Image Stitching System Design Using Corner Detection on FPGA

O.Sai Lahari

August 2020

National Institute of Technology, Andhra Pradesh

Supervised by
Dr. V. Sandeep *Dept. EEE NIT AP, India*
K. Prasanna Kumar *Dept. ECE IISC, India*

Índice

Department of EEE ,NIT AP,Andhra Pradesh, India

1. Introduction

Image stitching or photo stitching is the process of combining multiple photographic images with overlapping fields of view to produce a segmented panorama or high-resolution image.It is also known as image mosaicing. Image stitching is basically combining two or more different images to form a single panorama view. Panorama is derived from a Greek word, Pan mean Everything and Horoma means to view. The entire image stitching process is done by images taken from a camera and then applying the process.

The main steps include image acquisition, image registration image warping.Image stitching techniques can be categorized into two general approaches: 1.direct and 2.feature based techniques. Direct techniques compare all the pixel intensities of the images with each other, whereas feature based techniques aim to determine a relationship between the images through distinct features extracted from the processed images.The feature based approach had the advantage of being more robust against scene movement and has the ability to automatically discover the over lapping relationships among an un ordered set of images.The feature detection step can be executed in a number of methods by selecting various features in the images which are unique and robust. Out of all the feature detected, the corners are most versatile and gives the very good results.

There are many feature detector techniques exist some of which are, Harris, Scale-Invariant Feature Transform (SIFT) , Speeded Up Robust Features (SURF) , Features from Accelerated Segment Test (FAST) , PCA-SIFT and ORB techniques.

Among all these techniques ORB is the best approch which is based on corner detetion and is proposed by Ethan Rublee in 2011.

Our main objective is to build **Image Stitching System Design Using Corner Detection on FPGA**. In this report we explaied our approach using ORB and RANSAC.

2. Our Approach

Flow Chart Representing Different Steps of Image Stitching

2.1. Image Acquisition

The first step of any image stitching system is image acquisition. Image acquisition can be broadly be defines as the action of retrieving an image from some sources. Typically, the images required for image stitching can be acquired by three different methods. These methods include translating a camera parallel to the scene, rotating a camera about its vertical axis by keeping the optical centre fixed or by a handheld camera. The acquired images are assumed to have enough overlapping that the stitching can be done and also some other camera parameters are known.

2.2. Feature Detection

The ORB algorithm uses the improved FAST (features from accelerated segment test) algorithm to detect feature points. The idea is that if a pixel is significantly different from the neighborhood pixels then it is more likely to be a corner point. The detection process is as follows:

2.2.1. Image feature point detection

Image feature point detection. First, select the pixel p in the image and assume its brightness is I_P. Set a brightness threshold T Then, take pixel p as the center, select 16 pixels on a circle with a radius of 3 and compare the gray value between pixel p and other pixels on the circle. If the bright-ness of consecutive N points on the selected circle is greater than $I_p + T$ less than $I_p - T$ then pixel$p$ an be considered as a feature point.

2.2.2. Feature point screening

Since the calculation of the FAST corner point is only to compare the difference in brightness between pixels, the number is large and uncertain and there is no direction information. Therefore, the ORB algorithm improves the original FAST algorithm which calculates the Harris response values for the original FAST corner points and sorts them according to the gray value and take the first N points. Harris response value calculation formula is as shown in equations below.

$$R = det(M) - k(trace(M))^2$$

$$\sum(x, y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

where R s the Harris response value, $M$is a2×2 matrix, k ranges from 0.04 to 0.06, $w(x, y)$ the image window function, I_x is the variation of the feature point in the horizontal direction, and I_y is the variation of the feature point in the vertical direction.

Image scale pyramids are constructed and sampled on each layer of the pyramid to extract FAST features and add scale invariance to feature points.

Determine the feature point direction. In order to make the extracted feature points have rotational invariance, the direction of the feature points is obtained by using the Intensity Centroid method. First, in a small image block B, the moment of the image block is defined as

$$m_{pq} = \sum_{x,y \in B} x^p y^q I(x, y), \quad p, q = 0, 1$$

where xnd y are pixel coordinates, and $I(x, y)$ the gray value of the corresponding pixel. Then, find the centroid of the image block by the moment:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

where the 0th moment (m_{00} the mass of the image block and the 1st moment ($m_{10}, m_{01}$the centroid of the image block. Finally, the geometric center O and the centroid C of the image block are connected to obtain a direction vector OC direction of the feature point is defined as:

$$\theta = arctan(\frac{m_{01}}{m_{10}})$$

Through the above steps, the FAST corner points have scale invariance and rotation invariance, which greatly improves their robustness in different images.

2.2.3. Generate feature point descriptors

After extracting the Oriented FAST feature points, the ORB algorithm uses the improved BRIEF algorithm [8] to calculate the descriptors for each point. BRIEF is a binary vector descriptor whose vector consists of a number of 0 and 1:

$$\tau(p; x, y) = \begin{cases} 1 & p(x) < p(y) \\ 0 & p(x) \geq p(y) \end{cases}.$$

where $p(x)$ the gray value at the field x around the image feature point, and $p(y)$ is the gray value at the field y around the image feature point. To reduce the effects of noise, Gaussian filtering is first performed on the image.

2.3. Feature Matching

Brute Force Matcher will match keypoints based on distance.Here it finds hamming distance between the pairs of points and gives the least distance points as matched points.For BF matcher we give the descriptor as input.From Brute Force matcher we get the matched points between two images.Then we give this matched points to RANSAC.

2.3.1. RANSAC

We use RANSAC(Random Sample Consensus) algorithm to find homography matrix for the best matches we get from BF matcher. A Homography is a transformation that maps the points in one image to the corresponding points in the other image.We find the corresponding transformation matrix called homography matrix.RANSAC algorithm is based on least square curve fitting.It finds the homography matrix for the best fitted feature points in a curve with less error. It is an iterative method to estimate the parameters of a mathematical model from a set of observed data which contains outliers.

The input to RANSAC algorithm is a set of observed data values, a parameterized model which can explain or be fitted to the observations and some confidence parameters, RANSAC achieves its goal by iteratively selecting a random subset of the original data. The basic RANSAC algorithm is as summarized below:

- Select randomly the minimum number of points required to detect model parameters.
- Solve for the parameter of the model.
- Determine how many points from the set of all points fit with a predefined tolerance ϵ.
- If the fraction of the number of inliers over the total number points in the set exceeds a predefined threshold , estimate the model parameters using all the identified inliers and terminate.
- Otherwise, repeat steps 1 through 4.

This is the process involved in RANSAC .

After finding the homography matrix we need to perform image warping to combine both images.

2.4. Image Warping

Image warping is a transformation which maps all positions in one image plane to positions in a second plane. It arises in many image analysis problems, whether in order to remove optical distortions introduced by a camera or a particular viewing perspective. The two images that will form the mosaic are warped, by using the geometric transformation.

Types of warping :

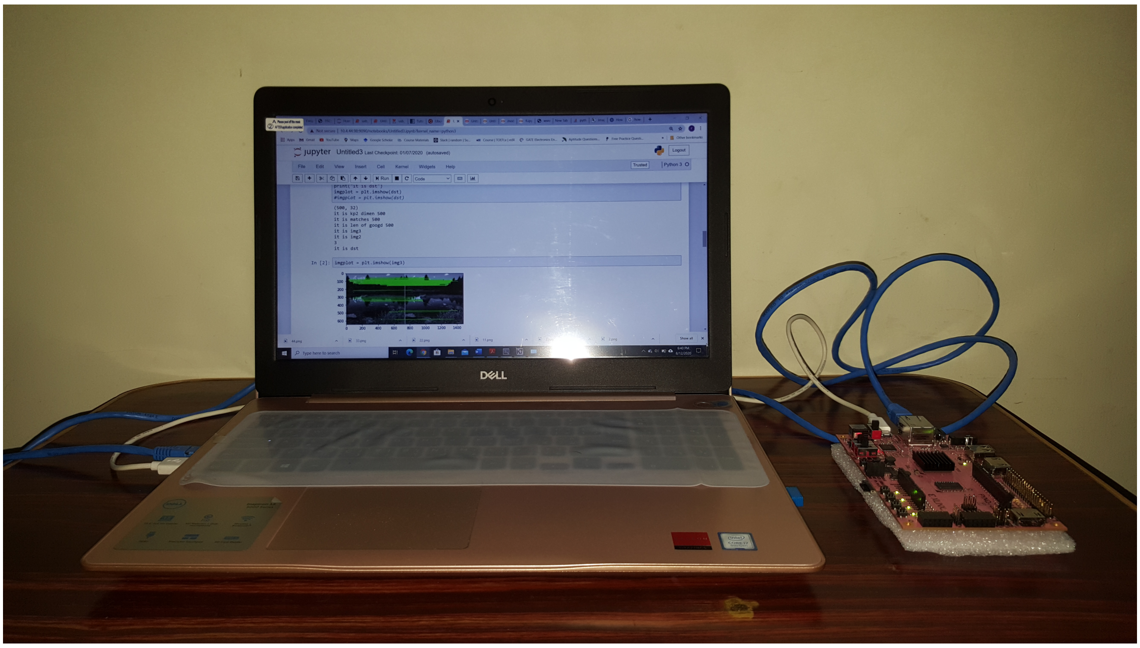
- Planar : where in every image is an element of a plane surface, subject to translation and rotations.

- Cylindrical : wherein every image is represented as if the coordinate system was cylindrical. and the image was plotted on the curved surface of the cylinder.
- Spherical : the above appends, instead of a cylinder, the reference model is a sphere.

Each model has its' own application.Here planar homography and warping are used.We used the homography matrix generated by RANSAC to do the transformation.This image warping combines the both the images based on homography matrix.

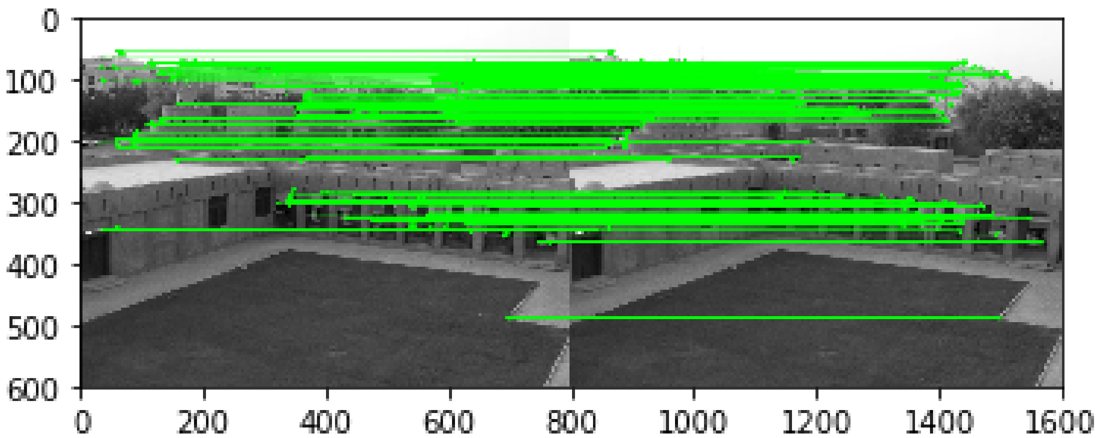
3. Results

For testing our algorithm based on Color Eigen Structures we taken two sets of images one is the fort and other is some scenery. We used the harware(FPGA) **PYNQ-Z2** which is a Python productivity for ZYNQ, it has ZYNQ-7020 SOC on the board.And we taken a PYNQ img of version 2.5 to boot the board. We uploaded these test images on to board and accessed them through code. We implemented the above algorithm using Python programming language and we run the code on PYNQ board. These are the output results for our input images. We showed the image representing the matched keypoints in both the images and the final output stitched image for given two inputs.



Hardware Set Up Using PYNQ-Z2 Board

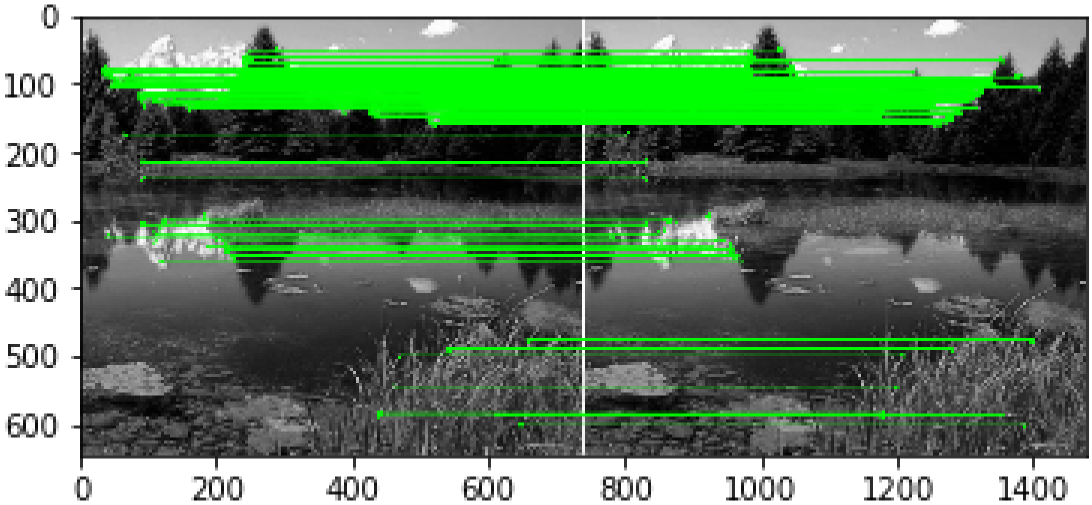
Ex1: Two Images Taken for Stitching



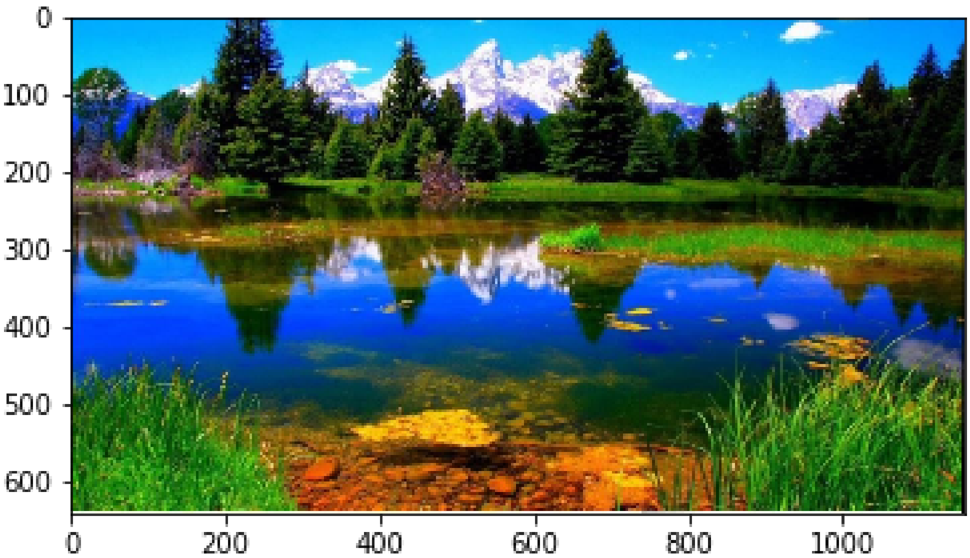
Ex1: Matched the Common Points in Both Images

Ex1: Output of the Image Stitching

Ex2: Two Images Taken for Stitching



Ex2: Matched the Common Points in Both Images



Ex2: Output of The Image Stitching

4. References

https://en.wikipedia.org/wiki/Harris$_C orner_D etector$
$https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_orb/py_orb.html$
$https://iopscience.iop.org/article/10.1088/1742-6596/1237/3/032020/pdf$
$https://www.irjet.net/archives/V3/i4/IRJET-V3I4439.pdf$
$https://en.wikipedia.org/?title=RANSACredirect=no$
$https://en.wikipedia.org/wiki/Image_warping$

Contact Details
O.Sai Lahari
ogireddysailahari@gmail.com
M.no : 9515426799
Dept.Electronics and Communication Engineering,
RGUKT Nuzvid, Krishna District,
AndhraPradesh, Pin : 521202,
India