FPGA Implementation of Earth Mover’s Distance

B Chandra Sekhar Naik1, Anudeep J2, Srinivas Lakavath3, Pinal J Engineer4
1,2,3Electronics Department, SVNIT, Surat, Gujarat, India
4Assistant Professor, Electronics Department, SVNIT, Surat, Gujarat, India

Abstract—Image processing has become a vibrant area of research over the last few years and distance measurement between two images is needed in many applications. Earth Mover’s Distance (EMD) is the algorithm for measuring similarity between images. However, application of the current software implementations of the EMD algorithm to real time optimization problems are time consuming when used as Image Retrieval and many other applications as shown in many research papers. This Paper describes Hardware implementation of EMD algorithm, which is reduced the time complexity, designed with VHDL language and synthesized on Virtex4 FPGA and also shows comparison of time analysis with MATLAB and C. The result shows that the architecture is able to operate at 300.144 MHz speed which is faster than any software solution.

Keywords— Image Processing, EMD, Distance Metric, VHDL.

I. INTRODUCTION

Measure the distance between two images is needed in many applications. There exist many distance metrics to measure the distance between two images of same size i.e. Manhattan Distance [1], Euclidean Distance [1], Vector Cosine Angle Distance [2] and Earth Mover’s Distance [3] etc. Among these distances EMD gives better performance for similarity measures and also for partial matching.

The Earth Mover’s distance (EMD) was first introduced by Rubner et al. [3] for color and texture images, is cross-bin distance, defined as the minimal cost that must be paid to transform one histogram into the other, where there is “ground distance” between the basic features that are aggregated into the histogram. Computing the EMD is based on a solution to the well-known Linear Programming problem. The Earth Mover’s Distance has been used successfully in many applications such as Image Retrieval [3], Edge and Corner Detection [4], Key-point Matching [5] [6], near duplicate image identification [7] and Contour matching [8].

A. Linear Programming

Linear Programming is a scientific computing application provides a general framework for describing optimizations problem as a linear objective function and a set of linear constraints.

Linear Programming is applied to large variety of scientific computing applications and industrial optimization problems. A formulation for a maximization problem is as shown in (1), where x is a vector with the variables, A is the matrix, and c and b are vectors of coefficients. A minimization problem is can be formed by negating the objective function coefficients of the maximization problem.

$$\text{max} \quad c^T x$$
subject to $Ax \leq b$
$x \geq 0$

The Simplex Algorithm [9] provides a robust tool for solving problems modelled using a linear programming frame work. Almost all the commercial and research tools available for linear programming use some variant of the Simplex algorithm.

B. EMD Definitions

The Earth Mover’s Distance is defined for “signatures” of the form $\{(x_i, p_i) \ldots , (x_m, p_m)\}$, where $x_i$ is the center of data cluster $i$ and $p_i$ is the number of points in the cluster. The signatures are not normalised, so the total masses of two signatures may not be equal. Given two signatures $P = \{(x_1, p_1) \ldots , (x_m, p_m)\}$ and $Q = \{(y_1, q_1) \ldots , (y_n, q_n)\}$, the EMD is defined in terms of optimum flow $F = \{(f_{ij})\}$, which minimizes[10]

$$W(P, Q, F) = \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}$$

Where $d_{ij} = d(x_i, y_j)$ is some measure of dissimilarity between $x_i$ and $y_j$, $W(P, Q, F)$ is the work required to move earth from one signature to another. The flow $(f_{ij})$ must satisfy the following constraints:

$$f_{ij} \geq 0, \quad 1 \leq i \leq m, 1 \leq j \leq n$$

$$\sum_{j=1}^{n} f_{ij} \leq p_i, 1 \leq i \leq m$$

$$\sum_{i=1}^{m} f_{ij} \leq q_j, 1 \leq j \leq n$$
Once the optimal flow $f_{ij}^*$ is found, the Earth Mover’s Distance between P and Q is defined as

$$EMD(P, Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}^* d_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}^*}$$ (7)

C. FPGA and Coding

This paper mainly focuses on FPGA Implementation. FPGA (Field Programmable Gate Array) design allows designers to design their own modules according to their needs and upgrade the system conveniently. The system design based on FPGA is flexible with the advantages of parallelism, low cost and low power consumption. Hardware Description Language (HDL) is a specialized computer language used to program structure, design and operation of electronic circuits, and most commonly, digital logic circuits. VHDL and Verilog HDL are most commonly used HDL’s. In our design we are using VHDL as programming language.

This paper is organized as follows: The section II presents HDL Implementation Architecture. Section III presents Simulation Results and Synthesis Report. Section IV presents Conclusion and Future Scope.

II. HARDWARE IMPLEMENTATION

The proposed method has the advantage of being computationally simple without decreasing the efficiency and increasing the computational time. The main purpose our work is to design a feasible hardware circuits based on FPGA for Earth Mover’s Distance to measure distance between two images of same size to improve the processing speed. First the EMD is implemented with C, then transform to VHDL Coding. The Hardware include four major modules: Histogram, Ground distance, Equality matrix and Simplex algorithm. To our knowledge, this is the first FPGA implementation. This implementation is applicable to only number of bins 2. The simplified block diagram of EMD implementation as shown in Fig.1

The implementation is divided into two parts: First one is to create EMD matrix for Simplex algorithm and second is solve Linear programming problem. Fig 2 shows First part and Fig3 shows second part. Fig 4 shows top module of Earth Mover’s Distance.

D. Histogram Calculation

Histogram calculation is main task in the implementation. To read images directly on FPGA is not possible so, Image reading is implemented and created Xilinx coefficient (.Coe) files of images. These .Coe files are stored in memory preferably, Single Port ROM. The Single Port Block Memory module is generated based on user-specified width and depth.
When block memory is enabled, all memory operations occur on the active edge of the clock input (CLK). When the block memory is disabled the memory configuration and output value remain unaltered. Fig. 5 show block diagram Histogram calculation.

A. Weights Calculation

Weight is equal to histogram value divided by sum of all histogram values. The division algorithm is implemented by using shift logic. For fixed size image i.e., 256x256, the sum of histogram values should be power of 2. So, the division is implemented by shifting 16 bits to the left. The result should be in fractional. The fractional values are represented in the form of fixed-point representations By implementing algorithms using fixed-point (integer) mathematics, a significant improvement in execution speed can be observed because of inherent integer math hardware support in a large number of processors, as well as reduced software complexity for emulated integer multiply and divide. To more accurately construct an algorithm, double and single precision floating point data and coefficient values should be used.
However there is significant processor overhead required to perform floating-point calculations resulting from the lack of hardware based floating-point support. To improve mathematical throughput or increase the execution rate, calculations can be performed using fixed-point representations. The fixed-point representations used in our design as shown in Fig. 6.

**B. Equality Constraints Matrix**

Equality Constraints Matrix values are assigned directly. Because all the values should be 0’s and 1’s.

**C. Setup LP(Linear Programming)**

The Setup LP stage reads out the values coming from EMD matrix which includes A matrix (Equally Constraints Matrix), two cost vectors (Ground Distance Values) and the b vector (Weight Values). All of these inputs are giving parallel, saving time. The parallelism incurred here requires little overhead, because the data are not dependent on each other. Add each column elements of A matrix and b vector, stored in other registers, which used for finding pivot column.

**D. Find Pivot Column and Check for Optimality**

The ‘Find Pivot Column’ stage searches for the minimum value within the registers, returns the location of the column to the next stage, which becomes the location of the pivot column. The ‘Check for Optimality’ stage detects whether the current solution is the optimal solution to the LP and indicates to the control logic what to do next. If the solution is non-optimal, the simplex method is carried on. If the solution is optimal, then the simplex algorithm is completed and the solution can be calculated by reading out the b vector and the list of basic variable. The simplified one variable Hardware structure to find minimum and optimality as shown in Fig. 6.
E. Find Pivot Row

The ‘Find Pivot Row’ stage executes the minimum ratio test to determine the existing basic variable. The minimum ratio test has been implemented using a simple search through the b vector. It suffices to find minimum ratios. When the minimum value is found, returns the location of the row to the next stage. Fig. 8 shows one variable simplified Hardware structure. In this A0, A1 are from A matrix and B0, B1 are from b vector.

F. Gaussian Elimination

Finally, the Gaussian Elimination stage turns the pivot element in the simplex table into the single 1 within its column by Gaussian Elimination. The simplified hardware structure for one variable is shown in Fig. 9.

III. RESULTS

Simulation for the FPGA based Earth Mover’s Distance calculation architecture described in this paper is done with the Isim Simulator. The images taken in this design as shown in Fig. 10.
The Simulation results of Earth Mover’s Distance between Cameraman image and Lena image for numbers of bins 2 as shown in Fig. 13

Fig. 12 Weight calculation of Lena image

Fig. 13 EMD value between Cameraman and Lena images for number of bins 2

G. Experiment Results

The FPGA based EMD was implemented on Xilinx Virtex 4, XC4VFX100-12-FF1152 device. The architecture is capable of operating at a clock frequency of 300.144 MHz. The Device Utilization summary is given in Table 1.

Table II summarizes the comparison of time analysis between our work with existing work i.e., MATALB.

IV. Conclusion and Future Scope

The Earth Mover’s Distance for number of bins 2 has been implemented in hardware which allows for calculating distance between two images. Both the histogram calculation and he simplex algorithm have been analysed in depth to perform speed optimizations at both the algorithmic and architectural level. Thus vast improvements in the time analysis of the distance calculation are achieved in the hardware implementation. This implementation has shown a significant improvement in speed over MATLAB and C.

In this design we proposed only for restricted bins i.e., 2. It is suggested to develop more number of bins. For more number of bins it is suggested to implement with memory.
REFERENCES


[9] Samuel Bayliss, Christos, B. and George, A. An FPGA implementation of Simplex Algorithm.
