Motivation: The Wheel Detector

The input image, FPGA output – the classifier response map, response after spatial post-processing.

Classifier with Structured Features

1. image patch normalization
   \[ S_{x,y} = n(x,y) \cdot I_{x,y} \]
2. dot product
   \[ d_t(x,y) = \langle S_{x,y} , M_t \rangle \]
3. weak classifier
   \[ \psi_t = \begin{cases} 
   d_t, & \text{if } M_t \in \mathbb{R}^{w \times h} \\
   d_t | M_t \in \mathbb{C}^{w \times h} 
   \end{cases} \quad y_t = \begin{cases} 
   +1, & \text{if } \psi_t > t \\
   -1, & \text{if } \psi_t \leq t 
   \end{cases} \]
4. overall classifier response
   \[ r(S_{x,y}) = \sum_{t=1}^{T} \alpha_t y_t \]

Resource Consumption for different number of DPCBs for the image patch of size 25 x 29. Each block adds 11 BRAMs and about 2000 LUTs.

Conclusions

We have proposed an approach for implementing a dense linear classifier on an FPGA. The proposed architecture is scalable and quite generic, it is up to the designer how many blocks are needed to place in the FPGA. The proposed wheel classifier response map computation is used in a car detector running in an intelligent vehicle as a part of a more complicated collision mitigation system [1], that requires processing cycle of 20-30fps and a maximum latency of 200 ms.

Acknowledgement

This work was supported by the European Commission under interactive, a large scale integrated project, part of the FP7-ICT-298387 for Safety and Energy Efficiency in Mobility. The authors would like to thank all partners within interactive for their support.

References
