Continuous Tracks CNN and Non-local Gating for Multi-class Video Understanding

Youngjae Yu
RippleAI & SNU
Seoul, Korea
yj.yu@vision.snu.ac.kr

Hyeokjun Kwon
SNU
Seoul, Korea
emckwon@gmail.com

Hankyol Lee
RippleAI
Seoul, Korea
hklee@rippleai.co

GunheeKim
RippleAI & SNU
Seoul, Korea
gunhee@snu.ac.kr

Abstract

We present (1) a new CNN architecture named CT-CNN and (2) Non-local Gating ensembles that can infer the multiple actions in short clip videos. In order to learn effective multi-label actions for the video clip, our model aggregates slow and fast network [2] informations and the Non-local Gating inference. Thus, our model achieves more accurate final class confidence prediction of each segments in video. We ensembled multiple predictions of Video CNN models including ours, and 12 variants of Nonlocal Gating layers. We participate in the first Multi Moment in Time challenge [10, 9] in ICCV 2019, for which ensemble of our model achieves one of the best performances.

1. Challenge Introduction

Multi-Moments in Time Challenge 2019 presents a multilabel extension to the Moments in Time Dataset [10, 9] which includes annotation of multiple actions in each video. The goal of this challenge is to detect multiple event labels depicted in a 3 second video clip.

2. Approach

2.1. Preprocessing

Video Frame The size of each frame of video frame data consisting of 3 channels of RGB was all resized in 128 by 128. Resized video goes through the input of the video cnn after some processing in the data loader module. Since the number of frames/fps of each video is not constant, we proceeded to correct this. the temporal depth of the input into the CNN is fixed between 16 and 64. At this time, if the total number of frames is larger than the fixed temporal depth, we uniformly sample frames from the entire video. After adjusting the temporal depth, random cropping was performed so that each frame had a height and width of 112 for data augmentation. In the training session, additional horizontal random flipping was performed.

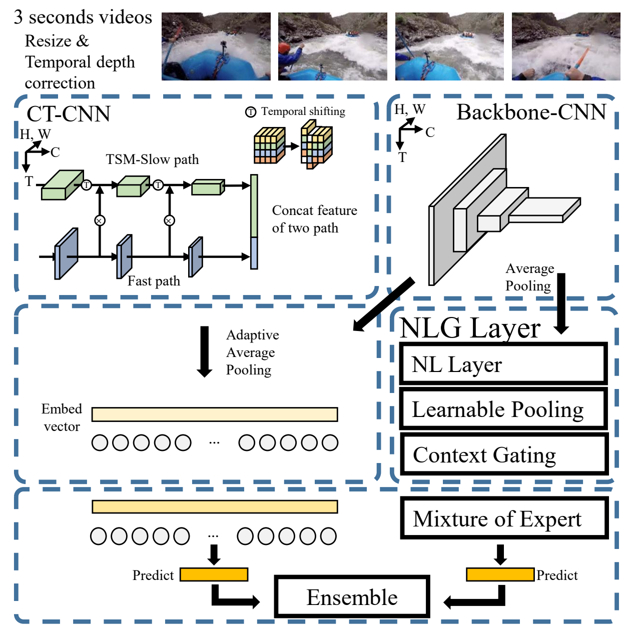


Figure 1. The intuition of the our final model. Our model is easily adaptable to many video multi-label prediction tasks.

Between-Class Learnning. We applied the idea of between-class learning [13] To do this, the data loader loads two resized videos, matches temporal depth and spatial dimensions, and then mixes them frame-wise at a random rate. In the same way, the labels of two videos are also class-wise mixed at the same rate as the video, producing a real value between 0 and 1. We then learn these values as labels for the video that are frame-wise mixed.

Video Sound. Sound waveforms are obtained from the video data. The waveform was used to train the Envnetv2 model [12].

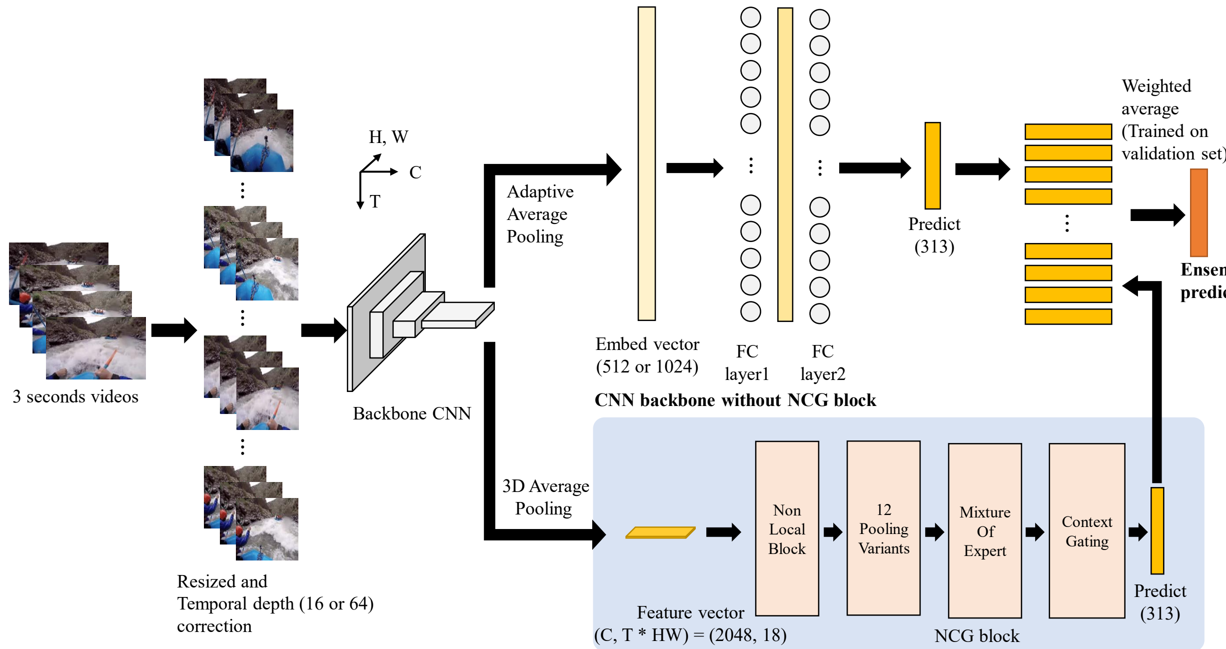


Figure 2. The architecture of our ensembled model. We omit some fully-connected layers for visualization purpose.

2.2. Video CNN Backbones

We extract feature maps from CNN as the backbone of a custom multilabel prediction layer variant, including the NonLocal Gating layer.

CT-CNN : Temporal Shift for Slow Fast Network. As one of the backbones for processing video data, we used variant of SlowFast network [2]. SlowFast network is a video CNN based on ResNet in two paths with different sampling rates of frames. Fast path concentrates on temporal features at high frame sampling rates, and slow path concentrates on spatial features at low frame sampling rates. We implement variant of SlowFast network so that spatial features flows across slow path with Temporal Shift Module layers [6].

Pretrained CNN We use R(2 + 1)D-152 [14] pre-trained with IG-65M [3] as backbone.

2.3. Video Nonlocal Gating

We add NonLocal layer [15] after CNN feature map and following context gating layer [8] for multi-label prediction. Putting learnable pooling before context gating [8] generally helps to improve performance. For example, we denote NLG(NVLAD) as sequential layers of [NonLocal Layer [15], NetVLAD [1], MLP+ContextGating [8]]

2.4. Training

BCE Loss The CNN prediction value is passed through the sigmoid function to calculate Binary Cross Entropy(BCE) loss with ground truth label for 313 classes and to train the model in the direction of minimizing it. BCE loss was used by default in all cnn backbone and fully-connected layers and in Envnetv2 training sessions. We use batch shuffling in every training epoch.

LSEP Loss We use LSEP(log-sum-exp pairwise) loss and threshold estimation for each classes [5] to train CT-CNN model. LSEP loss is differentiable and smooth everywhere, which make model easier to optimize. The estimated thresholds of each classes are used to calculate another binary cross entropy loss, L_{thresh}. L_{thresh} can be calculated as BCE loss but, its input logits are sigmoid activation of difference between confidence of each classes and estimated thresholds.

We use the SGD-momentum optimizer. We set the Initial learning rate as $lr = 0.001$ in our experiments. For regularization, we apply batch normalization [4], and use dropout [11] after dense layers.

3. Experiments

We report the experimental results of our models for the Multi-Moment in Time challenge challenge. The challenge provides a multi-class video data set that includes 313 action classes, 1M training labels for 1M video, 10K valida-

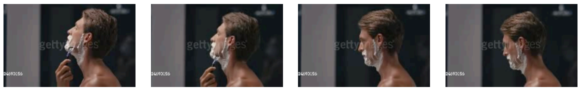
Correct Examples **Wrong Examples**

Pred : Floating(218)-Boating(37)-Rafting(201)-Rowing(289)-Flowing(304)-Paddling(1)

(a)

Pred : Baking(120)-Descending/Lowering(147)-Jumping(170)-Skipping(211)-Leaping(87)-Bouncing(189)

(b)



Pred : Cutting(246)-Grooming(216)-shaving(80)-Trimming(144)-Clipping(234)-Removing(293)

(c)

Pred : Throwing(84)-Pitching(89)-Playing sports(298)-Competing(278)-Officiating(302)-Catching(224)

(d)

Figure 3. Qualitative examples of the multi-label action recognition task. The left column shows correct examples, while the right column shows wrong examples. In each case, we show our best scored 6 predictions in descending order. We denote each action class as Name of lable(index of label).

tion video, and 10K test video. mAP (mean average precision) on the testing set is the official metric for this challenge. We strictly follow the evaluation protocols of the challenge. We defer more details and challenge rules to the challenge homepage[1].

3.1. Qualitative Results

Figure 3 illustrates qualitative results of our ensemble model results with correct (left) or wrong (right) examples for each task.

3.2. Quantitative Results

Table 1–2 summarize the quantitative results of our experiments. Table 1 show the public validation results and ensemble weight for final submission. At Table 1 weights of model R(2+1)D is trained 2 more epoch than the other R(2+1)D models with NLG layer.

4. Conclusion

We proposed the NonLocal Gating model for 3D spatio-temporal video feature maps. We have observed performance improvement in ensemble of various NLGating + Pooling layers, even they based on the same CNN feature map. We plan to expand the applicability of the CT-CNN, NLG models; Unfortunately, it was not enough time to run a variety of experiments until the challenge due, but more sophisticated experiments will be updated to compare the various cnn models. Since our method is applicable to any multi-label prediction tasks, we plan to test our model on other large scale video datasets.

[1] http://moments.csail.mit.edu/challenge_iccv_2019.html.

Model	ensW	GAP
CT-CNN(no Pretrain)	0.1614	0.35082
R(2+1)D	0.2134	0.52823
CNN+NLGating	0.1793	0.45770
CNN+NLG(BLSTM) [1]	0.3968	0.46304
CNN+NLG(LSTM)	0.1507	0.43770
CNN+NLG(NeXTVLAD) [7]	0.3809	0.49452
CNN+NLG(NeXTVLAD(med)) [7]	0.5383	0.50102
CNN+NLG(NeXTVLAD(big)) [7]	0.6764	0.50879
Ensemble	–	0.78371

Table 1. Performance comparison for valdiation datset. ensW means ensemble weight

Model	mAP
TRN [10]	0.24
CT-CNN	0.32
CT-CNN+Envnetv2+R(2 + 1)D-152 [14]	0.47
Ours + NLGating Models (Ensemble)	0.4857

Table 2. Performance comparison for official test leaderboard.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, 2016.

[2] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2018.

[3] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, 2019.

[4] Sergey Ioffe and Christian Szegedy. Batch Normalization:

Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, 2015.

[5] Yuncheng Li, Yale Song, and Jiebo Luo. Improving pairwise ranking for multi-label image classification. *CVPR*, 2017.

[6] Ji Lin, Chuang Gan, and Song Han. Temporal shift module for efficient video understanding. In *ICCV*, 2019.

[7] Rongcheng Lin, Jing Xiao, and Jianping Fan. Nextvlad: An efficient neural network to aggregate frame-level features for large-scale video classification. In *ECCV*, 2018.

[8] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. *arXiv*, 2017.

[9] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *TPAMI*, 2019.

[10] Mathew Monfort, Kandan Ramakrishnan, Dan Gutfreund, and Aude Oliva. A large scale multi-label action dataset for video understanding. In *CCN*, 2018.

[11] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *JMLR*, 2014.

[12] Yuji Tokozume and Tatsuya Harada. Learning environmental sounds with end-to-end convolutional neural network. In *ICASSP*, 2017.

[13] Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. Learning from between-class examples for deep sound recognition. In *ICLR*, 2018.

[14] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[15] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018.