Exploring Visual Features Through Gabor Representations for Facial Expression Detection

Sien W. Chew\textsuperscript{1}, Patrick Lucey\textsuperscript{2}, Sridha Sridharan\textsuperscript{1}, Clinton Fookes\textsuperscript{1}

\textsuperscript{1}Image and Video Research Laboratory, Queensland University of Technology, GPO Box 2424, Brisbane 4001, Australia
\textsuperscript{2}Robotics Institute, Carnegie Mellon University/Department of Psychology, University of Pittsburgh, Pittsburgh, USA
s4.chew@student.qut.edu.au, patlucey@andrew.cmu.edu, \{s.sridharan;c.fookes\}@qut.edu.au

Abstract

Gabor representations have been widely used in facial analysis (face recognition, face detection and facial expression detection) due to their biological relevance and computational properties. Two popular Gabor representations used in literature are: 1) Log-Gabor and 2) Gabor energy filters. Even though these representations are somewhat similar, they also have distinct differences as the Log-Gabor filters mimic the simple cells in the visual cortex while the Gabor energy filters emulate the complex cells, which causes subtle differences in the responses. In this paper, we analyze the difference between these two Gabor representations and quantify these differences on the task of facial action unit (AU) detection. In our experiments conducted on the Cohn-Kanade dataset, we report an average area under the ROC curve (A’) of 92.60% across 17 AUs for the Gabor energy filters, while the Log-Gabor representation achieved an average A’ of 96.11%. This result suggests that small spatial differences that the Log-Gabor filters pick up on are more useful for AU detection than the differences in contours and edges that the Gabor energy filters extract.

Index Terms: action unit detection, AdaBoost feature selection, Log-Gabor filter, Gabor energy filter

1. Introduction

Facial expressions provide cues about emotion and the cognitive processes associated with them [1, 2]. The decomposition of visible muscular movements in the face using the Facial Action Coding System (FACS) had been the de-facto standard in automatic facial expression recognition [3]. An entire hour of coding can translate to just one minute of ground-truthed video [4], hence providing the motivation for the automatic coding of action units (AUs). However, automatic facial expression recognition is still deemed a difficult task plagued by similar difficulties experienced in the pattern recognition and face processing communities.

Gabor filters have been frequently applied to facial expression recognition. The superior performance of these filters can be attributed to two positive characteristics: i) its excellent representation of texture, and ii) its invariance to changing illumination conditions. Several instances have been reported on Gabor filters producing good performance [5, 6]. Zhang et al. [7] applied multilayer perceptrons (MLP) on a Gabor filterbank, composed of 3 spatial frequencies and 8 orientations, to recognise the seven facial expressions that are shared universally among people [8] (neutral, happiness, sadness, surprise, anger, disgust and fear). Reduction of the large dimensionality of Gabor features was accomplished through principal component analysis. An accuracy of 80% was achieved overall, but this improved to 85.6% after data containing fear expressions were omitted.

The work of Bashyal and Venayagamoorthy [9] closely paralleled that of Zhang et al. Both procedures were almost identical; with the only exception being that their system used learning vector quantization (LVQ) instead of MLP. An accuracy of 90.22% was achieved on the Japanese Female Facial Expression dataset [10]. The interesting finding surfacing from these two studies showed that the MLP learning algorithm had difficulty classifying “fear” emotions, but this problem could be circumvented by LVQ. Littlewort et al. [11] were able to achieve a 93% emotion classification accuracy on the Cohn-Kanade dataset using a system based on Gabor filters and support vector machines. Whitehill et al. [12] utilized this same system for the recognition of smiles. A Gabor energy filterbank (5 spatial frequencies, 8 orientations) was applied to distinguish between smiling and non-smiling faces, which managed to achieve an accuracy of 98%.

Across these various works described above, there has been little attention paid to the various types of Gabor representations that can be used. Two of these are: 1) Log-Gabor and 2) Gabor energy filters. Even though these representations are somewhat similar, they also have distinct differences as the Log-Gabor filters mimic the simple cells in the visual cortex while the Gabor energy filters emulate the complex cells, which causes subtle differences in the responses. In this paper, we analyze the difference between these two Gabor representations on the task of facial action unit (AU) detection.

2. Gabor Representations: Log-Gabor vs Gabor Energy

The primates’ primary visual cortex system (commonly abbreviated to V1) have been frequently modeled using Gabor filters. Neurons within V1, comprised of simple and complex cells [13], administer visual processing tasks by means of edge and line detection. In an ideal sense, Gabor wavelets provide a sufficient representation of these cells [14, 15, 16]. Simple cells are sensitive to an image’s orientation as well as its position, whereas complex cells are sensitive only to its orientation. The transfer function of a two-dimensional Gabor filter consists of a Gaussian envelope modulated by a sine carrier,
In 1987, Field [17] proposed that the coding of natural images could be improved through the modification of the Gaussian envelope’s transfer function. The suggestion was to design the Gabor filter so that the shape of the envelope was retained on the logarithmic frequency scale instead of on the linear frequency scale.

2.1. Log-Gabor Features

In 1987, Field [17] proposed that the coding of natural images could be improved through the modification of the Gaussian envelope’s transfer function. The suggestion was to design the Gabor filter so that the shape of the envelope was retained on the logarithmic frequency scale instead of on the linear frequency scale.

$$g(x, y) = K \exp \left\{ -\pi \left[ a^2(x - x_0)^2 + b^2(y - y_0)^2 \right] \right\} \exp \left\{ j2\pi(\mu_0 x + \nu_0 y) + P \right\}$$

Figure 1: Computation of Log-Gabor features by convolution of the original image with a spatial Log-Gabor filter

where $g(x, y)$ denotes a two-dimensional Gabor filter. The first exponential term in Equation 1 defines the two-dimensional Gaussian envelope, and the second exponential term defines the sinusoid carrier. Constant $K$ determines the scale of the Gaussian envelope; the elongation factor of the envelope is manipulated by variables $a$ and $b$. The subscript $r$ stands for a rotation operation of the envelope. Horizontal and vertical spatial frequencies of the sinusoid carrier are determined by $\mu_0$ and $\nu_0$; and $P$ denotes the phase of the sinusoid carrier. Upon closer inspection of Equation 1, filtering operations through a Gabor filter may be perceived to be synonymous to that of a spatial bandpass filter. In view of the fact that a single Gabor filter is only capable of representing a single cell in V1, multiple Gabor filters implemented as filterbanks have been common practice. In order to permit the detection of bars from multiple orientations and positions, the implementation of Gabor filterbanks becomes imperative.

2.2. Gabor Energy Features

Energy mechanisms operate by summing the squares of the outputs of a quadrature pair [15]. The quadrature counterpart of a linear operator is equivalent to its Hilbert transform. Hence, taking the square-root of the sum of squares between a Gabor feature and a 90-degree phase-shifted version of itself would produce the corresponding Gabor energy feature. The complex outputs of a quadrature pair [15]. The quadrature counterpart of a linear operator is equivalent to its Hilbert transform. Hence, taking the square-root of the sum of squares between a Gabor feature and a 90-degree phase-shifted version of itself would produce the corresponding Gabor energy feature. The complex output of the input Gabor feature and its quadrature through Equation 3.

$$e(x, y) = \sqrt{g_0^2(x, y) + g_r^2(x, y)}$$

Figure 2: Formation of Gabor energy features from Gabor features and its quadrature through Equation 3. a) Gabor features, b) Quadrature Gabor features, c) Gabor energy features

The advantage of Gabor energy filters over Gabor filters is that the former is able to give a smooth response to an edge or line [19, 20]. This phenomenon can be observed in Figure 2 where the resultant Gabor energy features are observed to be sharper around edges (cheek contours and nose contours) as compared to either the Gabor features or from its quadrature.

3. AdaBoost for Feature Selection

Large representations of features inevitably result from the application of large filterbanks, often necessitating the use of feature selection. This is important as the redundancy of insignificant features would prolong training times of the classifier otherwise, and yet offer little performance benefits. In this study, the feature selection capability of the AdaBoost algorithm was adopted.

Boosting [24] is a type of committee method commonly applied to improve the performance of an individual classifier. A multitude of weak learners (also known as weak classifiers) are combined to reweighted versions of the data at each iteration. A common realization of weak learners is via classification and
For each of the 64 filter outputs, the 60 most discriminant features were selected by AdaBoost. This gave a total of 60 × 8 × 8 = 3840 discriminant features for each image. Image dimensions were 80 × 100 pixels, which produced a total of 80 × 100 × 8 × 8 = 512,000 features per image. Hence, only 60/512,000 = 0.0118% of the feature space was utilised as discriminant features. No significant differences in performance was observed when the number of discriminant features selected per image was increased ten-fold from 60 to 600.

During initialisation, each data point is assigned a weighting parameter $w_0$ with a value of $\frac{1}{N}$. At each iteration, a new classifier is trained on the data set using weights adjusted according to the performance of the previously trained classifier. A data point classified incorrectly would have its corresponding weight increased. Hence, this would mean that the weights of the most discriminant features would converge in the least number of iterations. In our experiments, we found ten iterations to be sufficient for the task of feature selection.

Visual inspection of the most discriminant features (between these two images) permitted further insight into the different mechanisms by which these two filters operate. AU 6 and 12 are used here for illustration (Figures 5 and 6). Blue markers indicate the top 60 most discriminant features selected by AdaBoost.

The “cheek-raiser” action unit (AU6) is illustrated in Figure 5. The most discriminant Gabor energy features were located in the region near the left-cheek. This region corresponded to an area where distinct changes in texture and curvature occurred. In the case of Log-Gabor features, however, the locations of the most discriminant features were scattered around the mouth area. This region does not correspond to the same landmarks as the Gabor energy features.

This prompted the hypothesis that simple and complex cells had indeed possessed different mechanisms of operation. Gabor energy features were well-adept at representing macro structural changes as well as variations in texture. On the other hand, regression trees [21]. Improvements in classification can be obtained at the final stage after a weighted voting scheme is applied on the weak learners.

AdaBoost [22] is one of the most widely used form of boosting. In a binary classification problem comprising $N$ data points, a set of training data $x_1, \ldots, x_N$ is prepared to correspond to a set of binary target variables $t_1, \ldots, t_N$ holding only either the value of +1 or -1. For example, $t_1$ assigned a value of +1 would label $x_1$ as a positive instance; and a value of -1 would label $x_1$ as a negative instance.

During initialisation, each data point is assigned a weighting parameter $w_0$ with a value of $\frac{1}{N}$. At each iteration, a new classifier is trained on the data set using weights adjusted according to the performance of the previously trained classifier. A data point classified incorrectly would have its corresponding weight increased. Hence, this would mean that the weights of the most discriminant features would converge in the least number of iterations. In our experiments, we found ten iterations to be sufficient for the task of feature selection.

In Figure 3, $|\alpha_t|$ represents the absolute value of the weight assigned by AdaBoost to a feature. The figure on the left-hand side of Figure 3 illustrates an ideal case — observe that the absolute value of the weights $|\alpha_t|$ assigned to the more discriminative feature (denoted by the black curve) converges to zero quicker than the less discriminative feature (denoted by the red curve). Shown on the right-hand side of Figure 3 is a practical example where convergence of $|\alpha_t|$ to zero is not so obvious. However, the features’ discriminative power can still be determined by calculating the numerical gradients of the two curves, which may be represented compactly through the sum of the numerical gradients. Hence, a smaller sum would indicate that a feature is more discriminative than a feature which outputs a larger sum.

The “GML AdaBoost Matlab Toolbox” [23] was modified to perform feature selection using the Gentle AdaBoost algorithm [24]. For each of the 64 filter outputs, the 60 most discriminant features were selected by AdaBoost. This gave a total of 60 × 8 × 8 = 3840 discriminant features for each image. Image dimensions were 80 × 100 pixels, which produced a total of 80 × 100 × 8 × 8 = 512,000 features per image. Hence, only 60/512,000 × 100 = 0.0118% of the feature space was utilised as discriminant features. No significant differences in performance was observed when the number of discriminant features selected per image was increased ten-fold from 60 to 600.

Figure 3: Feature selection - determining how discriminative a feature is. (a): ideal case clearly showing the black curve (more discriminative feature) converges to zero quicker than the red curve (less discriminative feature). (b): practical example where convergence is not so obvious, but still discernable through the curves’ numerical gradients.

Figure 4: Discriminant Features Computed by AdaBoost. Left: Fear Expression, Right: Neutral Expression, the most discriminant features are highlighted with blue markers.
Log-Gabor features were particularly good at distinguishing micro structural changes. On this premise, we propose in Section 5 that action units involving large facial structures are best represented using Gabor energy filters, and action units involving smaller facial structures are best represented using Log-Gabor filters.

4. Experimental Setup

In situations where data is scarce, the $S$-fold cross validation technique is suitable for training classifier models. A special case of $S$-fold cross validation is the leave-one-out strategy, where the number of folds equals the total number of data points. This strategy was employed in our experiments to train 17 selected action unit models on 97 subjects from the Cohn-Kanade [25] dataset. The action units selected were - AU 1 2 4 5 6 7 9 11 12 15 17 20 23 24 25 26 27. Support vector machines (LibSVM [26] used) with a linear kernel had been used for training. Images were rescaled to a size of $100 \times 80$ pixels. The subjects in this dataset ranged from 18 to 30 years of age, 65% were female, 15% African-American and 3% Asian or Latino. Instructions were given to the subjects to perform various facial displays which comprised of both single action units as well as combinations of several action units. The six emotions — joy, surprise, anger, fear, disgust, and sadness — were a part of these displays. All action units displayed by the subjects had been annotated by certified FACS coders.

Further details on the leave-one-out configuration implemented in this study ought to be addressed — if AU1 was to be trained on subject S010, then the frames containing the peak AU intensity from all other subjects in the dataset that contained AU1 would be used as positive instances. The negative instances were selected from frames that contained the neutral expression of the corresponding positive sequences. The ratio between positive instances to negative instances was held at 1 : 1. This procedure was repeated for every other subject in the dataset, and for each of the 17 selected action units selected. A total of $97 \times 17 = 1649$ action unit models were trained.

In the testing stage, all frames belonging to a particular subject were used to evaluate the action unit models. To test the AU1 model of subject S010, all peak frames containing AU1 were chosen to be positive instances, and all neutral frames and peak frames containing other AUs were chosen for negative instances.

5. Results and Discussion

We report the performance of our action unit detection system using a metric derived from receiver-operating characteristic (ROC) curves. ROC curves are generated by plotting the number of true-positives against the number of false-positives, as a decision threshold is varied. The area under the ROC curve (denoted $A'$) is a reliable performance measure which we used. This area indicates the probability that a positive instance selected at random will be ranked higher than a negative instance also selected at random. An upper-bound on the uncertainty of $A'$ was included to approximate the reliability of performance. A statistic commonly used for this purpose is $s = \sqrt{\frac{A'(1-A')}{n_p \cdot n_n}}$, where $n_p$, $n_n$ are the number of positive and negative examples [27]. The system obtained a mean $A'$ of 0.9504, 0.9611 and 0.9260 respectively for pixels-only, Log-Gabor and Gabor energy features.

Results shown in Table 1 indicate that the performances obtained from using all three features were very similar (note that small test sample sizes would effect a diminished impact on percentage differences). The Gabor features did not offer significant advantages over pixels-only features due to 2 reasons — i) good texture properties and little variation in illumination were already present in the images, ii) the Gabor features were subjected to high reduction in dimensionality.

An interesting finding from this study was that Log-Gabor and Gabor energy filters employed contrasting modes of operation. From the most discriminant features selected in Figures 5 and 6, we observed that Gabor energy filters detected...
Table 1: Results showing the area under the ROC curve for pixels-only, Log-Gabor and Gabor energy features. \( N \) indicates the number of positive examples available.

<table>
<thead>
<tr>
<th>AU</th>
<th>( N )</th>
<th>Pixels-only</th>
<th>Log-Gabor</th>
<th>Gabor Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>142</td>
<td>0.93±0.02</td>
<td>0.96±0.02</td>
<td>0.91±0.02</td>
</tr>
<tr>
<td>2</td>
<td>103</td>
<td>0.93±0.03</td>
<td>0.94±0.02</td>
<td>0.93±0.03</td>
</tr>
<tr>
<td>4</td>
<td>148</td>
<td>0.92±0.02</td>
<td>0.92±0.02</td>
<td>0.87±0.03</td>
</tr>
<tr>
<td>5</td>
<td>76</td>
<td>0.97±0.02</td>
<td>0.98±0.02</td>
<td>0.94±0.03</td>
</tr>
<tr>
<td>6</td>
<td>106</td>
<td>0.99±0.01</td>
<td>0.99±0.01</td>
<td>0.99±0.01</td>
</tr>
<tr>
<td>7</td>
<td>105</td>
<td>0.94±0.02</td>
<td>0.95±0.02</td>
<td>0.90±0.03</td>
</tr>
<tr>
<td>9</td>
<td>49</td>
<td>0.98±0.02</td>
<td>0.99±0.01</td>
<td>0.95±0.03</td>
</tr>
<tr>
<td>11</td>
<td>33</td>
<td>0.97±0.03</td>
<td>0.96±0.03</td>
<td>0.94±0.04</td>
</tr>
<tr>
<td>12</td>
<td>91</td>
<td>0.99±0.01</td>
<td>0.99±0.01</td>
<td>0.99±0.01</td>
</tr>
<tr>
<td>15</td>
<td>71</td>
<td>0.94±0.03</td>
<td>0.96±0.02</td>
<td>0.90±0.04</td>
</tr>
<tr>
<td>17</td>
<td>147</td>
<td>0.91±0.02</td>
<td>0.94±0.02</td>
<td>0.89±0.03</td>
</tr>
<tr>
<td>20</td>
<td>67</td>
<td>0.96±0.02</td>
<td>0.95±0.03</td>
<td>0.92±0.03</td>
</tr>
<tr>
<td>23</td>
<td>41</td>
<td>0.95±0.03</td>
<td>0.96±0.03</td>
<td>0.91±0.04</td>
</tr>
<tr>
<td>24</td>
<td>41</td>
<td>0.95±0.03</td>
<td>0.98±0.02</td>
<td>0.92±0.04</td>
</tr>
<tr>
<td>25</td>
<td>285</td>
<td>0.93±0.02</td>
<td>0.97±0.01</td>
<td>0.91±0.02</td>
</tr>
<tr>
<td>26</td>
<td>36</td>
<td>0.89±0.05</td>
<td>0.91±0.05</td>
<td>0.89±0.05</td>
</tr>
<tr>
<td>27</td>
<td>75</td>
<td>1.00±0.00</td>
<td>0.99±0.01</td>
<td>1.00±0.00</td>
</tr>
</tbody>
</table>

Mean Variance – 0.95±0.02 9.3 0.96±0.02 6.39 0.92±0.03 13.78

texture and contour variations. On the other hand, Log-Gabor filters detected small spatial differences. The high resolution of the images enabled finer spatial details of the images to be represented by Log-Gabor filters. As the availability of high-resolution images grows with the advent of modern video and image acquisition equipment, the utilisation of Log-Gabor filters could contribute positively to facial analysis. The experiments conducted in this study showed that a reduction of the variance encountered classification accuracies on these three representations for the detection of action units in facial expressions. The roles that simple and complex cells of the visual cortex system play in the detection of action units were examined. We chose Log-Gabor filters for the representation of simple cells, and Gabor energy filters for the representation of complex cells.

An interesting outcome from this study was that simple and complex cells had different modes of operation. The simple cells identified small spatial differences, while complex cells identified differences in contours and edges. An alternate interpretation could be that simple cells operate by searching an image for micro-structures, and complex cells operate by searching for macro-structures. Another finding was that Log-Gabor filters proved well-suited for the analysis of high-resolution images. No significant deterioration in performance was observed even after a 99.25% reduction in feature space, thus making these filters a fitting candidate for real-time implementations.

The mean accuracy obtained by the system on the Cohn-Kanade posed facial expression dataset was 96.11% and 92.60% respectively for Log-Gabor and Gabor energy filters. Our results suggest that simple cells are slightly more adept than complex cells at detecting action units under optimal conditions (i.e. limited image noise). Performance of the Gabor energy filters was inferior to that of pixels-only features. This phenomenon was the result of the curse of dimensionality, in which a larger representation in feature space deteriorates performance instead of bringing about improvement. Another reason given was that the images in the dataset were already of high-quality, leaving little room for improvement in terms of texture representation.

An interesting outcome from this study was that simple and complex cells had different modes of operation. The simple cells identified small spatial differences, while complex cells identified differences in contours and edges. An alternate interpretation could be that simple cells operate by searching an image for micro-structures, and complex cells operate by searching for macro-structures. Another finding was that Log-Gabor filters proved well-suited for the analysis of high-resolution images. No significant deterioration in performance was observed even after a 99.25% reduction in feature space, thus making these filters a fitting candidate for real-time applications.

The diverse functionalities of biological cells in the visual cortex system cannot be contained within this single study. Log-Gabor and Gabor energy filters serve, at best, as an approximation to these cells. Images acquired under more realistic conditions that have been contaminated by varying illumination and pose conditions, or the analysis of spontaneous facial expressions, or utilizing the information embodied in the time evolution of action units; all serve as interesting avenues for further research. Significant effort had been put in by the research community toward understanding these two types of filters. However, there is still much left unknown about these cells in image understanding. The benefit of pursuing a deeper comprehension about the behaviour of these cells could create a hybrid filter drawing on the benefits of both simple and complex cells, and again enabling nature as the conduit for the continual improvement in machine vision.

6. Conclusion
7. Acknowledgments

This research was supported by the Cooperative Research Centre for Advanced Automotive Technology (AutoCRC).

8. References


