Genetic Based LBP Feature Extraction and Selection for Facial Recognition

Joseph Shelton, Gerry Dozier, Kelvin Bryant, Joshua Adams, Khary Popplewell,
Tamirat Abegaz, Kamilah Purrington, Damon L. Woodard, Karl Ricanek,

Abstract— This paper presents a novel approach to LBP feature extraction. Unlike other LBP feature extraction methods, we evolve the number, position, and the size of the areas of feature extraction. The approach described in this paper also attempts to minimize the number of areas as well as the size in an effort to reduce the total number of features needed for LBP-based face recognition. In addition to reducing the number of features by 63%, our approach also increases recognition accuracy from an average of 99.04% to 99.84%.

Keywords— Local Binary Pattern (LBP), Steady State Genetic Algorithm (SSGA), Feature Extraction, Manhattan Distance.

I. INTRODUCTION

Local Binary Pattern (LBP) feature extraction is a method proposed by Ojala [1] that has been used successfully in a number of applications [2,3]. In [2], Porebski et al. used LBP in an effort for classifying tree trunks while Sun et al. [3] used LBP for iris feature extraction.

The standard way of using LBP-based feature extraction is to evenly distribute patches across an image, so that the whole image is covered. Each patch is of uniform size, and no patches overlap [10]. Although this method has proven to be reliable, no researchers, as of yet, have developed LBP-based feature extraction methods that differ from the standard approach. The standard method generally uses the entire space of an image to perform calculations on, which this paper proves is unnecessary. In this paper, we develop a method based on genetic and evolutionary computing (GEC) [7,8,9,11,12] that evolves LBP extractors that have unevenly distributed, overlapping patches that may not cover the entire image.

GEC is a general problem solving technique based on simulated evolution [7,8,9,11,12]. A typical GEC creates a population of individuals, and then breeds these individuals to create a number of children. The children are incorporated into the population by replacing a number of weaker individuals. The evolutionary process of selecting parents, creating offspring, and incorporating the offspring into the population is repeated until a user-specified stopping condition has been reached.

The remainder of this paper is organized as follows. In Section II, we will introduce standard LBP-based feature extraction as well as introduce our genetic-based method, in Section III, we present our experiments, in Section IV, we present our results, and in Section V, we present our conclusions and future work.

II. STANDARD AND GENETIC-BASED LBP FEATURE EXTRACTION

LBP can be introduced as follows. For the LBP method, typically a gray-scale image (as shown in Figure 1a) of a subject is initially segmented into a number of uniform, evenly distributed patches that cover the entire image as shown in Figure 1.

Figure 1a: Subject image          Figure 1b: Segmented

LBP is then applied to each pixel of a patch resulting in a histogram representing the feature characteristics for that particular patch. A feature vector is created by simply concatenating all of the histograms associated with each patch.

Figure 2 shows an example of how an LBP is calculated for one pixel within a patch. In Figure 2, consider a pixel surrounded by eight neighbors. The matrix represents the normal pixel values in an image. The values in the pattern matrix are obtained after using Equation 1. The weights are the worth of each value in the pattern matrix. To extract the LBP for the center pixel, C, the difference between it and each of its neighbor pixels, W, are calculated.
\[
T = \{(W_0 - C), \ldots, (W_{m-1} - C)\}
\] (1)

If the difference is zero or greater, then a value of ‘1’ is associated with the difference. If the difference is negative, then a ‘0’ is associated with the difference (as shown in Equation 2). The sequence of differences forms what we refer to as a pattern matrix.

\[
S = \begin{cases} 
0 & \text{if } T(i) < 0; \\
1 & \text{if } T(i) \geq 0; 
\end{cases}
\] (2)

After the pattern matrix has been created, the binary number associated with the pattern matrix is computed starting with the top right neighbor and calculating clockwise. The values in the resulting matrix are summed together to get the LBP.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Pattern</th>
<th>Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 7 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 6 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 5 1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Figure 2: Computing LBP](image)

Since there are a total of 8 neighbors there exists a total of 256 possible patterns that can be extracted, ranging from 0 to 255. A LBP bit sequence corresponding to a pattern matrix is said to be uniform if the sequence (with wrap around) changes values less than two times. For example, the bit sequence, 11111111, has zero changes, the bit sequence, 11100011, has two changes: between the fourth and fifth bit, and between the seventh and eighth bit. The bit sequence, 11100010, has more than two changes and would not be considered uniform.

As stated earlier, each patch has an associated histogram. Each histogram has 59 components referred to as bins: there are 56 bins that correspond to the 56 uniform bit sequences that have exactly two changes, 2 bins that correspond to the two bit sequences that have no change, namely, 00000000 and 11111111. These two sequences are assigned to Bin 0 and Bin 58 respectively. Finally, all non-uniform sequences are assigned to Bin 59. Thus, the histogram associated with a patch is a count of the number of sequences (of the pixels within a patch) that are associated with one of the 59 bins.

A feature vector or template is a concatenation of all the histograms corresponding to the patches on an image. For example, if there are 24 patches and each patch has 59 bins, the total size of the corresponding feature vector would be 1416. In the feature vector, the features corresponding to the first patch would start at 1 and go to 59. Similarly, the features of the second patch would begin at 60, while the 59 features associated with the 24th patch would begin at 1358.

Recognition is performed by comparing a captured probe template, \( p \), with all the vectors in a gallery set \( H = \{ h_0, h_1, \ldots, h_{q-1} \} \) using the Manhattan distance metric (City Block). The subject, \( h_s \), from the gallery set that is closest to \( p \) is considered to be its match.

The GEC used in this paper is a Steady State Genetic Algorithm (SSGA) [7,8]. This algorithm was chosen because of its simplistic nature. The SSGA is used to evolve a population of candidate feature extractors (FEs). A candidate FE, \( f_{\text{fe}} \), is a 6-tuple, \( \langle X_i, Y_i, W_i, H_i, M_i, f_{\text{fe}} \rangle \), where \( X_i = \{ x_{i,0}, x_{i,1}, \ldots, x_{i,n-1} \} \) represents the x-coordinates of the upper leftmost pixel of the \( n \) possible patches, \( Y_i = \{ y_{i,0}, y_{i,1}, \ldots, y_{i,n-1} \} \) represents the y-coordinates of the upper leftmost pixel of the possible patches, \( W_i = \{ w_{i,0}, w_{i,1}, \ldots, w_{i,p} \} \) represents the widths of the \( n \) possible patches measured in pixels, \( H_i = \{ h_{i,0}, h_{i,1}, \ldots, h_{i,n-1} \} \) represents the heights of the \( n \) possible patches, \( M_i = \{ m_{i,0}, m_{i,1}, \ldots, m_{i,p} \} \) represents the mask for each patch (1 means to extract features from the corresponding patch, 0 means do not extract features from the corresponding patch), and \( f_{\text{fe}} \) represents the fitness of \( f_{\text{fe}} \).

Given a probe set and a gallery set, recognition is performed by comparing a captured probe template, \( p \), with all the vectors in a gallery set \( H = \{ h_0, h_1, \ldots, h_{q-1} \} \) using the Manhattan distance metric (City Block). The subject, \( h_s \), from the gallery set that is closest to \( p \) is considered to be its match.

The SSGA used to evolve candidate FEs works as follows. First a population of candidate FEs is randomly generated. Each candidate FE is then evaluated and assigned a fitness. After the initial population has been created, two parents are selected via binary tournament selection [8,12] and create one offspring, and allowing the offspring to replace the worst fit candidate FE in the population. The evolutionary process of selecting parents, allowing them to create an offspring, and allowing the offspring to replace the worst fit FE in the population is repeated a user specified number of times (see Figure 3).

![Figure 3: Psuedo-code for SSGA](image)

### III. EXPERIMENTS

The dataset used in our experiment was composed of 105 subjects taken from the FRGC dataset [13]. Each subject had a total of three images taken that varied slightly. Our experiment used a probe set, which contained one image of a subject, and a gallery set, which contained the other two
images of a subject. There were a total of 105 images in the probe set and 210 images in the gallery set. All images had a width of 100 pixels and a height of 127 pixels.

For LBP feature extraction, only the interior pixels within a patch were processed. The position of a patch was corrected if it exceeded the boundaries of an image by shifting the patch towards the center of the image. The distance that a patch was shifted was the amount that the patch surpassed the image dimensions.

For this experiment, we compared the standard Local Binary Pattern Method (SLBPM) with the Genetic-Based LBP method. The SLBPM used a set of 24 patches (four rows by six columns) that were uniform, non-overlapping, and covered the entire image.

IV. RESULTS

For our results, a SSGA was used to evolve a population of 20 candidate FEs using uniform crossover and Gaussian mutation, (where the Gaussian mu $\sigma = 0.1$). The SSGA was run 30 times. For each run, a total of 1000 function evaluations were allowed.

The standard Local Binary Pattern Method only needed to be run once since the patch characteristics were deterministic. In Table I, the average performance of the SSGA is compared to the SLBPM. The SLBPM used all 100% of the patches, and had an accuracy of 99.04%. The SSGA used an average of 36.90% of patches, with an average accuracy of 99.84%.

A t-Test comparing SLBPM and SSGA was performed. The results of the t-test were statistically significant. The best individual out of all 30 runs of the SSGA had a 100% recognition rate and used only seven patches. Figure 4 shows a representation of the patch positions and sizes of the best feature extractor out of the 30 runs.

The results suggest that evolving patches is superior to placing uniform, non-overlapping patches over an entire image. The positioning of the patches for the best individual evolved by the SSGA method also suggests that the periocular, nasal, and eyebrow regions are perhaps the better distinguishing areas between subjects.

As seen in Figure 4, most of the patches for the best individual are centered on the nose and eyebrow regions. The patches in the bottom right corner seem to serve no contribution, but a plausible theory is that too much weighting of patches around the ocular region proved to be problematic for certain individuals in the dataset.

IV. RESULTS

For our results, a SSGA was used to evolve a population of 20 candidate FEs using uniform crossover and Gaussian mutation, (where the Gaussian mu $\sigma = 0.1$). The SSGA was run 30 times. For each run, a total of 1000 function evaluations were allowed.

The standard Local Binary Pattern Method only needed to be run once since the patch characteristics were deterministic. In Table I, the average performance of the SSGA is compared to the SLBPM. The SLBPM used all 100% of the patches, and had an accuracy of 99.04%. The SSGA used an average of 36.90% of patches, with an average accuracy of 99.84%.

A t-Test comparing SLBPM and SSGA was performed. The results of the t-test were statistically significant. The best individual out of all 30 runs of the SSGA had a 100% recognition rate and used only seven patches. Figure 4 shows a representation of the patch positions and sizes of the best feature extractor out of the 30 runs.

The results suggest that evolving patches is superior to placing uniform, non-overlapping patches over an entire image. The positioning of the patches for the best individual evolved by the SSGA method also suggests that the periocular, nasal, and eyebrow regions are perhaps the better distinguishing areas between subjects.

As seen in Figure 4, most of the patches for the best individual are centered on the nose and eyebrow regions. The patches in the bottom right corner seem to serve no contribution, but a plausible theory is that too much weighting of patches around the ocular region proved to be problematic for certain individuals in the dataset.

V. CONCLUSION AND FUTURE WORK

In this paper, a genetic-based LBP feature extractor was presented that successfully minimize the number of patches needed for LBP-based facial recognition, while increasing recognition accuracy. An extractor was created that has a 100% recognition accuracy. Our future work will be devoted towards the investigation of the performance of other forms of GEC for the purpose of genetic and evolutionary-based feature extraction.

ACKNOWLEDGMENT

This research was funded by the office of the Director of National Intelligence (ODNI), Center for Academic Excellence (CAE) for the multi-university, Center for Advanced Studies in Identity Sciences (CASIS) and by the National Science Foundation (NSF), Science & Technology Center: Bio/Computational Evolution in Action Consortium (BEACON). The authors would like to thank the ODNI and the NSF for their support of this research.
TABLE I
EXPERIMENTAL RESULTS FOR LBP (EVEN DISTRIBUTION) AND SSGA METHODS

<table>
<thead>
<tr>
<th>Methods</th>
<th>Patches Used</th>
<th>Average Accuracy</th>
<th>Best Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLEPM</td>
<td>100%</td>
<td>99.04%</td>
<td>99.04%</td>
</tr>
<tr>
<td>SSGA(0.01)</td>
<td>36.90%</td>
<td>99.84%</td>
<td>100.0%</td>
</tr>
</tbody>
</table>

REFERENCES


[3] Zhenan Sun, Tieniu Tan, Xianchao Qiu Graph Matching Iris Image Blocks with Local Binary Pattern; National Laboratory of Pattern Recognition, 2006


