StillFast: An End-to-End Approach for Short-Term Object Interaction Anticipation Supplementary Material

Francesco Ragusa Giovanni Maria Farinella Antonino Furnari

FPV@IPLab - University of Catania
Next Vision s.r.l. - Spin-off of the University of Catania
{francesco.ragusa, giovanni.farinella, antonino.furnari}@unict.it

Abstract

This document is intended for the convenience of the reader and reports additional information about the implementation details. This supplementary material is related to the following submission:

- *F. Ragusa, G. M. Farinella, A. Furnari, "StillFast: An End-to-End Approach for Short-Term Object Interaction Anticipation", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. 2023.*

1. Implementation Details

We train the model on the training data, whereas we use the validation set to choose the best performing checkpoint according to the overall mAP measure. We then report the results of such model on both the validation and the test set. At training time, we follow the standard Faster R-CNN multi-scale procedure and feed high-resolution images with a short side in the range $[640, 672, 704, 736, 768, 800]$ and a maximum long side of 1333. As a result, we obtain a still image with height H. The low resolution video is obtained by re-scaling the input high resolution video with linear interpolation in such a way that the height of re-scaled video is equal to $h = \alpha \times H$. We set $\alpha = 0.32$ in all our experiments. In this way, a still image of height $H = 800$ pixels will correspond to a video of height $h = 256$ pixels, which is a standard resolution for video backbones. At test time, we feed to the networks still images of height $H = 800$ pixels and videos of height $h = 256$ pixels. We sample video clips of 16 frames with a sampling stride of 1 frame. During training, we weigh the $\mathcal{L}_v$ loss with 0.1 and the $\mathcal{L}_{ttc}$ loss with 0.5. The 2D backbone is a ResNet-50 architecture. The weights of this backbone and the ones of the standard feature pyramid layer are initialized from a Faster R-CNN model pre-trained on the COCO dataset [3]. The 3D network is an X3D-M model [2] pre-trained on Kinetics [1]. The global-local fusion network included in the prediction head has two connected layers with a ReLU activation in between. The first layer maps features from $256 + 1024$ features to 1024 features, whereas the second layer maps features from 1024 to 1024 dimensions. The weights of the local-global module are initialized randomly. The model is trained with a base learning rate of 0.001 and a weight decay of 0.0001. The learning rate is lowered by a factor of 10 after 15 and 30 epochs. The model is trained in half precision on four NVIDIA V100 GPUs with a batch size of 8. The convolutional layers included in the Combined Feature Pyramid Layers are randomly initialized and have a 3×3 kernel with a padding equal to 1. The first convolutional layer (pre-sum) maps features from the numbers of channels of the 3D network ($[24, 48, 96, 192]$) to numbers of channels of the 2D network ($[256, 512, 1024, 2048]$), whereas the second convolutional layer (post-sum) maps features from the number of channels of the 2D network ($[256, 512, 1024, 2048]$) to the same number of channels. The standard feature pyramid layer maps features to 256 channels. Following the 2D and 3D backbone branch initialization, input still images are normalized with $[0.485, 0.456, 0.406]$ means and $[0.229, 0.224, 0.225]$ standard deviations, whereas the input videos are normalized with $[0.45, 0.45, 0.45]$ means and $[0.225, 0.225, 0.225]$ standard deviations.

References

[1] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern*

Recognition, pages 6299–6308, 2017. 1

[2] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020. 1

[3] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1