Image Processing*, vol. 31, pp. 1789–1804, 2022.
- [32] X. Gu, L. Lu, S. Qiu, Q. Zou, and Z. Yang, "Sentiment key frame extraction in user-generated micro-videos via low-rank and sparse representation," *Neurocomputing*, vol. 410, pp. 441–453, 2020.
- [33] S. Mei, M. Ma, S. Wan, J. Hou, Z. Wang, and D. D. Feng, "Patch based video summarization with block sparse representation," *IEEE Transactions on Multimedia*, vol. 23, pp. 732–747, 2021.

- [34] Z. Ji, K. Xiong, Y. Pang, and X. Li, "Video summarization with attention-based encoder-decoder networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 6, pp. 1709–1717, 2020.
- [35] R. Zhong, R. Wang, Y. Zou, Z. Hong, and M. Hu, "Graph attention networks adjusted bi-lstm for video summarization," *IEEE Signal Processing Letters*, vol. 28, pp. 663–667, 2021.
- [36] B. Zhao, X. Li, and X. Lu, "Hierarchical recurrent neural network for video summarization," in *Proceedings of the 25th ACM International Conference on Multimedia, MM '17*, (New York, NY, USA), p. 863–871, Association for Computing Machinery, 2017.
- [37] Y. Yuan, H. Li, and Q. Wang, "Spatiotemporal modeling for video summarization using convolutional recurrent neural network," *IEEE Access*, vol. 7, pp. 64676–64685, 2019.
- [38] M. Rochan, L. Ye, and Y. Wang, "Video summarization using fully convolutional sequence networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- [39] B. Mahasseni, M. Lam, and S. Todorovic, "Unsupervised video summarization with adversarial lstm networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2982–2991, 2017.
- [40] G. Liang, Y. Lv, S. Li, S. Zhang, and Y. Zhang, "Video summarization with a convolutional attentive adversarial network," *Pattern Recognition*, vol. 131, p. 108840, 2022.
- [41] H. Xia and Y. Zhan, "A survey on temporal action localization," *IEEE Access*, vol. 8, pp. 70477–70487, 2020.
- [42] X. Chen, C. Gao, C. Li, Y. Yang, and D. Meng, "Infrared action detection in the dark via cross-stream attention mechanism," *IEEE Transactions on Multimedia*, vol. 24, pp. 288–300, 2022.
- [43] P. Zhao, L. Xie, Y. Zhang, and Q. Tian, "Actionness-guided transformer for anchor-free temporal action localization," *IEEE Signal Processing Letters*, vol. 29, pp. 194–198, 2022.
- [44] H. Xu, A. Das, and K. Saenko, "Two-stream region convolutional 3d network for temporal activity detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 10, pp. 2319–2332, 2019.
- [45] M.-G. Gan and Y. Zhang, "Temporal attention-pyramid pooling for temporal action detection," *IEEE Transactions on Multimedia*, pp. 1–1, 2022.
- [46] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan, "Graph convolutional module for temporal action localization in videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6209–6223, 2022.
- [47] K. Xia, L. Wang, S. Zhou, G. Hua, and W. Tang, "Dual relation network for temporal action localization," *Pattern Recognition*, vol. 129, p. 108725, 2022.
- [48] R. Girdhar, J. João Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 244–253, 2019.
- [49] Q. Wang, Y. Zhang, Y. Zheng, and P. Pan, "Rcl: Recurrent continuous localization for temporal action detection," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13556–13565, 2022.
- [50] F. Cheng and G. Bertasius, "TALLFormer: Temporal action localization with long-memory transformer," in *Computer Vision—ECCV 2022: 17th European Conference*, vol. abs/2204.01680, 2022.
- [51] X. Liu, S. Bai, and X. Bai, "An empirical study of end-to-end temporal action detection," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19978–19987, 2022.
- [52] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- [53] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," in *NeurIPS*, 2021.
- [54] C.-Y. Wu, Y. Li, K. Mangalam, H. Fan, B. Xiong, J. Malik, and C. Feichtenhofer, "MeMViT: Memory-augmented multiscale vision transformer for efficient long-term video recognition," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13577–13587, 2022.
- [55] 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," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9992–10002, 2021.
- [56] L. Yang, J. Han, T. Zhao, N. Liu, and D. Zhang, "Structured attention composition for temporal action localization," *IEEE Transactions on Image Processing*, pp. 1–1, 2023.
- [57] H. Alwassel, S. Giancola, and B. Ghanem, "TSP: Temporally-sensitive pretraining of video encoders for localization tasks," in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 3166–3176, 2021.
- [58] C. Zhang, T. Yang, J. Weng, M. Cao, J. Wang, and Y. Zou, "Unsupervised pre-training for temporal action localization tasks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14031–14041, June 2022.
- [59] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 4037–4058, 2021.
- [60] X. Liu, Z. Zhang, L. Lyu, Z. Zhang, S. Xiao, C. Shen, and P. Yu, "Traffic anomaly prediction based on joint static-dynamic spatio-temporal evolutionary learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 5356–5370, 2023.
- [61] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W. Chun Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *NIPS*, 2015.
- [62] C. Gao, H. Ye, F. Cao, C. Wen, Q. Zhang, and F. Zhang, "Multiscale fused network with additive channel-spatial attention for image segmentation," *Knowledge-Based Systems*, vol. 214, p. 106754, 2021.
- [63] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-Unet: Unet-like pure transformer for medical image segmentation," in *ECCV Workshops*, 2021.
- [64] J. Xie, J. Xiang, J. Chen, X. Hou, X. Zhao, and L. Shen, "C2 am: Contrastive learning of class-agnostic activation map for weakly supervised object localization and semantic segmentation," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 979–988, 2022.
- [65] C. Jung, G. Kwon, and J.-C. Ye, "Exploring patch-wise semantic relation for contrastive learning in image-to-image translation tasks," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18239–18248, 2022.
- [66] T. Yang, Z.-Y. Huang, J. Cao, L. Li, and X. Li, "Deepfake network architecture attribution,"
- [67] Y. Xu, B. Du, F. Zhang, and L. pei Zhang, "Hyperspectral image classification via a random patches network," *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018.
- [68] W. Huang, Y. Huang, Z. Wu, J. Yin, and Q. Chen, "A multi-kernel mode using a local binary pattern and random patch convolution for hyperspectral image classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 4607–4620, 2021.
- [69] T.-Y. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, pp. 318–327, 2020.
- [70] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-iou loss: Faster and better learning for bounding box regression," in *AAAI Conference on Artificial Intelligence*, 2019.
- [71] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-nms — improving object detection with one line of code," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5562–5570, 2017.
- [72] F. C. Heilbron, V. Escorcia, B. Ghanem, and J. C. Niebles, "ActivityNet: A large-scale video benchmark for human activity understanding," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 961–970, 2015.
- [73] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733, 2017.
- [74] T. Zhang, Y. Zhao, W. Jia, and M. Chen, "Collaborative algorithms that combine ai with iot towards monitoring and control system," *Future Generation Computer Systems*, vol. 125, pp. 677–686, 2021.
- [75] X. Huang, B. Zhang, S. Feng, Y. Ye, and X. Li, "Interpretable local flow attention for multi-step traffic flow prediction," *Neural Networks*, vol. 161, pp. 25–38, 2023.