Introduction
The possibility of enabling our electronic devices to visually understand their environment gave rise to a number of novel applications. Convolutional Neural Networks (CNN) have achieved great results in many connected computer vision problems recently. However, their computational complexity makes them hard to train, and their deployment on embedded platforms particularly difficult. This Master thesis explored a wide range of different techniques addressing these problems and evaluated their performance on the example of a real industrial application.

Methodology
Optimization the training process and embedded inference of a given CNN based image classification application. Exploration of parallelism on multiple levels for accelerating the training process and reducing memory consumption:
- Mini-batch training for better vectorization
- Multi-GPU training [1]
- Online data loading and preprocessing (CPU-GPU parallelism)

Improvemnt of accuracy by extended data augmentation [2]:
- Infinite training set using on-the-fly augmentation
- Creation of artificial samples for training [3]
- Affine transformations to simulate variance in position

Acceleration of inference on an ARM Cortex-A7 based quad-core embedded processor:
- Hardware accelerated floating point calculations
- Multi-core parallelization
- SIMD execution with ARM NEON engine
- Different convolution realizations for efficient memory access patterns and reduced complexity [4][5]

Study of quantized approximation of original model [6]:
- Embedded execution using fixed-point arithmetic
- Training with quantized parameters in Torch

Acknowledgements
I would like to thank D. Atienza, R. Braojos and M. Ruggiero for their support all along this project.

References

Conclusion
- Significantly accelerated training process while improved model accuracy and reduced memory consumption
- Enabled real-time classification by exploiting hardware resources
- Demonstrated that a low-precision fixed-point implementation can be used for reducing hardware requirements

Real-Time Classification of Objects with Visual Artifacts in Multi-Core Embedded Platforms
Ikke Balasi
Ikke.balasi@epfl.ch
Supervisors: Prof. David Atienza Alonso, Dr. Rubén Braojos López
Embedded Systems Laboratory

Results

Reduced training time from 1 week to 12-24 hours:
- Eliminated data loading and preprocessing delay
- Mini-batch (3.8x) and dual-GPU (1.8x) acceleration

Reduced and highly scalable memory consumption while training

Improved model accuracy reaching 100% on testing set
Accelerated embedded inference by factor of 26 using:
- Hardware Floating Point Unit (FPU) - 4.3x
- Quad-core execution - 3.3x
- NEON acceleration - 1.8x

Selected optimal convolution implementation for different model architectures

Quantized network implementation (8-bit weights, 16-bit activations):
- No loss in testing accuracy after fine-tuning
- Reduced model size by 75% and dynamic memory consumption by 50%
- 7.6x speed up compared to 32-bit floating-point variant when no FPU available (using NEON engine)