Friedrich-Alexander-Universität
Technische Fakultät

µYOLO: Towards Single-Shot Object Detection on Microcontrollers

Mark Deutel[1,2], Christopher Mutschler[2], and Jürgen Teich[1]

1. Friedrich-Alexander-Universität Erlangen-Nürnberg
2. Fraunhofer Institute for Integrated Circuits, Fraunhofer IIS

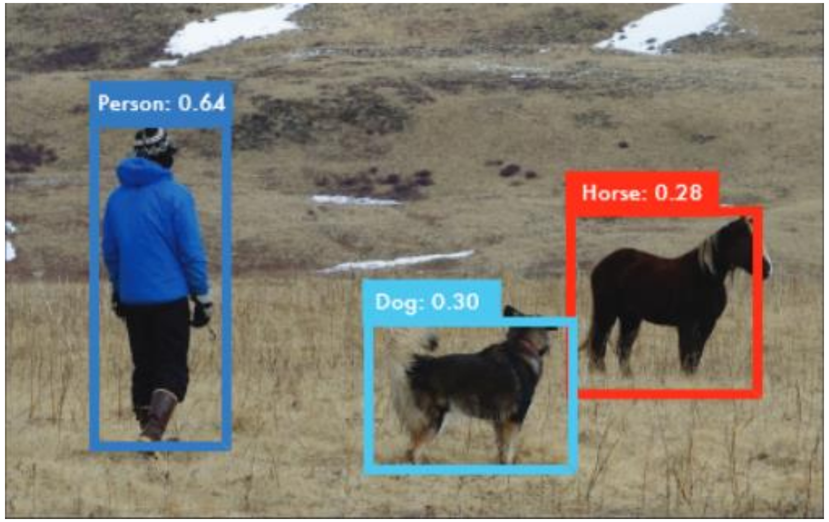
Outline

Object Detection

Introduction

Objective: Detect and Classify Objects in Scenes

- Two problems to solve:
  - Bounding Box Regression (where and how many objects)
  - Classification (what kind of objects)
- There can be any number of objects in a scene
- **Region-based CNNs** (R-CNNs):
  - Select a set of **region proposals** from the image using a **selective search algorithm**
  - **Classify** each proposed region with a **CNN**
  - Fast R-CNN[1], Faster R-CNN[2], Mask R-CNN[3], Mesh R-CNN[4], …



Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

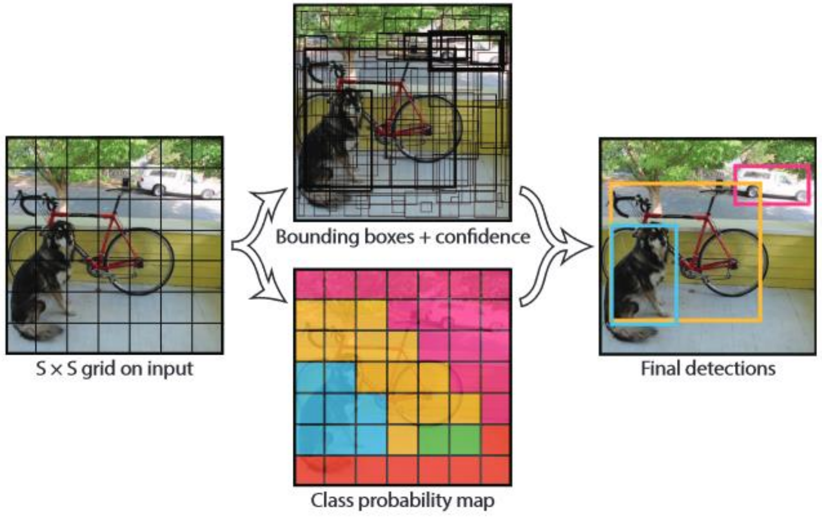
1. Girshick, Ross. "Fast r-cnn." *Proceedings of the IEEE international conference on computer vision*. 2015.
2. Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* 28. 2015.
3. He, Kaiming, et al. "Mask r-cnn." *Proceedings of the IEEE international conference on computer vision*. 2017.
4. Gkioxari, Georgia, Jitendra Malik, and Justin Johnson. "Mesh r-cnn." *Proceedings of the IEEE/CVF international conference on computer vision*. 2019.

Object Detection

Single-Shot Detectors

Can we solve object detection “in one go”?

- Yes, we can: **YOLO[1] – You only look once**
  - Combines region proposal and classification in a single CNN using a grid-based approach → only one “forward pass” required
  - Each grid cell proposes a number of regions (bounding boxes) and a confidence score for each of them
  - CNN is trained with a combined, additive loss function
- Improvement over time:
  - YOLOv5/v8[2], and recently YOLOv7[3]



Center Point

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

Size

$$+ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

Confidence

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2$$

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2$$

Classification

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

1. Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
2. https://ultralytics.com/
3. Wang, Chien-Yao, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023.

Object Detection

Single-Shot Detectors on Microcontrollers

Single Shot Detectors (SSDs) are well suited for use on embedded systems

- **Efficient** and **versatile** method for solving detection problems with remarkable precision
- Since “any” CNN can be trained as an SSD, they easily **scale** to **meet resource constraints**
- Widely used, especially on GPU/TPU-accelerated embedded systems[1,2]
- However, little research has been done exploring the feasibility of SSDs on microcontrollers

Yolov5 nano, https://github.com/ultralytics/yolov5

- Today’s topic: A highly **compact CNN architecture** “µYOLO” for efficient object detection on **Cortex-M based microcontrollers**

1. Chiu, Yu-Chen, et al. "Mobilenet-SSDv2: An improved object detection model for embedded systems." *2020 International conference on system science and engineering (ICSSE)*. IEEE, 2020.
2. Wong, Alexander, et al. "YOLO nano: A highly compact you only look once convolutional neural network for object detection." *2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition (EMC2-NIPS)*. IEEE, 2019.

µYOLO: A SSD for Microcontrollers

Architecture Design and Training

- Compact **CNN architecture**
  - **1 convolution** followed by **7 depth-wise separable convolutions** with **ReLU** and **batch normalization**
  - **Linear** object detection head
  - Designed to achieve **3-5 FPS** on **OpenMV Cam H7 R2**[1]
  - **Training** and **Compression**
    - Can be trained with any regular „YOLO loss function" on any object detection dataset, e.g. COCO, Pascal VOC, …
    - We apply **filter pruning** and **weight quantization** to further optimize the architecture and perform deployment using our own pipeline and framework[2,3]

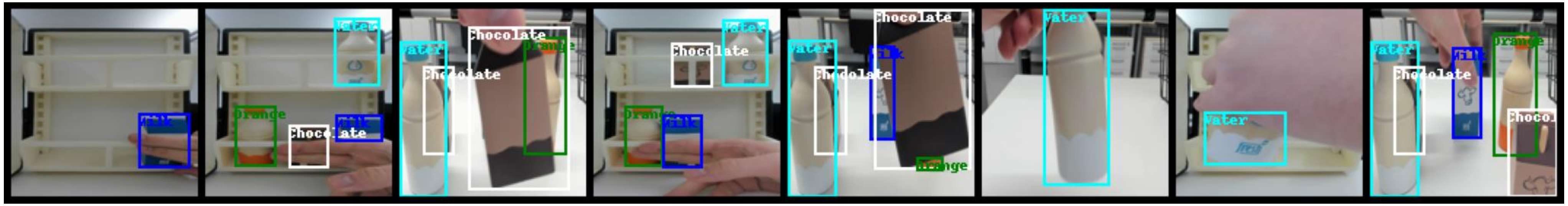
µYOLO

Input: $3 \times 128 \times 128$
C: [3, 64, 4, 2, 0]
MaxPool: [2]
D: [64, 128, 3, 1, 0] D: [128, 128, 3, 1, 1] D: [128, 128, 3, 1, 0]
MaxPool: [2]
D: [128, 128, 3, 1, 1] D: [128, 64, 3, 1, 0] D: [64, 64, 3, 1, 1] D: [64, 64, 3, 1, 0]
MaxPool: [2]
L: [1024, 1024] L: [1024, $S \times S \times N + B * 5$]
Output: $S \times S \times N + B * 5$

S: grid size, N: num. classes, B: num. bounding boxes

1. STM32H743VI ARM Cortex M7 processor @480 MHz, 1MB SRAM, and 2MB Flash
2. Deutel, Mark, et al. "Energy-efficient Deployment of Deep Learning Applications on Cortex-M based Microcontrollers using Deep Compression." *MBMV 2023; 26th Workshop*. VDE, 2023.
3. Deutel, Mark, et al. "Multi-Objective Bayesian Optimization of Deep Neural Networks for Deployment on Microcontrollers." *WEML 2023; 4th Workshop*. 2023

Datasets



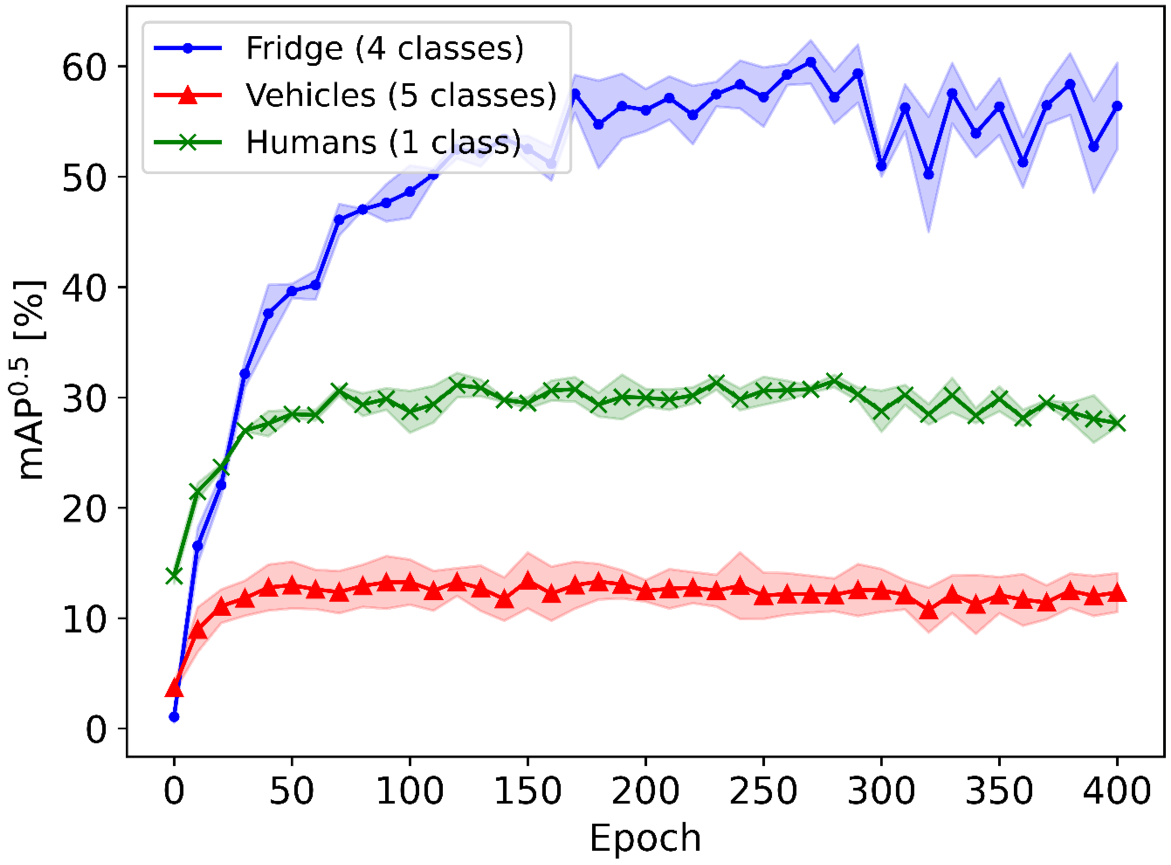
Fridge: 4 classes (white – chocolate, blue – milk, green – orange juice, cyan – water)

Vehicles: 5 classes (red – bicycle, blue – car, green – motorcycle, cyan – bus, orange – truck)

Humans: 1 class (red – human)

Experimental Results

Performance on Datasets

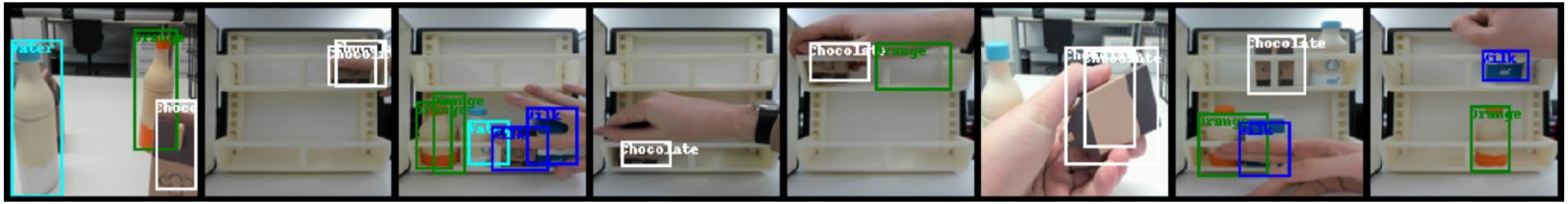


	Fridge	Vehicles	Humans
Flash	752 KB	771 KB	706 KB
RAM	324 KB	324 KB	324 KB
FPS	3.46	3.45	3.49

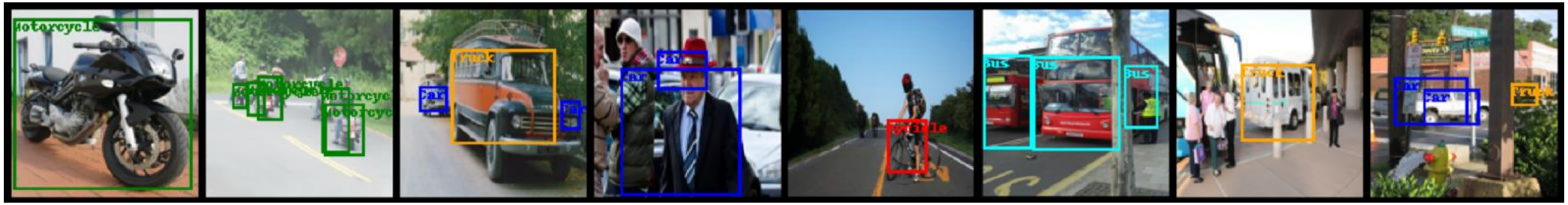
- With increasing scene complexity present in the datasets detection performance (mAP) degrades significantly
- However, qualitatively µYOLO seems to perform equally well for all datasets
- The main problem seems to be that µYOLO “misses” extremely small objects, or objects that are surrounded or partially occluded by other objects

Experimental Results

Qualitative Evaluation



Fridge: 4 classes (white – chocolate, blue – milk, green – orange juice, cyan – water)



Vehicles: 5 classes (red – bicycle, blue – car, green – motorcycle, cyan – bus, orange – truck)

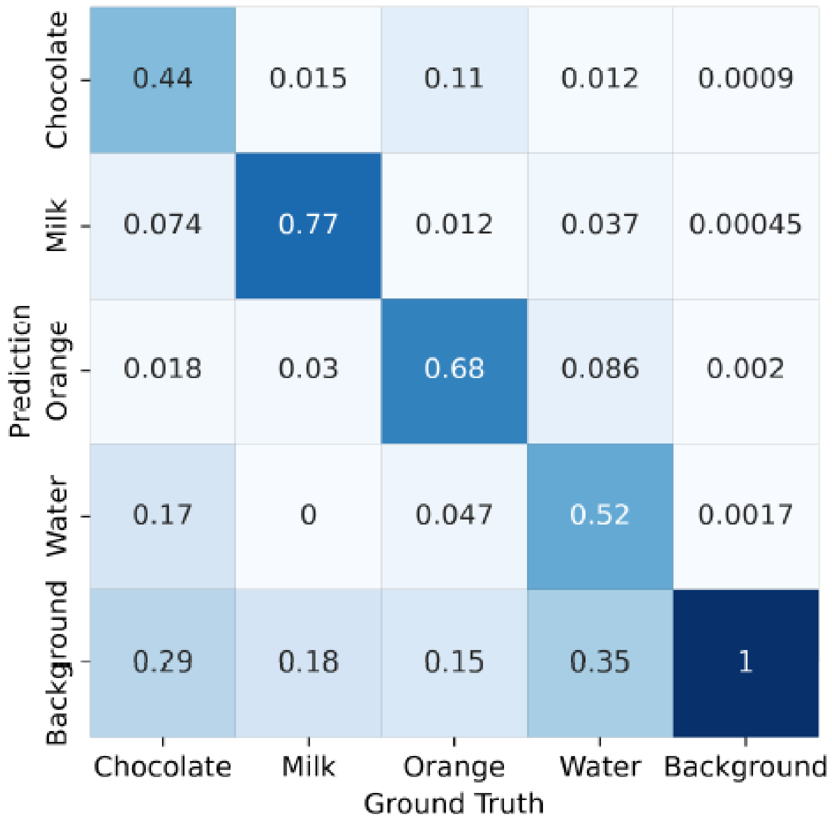
Humans: 1 class (red – human)

Error Analysis

Confusion Matrices

- Errors can be analyzed using a confusion matrix
  - Rows describe predicted bounding boxes by class, columns describe ground truth boxes by class
  - The diagonal of the matrix describes correctly detected and classified bounding boxes
  - The last row denotes false negative bounding box detections
  - The last columns describe false positive detections
  - All other fields describe correctly detected but misclassified bounding boxes

Fridge Dataset (4 classes)



Error Analysis

Complex vs. Simple Scenes

Vehicles Dataset (5 classes)

Prediction \ Ground Truth	Bycicle	Car	Motorcycle	Bus	Truck	Background
Bycicle	0.14	0.002	0.051	0	0	0.00046
Car	0.025	0.15	0.032	0.043	0.1	0.0046
Motorcycle	0.056	0.0039	0.24	0.0066	0.0068	0.001
Bus	0.0028	0.0079	0.0023	0.23	0.018	0.00069
Truck	0.0028	0.01	0.0069	0.046	0.056	0.00071
Background	0.78	0.83	0.67	0.67	0.82	0.99

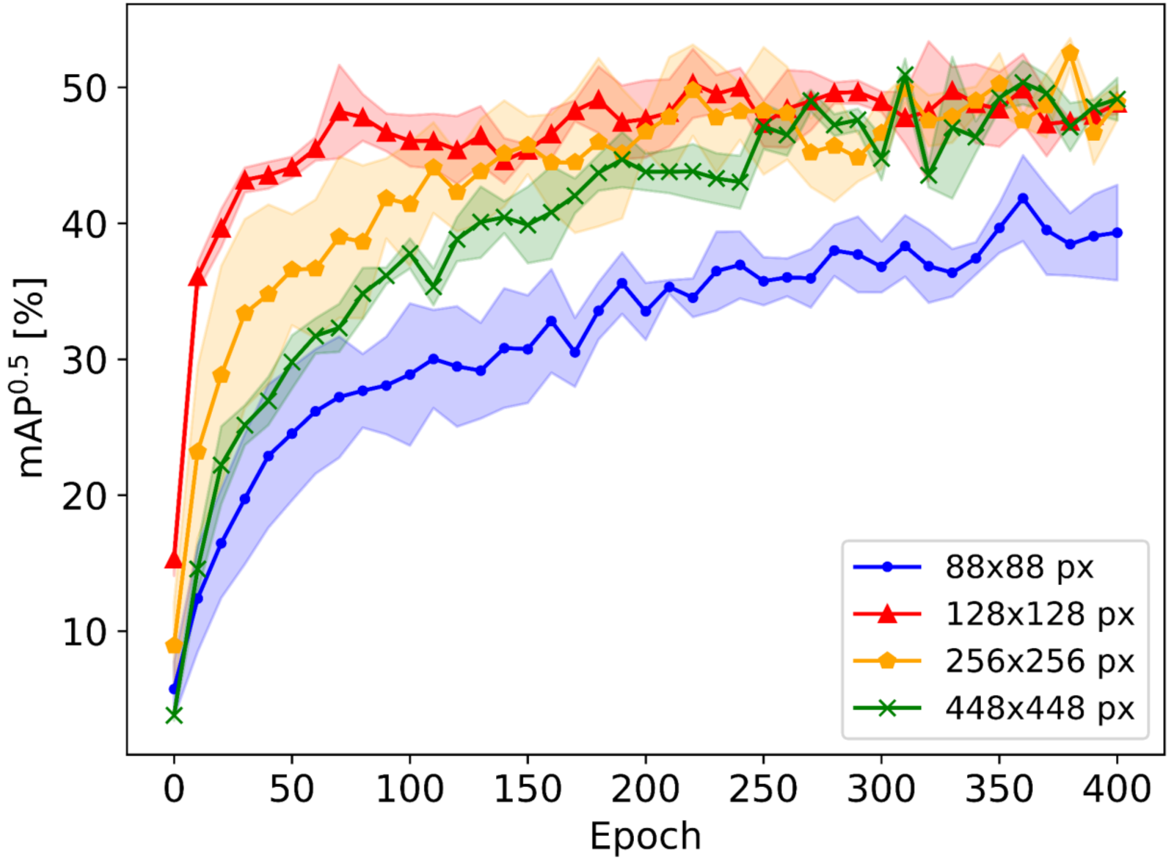
Vehicles Dataset (5 classes, max. 3 boxes)

Prediction \ Ground Truth	Bycicle	Car	Motorcycle	Bus	Truck	Background
Bycicle	0.42	0	0.013	0	0	0.00014
Car	0	0.29	0.013	0.026	0.089	0.00032
Motorcycle	0.11	0.057	0.51	0.013	0.022	0.00018
Bus	0	0.029	0	0.58	0.089	0.00046
Truck	0	0.029	0.013	0.051	0.24	0.00028
Background	0.47	0.6	0.45	0.33	0.56	1

Error Analysis

Performance for Different Image Resolutions

Vehicles Dataset (5 classes, max. 3 boxes)



	Flash	RAM	FPS
88x88 px	133 KB	152 KB	8.40
128x128 px	771 KB	324 KB	3.45
256x256 px	774 KB	1.01 MB	-
448x448 px	774 KB	3.97 MB	-

- Simplifying the dataset significantly improved mAP
- Increasing the input image resolution did not improve mAP (CNN would require more depth and capacity)
- A 128x128 pixel resolution proved to be a good compromise

Conclusion

µYOLO: A compact SSD for Microcontrollers

- Enables object detection on microcontrollers <800 KB Flash, <400 KB of RAM and at about 3-5 FPS
- It can achieve good quite good results, but be aware of its limitations …
  - Struggles with complex scenes or scenes with a lot of small objects in the background
  - As the image resolution increases, µYOLO is limited by its small model capacity, but offers an excellent trade-off between accuracy and resource consumption at lower resolutions

Thank you for your attention!

Contact: mark.deutel@fau.de