Achieving Real-time Visual Tracking with Low-Cost Edge AI

Van Minh Do, Meiqing Wu, Siew-Kei Lam, Thambipillai Srikanthan
Nanyang Technological University, Singapore
{vmdo,meiqingwu,assklam,astsrikan}@ntu.edu.sg

Abstract

Visual multiple object tracking (MOT) algorithms based on deep learning are computationally intensive, and often cannot achieve real-time performance on low-cost edge computing platforms. We propose an algorithmic-hardware co-design methodology that combines novel algorithm augmentations and architecture mapping of state-of-the-art visual MOT on heterogeneous multi-core processor. We applied the proposed algorithm augmentations to two deep visual MOT pipelines. Experiments based on widely-used datasets demonstrate that the proposed methods outperform the baselines. We also show that the proposed methodology is able to achieve high performance on a low-cost embedded device (Odroid N2+), making it viable for real-time automated traffic surveillance with edge AI.

***Keywords*—** object tracking, real-time, embedded devices, heterogeneous multi-core.

1. Introduction

With the increase in roadside cameras and advancements in deep learning for multiple object tracking (MOT), there's a pressing need for real-time traffic surveillance to enhance urban mobility. However, central processing of traffic data requires extensive bandwidth, storage, and computing power, raising costs and privacy concerns. Edge computing offers a solution by processing data on-site but struggles to realize complex MOT algorithms on resource-constrained devices without sacrificing accuracy.

Our research introduces an algorithmic-hardware co-design methodology that optimizes the application of state-of-the-art MOT algorithms on low-cost, multi-core processors. This methodology focuses on reducing computational demands through adaptive region-of-interest (ROI) determination, lightweight change detection, and a novel motion prediction model. These innovations allow for real-time, accurate traffic monitoring on embedded platforms, offering a scalable and privacy-preserving approach to traffic surveillance.

2. Proposed Methodology

Adaptive ROI for Object Detection: We propose a lightweight change detection algorithm (see Figure 1) to rapidly identify the foreground mask in each camera frame, to determine the ROI for accelerating object detection. Our algorithm extends the visual background extractor (ViBe) [1] technique, which is fast and well-suited for embedded devices. However, ViBe has certain shortcomings for static objects. It is unable to classify static objects (e.g., vehicles waiting at the traffic light) as foreground when they first appear in the camera view. In addition, when the static objects start to move, the ghost phenomenon occurs (see Figure 2a) and remains in the foreground for a long time. This is because the background model of ViBe has incorporated the static objects. This may lead to a larger ROI that contains false foregrounds. We deal with this issue by applying object detection to generate a foreground mask of target objects to enhance the background model. Pixels that are not associated with the foreground mask will be included in the background model; otherwise, the inverted value of the current pixel will be used in the corresponding positions. This helps the foreground detection module to classify pixels of the static objects as foreground in the subsequent frames.

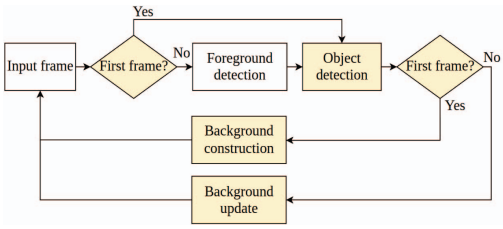

Figure 1. Improved change detection by incorporating feedback from object detection. The background model is built and updated based on the foreground mask of object detection. Our contributions are highlighted in yellow.

Figure 2. Adaptive ROI. a) Ghosts and background noises appear in foreground mask of original ViBe. b) Ghosts and background noises are removed from the foreground mask with our change detection. c) ROI for object detection indicated in yellow box.

Lightweight Motion Model: Our lightweight motion model addresses the challenge of large object displacements in edge-based MOT frameworks by using optical flow. By tracking Shi-Tomasi corners across frames, it accurately predicts object locations, overcoming the limitations of traditional models like the Kalman filter in scenarios of skipped frames. The final track predictions at the current frame are selected from either our optical flow method or the motion model of the existing MOT framework, by comparing RGB histogram similarity between patches of the previous track and the prediction of two motion models.

Heterogeneous Multi-core Processing: In our multi-core pro-

979-8-3503-6927-4/24/$31.00 ©2024 IEEE
DOI 10.1109/ICCPS61052.2024.00035

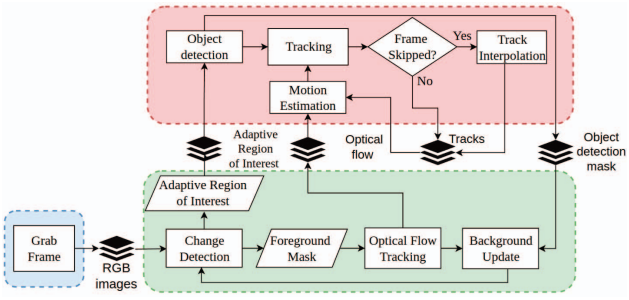

Figure 3. Overview of our proposed multi-core processing of MOT on Odroid N2+ [6]. All three processes run concurrently. First process (blue) obtains the camera stream. Second process (green) executes change detection and optical flow at high speed. Third process (red) performs object detection and tracking.

cessing strategy, the algorithm augmentations (adaptive ROI and motion model) are processed at high frame rates on the slower cores to compensate for the skipped frames encountered by the deep learning-based methods that run on powerful cores. The architecture (see Figure 3) has three main processes working concurrently at different speeds on multiple cores. The blue process obtains video frames at the camera frame rate and stores them in a shared buffer. The green process consists of our change detection and sparse optical flow method that can work at high frame rate to compensate for the skipped frames encountered in deep learning-based object detection and tracking. The red process contains object detection and tracking that runs on powerful cores but operate at a lower rate than the green process. These processes are synchronized and exchange data through shared buffers.

3. Experiments and Results

Experiment Settings: Our experiments employed the MOT15 [2] and EPFL [3] datasets, divided into 60% for training, 20% for validation, and 20% for testing, tailored for traffic surveillance applications. We employed the yolov5s model for object detection and the SORT [4] and HES-Track [5] algorithms for tracking. We assessed the accuracy and performance on Odroid N2+ using MOT metrics (HOTA, MOTA, and IDF1).

Results: Table 1 compares the accuracy between our method and the baselines on the Odroid N2+, using a track interpolation mechanism to address frame skipping due to computational constraints. The results demonstrate that interpolation boosts the accuracy across all methods. Notably, our proposed method outperforms the baselines even in the absence of interpolation, thanks to its adaptive ROI and robust motion model. These features effectively enhance object tracking across significant displacements, counteracting the challenges posed by skipped frames.

Figure 4 analyses the tracking performance on Odroid N2+ at different camera frame rates for the PETS09-S2L1 sequence. The results show that our change detection and optical flow method can operate at the camera frame rate, which helps to compensate for the large displacement of target objects due to skipped frames. In addition, due to the adaptive ROI method, our detection and tracking are faster than the baselines and result in fewer skipped frames, which in turn, contributes to higher tracking accuracy.

					Without Interpolation			With Interpolation		
Tracking	Dataset	Sequence	FPS	Method	HOTA	MOTA	IDF1	HOTA	MOTA	IDF1
SORT	MOT15	KITTI-17	10	Baseline	0.17	0.14	0.25	0.47	0.40	0.57
				Our method	**0.22**	**0.22**	**0.38**	**0.52**	**0.55**	**0.71**
		PETS09-S2L1	7	Baseline	0.07	0.09	0.12	0.26	0.34	0.36
				Our method	**0.21**	**0.26**	**0.36**	**0.48**	**0.62**	**0.64**
		TUD-Campus	25	Baseline	0.07	0.06	0.11	0.20	0.14	0.22
				Our method	**0.09**	**0.09**	**0.14**	**0.38**	**0.38**	**0.40**
		Vience-02	30	Baseline	0.03	0.02	**0.08**	0.39	0.29	**0.55**
				Our method	**0.04**	**0.03**	0.07	**0.43**	**0.30**	0.52
	EPFL	Passageway1-c1	25	Baseline	0.02	0.02	0.03	0.18	0.17	0.29
				Our method	**0.09**	**0.10**	**0.19**	**0.65**	**0.81**	**0.90**
HES-Track	MOT15	KITTI-17	10	Baseline	0.19	0.18	0.34	0.44	0.46	0.63
				Our method	**0.27**	**0.25**	**0.45**	**0.55**	**0.59**	**0.76**
		PETS09-S2L1	7	Baseline	0.13	0.15	0.24	0.48	0.60	0.63
				Our method	**0.30**	**0.37**	**0.47**	**0.57**	**0.76**	**0.71**
		TUD-Campus	25	Baseline	0.07	0.06	0.11	0.07	0.06	0.11
				Our method	**0.10**	**0.10**	**0.19**	**0.48**	**0.41**	**0.57**
		Vience-02	30	Baseline	0.03	0.01	**0.08**	0.41	0.27	0.55
				Our method	**0.04**	**0.03**	0.07	**0.43**	**0.28**	**0.56**
	EPFL	Passageway1-c1	25	Baseline	0.03	0.03	0.05	0.35	0.46	0.55
				Our method	**0.11**	**0.13**	**0.24**	**0.78**	**0.94**	**0.97**

Table 1. Accuracy of baselines and our method on Odroid N2+. Impact of interpolation mechanism is evaluated for all methods.

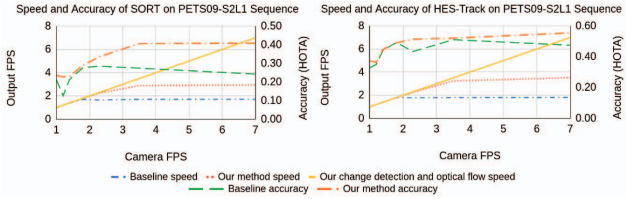

Figure 4. Processing speed and accuracy of baselines and our method on Odroid N2+ for different camera frame rates.

4. Conclusions

We presented an algorithmic-hardware co-design methodology that enables state-of-the-art MOT algorithms to be realized on low-cost edge AI devices, comprising of heterogeneous multi-core processors, without relying on GPUs. We addressed the inherent problem of skipped frames on resource-constrained devices, by dynamically reducing the computational requirements of the object detector through adaptive ROI, improving the correlation of objects with large displacements between consecutively processed frames, and a multi-core processing strategy that enables tightly coupled operations between our fast algorithm augmentations with accurate but slow deep learning methods.

References

[1] Barnich Olivier et al. Vibe: A universal background subtraction algorithm for video sequences. *IEEE Transactions on Image processing*, 2010. 1

[2] Dendorfer Patrick et al. Motchallenge: A benchmark for single-camera multiple target tracking. *International Journal of Computer Vision*, 2021. 2

[3] Fleuret Francois et al. Multicamera people tracking with a probabilistic occupancy map. *IEEE Transactions on pattern analysis and machine intelligence*, 2007. 2

[4] Nicolai Wojke et al. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing*. 2

[5] Wu Meiqing et al. Robust and low complexity obstacle detection and tracking. In *IEEE 19th International Conference on Intelligent Transportation Systems*, 2016. 2

[6] LTD. HARDKERNEL CO. Odroid-n2+ with 4gbyte ram. [Accessed 25-01-2024].