AN IMPROVEMENT TO THE NEAREST NEIGHBOR CLASSIFIER AND FACE RECOGNITION EXPERIMENTS

YONG XU¹, QI ZHU¹, YAN CHEN¹ AND JENG-SHYANG PAN²

¹Bio-Computing Research Center
²Innovative Information Industry Research Center
Harbin Institute of Technology Shenzhen Graduate School
HIT Campus of Shenzhen University Town, Xili, Shenzhen 518055, P. R. China
laterfall2@yahoo.com.cn; ksqiqi@sina.com; jadechenyan@gmail.com; jspan@cc.kuas.edu.tw

Received November 2011; revised March 2012

Abstract. The conventional nearest neighbor classifier (NNC) directly exploits the distances between the test sample and training samples to perform classification. NNC independently evaluates the distance between the test sample and a training sample. In this paper, we propose to use the classification procedure of sparse representation to improve NNC. The proposed method has the following basic idea: the training samples are not uncorrelated and the “distance” between the test sample and a training sample should not be independently calculated and should take into account the relationship between different training samples. The proposed method first uses a linear combination of all the training samples to represent the test sample and then exploits modified “distance” to classify the test sample. The method obtains the coefficients of the linear combination by solving a linear system. The method then calculates the distance between the test sample and the result of multiplying each training sample by the corresponding coefficient and assumes that the test sample is from the same class as the training sample that has the minimum distance. The method elaborately modifies NNC and considers the relationship between different training samples, so it is able to produce a higher classification accuracy. A large number of face recognition experiments on three face image databases show that the maximum difference between the accuracies of the proposed method and NNC is greater than 10%.

Keywords: Face recognition, Nearest neighbor classifier, Sparse representation, Classification

1. Introduction. The image recognition technique [1-3] can be used for a variety of applications such as objection recognition, personal identification and facial expression recognition [4-14]. For many years researchers in the field of image recognition have adopted the following procedures to perform recognition: image capture, feature selection or feature extraction and classification. Usually these procedures are consecutively implemented and each process is necessary. The nearest neighbor classifier (NNC) is an important classifier. NNC is also one of the oldest and simplest classifiers [15,17]. The nearest neighbors of the sample were used in a number of fields such as image retrieval, image coding, motion control and face recognition [19,22]. NNC first identifies the training sample that is the closest to the test sample and assumes that the test sample is from the same class as this training sample. Since “close” means “similarity”, we can also say that NNC exploits the “similarity” of the test sample and each training sample to perform classification. To determine the nearest neighbors of the sample is the first step of NNC, so it is very crucial. In the past, various ideas and algorithms were proposed for determining the nearest neighbors. For example, D. Omercevic et al. proposed the idea of meaningful nearest neighbors [23]. H. Samet proposed the MaxNearestDist algorithm for
finding K nearest neighbors [24]. J. Toyama et al. proposed a probably correct approach for greatly reducing the searching time of the nearest neighbor search method [25]. The focus of this approach is to devise the correct set of k-nearest neighbors obtained in high probability. Y.-S. Chen et al. proposed a fast and versatile algorithm to rapidly perform nearest neighbor searches [26]. Besides the methods in these works, many other methods [26-30] have also been developed for computationally efficiently searching the nearest neighbors. We note that most of these methods focus on improving the computation efficiency of the nearest neighbor search.

We note that recently a distinctive image recognition method, the sparse representation (SR) method was proposed [31]. The applications of SR on image recognition such as face recognition have obtained a promising performance [32-34]. However, it seems that it is not very clear why SR can outperform most of previous face recognition methods and different researchers attribute the good performance of SR to different factors. In our opinion one of remarkable advantages of SR is that it uses a novel procedure to classify the test sample. Actually, this method first represents a test sample by using a linear combination of a subset of the training samples. Then it takes the weighted sum of the training sample as an approximation to the test sample and regards the coefficients of the linear combination as the weights. SR calculates the deviation of the test sample from the weighted sum of all the training samples from the same class and classifies the test sample into the class with the minimum deviation. As the weighted sum of a class is also the sum of the contribution in representing the test sample of this class, we refer to this classification procedure as representation-contribution-based classification procedure (RCBCP). We also say that SR consists of a representation procedure and a classification procedure.

The main rationale of RCBCP is that when determining the distances between the test sample and training samples, it takes into account the relationship of different training samples. If some training samples are collinear, RCBCP will use the weights to reflect the collinear nature and will classify the test sample into the class the weighted sum of which provides the best approximation to the test sample. However, the conventional NNC usually separately evaluates the distances between the test sample and training samples, ignoring the similarity and potential relationship between different training samples. The following example is very helpful to illustrate this difference between RCBCP and the conventional NNC: if two training samples have the same minimum Euclidean distances to the test sample, then NNC will be confused in classifying the test sample. However, under the same condition, RCBCP usually obtains two different “distances” and is still able to determine which training sample is closer to the test sample.

In this paper, motivated by RCBCP, we propose to exploit RCBCP to modify NNC. The basic idea is to use a dependent way to determine the “distances” between the test sample and training samples. We first use all of the training samples to represent the test sample, which leads to a linear system. We directly solve this system to obtain the least-squares solution and then exploit the solution and the classification procedure of NNC to classify the test sample. Differing from the conventional NNC, the proposed method calculates the distance between the test sample and the result of multiplying each training sample by the corresponding weight (i.e., a component of the solution vector) and assumes that the test sample is from the same class as the training sample with the minimum distance. The proposed method is very simple and computationally efficient. Our experiments show that the proposed method always achieves a lower rate of classification errors than NNC. This paper also shows that one modification of the proposed method is identical to NNC.

This paper not only proposes an improvement to NNC but also has the following contributions: it confirms that RCBCP is very useful for achieving a good face recognition
AN IMPROVEMENT TO THE NNC AND FACE RECOGNITION EXPERIMENTS

performance. Moreover, it also somewhat illustrates that RCBCP is one of the most important advantages of SR.

The rest of the paper is organized as follows. Section 2 describes our method. Section 3 shows the difference between NNC and the proposed method. Section 4 presents the experimental results. Section 5 offers our conclusion.

2. Problem Statement and Preliminaries. Let \( A_1, \ldots, A_n \) denote all \( n \) training samples in the form of column vectors. We assume that in the original space test sample \( Y \) can be approximately represented by a linear combination of all of the training samples. That is,

\[
Y \approx \sum_{i=1}^{n} \beta_i A_i. \tag{1}
\]

\( \beta_i \) is the coefficient of the linear combination. We can rewrite (1) as

\[
Y = A \beta, \tag{2}
\]

where \( \beta = (\beta_1, \ldots, \beta_n)^T, A = (A_1, \ldots, A_n). \)

As we know, if \( A^T A \) is not singular, we can obtain the least squares solution of (2) using \( \beta = (A^T A)^{-1} A^T Y \). If \( A^T A \) is nearly singular, we can solve \( \beta \) by \( \beta = (A^T A + \mu I)^{-1} A^T Y \), where \( \mu \) is a positive constant and \( I \) is the identity. After we obtain \( \beta \), we calculate \( Y' = A \beta \) and refer to it as the result of the linear combination of all of the training samples.

From (1), we know that every training sample makes its own contribution to representing the test sample. The contribution that the \( i \)th training sample makes is \( \beta_i A_i \). Moreover, the ability, of representing the test sample, of the \( i \)th training sample \( A_i \) can be evaluated by the deviation between \( \beta_i A_i \) and \( Y' \), i.e., \( e_i = \|Y - \beta_i A_i\|^2 \). Deviation \( e_i \) can be also viewed as a measurement of the distance between the test sample and the \( i \)th training sample. We consider that the smaller \( e_i \) is, the greater ability of representing the test sample the \( i \)th training sample has. We identify the training sample that has the minimum deviation from the test sample and classify \( Y \) into the same class as this training sample.

3. Analysis of Our Method. In this section, we show the characteristics and rationale of our method.

3.1. Difference between our method and NNC. Superficially, our method performs somewhat similarly with NNC, because both of them first evaluate the “distances” between the test sample and each training sample and classify the test sample into the same class as the training sample that has the minimum “distance”. However, our method is different from NNC as follows: it does not directly compute the distance between the test sample and each training sample but calculates the distance between the test sample and the result of multiplying each training sample by the corresponding coefficient. Since the sum of all the training samples weighted by the corresponding coefficients well approximates to the original test sample, the result of multiplying each training sample by the corresponding coefficient can be viewed as an optimal approximation, to the original test sample, generated from the training sample. Thus, the deviation between this approximation and the original test sample can be taken as the “distance” between the training sample and test sample. Intuitively, the smaller the “distance”, the more “similar” to the test sample the training sample.

As the weighted sum (i.e., a linear combination) of all the training samples well represents the test sample, we say that all the training samples provide a good representation for the test sample in a competitive way. According to the classification procedure of our
Table 4. Rates of the classification errors (%) of our method and NNC on the AR database. We took the first 2, 4, 6 and 8 training samples per class and the others as training samples and test samples, respectively.

<table>
<thead>
<tr>
<th>Database</th>
<th>Training samples</th>
<th>Our method</th>
<th>NNC</th>
<th>CBNNC</th>
<th>NNL</th>
<th>Difference between our method and NNC</th>
<th>Difference between our method and CBNNC</th>
<th>Difference between our method and NNL</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>2 per class</td>
<td>30.38</td>
<td>39.93</td>
<td>40.03</td>
<td>40.80</td>
<td>9.55</td>
<td>9.65</td>
<td>10.42</td>
</tr>
<tr>
<td>AR</td>
<td>4 per class</td>
<td>31.55</td>
<td>42.69</td>
<td>42.69</td>
<td>42.50</td>
<td>11.14</td>
<td>11.14</td>
<td>10.95</td>
</tr>
<tr>
<td>AR</td>
<td>6 per class</td>
<td>30.92</td>
<td>38.88</td>
<td>38.92</td>
<td>38.25</td>
<td>7.96</td>
<td>8.0</td>
<td>7.33</td>
</tr>
<tr>
<td>AR</td>
<td>8 per class</td>
<td>33.89</td>
<td>41.76</td>
<td>41.76</td>
<td>41.34</td>
<td>7.87</td>
<td>7.87</td>
<td>7.45</td>
</tr>
</tbody>
</table>

When computing the distance between the test sample and each training sample, the proposed method not only exploits these two samples but also takes into account the relationship between different training samples. As a result, the proposed method can identify the training sample that has the greatest contribution in representing the test sample. A large number of face recognition experiments show that our method always achieves a higher classification accuracy than NNC and the maximum difference between the accuracies of our method and NNC is greater than 10%.

Acknowledgment. This article is partly supported by Program for New Century Excellent Talents in University (Nos. NCET-08-0156 and NCET-08-0155), NSFC under Grant nos. 61071179, 61173086, 61020106004, 61001037 and 61173086 as well as the Fundamental Research Funds for the Central Universities (HIT.NSRIF. 2009130).

REFERENCES


