Human Interaction Learning on 3D Skeleton Point Clouds for Video Violence Recognition

Yukun Su[1,2], Guosheng Lin[3*], Jinhui Zhu[1,2], and Qingyao Wu[1,2*]

[1] School of Software Engineering, South China University of Technology
[2] Key Laboratory of Big Data and Intelligent Robot, Ministry of Education
suyukun666@gmail.com,{csjhzhu,qyw}@scut.edu.cn
[3] Nanyang Technological University, Singapore
gslin@ntu.edu.sg

Abstract. This paper introduces a new method for recognizing violent behavior by learning contextual relationships between related people from human skeleton points. Unlike previous work, we first formulate 3D skeleton point clouds from human skeleton sequences extracted from videos and then perform interaction learning on these 3D skeleton point clouds. A novel **S**keleton **P**oints **I**nteraction **L**earning (SPIL) module, is proposed to model the interactions between skeleton points. Specifically, by constructing a specific weight distribution strategy between local regional points, SPIL aims to selectively focus on the most relevant parts of them based on their features and spatial-temporal position information. In order to capture diverse types of relation information, a multi-head mechanism is designed to aggregate different features from independent heads to jointly handle different types of relationships between points. Experimental results show that our model outperforms the existing networks and achieves new state-of-the-art performance on video violence datasets.

1 Introduction

Generally, the concept of video-based violence recognition is defined as detecting violent behaviors in video data, which is of vital importance in some video surveillance scenarios like railway stations, prisons or psychiatric centers. Consider some sample frames from public datasets as shown in Fig 1(a). When we humans see the sequences of images, we can easily recognize those violent actions through the human body's torso movements such as "kick", "beat", "push", etc.

However, current deep learning approaches fail to capture these ingredients precisely in a multi-dynamic and complex multi-people scene. For instance, the approaches based on two-stream ConvNets [27,32] are learning to classify actions based on individual video frames or local motion vectors. However, such local motions that are captured by optical flow [2] sometimes fail to satisfy the dynamics modeling of shape change in multiple motion states. To tackle this limitation,

* Corresponding authors.

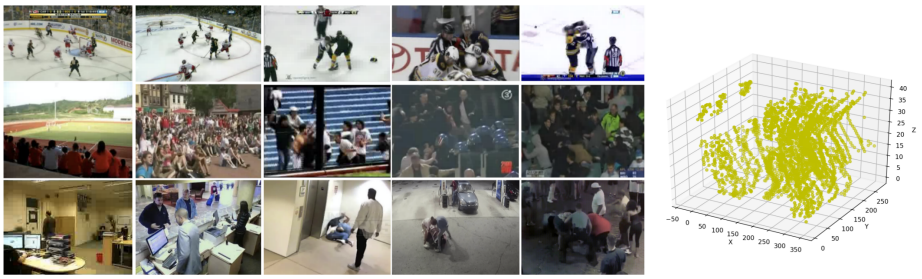

Fig. 1. (a) Sample frames from the Hockey-Fight [21] dataset (first row), the Crowd Violence [15] dataset (second row) and the RWF-2000 Violence [5] dataset (third row). In each row, the left two columns are non-violent scenes while the right three columns are violent scenes. (b) Skeleton point clouds for a certain video.

recent Recurrent Neural Networks [10,38] and 3D Convolutions [3,29,30] works have also focused on modeling long term temporal information. However, all these frameworks focus on the features extracted from the whole scenes, leading to the interference by irrelevant information in the scenarios, and fail to capture region-based relationships. Meanwhile, the existing vision-based methods are mainly based on hand-crafted features such as statistic features between motion regions, leading to poor adaptability to another dataset. In violence recognition, extracting such appearance features and dynamics information of objects suffer from a number of complexities. Therefore, the above methods are often not very effective.

The movements of people in the video are reflected in human skeletal point sequences, which can be converted into 3D point clouds. We can then perform feature extraction on this data. Our experiments show that existing 3D point clouds methods [25,34,36] can readily be applied to the violence recognition task. However, current methods, while excellent at extracting pertinent features on ordinary point clouds, they lack the ability to focus on relevant points and their interactions.

Based on these observations, we introduce a novel approach to perform video violence recognition via a human skeleton point convolutional reasoning framework. We first represent the input video as the cluster of 3D point clouds data as shown in Fig 1(b) through extracting the human skeleton sequences pose coordinates from each frames in the video. In order to better observe the characteristics of the skeleton points and determine whether there is violence, specifically, (i) **Skeleton Points Interaction Learning (SPIL) module**: the weight distribution among regional points with high coupling degree or strong semantic correlation is relatively high. Based on this, we can capture the appearance features and spatial-temporal structured position relation of skeleton points uniformly

avoiding feature contamination between objects, and model how the state of the same object changes and the dependencies between different objects in frames. (ii) **Multi-head mechanism**: the single head is responsible for processing the action information for the skeleton points without interference from the scene. Multiple heads attend to fuse features from different independent heads to capture different types of information among points parallelly, which can enhance the robustness of the network.

In summary, we highlight the major contributions of this paper in three folds:

- We formulate the video violence recognition task as 3D skeleton point clouds recognition problem, and we propose an effective interaction learning method on skeleton point clouds for video recognition.
- We propose a novel SPIL module, which can be learned on the human skeleton points to capture both feature and position relation information simultaneously. And the multi-head mechanism allows SPIL to capture different types of points interactions to improve robustness.
- Different from the previous methods, we use the skeleton point clouds technique on recognition in violent videos, and our approach significantly outperforms state-of-the-art methods by a large margin.

2 Related Work

Video classification with deep learning: Most recent works on video classification are based on deep learning. Initial approaches explored methods to combine temporal information based on pooling or temporal convolution [17,38]. To jointly explore spatial and temporal information of videos, 3D convolutional networks have been widely used. Tran et al. [29] trained 3D ConvNets on the large-scale video datasets, where they experimentally found that a $3 \times 3 \times 3$ convolutional kernel can learn both appearance and motion features. In a later work, Hara et al. [14] studied the use of a Resnet architecture with 3D convolutions and Xie et al. [37] exploited aggregated residual transformations to show the improvements. Two-stream networks [3,4,32] have also been attracting high attention, they took the input of a single RGB frame (captures appearance information) and a stack of optical flow frames (captures motion information). An alternative way to model the temporal relation between frames is by using recurrent networks [19,6]. However, these above approaches encountered bottlenecks in feature extraction when faced with more complex scenes and more irregular dynamic features in video violence recognition tasks. They fail to fully capture the comprehensive information in the entire video and are difficult to focus on distinguishing violent behavior in multiple characters and action features.

3D Point Clouds: To adapt the 3D points coordinates data for convolution, one straightforward approach is to voxelize it in a 3D grid structure [12,28]. OctNet [26] explored the sparsity of voxel data and alleviated this problem. However, since voxels are the discrete representations of space, this method still requires high-resolution grids with large memory consumption as a trade-off to keep a level of representation quality. Because the body keypoints themselves are a very

sparse spatial structure, the application of the 3D voxel method to the skeleton points will lead to insufficient data characterization and make it difficult to train the model. In this trend, PointNet [24] first discusses the irregular format and permutation invariance of point sets, and presents a network that directly consumes point clouds. PointNet++ [25] extends PointNet by further considering not only the global information but also the local details with a farthest-sampling-layer and a grouping-layer. Deep learning in graph [1] is a modern term for a set of emerging technologies that attempt to address non-Euclidean structured data (e.g., 3D point clouds, social networks or genetic networks) by deep neural networks. Graph CNNs [20,7] show advantages of graph representation in many tasks for non-Euclidean data, as it can naturally deal with these irregular structures. [41] builds a graph CNN architecture to capture the local structure and classify point clouds, which also proves that deep geometric learning has enormous potential for unordered point clouds analysis. Nonetheless, these works ignore the different importance of each point's contribution, especially in skeleton points. Even though some works [31,13] suggest the use of attention, they seem to be of little use in the processing of specific spatial-temporal relation in the skeleton point clouds.

To this end, different from these works, our approach encodes dependencies between objects with both feature and position relations, which focus on specific human dynamic action expressions ignoring action-independent information. This framework provides a significant boost over the state-of-the-art.

3 Proposed Method

Our goal task is to represent the video as human skeleton point clouds of objects and perform reasoning for video violence recognition. To this end, we propose the multi-head Skeleton Points Interaction Learning (SPIL) module to deal with the interrelationships between points. In this section, we will give detailed descriptions of our approach. 3.1 presents an overview of our framework. 3.2 introduces the detail of the SPIL module . 3.3 shows the multi-head mechanism and 3.4 describes how to train and infer this method for skeleton point clouds.

3.1 Framework

We propose to tackle this problem with an architecture, as illustrated in Fig 2. Our approach takes raw video clips as input streams. First, we follow the human pose extraction strategy used in [11], which can detect body points from each frames in the video. The coordinate (x, y, z) of each keypoint represents the position of the current point in each frame, where z represents the t^{th} frame. Then we collect all the skeleton points sequences and transform the dynamic representation of the people in the video into a point clouds structure.

Subsequently, a SPIL abstraction level module takes an $N \times (3 + C)$ matrix as input that is from N centroid points which are sampled following the scheme of [25] with 3-dim coordinates and C-dim points feature, and it outputs an

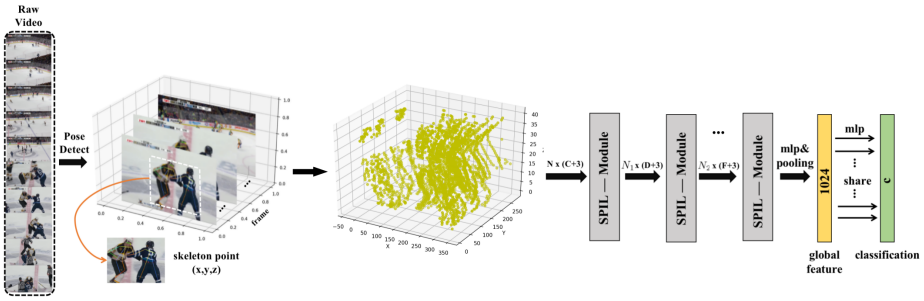

Fig. 2. Model Pipeline Overview. Our model uses the pose detection method to extract skeleton coordinates from each frames of the video. These human skeleton points are provided as point clouds inputs to the SPIL modules which perform information propagation based on assigning different weights to different skeleton points. Finally, a global feature is extracted to perform classification.

$N_1 \times (3 + D)$ matrix of N_1 subsampled points with 3-dim coordinates and new D-dim feature vectors representing local features. To obtain various relation information between points, we design a multi-head mechanism in SPIL.

With this representation, we apply several SPIL modules to sample and extract the skeleton point clouds, and finally, classify the global feature through a fully connected layer to complete the video recognition.

3.2 Skeleton Points Interaction Learning Module

To compute the interaction weights between points, the standard general approach is determined by the K neighbors. However, not all nearby points have an effect on the current point. For example, if there are multiple characters in a scene, irrelevant skeleton points can sometimes be confusing and the learned feature characterizes all of its neighbors indistinguishably. To address this problem, in our SPIL module, as shown in Fig 3 intuitively, the points interaction weights on different skeleton points are distributed based on the relationships between points. We learn to mask or weaken part of the convolution weights according to the neighbors' feature attributes. In this way, the network can focus on the skeleton points for prediction.

Consider a point set $\{p_1, p_2, ..., p_K\} \in \mathbb{R}^3$ according to a centroid point's K neighbors, where the local region is grouped within a radius. We set the radius to $(r \times T_{frame})$ that guarantees local region to cover more inter- and intra-frames points across space and time. Particularly, the pair-wise interaction weights between points can be mathematically considered to use W to represent, where the weight W_{ij} indicates the connection of point j to point i. In the case of $W \in \mathbb{R}^{K \times K}$, we define the set of points as $\mathcal{S} = \left\{(p_i^f, p_i^l)|i = 1, ..., K\right\}$. Among

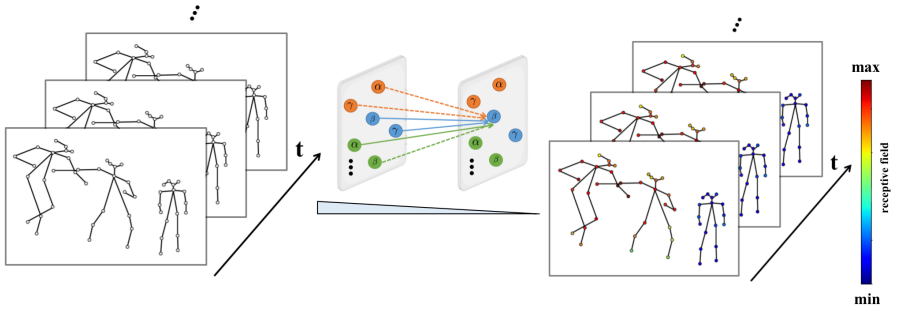

Fig. 3. Illustration of a single SPIL module on a subgraph of the human skeleton point clouds. Input points with a constant scalar feature (in grey) are convolved through a human skeleton points interaction filter. The output is a dynamically attentional weighted combination of the neighbor's points. The weights on irrelevant points (the dotted arrows) are masked so that the convolution kernel can focus on the correlate points for prediction.(α,β,γ of different colors denote parts of different human joint points respectively, such as left hand, right elbow, left ankle).

them, $p_i^f \in \mathbb{R}^C$ is point i's C-dim feature, here we take points' confidence as initial feature, and $p_i^l = (l_i^{\mathrm{x}}, l_i^{\mathrm{y}}, l_i^{\mathrm{z}}) \in \mathbb{R}^3$ is the 3-dim position coordinates.

For the distribution of the weights W, unlike the traditional point operation, in order to obtain distinguishable representation ability to capture the correlation between different skeletal points, it is necessary to consider both feature similarity and position characteristic relation. To this end, we separately explore the feature and position information and then perform high-level modeling of them to dynamically adapt to the structure of the objects. Concretely, the interaction weight of each neighboring point is computed as follows:

$$W_{ij} = \Phi(R^F(p_i^f, p_j^f), R^L(p_i^l, p_j^l)), \tag{1}$$

where $R^F(p_i^f, p_j^f)$ implies the feature relation between points and $R^L(p_i^l, p_j^l)$ denotes the position relation. Φ function plays the role of a combination of feature information and position information.

In this work, we follow [35] and use the differentiable architecture but different in building R^F and R^L functions detailedly to compute points interaction value, which can be formulated as:

$$W_{ij} = \frac{R^L(p_i^l, p_j^l)\ exp(R^F(p_i^f, p_j^f))}{\sum_{j=1}^{K} R^L(p_i^l, p_j^l)\ exp(R^F(p_i^f, p_j^f))}, \tag{2}$$

where the points interaction weights are normalized across all the neighbors of a point i to handle the size-varying neighbors across different points and spatial scales.

Feature Term: Intuitively, the points of different features in the local area exert various influences to enhance the expressive power of each point. Our feature modulator solves this problem by adaptively learning the amount of influence each feature in a point has on other points. we can utilize the dot-product operation to compute similarity in embedding space, and the corresponding function R^Fcan be expressed as:

$$R^F(p_i^f, p_j^f) = \phi(g(p_i^f))^\intercal \theta(g(p_j^f)), \tag{3}$$

where $g(\cdot)$: $\mathbb{R}^C \rightarrow \mathbb{R}^{C'}$ is a feature mapping function. $\phi(\cdot)$ and $\theta(\cdot)$ are two learnable linear projection functions, followed by ReLU, which project the features relation value between two points to a new space.

Position Term: In order to make full use of the spatial-temporal structure relation of points, the position characteristics of points should be taken into account. In our work, we consider the following three choices:

(1) Eu-distance in space: Considering the Euclidean distance between the points, the relatively distant points contribute less to the connection of the current point than the local points. With this in mind, we directly calculate the distance information and act on the points. The R^P is formed as:

$$R^L(p_i^l, p_j^l) = -ln(\sigma(\mathcal{D}(p_i^l, p_j^l))), \tag{4}$$

where $\sigma(\cdot)$ is a sigmoid activation function and the output range is controlled between [0, 1] to fed to the ln function. $\mathcal{D}$ is a function to calculate the distance between points.

(2) Eu-distance Spanning: Alternatively, we can first encode the relations between two points to a high-dimensional representation based on the position distance. Then the difference between two terms encourages to span the relations to a new subspace. Specifically, the position relation value is computed as:

$$R^L(p_i^l, p_j^l) = \frac{\psi(M_1(p_i^l) - M_2(p_j^l))}{\mathcal{D}(p_i^l, p_j^l)}, \tag{5}$$

where $M_1(\cdot)$ and $M_2(\cdot)$ are two multilayer perceptrons functions and $\psi(\cdot)$ is a linear projection function followed by ReLU that generates the embedded feature into a scalar.

(3) Eu-distance Masking: In addition, a more intuitive approach is to ignore some distant points and retain the characteristic contribution of the relative local points. For this purpose, we set a threshold to ignore the contribution of points and the function can be defined as:

$$R^L(p_i^l, p_j^l) = \begin{cases} 0 & , \ if \underset{(l_i^z = l_j^z)}{\mathcal{D}}(p_i^l, p_j^l) > d, \\ \psi(M_1(p_i^l)||M_2(p_j^l)) & , \ else. \end{cases} \tag{6}$$

Note that the radius ($r \times T_{frame}$) ensures that the local region cover points spatially and temporally. However, spatially, we try to mask out some weak

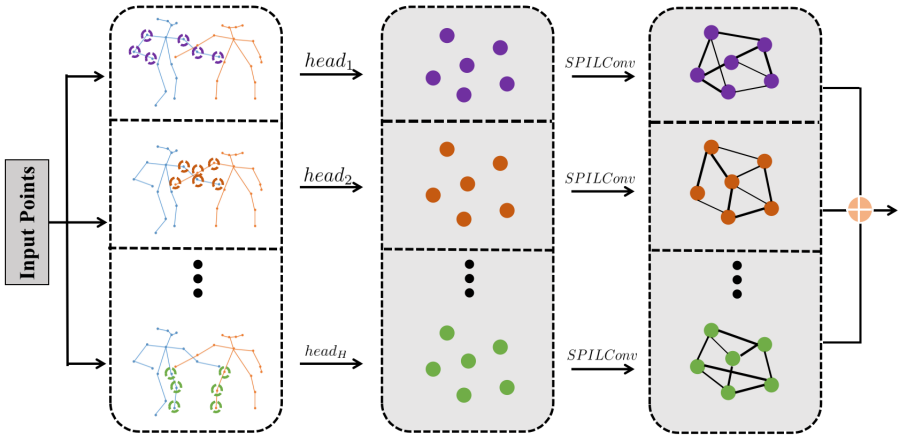

Fig. 4. Illustration of the multi-head SPIL. A single head encodes the skeleton point clouds from the input independently, and multiple different headers are responsible for processing different types of information from the points and eventually aggregate them together. Each node denotes a skeleton joint point and each edge is a scalar weight, which is computed according to two points' features and their relative position.

correlation points within the same frames. The implication is that we preserve the globality in time and the locality in space. || is the concatenation operation and the embedded feature between two points is transformed into a scalar by a learnable linear function, followed by a ReLU activation. d acts as a distance threshold which is a hyper-parameter.

3.3 Multi-head mechanism

Although a single head SPIL module can perform interaction feature extraction on skeleton points, because the connection between the joint points of the human body is ever-changing, each points may have different types of features. For example, a joint point has an information effect on its own posture and also a dynamic information effect on the interaction between human bodies at the same time, we call it a point with different types of features. Specifically, as shown in Fig 4, for a certain skeletal point such as the elbow joint, the first head may be sensitive to the information of the human elbow joint's own posture, such as judging whether it is a "punch" posture; while the second head is more concerned with the connection of elbow joint motion information between people to extract dynamic features. In the same way, the remaining (H-2) heads extract different features for other types that may be related.

For this reason, the designed multi-head mechanism allows the SPIL module to work in parallel to capture diverse types of relation points. Every weights W_ι is computed in the same way according to Eq 2, where $\iota \in H$ is the number of

heads. It should be noted that independent headers do not share weights during the calculation. By using the multi-head mechanism, the model can make more robust relational reasoning upon the points.

3.4 Skeleton Point Convolution

To perform reasoning on the skeleton points, unlike the standard 2D or 3D convolutions that run on a local regular grid. For a target point, the outputs are updated features of each object points from all its neighbors. We can represent one layer of convolutions as:

$$X^{(l+1)} = WX^{(l)}\mathcal{Z}^{(l)}, \tag{7}$$

where $W \in \mathbb{R}^{K\times K}$ represents the interaction weights we have introduced in Sec 3.2. $X^{(l)} \in \mathbb{R}^{K\times C'}$ is the input feature projected by $g(\cdot)$ mapping function of a centroid grouping skeleton point set. $\mathcal{Z}^{(l)} \in \mathbb{R}^{C'\times d}$ is the layer-specific learnable weight matrix. After each layer of convolutions, we adopt non-linear functions for activating before the feature $X^{(l+1)}$ is forwarded to the next layer. We stack the convolution operation into two layers in our work.

To combine multi-head weights, in this work, we employ the concatenation fusion function. We can extend Eq 7 as:

$$X^{(l+1)} = \mathop{\Big\|}_{\iota}^{H}(W_\iota X^{(l)}\mathcal{Z}_\iota^{(l)}, dim = 1), \tag{8}$$

where W_ι indicates different types of weights and the different $\mathcal{Z}_\iota$ are not shared. $\|(\cdot)$ function aggregates and fuses the output information of all H heads. Namely, all $K \times d$-dim features will concatenate together to form a new feature $\in \mathbb{R}^{K\times D}$, where $D = \sum_1^H d$. Thus the points can be increased in dimensionality to obtain more characteristic information. Afterward, for each subsampled centroid grouping points, $Maxpooling$ is applied for fusing K local region points that are updated by Eq 8.

Finally, as illustrated in Fig 2. the output relational features are forwarded to an average pooling layer, which calculates the mean of all the proposal features and leads to a 1×1024 dimensions representation. Then it is fed to the classifier to generate predictions. Y_W denotes the labels and $\hat{Y}_W$ are predictions, with standard cross-entropy loss, the final loss function is formed as:

$$\mathcal{L} = \mathcal{L}(Y_W, \hat{Y}_W). \tag{9}$$

4 Experiments

In this section, we evaluate our method on public violent video datasets. The results show that our method outperforms representative baseline methods and achieves state-of-the-art performance.

Dataset Details: We train and evaluate our model on four datasets (Hockey-Fight dataset [21], Crowd Violence dataset [15], Movies-Fight dataset [22] and

the RWF-200 Violence dataset [5]) for video violence recognition. To the best of our knowledge, RWF-2000 is the largest dataset which consists of 2,000 video clips captured by surveillance cameras in real-world scenes. Each video file is a 5-second video clip with 30 fps. Half of the videos contain violent behaviors, while others belong to non-violent actions. All videos in this dataset are captured by surveillance cameras in the real world, none of them are modified by multimedia technologies. Unlike ordinary action video datasets, where there are only one or two people, these violent video datasets usually have a lot of characters in each video and they are not fixed. The background information is complicated and the dynamic information of each character is also more changeable.

Implementation Details: Our implementation is based on PyTorch deep learning framework. Under the setting of batch-size 8 and 2048 sampled points, dropout [16] is applied to the last global fully connected layer with a ratio of 0.4. We train our model from scratch for 200 epochs using SGD with momentum and a learning rate of 10-3 on 4 Nvidia 2080Ti GPUs. Empirically, we set the layer number of our SPIL modules to 3 and $T_{frame} = 5$. Augmentation is used during the training with randomly jittering and rotating($\pm 10\%$). For the baseline of our skeleton point convolution network, we give a vanilla model without using SPIL and multi-head mechanism. In other words, we simply use the feature information of the points and single header, ignoring the position information, and then use the multi-layer perceptrons to perform operations. We run all experiments four times with different random seeds and report mean accuracies. For simplicity, we define the method using Eq 4 as **SPIL-Space**, and **SPIL-Span** represents using Eq 5 while **SPIL-Mask** implies Eq 6's stratgey.

4.1 Ablation Study

To show the impact of the SPIL module and the hyper-parameter in our network, we conduct several ablation studies on the RWF-2000 dataset. First, we compare the baseline model with the SPIL convolution of the three discussed solutions we designed. Among them, we all use a single head method for a fair comparison. At the same time, we set the default value as 0.02 in the hyper-parameter of the threshold d in Eq 6.

Method	Baseline	SPIL-Space	SPIL-Span	SPIL-Mask
Accuracy(%)	84.3	86.4	88.0	**88.7**

Table 1. Exploration of different skeleton points interaction learning strategies.

As shown in table 1, In the case of point convolution operation without using SPIL to extract both feature and position information of points, our baseline has an accuracy of only **84.3%**. After using our proposed SPIL module, base on a single head, all methods outperform the based model, demonstrating the effectiveness of modeling interaction weights between points. And the **SPIL-Mask**

Num-head	1	2	4	8	16	32	64
SPIL-Mask (%)	88.7	88.9	89.0	**89.2**	89.1	88.9	89.0
Baseline (%)	84.3	84.6	**84.8**	84.7	84.7	84.6	84.6

Table 2. Exploration of number of multi-heads.

yields the best accuracy with **88.7%** than the other two ways. We conjecture that the function of the mask can filter out some redundant information and make the information more stable. In the rest of the paper, we choose **SPIL-Mask** to represent our main method.

We also reveal the effectiveness of building a multi-head mechanism to capture diverse types of related information. As depicted in table 2, we compare the performance of using different numbers of heads in baseline and our model, which the results indicate that multi-head mechanism can improve both methods in different degrees. When head number is set to 8, **SPIL-Mask** is able to further boost accuracy from **88.7%** to **89.2%**. Meanwhile, too many heads will lead to redundant learning features and will increase computational costs.

Eu-Distance	$d_1 = 0.01$	$d_2 = 0.02$	$d_3 = 0.04$	$d_4 = 0.08$
Accuracy(%)	88.7	89.2	**89.3**	89.0

Table 3. Exploration of different distance threshold d in Eq 6.

Furthermore, we also implement experiments to reveal the effect of the hyper-parameter threshold of d on network performance. As shown in table 3, We see that when $d = 0.04$ achieves the best performance approaching **89.3%**. We conjecture that a too-small threshold will cause network information to be lost, and a too-large threshold will cause a negative effect of excess network information. Thus, we adopt head number H= 8, d= 0.04 in the following experiments.

4.2 Comparison with the State of the Art

We compare our approach with state-of-the-art approaches. Here, we adopt several models that perform well in traditional video action recognition tasks, and apply them to the violent video recognition tasks in this paper (they all are pre-trained on Kinetics [18]). At the same time, we also compared the excellent models in processing 3D point clouds tasks with our method.

As shown in table 4 on RWF-200 dataset, above the solid line it can be seen that although these methods reach the state-of-the-art level in general video action recognition tasks, their best results can only reach **87.3%** in violent video recognition which is 2% lower than our proposed method with accuracy in

Method	Core Operator	Accuracy(%)
TSN [33]	Two-Stream	81.5
I3D [4]	Two-Stream	83.4
3D-ResNet101 [14]	3D Convolution	82.6
ECO [42]	3D Convolution + RGB	83.7
Representation Flow [23]	Flow + Flow	85.3
Flow Gated Network [5]	3D Convolution + flow	87.3
PointNet++ [25]	Multiscale Point MLP	78.2
PointConv [36]	Dynamic Filter	76.8
DGCNN [34]	Graph Convolution	80.6
Ours	**SPIL Convolution**	**89.3**

Table 4. Comparison with state-of-the-art on the RWF-2000 dataset.

89.3%. This shows that traditional action video recognition methods lack the ability to extract the dynamic characteristics of people in the violence videos and the long-term correlation performance in each frame. At the same time, in violent videos, due to the diversity of characters and the variety of scenes, as a result, these methods rely on information such as optical flow information, global scene characteristics are invalidated, resulting in low recognition results. When we use human skeleton points to identify action characteristics, previous point clouds methods are not targeted and are not sensitive to the skeleton points, therefore, the recognition results are not accurate enough. By learning different skeleton points interactions through our method, we can reach the leading level.

Method	Hockey-Fight(%)	Method	Crowd (%)
3D CNN [9]	91.0	3D CNN [9]	-
MoWLD + BoW [40]	91.9	MoIWLD + KDE [39]	93.1
TSN [33]	91.5	TSN [33]	81.5
I3D [4]	93.4	I3D [4]	83.4
ECO [42]	94.0	ECO [42]	84.7
Representation Flow [23]	92.5	Representation Flow [23]	85.9
Flow Gated Network [5]	**98.0**	Flow Gated Network [5]	88.8
PointNet++ [25]	89.7	PointNet++ [25]	89.2
PointConv [36]	88.6	PointConv [36]	88.9
DGCNN [34]	90.2	DGCNN [34]	87.4
Ours	96.8	**Ours**	**94.5**

Table 5. Comparison with state-of-the-art on the Hockey-Fight and Crowed Violence dataset.

We further evaluate the proposed model on the Hockey-Fight and Crowd Violence dataset. As shown in table 5, specifically, in terms of accuracy, the average level on the crowd dataset is lower than the hockey dataset. This is

because the crowd dataset has more people and its background information is relatively complicated. Therefore, some previous work is difficult to have a high degree of recognition. Our SPIL module is not affected by complex scenes. It obtains features by extracting the skeleton point clouds information of the people, which can achieve a high accuracy rate. This outstanding performance shows the effectiveness and generality of the proposed SPIL for capturing the related points information in multiple people scene.

Method	Accuracy(%)
Extreme Acceleration [8]	85.4
TSN [33]	94.2
I3D [4]	95.8
3D-ResNet101 [14]	96.9
ECO [42]	96.3
Representation Flow [23]	97.3
PointNet++ [25]	89.2
PointConv [36]	91.3
DGCNN [34]	92.6
Ours	**98.5**

Table 6. Comparison with state-of-the-art on the Movies-Fight dataset.

Finally, we validate our model on the Movies-Fight dataset. As can be seen in table 6, our method can outperform the existing methods. Under different datasets and different environmental scenarios, our method does not rely on other prior knowledge and will not overdone in one dataset and causes inadaptability in other scenarios.

4.3 Failure Case

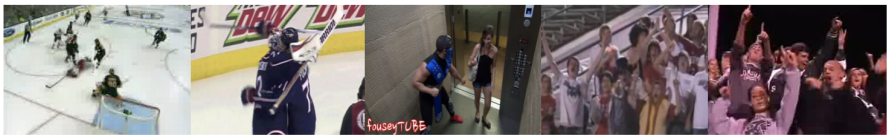

Fig. 5. From left to right: "hockey ball scrambling", "hug", "dance", "high five" and "wave".

To further analyze the network effect, we study the misclassification results of the above four datasets. We found that some similar to violent actions are

misclassified as violence. As shown in Fig 5, some movements in the picture that have physical contact, models consider they are fighting or some other violence. Therefore, we collect all these easily confused samples from 4 datasets evenly and formed them into a new small dataset for classification.

Fig 6 shows the confusion matrix comparison. In those actions that are very similar to violence but are not actual acts of violence, compared with the traditional video action recognition technique, our method can have a large degree of discrimination to classify these actions. The above results illustrate that the proposed network is an effective method for video violence recognition. Essentially paying more attention to the information about the skeletal modality movements of the characters will help to distinguish the behaviors of the characters.

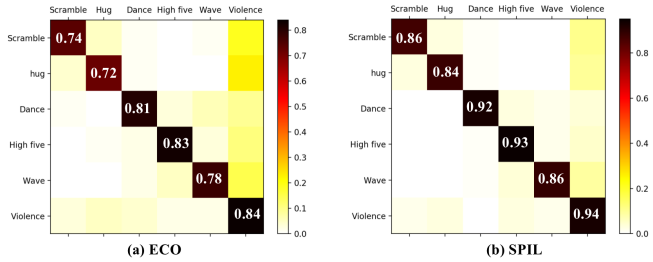

Fig. 6. Confusion matrix comparison on the confused sample. (a) ECO [42] method. (b) Our SPIL method.

5 Conclusion

In this paper, we propose a novel and effective approach for video violence recognition. To the best of our knowledge, we are the first one to solve this task by using a 3D point clouds technique to extract action feature information of human skeleton points. By introducing the Skeleton Points Interaction Learning (SPIL) module, our model is able to assign different weights according to different skeleton points to obtain the motion characteristics of different people. Furthermore, we also design a multi-head mechanism to process different types of information in parallel and eventually aggregate them together. The experiment results on four violent video datasets are promising, demonstrating that our proposed network outperforms the existing state-of-the-art violent video recognition approaches.

Acknowledgment. This work was supported by NSFC 61876208, Key-Area Research and Development Program of Guangdong 2018B010108002, National Research Foundation Singapore under its AI Singapore Programme (AISG-RP-2018-003) and the MOE Tier-1 research grants: RG28/18 (S) and RG22/19 (S).

References

1. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P.: Geometric deep learning: going beyond euclidean data. IEEE Signal Processing Magazine **34**(4), 18–42 (2017) 4
2. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: European conference on computer vision. pp. 25–36. Springer (2004) 1
3. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017) 2, 3
4. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017) 3, 12, 13
5. Cheng, M., Cai, K., Li, M.: Rwf-2000: An open large scale video database for violence detection. arXiv preprint arXiv:1911.05913 (2019) 2, 10, 12
6. Christoph, R., Pinz, F.A.: Spatiotemporal residual networks for video action recognition. Advances in Neural Information Processing Systems pp. 3468–3476 (2016) 3
7. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems. pp. 3844–3852 (2016) 4
8. Deniz, O., Serrano, I., Bueno, G., Kim, T.K.: Fast violence detection in video. In: 2014 international conference on computer vision theory and applications (VISAPP). vol. 2, pp. 478–485. IEEE (2014) 13
9. Ding, C., Fan, S., Zhu, M., Feng, W., Jia, B.: Violence detection in video by using 3d convolutional neural networks. In: International Symposium on Visual Computing. pp. 551–558. Springer (2014) 12
10. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2625–2634 (2015) 2
11. Fang, H.S., Xie, S., Tai, Y.W., Lu, C.: Rmpe: Regional multi-person pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2334–2343 (2017) 4
12. Garcia-Garcia, A., Gomez-Donoso, F., Garcia-Rodriguez, J., Orts-Escolano, S., Cazorla, M., Azorin-Lopez, J.: Pointnet: A 3d convolutional neural network for real-time object class recognition. In: 2016 International Joint Conference on Neural Networks (IJCNN). pp. 1578–1584. IEEE (2016) 3
13. Gehring, J., Auli, M., Grangier, D., Dauphin, Y.N.: A convolutional encoder model for neural machine translation. arXiv preprint arXiv:1611.02344 (2016) 4
14. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6546–6555 (2018) 3, 12, 13
15. Hassner, T., Itcher, Y., Kliper-Gross, O.: Violent flows: Real-time detection of violent crowd behavior. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. pp. 1–6. IEEE (2012) 2, 9
16. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580 (2012) 10

17. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1725–1732 (2014) 3
18. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017) 11
19. Lev, G., Sadeh, G., Klein, B., Wolf, L.: Rnn fisher vectors for action recognition and image annotation. In: European Conference on Computer Vision. pp. 833–850. Springer (2016) 3
20. Niepert, M., Ahmed, M., Kutzkov, K.: Learning convolutional neural networks for graphs. In: International conference on machine learning. pp. 2014–2023 (2016) 4
21. Nievas, E.B., Suarez, O.D., Garcia, G.B., Sukthankar, R.: Hockey fight detection dataset. In: Computer Analysis of Images and Patterns. pp. 332–339. Springer (2011) 2, 9
22. Nievas, E.B., Suarez, O.D., Garcia, G.B., Sukthankar, R.: Movies fight detection dataset. In: Computer Analysis of Images and Patterns. pp. 332–339. Springer (2011) 9
23. Piergiovanni, A., Ryoo, M.S.: Representation flow for action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9945–9953 (2019) 12, 13
24. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017) 4
25. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in neural information processing systems. pp. 5099–5108 (2017) 2, 4, 12, 13
26. Riegler, G., Osman Ulusoy, A., Geiger, A.: Octnet: Learning deep 3d representations at high resolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3577–3586 (2017) 3
27. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems. pp. 568–576 (2014) 1
28. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1746–1754 (2017) 3
29. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015) 2, 3
30. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6450–6459 (2018) 2
31. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017) 4
32. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016) 1, 3
33. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016) 12, 13

34. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (TOG) **38**(5), 1–12 (2019) 2, 12, 13
35. Wu, J., Wang, L., Wang, L., Guo, J., Wu, G.: Learning actor relation graphs for group activity recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9964–9974 (2019) 6
36. Wu, W., Qi, Z., Fuxin, L.: Pointconv: Deep convolutional networks on 3d point clouds. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9621–9630 (2019) 2, 12, 13
37. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1492–1500 (2017) 3
38. Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4694–4702 (2015) 2, 3
39. Zhang, T., Jia, W., He, X., Yang, J.: Discriminative dictionary learning with motion weber local descriptor for violence detection. IEEE transactions on circuits and systems for video technology **27**(3), 696–709 (2016) 12
40. Zhang, T., Jia, W., Yang, B., Yang, J., He, X., Zheng, Z.: Mowld: a robust motion image descriptor for violence detection. Multimedia Tools and Applications **76**(1), 1419–1438 (2017) 12
41. Zhang, Y., Rabbat, M.: A graph-cnn for 3d point cloud classification. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6279–6283. IEEE (2018) 4
42. Zolfaghari, M., Singh, K., Brox, T.: Eco: Efficient convolutional network for online video understanding. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 695–712 (2018) 12, 13, 14