Appendix A. Training Datasets

The **YouTube-BoundingBoxes** [10] is a large-scale dataset of videos. The dataset consists of approximately **380,000** video segments of 15-20s with a recording quality often akin to that of a hand-held cell phone camera.

The **LaSOT** [5] consists of 1,400 sequences with more than **3.5M** frames in total. Each sequence contains 2,500 frames on average and the dataset represents 70 different object categories.

The **GOT-10k** [7] is built upon the backbone of WordNet structure [6] and it populates the majority of over 560 classes of moving objects and 87 motion patterns. It contains more than 10,000 of short video sequences with more than **1.5M** manually labeled bounding boxes, annotated at 30 frames per second, enabling unified training and stable evaluation of deep trackers.

The **ImageNet-VID** [4] is a benchmark created for video object detection task. It contains 30 object categories. Overall, benchmark consists of near **2M** annotations and over 4,000 video sequences.

In addition, similar to other tracking models [2], [12], [11], we use a part of the **COCO** [8] dataset for object detection with 80 different object categories to diversify the training dataset for visual object tracking. In our setup, we set $I_S = I_T$ to let the network efficiently predict the object's location in a larger context.

Appendix B. Technical details

B.1. Pixel-wise correlation implementation

Classical cross-correlation cannot be executed by most mobile neural network inference engines such as CoreML [3] due to unsupported convolutional operation with dynamic weights from the template features. Thus, we reformulated the pixel-wise cross-correlation operation as a matrix multiplication operation that is better supported on mobile devices.

Given input image features Φ_S and template image features Φ_T flattened along the spatial dimensions to shapes $C \times WH$ and $C \times wh$ respectively, we compute pixel-wise cross-correlation features Φ_{corr} as:

$$\Phi_{corr} = \Phi_T^{\top} \Phi_S \tag{1}$$

The resulting Φ_{corr} will be a tensor of shape $wh \times WH$.

B.2. Smartphone-based Implementation

The models are trained offline using PyTorch [9] and then ported with an optimal model snapshot to mobile devices for inference. All models are executed in *float16* mode for faster execution comparing to *float32* computations. The precision loss of *float16* computations is negligible, we observe that the results differ only by $\pm 0.5\%$ depending on the experiment.

We use Core ML [3] framework to run FEAR tracker on iPhone devices. Core ML is a machine learning API from Apple that optimizes on-device neural network inference by leveraging the CPU, GPU and Neural Engine.

For Android devices, we employ TensorFlow Lite [1] which is an open-source deep learning framework for on-device inference from Google supporting execution on CPU, GPU and DSP.

Appendix C. Qualitative comparison

The comparison of FEAR tracker with the state-of-the-art methods is presented in Figure 1. We display the tracking results of every 200 frames (0 - 1000) on the challenging cases from LaSOT benchmark where the object appearance and scale change throughout the video.

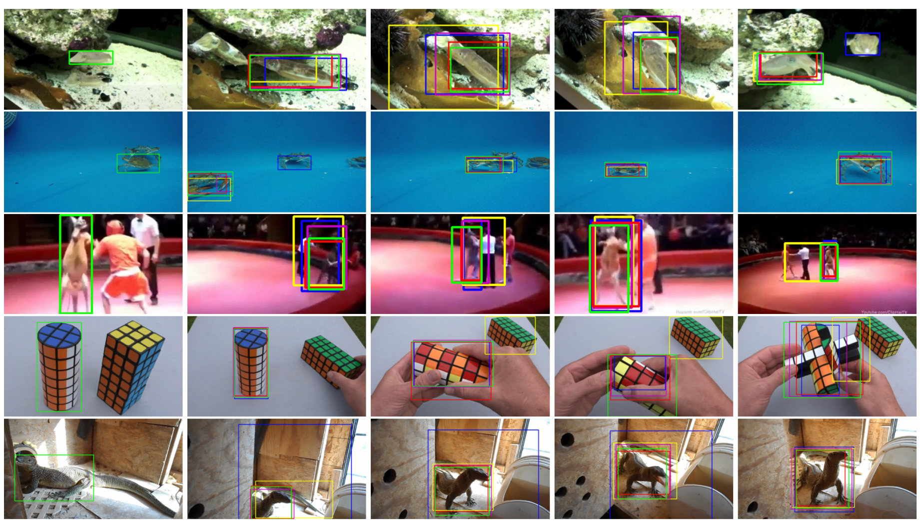

Fig. 1: Qualitative comparison of FEAR tracker with state-of-the-art methods on challenging cases of variations in tracked object appearance from LaSOT benchmark [5]. Green: Ground Truth, Red: FEAR-L, Yellow: STARK Lightning, Blue: Ocean, Purple: Stark-ST50.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016) 2
2. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking. In: European conference on computer vision. pp. 850–865. Springer (2016) 1
3. Core ML. `https://developer.apple.com/documentation/coreml` 1, 2
4. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009) 1
5. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C., Ling, H.: Lasot: A high-quality benchmark for large-scale single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019) 1, 2
6. Fellbaum, C.: WordNet: An Electronic Lexical Database. Bradford Books (1998) 1
7. Huang, L., Zhao, X., Huang, K.: Got-10k: A large high-diversity benchmark for generic object tracking in the wild. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**(5), 1562–1577 (2021) 1
8. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollar, P., Zitnick, L.: Microsoft coco: Common objects in context. In: ECCV (September 2014) 1
9. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019) 1
10. Real, E., Shlens, J., Mazzocchi, S., Pan, X., Vanhoucke, V.: Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5296–5305 (2017) 1
11. Zhang, Z., Peng, H., Fu, J., Li, B., Hu, W.: Ocean: Object-aware anchor-free tracking. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16. pp. 771–787. Springer (2020) 1
12. Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., Hu, W.: Distractor-aware siamese networks for visual object tracking. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 101–117 (2018) 1