SMARTADAPT: Multi-branch Object Detection Framework for Videos on Mobiles

Ran Xu*, Fangzhou Mu^, Jayoung Lee*, Preeti Mukherjee*, Somali Chaterji*, Saurabh Bagchi*, and Yin Li^
* Purdue University ^ University of Wisconsin-Madison

Introduction

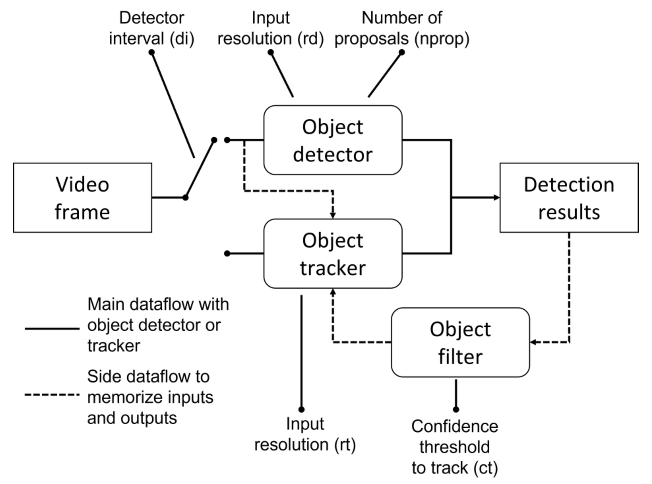
Multi-branch Object Detection Framework (MBODF): A solution that has many execution branches and one can dynamically choose from among them at inference time to satisfy varying latency requirements.

Key Observation: If one is allowed to select, at inference time, from a large set of fine-grained execution branches, the detection accuracy and latency can be significantly improved.

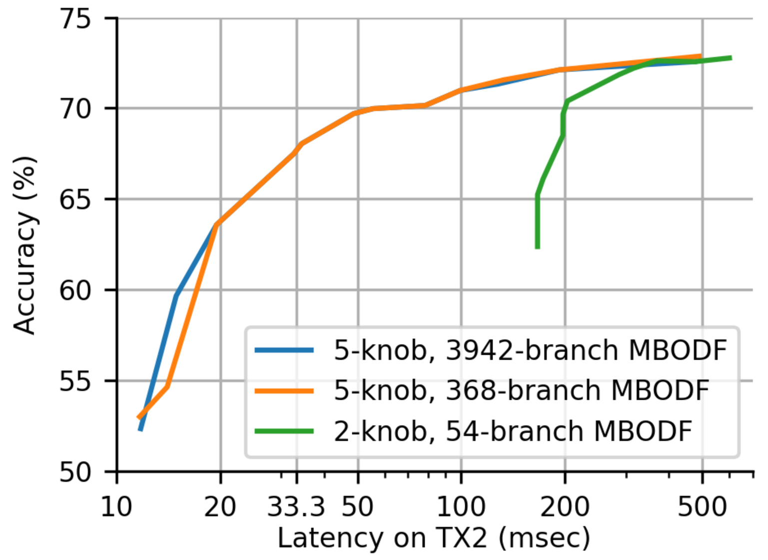
Problem Statement:
- How to expose the right set of execution branches in many existing object detectors
- How to schedule the optimal one at inference time

Multi-branch Object Detection Framework (MBODF)

- Tracking-by-detection scheme
- Tuning knobs at inference time
  - Detector interval (di)
  - Input resolution of the detector (rd)
  - Number of proposals ($nprop$)
  - Input resolution of the tracker (rt)
  - Confidence threshold to track (ct)



- MBODF: multi-knob tracking-by-detection scheme, with the range and step sizes for each knob
  - More knobs are better.
    - 5 knobs: 6.1x speed up with 2.4% mAP reduction
    - 2 knobs: 3.0x speed up with 2.4% mAP reduction
  - More branches are slight better.
    - Because of lack of smarts in choosing the execution branch conditioned on the video content.

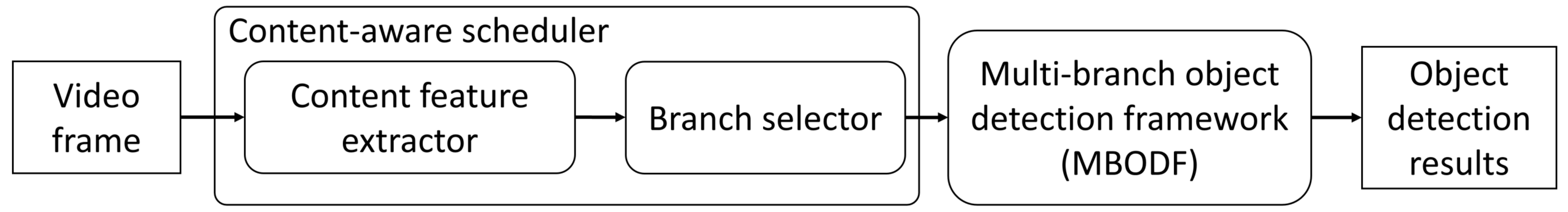


Branch Selection Problem

Optimization Goal of the Branch Selection Based on the Content:

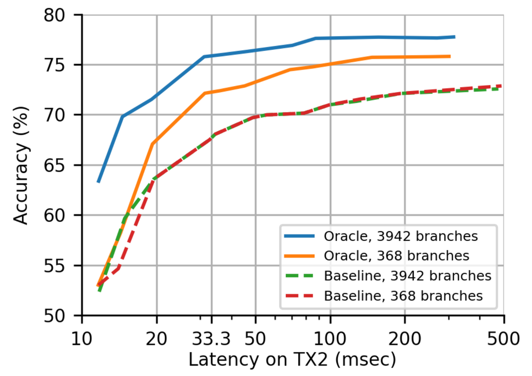
$$b_{opt} = \operatorname*{argmax}_{b} a(b, \hat{X}), s.t.\ l(b, \hat{X}) \le l_0$$

The Workflow of SMARTADAPT:
- The scheduler extracts the content features.
- The scheduler predicts the accuracy with a content-aware accuracy predictor.
- The scheduler uses a branch selector to choose the optimal branch.
- The MBODF takes the branch and detects objects in the video frame.



Content-aware Scheduler (CAS)

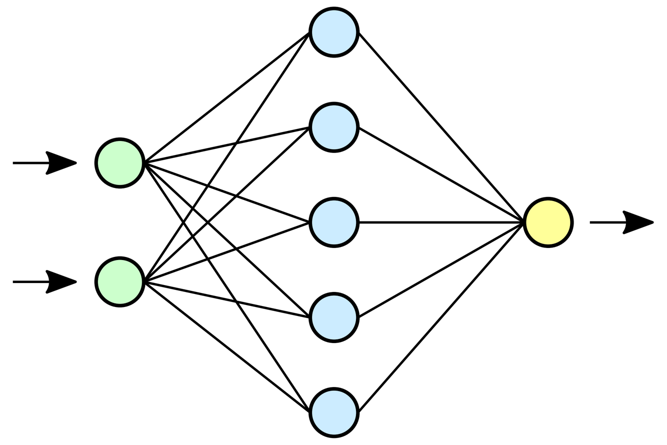
Oracle CAS: An Upper Bound
- 368-branch: 3.2%-4.6% mAP above w/o CAS
- 3942-branch: 6.6%-8.3% mAP above w/o CAS



Our CAS:
- Content feature extractors
- Content-aware accuracy predictor
  - 5-layer fully connected neural network (NN)
  - Multi-layer perceptrons (MLPs): joint modeling of content and latency requirement
- Candidate branches

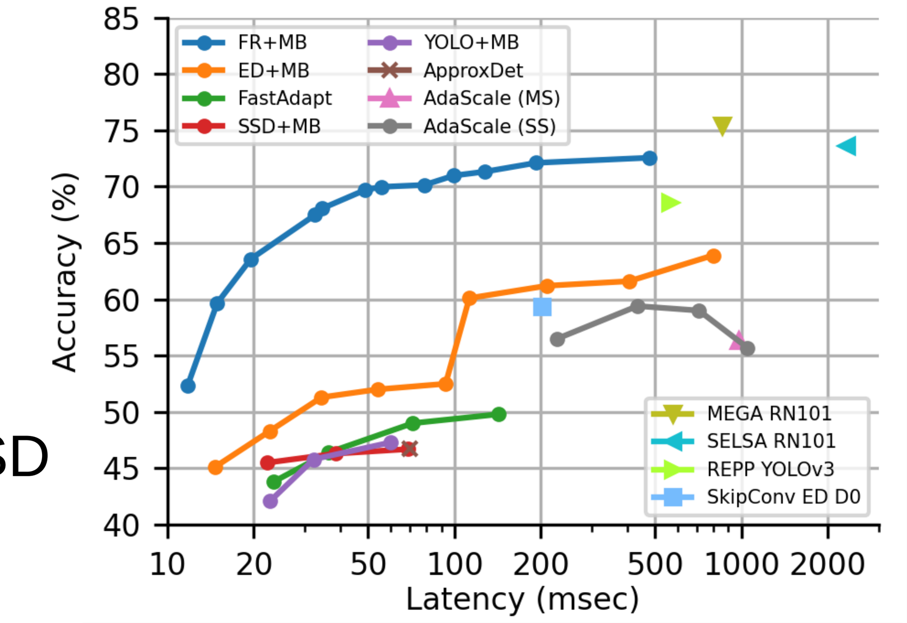
$$b_{opt} = \operatorname*{argmax}_{b} a(b, f(\hat{X})), s.t.\ l(b, \hat{X}) \le l_0$$

Name	Dim.	Trainable	Description
light	4	No	Composed of height, width, number of objects, averaged size of the objects
HoC	768	No	Histograms of Color on red, green, blue channels
HOG	5400	No	Histograms of Oriented Gradients
ResNet50	1024	No	ResNet50 features from the object detector in the MBODF, average pooled over height and width dimensions, and only preserving the channel dimension
CPoP	31	No	"Class Predictions on the Proposal" feature (CPoP) from the object detector of the MBODF, averaged pooled over all region proposals, and only preserving the class dimension (including a background class)
MobileNet	1280	**Yes**	Efficient, effective feature extractor, average pooled from the feature map before the fully-connected layer, and only preserving the channel dimension



Experimental Evaluation (ILSVRC VID)

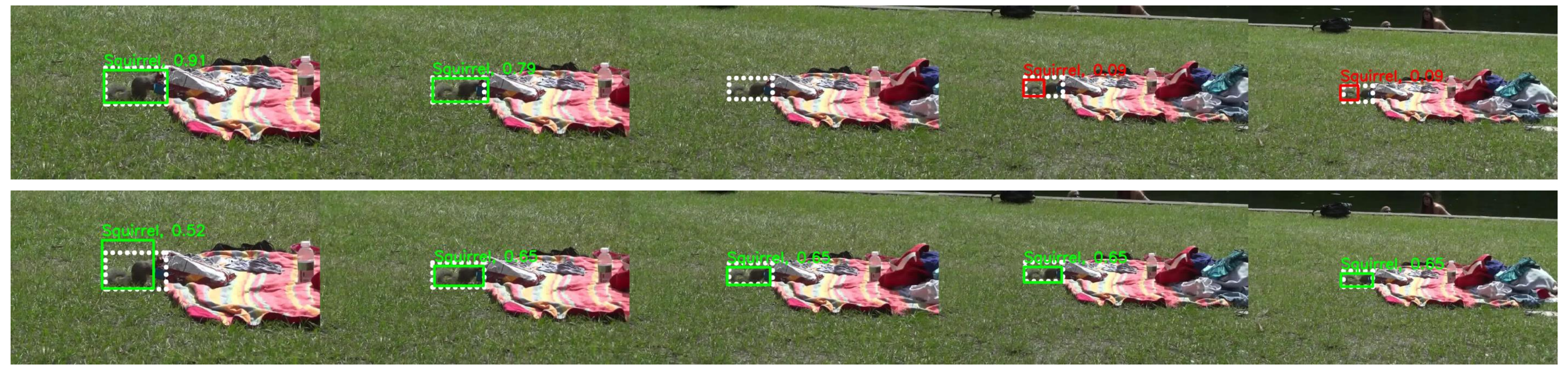
Better Accuracy-latency Frontiers
- 67.5%/69.7%/71.0% mAP at 30/20/10 FPS
- 40.5x adaptation range in latency 9.8x within 3% mAP reduction
- The MBODF enhances FRCNN, EfficientDet, YOLO, and SSD



Protocols	20.0 ms	33.3 ms	50 ms	100 ms
FR+MB+Oracle (3,942 br.)	71.5%	75.8%	76.3%	77.6%
FR+MB+Oracle (368 br.)	67.1%	72.1%	72.9%	74.8%
FR+MB+CAS	**64.1%**	**68.3%**	**69.8%**	**71.1%**
FR+MB	63.6%	67.5%	69.7%	71.0%
FastAdapt+CAS	N/A	**46.1%**	**47.1%**	**50.3%**
FastAdapt	N/A	43.8%	46.4%	49.0%
ED+MB	45.1%	51.3%	52.0%	52.5%
SSD+MB	N/A	45.5%	46.3%	46.7%
YL+MB	N/A	42.1%	45.8%	47.3%
ApproxDet	N/A	N/A	N/A	46.8%

Our CAS on top of MBODFs
- FR+MB+CAS +0.1%-0.8% mAP.
- FastAdapt+CAS +0.7%-2.3% mAP.

Visual Results (50 FPS Speed)
- FR(FRCNN)+MB (top figure)
  - Chooses $di = 20, rd = 288, nprop = 100, rt = 25\%, ct = 0.05$
  - Misses on the 30th frame and localizes wrongly on the 40th and 50th
  - Due to the small resolution and too frequent calibration with the OD.
- FR(FRCNN)+MB+CAS (bottom figure)
  - Chooses $di = 50, rd = 384, nprop = 100, rt = 25\%, ct = 0.05$
  - Detects correctly on all frames
  - Due to the large resolution and using object tracker on more frames.



Key Takeaways

- MBODF for object detection -- higher accuracy and stronger adaptation
- Scheduling the optimal branch conditioned on the video content
- Much higher accuracy upper bound (oracle CAS) and moderate accuracy improvement (our CAS)

Computer Vision and Pattern Recognition (CVPR), 2022 Acknowledge to NSF CCF-1919197, CNS-2038986, CNS-2038566, CNS-2146449, AWS AI award, ARL W911NF-20-2-0026, W911NF-2020-221 xu943@purdue.edu