

The stages are used to indicate the landmarks automatically on image as follows:

- Load the image to matrix
- Convert the image into Gray Scale mode and calculate the gray histogram of image.
- Apply binary threshold to get the binary image
- Apply the Canny algorithm to detect the edges in the image.
- Extract the edges from image which were detected by Canny algorithm
- Present the edges with the list of approximated lines.
- Construct the pairwise geometric histogram (PGH) of image by using the geometric histogram of list of approximated lines.
- Measure the similar metric between model image and scene image by Bhattacharyya metric
- Determine the presence and location of model image in the scene image by Probabilistic Hough Transform (PHT).
- Verify the estimated landmarks from PHT by using template matching

In next section, we will describe the architecture of program. Follows, we go to deeper in each stage of method.

1 Architecture

The architecture of program is followed 3-tier model. There-tier architecture is an architecture that each tier is designed, developed and maintained as independent. The advantage of this architecture is intended to allow any upgraded or replaced independent between the tiers. When user want to change the requirements or technology of a tier, it will non-affect to other tiers.

The architecture of three-tiers includes:

- **Data tier:** includes the classes which were designed for the data structure of program. It also provides the persistence mechanism to access the data.
- **Logic tier:** controls the functionality of application by performing detailed processing.
- **Presentation tier:** displays information related to user. It is a layer which received the require from user to program or return the result from program to user.

2 Methods

2.1 Loading image

The program using **LibJpeg** to load the image. **LibJpeg** is a open source library with the functions for operating on image under JPEG data format. It had implemented the functions to encode and decode the JPEG image. Besides, LibJpeg also contains the various functions for handling JPEG data. It is written by C language and distributed as opensource.

The image after decompressing is saved into a suitable data structure. Following the design of program, we used the matrix of color space (RGB) to store the value from decompressing step. This matrix will use for the next steps.

2.2 Converting the image into gray scale

The image after decompressing will be converted into 1-channel(gray scale) of color space to calculate the histogram. With 1-channel mode, the image can easily be processed. To convert the image to gray scale mode, we measure the intensity of light at each pixel in a single band (0-255). Having many ways to convert a color image to grayscale image, and in this case to suitable with Luma coding in video systems, the values of grayscale pixel is computed as:

$$Gray_value = 0.299 * R + 0.587 * G + 0.114 * B$$

2.3 Calculating the histogram, threshold value

After converting the RGB image into grayscale, a histogram is calculated to determine the distribution of pixels in the image. The histogram used to calculate the threshold value which was used to convert the grayscale image into binary image. This value is also a importance parameter in Canny algorithm. The process to compute threshold value as follows:

- Calculate the mean and median of grayscale histogram
- Calculate the mean values of peak regions value based on the mean and median of global grayscale histogram
- The threshold value is mean of the peak's mean values.

2.4 Thresholding the image

After that, the grayscale image is converted into binary image by using binary thresholding value.

$$b(x, y) = \begin{cases} 255 & \text{if } Wsrc(x, y) > \text{threshold value} \\ 0 & \text{otherwise} \end{cases}$$

2.5 Canny algorithm

Canny algorithm is varied for determine the edge in image. This is a technique to extract the useful structure of the object in image. It was developed by **John F.Canny** in 1986. The processes of Canny algorithm are described in 4 steps as:

- Apply Gaussian filter to smooth the image in order to reduce the noise
- Find the intensity gradients and direction gradients of the image
- Apply non-maximum suppression to eliminate the weak edges, which are parallel with the strong edges.
- Apply double threshold to determine the edges.

2.5.1 Gaussian filter

To smooth the image, a Gaussian filter was applied to convolve with the image. This step will smooth the image to reduce the effects of the noises on the edge detector. The equation for Gaussian kernel of size $(2k + 1) \times (2k + 1)$ is computed as:

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i-(k+1))^2 + (j-(k+1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k + 1)$$

In this method, the Gaussian filter is combined into the Sobel operation, which used to finding the intensity and direction of gradients.

2.5.2 Finding the intensity and direction of gradients

The Sobel operation is used to detect the edge in image. This operator returns a value for the first derivative in horizontal direction (G_x) and the vertical direction (G_y). From these value, the gradient and direction of edge can be determined as:

$$G = \sqrt{G_x^2 + G_y^2} \text{ and } \phi = \text{atan2}(G_y, G_x)$$

The Sobel kernel used in this case is 3x3. The edge direction angle is rounded to one of four angles which were represented for four directions: vertical, horizontal and two diagonals (90,0,45 and 135 degree).

2.5.3 Non maximum supression

Non-maximum suppression is applied to thin the edge in image. Thus, this operation help suppress all the gradient value to 0 except the local maximal, which indicates changing the location of intensity value. At every pixel, it suppress the gradient value of the center pixel if its magnitude is smaller than the magnitude of one out of two neighbors in the gradient direction. In details:

- If the gradient direction angle is **0** degree, the point will be considered to be on the edge if the gradient magnitude is greater than the magnitude at pixels in the **east** and **west** directions.

- If the gradient direction angle is **45** degree, the point will be considered to be on the edge if the gradient magnitude is greater than the magnitude at pixels in the **north east** and **south west** directions.
- If the gradient direction angle is **90** degree, the point will be considered to be on the edge if the gradient magnitude is greater than the magnitude at pixels in the **north** and **south** directions.
- If the gradient direction angle is **135(-45)** degree, the point will be considered to be on the edge if the gradient magnitude is greater than the magnitude at pixels in the **north east** and **south west** directions.

2.5.4 Double threshold

After applying the non-maximum suppression, the edges pixels are present. However, there are still some edge pixels effected by noise. Double threshold will filter out the edge pixels with the weak gradient value and preserve the edge with the high gradient value.

- A pixel called strong pixel (hence, it belong to the edge), if the edge pixel's gradient value is higher than the high threshold value.
- A pixel will be suppressed, if the edge pixel's gradient value is smaller than the low threshold value.
- A pixel called weak pixel (can be belong to the edge or not), if the edge pixel's gradient value is larger than low threshold value and smaller than high threshold value. A weak pixel can be belong to the edge if it connected with a strong pixel in 8-connected; else, it will be suppressed.

Thus, the accuracy of algorithm is depended on two parameters: the kernel of Gaussian filter and thresholds value. In the content of this implementation, the size of Gaussian kernel is 3x3 and the threshold values was determined by analysis the histogram of image with the ratio 1:3.

2.6 Extract the edge with Suzuki algorithm