IFERP connecting engineers... developing research

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 5, May 2016

Design and Implementation of Face Recognition System Using CORDIC-FFT

[1] Pradeep Hulihalli [2] M.C.Chandrashekhar [3] Dr. M Z Kurian
[1][2][3] Dept. of ECE, Sri Siddhartha Institute of Technology
Tumakuru, Karnataka, India
[1]pshulihalli27@gmail.com [2] mccshekhar@gmail.com [3] mzkurianvc@yahoo.com

***Abstract*— In the real world applications like access control and identification of an individual from a group of people which uses parts of human body called as biometrics, like finger prints, face, iris, Deoxyribo Nucleic Acid(DNA), palm print etc which are different for every individual, are being used abundantly in recent times due to their overwhelming features like uniqueness, the reliable results obtained by using them. CO-ordinate Rotation DIgital Computer (CORDIC) is used to find the trigonometric functions like sin θ,cos θ, tan θ, sinh θ, cosh θ, tanh θ of a given angle θ, logarithmic functions. Fast Fourier Transform(FFT) algorithm is used to compute the Discrete Fourier Transform(DFT) of a sequence or its inverse. Fourier analysis converts from time domain to the representation in frequency domain or vice versa. In this proposed work an efficient face recognition system is designed and implemented on FPGA using simulink.**

***Keywords:* Biometric, CORDIC, FFT, FPGA, simulink**

I. INTRODUCTION

The CO-ordinate Rotation DIgital Computer (CORDIC) was first described by Jack.E.Volder[1] in 1959. CORDIC algorithm is an iterative algorithm, which is used to find the trigonometric functions like sin θ,cos θ, tan θ, sinh θ, coshθ, tanh θ of a given angle θ, logarithmic, complex number multiplications, matrix inversion.

CORDIC is implemented using simple hardware through repeated shift-add operations. There are two computing modes for CORDIC namely, rotation and vectoring. In the rotation mode, the components of a vector and an angle of the rotation are given and the coordinate components of the original vector, after rotation through the given angle, are computed. In vectoring mode the coordinate components of the vector are given and the magnitude and angular argument of the original vector are computed.

All of the trigonometric functions can be computed or derived from functions using vector rotations can also be used for polar to rectangular and rectangular to polar conversions, for vector magnitude, and as a building block in certain transforms such as the DFT and DCT. The CORDIC algorithm provides an iterative method of performing vector rotations by arbitrary angles using only shifts and adds operations. Figure 1 shows the rotation of a

A two-dimensional vector $p_0=[x_0\ y_0]$ through an angle θ to obtain a rotated vector $p_n= [x_n y_n]$ could be performed by the matrix product $p_n=R.p_0$ whereRis the rotation matrix:

$$R=\begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix} \quad (1.1)$$

By factoring out the cosine term in (1.1), the rotation matrix R can be rewritten as

$$R=[(1+\tan^2\theta)^{-1/2}]\begin{bmatrix}1 & -\tan\theta\\ \tan\theta & 1\end{bmatrix} \quad (1.2)$$

and can be interpreted as a product of a scale-factor

$$R_C=\begin{bmatrix}1 & -\tan\theta\\ \tan\theta & 1\end{bmatrix} \quad (1.3)$$

$K=[(1+\tan^2\theta)^{-1/2}]$ with a pseudo rotation matrix R_c, given by

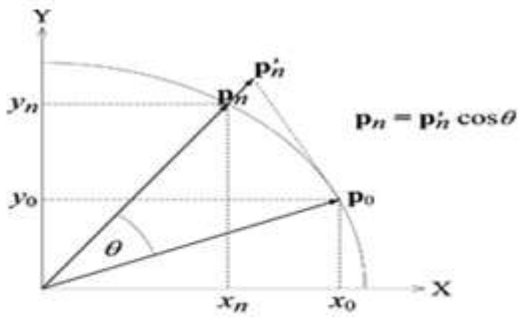

Fig. 1: Rotation of vector on a two-dimensional plane[1]

The pseudo rotation operation rotates the vector p_0 by an angle θ and changes its magnitude by a factor K=cos$\theta$to produce a pseudo-rotated vector $p_n^`$ =Rc.p0.

II. RELATED WORK

Jack E. Volder **[**1] proposed that the CORDIC computing technique is especially suitable for use in a special-purpose computer where the majority of the computations involve trigonometric relationships. In general, the *Rotation* and *Vectoring* operations should be considered constant-length routines in which the number of word times per operation is equal to the word length. Similar algorithms have been developed for multiplication, division, conversion between binary and mixed radix systems, extractions of square root, hyperbolic coordinate transformations, exponentiation and generation of logarithms. It is believed that similar algorithms based on this for many other computing requirements.

J.S.Walther[2] proposed a single unified algorithm for the calculation of elementary functions including multiplication, division, sin, cos, tan, arctan, sinh, cosh, tanh, arctanh, ln, exp, and square-root. The basis for the algorithm is coordinate rotation in a linear, circular, or hyperbolic coordinate system depending on which function is to be calculated. The operations required are only shifting, adding, subtracting.

Milos D. Ercegovac et al.,[3] presented several modifications to the CORDIC method in order to improve speed and efficiency of its implementation. The main contributions are

1) The introduction of redundant (carry-free) addition to replace time-consuming conventional additions;

2) The use of on-line arithmetic to reduce the communication bandwidth, maximize the overlap between successive operations, and replace area-expansive shifters by delays;

3) The use of angles in decomposed forms to eliminate angle accumulation recurrences. These modifications contribute to a speedup of about 4.5 with respect to a conventional CORDIC in the given triangularization algorithms, and to a speedup of about 4in the Singular Value Decomposition (SVD) case. No attempt has been made at this time to estimate the savings in the area since no VLSI realizations are done.

Subit Abraham and PuranGour[4] proposed an area-time efficient CORDIC algorithm that completely eliminates the scale-factor by microrotation selection which is done by the most significant one detector in the micro rotation sequence generation circuit. Proper order of approximation of Taylor series is selected to meet the accuracy requirement, and the desired range of convergence is achieved. A new CORDIC algorithm, they show an enhanced version of the scaling-free CORDIC algorithm. Booth recoding algorithm is used to eliminate scaling and domain folding is avoided which reduces scaling to minimum but not eliminate it.

HunnyPahuja et al.[5], proposed theoretical and practical aspects of implementing sine/cosine CORDIC-based generators in FPGAs. A trade-off speed/area will determine the right structural approach to CORDIC FPGA implementation for an application. Module count and operating speed depend significantly on the used synthesis tool. Simulation has shown that the redundant adder can improve the efficiency of CORDIC FPGA implementations for bit-lengths higher than 32-bit.

P.Malathi et al.,[6] proposedan efficient CORDIC algorithm for FFT processor is implemented on FPGA. Due to use of Radix-4 speed increases when compared with Radix-2FFT computation. For twiddle factor calculation CORDIC algorithm is used, which help to reduce the computation time and make processor faster. The CORDIC provides the opportunity to calculate all the required functions in a rather simple and elegant fashion.

Narayanam Rangnadh and Muni Guravaiah [7] proposed a large family of signal processing techniques which consist of Fourier-transforming a signal, manipulating the Fourier transformed data in a simple way, and reversing the transformation. Fourier frequency analysis is widely used in equalization of audio recordings, X-ray crystallography, artifact removal in Neurological signal and image processing, Voice Activity Detection in Brain stem speech evoked potentials, speech processing spectrograms are used to identify phonetic sounds and so on. Discrete Fourier

Transform (DFT) is a principal mathematical method for the frequency analysis. The way of splitting the DFT gives out various fast algorithms. In this paper, wepresent the implementation of two fast algorithms for the DFT for evaluating their performance. One of them is the popular radix-2 Cooley-Tukey fast Fourier transform algorithm (FFT) and the other one is the Grigoryan FFT based on the splitting by the paired transform.

Vinita Makwana and Neha Parmar[8] proposed a fast fourier transformation for an audio file in which they explained about FFT which is a method to compute DFT and also we can find DFT's inverse. Now days, science introduce many FFT algorithms for computing DFT. It is also very effective. In past, we are using DFT, Discrete Fourier Transform. It is used for same purpose as FFT But it is slow related to FFT.We are considering only frequency coefficients of an signal. DFT and FFT are techniques which can convert samples into its equivalent frequency coefficients. The DFT is obtained by decomposing the sequence of values into modules of frequencies. Computing the DFT of N point takes O(N2) arithmetical operation, where as a FFT can takes only O(N log N) operation.

Renu Bhatia[9] proposed different biiometric techniques. Biometrics is automated methods of recognizing a person based on a physiological or behavioral characteristic. The past of biometrics includes the identification of people by distinctive body features, scars or a grouping of other physiological criteria, such like height, eye color and complexion. The present features are face recognition, fingerprints, handwriting, hand geometry, iris, vein, voice and retinal scan. Biometric technique is now becoming the foundation of a wide array of highly secure identification and personal verification. As the level of security breach and transaction scam increases, the need for well secure identification and personal verification technologies is becoming apparent. Recent world events had lead to an increase interest in security that will impel biometrics into majority use. Areas of future use contain Internet transactions, workstation and network access, telephone transactions and in travel and tourism. There have different types of biometrics: Some are old or others are latest technology. The most recognized biometric technologies are fingerprinting, retinal scanning, hand geometry, signature verification, voice recognition, iris scanning and facial recognition

III. PROPOSED WORK

3.1 Cordic:

The CORDIC algorithm will be used to find the cos and sin of the given angle θ is implemented using code written in VHDL and Xilinx v14.5.

The flowchart of the proposed CORDIC is shown in the figure 2.

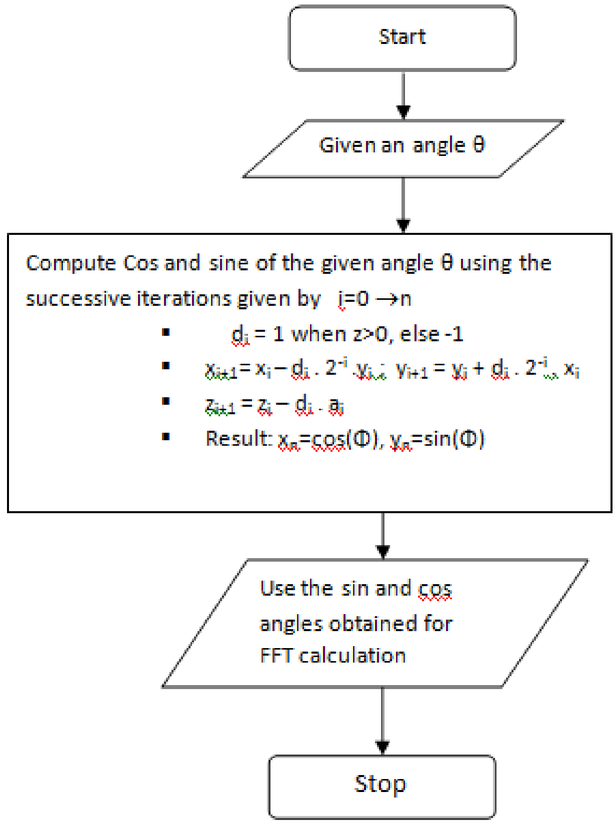

Fig. 2: Flowchart of proposed CORDIC

3.2 FFT:

Decimation in Frequency DIT –FFT

The decimation-in-time (DIF) radix-2 FFT partitions the DFT computation in the correct order which can each be computed by shorter-length DFTs of different combinations of input samples placed as seperate even and odd indexed. Recursive application of this decomposition to the shorter-length DFTs results in the full radix-2 decimation-in-time FFT. The figure 3 shows the first stage of the 8-point DIT algorithm.

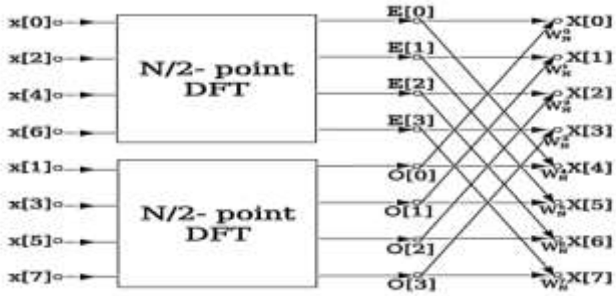

Fig 3: First stage of 8-point DIT algorithm

The decimation, however, causes shuffling in data. The entire process involves $v = \log_2 N$ stages of decimation, where each stage involves N/2 butterflies of the type shown in the figure 4.

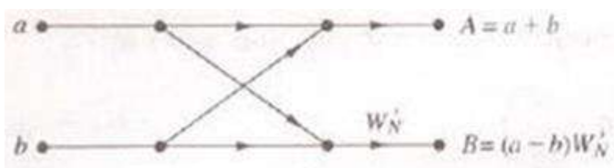

Fig.4: Butterfly Structure

Here $W_N = e^{-j2/N}$, is the Twiddle factor.

The signal flow graph of the 8-point DIT-FFT is as shown in figure 5. The N-point DFT requires $(N/2)\log_2 N$ complex multiplications.

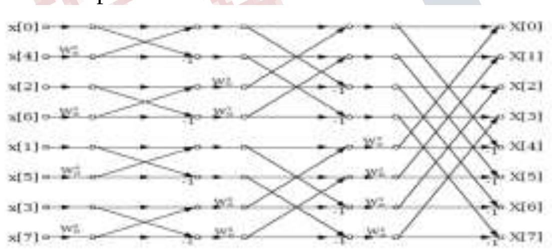

Fig. 5: SFG of 8-point DIT-FFT

3.3 Face Recognition Model:

The proposed Face recognition model is as shown in figure 6.

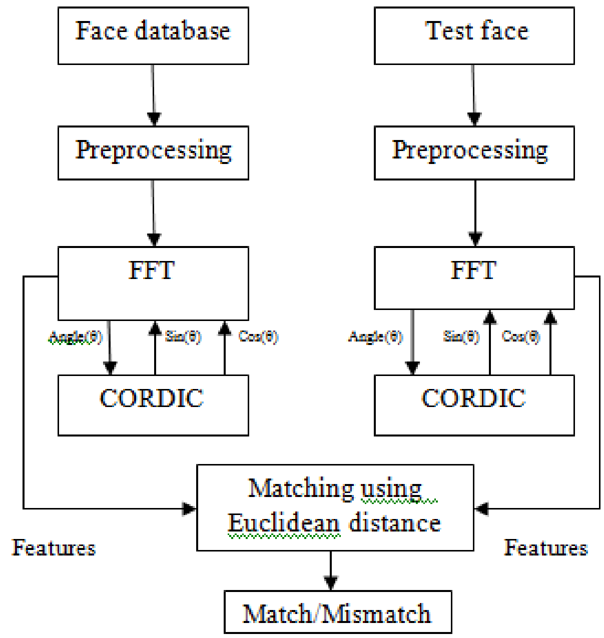

Fig.6:Face Recognition Model

The test image that is image of the face is preprocessed and features are extracted using the CORDIC and features of test image is compared with database features obtained from images of the faces present in the database using the Euclidean distance. Based on the distance obtained the condition for match or mismatch of the test image is decided. In this proposed work features of test image are extracted using feature based approach and CORDIC FFT is used in the rotational mode.

Preprocessing: The preprocessing of the face are carried out in various steps like

(i) Image segmentation
(ii) Applying filter coefficients in order to remove the noise if any present in the acquired image
(iii) Removal of high frequency edges to get more efficient face images

***CORDIC-FFT*:** The CORDIC is used to find the sine and cosine of the given angle θ such that from the obtained trigonometric values the twiddle factor (W_N^{kn}) is obtained which is given by

$$W_N^{kn} = e^{-j\theta} = \cos\theta - j\sin\theta \qquad (1.4)$$

The twiddle factor obtained by the use of CORDIC is used in the calculation of DIF-FFT algorithm used to obtain the magnitude and phase of the acquired image.

IV. RESULTS AND DISCUSSION

The result obtained for the given angle with decimal value 6800 is shown in the figure 7.

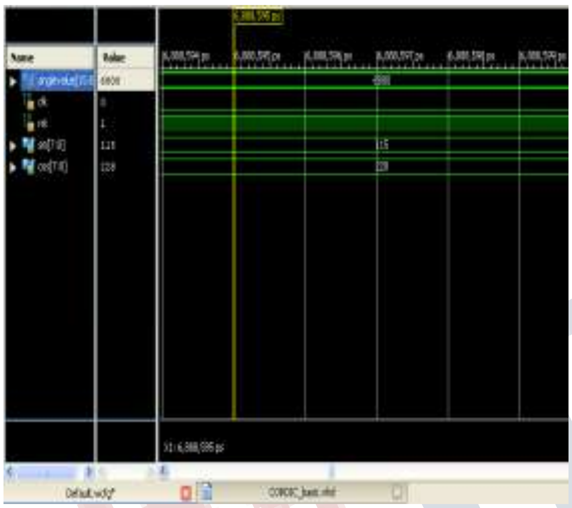

Fig. 7: Result obtained for the proposed CORDIC

The given angle in decimal that is 6800 is divided by 2^8=256 which gives us the value as 26.5625.

The sine of given angle is sin (26.5625) = 0.4472

The cosine of given angle is cos(26.5625)=0.8944

In the results obtained the sin and cos values are 115 and 228 which are in decimal so dividing the values by the number of logic levels used that is 2^8=256

The sine value obtained=115/256=0.449The cos value obtained= 228/256=0.8906.

The number of slice LUT's, fully used LUT-FF pairs, bonded IOB's, BUFG/BUFGCTRL/BUFHCEs is shown in table 1.

Table 1: Logic utilization for the proposed CORDIC

Logic utilization	Used
Number of slice LUT's	273
Number of fully used LUT-FF pairs	0
Number of bonded IOB's	34
Number of BUFG/BUFGCTRL/BUHFCEs	1

The basic RTL schematic of 4-point FFT is shown in figure 8.

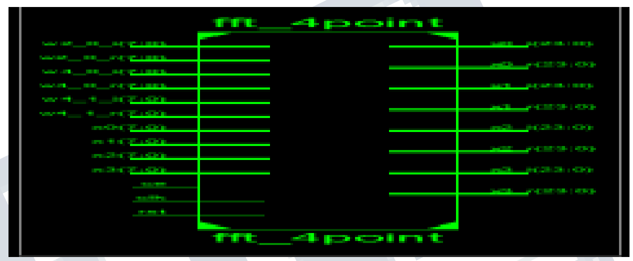

Fig.8: RTL schematic of 4-point FFT

The number of slice LUT's, fully used LUT-FF pairs, bonded IOB's, BUFG/BUFGCTRL/BUFHCEs is shown in table 2.

Table 2: Logic utilization for the proposed 4-point FFT

Logic utilization	Used
Number of slice LUT's	257
Number of fully used LUT-FF pairs	0
Number of bonded IOB's	274
Number of BUFG/BUFGCTRL/BUHFCEs	1
Number of DSP48A1s	8

The magnitude indicates the distance between different parts of the face and the phase is used to see by how much angle the face is tilted and etc. The features extracted using CORDIC-FFT are stored in the database memory.

For the test image the pre-processing and feature extraction using the COORDIC-FFT are done as in the same way above and the features of the test image and images in the database is compared using the Euclidean distance which is the combination of subtraction and squaring operations.

Based on the Euclidean distance obtained the match or mismatch between the test image and the database images

is decided. If Euclidean distance=1 then there is a To find the match or mismatch the XNOR operation is used. As XNOR operation gives the output equal to one which means both the test image and database images are the same otherwise it produces the output equal to zero which means there is mismatch between the test image and the database image.
Since the images shown in figure 10 are both same it gives output as one in the simulink model shown in fig.9

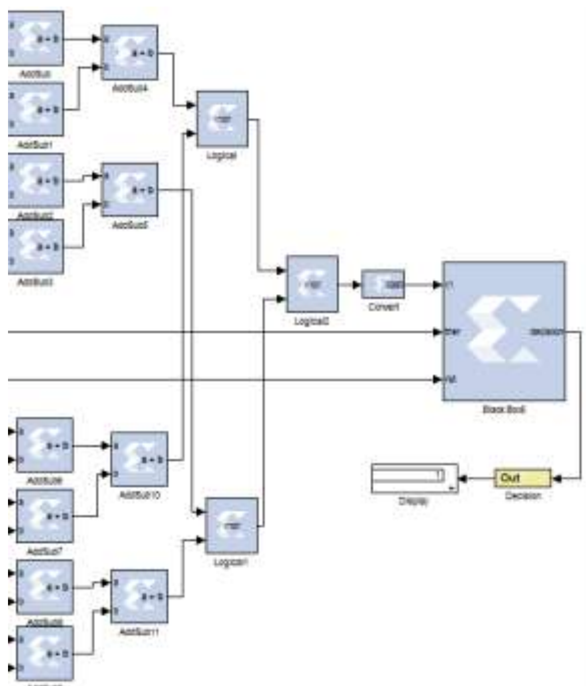

Fig.9: Simulink model for match or mismatch condition

Fig.10: Database and the test image which are identical

Fig.11: Database and test image are not same

Advantages and Disadvantages

Advantages:

- ❖ Hardware requirement and cost of CORDIC processor is less as only shift registers, adders and look-up table (ROM) are required
- ❖ Number of gates required in hardware implementation, such as on an FPGA, is minimum as hardware complexity is greatly reduced compared to other processors such as DSP multipliers
- ❖ It is relatively simple in design
- ❖ No multiplication and only addition, subtraction and bit-shifting operation ensures simple VLSI implementation.
- ❖ Delay involved during processing is comparable to that during the implementation of a division or square-root operation.

Disadvantages:

- For the real time implementation it requires CMOS sensor camera and high version PCB boards as the camera cost is very high it is a constraint.

V. CONCLUSION

The implementation of the CORDIC processor is carried out in which based on the given angle θ the corresponding sinθ and cosθ are calculated. The cos and sine values obtained from CORDIC are very much near to those values obtained from the conventional methods. The error in the values obtained using CORDIC are less than 1%. The error is very small and for the implementation of CORDIC it requires less hardware compared to the conventional methods. Using the obatined sine and cosine values the DIT-FFT is calculated using which the features are extracted. The features obtained for the database image and test image is compared using Euclidean distance and the state of match or mismatch between the images is found out.

REFERENCES

[1] Jack E.Volder, "*The CORDIC trigonometric computing technique*" IRE Transactions on Electronic Computers, volume EC-8, pp. 330–334, Sept. 1959.

[2] J.S.Walther "*A Unified Algorithms for Elementary Functions*" Spring Joint Computer Conference, pp. 379-385, 1971.

[3] Milos D. Ercegovac and Tomas Lang *"Redundant and On-Line CORDIC: Application to Matrix Triangularization and SVD",* IEEE transactions on Computers, Vol. 39, Issue-6, pp. 725-740, Jun 1990.

[4] Subit Abraham and PuranGour *"Review on Area Efficient CORDIC Algorithm and Reduction in Scale Factor"* International Journal of Engineering Research and Applications , Vol. 3, Issue-4, pp. 597-599, Jul-Aug 2013.

[5] HunnyPahuja, Lavish Kansal and Paramdeep Singh "*CORDIC algorithm implementation in FPGA for computation of Sine and Cosine signals"* International Journal of Scientific and Engineering Research , Vol. 2, Issue-12, Dec-2011.

[6] P.Malathi, Manish Sharma and Chetan Korde,"*Pipelined CORDIC architecture for FFT processor implementation of FPGA"* International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering , Vol. 4, Issue-6,June-2015.

[7]Narayanam Ranganadh and Muni Guravaiah P *"Performance Evaluations of Grigoryan FFT and CooleyTukey FFT"* International Journal of Scientific and Research Publications, Volume 3, Issue 1, January 2013 ISSN 2250-3153

[8]Vinita Makwana and Neha Parmar " Analysis of Performance of Fast Fourier Transformation of an Audio File" International Journal of Application or Innovation in Engineering & Management Volume 2, Issue 11, November 2013.

[9] *Renu Bhatia"Biometrics and Face Recognition Techniques"* International Journal of Advanced Research in Computer Science and Software Engineering" Volume 3, Issue 5, May 2013