HUMAN PRESENCE DETECTION: FROM BOOSTING TO NEURAL NETWORKS
CURRENT SOLUTION

- Long running project for external customer
- Embedded camera detecting presence of people in indoor
- Mounted in approx. 2.5 m height on wall
- Our target is moving or stand still person from any side
- Many algorithms with significant computational demands
  - Motion detection
  - Online model
  - Static model
AUXILIARY ALGORITHMS

- Motion detection
  - Based on edge magnitude and angle
  - Long term detection with people like behaviour confirmation
- Online model
  - Kind of background subtraction
  - Based on HoG and Haar classifiers
  - Background online updated
  - Foreground pre-learned using boosting
MAIN ALGORITHM

- Static model
  - AdaBoost, WaldBoost, both discrete
  - Experiments with real versions of boosting, also with other classifiers (SVM, …)
  - Based on HoG and Haar (eventually MBLBP) classifiers
  - Experiments done also with more complex classifiers used as weak ones: AdaBoost, SVM, … : classifiers tree
ALGORITHM LIMITS

- Wide angle + long detection range: small detection window
- Person detection from any side, also occluded
- Detection during bad light conditions
- The base part of algorithm, static model, on the edge – model growing, small results improvements
- Experiments with model dedicated to scene
  - Better results, but difficult for users
- Decision to change completely algorithm core
CURRENT HW

- Zynq Z-7010
- 1.2 Mp CMOS camera
- IR lighting
- Power consumption below 4 Watts (2.5 W without lighting)
CURRENT HW

Zynq Z010
2xARM @800MHz

FPGA series 7

3.2GBps

LPDDR2 512 MB

FLASH 32 MB

Temp / Humidity

AR0132 CMOS camera, 1280x960

LPDDR2 128MB

10bit

3.2GBps

LAN + PoE

Accelerometer

LED

4 x IR LED

1.6GBps

KNX TP
CURRENT HW
NEURAL NETWORKS

- Single shot architecture (no location proposals, object detection in one forward pass)
- Single-class detector
- Network based on Yolo detector (by Joseph Redmon) currently gives us best results
- 22 convolutional layers, approx. 50M parameters
- Input image 480x480px
- We were able to quantize parameters to 9 bits in dynamic fixed point format with no or acceptable loss in accuracy
YOLO
FPGA IMPLEMENTATION

- First design for CNN implemented
- Consist of two computational units
- „Pseudo assembly code“ used to describe CNN layers
- Speed up in progress
- Expected speed is 1 FPS without other optimizations
- Motion detection part of implementation
OLD VS. NEW
NEXT STEPS

- Learn better models
- Re-learn quantized models
- Higher quantization