Hardware synthesis methodology of Vertex Component Analysis Matlab algorithm

Javier Martín Abasolo, Gustavo Marrero Callicó, Sebastián López Suárez
Integrated System Design Division, IUMA
Las Palmas de Gran Canaria, Spain
javier.martin.abasolo@gmail.com, gustavo@iuma.ulpgc.es, seblopez@iuma.ulpgc.es

***Abstract*—This paper presents a hardware synthesis methodology for the implementation of the Vertex Component Analysis (VCA) using MATLAB. In order to implement hardware prototypes following the time to market philosophy, we have developed a hardware synthesis methodology for high-level languages. Using software like Embedded C and Fixed Point MATLAB Toolbox and Catapult C we have achieved good results in the synthesized hardware for FPGAs with reduced execution results. In this article we used the fixed point arithmetic as an alternative to floating point which is not suitable for electronic devices such as FPGAs.**

Keywords: hyperespectral, endmember, methodology, VCA.

During the last years, several algorithms for endmember extraction have been published in scientific literature. Despite the different nature of these algorithms, they all demand a huge computational effort in order to extract the endmembers of a hyperspectral image. Combined with the high dimensionality of hyperspectral data, they cause serious complications in the use of these algorithms in application domains under real time constraints.

This paper proposes a fast implementation methodology of one of the most popular endmember extraction algorithm, the Vertex Component Analysis (VCA) [1]. This methodology is based in the synthesis of the VCA MATLAB algorithm.

To achieve this, software tools have been used to manage embedded C code generation from MATLAB's original code. Having obtained the Embedded C code, we proceed to perform the hardware synthesis from high-level language. Before synthesizing the algorithm, we have to use fixed point arithmetic instead of the inefficient floating point.

The Embedded C program generated is able to reproduce the results of its original, while exhibiting a lower computational complexity in comparison of the Matlab algorithm. Using the proposed solution of fixed point for the embedded C code obtain good results from frequency and latency in the hardware synthesized by the example of Altera Stratix III FPGA.

The rest of this paper is organized as follows. Section II describes the VCA algorithm, while Section III exposes the methodology developed for synthesis MATLAB code algorithm. Section IV presents the most significant results obtained and, finally, Section V outlines the conclusions extracted from this work.

I. THE VCA ALGORITHM

The VCA algorithm has demonstrated to be a more effective solution than other classical endmember extraction algorithms, such as the pixel purity index (PPI) algorithm [2] or the N-FINDR algorithm [3], in the sense that it provides similar results to the ones provided by these two algorithms but demanding a lower computational effort.

The VCA algorithm is based on the algebraic fact that the endmembers are the vertices of a simplex, being the affine transformation of a simplex also a simplex. VCA uses a positive cone defined by the hyperspectral data to be processed, which projected on a properly chosen hyperplane gives a simplex with vertices corresponding to the endmembers. After projecting the data onto the selected hyperplane, the VCA projects all image pixels to a random direction, obtaining the first endmember as the pixel with the largest projection. The other endmembers are identified by iteratively projecting the data onto a direction orthonormal (given by a vector named *f*) to the subspace spanned by the endmembers already determined. The new endmember is then selected as the pixel corresponding to the extreme projection, and the procedure is repeated until the whole set of *p* endmembers is found.

II. MATLAB CODE METHODOLOGY SYNTHESIZATION

Programming in high level languages allows quick development of algorithms. New programming tools like Embedded and Catapult C allow fast hardware synthesis from MATLAB code. For the best results of the synthesis it is necessary to study the algorithm to simplify and adapt it to the requirements of Embedded C. Furthermore, we have tested it for the use of fixed point arithmetic and integer arithmetic to improve the performance of the VCA that runs on devices without Floating Point Units. The proposed methodology for the algorithm aims to simplify the process of hardware synthesizing other algorithms developed in MATLAB. The presented method selects the use of integer arithmetic, fixed point or floating depending on the algorithm, the restrictions imposed by Embedded MATLAB [4] and taking into account the possible generation of overflow in math calculations.

III. RESULTS

In this section, the performances of the different versions of VCA algorithms are compared.

Artificial hyperspectral images represent an excellent test bench for the purpose of comparing the different algorithms. In particular, the hyperspectral images used in this work were generated by the *demo_vca* software tool available at [5].

In order to evaluate the accuracy of the algorithms, the Spectral Information Divergence (SID) measures the difference between the extracted endmember $\widehat{m}_i$ and its correspondent real endmember signature m_i is calculated as follows:

$$SID_{m_i,\widehat{m}_i} \equiv D(m_i|\widehat{m}_i) + D(\widehat{m}_i|m_i)$$

$$D(m_i|\widehat{m}_i) \equiv \sum_{j=1}^{L} p_j \, log\left(\frac{p_j}{q_j}\right)$$

$$p_j = \frac{m_{ij}}{\sum_{k=1}^{L} m_{ik}} \text{ y } q_j = \frac{\widehat{m}_{ij}}{\sum_{k=1}^{L} \widehat{m}_{ik}}.$$

TABLE I
VCA SID ERROR

	5 *endmembers*	10 *endmembers*	15 *endmembers*
VCA original	0.00185	0.000213	0.00047
VCA mod	0.00185	0.000213	0.00047
VCA Fixed Point	0.00185	0.00118	0.000519
VCA C++ fixed	0.0518	0.0018	0.000519
VCA integer arithmetic	0.00185	0.0018	0.000519

TABLE II
MAXIMUM FREQUENCY OF HARDWARE SYNTHESIS

	Freq. MHz
VCA original	5.56
VCA mod	178.635
VCA Fixed Point	172.295
VCA C++ fixed	102.6690
VCA integer arithmetic	90.220

IV. CONCLUSION

As seen from both tables, VCA algorithm implementations have an error similar to that obtained by the original algorithm. High frequencies have been obtained for barium synthesis versions of the algorithm, floating point and fixed point.

The results obtained in synthesizing do not have large differences which can be obtained in the synthesis of hardware description languages. In contrast, the implementation time of our methodology is much lower, closer to the time to market.

REFERENCES

[1] J. Nascimento, J. Bioucas (2005). “Vertex component analysis: a fast algorithm to unmix hyperspectral data”, IEEE Transactions on Geoscience and Remote Sensing, v. 43, nº. 4, pp. 898-910.

[2] F. Chaudhry, C. Wu, W. Liu, C. Chang, A. Plaza (2006). “Pixel purity index-based algorithms for endmember extraction from hyperspectral imagery”. Transworld Research Network, pp.30-62.

[3] M. Zortea, A. Plaza (2009). “A quantitative and comparative analysis of different implementations of N-FINDR: A fast endmember extraction algorithm”, IEEE Geoscience and Remote Sensing Letters, v. 6, nº. 4, pp. 787-791.

[4] Embedded Matlab Function Library Reference. [Citado el: 1 de 6 de 2011.] http://www.kxcad.net/cae_MATLAB/toolbox/eml/ug/bq1h2z7-9.html.

[5] http://www.lx.it.pt/~bioucas/code.htm. Last visited: 20/07/11.