Title: Ensemble Transformer and CNN for endoscopic images recognition

Authors: Atsushi Yamada, Yuriko Harai, Yuto Ishikawa, Kazuyuki Hayashi.

Affiliations: Department of Medical Device Innovation, Office of AI Digital Device Innovation, National Cancer Center Hospital East.

Team name: NCC NEXT.

Do you agree to make your submission public as part of the challenge archive? Yes/No → **Yes.**

1. Introduction

Our team is one of the hospital's research teams, and we won the MICCAI CHALLENGE 2021 (PETRAW) last year. We participated this year again and worked on all challenges (action recognition, segmentation, and multitask approaches).

In action recognition task, we approached sequence-wise prediction. We used Video-Swin-Transformer[1] and SlowFast[2]. We aimed at ensemble effects between Transformer and Convolutional Neural Network (CNN). Our result of action recognition is 0.752. In segmentation task, we approached frame-wise prediction. We used Swin-Transformer[3], SegFormer[4] and OCRNet[5]. We ensembled because of former reason. Our result of segmentation is 0.796. In multitask, we approached independent task, so we used same model of action recognition and segmentation again. Our result of multitask is 0.774.

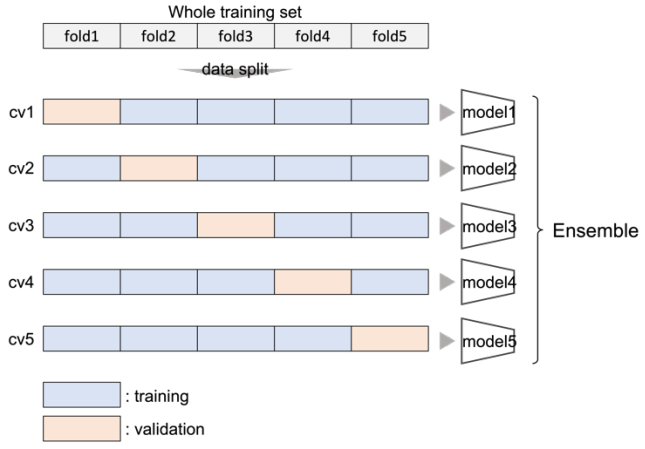

Figure 1. Cross validation ensemble. When testing, we ensembled each model.

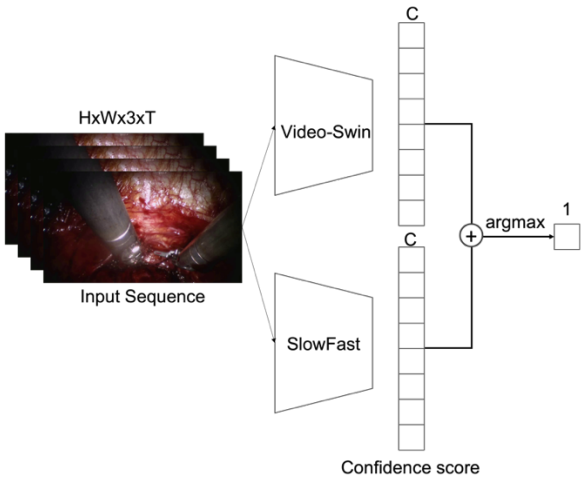

Figure 2. Architecture of action recognition task. H, W, T, C is height, width, number of temporal frames, number of classes, respectively. In this task, we set H is 256, W is 512, T is 6, C is 8.

2. Methods and results

Datasets preparation and how to test

We divided training dataset to 5 folds for cross validation. When dividing the data, the data was divided by video unit so that the amount of data in each class would be the same. We evaluate models by 5-fold cross validation (CV) score. At the test, each iteration model was ensembled by averaging (Figure 1). Image size of height and width size are resized to 256 and 512 pixels, respectively. Input images are normalized by dataset mean and standard deviation of RGB value.

2.1 Action Recognition task

We ensemble two models, Video-Swin-Transformer and SlowFast Network (Figure 2). These models are sequence frames input, so the sequence is made by taking a fixed number of frames as input and shifting them one frame at a time to make a sequence, which enables frame-by-frame prediction. We applied post-process that transition rules based to predictions (Figure 3).

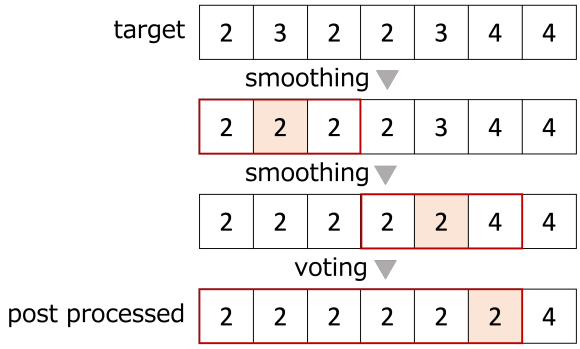

Figure 3. The transition rules. We applied voting with window size 15 after smoothing with window size 3 prediction.

Video-Swin-Transformer model

We used tiny version of Video Swin Transformer (Swin-T) pretrained Kinetics400[6]. We applied data augmentation that random crop, horizontal flip, shear, random shift, random rotate, posterize, solarize, color enhancement, random contrast, random brightness, sharpen, contrast adjustment, histogram equalization, CutOut[7]. There are implementation details of Swin-T as follow: learning rate is 0.01, weight decay is 0.02, optimizer is AdamW[8], scheduler is cosine annealing with warmup, batch size is 16, number of training epochs is 10, loss is cross entropy.

SlowFast model

We used SlowFast-R50[2] pretrained Kinetics400. We applied the same data augmentation as Video-Swin-Transformer. There are implementation details of SlowFast as follow: resample rate to slow pathway is 2, sampling rate between fast pathway and slow pathway is 2, channel rate between fast pathway and slow pathway is 8, learning rate is 0.1, weight decay is 10^{-4}, optimizer is Stochastic Gradient Decent (SGD) with momentum 0.9, scheduler is cosine annealing with warmup, batch size is 16, number of training epochs is 10, loss is cross entropy.

Results

We got frame-wise accuracy (FWA) is 0.711, segmental F1@10 score (F1@10) is 0.795, final score is 0.752 on CV.

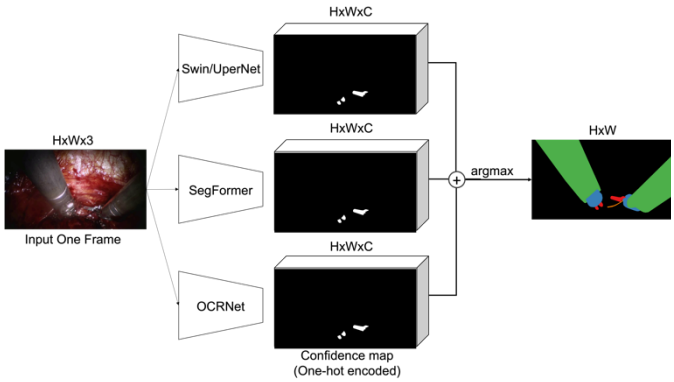

Figure 4. Architecture of segmentation task. H, W, C is height, width, number of classes, respectively. In this task, we set H is 256, W is 512, C is 10 included background class.

2.2 Segmentation task

We ensemble three models, Swin-Transformer, SegFormer and OCRNet (Figrue 4). We expected that the models extracted several features between Transformer and CNN.

Swin-Transformer model

We used Swin-Transformer-Base (Swin-B)[3] pretrained ImageNet as an encoder and UperNet[9] as a decoder. We set Focal loss[10] and Dice loss[11] as losses. We set RAdam[12] as an optimizer. We applied data augmentation that horizontal flip, random shift, random scale, random rotate. There are implementation details of Swin-B as follow: window size of Swin-B is 7, patch size of Swin-B is 4, gamma of Focal loss is 2.0, loss weight between Focal loss and Dice loss is 1.0, learning rate is $5\text{x}10^{-5}$, weight decay is 10^{-5}, scheduler is cosine annealing with warmup, batch size is 32, number of training epochs is 30.

OCRNet model

We used HRNetV2[13] pretrained ImageNet as an encoder and OCRNet as a decoder. Implementation details are the same as for Swin-B.

SegFormer model

We used Mix-Transformer (MiT)-B3[4] pretrained ImageNet as an encoder and SegFormer as a decoder. We set Cross entropy as losses. We set RAdam as an optimizer. We applied data augmentation that horizontal flip, random shift, random scale, random rotate. There are implementation details of SegFormer as follow: learning rate is $5x10^{-5}$, weight decay is 10^{-4}, scheduler is cosine annealing with warmup, batch size is 8, number of training epochs is 30.

Results

We got mean intersection over union (mIoU) is 0.768, normalized surface distance (NSD) is 0.824, final score is 0.796 on CV.

2.3 Multitask of Action Recognition and Segmentation

We approached this task to divide independent tasks of action recognition task and segmentation task, so methods are all same below section 2.2 and 2.3.

Results

We got final score is 0.774 on CV.

3. Discussion and Conclusions

In this challenge, participants were given 40 videos as a training, and 10 videos as a test. The training videos are annotated action and mask of instruments, participants should predict actions every 6 frames and masks every 30 frames. We participated in all tasks.

In action recognition task, what was important to improve accuracy was input multiple frames as a sequence. The ensemble of models outperformed the accuracy of the individual models. On the other hand, large backbone worked worse than small model. It was difficult to recognize action “Tying a knot” and “Cutting the suture”, because it considered these actions are rare cases, so the model didn’t be well-trained.

In segmentation task, small backbone like MiT-B3 works better than large backbone like MiT-B5. We considered it that video frames have many similar scenes, so a large backbone becomes overfitting. On the other hand, it was difficult to recognize threads and clamps because it considered these objects are smaller than other classes, so the model disregard such small area to reduce frame size for input.

In multitask, we didn’t make any special effort. In the future works, we try to build multi head Swin-Transformer model which has action prediction branch and segmentation branch and apply multi-task learning. We assume that it would more accurate if segmentation mask is used as input for the action recognition model.

Citation

[1] Liu, Ze, *et al.* "Video swin transformer." *CVPR*. 2022.
[2] Feichtenhofer, Christoph, *et al.* "Slowfast networks for video recognition." *ICCV*. 2019.
[3] Liu, Ze, *et al.* "Swin transformer: Hierarchical vision transformer using shifted windows." *ICCV*. 2021.
[4] Xie, Enze, *et al.* "SegFormer: Simple and efficient design for semantic segmentation with transformers." *NeurIPS* (2021).
[5] Yuan, Yuhui, *et al.* "Segmentation transformer: Object-contextual representations for semantic segmentation." *arXiv:1909.11065* (2019).
[6] Kay, Will, *et al.* "The kinetics human action video dataset." *arXiv:1705.06950* (2017).
[7] DeVries, Terrance, and Graham W. Taylor. "Improved regularization of convolutional neural networks with cutout." *arXiv:1708.04552* (2017).
[8] Loshchilov, Ilya, and Frank Hutter. "Decoupled weight decay regularization." *arXiv:1711.05101* (2017).
[9] Xiao, Tete, *et al.* "Unified perceptual parsing for scene understanding." *ECCV*. 2018.
[10] Lin, Tsung-Yi, *et al.* "Focal loss for dense object detection." *ICCV*. 2017.
[11] Milletari, Fausto, *et al.* "V-net: Fully convolutional neural networks for volumetric medical image segmentation." *3DV* 2016.
[12] Liu, Liyuan, *et al.* "On the variance of the adaptive learning rate and beyond." *arXiv:1908.03265* (2019).
[13] Sun, Ke, *et al.* "High-resolution representations for labeling pixels and regions." *arXiv:1904.04514* (2019).