A Wavelet Approach to Facial Recognition

Alex Nguyen & Hao Lin

December 11, 2020

1 Introduction

Facial recognition is a common technology used in many applications today. There are many examples in real life that make usage of facial recognition. For example, face filter applications are used in many social networking apps. In order to appropriately apply the filters on the face, it uses facial recognition to locate the face and the features of the face. Similarly, in security systems such as unlocking a smartphone, a facial recognition system is applied to verify if the current user is allowed to access the phone with the list of verified faces stored in the phone.

2 Related Work

In the field of facial recognition, there is an approach that has high performance using a neural network. In particular, the image data set is collected and labeled (either by manually drawing a bounding box or drawing a heat-map). This data set goes through a training process for the machine to learn the kernel (which is mostly high dimensional matrices) that maps from the input to the output data. After that, the model (kernel) is then applied in the test data (new data) to predict (classify) an actual bounding box or facial heat-map.
One way for the machine to learn the kernel is to use convolutional layers in the neural network that have a filter of 2x2 or 3x3, striding through every row of the input image, then for each stop, the filter uses its function to compute all the pixels covered by the filter to compute 1 value. This serves as the function that maps from one space to another. Then after the output is computed, a log-likelihood error is computed given that predicted output and the labeled image. The result is then propagated back to the kernels for the mapping matrices to be changed in value [3].

2.1 Discrete Wavelet Decomposition

Discrete Wavelet Decomposition has been found useful in many signal and image processing applications. The application to an image is performed by applying a low pass and a high pass frequency filter on an image. The convolution

with the low pass filter results in a blurred image by taking the averages of two neighboring pixels. The convolutions with the high pass filter are taking the difference between two neighboring pixels to create three detailed images, horizontal, vertical, and diagonal. The analysis of images is processed with scaling function ϕ and wavelet function ψ and the orientation information is obtained by following combinations of scaling and wavelet function.

$$\begin{aligned}\phi(t) &= \phi(2t) + \phi(2t-1)\\ \psi(t) &= \phi(2t) - \phi(2t-1)\\ \phi_B(x,y) &= \phi(x)\phi(y)\\ \psi_H(x,y) &= \phi(x)\psi(y)\\ \psi_V(x,y) &= \psi(x)\phi(y)\\ \psi_D(x,y) &= \psi(x)\psi(y)\end{aligned}$$

In a typical decomposition, the image is decomposed and the blurred image is split further to achieve the next level of decomposition. But instead, both the blur and the detail image is split to achieve the next level, resulting in 16 images. Typically, the evaluation of the entropy of the image is performed to select the level the image is being processed, but here two-level decomposition is enough to ensure the images contains the relevant information of the face and avoid losing important detail of the facial feature.

2.2 Integral Projection

Many algorithms have been proposed to solve facial feature extraction, which is based on template matching and image normalization. However, facial features may differ and may require several templates to correctly extract each facial feature. It can be very time and space consuming. A useful technique that is proposed here is to use integral projection. For an image I(x,y), x represents the rows and y represents the columns of the image, sum the total pixels for all the rows and columns. H(y) represents the horizontal projection or the sum of every column, and V(x) represents vertical projection or the sum of every row.

$$H(y) = \sum_{x=x_1}^{x_2} I(x,y) \tag{1}$$

$$V(x) = \sum_{y=y_1}^{y_2} I(x,y) \tag{2}$$

The results from the horizontal and vertical projection contribute to locating the position of the face and the facial feature. Using the vertical projection on the blurred image, the border of the face and identify by finding the two local maxima one on each side of the image, representing the broader of the face. The facial features such as eyes, nose, and mouth can be identified by using the horizontal projection on the blurred image. The area around facial features have higher coefficient values compare to the rest of the face, so by locating the

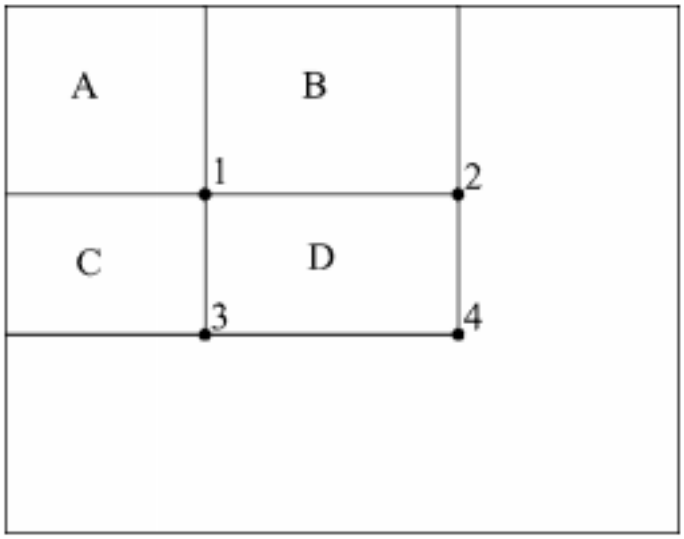
local maxima on the horizontal projection to find the baseline of the feature. In additionally, the process can be further checked by using the vertical projection on the horizontal detail image [1].

2.3 Integral Image

A space-efficient technique in extracting the pixels from a portion of the image is through the integral image. By using the integral image, it reduces the pixel being processed and boosts the computation speed. The integral image at position x, y contains the sum of all the pixels from the upper left of the image inclusively, where

$$ii(x, y) = \sum_{x'<x, y'<y} i(x'y')$$

ii(x,y) is the integral image and i(x,y) is the original image



A section of the feature can be computed through four array references. The sum of the pixel of rectangle D can be subtracted from the sum of pixels at location 2 and 3, which are the rectangle A + B and A + C, since rectangle A is subtracted twice, it is important to add back the sum of the pixel from rectangle A [2].

3 Approach

In order to draw the to draw the bounding box, we decide to use a combination of the use of vertical projection and edge detection. Particularly, to get the vertical bounds of the face, we first perform edge detection on the face, then scan the whole image from left to right and right to left to search for the very first occurrence of the vertical indices where there exist a sharp change in pixel

value. To get the horizontal bounds of the face, we perform vertical projection and set a threshold to spaced out the baselines on the face. After that, we get the very first and last horizontal baselines. After obtaining two horizontal and two vertical bounds, we consider the cross-section of the four lines the bounding box of the face [1].

3.1 Sobel Filters

There are many related work that concerns edge detection, however, in order to efficiently detect the edge of the face without noise in the constant-pixel area, we have decided to use a high-level convolution filter (sobel) that act similarly to the undecimated wavelet transformation.

-1	0	1
-2	0	2
-1	0	1

Horizontal

-1	-2	-1
0	0	0
-1	-2	-1

Vertical

The sobel filter consists of two convolutional filters, one for detecting the change in the horizontal direction, h_x, and one for detecting the change for the vertical direction, h_y. We define the horizontal change of the image, G_x, and the vertical change of the image, G_y as the convolutional product of each corresponding filter with the image, I:

$$G_x = h_x * I \tag{3}$$

$$G_y = h_y * I \tag{4}$$

We define the result edge image G as below:

$$G = \sqrt{G_x^2 + G_y^2} \tag{5}$$

The sobel filter detects edge by computing the gradient in each pixel with the perspective of the surrounding area, so the area with no small change will have the gradient close to 0, while the area with more changes in pixel value (area where there happen to be face-edge) will have large gradient.

3.2 Facial Features Localization

After the image transformation, we can draw bounding lines by localizing the face and facial features using the sobel image. We draw the horizontal baseline to locate the face with respect to the entire image. Since there is a significant difference in gradient between the background to the face, it allows easy edge detection for finding the bounding box. The left border can be located by searching through for a sudden leap in pixel from the left to right. Similarly, the right border can be located by doing the same process by starting from the right. Next is locating the features of the facial. Facial features have the highest pixel value compare to the rest of the face. So by using the vertical projection, we find the six highest local maxima. Since some of the coefficient values around the facial features are the highest, we set a threshing floor of 20 to avoid clustering.

3.3 Feature Vector Extraction

We perform the discrete wavelet decomposition propose before. The first level returns four images and the second level returns a total of 16 images. We ignore the first image, which is the image that went through two low pass transformation, and only use the other 15 images. Additionally, we locate the face on the image and get the top half of the face and bottom of the face by using the bounding box we set up from the original image. In total there should be 17 images, we calculate the mean and variances for each of the images. With the means and variances of these vector features, we calculate the Bhttacharrya distance between two images denoted as $D(v_k, v_l)$. D_i is the distances between the component pairs i of the two feature vectors v_k and v_l [1].

$$\mathscr{D}_i(v_k, v_l) = \frac{1}{4}\frac{(\mu_{ik} - \mu_{il})^2}{(\sigma_{ik}^2 + \sigma_{il}^2)} + \frac{1}{2}\ln\left[\frac{1/2(\sigma_{ik}^2 + \sigma_{il}^2)}{\sqrt{\sigma_{ik}^2\sigma_{il}^2}}\right]$$

The Bhttacharrya distance is statistically measuring the probability distribution of two vector feature. We process all 17 feature vectors and sum the total distance to evaluate if the two images is corresponding to the same face an. We set the threshold of 0.1, where any value greater than 0.1 means the two images does not represent the same person.

4 Conclusion

Given the data set with n data instance and p number of accurate predictions, we compute the accuracy A by the equation below:

$$A = \frac{p * 100\%}{n} \tag{6}$$

We tested a total of 100 random different pairs of images and resulted in an accuracy of 87%. We are even able to correctly match a person with their glasses on and off. However, we sometimes run into difficulty, when there is a shadow or any other disturbances on the image which affect our statistics measurement to correctly evaluate two images. As an extension of this work, it would be interesting to implement would be directly extracting all the individual features from the face for statistical analysis. While we located the general location of the features, we do no have the bounding box of each specific feature. Since integral projection creates a clustering around one feature, it will be helpful to include the integral image to determine which portion within the bounding box has the highest intensity in respect to the size of that portion. Overall, our approach can be further improved by including other data science techniques that can improve our accuracy.

References

[1] C. Garcia, G. Zikos, and G. Tziritas. Wavelet packet analysis for face recognition. *Image and Vision Computing*, 18(4):289 – 297, 2000.

[2] MICHAEL J. JONES and PAUL VIOLA. Robust real-time face detection. *International Journal of Computer Vision*, pages 137 – 154, 2003.

[3] S. Khan, M. H. Javed, E. Ahmed, S. A. A. Shah, and S. U. Ali. Facial recognition using convolutional neural networks and implementation on smart glasses. In *2019 International Conference on Information Science and Communication Technology (ICISCT)*, pages 1–6, 2019.