IMPLEMENTATION OF EMOTIONAL BEHAVIOR OF AUTISTIC CHILDREN USING SIFT TARGET FPGA

A Thesis submitted in partial fulfillment of the requirements for the

degree of

Bachelor of Technology

in Electronics and Communication Engineering

By

Ananya Ipsita (110EC0165)

Under supervision
Of

Prof. K. K. Mahapatra

Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela
Session 2013-2014

CERTIFICATE

This is to certify that **Ananya Ipsita (Roll. No. 110EC0165) of B. Tech** has worked under my supervision and guidance on the project entitled '**IMPLEMENTATION OF EMOTIONAL BEHAVIOR OF AUTISTIC CHILDREN USING SIFT TARGET FPGA'** in partial fulfillment of the requirements for the award of **Bachelor of Technology Degree** in **Electronics and Communication Engineering** at the **National Institute of Technology, Rourkela.**

Prof. K. K. Mahapatra
Professor
Department of Electronics and Communication Engineering
National Institute of Technology, Rourkela-769008

Abstract

Children suffering from autism have difficulties in recognizing the emotions of others which hampers their interpersonal communication. A stable detection system can help such children to easily interact with the external world. In this thesis, an efficient way is proposed which can successfully detect the signs (facial or hand gestures) of such children using Scale Invariant Feature Transform (SIFT) algorithm. Ways of implementing the algorithm on Field Programmable Gate Arrays(FPGA) is also mentioned briefly in this thesis. The algorithm helps in extracting distinctive features from an image using scale invariant features and obtains feature descriptors from the keypoints.

Keywords: Scale Invariant Feature Transform, Field Programmable Gate Arrays, keypoint, scale invariant, feature descriptor

ACKNOWLEDGEMENT

I am extremely grateful to my project guide, **Prof. K. K. Mahapatra**, **Professor, Department of Electronics and Communication Engineering, NIT Rourkela**, for his insightful suggestions on the project work and for guiding me during the project with his encouragement, support and cooperation. I am thankful to **Mr. Vijay Sharma**, PG student in the Department of Electronics and Communication Engineering, NIT Rourkela, for his generous help and continuous encouragement in various ways towards the completion of this project. I would like to convey my sincerest gratitude and indebtedness to all the faculty members and staff of Department of Electronics and Communication Engineering, NIT Rourkela, who showed their great effort and guidance during required times without which it would have been very difficult to carry out my project work. Moreover, an assemblage of this nature could never have been attempted with my reference to the works of others. I acknowledge my indebtedness to all of them. Finally, I would also like to extend my heart- felt thanks to my family for their moral support, love and affection.

Ananya Ipsita

(B.Tech in Electronics and Communication Engineering)

LIST OF FIGURES

CONTENTS

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

Children suffering from autism disorder possess difficulties in reading and understanding the emotions of others. In order to help the people suffering from such problems, we need an efficient emotion recognizer. Many techniques have been put forth by the researchers throughout the world to improve the human computer interaction by using emotion recognition. The image depicting the emotions are pre-processed by using efficient algorithms and the desirable features are extracted and finally used to get the result. In this paper, an efficient method Scale Invariant Feature Transform(SIFT) is used in which the image is processed to obtain local features which are invariant to translation, rotation, scale, and other imaging parameters. In this thesis, methods have been proposed for a hardware efficient system to be implemented on FPGA which can help the autistic children to communicate successfully with their environment.

1.2 Problem Statement

The primary goal of this project is to design a stable emotion recognition system to be implemented on FPGA which can successfully help the autistic children to interact with their external environment.

1.3 Organization of the Thesis

The entire thesis is divided into 5 chapters. The first one give the basic aim, motivation as well as the work intended. In chapter 2, the various steps in which The Scale Invariant Feature Transform(SIFT) can be implemented is explored. Chapter 3 deals with the implementation of the SIFT algorithm on FPGA. The fourth chapter shows the work carried out throughout the entire session, the output figures and results on MATLAB. It also mentions about the tasks completed in VHDL for FPGA implementation The conclusion and future work has been proposed in chapter 5.

CHAPTER 2

SIFT OVERVIEW

2.1 BACKGROUND:

The Scale Invariant Feature Transform is an efficient method proposed by David Lowe in 1999[4]. For any particular portion of an image, interesting points can be obtained for describing the features of that portion. While comparing the query image with the images present in the database, the feature descriptor can be used to locate the similar objects in the images. The main idea behind this algorithm is to transform the image content to local feature coordinates invariant to rotation, translation, scale and other imaging parameters.

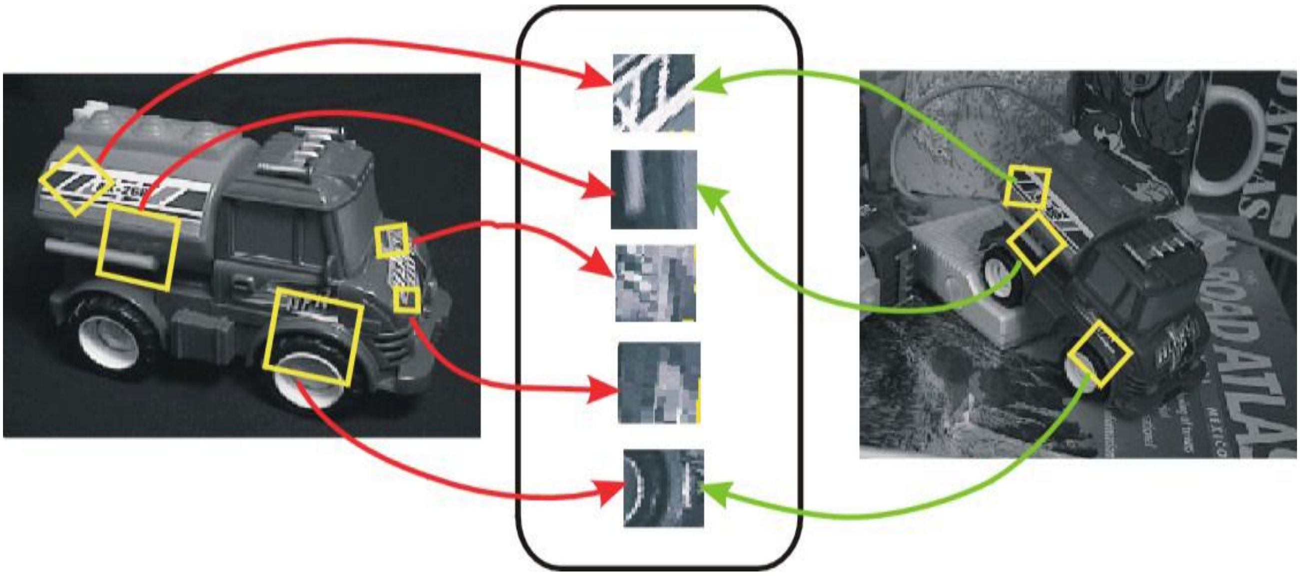

Figure 1. SIFT features

In Figure 1 , it can be marked that the second image contains some objects which are the rotated versions of the first image[5]. The SIFT algorithm is used to match the features in both the images.

2.2 SIFT ALGORITHM IN DETAIL:

2.2.1 ADVANTAGES OF SIFT ALGORITHM:

Some of the most important advantages of SIFT algorithm are as follows:

1. **Locality**: SIFT involves local features and thus this makes the algorithm more prone to occlusion and clutter.

2. **Distinctiveness**: the features extracted can be compared to objects in a large database.

3. **Quantity**: In this algorithm, even for minute objects, large number of features can be found.

4. **Efficiency**: The output of the SIFT algorithm is very close to real-time.

5. **Extensibility**: This algorithm can be used for wide variety of different types of features and thus adding robustness to the algorithm

2.2.2 OVERALL PROCEDURE:

The entire algorithm can be divided into 4 subparts:

1. Extrema detection at Scale space:
2. Keypoint localization
3. Orientation Assignment
4. Keypoint description

The methods are described as follows:

2.2.2.1 **Extrema detection at Scale Space:**

The aim of this method is to identify the scales and locations that retain their features under different views. In this step, such features are searched over the entire image across multiple scales. A continuous scale of Gaussian function is used over here. The function that is used to represent the scale space is $G(x,\sigma)$ which is the result of the convolution of the input image with the Gaussian kernel.

Figure 2: Gaussian Pyramid

Figure 2 shows the Gaussian Pyramid which is formed as a result of repeated application of the Gaussian kernel and sub-sampling of the image[5].

The image in scale space, represented by $G(x,\sigma)$ depicts the same information at different scale values. The image is scaled at different values in order to reduce redundancy[4]. The domain of the scale space variable is sampled at discretized steps in logarithmic values organized in a specified number of octaves which is then further subdivided into S sub-levels. The image is sub sampled at each octave.

The scale function can be given by the equation:

$$\sigma(o,s) = \sigma_0 2^{o+s/S},$$

Where o and s are the octave and the sub level that are currently used

S is the total no of sublevels per octave and

σ_0 is the base scale level.

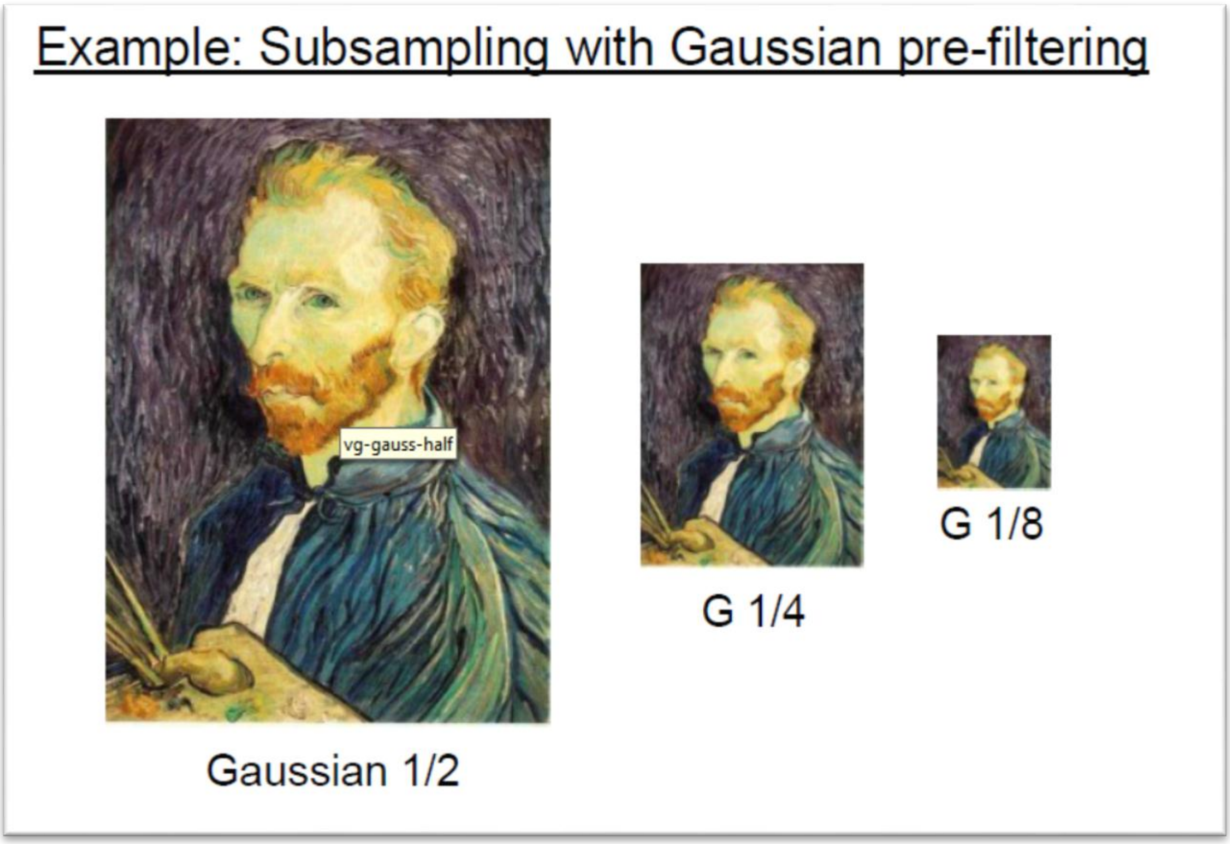

Figure 3 : Subsampling with Gaussian pre-filtering

Figure 3 shows the image at different levels of octave[5]. At each successive level, the image is sub-sampled by ½.

The entire scale space is divided into O octaves, each octave being further subsampled into S sublevels. In each octave, the starting image is convolved with Gaussian kernel producing a set of smoothed images in scale space.

Then the next step is to subtract the Gaussians at each level to produce the DOG (Difference of Gaussians).

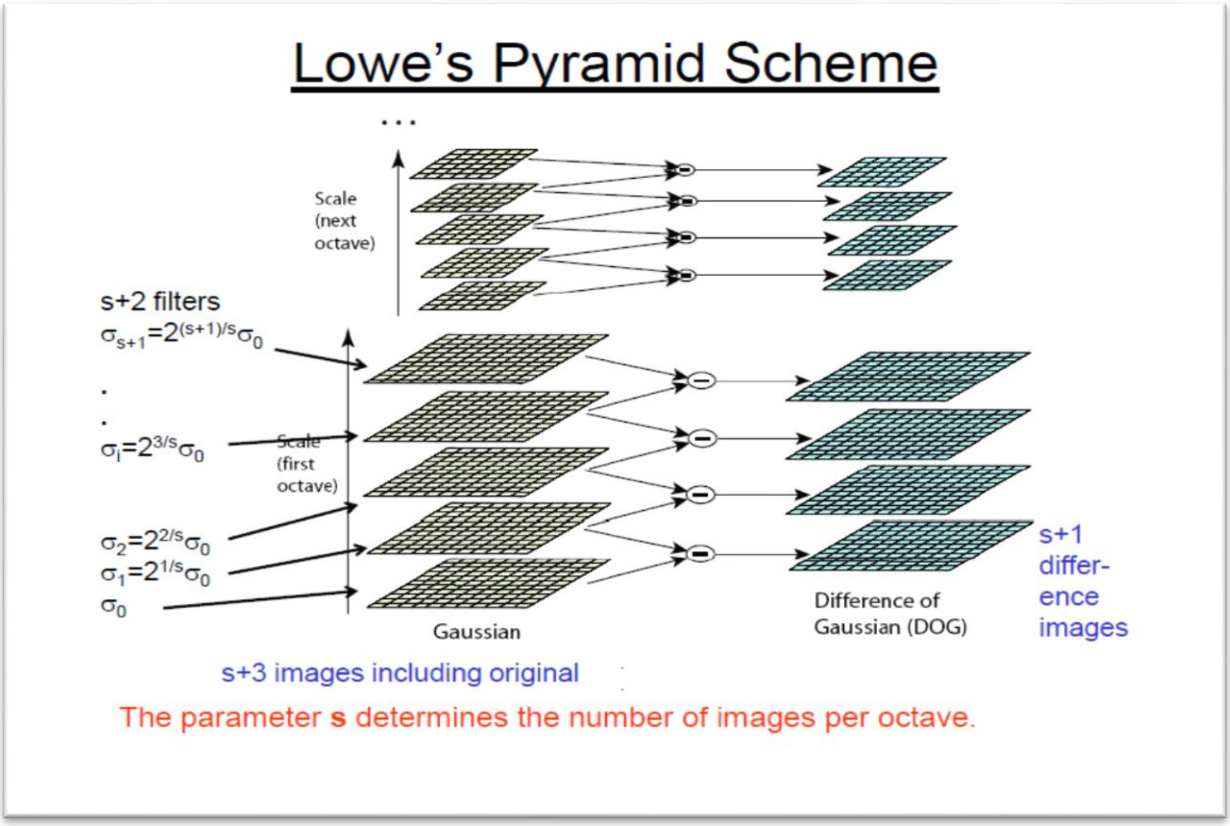

Figure 4. Images at Gaussian Scale and Difference of Gaussian Scale Space

Figure 4 shows the images blurred at different scales and the Difference of Gaussians of the Gaussian scale space images[5].

2.2.2.2 Keypoint localization:

The maximum is determined by the comparison of the pixel at the center of a 3 X 3 window with the surrounding 8 pixels as well as the 9 pixels in the layer above and below as shown in Figure 5. In total, the local maximum is found out after performing 27 comparisons[5].

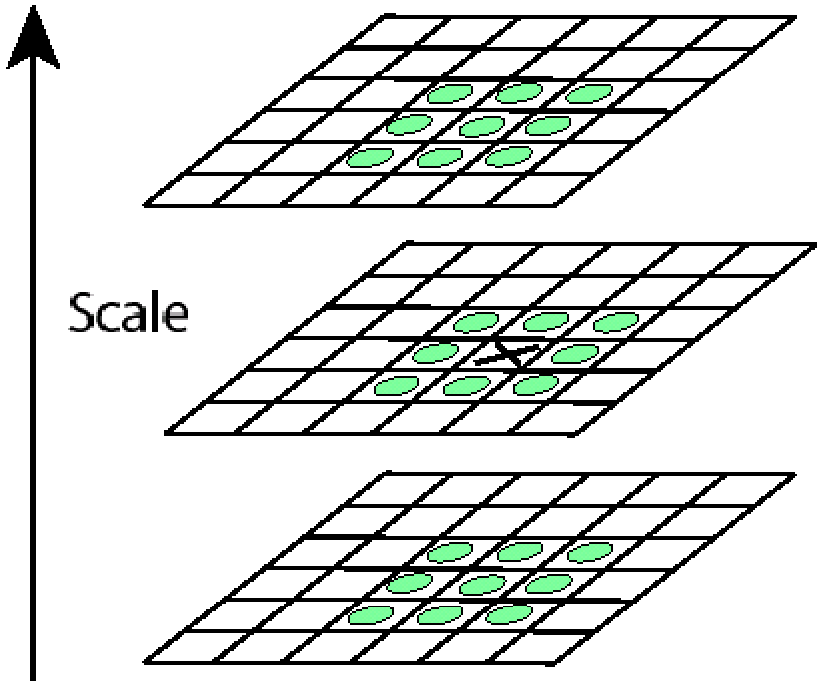

Figure 5. Maxima Detection

2.2.2.3. Orientation assignment:

After a keypoint is found, the next step is to compute the information regarding

- Location,
- Scale, and
- The ratio of the principal curvatures.

After the information about the keypoint is retrieved, a histogram is created showing the directions of the local gradient at selected scale values. At peaks of the smoothed histogram, the canonical orientation is assigned[4]. Stable 2- dimensional coordinates i.e. (x, y, scale, orientation) are specified by the keypoints.

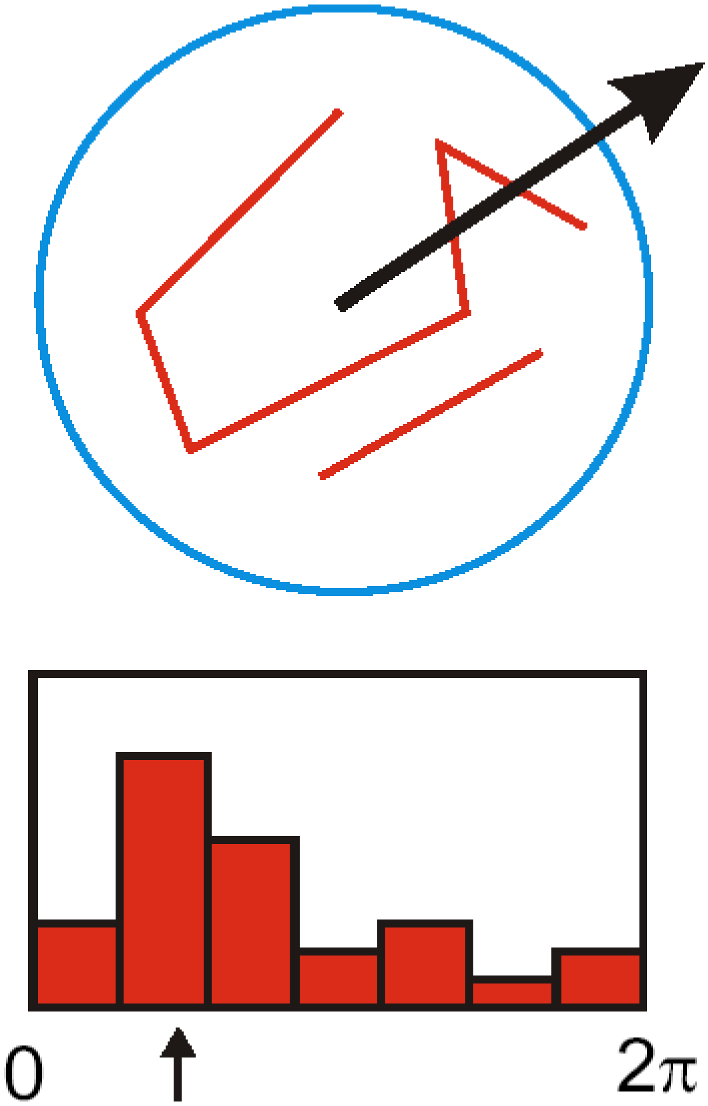

Figure 6. Orientation assignment to the keypoints

The keypoints are chosen based on their stability. The best orientations are chosen for each keypoint region as shown in Figure 6[5].

2.2.2.4. Keypoint description :

In the Orientation assignment step, each keypoint is assigned a scale, location and orientation. The task of this step is to determine a descriptor for the region local to that keypoint which should be highly distinctive and it should retain its features after facing variations in illumination and viewpoint.

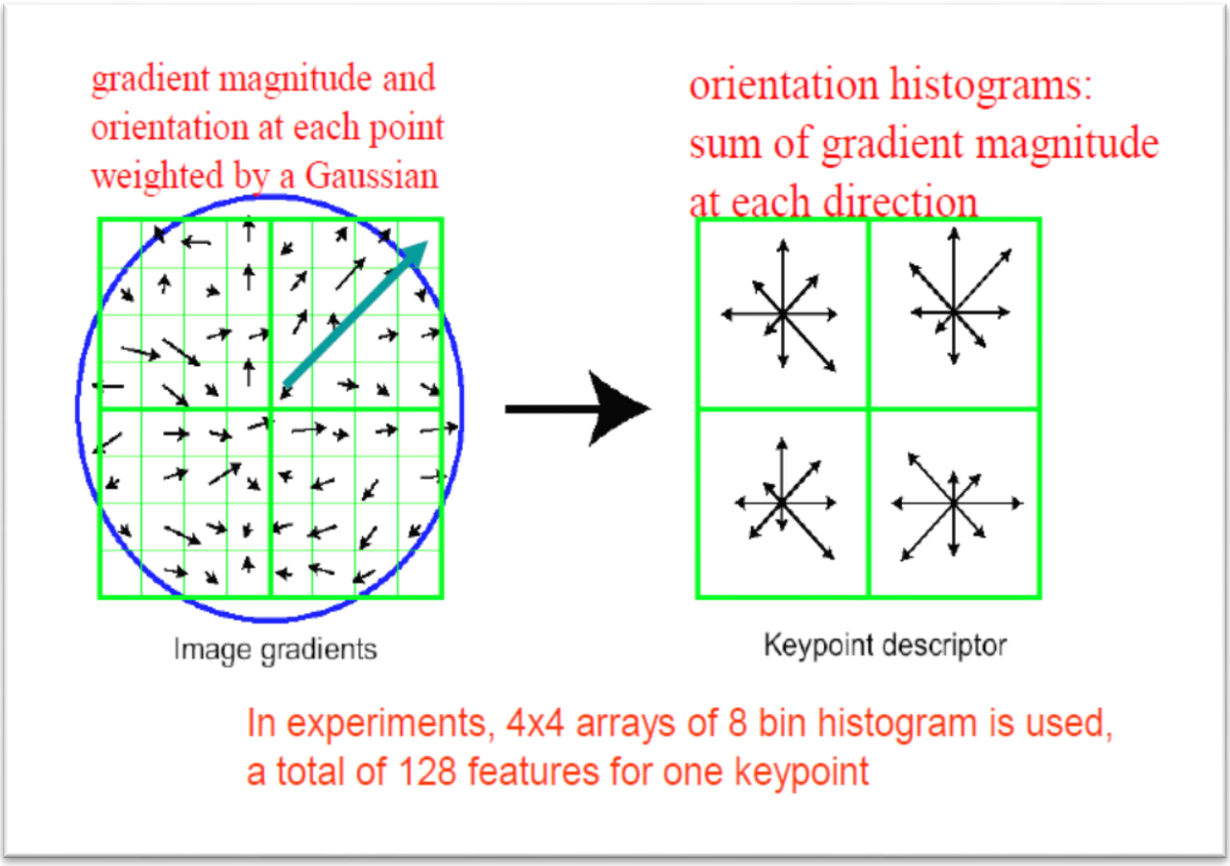

Figure 7. Keypoint description

The key point description uses the normalized region surrounding the keypoint. The gradient magnitude and orientation are calculated at each point in the region.

In Figure 7, an orientation histogram is created is created over the 4 X 4 subregions[5]. A vector of 128 values is produced as a result of 4 X 4 times 8 directions.

CHAPTER 3

FPGA IMPLEMENTATION

3.1 FPGA IMPLEMENTATION OF THE SIFT ALGORITHM:

The entire overview of the SIFT algorithm can be shown as in Figure 8[2].

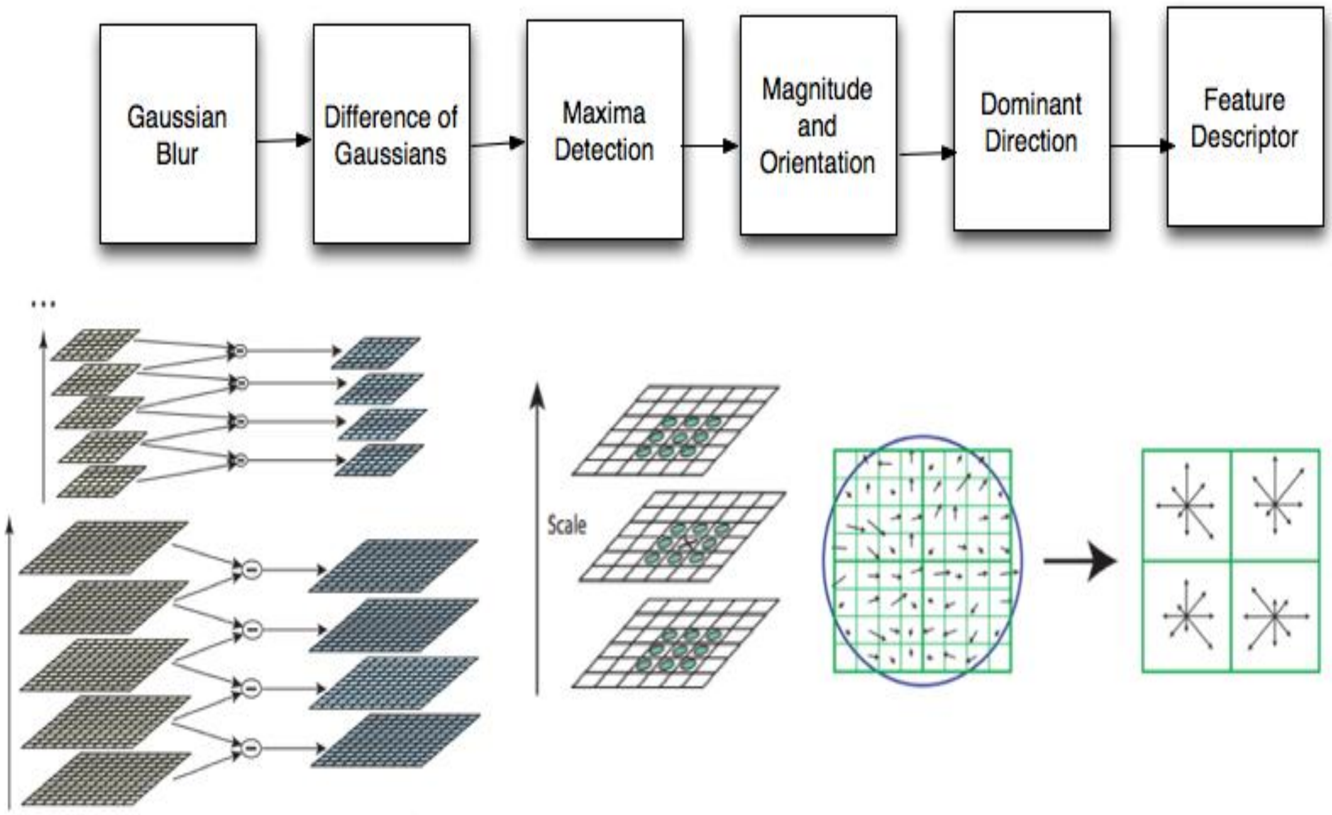

Figure 8. Overview of SIFT algorithm

3.2 IMPLEMENTATION ON FPGA

The implementation of SIFT on FPGA can be categorized under the following groups:

1. Gaussian Blur

2. Difference of Gaussians

3. Maxima Detection

4. Magnitude and Orientation

5. Dominant Direction

6. Feature descriptor

The above steps can be described as under:

3.2.1 **Gaussian Blur**:

The implementation of the Gaussian Blur is done using cyclic buffers which can buffer image data lines concurrently as shown in Figure 9[2].

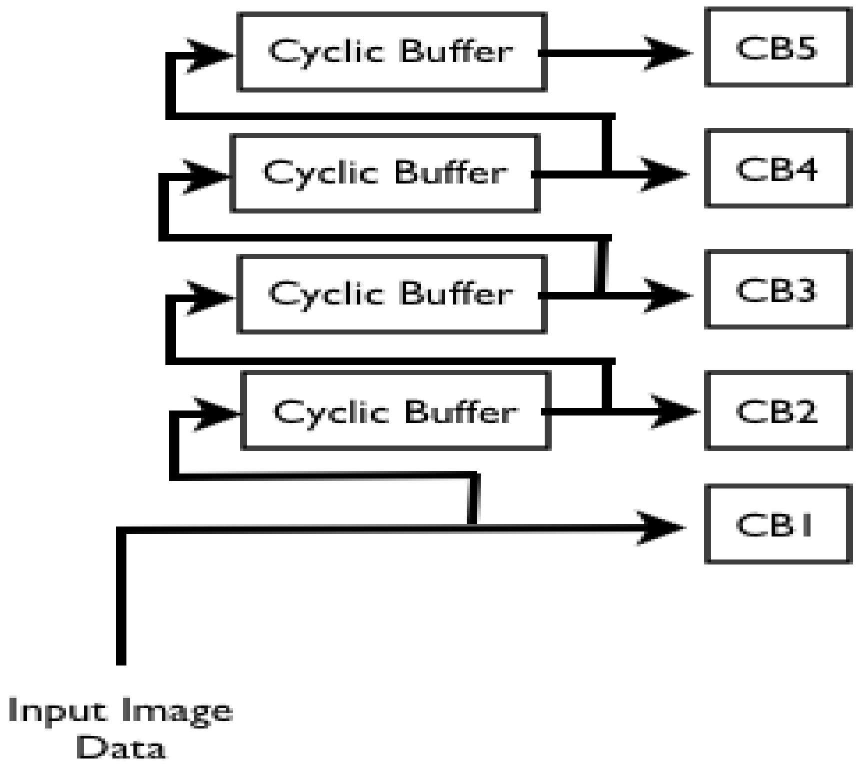

Figure 9. Processing of image data lines using cyclic buffers

The original image is blurred at different levels. Different window sizes of 1 X 1, 3 X 3, etc. are used upon which blur is performed. The blur vector used for convolution with the n X n window corresponds to the nth row of the Pascal's triangle[1]. The matrix used for the convolution can be depicted as:

$$\begin{bmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_n \end{bmatrix} \cdot \begin{bmatrix} p_1 & p_2 & \cdot & \cdot & \cdot & p_n \end{bmatrix} = \begin{bmatrix} p_1p_1 & p_1p_2 & \cdot & \cdot & \cdot & p_1p_n \\ p_2p_1 & p_2p_2 & \cdot & \cdot & \cdot & p_2p_n \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ p_np_1 & p_np_2 & \cdot & \cdot & \cdot & p_np_n \end{bmatrix}$$

Where p_i denotes the ith number in the nth row of the Pascal's triangle.

3.2.2 **Difference of Gaussians:**

The Difference of Gaussians(DoG) are computed by subtracting the blurred images from one another. Figure 10 shows the entire process[2].

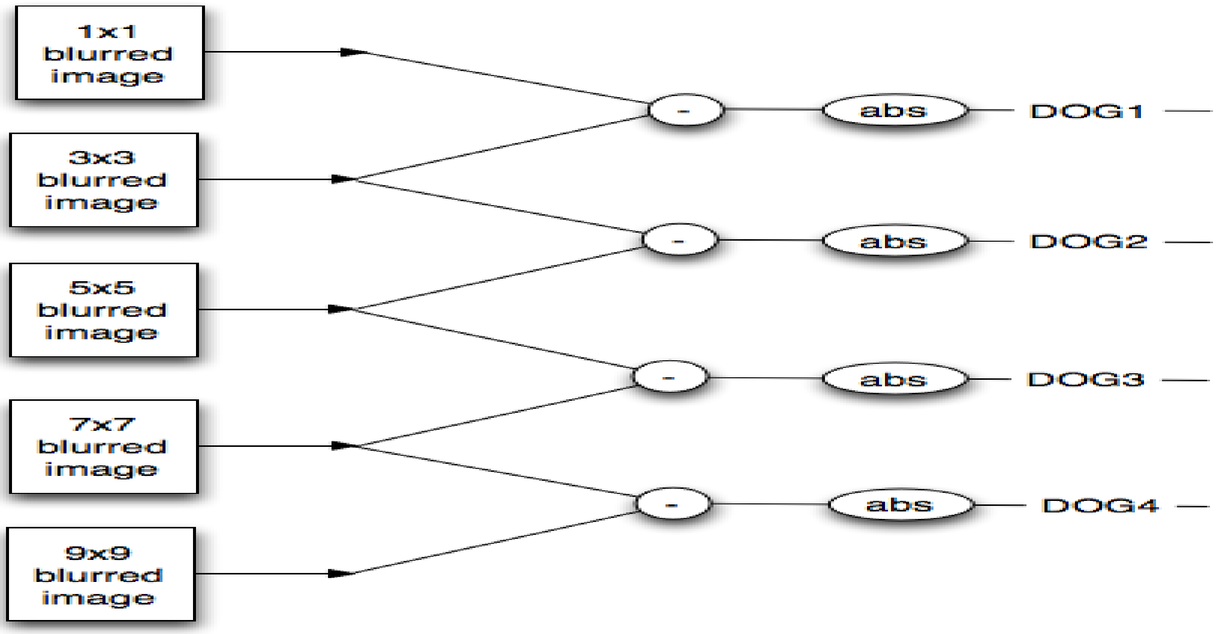

Figure 10. Difference of Gaussians(DoG)

3.2.3 **Maxima Detection**:

The maxima is computed by comparing the pixel at the centre with the 26 pixels surrounding it in the 3 X 3 X 3 three dimensional window surrounding the pixel as shown in Figure 11[2].

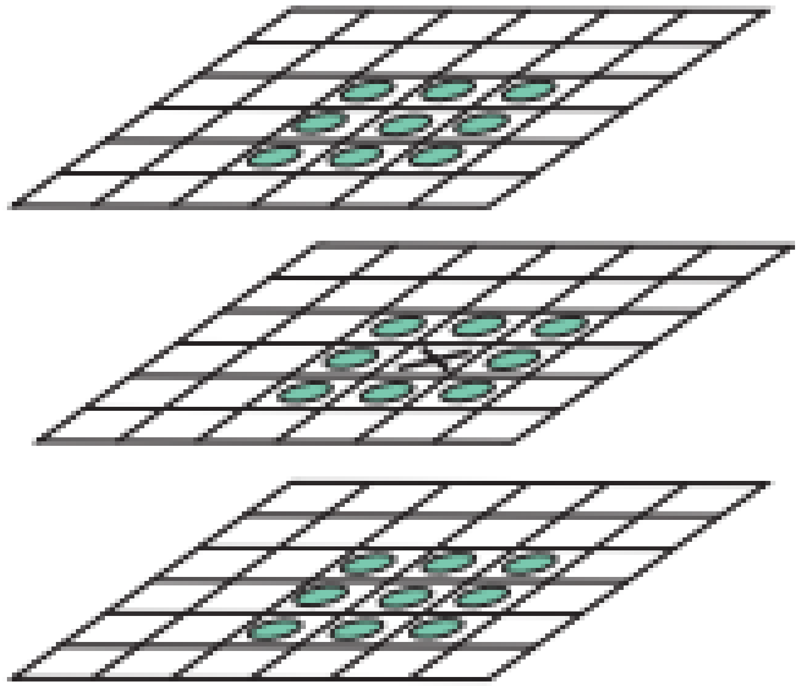

Figure 11. 3 X 3 X 3 window surrounding the center pixel

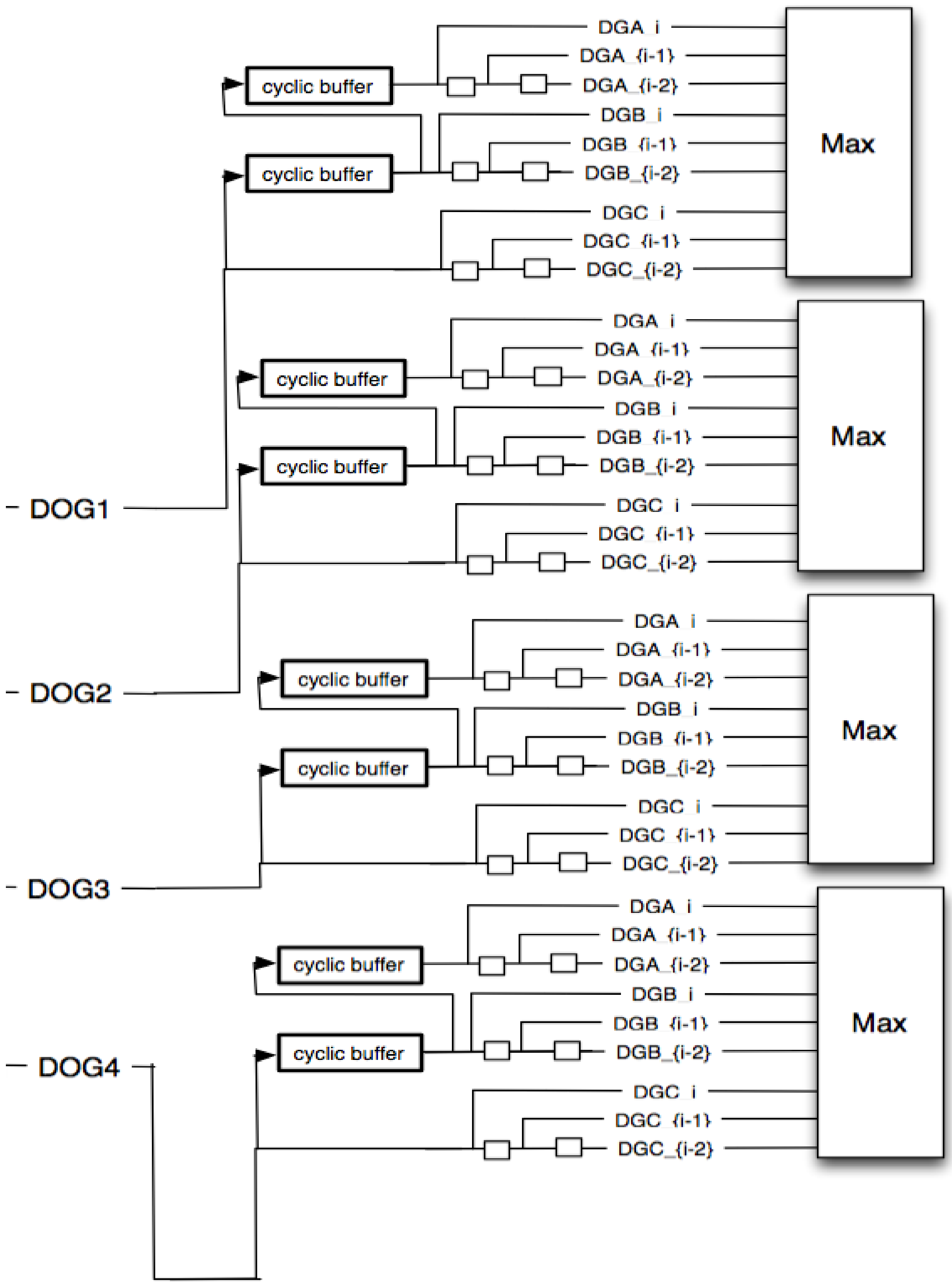

Figure 12. Maxima Detection

Figure 12 depicts the entire process of maxima detection[2]. The image data is streamed by using the cyclic buffers and then the maxima are then obtained by comparing the pixels of 3 consecutive DOGs.

3.2.4 **Magnitude and Orientation Calculation**:

The magnitude and orientation of the keypoint are calculated by using the finite differences in a 16 X 16 window around the maximum. BRAM stores the differences in x, or dx and the differences in y, or dy and the magnitude is calculated by using the CORDIC. Figure 13 depicts the process and hardware[2].

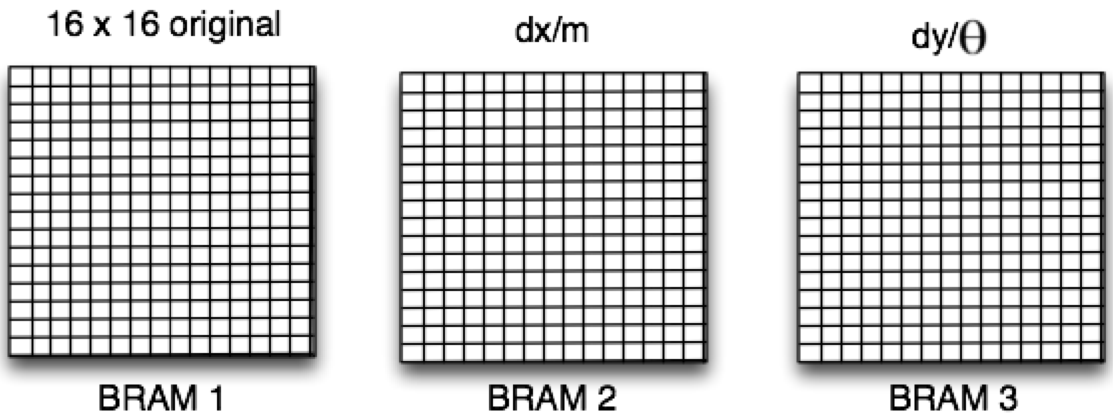

Figure 13. Magnitude and Orientation Calculation

3.2.5 **Dominant Direction:**

After the magnitude and orientation are calculated, the dominant direction is determined which passes a certain level.

3.2.6 **Feature descriptor:**

It is generated by dividing the 16 X 16 window surrounding the keypoint. The 16 X 16 window is split into 16 4 X 4 windows. In the next step, the orientations are calculated and fed to 8 bin histograms. Then the 16 X 16 window is converted to obtain a feature descriptor which is obtained as a result of normalizing the 4 X 4 X 8 feature vector.

CHAPTER 4

OBSERVATIONS

4.1 SIFT IMPLANTATION ON MATLAB:

Experimental Set-up: The experiment has been done on MATLAB 2013a in Intel i3-370M Processor of 2.40 HHz.

4.1.1 **Query image:** The query image is the image showing the hand gesture of the user .Figure 14 is one such query image.

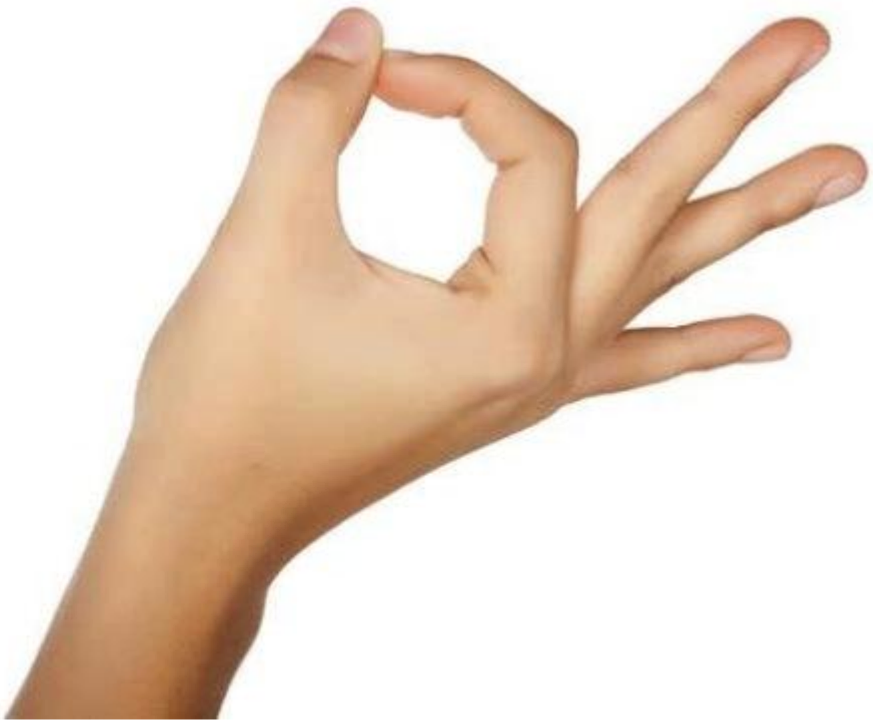

Figure 14. Query Image

4.1.2. Matching image from database: The image in the database file that matches with the query image is shown in Figure 15.

Figure 15. Database image

4.1.3.Gaussian Blurring images at different scales and Difference of Gaussians:

The images are blurred at different scales using Gaussian function and then the Difference of Gaussians is calculated. The Difference of Gaussians of the Query image at different scales are depicted as shown in Figures 16 and 17.

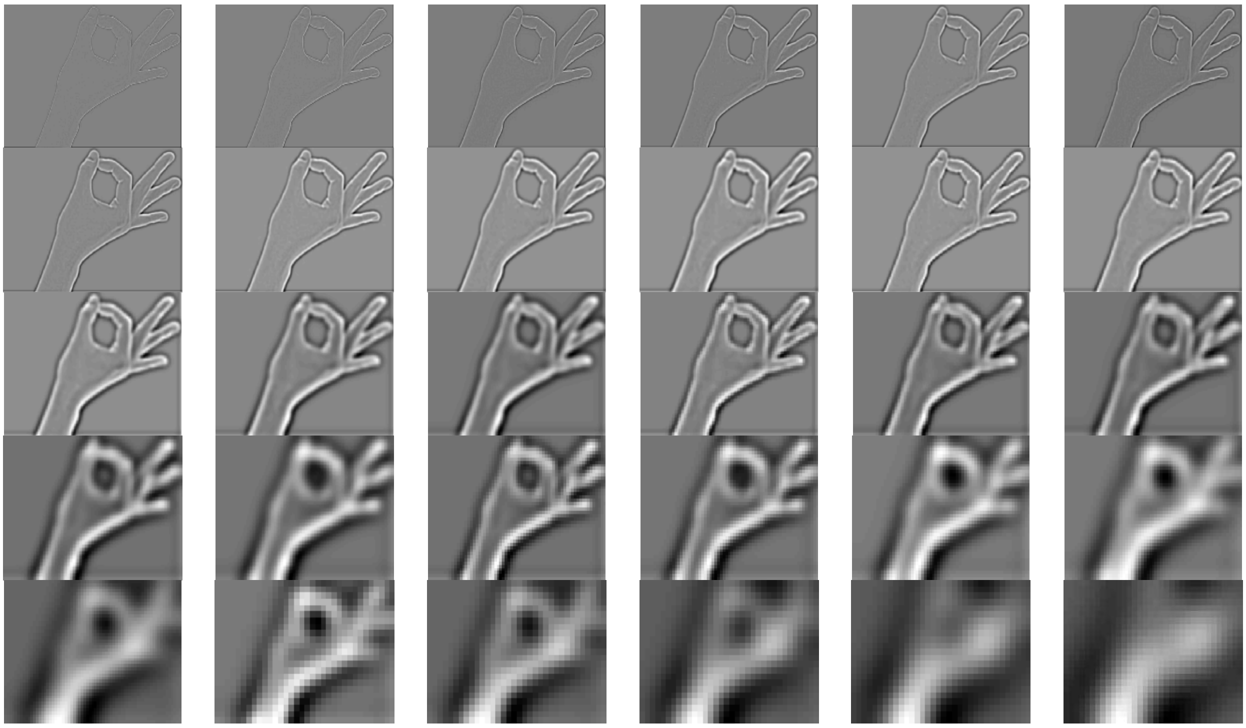

Figure 16. Difference of Gaussians after performing Gaussian Blur of Query image

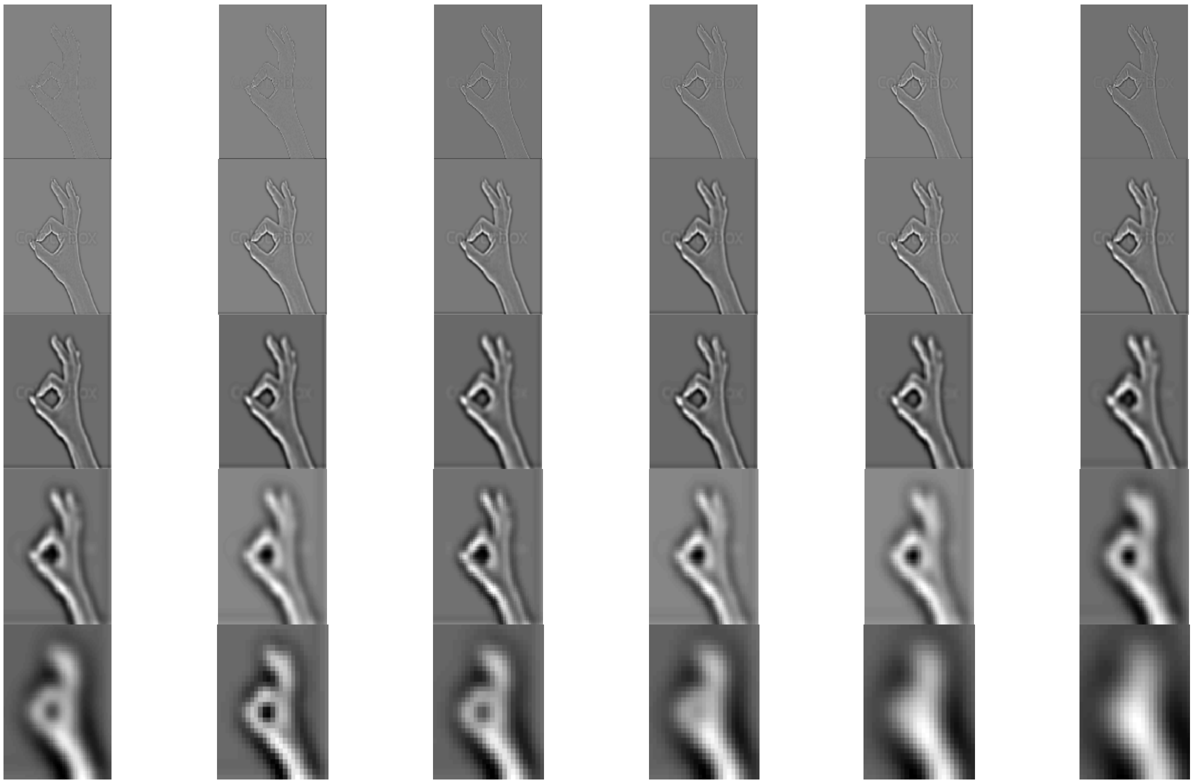

Figure 17. Difference of Gaussians after performing Gaussian Blur of Database image

4.1.4. Keypoints determination of the Query and the Database images:

Figure 18 shows the keypoints for the database image

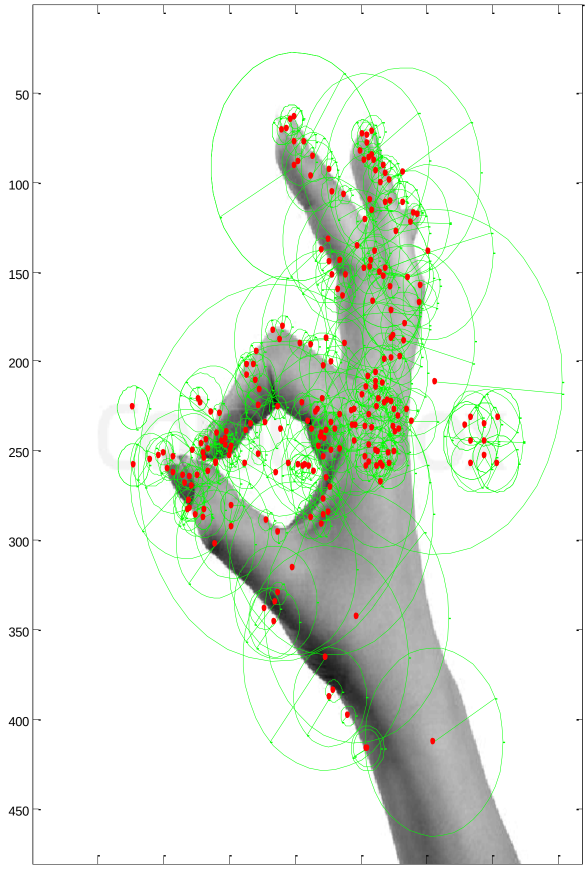

Figure 18. Keypoints of the Database image

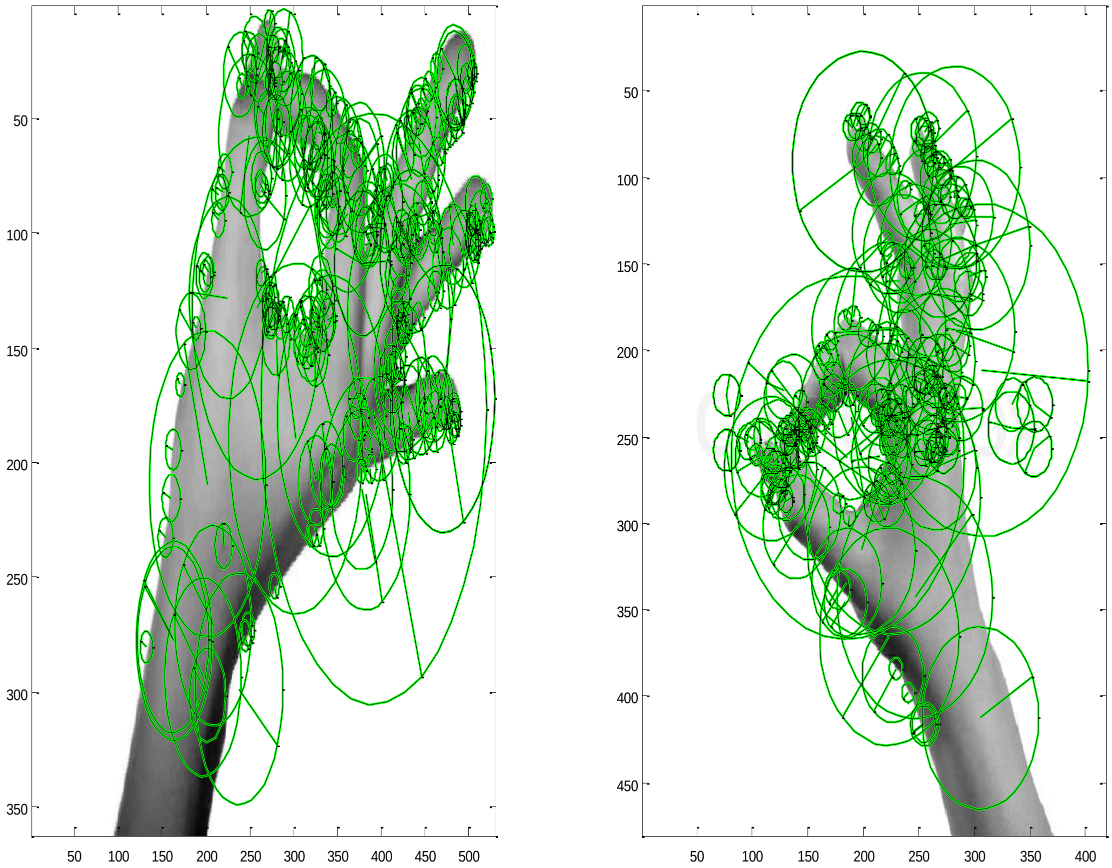

Figure 19. Keypoints in the query image and the database image

Figure 19 shows the keypoints in the query image and the database image.

4.1.5. Orientation and magnitude assignment

The magnitude and orientation of the keypoints are computed.

Figure 20 shows the orientation histogram at a particular keypoint in the database image.

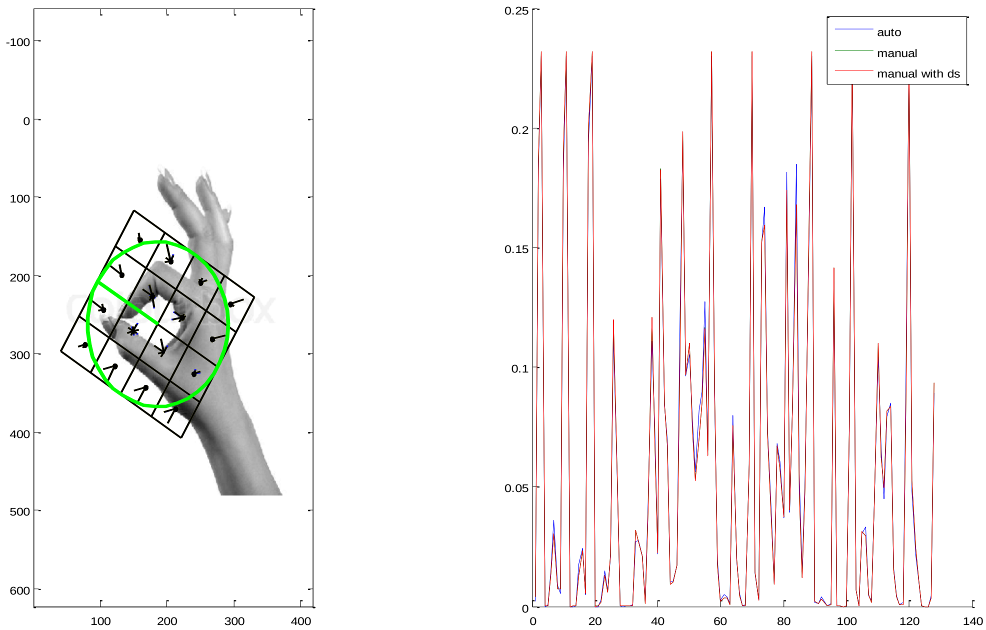

Figure 20. Orientation Histogram

Figure 21 shows the orientation assignment of the keypoints in the query image.

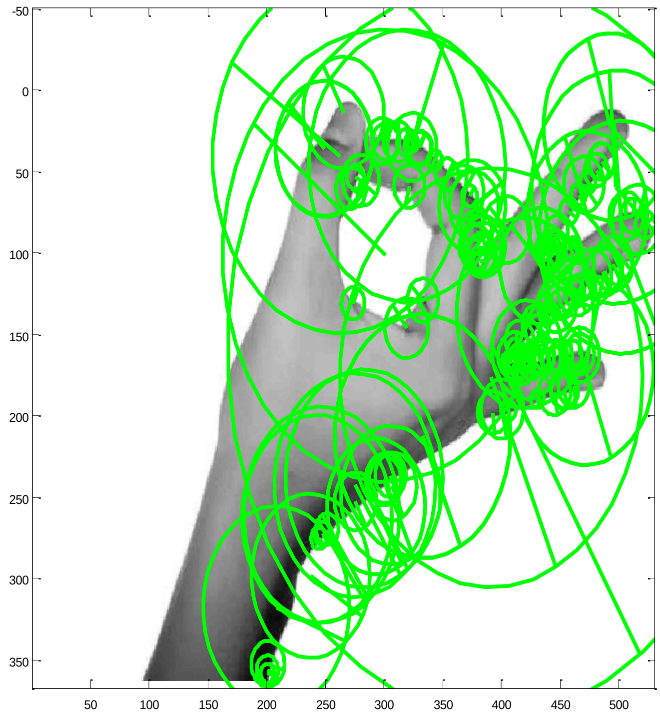

Figure 21. Orientation assignment of the keypoints

4.1.6. Matching of the keypoints

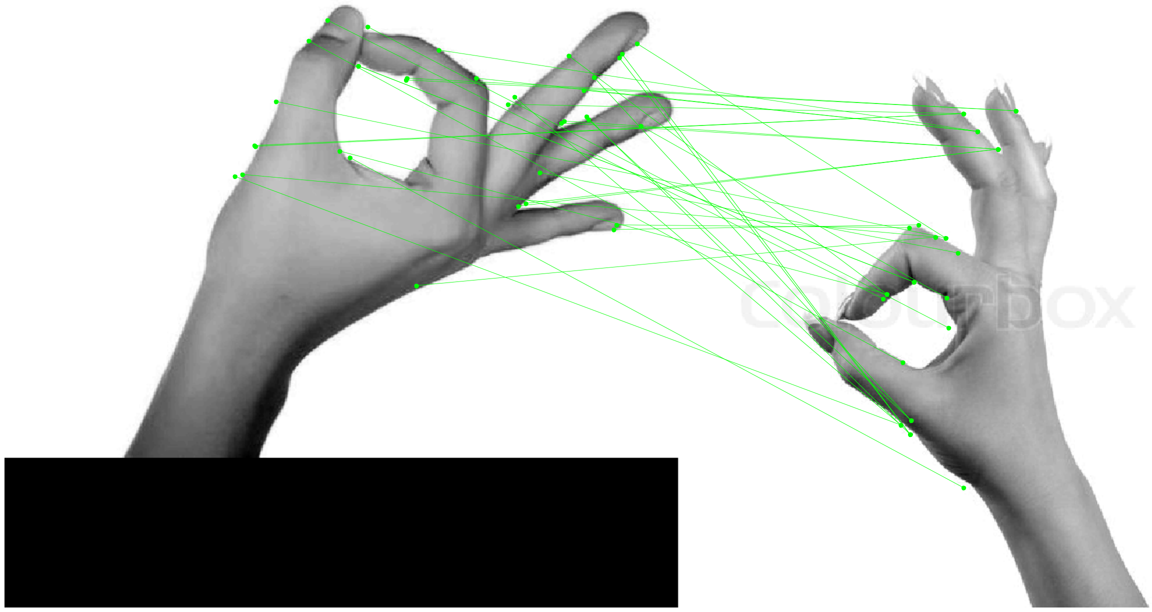

Figure 22. Matching images

The keypoints of the query and the database image are matched as depicted in Figure 22.

4.2 FPGA Implementation:

VHDL programs are written for finding the Gaussian Blur of a window of an image using the convolution of the image window with the Gaussian kernel.

Then the Gaussians of the images are subtracted to find the Difference of Gaussians (DOGs).

CHAPTER 5

CONCLUSION AND FUTURE WORK

SIFT is a powerful algorithm for computer vision. It can be used to detect shapes efficiently and is invariant to changes in scale, illumination and rotation. In this project , the algorithm is implemented in MATLAB. The query image was successfully matched with the similar image present in the database. Codes for Gaussian Blur and Difference of Gaussians were written in VHDL for implementing the algorithm in FPGA.

The advantages of using SIFT for the feature detection are:

1.**Locality**: SIFT involves local features and thus this makes the algorithm more prone to occlusion and clutter.

2.**Distinctiveness**: the features extracted can be compared to objects in a large database.

3.**Quantity**: In this algorithm, even for minute objects, large number of features can be found.

4.**Efficiency**:The output of the SIFT algorithm is very close to real-time.

5.**Extensibility**: This algorithm can be used for wide variety of different types of features and thus adding robustness to the algorithm

There are some disadvantages however such as:

1.**Complexity of the algorithm:** The algorithm is quite complex. So writing the code in VHDL for this algorithm is a bit difficult.

2.**Hardware requirement:** The hardware requirement for the FPGA implementation of the SIFT algorithm can be quite high.

FUTURE SCOPE:

The entire SIFT algorithm can be implemented in FPGA . An efficient emotional recognizer can be fully designed on FPGA which can successfully aid the autistic children to improve their interpersonal communication.

REFERENCES:

[1] Bonato, Vanderlei, Eduardo Marques, and George A. Constantinides. "A parallel hardware architecture for scale and rotation invariant feature detection."*Circuits and Systems for Video Technology, IEEE Transactions on* 18.12 (2008): 1703-1712.

[2] http://www.cs.cmu.edu/afs/cs/user/mjs/ftp/thesis-program/2012/theses/tabib.pdf

[3] Huang, Feng-Cheng, et al. "High-performance SIFT hardware accelerator for real-time image feature extraction." *Circuits and Systems for Video Technology, IEEE Transactions on* 22.3 (2012): 340-351.

[4] Lowe, David G. "Distinctive image features from scale-invariant keypoints."*International journal of computer vision* 60.2 (2004): 91-110.

[5] http://www.robots.ox.ac.uk/~vedaldi/assets/sift/sift.pdf

[6] http://www-inst.eecs.berkeley.edu/~cs150/sp13/agenda/lec/lec01-intro.pdf

[7] Zhong, Fang, David W. Capson, and Derek C. Schuurman. "Parallel architecture for PCA image feature detection using FPGA." *Electrical and Computer Engineering, 2008. CCECE 2008. Canadian Conference on*. IEEE, 2008.

[8] Yao, Lifan, et al. "An architecture of optimised SIFT feature detection for an FPGA implementation of an image matcher." *Field-Programmable Technology, 2009. FPT 2009. International Conference on*. IEEE, 2009.

[9] Bouris, Dimitris, Antonis Nikitakis, and Ioannis Papaefstathiou. "Fast and efficient FPGA-based feature detection employing the SURF algorithm." *Field-Programmable Custom Computing Machines (FCCM), 2010 18th IEEE Annual International Symposium on*. IEEE, 2010.