Pattern Recognition Letters 33 (2012) 1689–1694

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

# Component-based global *k*-NN classifier for small sample size problems

Nan Zhang *, Jian Yang, Jian-jun Qian

*School of Computer Science and Technology, Nanjing University of Science & Technology, Nanjing 210094, China*

ARTICLE INFO

*Article history:*
Received 7 May 2011
Available online 7 June 2012

Communicated by S.Sarkar

*Keywords:*
*k*-NN classifier
LRC
SRC
Tikhonov regularization
Pattern classification

ABSTRACT

The classical *k*-NN classifier has been widely used in pattern recognition. However, it does not take into account the structural information of local samples. This paper presents a novel classifier named component-based global *k*-NN classifier (CG-*k*-NN), which takes advantage of the structural information of the local neighbors for enhancing the classification performance. We choose *k* nearest neighbors of a given testing sample globally at first, and then use these neighbors to represent the testing sample via ridge regression. In the further step, we construct the component image of each class by using the intra-class images from the *k* nearest neighbors and the corresponding representation coefficients. Finally, the testing sample is assigned to the class that minimizes reconstruction residual. The proposed method CG-*k*-NN is evaluated using the ORL, FERET, AR face image database and PolyU palmprint databases. The experiment results demonstrate that our method is more efficient and effective than the state-of-the-art methods such as sparse representation based classifier (SRC) and linear regression based classifier (LRC).



## 1. Introduction

The nearest neighbor (NN) classifier is widely-used in pattern classification due to its simplicity and effectiveness. Cover and Hart have shown that in large sample cases, the error rate of the NN classifier is bounded above by twice the Bayes error rate (Cover and Hart, 1967). The method was subsequently generalized to be *k* nearest neighbor classifier (Fukunaga, 1990). But the *k*-NN classifier decides the class label of the testing sample just based on the maximum number of training samples that belongs to the same class, without considering the structure of the data.

Rather than looking at the *k*-NN from all training samples, some recent works focus on the local neighborhood within classes. Li and Lu (Li and Lu, 1999) proposed the nearest feature line (NFL) method to generalize the representational capacity of the available limited prototypes. In a feature space, the NFL method uses a feature line to interpolate and extrapolate each pair of prototype feature points belonging to the same class. The feature line virtually provides an infinite number of prototype feature points of the class. The representational capacity of the prototypes is thus expanded. Chien and Wu (Chien and Wu, 2002) further extended Li and Lu's work on NFL and proposed the nearest feature plane (NFP) and the nearest feature space (NFS) methods for pattern classification.

Since the NFL method is conducted for each pair of available prototypes, it faces the large computation complexity problem when there are many prototypes in each class. The NFP method also faces the similar problem due to its increased computational requirement. The nearest neighbor line (NNL) and the nearest neighbor plane (NNP) (Zheng et al., 2004) methods were suggested to alleviate the computation complexity of the NFL and NFP methods. Because only one feature line or a feature plane in each class need to be computed, The NNL and NNP methods are computationally more efficient than the NFL and NFP methods. Recently, the hit-distance based nearest neighbor classifiers was proposed to enhance the generalization power of the NNL and NNP methods (Lou and Jin, 2006). The connection between the tangent distance (Simard et al., 2001) and the NFL is discussed in (He et al., 2008).

Recently, Linear Regression Classifier (LRC) was presented for face classification (Naseem et al., 2010). The method formulates the identification task as a problem of linear regression. LRC is actually a special case of local subspace based classifier. It operates respectively by least-squares estimation using all samples of each class. In addition, the sparse representation-based classifier (SRC), which is presented by Wright et al., has been successfully applied to real-world face recognition problems (Wright et al., 2009a, 2009b). The basic idea of SRC is to represent a given testing sample as a sparse linear combination of all training samples. Both of the LRC and SRC yield high recognition accuracies of face identification in the presence of disguise. However, testing sample may be represented by training samples that are far away from it via the representation of SRC and LRC. So, they will lead to error classification

* Corresponding author. Tel.: +86 13813991892.
*E-mail address:* 554961959@qq.com (N. Zhang).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.05.020

when the supporting samples have different class label with testing sample.

To address the problems of $k$-NN, LRC and SRC, we present a new method called component-based global $k$-NN classifier (CG-$k$-NN). Our method using the $k$ nearest neighbors of the testing sample from all training samples to represent it. In this way, we can avoid the case that the testing sample is represented by the training samples that are far away from it, which might be encountered in LRC and SRC. In addition, since we only use a small number of training samples to represent the testing sample, the scale of the involved ridge regression problem is small. So it is computationally efficiency to calculate the representation coefficients. In contrast, SRC uses entire training samples to represent a testing sample and needs to solve a $L_1$ optimization problem, so it is generally time-consuming when the number of training samples is large. In addition, compared with the $k$-NN classifier which determines the class label of testing sample only based on the number of samples belonged to each class, the proposed CG-$k$-NN takes into account the structural information of samples by using the class reconstruction residual as a decision measure. The evaluation intentionally tested that the new approach is suited to a variety of scenarios and data, and is more effective and computationally efficient than $k$-NN, LRC and SRC in most cases.

## 2. Outline of LRC and SRC

### 2.1. Linear regression classifier

Suppose there are $c$ known pattern classes. Let $A_i$ be the matrix formed by the training samples of the $i$th class, i.e., $A_i = [\mathbf{x}_{i1}, \mathbf{x}_{i2}, \cdots, \mathbf{x}_{iM_i}] \in R^{N\times M_i}$, where $M_i$ is the number of training samples of the $i$th class. Let us define a matrix $A = [A_1, A_2, \ldots, A_c] \in R^{N\times M}$, where $M = \sum_{i=1}^{c} M_i$.

In LRC, the linear equation system $y = Aw$ is solved in each class. For a given testing sample $y$, we use the training samples of the $i$th class to represent it. The least-square estimation of the representation weights is given by Hastie et al. (2001), Seber (2003), Ryan (1997):

$$\hat{w}_i = (A_i^T A_i)^{-1} A_i^T y, \tag{1}$$

We can reconstruct $y$ based on $\hat{w}_i$ and the training samples of the $i$th class, i.e. $\hat{y}_i = A_i \hat{w}_i$. The distance between y and the $i$th class is defined as $d_i = ||y - \hat{y}_i||_2$. The label of the testing sample is decided by the class with the minimum distance $d_i$ .

### 2.2. Sparse representation-based classifier

The sparsest solution to $y = Aw$ can be sought by solving following $L_1$ optimization problem (Donoho, 2006; Candès et al., 2006; Candès and Tao, 2006):

$$(L_1) \quad \hat{w} = \arg\min ||w||_1, \text{subject to } y = Aw. \tag{2}$$

This problem can be solved in polynomial time by standard linear programming algorithms (Chen et al., 2001).

After obtaining the sparsest solution $\hat{w}$, we can design a sparse representation based classifier (SRC) in terms of the class reconstruction residual. Specifically, for each $ith$ class, let $\delta_i: R^N \to R^N$ be the characteristic function that selects the coefficients associated with the $ith$ class. For $w \in R^N$, $\delta_i(w)$ is a vector whose only nonzero entries are the entries in $w$ that are associated with the $ith$ class. Using only the coefficients associated with the $i$th class, one can reconstruct a given testing sample $y$ as $\hat{y}_i = A\delta_i(w)$. The corresponding class reconstruction residual is defined by

$$r_i(y) = ||y - \hat{y}_i||_2 = ||y - A\delta_i(w)||_2 \tag{3}$$

The SRC decision rule is: If $r_l(y) = \min_i r_i(y)$, $y$ is assigned to the $l$th class.

## 3. Component-based global $k$-NN classifier

This section presents a novel classifier coined component-based global $k$-NN classifier (CG-$k$-NN), which takes advantage of the structural information of the local neighbors for enhancing the classification performance of the $k$-NN classifier. CG-$k$-NN uses the $k$ nearest neighbors to represent the testing sample, and calculates the representation weights via Tikhonov regularization. CG-$k$-NN can successfully avoid the error that LRC encounters and reduce the computational complexity of SRC.

### 3.1. Tikhonov regularization of linear regression

Tikhonov regularization was initially presented by Andrey Tychonoff and applied to ill-posed problems (Tikhonov and Arsenin, 1977). It is also known in the statistical literature as ridge regression (Hoerl and Kennard, 1970).

As we know, the least square solution of the ordinary linear regression can be given by:

$$\hat{w} = (A^T A)^{-1} A^T y. \tag{4}$$

Obviously, the inverse of the matrix $A^T A$ in (4) must exist. However, the covariance matrix is sometimes on the verge of singularity or ill-conditioned. Tikhonov Regularization can overcome the singularity by appending the regularization term. The solution of Tikhonov regularization is gained by minimizing:

$$||Aw - y||^2 + ||\Gamma w||^2, \tag{5}$$

where $\Gamma$ is the Tikhonov matrix. The Tikhonov matrix is generally chosen as $\Gamma = \alpha I$, where I is the identity matrix, and $\alpha$ is the Tikhonov factor, a parameter to be chosen.

The covariance matrix $A^T A$ in Tikhonov regularization turns into $A^T A + \Gamma^T \Gamma$, for some suitably chosen Tikhonov matrix $\Gamma$. An explicit solution of Eq. (5) is given by:

$$\hat{w} = (A^T A + \Gamma^T \Gamma)^{-1} A^T y. \tag{6}$$

### 3.2. Component-based global k-NN classifier

The steps of our method are described as follows. First, find its $k$ nearest neighbors of a given testing sample from all training samples. Then, represent the testing sample with these $k$ nearest neighbors. The representation coefficients can be obtained by solving a Tikhonov Regularization problem. Subsequently, we can construct the component image of each class by using the intra-class images from the $k$ nearest neighbors and the corresponding representation coefficients. Finally, compute the residual between the testing sample and the component images of those classes. We can make a decision in favor of the class with the minimum residual.

Specifically, given a testing sample $y$, let us represent it by its $k$ nearest neighbors. Suppose there are $k_i$ nearest neighbors $\tilde{x}_{i1}, \tilde{x}_{i2}, \cdots, \tilde{x}_{ik_i}$ belonging to the $ith$ class, i.e., where $k = \sum_{i=1}^{c} k_i$. Let $\widetilde{A}_i = [\tilde{x}_{i1}, \tilde{x}_{i2}, \cdots, \tilde{x}_{ik_i}] \in R^{N\times k_i}$ and $\widetilde{A}$ be the matrix composed of all $k$ nearest neighbors. Further, the representation coefficients of a given testing sample $y$ are learned by these $k$ nearest neighbors. The representation coefficients can be calculated by solving (6). For each $ith$ class, let $\widetilde{W}_i = [\widetilde{W}_{i,t_1}, \widetilde{W}_{i,t_2}, \cdots, \widetilde{W}_{i,t_{k_i}}] \in R^{k_i}$, $where t_s \in (1, 2, \cdots, M_i), s = 1, 2, \cdots, k_i$ is the coefficients and only associated with the $ith$ class. The component image $\hat{y}_i$ of Class $i$ for the given testing sample $y$ can be calculated as following:

$$\hat{y}_i = \widetilde{A}\widetilde{W}_i. \tag{7}$$