Multi-View and Multi-Modal Action Recognition with Learned Fusion

Sandy Ardianto[1] and Hsueh-Ming Hang[2]
[1,2]College of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu, Taiwan, ROC
E-mail: sandyardianto.03g@g2.nctu.edu.tw, hmhang@mail.nctu.edu.tw

***Abstract*— In this paper, we study multi-modal and multi-view action recognition system based on the deep-learning techniques. We extended the Temporal Segment Network with additional data fusion stage to combine information from different sources. In this research, we use multiple types of information from different modality such as RGB, depth, infrared data to detect predefined human actions. We tested various combinations of these data sources to examine their impact on the final detection accuracy. We designed 3 information fusion methods to generate the final decision. The most interested one is the Learned Fusion Net designed by us. It turns out the Learned Fusion structure has the best results but requires more training.**

***Keywords*—human action recognition, neural nets, deep learning, multi-view video, multi-modal video, information fusion**

I. INTRODUCTION

In this research, we try to recognize human action based on the multi-view and multi-modal information. First, we use standard RGB video input system as the base model, then we develop more complicated algorithms to take advantages of multi-view and multimodal information. Four modalities will be used in our study, namely RGB, depth, infrared and optical flow. All except optical flow are provided by the dataset. Our experimentviews show that multiple modalities can can improve the accuracy. We hope to achieve high accuracy and do not need to compromise in speed, which is essential for real-time application. We first choose the Temporal Segment Networks (TSN) as our starting point. We modify it to meet our objectives. Multi-modalities and multi-view information fusion is handled by adding one learned fusion layer at the end of multi-view TSN.

II. RELATED WORKS

Public space like airport, shopping mall, train station already having many surveillance cameras installed. However, the human operator needed to watch them in order to detect a threat. A human can get tired and may miss some threats. Casualties caused terrorist attack can be minimalized if it can be detected earlier. Deep learning becomes a popular and accurate technique in this decade to detect and classify objects such as human faces and pre-defined objects. It can also be used to identify human actions. Many research groups are actively developing action recognition techniques, such as ActivityNet [1], an annual action recognition competition. There are also researchers working on the multi-view and multi-modal dataset such as JCRRNN [2] and skeleton based CNN [3]. Most of them did not use all the available modalities to detect the human action. Thanks to the availability of Microsoft Kinect devices, we can relatively easily obtain the skeleton information. Using skeleton information for action recognition thus becomes another popular method.

III. DATASET: PKU-MMD

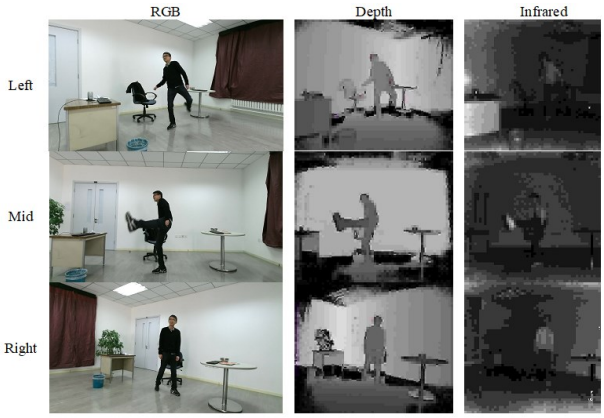

Figure 1. PKU-MMD sample images

PKU-MMD [4] is a multi-view, multimodality dataset. Researchers at Peking University captured human action videos using 3 Microsoft Kinect version 2. This dataset has 51 action classes, which include 41 daily actions (drinking, waving hand, putting on the glass, etc.), and 10 interaction actions (hugging, shaking hands, etc.). Since it was captured using Microsoft Kinect version 2, there are four modalities available in this dataset, namely, (1) RGB videos with a resolution of 1920 x 1080, (2) human skeleton joint information stored in 3-dimensional positions of 25 major body joints, (3) depth maps and (4) infrared at 512 x 424 resolution. In Figure 1, the depth and infrared images were preprocessed using histogram equalization just for display purpose. In the website announcement, there are two phases in constructing this dataset. However, by the time of writing, only Phase #1 is published. Phase #1 provides 1076 long video sequences with 51 action categories, performed by 66 subjects. Each video has 3-4 minutes duration. Phase #2 plans have 2000 short video sequences with 49 action categories, performed by 60 subjects.

Each video in that phase will have 1-2 minutes duration. In this study, we use the available Phase #1 data only.

IV. Adopted Approaches

Many machine/deep learning techniques have been proposed for action recognition, such as temporal segment networks [5], long-term recurrent convolutional networks [6], and 3D convolutional networks [7]. After surveying through the documents, we adopt the temporal segment network (TSN) proposed in [5] as the basis algorithm in this study. One of the motivations is that this approach was the winner of the Activity Net [1] action recognition competition in 2016, and it got the second place in 2017, although it uses the CNN structure instead of RNN. Since our target is using the multi-view and multi-modal information, we modify TSN to make use of the full advantages of multimodal data. It is expected that the multimodal dataset can produce more robust recognition results if they are properly used.

A. Temporal Segment Networks

The main idea of Temporal Segment Networks (Fig. 2) is to split the video input into several segments, then use them as input to the CNN. In their original work, the authors use RGB video and the optical flow information derived from the video. The target class was identified by combining the outputs from multiple CNNs using the segment consensus. In the segment consensus, they tried several combinations such as maximum, average, and weighted average pooling. The best result was achieved using the average pooling that has a 1-2% accuracy advantage. They also tried different number of segments, such as 3, 5, 7, and 9. However, the number of segments only affect less than 1% on the accuracy. Based on their experiments, we adopt the maximum pooling for segment consensus and divide the input videos into 3 segments. The reason is that we intend to implement it for real-time action detection. The maximum pooling is faster to compute than the average pooling. The number of CNNs matches the number of segments. Thus, increasing the number of segments also increases the CNN computation.

Temporal Segment Network (TSN) [5] adopts a modified version of InceptionNet [8] for the RGB and the optical flow CNN. This network contains 9 sub-networks, and each sub-net has 2x4 convolution layers, and 2 pooling layers. The output of a sub-network goes to the concat/normalize layer, then it becomes the input the next sub-network. There is softmax layer in the last stage of 4th, 7th, and 9th sub-network. The 4th and 7th softmax layers act as auxiliary classifiers, and the 9th softmax layer provides the output classifier. The optical flow information was generated from the RGB video based on the motion vectors produced by the TVL1 optical flow algorithm [9].

B. Multi-View Temporal Segment Networks

As mentioned in the introduction section, our target dataset has multi-view information. Therefore, we expand the TSN network into the multi-view TSN as shown below (Fig.4). We use one TSN for each view, and the output target class comes from a weighted fusion of all the TSN outputs. We first use the simple signal combination approach. We simply add an addition or max pooling layer, which collects information from all views. According to our experiments, different methods of fusion did not affect the accuracy much (around 1% accuracy).

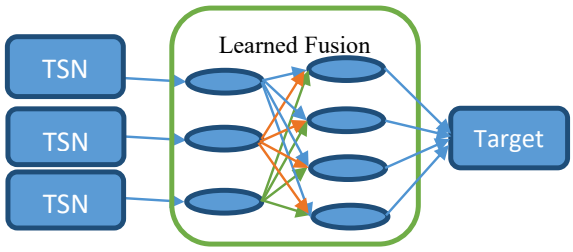

Figure 2. Multi-view temporal segment network architecture

Next, we proposed a learned weight approach, a neural network with one hidden layer was used to determine the weights among various views. The (classification) output from each TSN is the input to the Fusion Neural Network. There are two ways to train the weights. (1) We first train the TSN separately and the simply train the weights in the fusion net using the trained (fixed) TSN. (2) The other way is to train the weights along with the TSNs. At the moment, we have implemented only the first option because the second one needs more memory and computing power. Since one of our aims is to build a real-time action recognition, we choose a faster and yet a robust system.

In total, we tested 3 fusion methods: summation, max-pooling, and learned fusion. In the experimental results section, we will show the comparison results among them. The summation method adds up the probability of each class from three views (left, middle, right) as below.

$$c_{summation} = c_{left} + c_{mid} + c_{right}, \quad (1)$$

where c is the output class. In the summation method, we choose the highest summed probability among all the classes. For example, in Table I, the highest value of summation is the 'walking' action by summing the probability of each view. It is 2.64 (0.89+0.90+0.85). The 'Running' action has the sum of 2.61 (0.90+0.91+0.80), and the 'stand-up' has the lowest probability (1.36 = 0.32+0.92+0.12).

We also tried the max-pooling on all probabilities coming from each TSN as below.

$$c_{max-pooling} = max(c_{left}, c_{mid}, c_{right}) \quad (2)$$

This particular method has a weakness when one false negative class has very high probability, as shown in the example below. The final max-pooling result simply selects the maximum probability of each class. For example, in Table I, the highest probability is 0.92 from the stand-up action in the mid-view.

TABLE 1. FUSION METHOD EXAMPLES

ACTIONS	LEFT	MID	RIGHT	SUM-MATION	MAX-POOLING
RUNNING	**0.90**	0.91	0.80	2.61	0.91
WALKING	0.89	0.90	**0.85**	**2.64**	0.90
STAND-UP	0.32	**0.92**	0.12	1.36	**0.92**

Table 1 shows that different fusion methods give different results. The summation method votes 'walking', and the max-pooling picks 'stand-up' as the final output. This example shows that the fusion method needs to be carefully designed to produce the desirable result. When a person does 'slow jogging', it can be perceived as 'stand-up' from the middle camera, and 'running' or 'walking' from either left or right camera. A properly crafted weighted fusion scheme may solve this problem by giving more weight to the trusted source.

C. *Multi-Modal Temporal Segment Networks*

Our testing dataset was captured using Microsoft Kinect. Hence, there are four different modalities available, RGB, depth, infrared, and skeleton information. Since computing the optical flow information takes a lot of computing power and time, we do not include it in our networks. We intend to design real-time action detection system, and thus the schemes with time consuming calculation is considered as low priority.

Each of modalities has its own CNN. In this moment, we have not included the skeleton information into our processing yet, partially because the input of InceptionNet is an image. Typically, the skeleton information is stored in the 3-dimensional format, which needs to be converted into 2D images.

V. EXPERIMENTAL RESULTS

TSN originally had two inputs, namely, RGB and optical flow videos. Now, we expand TSN to support multi-view and multi-modal dataset. As Table II shows, our proposed methods can achieve 89.2% accuracy when using the RGB, infrared, and depth videos from each view on PKU-MMD. If we restrict to the use of the RGB information only, it can achieve around 83% accuracy. But using only the depth or the infrared inputs can lead to about 75% accuracy. If we combine depth and infrared together, the middle view reaches 86.3% accuracy.

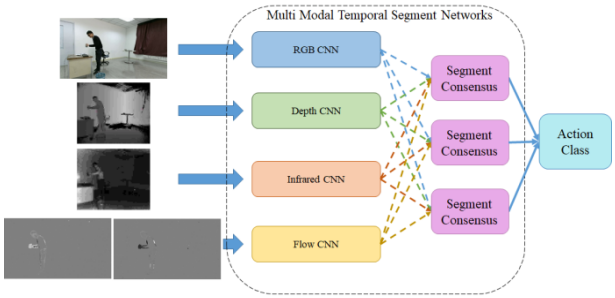

Figure 3. Multi-modal temporal segment networks architecture

Our training time is 12 hours for RGB, and 8 hours for depth and infrared, separately. First, we train the TSN with UCF101 dataset, then the trained model from UCF101 is used as the pre-trained model for the RGB CNN. The RGB CNN trained model is used as the pre-trained model for the depth and infrared CNN. The CNN used in depth and infrared images have the same structure as the RGB one. PKU-MMD has a total of 1,076 videos (each video has three views). We divide it into 944 videos for training, and 132 videos for testing. Since one video may contain more than one action, one segment of video was trimmed for each action based on the ground-truth file. Each of CNN was trained separately for each modalities and views. The experiments were done using a single NVIDIA Titan X GPU with 12GB memory. The segment number used in this experiments is 3 (similar to the original work on the Temporal Segment Networks by [5]).

TABLE II. MULTI VIEW AND MULTI MODAL ACCURACY RESULTS ON PKUMMD

VIEW	RGB	DEPTH	IR	DEPTH +IR	RGB +DEPTH	RGB +IR	RGB +IR +DEPTH
LEFT	80.2	76.3	72.2	80.5	81.1	80.3	**81.9**
MIDDLE	85.3	78.6	76.5	**86.3**	85.4	85.0	85.5
RIGHT	82.9	75.8	75.7	82.3	**85.3**	83.3	84.8
MULTI VIEW	85.3	78.2	73.2	84.9	88.1	86.1	**89.2**

We find that the right view has slightly better accuracy than the left view (about 2% better). This may be due to that most actions on the dataset are using the right hand. Using all modalities raises around 5% accuracy. However, it makes the process more complicated and time-consuming. Adding the optical flow data into our classifier increases around 8% accuracy on average.

Table III examines the optical flow data working together with other information sources. It can produce 89.2% accuracy just using the flow information alone, which is somewhat higher than the use of RGB, IR, and depth data alone. Combining flow with other modalities is able to increase 10% or more on the depth and the infrared data. This result can be interpreted that some modalities have more useful information for action recognition than the others. The optical flow seems beneficial to action recognition. However, computing optical flow is complex and time-consuming. The Max-pooling was used in the output fusion here, since it is faster than the other methods in computing complexity.

TABLE III. ADDING FLOW FEATURES IN MULTI-VIEW AND MULTI-MODAL ON PKU-MMD

VIEW	RGB	DEPTH	IR	FLOW	DEPTH +IR	RGB +DEPTH	RGB +IR	RGB +IR +DEPTH
MULTI VIEW	85.3	78.2	73.2	89.2	84.9	88.1	86.1	89.2
+FLOW	92.3	90.7	89.6	-	91.6	91.4	90.8	**94.6**
AVG: +8%	+7%	+12.5	+16.4	-	+6.7	+3.3%	+4.7	+5.4

In Section IV.B, we discussed 3 types of fusion methods to combine all gathered information (including optical flow) from different views. Finally, we can further improve our result by modifying the fusion used in the experiments. The learned fusion method can achieve 95.2% accuracy (0.8% better than the Max-pooling). The learned fusion method is the neural network with one hidden layer.

TABLE IV. WEIGHT FUSION COMPARISON RESULTS

METHOD	ACCURACY(%)
SUMMATION	93.2
MAX-POOLING	94.6
LEARNED FUSION	**95.2**

TABLE V. COMPARISON WITH OTHER WORKS ON PKU-MMD

METHOD	DATA TYPE	ACCURACY
MULTI VIEW TEMPORAL SEGMENT NETWORKS	RGB+DEPTH+IR+FLOW	95.2%
*TEMPORAL SEGMENT NETWORKS [5]	RGB+FLOW	85.3%
JCRRNN [2]	RGB	69.9%
CNN [3]	SKELETON JOINT	**95.8%**

*TESTED USING THEIR SOURCE CODE ON SINGLE VIEW ONLY

In Table V, we compare our approach with the other works using different data types on PKU-MMD. Our approach, the multi-view Temporal Segment Network, outperforms most of the other methods on the average accuracy (95.2%). The original temporal segment network is able to achieve 85.3% accuracy using the RGB and optical flow information. The JCRRNN method which uses RGB videos only is able to achieve 69.9%. This value is lower than our method when using the same RGB information only. Compared to the other works, the CNN approach proposed by [3] accomplishes the highest accuracy on PKU-MMD [4]. Their network input using only the skeleton information and is able to produce 95.8% accuracy. Therefore, in our future work, we plan to experiment on the skeleton based action recognition.

VI. CONCLUSIONS

In this study, we implemented the Temporal Segment networks and extended it to the multi-modal and multi-view dataset. Our proposed multi-modal system can achieve 95.2% accuracy on the PKU-MMD dataset. We observe that some views provide more information than the other views. In our experiments on PKU-MMD, the middle view has the highest accuracy than the other two. Also, the right view has slightly better accuracy than the left view. Finally, the multi-view data can improve the recognition performance for action recognition comparing to the single view but the difference is not very large. In the multi-modal experiments, we can see that combining different modality can improve the accuracy. Using only the RGB videos, we achieve 85.3% accuracy. Combining RGB with infrared and depth, only increases 1% accuracy. Nonetheless, when we include all of them, the accuracy raises around 4-5%. The optical flow data seems to be very informative. It alone can outperform several combined multimodal cases but its needs to calculate the optical flow maps. Based on this experiments, we can conclude that if the multi-view and multi-modal information is available, with carefully selected few modalities, we can develop a much faster action recognition system with a bit of compromise, say 5%, on accuracy.

ACKNOWLEDGMENT

This research was supported by Ministry of Science and Technology (MOST), Taiwan under Grant MOST 106-2221-E-009-125-MY3.

REFERENCES

[1] F. C. Heilbron, V. Escorcia, B. Ghanem and J. C. Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Computer vision and pattern recognition*, 2015.

[2] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan and J. Liu, "Online human action detection using joint classification-regression recurrent neural networks," in *European Conference on Computer Vision (ECCV)*, 2016.

[3] C. Li, Q. Zhong, D. Xie and S. Pu, "Skeleton-based action recognition with convolutional neural networks," in *International Conference on Multimedia & Expo Workshops (ICMEW)*, 2017.

[4] C. Liu, Y. Hu, Y. Li, S. Song and J. Liu, "PKU-MMD: A large scale benchmark for continuous multi-modal human action understanding," in *arXiv preprint:1703.07475*, 2017.

[5] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European Conference on Computer Vision*, 2016.

[6] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko and T. Darrel, "Long-term recurrent convolutional networks for visual recognition and description," in *computer vision and pattern recognition*, 2015.

[7] T. Du, L. Bourdev, R. Fergus, L. Torresani and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Intenational Conference on Computer Vision (ICCV)*, 2015.

[8] I. Sergey and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, 2015.

[9] C. Zach, T. Pock and H. Bischof, "A duality based approach for realtime TV-L 1 optical flow," in *Joint pattern recognition symposium*, Berlin, Heidelberg, 2007.

[10] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen and X. Xie, "Co-Occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *AAAI*, 2016.