A Multi-Modal Transformer Network for Action Detection

Matthew Korban[a], Peter Youngs[b], Scott T. Acton[a,*]

[a]*Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904*
[b]*Department of Curriculum, Instruction and Special Education, University of Virginia, VA 22904*

Abstract

This paper proposes a multi-modal transformer network for detecting actions in untrimmed videos. To enrich the action features, our transformer network utilizes a novel multi-modal attention mechanism that captures the correlations between different combinations of spatial and motion modalities. Exploring such correlations for actions effectively has not been explored before. We also suggest an algorithm to correct the motion distortion caused by camera movements. Such motion distortion severely reduces the expressive power of motion features represented by optical flow vectors. We also introduce a new instructional activity dataset that includes classroom videos from K-12 schools. We conduct comprehensive experiments to evaluate the performance of different approaches on our dataset. Our proposed algorithm outperforms the state-of-the-art methods on two public benchmarks, THUMOS14 and ActivityNet, and our instructional activity dataset.

Keywords: Action Detection, Transformer Network, Optical Flow, Motion Features

1. Introduction

Action Detection is temporally localizing action class instances, commonly in continuous-streaming videos. Action sequences are represented as two spatial and temporal components, which jointly can define the meaning of various actions. For example, the action "throwing a ball" is characterized by its spatial components, the image pixels of the ball, and its movement during the action sequence. A popular way to represent such spatial and temporal components of actions are *RGB* images and *optical flows*, respectively [1]. However, such spatial-temporal action detection using the RGB and optical flow modalities is challenging.

*Corresponding author
Email addresses: acw6ze@virginia.edu (Matthew Korban), pay2n@virginia.edu (Peter Youngs), acton@virginia.edu (Scott T. Acton)

The two main challenges are the separated RGB and optical flow modalities and camera movement [2]. We will discuss the challenges above and our solutions to handle them as follows:

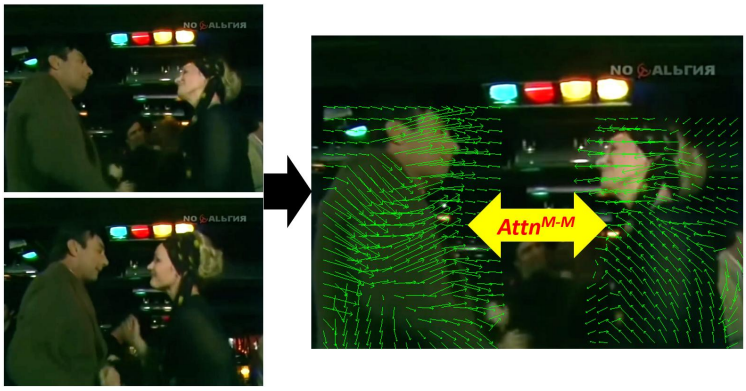
1.1. Multi-modal attention for solving separated RGB and optical flow

The optical flow is a powerful modality to model motions in action videos. As the optical flow is extracted independently, the current action detection algorithms exploit RGB and optical flow images separately [3]. Many actions however are defined by the correlative patterns among spatial (RGB) and motion (optical flow) pixels or among each modality themselves. Some examples are shown in Fig. 1. The first scenario is when both subjects/objects of interest move. An example of such an action "dancing" is shown in Fig. 1 - top, where two persons move toward each other. We compute such correlations using our Motion-Motion Attention ($Attn^{M-M}$) within our transformer network using a novel multi-modal attention mechanism. Next is when we have both moving and stationary subjects/objects in the scene that jointly define the action such as "kicking a ball" (Fig. 1 - middle). Here, our Spatial-Motion Attention ($Attn^{S-M}$) calculates the correlations between the spatial stationary features (for the person) and temporal motion features (for the ball) within our transformer network. And finally, our Spatial-Spatial Attention ($Attn^{S-S}$) computes the correlative patterns in the scene when all the objects/subjects are stationary such as persons and violins in the action "playing violin" shown in Fig. 1 - bottom.

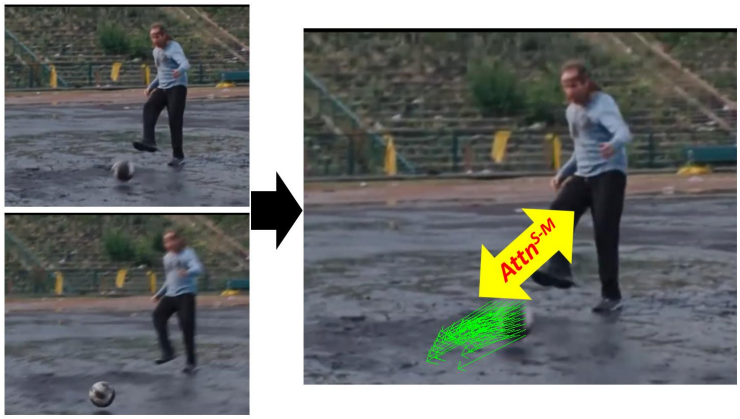
There have been some methods that evaluated several aspects of the optical flow in action recognition and detection. [4] suggests that some features such as velocity, gradient, and divergence represented in the optical flow are effective in action recognition. [5] investigates the correlations between optical flow and action recognition accuracy based on several well-known optical flow estimation methods. [6] proposes a deep network that extracts effective optical flow for action recognition. However, there has not been any approach to effectively capture the correlations between optical flow and RGB images. Such correlations, as stated above, are important in action modeling. So, we propose an effective strategy to capture the correlations between optical flow and RGB images.

1.2. Motion distortion correction for solving camera movement

Continuous-streaming action videos are often captured in the wild, where camera movement is common. Such camera movement can significantly distort the motion represented



Motion-Motion Attention (dancing)



Spatial-Motion Attention (kicking a ball)

Spatial-Spatial Attention (playing violin)

Figure 1: Our multi-modal attention mechanism covers a variety of actions. Top, Motion-Motion Attention ($Attn^{M-M}$): in this case, the goal is to find the correlations between both moving subjects/objects of interest such as the action "dancing" where the two persons get toward each other. Motion vectors (green vectors) are shown in the right large image based on two consecutive left (small) frames. Middle, Spatial-Motion Attention ($Attn^{S-M}$): this scenario aims to find the correlations between moving subjects/objects, such as a ball in this example, and stationary ones (the person here). In this example, the green motion vectors are only illustrated on the ball. The motion vectors for the ball are shown in the right (large) image. Bottom, Spatial-Spatial Attention ($Attn^{S-S}$): In this case, all the subjects/objects (persons and violins) are almost stationary such as "playing violin".

by the optical flow as it causes spatial-temporal inconsistency. An example is shown in Fig. 2 where the person is running toward the southwest, and getting closer to the orange street line in two consecutive frames (Fig. 2 (a) and (b)) and the car is stationary. A standard state-of-the-art optical flow algorithm [ref] fails to extract the correct optical flow motion vectors (Fig. 2 (c)) because of spatial-temporal inconsistency caused by the camera movements. In this example, the person's movement (orange arrows) is inconsistent with respect to his spatial location in the image (yellow arrows). To solve this issue, we propose a motion distortion correction algorithm whose corrected results for the moving person and the stationary car are shown in Fig. 2 (d).

There have been some approaches that tried to include the camera movement factor in optical flow extraction. [2] estimates the camera pose jointly with optical flow and depth maps using a complex network architecture. Similarly, [7] proposes that a collaboration between camera pose, optical flow, and depth map estimation is useful. However, these methods have some issues: (1) such methods require a complicated deep network design and training; (2) they need to have the ground truths for objects' poses which are challenging to obtain in real-world scenarios; (3) these approaches have not been validated in real-world applications such as action detection to show the practical reliability of the extracted optical flows. We, however, propose a simple yet effective approach that does not require any ground truths to improve the optical flow extraction. Furthermore, we validated our improved optical flows in real-world scenarios (action detection) on several public benchmarks.

A new instructional activity dataset. We created a new dataset of instructional activities gathered from K-12 schools. We used a trained and professional team of annotators to label 24 instructional activity classes in our collected videos. We annotate every second (30 frames) of the video with multiple class labels. Some frame examples from our annotated videos and computed optical flow are shown in Fig. 3. We will give more details about our dataset in Section 4.1.2.

The main **contributions** of this paper can be summarized as follows:

- We propose a novel transformer network for action detection.
- We suggest a new multi-modal attention mechanism to effectively capture spatial-temporal features from RGB and optical flow modalities.
- We introduce a novel motion distortion correction algorithm to handle camera move-

Figure 2: Illustration of the results of our motion distortion correction algorithm: (a) and (b) show two consecutive sampled frames of an action sequence [ref]. In this sequence, the person is running toward the southwest. Due to the spatial-temporal inconsistency caused by camera movements, the actual motion is distorted. Specifically, in reality, the person is getting closer to a reference orange street lines (shown as orange vectors) which is inconsistent with the person's spatial location with respect to the image origin (yellow vectors). (c) show the optical flow motion vectors corresponding to the person and the parked car obtained from a standard state-of-the-art algorithm [ref]. Due to the motion distortion caused by camera movements, the motion vectors for the person and stationary parked car are incorrect. (d) shows our corrected optical flow motion vectors for the person and the stationary car.

Figure 3: Some example frames of our annotated instructional activity dataset showing two consecutive sampled frames of t and frame $t+\tau$, and consequently, the computed optical flows.

ments in action videos.

- We collected a new large dataset including instructional activities from K-12 schools.

2. Related works

Two main related topics to our work are "Action Recognition"and "Temporal Action Detection". Action recognition aims to classify trimmed, and often short action sequences [8, 9]. In contrast, action detection intents to identify action instances in untrimmed, and usually long videos [10, 11]. While action recognition is a single-label classification problem, in action detection we can have multi-class labels for each action sequence [3, 8]. In this literature review, we mainly focus on the action recognition and detection works that are similar to ours in using both RGB images and optical flow. We also investigate the previous works that are based on the transformer network which is a state-of-the-art deep architecture that utilizes an attention mechanism to capture the correlations among its selective inputs, so-called tokens [12, 13].

2.1. Action recognition

[14] is one of the first deep architectures that use both RGB and optical flow images and suggested that combining optical flow and RGB images boosts the action recognition performance. [15] improved such as a two-stream network by redesigning its architecture such as the mechanisms for feature fusion and pooling.

Later, some researchers proposed strategies to improve the utilization of optical flow along with RGB images. For example, [16] exploited RGB and optical flow images to create more effective fine-grained action descriptors which are located in informative small regions in video frames. As another example, [17] proposed a multi-stream Convolutional Neural Network that extracts effective motion and spatial features which are concluded to be more centralized on the human body. [18] concluded that combining trajectory descriptors and optical flow can improve the action feature representation.

Some other works have been conducted to resolve some general issues in action recognition using two modalities. [19] introduced a 3-D-convNet Fusion to deal with varying spatial and temporal sizes of RGB and optical flow frames. [20] proposed a two-stream convolutional network that takes advantage of spatial and motion modalities while improving the efficiency of motion feature extraction. [21] suggested a method to transfer the knowledge obtained from the dataset with a large volume of RGB and optical flow action frames to smaller-scale real-world scenarios such as manufacturing.

2.2. Temporal action detection

To enhance the way the optical flow and RGB frames are used for temporal action detection the researchers in the field suggested various strategies. For example, [22] suggested that focusing on local regions in RGB and optical flow images (by using a motion region network) and stacking optical flow improves the modeling of actions. As another example, [23] proposed to use of multiple object tracking and person detection to capture better action proposals from the RGB and optical flow images. [24] suggested using appearance and motion detectors to improve the temporal cuboid representation around subjects in the two spatial and temporal modalities. [25] performed the tasks of spatial-temporal localization and action classification using a cross-stream cooperation strategy, where the RGB and optical flow streams jointly improve these tasks. [26] proposed reducing the number of optical flow and RGB frames needed for creating effective spatial-temporal action features utilizing long-term 3D CNNs. [27] suggested a two-stream network to distinguish between the actions and background in both RGB and optical flow frames which are weakly annotated. [28] proposed a convolution autoencoder to extract spatial and temporal features and effectively simulate the optical flow information by using consecutive frames.

2.3. Transformer network and attention

There have been some approaches that suggested using an attention mechanism or a transformer network for action recognition and detection. The attention mechanism has been proposed to handle some existing problems. For example, [29] suggested a 3D CNN with an attention agent to remove the redundant temporal information. [30] introduced an efficient action transformer network that combines the power of attention and recurrent mechanisms to shorten the temporal window required for action recognition.

The attention mechanism also has been used to improve the modeling of actions. [31] suggested a self-attention module to capture the interactions between different spatial-temporal feature maps. [32] proposed a video transformer network that utilizes a temporal attention module to improve the spatial feature representation. [33] suggested a two-stream network using LSTM and an attention module that focuses on selective effective spatial-temporal input features. [34] proposed a Markov decision process to train an attention mechanism that captures keyframes in action videos effectively.

Following the literature, our temporal action detection method also utilizes both RGB and optical flow modalities to effectively incorporate spatial and motion information. To improve the expressive power of spatial-temporal action features using RGB and optical flow images, we propose a transformer network with a multi-modal attention mechanism and enhanced motion features.

3. Methodology

3.1. Overview and terminology

Fig. 4 indicates the overview of our proposed method. Given a set of RGB frame sequence, $I^S = \{I_i^s, i = 0, 1, ..., T\}$, and corresponding optical flows $I^{M'} = \{I_i^{m'}, i = 0, 1, ..., T\}$, respectively, the goal is to find the action class scores, $\hat{Y}^C$. Here, T is the length of the temporal sequence. To do such, we first fix the motion distortion of the optical modality using our *Motion Distortion Correction* algorithm. We first embedded the features using [35], Our multi-modal transformer the corrected optical flows I^M and I^S to compute multi-modal attentions including motion-motion, spatial-motion, motion-spatial, and spatial-spatial attentions, $Attn^{M-M}$, $Attn^{S-M}$, $Attn^{M-S}$, and $Attn^{S-S}$, respectively. Then in the classification stage, after computing action class label scores for each frame $\hat{Y}^C = \{y_i, i = 0, 1, ..., T\}$, we calculate the class labels c^s for the sequence.

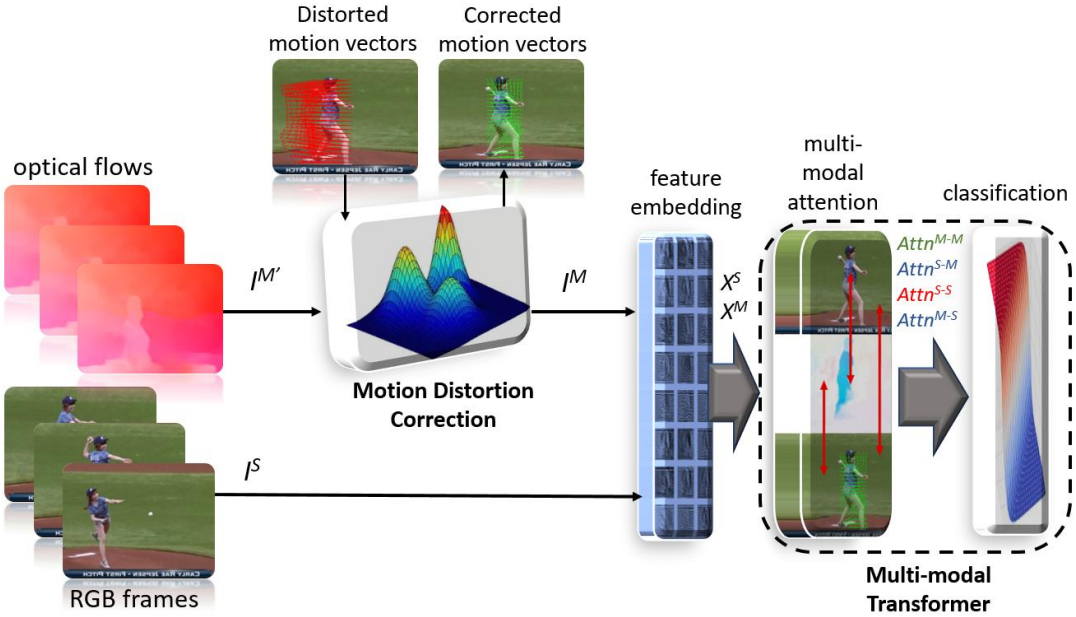


Figure 4: The main pipeline of our proposed action detection algorithm using both RGB frames (I^S) and optical flows ($I^{M'}$). Our motion distortion correction algorithm fixes the distorted optical flows (caused by camera movements) and produces the corrected optical flows (I^M). The feature embedding network [35] extracts spatial and motion features, X^S, and X^M, respectively. Our multi-modal transformer network uses both spatial and corrected motion modalities to detect the action classes in videos. Using our multi-modal attention mechanism we calculate motion-motion, spatial-motion, motion-spatial, and spatial-spatial attentions, $Attn^{M-M}$, $Attn^{S-M}$, $Attn^{M-S}$, and $Attn^{S-S}$, respectively. Finally, we identify the action sequence in the classification stage.

3.2. Motion distortion correction algorithm

In action videos captured in the wild such as that for popular action datasets (THUMOS14 [36] and ActivityNet [37]), camera movements happen often. Such camera movements significantly distort the motion information depicted in the optical flow which is a powerful and popular modality to represent actions. We previously showed some examples of motion distortion caused by camera movements in Fig. 2. To solve such distortion caused by camera movements and to use the optical flow in our multi-modal transformer effectively, we propose a motion distortion correction algorithm.

Given the distorted optical flow image, $I^{M'}(x, y)$ and the corresponding motion vectors, $I^{V'}(u', v')$, the goal of our motion distortion correction algorithm is to effectively define a function ψ so that $\psi : I^{V'}(u', v') \rightarrow I^{V}(u, v)$, where $I^{V}(u, v)$ is the corrected motion vectors. Here, x and y are image pixels and u' and v' are distorted movement displacements between the image pixels in the time t, as $(x^{(t)}, y^{(t)})$, and the time $t + \tau$, as $(x^{(t+\tau)}, y^{(t+\tau)})$. u, and v are the corrected movement displacements.

Our motion distortion correction algorithm includes three main steps: motion segmentation, background motion modeling, and motion restoration which are explained as follows:

Motion segmentation. Assuming the distorted motion as $I^{V'}(u', v')$, we first segment it to the background motion vectors, $I_B^{V'}(u', v')$, and foreground motion vectors, $I_F^{V'}(u', v')$, using an person detection algorithm [38]. We assume that the most important moving subjects are the persons in the scene as actions are often defined based on persons' motions. Persons also often have dominant movements and consequently can be considered as the foreground in action videos.

Background motion modeling. While the foreground's motion is the consequence of both the camera and local movements, the often static background is mainly affected by camera movements. So, modeling the background's motion is an effective way to interpret the camera movements. We use the Gaussian Mixture Models (GMM) for modeling motion displacement vectors of the background. It is because even the most complex camera movement is the result of several sub-random movements that affect various sub-regions in the images. Such sub-random movements can be effectively modeled by the GMM.

For the distorted background motion vectors, assuming $I_B^{V'}(s') = \{s'_n, n \in 0, ..., H\}$, where $s' = (u', v')$, and $H = h \times w$ is the image size (with the height and length of h and w, respectively), the GMM with M distributions can be formulated as:

$$P(s') = \sum_{m=1}^{M} \pi_m N(s'|\mu_m, \Sigma_m), \tag{1}$$

In the above, $N(s'|\mu_k, \Sigma_m)$ is a sub-Gaussian density with the mean of μ_m and the covariance of Σ_m, weighted with the mixing coefficient of π_m. We model the distorted background motion vectors $I_B^{V'}(s') = \{s'_n, n \in 0, ..., H\}$ using the maximum likelihood estimation of the GMM [39]. The algorithm is summarized as follows:

1. Initializing μ_m, Σ_m, and π_m

2. Computing the posterior probability:

$$P(z_{nm}) = \frac{\pi_m N(s'_n|\mu_m, \Sigma_m)}{\sum_{i=1}^{M} \pi_i N(s'_n|\mu_m, \Sigma_m)}, \tag{2}$$

3. Re-estimating the model parameters:

$$\hat{\mu}_m = \frac{1}{\lambda_k} \sum_{n=1}^{H} P(z_{nm}) s'_n, \qquad \hat{\Sigma}_m = \frac{1}{\lambda_k} \sum_{n=1}^{H} P(z_{nm})(s'_n - \hat{\mu}_m)(s'_n - \hat{\mu}_m)^T, \qquad \hat{\pi_m} = \frac{\lambda_k}{H}, \tag{3}$$

   where, $\lambda_m = \sum_{n=1}^{H} P(z_{nm})$

4. Obtaining the log-likelihood:

$$lnP(I_B^{V'}(s')|\mu, \Sigma, \pi) = \sum_{n=1}^{H} ln(\sum_{m=1}^{M} \pi_m N(s'_n|\mu_m, \Sigma_m)) \tag{4}$$

5. Repeating (2) till reaching convergence

Motion restoration. Previously, we modeled the camera movements using the GMM which we call M^{gmm}. To restore the motions, we assume that each motion vector $s'_n \in I^{V'}$ is affected by the camera movements' average values μ_m, where s_n is clustered as one of the $m \in M$ distributions as $m_n = M^{gmm}(s'_n)$. Here, we call such a clustered motion vector (to distribution m_n) as $s_n^{\prime(m)}$. The following shows how we formulate the motion restoration for each motion vector:

$$for \quad s_n^{\prime(m)} \in I^{V'} : \quad s^n = s_n^{\prime(m)} - \mu_m \tag{5}$$

In the above, $s^n \in I^V$ are the corrected motion vectors. As we discussed before, our GMM model, M^{gmm}, is parameterized based on 2D variables $s'_n = (u'_n, v'_n) \in \mathbb{R}^2$. So, the mean variable for the distribution of m is also parameterized as $\mu_m = (\mu_{u'm}, \mu_{v'm})$.

After converting the corrected motion vectors, I^V to the corrected optical flow image, I^M, we use it in the next step as a more effective motion modality.

3.3. Multi-modal transformer

Fig. 5 shows the architecture of our multi-modal attention transformer network. The embedding ρ maps the inputs I^M and $I^S \in \mathbb{R}^{T\times h\times w\times 3}$ to X^M and $X^S \in \mathbb{R}^{T\times Z}$, where T is the temporal length, Z is the embedding size, and h and w are frame sizes. ρ is a two-stream convolutional network [ref] that embeds spatial and temporal (motion) modalities separately.

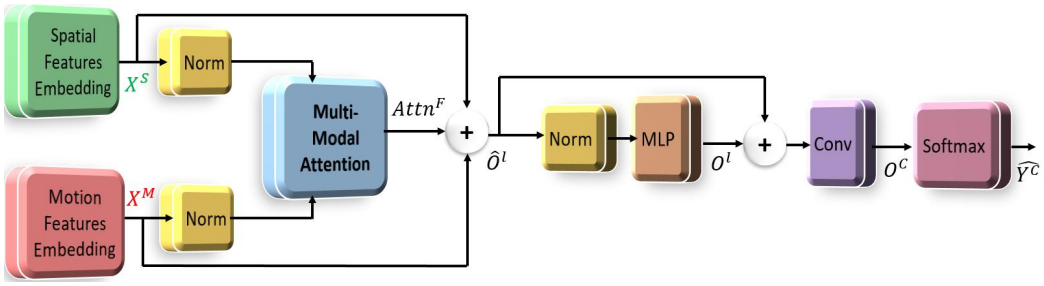


Figure 5: The architecture of our multi-modal transformer network includes the normalization layer (*Norm*), *Multi-Modal Attention*, *MLP*, and the classification modules including the *Conv* and *SoftMax* layers.

We feed the embedded spatial and motion inputs, X^M and X^S to our multi-modal transformer network. Our transformer network includes several layers whose relations between the consecutive layers $l-1$ and l are defined as follows:

$$\hat{O}^l = MMA(Norm(O^{l-1})) + O^{l-1}, \qquad l \in \{2, ..., L\}, \tag{6}$$

$$O^l = MLP(Norm(\hat{O}^l)) + \hat{O}^l, \qquad l \in \{2, ..., L\}, \tag{7}$$

In the above, $\hat{O}$ is the intermediate layer output, O is the layer output, *Norm* is the normalization layer, *MMA* is the Multi-Modal Attention, *MLP* is a Multilayer Perceptron layer, and L is the total number of layers.

For the first layer we have:

$$\hat{O}^1 = MMA(Norm(X^S, X^M)) + X^S + X^M, \tag{8}$$

And for the final layer, we will have:

$$\hat{Y}^C = p(W^F | X^S, X^M) = Softmax(Conv(O^L)), \tag{9}$$

In the above, $\hat{Y}^C$ is the action prediction scores for each frame, W^F is the transformer network model parameters, *Conv* is a convolutional layer, where $Conv : O^L \in \mathbb{R}^{T^L \times Z} \rightarrow O^C \in \mathbb{R}^C$, where C is the number of classes, O^C is the final output of the transformer before the Softmax layer, and T^L is the temporal length of the final layer.

3.3.1. Multi-modal attention

The transformer is a state-of-the-art deep network for solving spatial-temporal problems [12, 32]. One of the main advantages of the transformer network is the self-attention mechanism that computes the correlative patterns among selective inputs. As we discussed before in Section 1.1, finding the correlations among different spatial and motion modalities can empower the feature representation of actions. Hence, we propose a multi-modal attention mechanism to calculate such correlative patterns among our selective inputs which are spatial and motion modalities. Our multi-modal attention is illustrated in Fig. 6. An attention mechanism is defined as finding the correlations between the selective input, *Query* (Q), and other input candidates, *Keys* (K) which gives us the mapped correlative results, *Values* (V). To find the correlative patterns among each modality we introduce four attentions: (1) Spatial-spatial attention, $Attn^{S-S}$ computes the correlations between the spatial query, Q^S, and spatial keys, K^S, mapped to the spatial values, V^S; (2) The motion-motion attention ($Attn^{M-M}$) is obtained by mapping the correlations between motion query (Q^M) and motion keys (K^M) to motion values (V^M); (3) The spatial-motion attention ($Attn^{S-M}$) is calculated by first finding the correlations between Q^S and K^M, which then is mapped to V^S; (4) The motion-spatial attention ($Attn^{M-S}$) is obtained similarly, but the query, keys, and values for the two modalities are switched.

We formulate the query, keys, and values for both modalities as follows:

$$Q^S = X^S W_q^S, \qquad K^S = X^S W_k^S, \qquad V^S = X^S W_v^S, \tag{10}$$

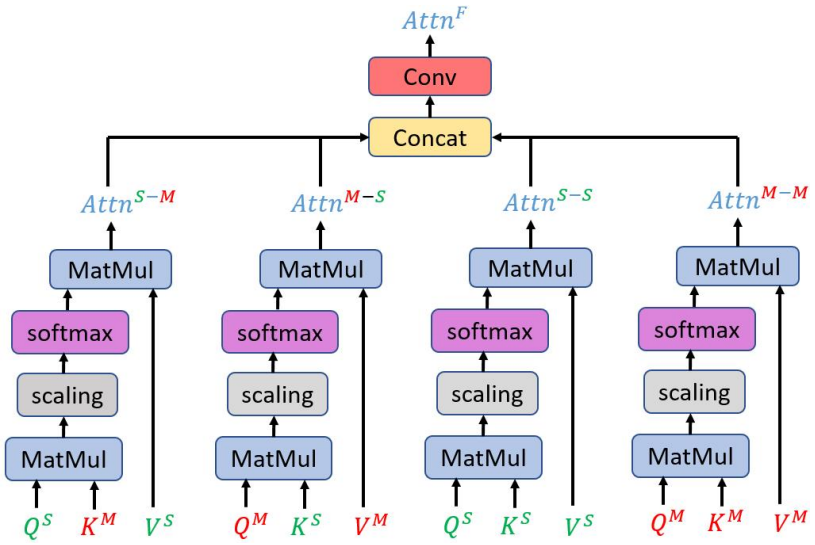


Figure 6: Our Multi-Modal Attention mechanism to compute the correlative pattern among spatial (RGB images) and motion (optical flow) modalities. It includes various multi-modal attentions such as motion-motion ($Attn^{M-M}$), spatial-motion ($Attn^{S-M}$), motion-spatial ($Attn^{M-S}$), and spatial-spatial ($Attn^{S-S}$).

$$Q^M = X^M W_q^M, \qquad K^M = X^M W_k^M, \qquad V^M = X^M W_v^M, \tag{11}$$

In the above $W_q^S \in \mathbb{R}^{Z \times Z_q}$ is the spatial query projection weights, $W_k^S \in \mathbb{R}^{Z \times Z_k}$ is the spatial keys projection weights, and $W_v^S \in \mathbb{R}^{Z \times Z_v}$ is the spatial values projection weights, and Z_q, Z_k, and Z_v are the projection sizes for query, keys, and values, respectively. $W_q^M \in \mathbb{R}^{Z \times Z_q}$ indicates the motion query projection weights, $W_k^M \in \mathbb{R}^{Z \times Z_k}$ is the motion keys projection weights, and $W_v^M \in \mathbb{R}^{Z \times Z_v}$ is the motion values projection weights.

The four multi-modal attentions (following Fig. 6) are formulated as follows:

$$Attn^{S-S} = Softmax(\frac{Q^S (K^S)^T}{\sqrt{Z_m}}) V^S, \tag{12}$$

$$Attn^{S-M} = Softmax(\frac{Q^S (K^M)^T}{\sqrt{Z_m}}) V^S, \tag{13}$$

$$Attn^{M-S} = Softmax(\frac{Q^M (K^S)^T}{\sqrt{Z_m}}) V^M, \tag{14}$$

$$Attn^{M-M} = Softmax(\frac{Q^M(K^M)^T}{\sqrt{Z_m}})V^M, \tag{15}$$

In the above, Z_m is the model size. The details for all the aforementioned parameter values are explained in Section 3.4.

The final attention, $Attn^F$ is obtained as:

$$Attn^F = Conv(Concat(Attn^{S-S}, Attn^{S-M}, Attn^{M-S}, Attn^{M-M})), \tag{16}$$

In the above $Conv$ is a convolutional layer where $Conv : \mathbb{R}^{4\times T\times Z} \rightarrow \mathbb{R}^{T\times Z}$ and $Concat$ is a concatenation operator.

Our network's loss function is shown as follows:

$$L = -\sum_{t=1}^{T}\sum_{c=1}^{C} y_t^{(c)} log\hat{y}_t^{(c)} + \alpha Loss_{tIOU} \tag{17}$$

In the above, y and $\hat{y}$ are ground truth and predicted values for each video frame and class c and time t, respectively. $Loss_{tIOU}$ is the temporal intersection loss that indicates the similarities between the predicted video segment frames and positive ground truth frames for the duration of $1 \leq t \leq T$. α is a loss adjustment term.

3.4. Implementation details

Table 1 indicates the implementation details of our proposed pipeline.

All the experiments are conducted using PyTorch 1.7 on a server PC with dual Nvidia RTX 3090 GPUs (24GB VRAM), AMD Ryzen Threadripper 3990X 64-Core Processor, and 256GB of RAM.

4. Experimental Results

4.1. Datasets and experimental setup

4.1.1. Public benchmarks

We used three THUMOS14 [36] ActivityNet [37], and our collected instructional activity datasets. THUMOS14 and ActivityNet datasets are the most well-known untrimmed activity datasets that have been used widely for action detection. THUMOS14 consists of 413

Table 1: Implementation details of our proposed pipeline with associated paper section references.

Parameter	Value	Section
Number of GMM distributions (M)	16	3.2
Spatial and Temporal Features Embedding Size (Z)	1024	3.3
Number of Transformer Layers (L)	6	3.3
Kernel Size for *Conv*	3	3.3
Learning Rate	$1e^{-5}$	3.3
Number of Training Epochs	100	3.3
Optimizer	ADAM	3.3
Weight Decay	$1e^{-6}$	3.3
Maximum Temporal Window Length (T)	2304	3.3
Projection Size for Query and Keys (Z_q, and Z_k)	512	3.3.1
Projection Size for Values (Z_v)	1024	3.3.1
Number of MMA Heads	3	3.3.1
Model size (Z_m)	512	3.3.1
Loss Adjustment Term (α)	1	3.3.1

untrimmed videos of 20 action classes. Following [40, 41, 42], we used 200 videos for training and 213 videos for testing. ActivityNet consists of 20,000 videos of 200 action classes. Following [40, 41, 42], we used 10,024 videos for training and 4,926 videos for testing.

4.1.2. Instructional activity dataset

We created a dataset of instructional activities recorded from K-12 schools. We annotated 240 hours of instructional activity videos with a professional team of 9 annotators. Our 24 instructional activity class labels are shown in Fig. 7. Some frame examples of our instructional activity dataset are shown in Fig. 8. The public link to download our dataset will be on our website [43] when it is available online. In this experiment, we used 50 hours of our videos with training and testing set proportions of 80% and 20%, respectively.

4.1.3. Evaluation Metric

We used the mean average precision (mAP) at different thresholds of temporal intersection over union (tIoU) which is the most used metric in action detection. For the THUMOS14 and ActivityNet, we reported the results for the threshold sets of {0.3, 0.4, 0.5, 0.6, 0.7} and {0.5, 0.75, 0.95}, respectively. The aforementioned thresholds are the most common ones that have been used for these two datasets in the literature [40, 41, 42].

ACTIVITY TYPE
Whole Class Activity
Individual Activity
Small Group Activity
Transition
TEACHER SUPPORTING
One Student
Multiple Students with SS Interaction
Multiple Students **without** SS Interaction

TEACHER LOCATION
Sitting
Standing (T)
Walking
STUDENT LOCATION
Sitting on the carpet or floor
Sitting at group tables
Sitting at desk
Student(s) Walking or Standing

DISCOURSE
On Task Student Talking with Student
Student Raising Hand
REPRESENTING CONTENT
Book - Using or holding book
Worksheet - Using or holding
Notebook - Using or holding
Instructional tool - Using or holding
Presentation with Technology
Laptop/tablet -Using or holding
Student Writing
Teacher Writing

Figure 7: Acivity class labels of our instructional activity dataset.

Figure 8: Some example frames of our instructional activity dataset.

4.2. Comparative results on public datasets

We compared our methods with the state-of-the-art strategies including **AF (ECCV 2022)** [44], **ReAct (ECCV 2022)**, [45], **TadTR (TIP 2022)** [46], **AFSD (CVPR 2021)** [40], **VSGN (ICCV 2021)** [40], **BMN-CSA (ICCV 2021** [47], **TCANet (CVPR 2021)** [48], **MUSES (CVPR 2021)** [49], **TSA-Net (CVPR 2021)** [48], **RTD-Net (ICCV 2021)** [50] , **TAL-MR (ECCV 2020)** [51], **A2Net (TIP 2020)** [52], **BMN (ICCV 2019)**, [53], and **P-GCN (ICCV 2019)** [42]. The comprative results for the THUMOS14 and ActivityNet datasets are shown in Table 2 and Table 3, respectively. As can be seen, our method outperformed the state-of-the-art approaches on these two public benchmarks based on different mAP thresholds.

Table 2: Comparison of our method, and the state-of-the-art methods on the THUMOS14 dataset.

Team (Year)	Method	maP@0.3	maP@0.4	maP@0.5	maP@0.6	maP@0.7	Avg
[52] (2020)	A2Net	58.6	54.1	45.5	32.5	17.2	41.6
[51] (2020)	TAL-MR	53.9	50.7	45.4	38.0	28.5	43.3
[50] (2021)	RTD-Net	68.3	62.3	51.9	38.8	23.7	49.0
[42] (2019)	P-GCN	69.1	63.3	53.5	40.4	26.0	50.5
[48] (2021)	TSA-Net	60.6	53.2	44.6	36.8	26.7	44.3
[46] (2022)	TadTR	62.4	57.4	49.2	37.8	26.3	46.6
[49] (2021)	MUSES	68.9	64.0	56.9	46.3	31.0	—
[53] (2019)	BMN	56.0	47.4	38.8	29.7	20.5	38.5
[48] (2021)	TCANet	60.6	53.2	44.6	36.8	26.7	44.3
[47] (2021)	BMN-CSA	64.4	58.0	49.2	38.2	27.8	47.7
[41] (2021)	VSGN	66.7	60.4	52.4	41.0	30.4	50.2
[40] (2021)	AFSD	67.3	62.4	55.5	43.7	31.1	52.0
[45] (2022)	ReAct	69.2	65.0	57.1	47.8	35.6	55.0
[44] (2022)	AF	82.1	77.8	71.0	59.4	43.9	66.8
We	**Ours(MMNet)**	**85.2**	**80.0**	**73.4**	**61.7**	**45.3**	**68.5**

4.3. Ablation study

We conducted an ablation study to evaluate the impact of the constituent components of our proposed method on the overall action detection performance.

Table 4 shows the impact of various multi-modal attentions on the overall action detection performance. As can be seen, the spatial-spatial attention, $Attn^{S-S}$, slightly led to a better performance than the motion-motion attention, $Attn^{M-M}$. On the other hand, the cross-modality attentions, spatial-motion $Attn^{S-M}$, and motion-spatial $Attn^{M-S}$, resulted in competitive performance compared to other attentions. Using all the attentions jointly, however, led to the maximum overall action detection performance.

Table 3: Comparison of our method, and the state-of-the-art methods on the ActivityNet dataset.

Team (Year)	Method	maP@0.5	maP@0.75	maP@0.95	Avg
[52] (2020)	A2Net	43.6	28.7	3.7	27.8
[51] (2020)	TAL-MR	43.5	33.9	9.2	30.2
[50] (2021)	RTD-Net	47.2	30.7	8.6	30.8
[42] (2019)	P-GCN	48.3	33.2	3.3	31.1
[48] (2021)	TSA-Net	48.7	32.0	9.0	31.9
[46] (2022)	TadTR	49.1	32.6	8.5	32.3
[49] (2021)	MUSES	50.0	35.0	6.6	34.0
[53] (2019)	BMN	50.1	34.8	8.3	33.9
[48] (2021)	TCANet	52.3	36.7	6.9	35.5
[40] (2021)	AFSD	52.4	35.3	6.5	34.4
[41] (2021)	VSGN	52.4	36.0	8.4	35.1
[47] (2021)	BMN-CSA	52.4	36.2	5.2	35.4
[45] (2022)	ReAct	49.6	33.0	8.6	32.6
[44] (2022)	AF	54.7	37.8	8.4	36.6
We	**Ours(MMNet)**	**58.1**	**39.5**	**9.1**	**39.0**

Table 4: Impact of different types of attention and their combinations on the overall action detection performance. Multi-modal attentions are motion-motion $Attn^{M-M}$, spatial-motion $Attn^{S-M}$, motion-spatial $Attn^{M-S}$, and spatial-spatial $Attn^{S-S}$

Attention type	maP@0.3	maP@0.4	maP@0.5	maP@0.6	maP@0.7	Avg
$Attn^{S-S}$	80.6	76.5	70.6	58.3	41.8	64.7
$Attn^{M-M}$	80.3	76.2	70.3	58.0	41.5	64.4
$Attn^{S-M}$	83.4	78.3	71.7	60.5	43.8	65.9
$Attn^{M-S}$	83.3	78.4	71.5	60.2	43.3	65.4
$Attn^{S-S}$ + $Attn^{M-M}$	83.6	78.7	72.1	60.9	44.2	66.4
$Attn^{S-S}$ + $Attn^{M-M}$ + $Attn^{S-M}$ + $Attn^{M-S}$	**85.2**	**80.0**	**73.4**	**61.7**	**45.3**	**68.5**

Table 5 illustrates the impact of our motion distortion correction algorithm on the overall action detection performance. As can be seen, using our motion distortion correction algorithm resulted in higher performance.

Table 5: Impact of our motion distortion correction algorithm on the overall action detection performance.

Option	maP@0.3	maP@0.4	maP@0.5	maP@0.6	maP@0.7	Avg
Without motion distortion correction	83.5	79.1	71.9	60.7	44.2	66.6
With motion distortion correction	85.2	80.0	73.4	61.7	45.3	68.5

4.4. Experimental results on instructional activity dataset

Table 6 shows the comparative results on our instructional activity dataset based on the average performance. As can be seen, our method outperformed the other methods by a large margin. Fig. 9 indicates the average performance of our proposed method on our instructional activity dataset separated for each class label.

Table 6: Comparison of our proposed method with the state-of-the-art approaches on our instructional activity dataset.

Method	MLAD [54]	SE [55]	BF [40]	GA [56]	LST [57]	COLA [58]	**Ours (MMNet)**
Avg	45.2	36.7	25.3	26.0	42.0	34.1	68.1

5. Conclusions

This paper proposed a novel transformer network for detecting actions in untrimmed videos. Our transformer network utilizes a new multi-modal attention mechanism to capture the correlative patterns between spatial (RGB) and motion (optical flow) features. Such correlative features improve the expressive power of action modeling. To be able to use the motion (optical flow) inputs more effectively, we also suggested a motion distortion correction algorithm to handle camera movements that can severely distort the motion vectors represented in the optical flow. We also introduced a new instructional activity dataset captured from K-12 schools. Our proposed method outperformed the state-of-the-art approaches on two public benchmarks, THUMOS14 and ActivityNet as well our instructional activity dataset.

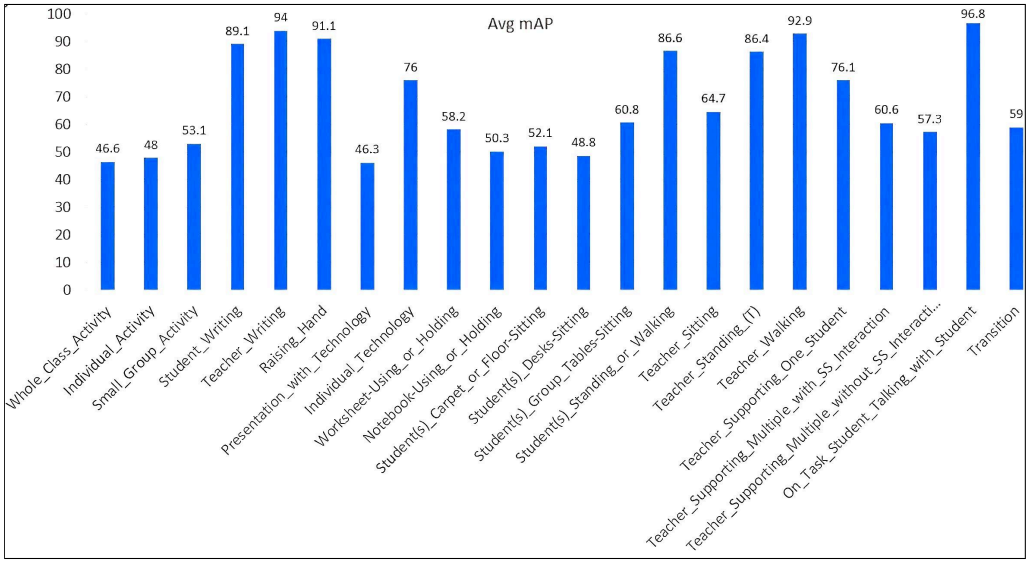


Figure 9: Average performance of our proposed method per action class evaluated on our instructional activity dataset.

Our study is beneficial for other researchers in the field as we are the first to suggest capturing the correlative patterns between RGB and optical flow using an effective multi-modal attention mechanism. Moreover, our novel motion distortion correction algorithm is highly advantageous in dealing with camera movement which is common in real-world scenarios and in the wild.

Future works. While our motion distortion algorithm is highly effective in dealing with camera movements, it still depends on a person detection algorithm to segment the background and foreground. We suggest modeling the background preferably within the action detection network itself. Moreover, for our multi-modal transformer, we suggest separating the semantics (both RGB and optical flow) in the scene to capture the correlative patterns among local objects/subjects instead of the whole action frames.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 2000487 and the Robertson Foundation under Grant No. 9909875. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

References

[1] S. Ali, M. Shah, Human action recognition in videos using kinematic features and multiple instance learning, IEEE transactions on pattern analysis and machine intelligence 32 (2) (2008) 288–303.

[2] Z. Yin, J. Shi, Geonet: Unsupervised learning of dense depth, optical flow and camera pose, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1983–1992.

[3] E. Vahdani, Y. Tian, Deep learning-based action detection in untrimmed videos: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[4] K. Guo, P. Ishwar, J. Konrad, Action recognition using sparse representation on covariance manifolds of optical flow, in: 2010 7th IEEE international conference on advanced video and signal based surveillance, IEEE, 2010, pp. 188–195.

[5] L. Sevilla-Lara, Y. Liao, F. Güney, V. Jampani, A. Geiger, M. J. Black, On the integration of optical flow and action recognition, in: German conference on pattern recognition, Springer, 2018, pp. 281–297.

[6] S. Sun, Z. Kuang, L. Sheng, W. Ouyang, W. Zhang, Optical flow guided feature: A fast and robust motion representation for video action recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1390–1399.

[7] A. Ranjan, V. Jampani, L. Balles, K. Kim, D. Sun, J. Wulff, M. J. Black, Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 12240–12249.

[8] S. Herath, M. Harandi, F. Porikli, Going deeper into action recognition: A survey, Image and vision computing 60 (2017) 4–21.

[9] H. Wang, C. Schmid, Action recognition with improved trajectories, in: Proceedings of the IEEE international conference on computer vision, 2013, pp. 3551–3558.

[10] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, D. Lin, Temporal action detection with structured segment networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2914–2923.

[11] J. Yuan, Z. Liu, Y. Wu, Discriminative video pattern search for efficient action detection, IEEE Transactions on pattern analysis and machine intelligence 33 (9) (2011) 1728–1743.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30.

[13] R. Girdhar, J. Carreira, C. Doersch, A. Zisserman, Video action transformer network, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 244–253.

[14] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, Advances in neural information processing systems 27.

[15] C. Feichtenhofer, A. Pinz, A. Zisserman, Convolutional two-stream network fusion for video action recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 1933–1941.

[16] M. Ma, N. Marturi, Y. Li, A. Leonardis, R. Stolkin, Region-sequence based six-stream cnn features for general and fine-grained human action recognition in videos, Pattern Recognition 76 (2018) 506–521.

[17] Z. Tu, W. Xie, Q. Qin, R. Poppe, R. C. Veltkamp, B. Li, J. Yuan, Multi-stream cnn: Learning representations based on human-related regions for action recognition, Pattern Recognition 79 (2018) 32–43.

[18] Y. Shi, Y. Tian, Y. Wang, T. Huang, Sequential deep trajectory descriptor for action recognition with three-stream cnn, IEEE Transactions on Multimedia 19 (7) (2017) 1510–1520.

[19] X. Wang, L. Gao, P. Wang, X. Sun, X. Liu, Two-stream 3-d convnet fusion for action recognition in videos with arbitrary size and length, IEEE Transactions on Multimedia 20 (3) (2017) 634–644.

[20] Y. Zhu, Z. Lan, S. Newsam, A. Hauptmann, Hidden two-stream convolutional networks for action recognition, in: Asian conference on computer vision, Springer, 2018, pp. 363–378.

[21] Q. Xiong, J. Zhang, P. Wang, D. Liu, R. X. Gao, Transferable two-stream convolutional neural network for human action recognition, Journal of Manufacturing Systems 56 (2020) 605–614.

[22] X. Peng, C. Schmid, Multi-region two-stream r-cnn for action detection, in: European conference on computer vision, Springer, 2016, pp. 744–759.

[23] M. Zhang, C. Gao, Q. Li, L. Wang, J. Zhang, Action detection based on tracklets with the two-stream cnn, Multimedia Tools and Applications 77 (3) (2018) 3303–3316.

[24] M. Zhang, H. Hu, Z. Li, J. Chen, Action detection with two-stream enhanced detector, The Visual Computer (2022) 1–12.

[25] R. Su, D. Xu, L. Zhou, W. Ouyang, Progressive cross-stream cooperation in spatial and temporal domain for action localization, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (12) (2020) 4477–4490.

[26] P. Zhang, Y. Cao, B. Liu, Multi-stream single shot spatial-temporal action detection, in: 2019 IEEE International Conference on Image Processing (ICIP), IEEE, 2019, pp. 3691–3695.

[27] Y. Zhai, L. Wang, W. Tang, Q. Zhang, N. Zheng, D. Doermann, J. Yuan, G. Hua, Adaptive two-stream consensus network for weakly-supervised temporal action localization, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[28] Y. Chang, Z. Tu, W. Xie, B. Luo, S. Zhang, H. Sui, J. Yuan, Video anomaly detection with spatio-temporal dissociation, Pattern Recognition 122 (2022) 108213.

[29] J. Kim, G. Li, I. Yun, C. Jung, J. Kim, Weakly-supervised temporal attention 3d network for human action recognition, Pattern Recognition 119 (2021) 108068.

[30] V. Mazzia, S. Angarano, F. Salvetti, F. Angelini, M. Chiaberge, Action transformer: A self-attention model for short-time pose-based human action recognition, Pattern Recognition 124 (2022) 108487.

[31] W. Hu, H. Liu, Y. Du, C. Yuan, B. Li, S. Maybank, Interaction-aware spatio-temporal pyramid attention networks for action classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (10) (2022) 7010–7028. doi:10.1109/TPAMI.2021.3100277.

[32] D. Neimark, O. Bar, M. Zohar, D. Asselmann, Video transformer network, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3163–3172.

[33] C. Dai, X. Liu, J. Lai, Human action recognition using two-stream attention based lstm networks, Applied soft computing 86 (2020) 105820.

[34] W. Dong, Z. Zhang, C. Song, T. Tan, Identifying the key frames: An attention-aware sampling method for action recognition, Pattern Recognition (2022) 108797.

[35] J. Carreira, A. Zisserman, Quo vadis, action recognition? a new model and the kinetics dataset, in: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6299–6308.

[36] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, M. Shah, The thumos challenge on action recognition for videos "in the wild", Computer Vision and Image Understanding 155 (2017) 1–23.

[37] F. Caba Heilbron, V. Escorcia, B. Ghanem, J. Carlos Niebles, Activitynet: A large-scale video benchmark for human activity understanding, in: Proceedings of the ieee conference on computer vision and pattern recognition, 2015, pp. 961–970.

[38] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, in: European conference on computer vision, Springer, 2020, pp. 213–229.

[39] C. M. Bishop, N. M. Nasrabadi, Pattern recognition and machine learning, Vol. 4, Springer, 2006.

[40] C. Lin, C. Xu, D. Luo, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, Y. Fu, Learning salient boundary feature for anchor-free temporal action localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3320–3329.

[41] C. Zhao, A. K. Thabet, B. Ghanem, Video self-stitching graph network for temporal action localization, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 13658–13667.

[42] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, C. Gan, Graph convolutional networks for temporal action localization, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 7094–7103.

[43] U. of Virginia, Aiai project (2022).
URL https://aiaiproject.weebly.com/

[44] C. Zhang, J. Wu, Y. Li, Actionformer: Localizing moments of actions with transformers, in: European Conference on Computer Vision, 2022.

[45] Q. C. J. Z. L. M. J. L. Dingfeng Shi, Yujie Zhong, D. Tao, React: Temporal action detection with relational queries, in: European Conference on Computer Vision, 2022.

[46] X. Liu, Q. Wang, Y. Hu, X. Tang, S. Zhang, S. Bai, X. Bai, End-to-end temporal action detection with transformer, IEEE Transactions on Image Processing 31 (2022) 5427–5441.

[47] D. Sridhar, N. Quader, S. Muralidharan, Y. Li, P. Dai, J. Lu, Class semantics-based attention for action detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 13739–13748.

[48] Z. Qing, H. Su, W. Gan, D. Wang, W. Wu, X. Wang, Y. Qiao, J. Yan, C. Gao, N. Sang, Temporal context aggregation network for temporal action proposal refinement, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 485–494.

[49] X. Liu, Y. Hu, S. Bai, F. Ding, X. Bai, P. H. Torr, Multi-shot temporal event localization: a benchmark, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 12596–12606.

[50] J. Tan, J. Tang, L. Wang, G. Wu, Relaxed transformer decoders for direct action proposal generation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 13526–13535.

[51] P. Zhao, L. Xie, C. Ju, Y. Zhang, Y. Wang, Q. Tian, Bottom-up temporal action localization with mutual regularization, in: European Conference on Computer Vision, Springer, 2020, pp. 539–555.

[52] L. Yang, H. Peng, D. Zhang, J. Fu, J. Han, Revisiting anchor mechanisms for temporal action localization, IEEE Transactions on Image Processing 29 (2020) 8535–8548.

[53] T. Lin, X. Liu, X. Li, E. Ding, S. Wen, Bmn: Boundary-matching network for temporal action proposal generation, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 3889–3898.

[54] P. Tirupattur, K. Duarte, Y. S. Rawat, M. Shah, Modeling multi-label action dependencies for temporal action localization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 1460–1470.

[55] A. Piergiovanni, M. S. Ryoo, Learning latent super-events to detect multiple activities in videos, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5304–5313.

[56] B. Shi, Q. Dai, Y. Mu, J. Wang, Weakly-supervised action localization by generative attention modeling, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1009–1019.

[57] M. Xu, Y. Xiong, H. Chen, X. Li, W. Xia, Z. Tu, S. Soatto, Long short-term transformer for online action detection, Advances in Neural Information Processing Systems 34 (2021) 1086–1099.

[58] C. Zhang, M. Cao, D. Yang, J. Chen, Y. Zou, Cola: Weakly-supervised temporal action localization with snippet contrastive learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16010–16019.

Matthew Korban received his BSc and MSc degree in Electrical Engineering in 2013 from the University of Guilan, where he worked on sign language recognition in video. He received his PhD in Computer Engineering from Louisiana State University. He is currently a Postdoc Research Associate at the University of Virginia, working with Prof. Scott T. Acton. His research interest includes Human Action Recognition, Early Action Recognition, Motion Synthesis, and Human Geometric Modeling in Virtual Reality environments.

Peter Youngs is a professor in the Department of Curriculum, Instruction and Special Education at the University of Virginia. He studies how neural networks can be used to automatically classify instructional activities in videos of elementary mathematics and reading instruction. He currently serves as co-editor of American Educational Research Journal.

Scott T. Acton received his M.S. and Ph.D. degrees at the University of Texas at Austin. He received his B.S. degree at Virginia Tech. Professor Acton is a Fellow of the IEEE "for contributions to biomedical image analysis." Currently, Acton is a program director in the Computer and Information Science and Engineering at the U.S. National Science Foundation. He is also professor of Electrical and Computer Engineering and of Biomedical Engineering at the University of Virginia. Professor Acton's laboratory at UVA is called VIVA - Virginia Image and Video Analysis. They specialize in bioimage analysis problems. Professor Acton has over 300 publications in the image analysis area including the books *Biomedical Image Analysis: Tracking* and *Biomedical Image Analysis: Segmentation*. He was the 2018 Co-Chair of the IEEE International Symposium on Biomedical Imaging. Professor Acton was recently Editor-in-Chief of the IEEE Transactions on Image Processing (2014-2018).