A Hardware-Efficient Architecture for Embedded Real-Time Cascaded Support Vector Machines Classification

Christos Kyrkou
KIOS Research Center,
University of Cyprus
kyrkou.christos@ucy.ac.cy

Theocharis Theocharides
KIOS Research Center,
University of Cyprus
ttheocharides@ucy.ac.cy

Christos-Savas Bouganis
Imperial College London
christos-savvas.bouganis@imperial.ac.uk

ABSTRACT
This work presents an optimized architecture for cascaded SVM processing, along with a hardware reduction method for the implementation of the additional stages in the cascade, leading to significant improvements. The architecture was implemented on a Virtex 5 FPGA platform and evaluated using face detection as the target application on 640×480 resolution images. Additionally, it was compared against implementations of the same cascade processing architecture but without using the reduction method, and a single parallel SVM classifier. The proposed architecture achieves an average performance of 70 frames-per-second, demonstrating a speed-up of 5× over the single parallel SVM classifier. Furthermore, the hardware reduction method results in the utilization of 43% less hardware resources, with only 0.7% reduction in classification accuracy.

Categories and Subject Descriptors
B.2.1 [**Arithmetic and Logic Structures**]: Design Styles—, *parallel*; C.3 [**Special-Purpose and Application-Based Systems**]: Real-time and embedded systems

General Terms
Performance, Design

Keywords
FPGA, Parallel Architecture, Support Vector Machines.

1. INTRODUCTION
Support vector machines (SVMs) [1] are a powerful supervised pattern recognition algorithm which has been used in object detection, amongst other applications, demonstrating high classification accuracies [1-3]. However, for large scale problems the high classification accuracy rates of SVMs come with the cost of longer classification times. The reason for this is that the run-time complexity of the algorithm is proportional to the number of support vectors; which are samples from the training set that specify the separating hyperplane that the SVM algorithm selects in order to construct its classification decision function. As such, embedded SVM-based classification systems with hundreds of support vectors, find it difficult to meet real-time processing demands, without sacrificing accuracy.

It is possible to speed-up SVM-based classification systems for a variety of applications, including object detection, by exploiting the facts that: (a) the majority of the samples presented to the classifier do not belong to the object class and (b) the majority of those samples can be easily distinguished from samples belonging to the object class. Cascade SVM classifiers, take advantage of these two observations by utilizing stages of classifiers, which are sequentially applied to the input data. The early stages, usually linear SVMs, have low complexity, meaning that they require less training data to be processed, and as such take less time to process. Conversely, the latter stages, typically RBF and polynomial kernels, have higher complexity as they require more training data to be processed, and hence have a longer classification-time. If at any stage the input data is classified as a non-object the classification process stops and the next sample is processed, otherwise it must go all the stages to be classified as an object. Under this scheme the stages at the beginning of the cascade discard a large amount of input samples very fast, resulting in significant speedups over single SVM classification [2], and is highly suitable for embedded applications such as object detection where a lot of data are generated from a single image and need to be classified.

This paper proposes a dedicated hardware architecture for cascaded SVM processing and a method for reducing the hardware implementation requirements of cascaded SVMs. The hardware architecture was implemented on a Virtex-5 FPGA platform, and evaluated using face detection on 640×480 resolution images. The implementation of the proposed hardware architecture for processing the adapted SVM cascade demonstrates an average performance of 70 frames-per-second, which is a 5× speedup over a parallel single SVM classifier implementation. Furthermore, by applying the hardware reduction method, the adapted hardware architecture for the implementation of the cascade consumes 43% less custom logic resources, with only a 0.7% reduction in classification accuracy.

2. PROPOSED ARCHITECTURE AND HARDWARE REDUCTION FRAMEWORK

2.1 Cascade SVM Hardware Reduction
The proposed hardware reduction method, outlined in Figure 1, is to round-off the support vector and alpha values of the low complexity kernels with the nearest power of two values. This will result in all the multiplication operations in the SVM classification phase (the kernel dot-product calculations and computations related to the alpha coefficients) becoming shift operations. Additionally, since the support vectors and alpha coefficients are now power of two values there is no need to store the binary representations of decimal numbers but only shift data (shift amount, shift direction, and number sign). Hence, this results in an *adapted cascade SVM* with reduced storage and computational demands.

2.2 Cascade SVM Hardware Architecture
The nature of the cascaded SVM classifiers means that each stage will have less input data to process and more support vector data to process than the previous. Accordingly then, the proposed hardware architecture for the cascaded SVM classifier consists of

GLSVLSI'13, May 2–3, 2013, Paris, France.
ACM 978-978-1-4503-1902-7/13/05.

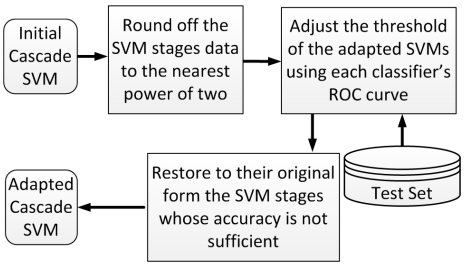

Figure 1. Cascade Hardware Reduction Method

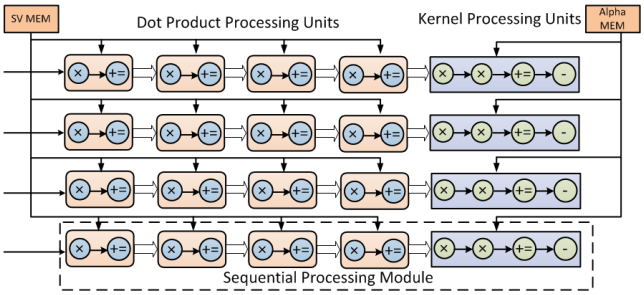

Figure 3. Architecture of single parallel SVM and Sequential Processing Module

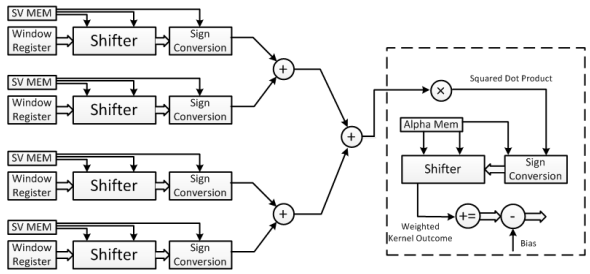

Figure 2. Architecture for the Parallel Processing Module

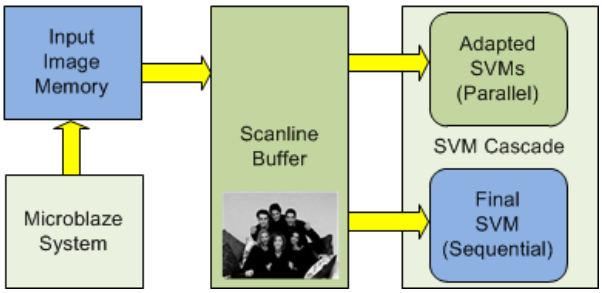

Figure 4. Cascaded Hardware Architecture for Object Detection

two main processing modules, which provide different parallelism with respect to the input data, in order to meet the different demands of the cascade stages. The first is a fully parallel module (Figure 2) which performs the processing necessary for all the adapted SVM stages and the parallelism focuses on processing all the input vector elements in parallel. The second is a sequential processing module (Figure 3), optimized for the high complexity SVM stages which demand processing a large number of support vectors and thus parallelism focuses on processing more support vectors in parallel. In addition, a shift register structure is used to provide sequential and parallel data access to the two processing modules, and also to take advantage of potential data overlap and reduce memory I/O.

3. EXPERIMENTAL PLATFORM AND RESULTS

The proposed hardware architecture (Figure 4) is evaluated using the embedded application of face detection on 640×480 resolution images [4,5] for a cascade comprised of four stages (2 linear SVMs followed by two non-linear SVMs). It is evaluated in terms of the detection accuracy, frame-rate, as well as requirements in terms of computing resources. Additionally, the proposed architecture for the *adapted cascade* is compared against two other systems in terms of the above metrics. The first system, which will be referred to as the *initial cascade* and is an implementation of the architecture that can process the same cascade SVMs, but without applying the hardware reduction method, and thus the parallel processing module is implemented using multipliers. The second system, referred to as *single parallel classifier*, consists of multiple sequential processing modules and implements only the most accurate SVM in the cascade, typically the final cascade stage, but in parallel (Figure 4). All systems were implemented on a Xilinx ML505 Virtex 5-LX110T FPGA and all use a Microblaze-based I/O system for I/O and verification. The following sections detail the evaluation process and results.

Through this optimization the total memory demands to store the SVM training data for the first three stages of the SVM cascade are reduced by 40%. Considering both the reduction in processing and storage resources together the adapted cascade implementation requires 43% less FPGA Look-Up Table (LUT) resources compared to the initial cascade implementation. Overall, both the adapted cascade and initial cascade implementations achieve a performance of 70 fps, resulting in a 5× speedup over the single parallel SVM classifier implementation which achieved 14 fps. Finally, the reduction method resulted in only a 0.7 % drop in overall classification accuracy resulting in a 84% of correct detections rate. Overall, the detection accuracy of the adapted cascaded SVM is similar to software works which range from 75% to 88% [1-3].

4. ACKNOWLEDGMENTS

This work was co-funded by the European Regional Development Fund and the Republic of Cyprus through the Research Promotion Foundation (Project NEA ΥΠΟΔΟΜΗ/ΣΤΡΑΤΗ/0308/26)

5. REFERENCES

[1] Osuna E., Freund R., and Girosi F., 1997, Training Support Vector Machines: An Application to Face Detection, Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 130-136.

[2] B. Heisele, T. Serre, S. Prentice, and T. Poggio. 2003, Hierarchical classification and feature reduction for fast face detection with support vector machines, Pattern Recognition, 2007–2017.

[3] I. Kukenys, B. McCane, 2008, Classifier cascades for support vector machines, Intl. Conf. on Image and Vision Computing, 1-6.

[4] "CBCL Face Database #1" [Online]. Available: http://cbcl.mit.edu/software-datasets/FaceData2.html

[5] Bao Face Database, [Online]. Available: http://www.facedetection.com/facedetection/datasets.htm