Design of High-Performance Unified Circuit for Linear and Non-Linear SVM Classifications

Soojin Kim, Seonyoung Lee, and Kyeongsoon Cho

Abstract—**This paper describes the design of a high-performance unified SVM classifier circuit. The proposed circuit supports both linear and non-linear SVM classifications. In order to ensure efficient classification, a 48x96 or 64x64 sliding window with 20 window strides is used. We reduced the circuit size by sharing most of the resources required for both types of classification. We described the proposed unified SVM classifier circuit using the Verilog HDL and synthesized the gate-level circuit using 65nm standard cell library. The synthesized circuit consists of 661,261 gates, operates at the maximum operating frequency of 152 MHz and processes up to 33.8 640x480 image frames per second.**

Index Terms—**Support vector machine, unified, high-performance, pattern recognition, classification**

I. INTRODUCTION

The support vector machine (SVM) [1] was proposed by Vladimir Vapnik and the AT&T Bell laboratory team for accurate binary classification. In order to determine the optimal hyper-plane, the SVM uses the support vectors on the boundaries of two groups. By considering the support vectors, the SVM can improve the accuracy of classification and provide great performance in complex pattern recognition and classification. Since the SVM is considered to be a state-of-the-art tool for linear and non-linear classifications, various algorithms and architectures of the SVM classifier circuit have been proposed [2-12].

This paper proposes a high-performance unified SVM classifier circuit that can support both linear and non-linear classifications. The proposed circuit shares most of the resources required for both linear and non-linear SVM classifications in order to reduce the circuit size. By adopting a parallel architecture to accelerate the operating speed, the proposed circuit processes up to 33.8 640x480 image frames per second. A 48x96 or 64x64 sliding window with 20 window strides is used for the proposed circuit to ensure efficient classification.

The rest of this paper is organized as follows. Section II briefly presents an overview of the algorithms of linear and non-linear SVM classifications. Section III describes the architecture and design of the proposed high-performance unified SVM circuit. Section IV represents the experimental results and the comparison results with other approaches. Finally, Section V concludes this paper.

II. SVM ALGORITHMS

SVM algorithm is comprised of two steps: SVM learning and SVM classification. Support vectors which make the maximum margin between two groups are found and the optimal hyper-plane is determined in SVM learning procedure. It is important that the optimal hyper-plane is determined to provide the best performance with the minimum errors for efficient classification. By using the designated optimal hyper-plane, a new incoming object is classified into one of two groups in SVM classification procedure. SVM classification should be performed in real time on a new obtained data for

Manuscript received Aug. 29, 2011; revised Dec. 5, 2011.
Dep. Electronics and Information Engineering, Hankuk University of Foreign Studies, Korea
E-mail : kscho@hufs.ac.kr

efficient classification [3, 4]. Therefore, it is important to increase the operating speed of SVM classification, resulting in large amount of computation.

Instead of considering the centers of gravity in two groups, SVM algorithm focuses on the support vectors which are on the boundaries of each group. The unique feature of SVM is that it determines the optimal hyper-plane by setting some limitations. The distance between two boundaries must be the maximum and there should not be any data between the boundaries for accurate classification. By using the designated optimal hyper-plane, SVM can provide great performance on the pattern recognition and classification.

In SVM classification, the objects are classified into one of two groups by the designated optimal hyper-plane. The linear SVM is used when the objects can be classified linearly into one of two groups as shown in Fig. 1, and the optimal hyper-plane is obtained by setting $d(x)$ in Eq. (1) to zero. In this equation, X represents the support vectors, Y represents the features of the objects, i represents the dimension of the support vectors and features, and b represents the bias.

$$d(x) = X_i^T Y_i + b \tag{1}$$

The non-linear SVM is used when it is difficult to classify the objects in a linear manner as shown in Fig. 2. However, those objects can be classified linearly if we map their original space into a higher dimension. While classification can be performed linearly by increasing the dimension, an enormous amount of computational resources is required. In order to overcome this problem, the kernel trick is adopted in the non-linear SVM. The kernel trick is an implicit method to increase the dimension of the features. This kernel trick enables

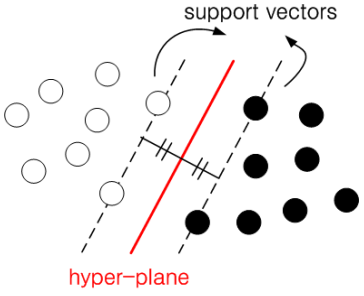

Fig. 1. Example of linear SVM classification.

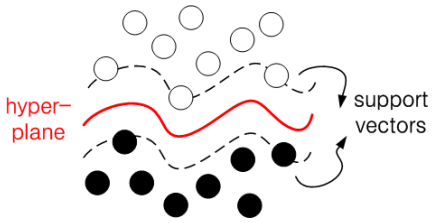

Fig. 2. Example of non-linear SVM classification.

efficient computation of the inner product in non-linear SVM classification. The optimal hyper-plane for non-linear SVM classification is obtained by setting $d(x)$ in Eq. (2) to zero. Eq. (3) is the radial basis function (RBF) kernel that is typically used for non-linear SVM classification. In Eq. (2), *svnum* represents the number of support vectors, α represents the Lagrange multiplier, and y is the parameter used in the Lagrange function. σ in Eq. (3) is the width of the Gaussian window.

$$d(x) = \sum_{N=1}^{svnum} \alpha y K(X_i, Y_i) + b \tag{2}$$

$$K(X_i, Y_i) = e^{-\|X_i - Y_i\|^2 / 2\sigma^2} \tag{3}$$

By applying Eq. (1) or (2) to the features of a new incoming object, the object is classified into one of two groups according to the value of $d(x)$. Note that $d(x)$ means the distance between the optimal hyper-plane and the new object.

III. Circuit Description

The features extracted from the objects by the histograms oriented gradients (HOG) algorithm [13] are used in the proposed SVM classifier circuit. The proposed high-performance unified SVM circuit uses support vectors and HOG features with 3,780 dimensions. A 48x96 or 64x64 sliding window with 20 window strides is used for each image frame to ensure efficient classification.

Fig. 3 shows the proposed high-performance unified circuit with the capability of linear and non-linear SVM classifications. It operates in linear and non-linear classification mode when 'kernel_type' is 0 and 1, respectively.

The proposed circuit is based on a parallel architecture with pipelines and it processes 112 dimensions for one

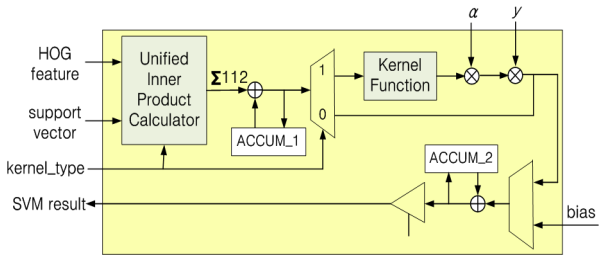

Fig. 3. Proposed unified SVM circuit.

pair of input data (support vectors and HOG features) per clock cycle. Since the proposed SVM circuit uses input data with 3,780 dimensions, 34 clock cycles are required for one pair of support vectors and HOG features. The 'ACCUM_1' register in Fig. 3 is used to accumulate the results of the 'Unified Inner Product Calculator' circuit for 34 clock cycles.

Since the operation for the kernel function is required in non-linear SVM classification, as shown in Eq. (2), the 'Kernel Function' circuit operates only in non-linear classification mode. In non-linear SVM classification mode, values obtained from processing 3,780 dimensions should be accumulated for *svnum* times, as shown in Eq. (2). The 'ACCUM_2' register in Fig. 3 is used in this accumulation operation.

1. Unified Inner Product Calculation

As shown in Eqs. (1, 3), inner product operations for vector data are necessary for both linear and non-linear SVM classifications. The inner product Eq. (4) and Euclidean distance Eq. (5) are used in the linear and non-linear SVM, respectively. In order to share the required resources for the proposed circuit, we rearranged Eq. (5) to obtain Eq. (6) for non-linear SVM classification.

$$X_i^T Y_i = X_i \bullet Y_i = X_0 \bullet Y_0 + X_1 \bullet Y_1 + \ldots\ldots + X_N \bullet Y_N \quad (4)$$

$$\| X_i - Y_i \|^2 = X_i \bullet X_i - 2X_i \bullet Y_i + Y_i \bullet Y_i \quad (5)$$

$$\| X_i - Y_i \|^2 = X_i \bullet X_i + Y_i(Y_i - 2X_i) \quad (6)$$

Fig. 4 shows the proposed 'Unified Inner Product Calculator' circuit using a parallel architecture with two-stage pipeline. As shown in the figure, the proposed 'Unified Inner Product Calculator' circuit shares

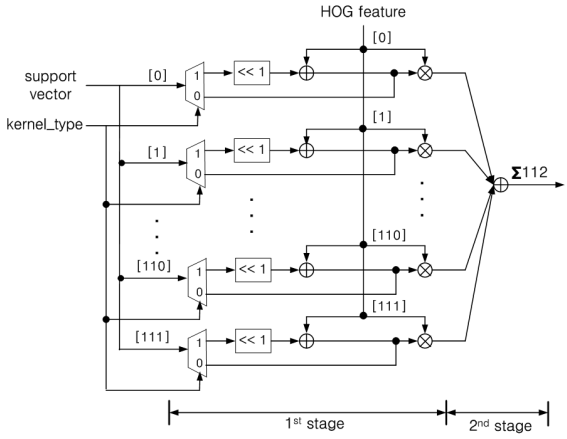

Fig. 4. Proposed unified inner product calculator circuit.

multipliers and adders to support both linear and non-linear SVM classifications. The adders in the first pipeline stage are only used for non-linear SVM classification.

By adopting a parallel architecture with two-stage pipeline, the proposed circuit can quickly process a large number of operations. The proposed circuit processes 112 dimensions for one pair of input data per clock cycle in order to accelerate the operating speed. Since three multipliers and two adders are required in Eq. (5), 336 (112x3) multipliers and 224 (112x2) adders are necessary to process 112 dimensions per clock cycle. Since the optimal hyper-plane and support vectors are already known at this point, the pre-computed values for the inner product of ($X_i \bullet X_i$) can be used. We rearranged Eq. (5) to obtain Eq. (6) in order to exclude the inner product operation for ($X_i \bullet X_i$) and applied this to the proposed 'Unified Inner Product Calculator' circuit. In that case, Eq. (6) requires only one multiplication and addition operations. Therefore, only 112 multipliers and adders are required in the proposed circuit. Furthermore, most of the resources required for both linear and non-linear SVM classifications are shared, as shown in Fig. 4, in order to decrease the circuit size. In this manner, the circuit size is significantly reduced. The values obtained from the 'Unified Inner Product Calculator' circuit are accumulated for 34 clock cycles.

2. Kernel Function Calculation

In non-linear SVM classification, the value obtained from the Euclidean distance operation is divided by $2\sigma^2$,

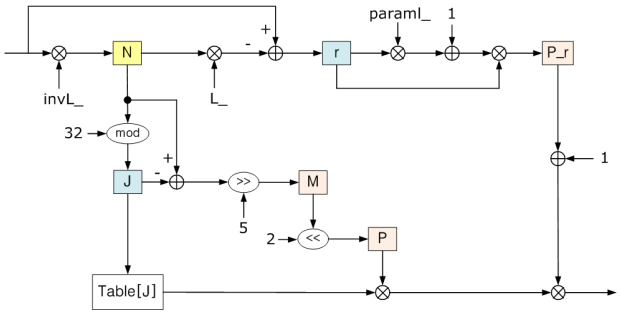

Fig. 5. Proposed kernel function circuit.

and this is used to calculate the kernel function in Eq. (3). The table-driven algorithm was proposed in [14] and we adopted it to the proposed 'Kernel Function' circuit for efficient calculation. The table-driven algorithm accelerates the operating speed for an exponential function where fixed-point arithmetic operations are required. Since a table-driven algorithm can increase not only the operating speed but also the accuracy, it has been widely used to accelerate fixed-point arithmetic operations.

Fig. 5 shows the proposed 'Kernel Function' circuit. The operation of the exponential function requires four clock cycles, and the value obtained from the proposed 'Kernel Function' circuit is multiplied by α and y as shown in Fig. 3 and Eq. (2).

IV. Experimental Results

We described the proposed high-performance unified SVM classifier circuit using the Verilog hardware description language (HDL) and synthesized the gate-level circuits using 65 nm standard cell library. Fig. 6 shows the timing diagram of the proposed SVM circuit in case that the circuit is processing one sliding window in non-linear SVM classification. Since it is required to process 213 support vectors per sliding window, a total of 7,248 clock cycles are required in non-linear SVM classification. In case of linear SVM classification, only one support vector is required to be processed. Therefore, a total of 36 clock cycles are required in linear SVM classification (34 clock cycles for the inner product calculation, one clock cycle for the latency, and one clock cycle for the last addition with the bias).

Table 1 shows the synthesis results and performance of the proposed circuit. The synthesized circuit consists of 661,261 gates and its maximum operating frequency is

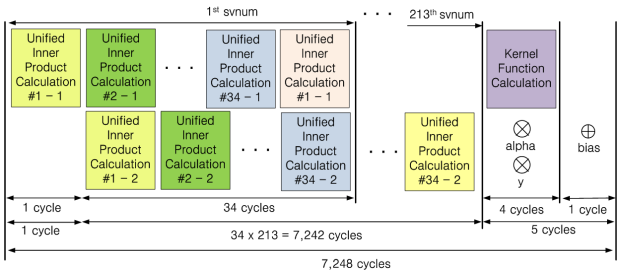

Fig. 6. Timing diagram of proposed circuit (non-linear mode).

Table 1. Synthesis results and performance

Image size	640x480
Sliding window size	48x96, 64x64
Window stride	20
# of sliding windows per frame	609
# of cycles per sliding window	7,248 cycles
Maximum delay	6.6 ns
Maximum operating frequency	152 MHZ
Speed	33.8 frames/s
Gate count	661,261

152MHZ. The circuit can process up to 33.8 640x480 image frames per second, since the maximum number of clock cycles required to process one sliding window is 7,248.

Table 2 shows the comparison results of the proposed SVM circuit with others. The circuit proposed in [10] is based on the parallel array architecture, and the main features of the circuit are resource sharing and efficient memory management. The parallel architectures are adopted in [11] and [12] in order to accelerate the operating speed. The circuit proposed in [11] processes six 16-bit fixed-point data per clock cycle, and the circuit proposed in [12] processes 512x512 image frames in real time by operating all dimensions for input data per clock cycle.

As shown in Table 2, our circuit can support both of linear and non-linear SVM classifications, while other circuits support only one of them. The image size, the number of support vectors and the dimension of the data have a direct impact on the processing time and the circuit resources. The size of the image that our circuit can process is larger than others. In order to increase the precision of the classification result, the proposed circuit uses 25-bit fixed-point data and processes 213 support vectors with 3,780 dimensions. As shown in the table, the precision and dimension of the input data for the proposed circuit are higher and the number of support

Table 2. Comparison results

	[10]	[11]	[12]	Proposed
SVM type	Non-linear	Linear	Non-linear	Linear, Non-linear
Archi-tecture	Parallel array	Parallel	Parallel	Parallel, Pipeline
Input data	8-bit grayscale	16-bit fixed-point	8-bit grayscale	25-bit fixed-point
Kernel	Poly-nomial, RBF	Linear	Poly-nomial	RBF
Image size	320x240	256x256	512x512	640x480
Window size	19x19 18x36 100x40	N/A	8x8 16x16	48x96 64x64
Window stride	5 8 10	N/A	N/A	20
# of sliding windows	4,405 467 74	N/A	N/A	609
# of support vectors	400 467 74	78	128	213
Dimension	N/A	6	256	3,780
# of multipliers	322	8	258	119
# of adders	1,601	24	260	584

vectors is larger than others. The circuit proposed in [10] supports RBF kernel by using look-up table (LUT). Our circuit, on the other hand, calculates RBF kernel operation itself in order to further improve the precision of the classification result. The proposed SVM circuit can process large amount of data in real time, but the number of required resources is smaller than others, as shown in Table 2. Note that the number of adders in Table 2 includes the number of adders for the 2's complement operation.

V. CONCLUSIONS

This paper proposed a high-performance unified SVM classifier circuit. The proposed circuit supports both linear and non-linear SVM classifications, and unification is achieved by sharing most of the circuit resources such as adders and multipliers in order to reduce the circuit size. A parallel architecture with pipelines is adopted in order to accelerate the processing speed. The proposed high-performance unified SVM classifier circuit processes up to 33.8 640x480 image frames per second when 65 nm standard cells are used.

ACKNOWLEDGMENTS

This work was supported by Hankuk University of Foreign Studies Research Fund of 2011.

REFERENCES

[1] Vapnik, V. N., *Statistical Learning Theory*, John Wiley & Sons, 1998.

[2] Gomes Filho, J., Raffo, M., Strum, M., and Wan Jiang Chau, "A General-purpose Dynamically Reconfigulable SVM," *2010 VI Southern Programmable Logic Conference*, pp.107-112, Mar., 2010.

[3] Papadonikolakis, M. and Bougamis, C., "A Novel FPGA-based SVM Classifier," *Field-programmable Technology, IEEE International conference on*, pp.283-286, Dec., 2010.

[4] Kyrkou, C. and Theocharides, T., "Scope: Towards a Systolic Array for SVM Object Detection," *IEEE Embedded Systems Letters*, Vol. 1, No. 2, pp.46-49, Aug., 2009.

[5] Demir, B. and Erturk, S., "Improving SVM Classification Accuracy using a Hierarchical Approach for Hyperspectal Images," *Image Processing, IEEE International Conference on*, pp.2849-2852, Nov., 2009.

[6] Hsu, C. F., Mong-Kai Ku, and Li Yen Liu, "Support Vector Machine FPGA Implemantation for Video Shot Boundary Detection Application," *IEEE International SoC Conference*, pp.239-242, Sep., 2009.

[7] Ruiz-Llata, M., Guarnizo, G., and Yebenes-Calvino. M, "FPGA Implementation of a Support Vector Machine for Classification and Regression," *Neural Networks, International Joint Conference on*, pp.1-5, Jul., 2010.

[8] Irick, K., DeBole, M., Narayanan, V., and Gayasen, A., "A Hardware Efficient Support Vector Machine Architecture for FPGA," *Field-programmable Custom Computing Machines, International Symposium on*, pp.304-305, Apr., 2008.

[9] Manikandan, J. and Venkataramani, B., "Design of a Modified One-against-all SVM Classifier,"

Systems, Man, and Cybernetics, IEEE International Conference on, pp.1869-1874, Oct., 2009.

[10] Kyrkou, C. and Theocharides, T., "A Parallel Hardware Architecture for Real-time Object Detection with Support Vector Machines," *Computers , IEEE Transactions on*, pp.1-12, 2011.

[11] Omar, P., Raquel, V., and Oscar. Y, "An FPGA Implementation of Linear Kernel Support Vector Machines," *Reconfigurable Computing and FPGA's*, *IEEE International Conference on*, pp.1-6, Sep., 2006.

[12] Reyna, R. A., Esteve, D., Houzet, D., and Albenge, M. F., "Implementation of the SVM Neural Network Generalization Function for Image Processing," *Computer Architectures for Machine Perception*, *Proceedings* of IEEE International Workshop on, pp.147-151, Sep., 2000.

[13] Dalal, N. and Triggs, B., "Histograms of Oriented Gradients for Human Detection," *Computer Vision and Patttern Recognition, IEEE Computer Society Conference on*, Vol. 1, pp.886-893, Jun., 2005.

[14] Tang, P. T. P., "Table-driven Implemantation of the Exponential Function in IEEE Floating-point Arithmetic," *Mathematical Software, ACM Trans. on*, Vol. 16, No. 2, pp.144-157, Jun., 1989.

Soojin Kim received her B.S. and M.S. degrees in Electronics and Information Engineering from Hankuk University of Foreign Studies, Korea, in 2007 and 2009, respectively. She is currently pursuing a Ph.D. degree at the Department of Electronics and Information Engineering from Hankuk University of Foreign Studies, Korea. Her research interests include the SoC architecture and design for multimedia and vision systems.

Seonyoung Lee received his B.S., M.S. and Ph.D. degrees in Electronic and Information Engineering from Hankuk University of Foreign Studies, Korea, in 1998, 2000 and 2009, respectively. From 2001 to 2006, he was a researcher at Enhanced Chip Technology, Inc. Since 2009, he has been a senior researcher at the Convergent SoC Center at Korea Electronics Technology Institute, Korea. His research interests include the SoC architecture and design for multimedia and vision systems.

Kyeongsoon Cho received his B.S. and M.S. degrees in Electronics Engineering from Seoul National University, Korea, in 1982 and 1984, respectively. He received his Ph.D. degree from the Department of Electrical and Computer Engineering at Carnegie Mellon University, U.S.A, in 1988. From 1988 to 1994, he was a senior researcher at the Semiconductor ASIC Division of the Samsung Electronics Company. He was responsible for the research and development of the ASIC cell library and design automation. Since 1994, he has been a professor at the Department of Electronics and Information Engineering at Hankuk University of Foreign Studies. In parallel with his academic research and education, he has also been very active in the industrial sector. From 1999 to 2003, he was a senior director at Enhanced Chip Technology. From 2003 to 2004, he was a head of the CoAsia Korea Research and Development Center. Since 2005, he has been a technical advisor of Dongbu HiTek and a vice director of the Collaborative Project for Excellence in System IC Technology sponsored by the Ministry of Knowledge Economy, Korea. His current research activities include the SoC architecture and design for multimedia and communications, SoC design and verification methodology, and very deep submicron cell library development.