tinyEOD: Small Deep Neural Networks and Beyond for Embedded Vision Applications

Christos Kyrkou and Theocharis Theocharides

KIOS Research and Innovation Center of Excellence and Department of Electrical and Computer Engineering, University of Cyprus

(kyrkou.christos, theocharides@ucy.ac.cy)

Summary

Dealing with computational cost of CNN
- Memory Footprint / Performance
  - Reduce the number of layers
  - Reduce number of filters
  - Reduce filter size
  - Increase stride
- Object Size
  - Affects the accuracy
  - Choose input based on object
- Input Size
  - Increase for better accuracy
  - Reduce for better performance

Case-Study: DroNet Architecture
- Trained with a custom database for vehicle detection
- Processes 512x512 images
- Make use of 3x3 filters and cheaper 1x1 convolutions
- Progressively reduce the feature maps size by a factor of 2
- Smaller number of filters at early layers

Results

Performance
- Up to 30 FPS on embedded CPU
- Comparing DroNet with different models demonstrates the effectiveness of the architecture.
  - Similar accuracy to tinyYolo + much faster.
  - 20% accuracy improvement over plain resizing approach
  - Less memory requiring only 253 KB.
- Overall, the tiling strategy can be slower than plain resizing but more efficient than processing the whole image - best trade-off between accuracy and performance

Conclusions, Ongoing and Future Work

- Deep Learning and Computer Vision are moving to the Edge
  - Drones are a prime example of a resource-constrained system with additional challenges for detectability at a distance
- Exploration of neural network architectures is key for deployment on hardware-constrained devices
  - Teacher-student models
  - Building the model ground-up: Use evolutionary methods/reinforcement to build a network with a minimal overhead
  - Apply quantization techniques
  - Investigate Cascade structures with hierarchical models.
  - Investigate real-time CNN model selection tailored to the region proposal result.
- Prior knowledge can further push performance
  - Apply informed region selection to discard regions using apriori knowledge.

Background

Why small deep-neural networks?
Small DNNs are more deployable on embedded processors
- Computation and even more so memory are at a premium
- Storing the model on chip saves on power, and improves performance
- Faster to go through training iterations
- More easily updatable over-the-air (OTA)
Small DNNs are more power efficient
- Less off-chip memory accesses which consumes order of magnitudes more power.
Small DNNs permit for multiple vision tasks to run on the same platform e.g. object detection

Single-Shot Detection
- Split the input image in a grid and for each grid generates bounding boxes and class probabilities.
- Outputs a confidence score that tells us how certain it is that the predicted bounding box encloses some object
- Predicts B bounding boxes, confidence for those boxes, and C class probabilities, encoded as an S x S x (B x 5+C) tensor
- More suitable for real-time applications

Tiling, Attention, and Memory

An object detection algorithm for UAVs that:
- Discard information and avoid unnecessary computations
- Avoid reducing the image accuracy and distorting the objects
- Make smaller objects detectable

1. Tiling
   - Separating the input image into smaller regions capable of being fed to the CNN in order to avoid resizing the input image and maintaining object resolution
2. Memory Mechanism
   - Keep track of detection metrics in each tile over time
   - Relative position of objects will not change significantly over a few successive frames
3. Attention Mechanism
   - Select which tiles to be processed by the CNN?
   - Select top N tiles above a threshold for processing based on statistical information

References


Acknowledgements

This work is funded by the European Union’s Horizon 2020 research and innovation programme under grant agreement No 739551 (KIOS CoE) and from the Republic of Cyprus through the Directorate General for European Programmes, Coordination and Development.