Mobile Vision Transformer-based Visual Object Tracking

Goutam Yelluru Gopal and Maria Amer
Concordia University, Montréal, Québec, Canada

BMVC 2023

Introduction

Motivation:

- Vision Transformer-based trackers
  - are computationally expensive due to a large number of model parameters.
  - require specialized hardware for real-time inference.

Tracker	GOT10k		TrackingNet		#params ↓	*fps*
	OR ↑	$SR_{0.50}$ ↑	AUC ↑	P_{norm} ↑	(in millions)	CPU ↑
DiMP-50	0.611	0.717	74.0	80.1	26.1	15.0
TransT	0.671	0.768	81.2	85.4	23.0	2.3
STARK-ST101	0.688	0.781	82.0	86.9	47.2	7.8
OSTrack-384	0.740	0.835	83.9	88.5	92.1	4.4
MixFormer-L	0.756	0.857	83.9	88.9	183.9	< 5

- **Our Solution:** Mobile Vision Transformer for fast tracking.

Key Contributions

Mobile Vision Transformer-based backbone:

- Cascade of Convolutional and Transformer blocks for feature extraction.
- Convolutional blocks model the spatially local information.
- Transformer blocks capture the long-range feature dependecies.

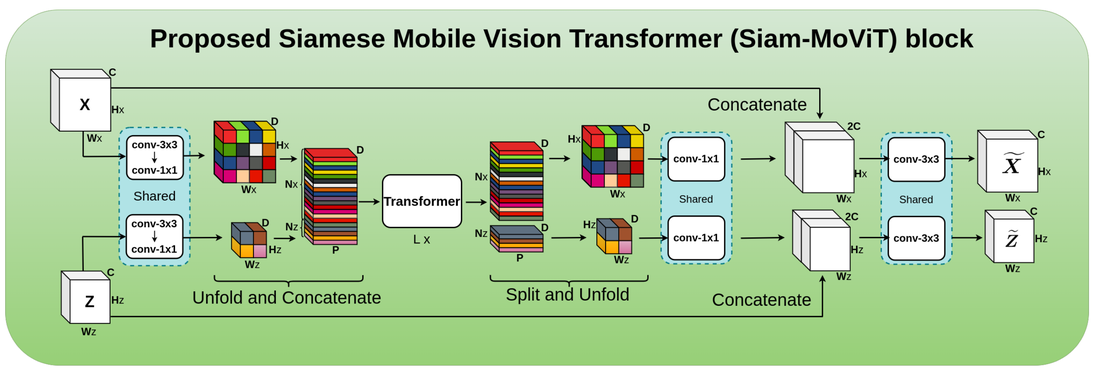
Feature fusion in tracker backbone:

- Self-attention on the concatenated template and search region features.
- Exchange of information ***within*** and ***between*** the two regions.

High inference speed:

- Joint feature extraction and fusion requires fewer attention operations.
- 175 *fps* on GPU and 29 *fps* on CPU (Pytorch).
- 300 *fps* on GPU (TensorRT) and 70 *fps* on CPU (ONNX Runtime).

Proposed Mobile Vision Transformer-based Tracker (*MVT*)



Proposed Backbone:

- Cascaded MobileNetV2 (or MV2) and Siam-MoViT blocks for feature extraction.
- Siam-MoViT block fuses features from the two branches.

Neck Module:

- Cross-correlation between template and search region features.

Head Module:

- Two fully-convolutional branches for classification and bounding box regression.

Loss function for training:

- Classification (L_{cls}) and regression (L_1 and L_{giou}) losses.
- Overall training loss,

$$L_{total} = L_{cls} + \lambda_1 \cdot L_1 + \lambda_2 \cdot L_{giou}.$$

Results

Implementation Details:

- The template and search region dimensions are 128×128 and 256×256.
- GOT10k-train dataset for training the model.
- Training for 100 epochs with a batch size of 128.
- The learning rate is set to 4×10^{-4} with cosine annealing as the scheduler.
- Initialization of our tracker backbone using pretrained MobileViT weights.
- During inference, we apply Hanning window on classification score map.

Comparison to Related Lightweight Trackers:

Tracker	**GOT10k** (server)			**TrackingNet** (server)			NfS30		LaSOT		*fps*
	OR ↑	$SR_{0.50}$ ↑	$SR_{0.75}$ ↑	AUC ↑	P_{norm} ↑	P ↑	AUC ↑	FR ↓	AUC ↑	FR ↓	(GPU)
LightTrack	0.582	0.668	0.442	72.9	79.3	69.9	0.582	0.146	0.524	**0.116**	99
Stark-Lightning	0.596	0.696	0.479	72.7	77.9	67.4	**0.619**	0.111	0.585	0.151	205
FEAR-XS	0.573	0.681	0.455	71.5	80.5	69.9	0.487	0.207	0.508	0.273	275
E.T.Track	0.566	0.646	0.425	74.0	79.8	69.8	0.589	0.172	**0.597**	0.162	53
MVT (ours)	**0.633**	**0.742**	**0.551**	**74.8**	**81.5**	**71.9**	0.603	**0.085**	0.553	0.137	175

- *MVT* has the best performance on server-based GOT10k and TrackingNet.
- Overall, *MVT* outperforms the related trackers in 7 out of 10 metrics.

Comparison to State-of-the-art:

Tracker	GOT10k		TrackingNet		#params ↓	*fps*	
	OR ↑	$SR_{0.50}$ ↑	AUC ↑	P_{norm} ↑	(in millions)	GPU ↑	CPU ↑
DiMP-50	0.611	0.717	74.0	80.1	26.1	61.5	15.0
TransT	0.671	0.768	81.2	85.4	23.0	87.7	2.3
STARK-ST101	0.688	0.781	82.0	86.9	47.2	80	7.8
OSTrack-384	0.740	0.835	**83.9**	88.5	92.1	74.4	4.4
MixFormer-L	**0.756**	**0.857**	**83.9**	**88.9**	183.9	45.2	< 5
MVT (ours)	0.633	0.742	74.8	81.5	**5.5**	**175.0** (**300***)	**29.4** (**70****)

***TensorRT**, ****ONNX-Runtime**

- **State-of-the-art:** Deployment of transformers has improved the performance, but at the cost of lowered tracking speed.
- In contrast, our *MVT* surpasses DiMP-50 with $4.7\times$ fewer parameters while running at $2.8\times$ and $2\times$ its speed on GPU and CPU, respectively.

Analysis

Ablation Study on Feature Fusion:

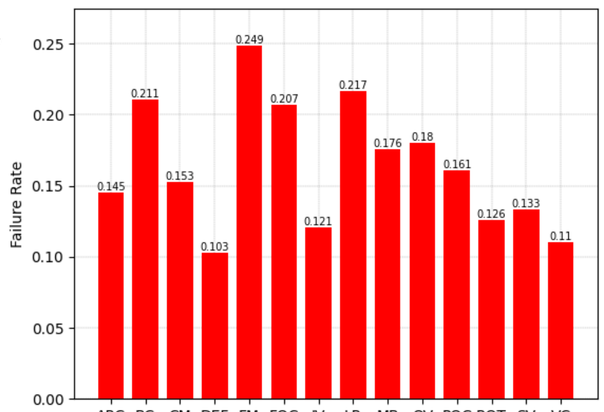
- We retrain our model without concatenating the template and search region features inside the proposed Siam-MoViT block.

feature fusion in backbone	GOT10k		TrackingNet		NfS30		LaSOT	
	OR ↑	$SR_{0.50}$ ↑	AUC ↑	P_{norm} ↑	AUC ↑	FR ↓	AUC ↑	FR ↓
✗	0.600	0.703	**74.9**	80.0	0.566	0.122	0.544	0.163
✓(ours)	**0.633**	**0.742**	74.8	**81.5**	**0.603**	**0.085**	**0.553**	**0.137**

- Proposed feature fusion improves AUC and reduces FR on average.

Robustness Analysis:

- We compare the FR of our *MVT* on attributes from the LaSOT dataset.
- *MVT* is robust to target deformation and appearance changes.
- *MVT* has a higher FR while tracking small, fast-moving target objects, *e.g.*, *volleyball*.



Conclusion

- We proposed a tracker that uses Mobile Vision Transformer, for the first time.
- Our tracker performed better than the related lightweight trackers, especially on server-based GOT10k and TrackingNet datasets.
- *MVT* runs at 70 *fps* on CPU, faster than second-best Stark-Lightning (50 *fps*).
- **Future work:** Deployment on embedded devices (*e.g.*, smartphones).

Project Webpage:

Tracker Code & Model