# CHAPTER 3

## ALGORITHM & IMPLEMENTATION

## 3.1. Algorithm

## The basic function of this program is to detect the defects and classify them accordingly. The Detected defects are then shown in a zoomed in version. Many operations are performed to get the desired outcome. The first operation is the basic image acquisition. In image acquisition a both the required images are acquired from the directory. Images can be either RGB or Grey. We are working on Grey images only so even if any RGB image is selected the function nDims (pic). This function checks the dimension of that image. A RGB will return an output of >=2 and grey image an output of <2. Hence using this function the images are differentiated. The RGB2GREY function then converts that image to Grey before performing further operations. After the images are acquired the next step is to register both the images. Registration is used to register two images one on another or pixel to pixel so that they can be compared pixel to pixel. Any changes that are present are then easy to detect. For registration of two translated images or two rotated images, the images are first compared if there are changes present then the SURF features are detected. This SURF features are nothing but speeded up Robust Features. It is a robust local feature detection function. This SURF features are then extracted for both template PCB image and Faulty PCB image. Using this features inlier and outlier points are showed. Using features from both this points features are then matched. After matching the features a spatial transform function tform is used. The function tform creates a multidimensional spatial transformation structure. Using this function further the angle of rotation or the scale by which image is translated is calculated. The angle of rotation is found by the phase formula and scale is found by the distance formula. After this a geometric transform function is used imwarp. The imwarp transform an image according to the tform value. According to this the image is rotated or translated or both. Hence the image is registered. After registration a threshold value is set to avoid the unwanted error which will be present. After setting threshold we need to detect the errors. Both the template image and faulty PCB images are then complemented. This complement turns any black pixel value to white and any white pixel value to black. By doing this it is easy to perform further functions. Hence image complement is used. Now there are three types of defects which can be found in a PCB. First is under etched. In this defect the PCB has extra copper remaining. Second Defect is over etched. In this defect some extra holes or some buses have been cut. And the third defect is it contains both under etched defect and over etched defect. Now to detect these defects we subtract both the images. For Under etched defect we subtract Faulty from Template image. This is done due to fact that under etched PCB has more number of pixels than over etched PCB. Similarly for Over Etched defect we subtract Template from Faulty PCB since Faulty PCB has more number of pixels than Template PCB. For the case of both, both the cases are performed. Hence we get the error which are present. Now to classify this errors some colors are used. For using this colors we use the function repmat. This function adds colors to those area which remain. Hence we get the error present in color. Now to show where this errors are the faulty PCB is laid over this errors image. Hence the areas which have error present are showed in colour.

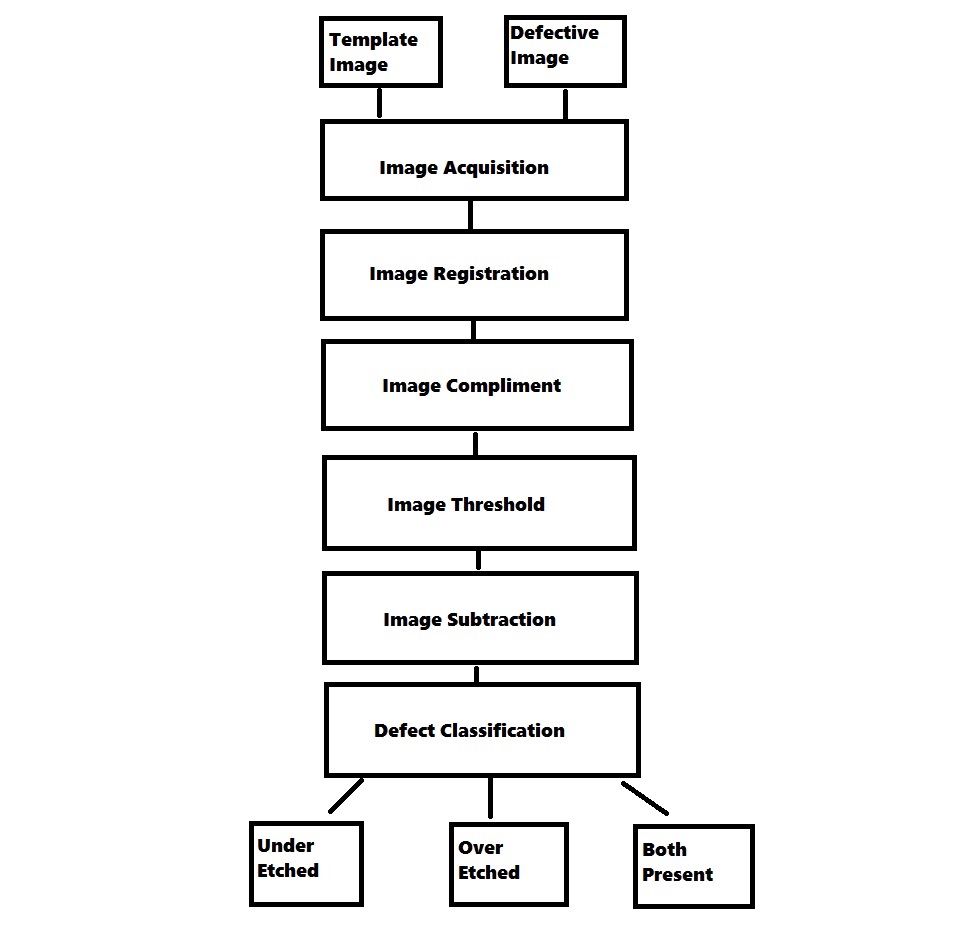


Figure 3. 1 Block Diagram of Algorithm

* 1. **Image Acquisition**

This step means acquiring the image that is to be checked for defects which could be of various image formats such as bitmap (.bmp), portable network graphics (.png), JPEG (.jpg), etc. Another Template/Reference image of a defect free PCB is taken as input which could also be of the same or different image format.

* 1. **Image Registration**

The template and Defective Image have different orientation and size, hence images are registered before the image operations. The pixels of both images are mapped according to their similar features in the image. For a rotated image the angle of rotation is calculated and the image is then re-rotated and resized for registration.

* 1. **Image Complement**

The Image of the PCB which is to be tested is to be complemented for subtraction and defect detection. The complementing of a 3D RGB image has to be done with the help of converting the image to Gray scale and then complementing. For Image complement Each pixel value of the image is subtracted from the highest value of the image i.e. if it is an 8-bit image maximum value is 28-1which equals 255.

* 1. **Image Threshold**

After Selection of the Defects the Images are Threshold according to their size and shape. Image threshold makes the pixel below a Threshold value zero i.e. black and pixel having value above the threshold value white. Hence this operation makes it easier for the appearance of PCB tracks more visible in an image. The image which is to be tested and the template image are given same threshold value to reduce complexity of the program.

* 1. **Image Subtraction**

After Image threshold is done the next step is image subtraction i.e. subtraction of defective image from the template image. The subtraction is done pixel by pixel of the template and the defective image. The Image registration provides easy access for image subtraction as an matrix operation.

* 1. **Pixel Manipulation**

The defects are seen as white spots in the resultant image caused by image subtraction operation. If no spots exist in the image it means that the PCB is not defective. If any spots are seen in the resultant image then defects exists in the image. The pixels having some value are detected and their value is manipulated to a specific color which is RED for UNDER-ETCH and GREEN for OVER-ETCH defects.

* 1. **GUI**

The GUI provides easy access to all the features of PCB defect detection system. The following Images displayed describe the GUI in detail.

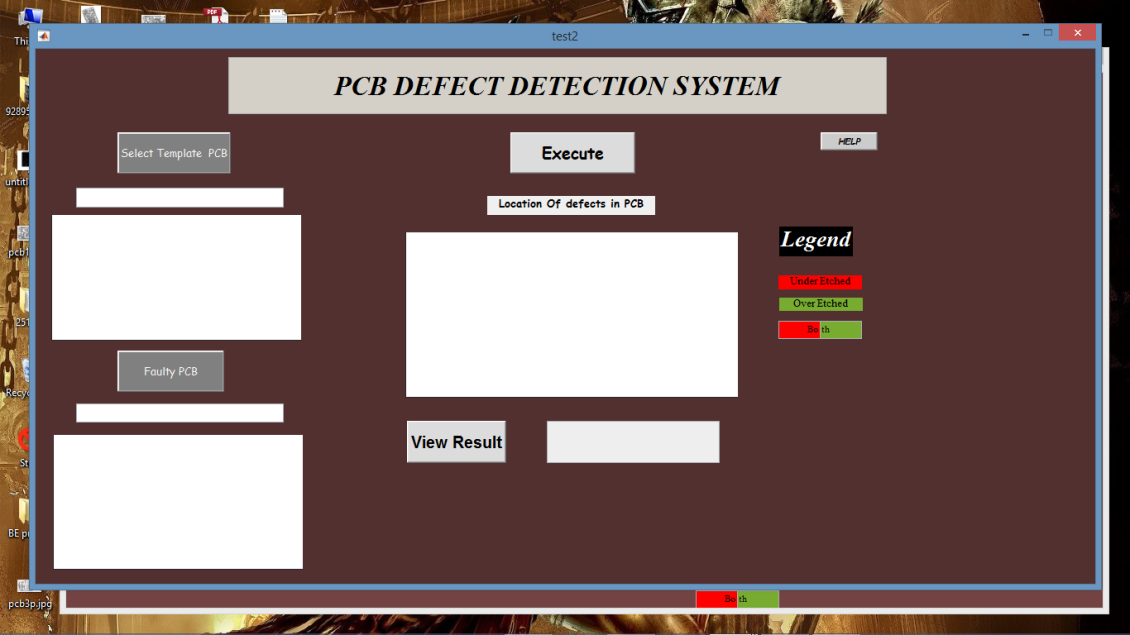


Figure 3. 2 GUI

The GUI consists of the two input images which are displayed on the left hand side of the image one above the other. The result is displayed in the center. Buttons are placed at respective places which help the user select and execute appropriate input and execute the output. A help menu and a Display result button is placed to get help and view the result in a larger size. The Legend placed helps us to know what type of defect is seen in the result i.e. Green for Over-etch and Red for Under-etch.

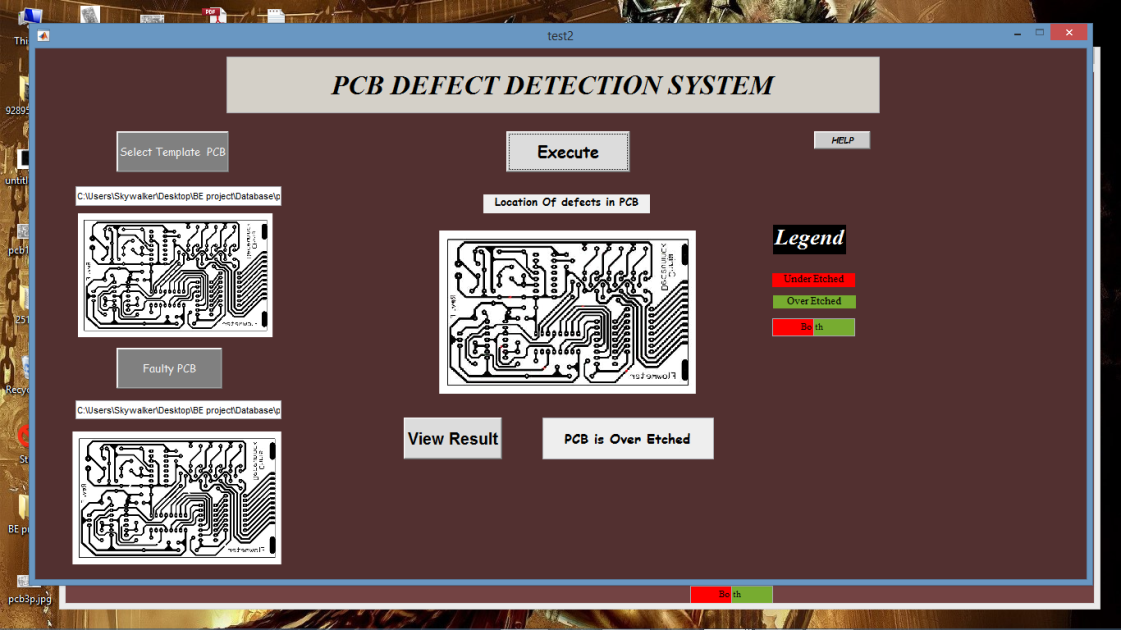


Figure 3. 3 GUI with Images loaded and executed

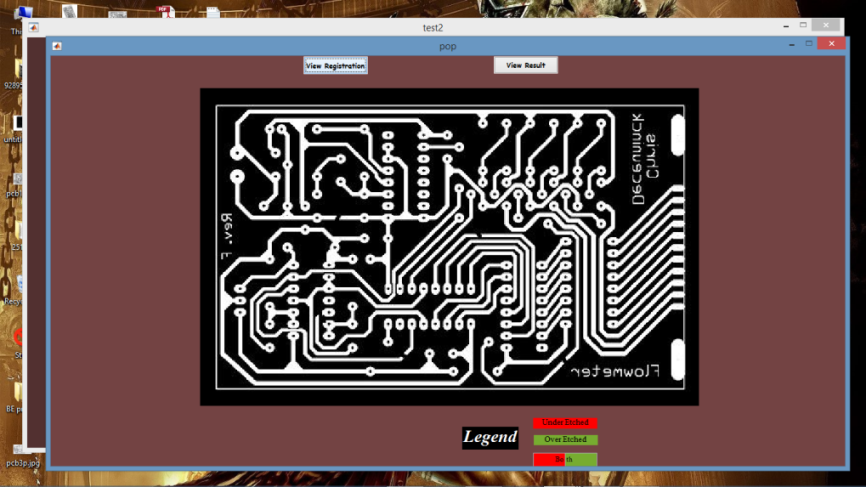


Figure 3. 4 Registration menu

The above menu shows whether the image is registered or there is a defect in the registration and the further operations cannot take place.

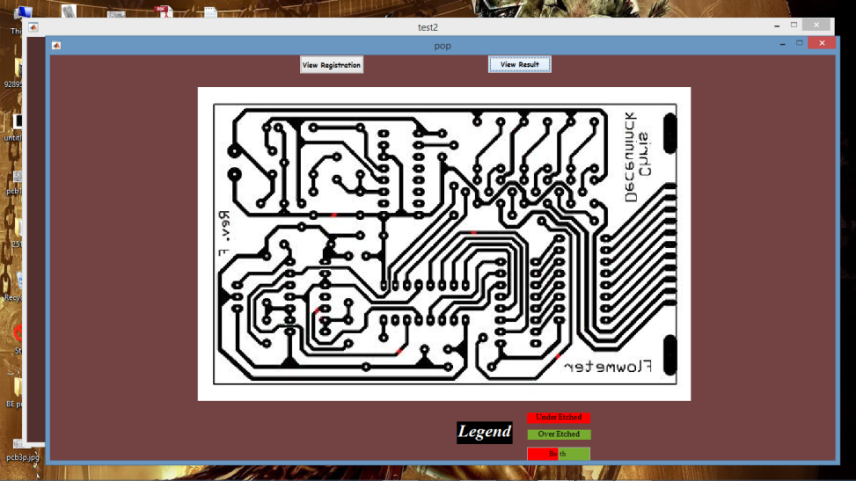


Figure 3. 5 Results Menu

The menu displays Result in a bigger, better and wider view this helps easy detection of error by the user.

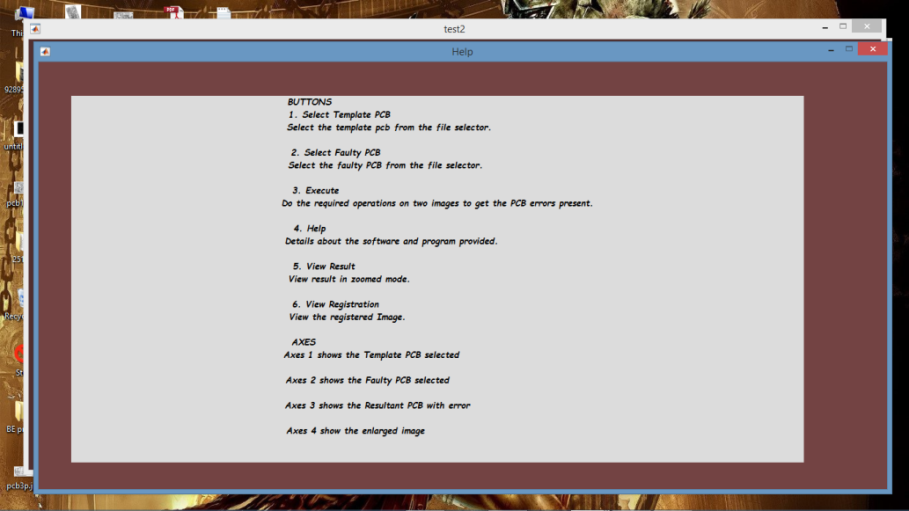


Figure 3. 6 Help menu

The help menu contains all the information about the GUI its buttons images and other things. It just ensures that the user may be able to troubleshoot if there is any problem with the application