FAST & ERROR RESILIENT PROCESSING UNIT OF CORNER DETECTION FOR THE APPLICATION OF NOISE BASED CORNER DETECTION SCHEME

PREETI BALA SAHU
Electronics & Telecommunication department, Chouksey Engineering College Bilaspur, Chhattisgarh
E-mail: preeti.sahu62@gmail.com

Abstract—In present era every multimedia device requires fast and good quality image/video. There is a rapid demand of real time transmission applications so for those applications there is need of some application specific processing unit which should also make justice with battery power consumption. As we know in present stage there are some areas where we need to find the corner points related to Defence, Aeronautics, Traffic. In this type of application there is need of some device which will detect the corner points. In this application there is need of fast processing unit which is not possible by pure accurate unit so for reduction of those issue in this work we will present a novel error resilient algorithm and architecture which will reduce all previous problems. Here we also design our own hardware unit using VLSI technology which is based on powerful HDL “verilog”. Our main motto is to make justice with SPAA(Speed, Power, Area & Accuracy) Metrics. For application analysis we will use Noise based Corner Detection because here we have to verify the quality level of our proposed corner detection. For quality analysis we will use Image quality parameters. In this thesis we are presenting an novel algorithm which is based on fixed point logic. I will compare my proposed algorithm & architecture with previous existing approach. Implementation of proposed algorithm will be done by Matlab and hardware implementation will be done by using of Verilog on Xilinx 14.2 simulator. Verification will be done on Modelsim.

Index Terms—SPAA, ERROR, HDL, FAST.

I. INTRODUCTION

1.1 Overview:

Digital Image:
A digital image is representation of a two dimensional image as a finite set of digital values, called picture elements or pixels. Pixel values typically represent gray levels, colors, heights; etc.

Digitization implies that a digital image is an approximation of a real scene. Common image formats include:
- 1 sample per point (B&W or Grayscale)
- 3 samples per point (Red, Green, and Blue)
- 4 samples per point (Red, Green, Blue, and “Alpha”)

1.2 Image Registration:
Image registration is an important step of remote sensing image processing, and it is pre-processing of image mosaic, image fusion, relative radiometric normalization, land use classification, and land use change detection, etc. Primary task of the remote sensing image registration is to find correct ground control point correspondences on the base image and the warp image. However, it is low efficient to select ground control point by manual work and the final precision is unstable. With the development of the computer science, pattern recognition, artificial intelligence, and image processing technology, many kinds of full-automatic or semi-automatic image registration algorithms are proposed, the key procedures including two aspects: one is to detect ground control point automatically.

1.3 Introduction to Corner Detection:
Because the gray gradient of corner is acutely mutative, it has a high accuracy in control point matching process based on gray level. For this reason, corner detection is the first step of many full-automatic or semi-automatic image registration algorithms. SUSAN operator, Harris operator, Wang operator and Shenjun operator are universal corner detection
methods. Especially SUSAN operator, it is non-directional, easy acceptable, noise eliminated, and high accuracy. But it requires so much image information also, it is difficult to choose correct luminance threshold and geometry threshold.

The computational processes of other operators are also more complicated. A new adaptive threshold discriminating algorithm for remote sensing image corner detection, which confirms the direction of corner by analyzing eight neighborhood direction gray gradients, then adopts the neighborhood tracking and uses two thresholds of gray gradient to detect the correct corner point. The thresholds of gray gradient are obtained by calculating probability density curvature extremum of gray gradient. Though it has certain directionality, this algorithm can effectively eliminate noise, its computation procedure is simple, the result of threshold extraction is more objective, and the final corner location is more accurate.

II. LITERATURE REVIEW

2.1 Previous Work:
Corner is the two-dimensional point of rapid change image brightness, or curve maximum curvature point in image edge. Corner is an important local feature of the image, it focused on a number of important information of the shape of the image to reflect the image of the local features, it can match images more reliable. Corner has such rich feature information, so that it hold important features of graphic images and effectively reduce the amount of data information, it improves the speed of operation, it makes easy to a image reliable matching, makes real-time processing possible. It has the rotational invariance and does not change with the light conditions change.

In 2010, Lei et al. proposed “A study on comparisons of corner detection algorithms:
1. Harris Multiscale Corner Detection algorithm
Harris is the successor algorithm of Morave algorithm.
Morave proposed corner detection formula is:
\[
E(u,v)_{(x,y)} = \sum w(x,y)|I(x+u, y+v) - I(x, y)|^2
\]
E is change in brightness value when move a (u, v) small window from the point (x, y); w(x, y) for the Gaussian smoothing factor. The nature of Style is autocorrelation of two-dimensional signal. The methods of Harris corner detection algorithm are: design a local detect window in the image, the window makes the minor movements in all directions, examining the average energy changes of the window, the energy changes in the value exceeds a set threshold, extracted the center pixel of the window for the corner. The gray scale of pixel (x, y) as f(x, y), the Gray-scale changes in intensity of each image pixel (x, y) after moving (u, v) is expressed as:
\[
E_{u,v}(x, y) = \sum_{u,v} W_{u,v} u, v \{f \circ (x+u, y+v) - f \circ (x, y)\}_2
\]
W u,v is coefficient of Gaussian windows in the position (u,v). Harris algorithm detects the corner through differential operator and self-correlation matrix.

2. SUSAN Corner Detection Algorithm formula is:
\[
c(x_0, y_0, x, y) = \begin{cases} 
1 & \text{when } |f(x_0, y_0) - f(x, y)| < T \\
0 & \text{when } |f(x_0, y_0) - f(x, y)| > T
\end{cases}
\]

SUSAN algorithm is a low-level image processing algorithms, directly using the image gray-scale effective detect the edge, corner. It is simple, anti-noise capability and faster processing speed and so on. (x, y) and (x_0, y_0) are separately coordinates of template center and other points in the template, c(x, y ; x_0,y_0) is the results of comparing, f (x, y) is the brightness value of the point, t is the gray-scale difference threshold.

In 2002 Azhar presents a technique for the corner detection with the help of wavelet transformation. According to this paper, author first find the wavelet transformation and after that they will apply corner detection algorithm for the detection of corners points.

In 2014 Shen et al. use another approach to find corner and that approach is Corner detection using Gabor filters. It was proved that the Gabor filters are suitable to be used as the model of human visual system. Shen and Bai demonstrated Gabor filters are the optimal filters to extract local feature for various applications, such as face recognition and edge detection. Imaginary part of Gabor filters (IPGFs) is used to smooth the pixels of edge contours and extract the fine grey variation information. The sum of the normalised magnitude response at each direction (SNMRED) is embedded into the framework of the contour-based methods to develop a new corner detector. Here, the SNMRED is defined as a new corner measure to detect corners from edge contours, which also conforms the corner’s definition. The new corner measure uses not only the contours’ geometric features well but also the directional grey-variation information of the pixels of contours completely.

2.2 Image Parameters:
Here for image quality point of view we are using some scientific parameters those are like:

- **PSNR**
- **SSIM[17]**
- **FSIM[18]**

1. PSNR (Peak signal-to-noise ratio)
Peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in...
terms of the logarithmic decibel scale. PSNR is most commonly used to measure the quality of reconstruction of lossy compression codecs (e.g., for image compression). PSNR is most easily defined via the mean squared error (MSE). Given a noise-free $mn$-monochrome image $I$ and its noisy approximation $K$, MSE is defined as:

$$\text{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

The PSNR is defined as

$$\text{PSNR} = 10 \cdot \log_{10} \left( \frac{\text{MAX}^2}{\text{MSE}} \right)$$

$$= 20 \cdot \log_{10} \left( \frac{\text{MAX}}{\sqrt{\text{MSE}}} \right)$$

$$= 20 \cdot \log_{10} (\text{MAX}_I) - 10 \cdot \log_{10} (\text{MSE})$$

2. SSIM (Structural-Similarity-Based Image Quality Assessment)[17]

Natural image signals are highly structured: their pixels exhibit strong dependencies, especially when they are spatially approximate, and these dependencies carry important information about the structure of the objects in the visual scene. The Minkowski error metric is based on point wise signal differences, which are independent of the underlying signal structure.

$$\text{SSIM}(x,y) = \left( \frac{2\mu_x \mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \left( \frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right).$$

The motivation of our new approach is to find a more direct way to compare the structures of the reference and the distorted signals, where $\mu_x$, $\mu_y$ are mean intensity, $\sigma_x$, $\sigma_y$ are standard deviation, $C_1$, $C_2$ are constants.

3. FSIM (Feature Similarity Index for Image Quality Assessment)[18]

The computation of FSIM index consists of two stages. In the first stage, the local similarity map is computed, and then in the second stage, pools the similarity map into a single similarity score. The separation of the feature similarity measurement between $f1(x)$ and $f2(x)$ into two components, each for PC or GM. First, the similarity measure for $PC1(x)$ and $PC2(x)$ is defined as

$$\text{SPC}(x) = 2 \cdot PC1(x) \cdot PC2(x) + T1/\text{PC12}(x) + T1 \cdot \text{PC22}(x) + T1$$

where $T1$ is a positive constant to increase the stability of $\text{SPC}$. Similarly, the GM values $G1(x)$ and $G2(x)$ are compared and the similarity measure is defined as

$$\text{SG}(x) = 2 \cdot G1(x) \cdot G2(x) + T2/\text{G12}(x) + G22(x) + T2$$

where $T2$ is a positive constant depending on the dynamic range of GM values. Then, $\text{SPC}(x)$ and $\text{SG}(x)$ are combined to get the similarity $\text{S}(x)$ of $f1(x)$ and $f2(x)$. We define $\text{S}(x)$ as $\text{S}(x) = \text{SPC}(x) \cdot \text{SG}(x)$. We use $\text{PCmax}(x) = \max(\text{PC1}(x), \text{PC2}(x))$ to weight the importance of $\text{SPC}(x)$ in the overall similarity between $f1$ and $f2$, and accordingly the FSIM index between $f1$ and $f2$ is defined as

$$\text{FSIM} = \frac{\sum_{x} \text{S}(x) \cdot \text{PC1}(x) \cdot \text{PC2}(x)}{\sum_{x} \text{PC1}(x) \cdot \text{PC2}(x)}$$

III. PROPOSED METHODOLOGY

Here we will propose a novel algorithm & architecture of corner detection with the motto of reducing the time complexity at algorithm level. Similar at architecture level we will make justification with SPAA metrics according to our thesis theme. According to that concept we will design fast algorithm and we will reduce the time complexity. After that we will design architecture of processing unit by using our proposed Corner Detection algorithm. At initial stage we will design our own novel algorithm after that we will design of our architecture of proposed novel algorithm. We will apply our proposed logic with previous existing logic & make a comparative analysis in terms of time & image quality. At algorithm level we use error resilient logic.

There we will compare the output image quality with different image quality parameters. Now at hardware stage we will use Verilog HDL and design our architecture on Xilinx 14.2. Virtually we will do analysis on Vertex 6 FPGA. Using verilog verification we will check the efficiency of our propose design. This project will develop in both level means algorithm and architecture level. So for algorithm level we will use MATLAB. Similar architecture level design will be done on Xilinx and verification will be done on Modelsim.

According to our proposed approach we will find the corners points for the noisy images. Here at initial stage we will resize image and reduce size with 25%, then we will apply Gaussian smooth filter. So according to this filter we are using 3X3 Gaussian window where it require only three pixel for the calculation of smooth value. After that we again resize image into original size, then we will find derivatives value which we will find by using of our proposed mask. According our proposed mask we will take only fraction value where no need of any kind of multiplier using we can use shifter logic which will generate output. At last step we will find the corner value by the help of existing formula but in our proposed approach we will modify the formula in terms of fixed point.

<table>
<thead>
<tr>
<th>Table 3.1 Required software and tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development Tools (Algorithm)</td>
</tr>
<tr>
<td>Development Tools (Architecture)</td>
</tr>
<tr>
<td>Simulation &amp; Verification Tool</td>
</tr>
</tbody>
</table>

Proposed Derivatives matrix:

$$p = [0 \ 0.25 \ 0.5 \ 0.25 \ 0]$$

$$d1 = [0.125 \ 0.25 \ 0.000000 \ -0.25 \ -0.125]$$

$$d2 = [0.25 \ 0 \ -0.5 \ 0 \ 0.25]$$

Fast & Error Resilient Processing Unit of Corner Detection For the Application of Noise Based Corner Detection Scheme
Proposed Corner Formula:
\[ k = \frac{1}{32} \]
\[ cim = (Ix^2 \cdot Iy^2 - Ixy^2) - k*(Ix^2 + Iy^2)^2 \]
In this section we present implementation of some existing corner detection approach like harris, gabor, wavelet. Here we also implement our proposed corner approach.

Implementation of Harris[1]:
According to this approach first we will find the derivatives value than we will apply filter approach after that we will apply corner detection formula which will generate the output in terms of corner value.

Implementation of Gabor Based Corner[3]:
According to this approach first we will filter the input with the help of gabor filter approach after that , we find the derivatives value than we will apply filter approach after that we will apply corner detection formula which will generate the output in terms of corner value.

Implementation of Wavelet Based Corner[16]:
According to this approach first we will filter the input with the help of wavelet filter approach after that , we find the derivatives value than we will apply filter approach after that we will apply corner detection formula which will generate the output in terms of corner value.

Implementation of Proposed Corner detection: As we already discussed about our proposed corner detection. So here we will implement proposed architecture using of verilog and all simulation will be perform of xilinx 14.2. In this proposed approach we will propose a multiplier less architecture and as compare to previous approach. In our design there is only need of four pixel which is able to generate corner based image.

Hardware Implementation: Proposed algorithm will also propose a novel hardware unit with using of Verilog HDL.
1 Harris[1]:
2 Wavelet Based[16]:
3 Gabor Based[3]:
4. Proposed Corner Detection:
Here we will present the gate level and Lut based design which is generated by using of Xilinx 14.2. Here we will implement our fixed point mask logic which will produce a small logic. As compared to previous approach.
Top Module of our Approximate 3X3 Gaussian Filter[20] Architecture:

LUT BLOCK MODULE:

Top Module of our proposed Approximate Hariss Corner Detection Architecture:

IV. RESULT & ANALYSIS

A new algorithm will be proposed and that algorithm will be implemented by using of MATLAB, for image quality measurement I will use some scientific parameters like PSNR, SSIM, FSIM, GMSD. I will also propose hardware unit for my proposed algorithm which will reduce the area, power and speed problem. I will compare my proposed algorithm with previous approach hardware implementation will be done by using of Verilog on Xilinx 14.2 simulator. Verification will be done on Model sim.

During first stage the proposed method is implemented on MATLAB to thoroughly investigate the required time to detect corner with in an object and compare output image with various parameters.

Test Image:

Noisy Image

Time Complexity Analysis:

Image Bike:

PSNR Analysis:

Image Bike:
Our proposed technique is far better than gabor & wave late based approach as it requires less time complexity with very less reduction in image quality, generates very few false corners.

**Comparative Result of FPGA:**

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>Harris</th>
<th>Gabor</th>
<th>Wavelet</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic Block</td>
<td>1794</td>
<td>2069</td>
<td>2525</td>
<td>122</td>
</tr>
<tr>
<td>Delay(µSec)</td>
<td>44.4</td>
<td>67.31</td>
<td>51.621</td>
<td>13.54</td>
</tr>
<tr>
<td>Frequency(MHz)</td>
<td>22.52</td>
<td>14.85</td>
<td>14.85</td>
<td></td>
</tr>
</tbody>
</table>

**CONCLUSION**

According to this thesis we will resolve the previous existing problem which are latency, power, area. The key contribution of this work is to develop a fast corner detection algorithm. Using this work we will develop a SPAA aware error tolerant corner detection Unit. This proposed corner detection unit will require less area, power and speed. In this approach I will propose a new approach of approximation which will reduce some amount of accuracy. In proposed approach I will use only fixed point logic. According to our proposed approach we will find the derivatives values which is defined by fixed point techniques similar we change the formula of corner detection. Using this approach we reduce the timing complexity and hardware complexity with 30-40%.

**REFERENCES**


★★★