You Better Look Twice: a new perspective for designing accurate detectors with reduced computations

Alexandra Dana
alex.dana@samsung.com
Maor Shutman
maor.shutman@samsung.com
Yotam Perlitz
perlitz@gmail.com
Ran Vitek
ran.vitek@samsung.com
Tomer Peleg
tomer.peleg@samsung.com
Roy J. Jevnisek
roy.jewnisek@samsung.com

Samsung Israel R&D Center
Tel Aviv, Israel

Abstract

General object detectors use powerful backbones that uniformly extract features from images for enabling detection of a vast amount of object types. However, utilization of such backbones in object detection applications developed for specific object types can unnecessarily over-process background regions. In addition, they are agnostic to object scales, thus redundantly process all image regions at the same resolution. In this work we introduce BLT-net, a new low-computation two-stage object detection architecture designed to process images with a significant amount of background and objects of variate scales. BLT-net reduces computations by separating objects from background using a very lite first-stage. BLT-net then efficiently merges obtained proposals to further decrease processed image regions and then dynamically reduces their resolution to minimize computations. Resulting image proposals are then processed in the second-stage by a highly accurate model. We demonstrate our architecture on the pedestrian detection problem, where objects are of different sizes, images are of high resolution and object detection is required to run in real-time. We show that our design reduces computations by a factor of $\times 4$-$\times 7$ on the Citypersons and Caltech datasets with respect to leading pedestrian detectors, on account of a small accuracy degradation. This method can be applied on other object detection applications to reduce computations.

1 Introduction

Object detection is a fundamental computer vision task with an abundant number of applications. General object detection architectures such as one-stage and two-stage detectors are

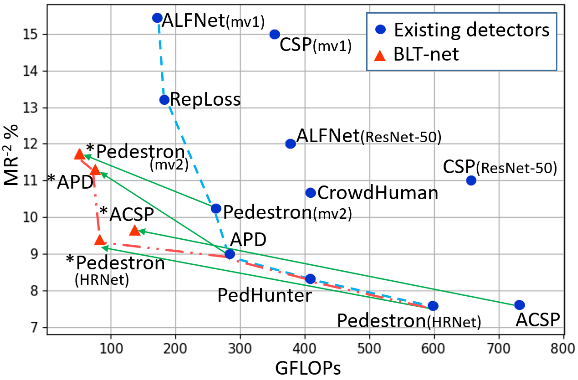


Figure 1: Accuracy (measured by MR^{-2}, lower is better) *vs*. computations of all pedestrian detectors under 10^{12} FLOPs for the Citypersons reasonable validation dataset (2048×1024 px). The dashed blue line depicts the existing empiric Pareto frontier [43, 75], green arrows show accuracy/computations change after detectors were integrated into the BLT-net architecture (red triangles), creating the new empiric Pareto frontier (dot-dashed red line).

designed to enable detection of a vast amount of object types in an image, without making any prior assumptions regarding objects' overall image coverage, objects size and amount of background. Over the years, these approaches achieved impressive detection accuracy results on general datasets such as PASCAL VOC [15] and MS COCO [36].

However, a variety of real-world applications are developed for detecting specific object classes, such as pedestrians and vehicles in autonomous driving [2, 13, 17, 71], buildings and roads in satellite images [8], humans and vehicles in aerial images [64], faces in surveillance, auto-focus and tagging applications [29, 67, 77] to name a few. In such scenarios, a non-negligible amount of image regions can be potentially referred to as background. These regions, when detected at a low computational cost and discarded from further processing, could significantly reduce the number of computations required for rich features extraction needed for accurate object detection. In addition, low-computation estimation of objects scale can be utilized to significantly reduce the resolution of candidate image regions, thus further decreasing the overall computations of the detector.

In this work we revisit the two-stage architecture to better utilize both object image coverage and objects scale. To this end, we suggest BLT-net, a two-stage architecture that **b**etter **l**ooks **t**wice at prospective objects of interest. BLT-net uses a very lite first-stage architecture to estimate objects location and their scale and a second-stage that processes these image proposals for accurate detection. To further decrease computations of the second-stage, we aim to dynamically produce an amount of proposals that is highly similar to the number of relevant objects in the image, with minimal overlap between proposals. In the second-stage, instead of directly extracting rich features for accurate object detection on proposals at full image resolution, we propose downscaling them to the minimum possible resolution to significantly reduce this stage calculations. For the second-stage one can develop an accurate detector to process image crops. General [3, 55, 60] and specialized object detectors [16, 41, 42, 58, 63, 66] were shown to achieve high detection accuracy on full-resolution images. We build upon their success and apply them directly on the first-stage proposals, with no additional fine-tuning.

We demonstrate both analytically and empirically the computations advantages of utilizing both image coverage and objects scaling estimation with respect to existing object detection architectures. To show the latter, we apply BLT-net for detecting pedestrians in driving scenes, a popular use-case that is highly sensitive to computations [35]. Our experiments cover the popular Citypersons [71] and Caltech [13]. This use-case can benefit tremendously from computations reduction due to its requirement to process high-resolution images for detecting objects at various distances (and thus of various sizes) in real-time [35] on edge-devices [28]. For example, even the latest edge-devices AI accelerators such as Snapdragon 865 and Exynos 990 have a maximal capacity of 250 GFLOPs/frame [1,2] [31]. However, as seen in Figure 1 none of the leading pedestrian detectors with a MR^{-2}<10% (see definition in Section 4) can meet this limitation. In contrast, BLT-net was able to reduce the number of FLOPs by a factor of $\times 4$-$\times 7$ with a small accuracy (MR^{-2}) degradation of 1.4%-2.3% of several leading pedestrian detectors, allowing for their real-time inference on edge-devices. In summary, the main contributions of our work are:

- A new perspective of the two-stage object detection architecture designed to better utilize image object coverage and object scales.
- A novel method for decreasing computations of existing object detectors and an estimation method for quantifying the method's computations reduction potential on any analyzed dataset.
- Theoretical and quantitative analysis of computation reduction aspects of our suggested architecture on the Citypersons and Caltech datasets.

2 Related work

Generally, one-stage or two-stage architectures can be applied to solve various specific object detection use-cases, such as face detection, cars and humans in aerial images, crops in agricultural images and pedestrians and vehicles in self-driving cars. To reduce latency and computations and in some cases to increase detection accuracy, over the years architectures were modified or redesigned to better utilize characteristics of the scenery and detected objects, such as foreground/background ratio, objects location and scale distributions.

For example, background region removal for improved computations and latency was demonstrated for pedestrian detection on the Caltech dataset using reinforcement learning techniques [16, 58]. In these solutions, the first-step processed downscaled images by a factor of $\times 2 - \times 5$ to detect candidate regions to be processed in the fine-step at their full resolution. The same detection architecture was used in both steps. A high downscaling factor of $\times 5$ resulted in a high number of proposed regions that were processed by the second stage to compensate for possible misses. Overall this approach processed $\sim 40\%$ of pixels for both downscaling factors. In our work we propose an opposite approach: we process in the first-stage full-resolution images by a very lite detector to enable detection of even small objects at their full image-resolution and then use objects' scaling estimation to reduce image resolution of proposed regions to significantly reduce computations of the accurate detector used in the second-stage, overall achieving a higher computations reduction factor.

Another approach commonly used for discarding background regions processes images using a cascade of two or more classical [59] or CNN-based steps [33, 42, 68]. This method

[1] FLOPs measure mult-add operations
[2] For 30 frames per second

was successfully applied for detecting faces [33, 68] or general objects [42]. To further reduce computations of later steps, some works united overlapping proposals [5, 42, 66], while others designed their first step architecture to detect regions containing possible clusters of objects [32, 49]. Nevertheless, these works did not optimize object clustering with respect to their background regions, which can be further reduced to decrease computations. In our work we demonstrate that a good scaling estimation of singular objects in the first-stage combined with a cost-efficient area merging criterion can significantly discard additional background areas, thus directly reducing computations of the second-stage detector. In addition, we show that in scenarios where object sizes significantly vary, it is highly beneficial to downscale obtained image proposals, to further decrease computations of the later stage.

In this work we demonstrate our suggested meta-architecture on the computations-sensitive pedestrian detection problem. The majority of pedestrian detectors are based on one-stage or two-stage architectures. One group of works adopted the accurate Faster R-CNN architecture [47] and introduced various adjustments to the common scheme. RepLoss [63] altered the training loss, AdaptiveNMS [38] refined the NMS step, MGAN [44] and PSC-Net [65] added additional branches, PedHunter [7] introduced a masked guided module for improved detections in crowded scenes, while Pedestron [20] and CrowdHuman [51] trained models on extensive datasets. Other works adapted single-stage architectures developed for general object detection such as SSD [39], YOLO [45, 46] and others [11, 37] to potentially decrease inference latency and computations [26]. These include ALFNet [40], CSP [41], APD [70] that use various backbones, loss terms, training dastasets and anchor-based/free approaches. Overall, these detectors use general backbones for feature extraction that result in an increased amount of computations, as seen in Figure 1 and in the Supplementary Material. In this work we will demonstrate how our proposed architecture enables computation reduction of such existing designs, significantly improving the Pareto frontier [43, 75].

3 BLT-net

The BLT-net architecture consists of two stages. The first-stage processes the full-resolution input image using a very lite architecture to extract features for a rough object localization and scaling estimation and thus separates objects of interest from background. Features from this stage are then discarded. Since detected objects may overlap each other, same image regions (ROIs) could be processed several times by the second-stage. To this end, overlapping ROIs are merged into mROIs using a cost-effective area criterion. Image crops are cut from the input image using the calculated mROIs and adaptively downscaled using objects' predicted scale to further reduce computations. The downscaled image crops are then fed into the second-stage. This stage is implemented as a separate object detector that uses a powerful backbone to extract rich features only from image crops for accurate detection.

In contrast, two-stage architectures such as Faster-RCNN use a single backbone to extract rich features from the entire image, which are used both in the first-stage for background removal (ROIs estimation) and accurate localization in the second-stage, creating a strong *coupling* between the final detection accuracy and the computations cost of the backbone, a property we directly tackle. An overview of the suggested architecture and comparison to Faster R-CNN [47] is described in Figure 2.

Let us formulate the number of computations of our approach and analyze the computations reduction factor with respect to an existing architecture integrated into our second-stage. Specifically, let us denote the number of FLOPs per pixel of a general object detector by A and the number of processed pixels in the original image by N. Thus the overall computations of a CNN based model are:

$$FLOPs(OD) = A \cdot N \tag{1}$$

Let us denote the number of FLOPs per pixel of a lite CNN first-stage architecture by B and the total number of pixels in the resulting downscaled mROIs by M. Thus the total number of FLOPs in our suggested architecture when incorporating an existing object detector in the second stage is:

$$FLOPs(BLTnet) = B \cdot N + A \cdot M \tag{2}$$

The FLOPs reduction factor of BLT-net is, therefore:

$$1/FLOPs_{reduction\_factor} = (B \cdot N + A \cdot M)/(A \cdot N) = B/A + M/N \tag{3}$$

Eq. (3) indicates that the number of computations could be decreased by reducing both the B/A and M/N factors. We suggest a highly efficient first-stage to reduce the B/A factor and a novel ROI merging and content-adaptive scaling algorithm to reduce the M/N factor.

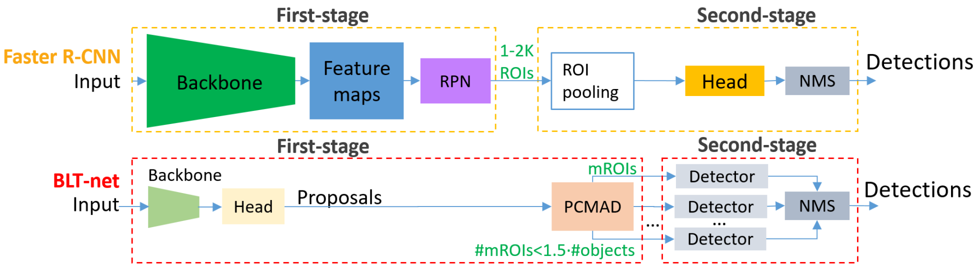


Figure 2: BLT-net *vs*. the Faster-RCNN two-stage architecture. The first-stage is replaced by a lite-CNN detector. The ROI pooling module is replaced by the PCMAD algorithm to merge and downscale image proposals. The second-stage head is replaced by an accurate detector that independently processes image crops based on calculated mROIs.

3.1 First-stage design

Ideally, we aim at designing a first-stage detector with a high sensitivity and a minimal false-positive rate to reduce the number of processed pixels by the second-stage. More formally, let us define two metrics to evaluate the performance of the first-stage detector:

1. Sensitivity: the percentage of ground truth bounding boxes in the dataset for which at least $k\%$ of their area is covered by mROIs. $k\%$ should be set high enough to ensure that adequate image object information is provided to the second-stage for successful detection.

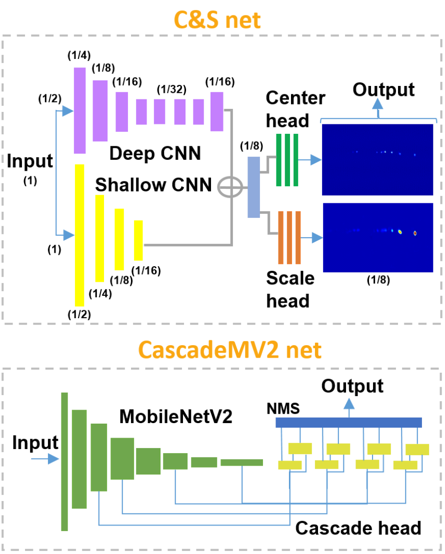
2. Relative processed area ($\overline{M}/N$): the average number of pixels in the dataset processed by the second-stage, relative to the full-resolution image.

In this work we experimented with two different lite-CNN detection architectures to meet the criteria, one anchor-free and the other anchor-based (see Figure 3). First, we denote the **anchor-free** architecture by Center & Scale net (C&S). The backbone of this architecture consists of two branches that process the image at its original resolution (shallower branch) and at half of the original resolution (deeper branch), a concept previously used [69, 76].

The output resolution of the shallow/deep branches was set to 1/8 and 1/16 respectively. The deep resolution branch output was upsampled to 1/8 of the original image resolution and fused to the output of the shallow branch output. The network heads output center-likelihood and object-scale maps, similarly to Liu *et al.* [41] (see details in Supplementary Material). The C&S net was trained on the Citypersons dataset with crops of size 768×768 px and a batch size of 8. We used random scaling in the range [0.5, 2], random uniform cropping and random mirroring with a probability of 0.5. The model was trained for 32 epochs with a learning rate of 10^{-4} and fine-tuned for an additional 350 epochs with a learning rate of 10^{-5}. While current pedestrian detectors with MR^{-2}<10% (see Supplementary Material) require computations in the range of 262-867 GFLOPs for images with a resolution of 2048×1024 px, C&S has 27.5 GFLOPs for image, achieving a maximum ratio B/A of 0.1.

Second, we developed an **anchor-based** first-stage detector, named CascadeMV2. This detector uses a MobilenetV2 [50] backbone for feature extraction with a cascade head for improved accuracy [3, 40]. In this work the detection head was designed using separable convolutions [25] to reduce computations. For additional network architectural details see Supplementary Material. Losses were set similarly to anchor-based object detectors [40]. The resulting architecture was trained for 350 epochs on the Citypersons training dataset with a learning rate of 10^{-4} and fine-tuned for an additional 100 epochs with a learning rate of 10^{-5}. For the Caltech dataset, the model trained on the Citypersons dataset was fine-tuned for an additional 100 epochs with a learning rate of 10^{-5}. Overall, this detector has 25 GFLOPs, similarly to the C&S net. For comparison purposes, the original Cascade R-CNN [3] has 262 GFLOPs and ALFNet [40] has 171 GFLOPs for the same image resolution.

Our experiments with designing lite CNNs demonstrated that various architectures could be developed to meet the first-stage criteria, thus we argue that such architecture is not unique and can be constructed using multiple variants or fully automated using NAS methods [14].



Algorithm 1 PCMAD

Input: $props_{all}$, $heightRanges$. Init: Empty($mROI_{all}$)
for $hRange$ in $heightRanges$ **do**
  $props$ = $props_{all} \in hRange$; $Q, mROIs$=INIT($props$)
  while not empty(Q) **do**
    (i,j)=pop(Q), s.t. min($dist_x(prop_i, prop_j)$)
    if area(ROI_{ij})<area(ROI_i)+area(ROI_j) **then**
      ROI_j=extended(BBox($prop_i$, $prop_j$))
      Update pairs in Q with updated ROI_j
      if cannot pair ROI_j to $ROIs \in Q$ **then** $ROI_j \rightarrow mROIs$
  $mROIs \rightarrow mROI_{all}$
return $mROI_{all}$
procedure INIT($props$)
  Empty(Q), Empty($mROIs$)
  for each prop i,j in $props$ **do**
    ROI_i,ROI_j=extended($prop_i$),extended($prop_j$)
    $ROI_{i,j}$=extended(BBox($prop_i$, $prop_j$))
    if area(ROI_{ij})<area(ROI_i)+area(ROI_j) **then**
      $(i,j) \rightarrow Q$
  $\triangleleft$ if $prop_i$ is not paired to any other $prop_j$
  for each $prop_i$ in $props$ **do**
    if $(i, ?) \notin Q$ **then** $ROI_i \rightarrow mROIs$
  Return Q, $mROIs$

Figure 3: (L) Two possible first-stage lite-CNN architectures. (R) PCMAD pseudo-code.

3.2 The PCMAD algorithm

To increase objects' context and compensate for possible localization and scaling errors of the first-stage, object proposals were extended by $p\%$ in each direction, creating ROIs. To avoid redundant processing of overlapping ROIs by the second-stage, we developed a **p**roposals **c**ontent aware **m**erging and **a**dptive **d**ownscaling algorithm (PCMAD) to produce a set of ROIs with minimal overlapping area. This hierarchical clustering algorithm iteratively merges overlapping proposals with similar heights if the sum of their ROIs area is larger than their union area (united proposals after extension). The proposal pair with the smallest horizontal distance between its centers is selected for merging first.

The resulting mROIs contain objects of similar heights, all with the original image resolution. However, high-resolution mROIs may be redundant when processing large objects for accurate detection, thus to further reduce computations we conduct mROIs adaptive scaling to a height h. This scaling imposes merging object proposals with heights in a specific range only, ensuring that mROIs containing smaller objects are downscaled by a smaller factor than mROIs containing larger objects. In this work height ranges were set to $[0,h)$, $[h,1.5h)$, $[1.5h,2h)$, $[2h,2.5h)$ and $[2.5h,3h)$. The h parameter is determined by the second-stage detector minimum resolution capability for achieving the required accuracy. Overall the final mROIs are of h height and their width was set as the minimum value of $\{0.5h,h,1.5h,2h\}$ that bounds the width of the resulting merged and downscaled proposals. PCMAD complexity is $O(P^3)$ with respect to the number of proposals P. PCMAD pseudocode is shown in Figure 3. A diagram demonstrating this concept can be found in the Supplementary Material.

3.3 Second-stage architecture

Generally, the second-stage architecture could be developed from scratch to learn to accurately detect objects from mROIs or one can use existing pre-trained detectors. As current pedestrian detectors achieve highly accurate results on the border of saturation for the Citypersons and Caltech datasets, we opted for integrating such leading pedestrian detectors for the second-stage. For the Citypersons dataset we integrated Pedestron [20], APD [70] and ACSP [62] and for the Caltech dataset we integrated the Pedestron detector. Figure 1 and Supplementary Material show detection accuracy and computations of these detectors.

The Pedestron detector is based on Faster R-CNN and configured to evaluate during inference 1000 ROIs for images of size 2048×1024 px. Since we apply this detector on smaller image crops (mROIs), we scale down to 64 ROIs per mROI. APD and ACSP are single-stage detectors, thus were applied with their original configuration. Final detections from all image crops are processed by a non-maximal-suppression (NMS) algorithm to remove duplicate detections from overlapping mROIs. The h parameter was set for Pedestron/ACSP/APD to 256/320/384 px accordingly, to conserve as much as possible detection accuracy.

4 Results

We start by analyzing the accuracy and computations of all published pedestrian detectors from recent years (see Figure 1 and Supplementary Material). Detectors were evaluated on the reasonable setup for full-resolution images. Pedestrian detection accuracy is measured using the MR^{-2} metric [2, 13, 71]. To calculate MR^{-2}, a detector's miss-rate (mr) is plotted against the number of its false positives per image ($fppi$) for a confidence value c. This plot is created from $(mr, fppi)$ pairs for various c values. The log-average miss-rate (MR^{-2}) is:

$$MR^{-2} = exp(\frac{1}{9}\sum_{f}(mr(\underset{fppi(c)\leq f}{\text{argmax}}\ fppi(c)))) \tag{4}$$

where the nine $fppi$ points f are equally spaced in the log space, $f \in \{10^{-2}, 10^{-1.75}, ..10^{0}\}$. A detection is considered true positive if its intersection over union was higher than 0.5.

FLOPs are used for estimating computations of various architectures [22, 24, 34, 50, 55, 74]. Previus works showed a strong correlation between latency and FLOPs [48]. In contrast to latency, which is highly platform dependent, we analyze FLOPs to allow a platform-independent comparison between various solutions.

4.1 Evaluating the first-stage architecture

The C&S and CascadeMV2 architectures resulted in a B/A factor of 0.046/0.038/0.097 for the Pedestron(HRNet)/ACSP/APD detectors, showing their very lite implementations. Both architectures also achieved a sensitivity of 99% when setting k to 85% and p to 10% (see Table 1). Moreover, for pedestrians higher than 100 px the C&S and CascadeMV2 architectures resulted in a sensitivity of 98.26%/99.47% respectively, while for pedestrians smaller than 100 px they obtained a sensitivity of 99.16%/99.08%, indicating detection robustness to object sizes. These results confirm that a first-stage architecture is not unique and could be replaced by even lighter designs, as long as they meet the first-stage criteria. Sensitivity could be further increased by selecting proposals with lower confidence levels, on account of processing more proposals in the second-stage, thus allowing to balance between sensitivity and computations reduction of the next stage, without additional training.

Since the lite CNN architecture is not unique, we also studied a more naïve approach that is sometimes used in coarse-to-fine solutions. In these designs an off-the-shelf detection architecture is applied on downscaled input images [16, 58]. Similarly, we adopted the single-stage APD detector [70] since it has the lowest number of FLOPs with MR^{-2} lower than 10%. To achieve a similar number of FLOPs as our solution, input images were downscaled by a factor of 3.2 on each axis. This resulted in a decreased sensitivity of 72.2% (see Table 1), that increased the overall miss rate to 31.9% (see Table 2). This accuracy deterioration is expected and impacts most small pedestrians. Specifically, the sensitivity for pedestrians higher than 100 px was 97.34%, while the sensitivity of pedestrians smaller than 100 px was 35.58%. Even when downscaling images by a factor of two on each axis, sensitivity was reduced to 92.1%, resulting in a miss rate of 15.4%, showing that using existing detectors for direct inference on downscaled input images is not an effective solution.

4.2 Gauging the PCMAD contribution

Although produced ROIs achieved high sensitivity, the maximum relative processed area (M/N) in some images exceeds 100% (see Table 1 and Supplementary Material). For h=265 px the PCMAD algorithm was able to limit M/N in the worst-case scenario to below 0.56 for the Citypersons dataset, with an average M/N below 9.6%. Overall, PCMAD significantly reduced the amount of processed pixels by the second-stage detector, contributing to the overall BLT-net computations reduction. Downscaling mROIs did not decrease detection accuracy (see Table 1) for the selected p and h parameters, but significantly decreased M/N.

Coarse detector	Dataset	PCMAD?	Avg. M/N	Max. M/N	Sensitivity%
C&S	CP	x	0.095	1.073	98.9
	CP	✓	0.091	0.38	98.9
CascadeMV2	CP	x	0.148	1.939	99.2
	CP	✓	0.096	0.56	99.3
	CT	x	0.13	1.31	98.58
	CT	✓	0.12	0.95	98.94
APD×1/3.2	CP	✓	0.095	0.55	72.2
APD×1/2	CP	✓	0.12	0.64	92.1

Proposals extension p	mROIs downscaled?	MR^{-2}	Avg. M/N
5%	✓	9.88%	0.085
10%	✓	9.39%	0.091
15%	✓	9.19%	0.105
20%	✓	8.8%	0.127
10%	x	9.37%	0.128

Table 1: Gauging PCMAD impact. Left: Relative processed area (M/N) and sensitivity of C&S and CascadeMV2, calculated for h=256 px, k=85% and p=10%, evaluated on both Citypersons (CP) and Caltech (CT) datasets. Right: Ablation study on the effect of p and mROIs downscaling on MR^{-2} and M/N, using C&S and Pedestron (HRNet) Citypersons.

Detection accuracy was impacted by p, where an increased p improved detection accuracy on account on increasing the M/N factor. Based on this analysis, we select for all the next analyses p=10%, that balances between a low MR^{-2} and a low N/M factor.

The number of mROIs divided by the number of pedestrians (Citypersons) when using the C&S / CascadeMV2 was smaller than 1.26/1.31 respectively for the different second-stage detectors. These results indicate that the first-stage was able to optimize the number of mROIs, similarly to the number of ground truth objects, as opposed to the thousands of ROIs used in general two-stage detectors [18, 47].

4.3 BLT-net impact on computations and detection accuracy

Overall, BLT-net significantly reduced the computations of the evaluated pedestrian detection architectures by a factor of ×4-×7 (see Table 2). Computations reduction was most significant for Pedestron, which had the highest detection accuracy for small pedestrians, that in turn enabled a more significant mROI downscaling. The total number of FLOPs when using Pedestron or APD in BLT-net second-stage was below 83 GFLOPs, well below all published pedestrian detectors so far. BLT-net increased MR^{-2} by 1.5%-2.3% when using C&S and by 1.5%-2.64% when using CascadeMV2 for the Citypersons dataset with respect to the original architectures, with Pedestron being the least impacted detector and APD the most impacted detector. Similar results were obtained for the Caltech dataset.

Although $\sim$ 1% of misses resulted from the first-stage detector, some additional misses resulted from applying the fine pedestrian detectors on mROIs. Such misses are characterized mostly by (a) pedestrians appearing in crowds/occluded or (b) small and poorly illuminated pedestrians. This deterioration is also reflected on the MR^{-2} of the heavy occluded pedestrians, (see Table 2). All additional misses created by BLT-net's first and second stages on the Citypersons validation dataset can be found in the Supplementary Material. Accuracy degradation in such scenarios was shown to improve by extending proposals margins (p) on account of increasing second-stage computations, as seen in Table 1. Such parameters optimization can be used for balancing between accuracy and computations reduction, without any additional training. Alternatively, detection accuracy could be improved by training second-stage detectors directly on mROIs [23, 32, 66] for given p and h parameters. Finally, in commercial solutions, accuracy degradation could be also compensated by already existing tracking modules used for ensuring detection stability between frames [9].

4.4 Determining the effectiveness of BLT-net on other datasets

The computation reduction factor depends on both A/B and M/N. While A/B solely depends on the selected first and second-stage architectures, M/N highly depends on the processed dataset characteristics, *e.g*. the amount of objects, their size and overlapping extent. To evaluate the effectiveness of BLT-net, an initial evaluation could be conducted per dataset by using only ground truth annotations (GTs; simulating maximum true positives). Specifically, GTs padding and downscaling bounded to a mROI of size $h \times w$ simulates PCMAD without its merging function. Given the first-stage average $fppi$ (false positives), M could be estimated using $(\#GTs + fppi \cdot \#images) \cdot h \cdot w$.

The Citypersons validation dataset contains 500 images and 1,579 reasonable objects. To achieve the defined sensitivity of the first-stage, for CascadeMv2 detections confidence was set to c=0.3, resulting in a $fppi$ of 2.4. For the obtained $fppi$ and a mROI of 256×128 px, M/N resulted in 8.7%, leading to similar results on par with the ones presented in Table 1. A similar analysis was conducted for the COCO dataset [36]. This dataset contains 118,287 images with a median size of 640×480 px and 849,941 objects (training dataset). For a selected mROI of 96×96 px and same $fppi$, M/N=28.7% and for a selected mROI of 64×64 px, M/N=12.8%, suggesting that this computations reduction method could be applied also to more general dataset types. The final computations reduction factor is determined by the actual $fppi$ (that meets the required sensitivity criterion), effect of merging overlapping proposals, used mROI size (determined by the second-stage detector ability) and A/B factor.

5 Conclusions

In this work we revise the two-stage architecture by leveraging object scales and coverage characteristics to reduce computations. Our design significantly reduced computations of existing pedestrian detectors while achieving similar detection accuracy, creating a new empirical Pareto frontier. This architecture could be applied on other object detection scenarios to benefit from significant computations reduction.

Table 2: Detection accuracy measured by MR^{-2} (smaller is better) and FLOPs of original pedestrian detection architectures and when integrated into BLT-net on full images. Small/large pedestrians are defined to be lower/higher than 100 px. Reasonable/bare/partial/heavy are as defined by the dataset.

Detector	Dataset	BLT-net first stage	GFLOPs	FLOPs reduct. factor	MR^{-2} Reas. %	MR^{-2} Bare %	MR^{-2} Partial %	MR^{-2} Heavy %	MR^{-2} Large %	MR^{-2} Small %
Pedestron	CP	-	596.8	-	7.6	5.8	6.3	34.3	5.4	7.6
(HRNet)	CP	C&S	82.2	7.3	9.4	7.3	8.3	38.5	7	9.5
(2-stage)	CP	CascMv2	82.7	7.2	9.1	6.9	8.2	38.9	6.6	8.8
Pedestron	CP	-	262.2	-	10.2	7.4	9.1	43.8	7.2	10.1
(mv2)	CP	C&S	51.5	5.1	11.7	8.2	11.4	48.2	8.1	12.6
(2-stage)	CP	CascMv2	50.2	5.2	11.8	8.2	10.9	47.4	8.6	10.7
ACSP	CP	-	730.5	-	7.6	4.9	6.9	42.2	4.7	7.2
(1-stage)	CP	C&S	136.7	5.3	9.6	6.3	9.3	47.5	5.9	10.6
	CP	CascMv2	128.2	5.7	9.2	5.9	9.0	45.4	5.6	10.1
APD	CP	-	283.3	-	9	5.2	8.98	47.0	4.5	9.8
(1-stage)	CP	C&S	75	3.8	11.3	7.8	10.1	51.3	7.8	11.6
	CP	CascMv2	72.7	3.9	11.6	7.3	10.7	50.0	7.4	11.7
	CP	APD×1/3.2	69.5	4.1	31.9	27.7	31.9	55.9	8.2	61.5
	CP	APD×1/2	125.5	2.3	15.4	10.7	15.2	47.2%	6.8	22.2
Pedestron	CT	-	87.4	-	2.22	-	-	-	-	-
(HRNet)	CT	CascMv2	13.87	6.3	3.63	-	-	-	-	-

References

[1] Anelia Angelova, Alex Krizhevsky, Vincent Vanhoucke, Abhijit Ogale, and Dave Ferguson. Real-time pedestrian detection with deep network cascades. 2015.

[2] Markus Braun, Sebastian Krebs, Fabian Flohr, and Dariu M Gavrila. Eurocity persons: A novel benchmark for person detection in traffic scenes. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1844–1861, 2019.

[3] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[4] Chen Chen, Mengyuan Liu, Xiandong Meng, Wanpeng Xiao, and Qi Ju. Refinedetlite: A lightweight one-stage object detection framework for cpu-only devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 700–701, 2020.

[5] Dong Chen, Gang Hua, Fang Wen, and Jian Sun. Supervised transformer network for efficient face detection. In *European Conference on Computer Vision*, pages 122–138. Springer, 2016.

[6] Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*, 2017.

[7] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, Xudong Zou, et al. Pedhunter: Occlusion robust pedestrian detector in crowded scenes. In *AAAI*, pages 10639–10646, 2020.

[8] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018.

[9] Qi Chu, Wanli Ouyang, Hongsheng Li, Xiaogang Wang, Bin Liu, and Nenghai Yu. Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. In *Proceedings of the IEEE international conference on computer vision*, pages 4836–4845, 2017.

[10] Michael F Cloutier, Chad Paradis, and Vincent M Weaver. A raspberry pi cluster instrumented for fine-grained power measurement. *Electronics*, 5(4):61, 2016.

[11] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[13] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311. IEEE, 2009.

[14] Thomas Elsken, Jan Hendrik Metzen, Frank Hutter, et al. Neural architecture search: A survey. *J. Mach. Learn. Res.*, 20(55):1–21, 2019.

[15] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[16] Mingfei Gao, Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Dynamic zoom-in network for fast object detection in large images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6926–6935, 2018.

[17] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[18] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[19] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[20] Irtiza Hasan, Shengcai Liao, Jinpeng Li, Saad Ullah Akram, and Ling Shao. Pedestrian detection: The elephant in the room. *arXiv preprint arXiv:2003.08799*, 2020.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 558–567, 2019.

[23] Sungeun Hong, Sungil Kang, and Donghyeon Cho. Patch-level augmentation for object detection in aerial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[24] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1314–1324, 2019.

[25] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[26] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311, 2017.

[27] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and<0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[28] Andrey Ignatov, Radu Timofte, Andrei Kulik, Seungsoo Yang, Ke Wang, Felix Baum, Max Wu, Lirong Xu, and Luc Van Gool. Ai benchmark: All about deep learning on smartphones in 2019. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3617–3635. IEEE, 2019.

[29] Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical report, UMass Amherst technical report, 2010.

[30] Soowon Kang, Hyeonwoo Choi, Sooyoung Park, Chunjong Park, Jemin Lee, Uichin Lee, and Sung-Ju Lee. Fire in your hands: Understanding thermal behavior of smartphones. In *The 25th Annual International Conference on Mobile Computing and Networking*, pages 1–16, 2019.

[31] MY Lee, J Chung, JH Lee, JH Han, and YS Kwon. Trends in ai processor technology. *Electronics and Telecommunications Trends*, 35(3):66–75, 2020.

[32] Changlin Li, Taojiannan Yang, Sijie Zhu, Chen Chen, and Shanyue Guan. Density map guided object detection in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 190–191, 2020.

[33] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5325–5334, 2015.

[34] Yixing Li and Fengbo Ren. Light-weight retinanet for object detection. *arXiv preprint arXiv:1905.10011*, 2019.

[35] Shih-Chieh Lin, Yunqi Zhang, Chang-Hong Hsu, Matt Skach, Md E Haque, Lingjia Tang, and Jason Mars. The architectural implications of autonomous driving: Constraints and acceleration. In *Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 751–766, 2018.

[36] 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.

[37] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[38] Songtao Liu, Di Huang, and Yunhong Wang. Adaptive nms: Refining pedestrian detection in a crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6459–6468, 2019.

[39] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[40] Wei Liu, Shengcai Liao, Weidong Hu, Xuezhi Liang, and Xiao Chen. Learning efficient single-stage pedestrian detectors by asymptotic localization fitting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 618–634, 2018.

[41] Wei Liu, Shengcai Liao, Weiqiang Ren, Weidong Hu, and Yinan Yu. High-level semantic feature detection: A new perspective for pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5187–5196, 2019.

[42] Mahyar Najibi, Bharat Singh, and Larry S Davis. Autofocus: Efficient multi-scale inference. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9745–9755, 2019.

[43] Marina Neseem and Sherief Reda. Adacon: Adaptive context-aware object detection for resource-constrained embedded devices. *arXiv preprint arXiv:2108.06850*, 2021.

[44] Yanwei Pang, Jin Xie, Muhammad Haris Khan, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Ling Shao. Mask-guided attention network for occluded pedestrian detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4967–4975, 2019.

[45] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[46] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[47] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[48] Albert Reuther, Peter Michaleas, Michael Jones, Vijay Gadepally, Siddharth Samsi, and Jeremy Kepner. Survey and benchmarking of machine learning accelerators. *arXiv preprint arXiv:1908.11348*, 2019.

[49] A. Royer and Christoph H. Lampert. Localizing grouped instances for efficient detection in low-resource scenarios. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1716–1725, 2020.

[50] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[51] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[52] Karen Simonyan and Andrew Zisserman. Vgg-16. *arXiv preprint arXiv:1409.1556*, 2014.

[53] Guanglu Song, Yu Liu, Ming Jiang, Yujie Wang, Junjie Yan, and Biao Leng. Beyond trade-off: Accelerate fcn-based face detector with higher accuracy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7756–7764, 2018.

[54] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[55] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10781–10790, 2020.

[56] Ujjwal Ujjwal, Aziz Dziri, Bertrand Leroy, and Francois Bremond. A one-and-half stage pedestrian detector. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 776–785, 2020.

[57] Burak Uzkent and Stefano Ermon. Learning when and where to zoom with deep reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12345–12354, 2020.

[58] Burak Uzkent, Christopher Yeh, and Stefano Ermon. Efficient object detection in large images using deep reinforcement learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1824–1833, 2020.

[59] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[60] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Scaled-yolov4: Scaling cross stage partial network. *arXiv preprint arXiv:2011.08036*, 2020.

[61] Hanrui Wang, Zhanghao Wu, Zhijian Liu, Han Cai, Ligeng Zhu, Chuang Gan, and Song Han. Hat: Hardware-aware transformers for efficient natural language processing. *arXiv preprint arXiv:2005.14187*, 2020.

[62] Wenhao Wang. Adapted center and scale prediction: More stable and more accurate. *arXiv preprint arXiv:2002.09053*, 2020.

[63] Xinlong Wang, Tete Xiao, Yuning Jiang, Shuai Shao, Jian Sun, and Chunhua Shen. Repulsion loss: Detecting pedestrians in a crowd. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7774–7783, 2018.

[64] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3974–3983, 2018.

[65] Jin Xie, Yanwei Pang, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Ling Shao. Psc-net: Learning part spatial co-occurence for occluded pedestrian detection. *arXiv preprint arXiv:2001.09252*, 2020.

[66] Fan Yang, Heng Fan, Peng Chu, Erik Blasch, and Haibin Ling. Clustered object detection in aerial images. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8310–8319, 2019.

[67] Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5525–5533, 2016.

[68] Zhenheng Yang and Ramakant Nevatia. A multi-scale cascade fully convolutional network face detector. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 633–638. IEEE, 2016.

[69] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018.

[70] Jialiang Zhang, Lixiang Lin, Yang Li, Yun-chen Chen, Jianke Zhu, Yao Hu, and Steven CH Hoi. Attribute-aware pedestrian detection in a crowd. *arXiv*, pages arXiv–1910, 2019.

[71] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. Citypersons: A diverse dataset for pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3221, 2017.

[72] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Occlusion-aware r-cnn: Detecting pedestrians in a crowd. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 637–653, 2018.

[73] Shifeng Zhang, Yiliang Xie, Jun Wan, Hansheng Xia, Stan Z Li, and Guodong Guo. Widerperson: A diverse dataset for dense pedestrian detection in the wild. *IEEE Transactions on Multimedia*, 22(2):380–393, 2019.

[74] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[75] Xiaofan Zhang, Haoming Lu, Cong Hao, Jiachen Li, Bowen Cheng, Yuhong Li, Kyle Rupnow, Jinjun Xiong, Thomas Huang, Honghui Shi, et al. Skynet: a hardware-efficient method for object detection and tracking on embedded systems. *arXiv preprint arXiv:1909.09709*, 2019.

[76] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, Jianping Shi, and Jiaya Jia. Icnet for real-time semantic segmentation on high-resolution images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–420, 2018.

[77] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2879–2886. IEEE, 2012.