Title:
Fast linear discriminant analysis using binary bases

Journal Issue:
Pattern Recognition Letters, 28(16)

Author:
Tang, Feng
Tao, Hai

Publication Date:
12-01-2007

Series:
UC Santa Cruz Previously Published Works

Permalink:
http://escholarship.org/uc/item/2tc2c68g

Additional Info:
The published version of this article is available at: www.elsevier.com/locate/patrec DOI 10.1016/j.patrec.2007.07.007

Keywords:
linear discriminant analysis, image representations, non-orthogonal binary subspace

Abstract:
Linear Discriminant Analysis (LDA) is a widely used technique for pattern classification. It seeks the linear projection of the data to a low dimensional subspace where the data features can be modelled with maximal discriminative power. The main computation in LDA is the dot product between LDA base vector and the data point which involves costly element-wise floating point multiplications. In this paper, we present a fast linear discriminant analysis method called binary LDA (B-LDA), which possesses the desirable property that the subspace projection operation can be computed very efficiently. We investigate the LDA guided non-orthogonal binary subspace method to find the binary LDA bases, each of which is a linear combination of, a small number of Haar-like box functions. We also show that B-LDA base vectors are nearly orthogonal to

each other. As a result, in the non-orthogonal vector decomposition process, the computationally intensive pseudo-inverse projection operator can be approximated by the direct dot product without causing significant distance distortion. This direct dot product projection can be computed as a linear combination of the dot products with a small number of Haar-like box functions which can be efficiently evaluated using the integral image. The proposed approach is applied to face recognition on ORL and FERET dataset. Experiments show that the discriminative power of binary LDA is preserved and the projection computation is significantly reduced. (c) 2007 Elsevier B.V. All rights reserved.

Fast Linear Discriminant Analysis using Binary Bases

Feng Tang and Hai Tao

Department of Computer Engineering
University of California
Santa Cruz, CA, USA
{tang|tao}@soe.ucsc.edu

Abstract

Linear Discriminant Analysis (LDA) is a widely used technique for pattern classification. It seeks the linear projection of the data to a low dimensional subspace where the data features can be modelled with maximal discriminative power. The main computation in LDA is the dot product between LDA base vector and the data point which involves costly element-wise floating point multiplication. In this paper, we present a fast linear discriminant analysis method called binary LDA (B-LDA), which possesses the desirable property that the subspace projection operation can be computed very efficiently. We investigate the LDA guided non-orthogonal binary subspace method to find the binary LDA bases, each of which is a linear combination of a small number of Haar-like box functions. We also show that B-LDA base vectors are nearly orthogonal to each other. As a result, in the non-orthogonal vector decomposition process, the computationally intensive pseudo-inverse projection operator can be approximated by the direct dot product without causing significant distance distortion. This direct dot product projection can be computed as a linear combination of the dot products with a small number of Haar-like box functions which can be efficiently evaluated using the integral image. The proposed approach is applied to face recognition on ORL and FERET dataset. Experiments show that the discriminative power of binary LDA is preserved and the projection computation is significantly reduced.

Key words: Linear discriminant analysis, image representations, non-orthogonal binary subspace.

1 Introduction and background

By finding the feature space that can best discriminate an object from others, discriminative methods have been successfully used in pattern classification applications including face recognition [2], image retrieval [17], tracking [10]. Linear discriminant analysis (LDA) is a widely used discriminative method. It provides a linear projection of the data into a low dimensional subspace with the outcome of maximum between-class variance and minimum within-class variances. LDA has been used for face recognition which is commonly called "Fisherface" [2].

1.1 Review of linear discriminant analysis

Linear Discriminant Analysis (LDA)is a class specific discriminative subspace representation that utilizes supervised learning to find a set of base vectors, denoted as $\mathbf{w}_i$, in such a way that the ratio of the between- and within-class scatters of the training sample set is maximized. This is equivalent to solving the following optimization problem:

$$\mathbf{E}_{opt} = \underset{\mathbf{E}=[\mathbf{e}_1,\mathbf{e}_2,\ldots,\mathbf{e}_K]}{\arg\max} \frac{|\mathbf{E}^T\mathbf{S}_b\mathbf{E}|}{|\mathbf{E}^T\mathbf{S}_w\mathbf{E}|} \tag{1}$$

where $\{\mathbf{e}_i | 1 \leq i \leq K\}$ are the LDA subspace base vectors, K is the dimension of the subspace. $\mathbf{S}_b$ and $\mathbf{S}_w$ are the between- and within-class scatter matrices, with the following forms,

$$\mathbf{S}_b = \sum_{i=1}^{c} M_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{2}$$

$$\mathbf{S}_w = \sum_{i=1}^{c} \sum_{\mathbf{x}_k \in \mathbf{X}_i} (\mathbf{x}_k - \mu_i)(\mathbf{x}_k - \mu_i)^T \tag{3}$$

where c is the number of classes, $\mathbf{x} \in \mathbf{R}^N$ is a data sample, $\mathbf{X}_i$ is the set of samples with class label i, μ_i is the mean for the all the samples in class-i, M_i is the number of samples in the class i. The optimization problem in Eq.(1) is equivalent to the generalized eigenvalue problem: $\mathbf{S}_b\mathbf{x} = \lambda\mathbf{S}_w\mathbf{x}$, for $\lambda \neq 0$. The solution can be obtained by applying an eigen-decomposition to the matrix $\mathbf{S}_w^{-1}\mathbf{S}_b$, if $\mathbf{S}_w$ is non-singular. The base vectors $\mathbf{E}$ sought in the above equation correspond to the first M most "significant" eigenvectors of $\mathbf{S}_\mathbf{w}^{-1}\mathbf{S}_b$ that corresponds to the K largest eigenvalues $\{\lambda_i | 1 \leq i \leq K\}$. These base vectors are orthogonal to each other. There are at most $c - 1$ eigenvectors corresponding to nonzero eigenvalues, since the rank of $\mathbf{S}_b$ is bounded by $c-1$. Therefore, the reduced dimension by LDA is at most $c-1$. A stable way to compute the eigen-decomposition is to apply SVD on the scattered matrix. In the case when the number of training samples is smaller than the dimensionality of the samples, the scatter matrices will be degenerated and will lead to the so called "small sample size" (SSS) problem. The SSS problem can be solved by incorporating a PCA step into the LDA framework [2]. PCA is used as a preprocessing step for dimensionality reduction so as to discard the null space of the within class scatter matrix of the training data set. Then LDA is performed in the lower dimensional PCA subspace [2]. Many methods [6, 9, 11, 7, 3, 23] have been proposed to solve this problem. In this paper, we focus on reducing the computational cost of LDA and assume the SSS problem is well solved by applying PCA on the data. The subspace spanned by the base vectors $\mathbf{E}$ is called LDA subspace. For a given test sample $\mathbf{x}$, we can obtain its representation in LDA subspace by a simple linear projection $\mathbf{E}^T\mathbf{x}$.

The main computation in LDA is the dot product of a data vector with all the LDA base vectors which involves element-by-element floating point multiplications. This can be computationally expensive especially when the original data is of high dimension or when there are many LDA base vectors.

1.2 Haar-like features and non-orthogonal binary subspace

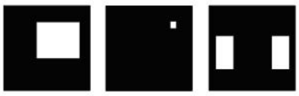

Fig. 1. Three typical one- and two-box functions. (a) and (b) are one-box functions and (c) is a symmetric two-box function.

In recent years, Haar-like box functions became a popular choice as image features due to the efficiency [21, 22]. Examples of such box functions are shown in Fig. 1. Formally, the binary function is defined as : $f(u,v) \in \{0,1\}$, $1 \le u \le w$, $1 \le v \le h$, w and h are the dimension of the binary function. The single Haar-like box function is defined as:

$$f(u,v) = \begin{cases} 1 & u_0 \le u \le u_0 + w' - 1, \\ & v_0 \le v \le v_0 + h' - 1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where w', h' are the size of the white box in Fig. 1, u_0, v_0 are the left up corner of the white box. For some symmetric objects like human faces, we can similarly define the vertically symmetric two-box binary function as:

$$f(u,v) = \begin{cases} 1 & u_0 \leq u \leq u_0 + w' - 1, \\ & v_0 \leq v \leq v_0 + h' - 1 \\ 1 & w - u_0 - w' + 1 \leq u \leq w - u_0, \\ & h - v_0 - h' + 1 \leq v \leq h - v_0 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

For an image of $w \times h$ pixels, there are $h(h+1)w(w+1)/4$ one-box base vectors and $h(h+1)w(w-1)/16$ symmetric two-box base vectors, we will denote this binary box dictionary as D.The main advantage of using these base functions is that the inner product of a data vector with each of them can be performed by several integer additions, instead of N floating point multiplications, where N is the dimension of the base vectors. This is achieved by computing the integral image $f_{\text{int}}(i,j)$ of the original image $f(i,j)$, which is defined as

$$f_{\text{int}}(i,j) = \sum_{m=1}^{i} \sum_{n=1}^{j} f(m,n) \tag{6}$$

The dot product of the image with a one-box base function is the summation of a rectangular area of the image, which can be computed efficiently as

$$\begin{aligned} &\sum_{i=top}^{bottom} \sum_{j=left}^{right} f(i,j) = f_{\text{int}}(bottom, right) - f_{\text{int}}(bottom, left-1) - \\ &f_{\text{int}}(top-1, right) + f_{\text{int}}(top-1, left-1) \end{aligned} \tag{7}$$

where $f(\cdot,\cdot)$ is the image function, $f_{int}(\cdot,\cdot)$ is the integral image of f. *top*, *bottom*, *left*, *right* are the coordinates that define the rectangular area. This technique has been used in many applications [21, 22, 20, 19, 8, 16, 13]. These binary box functions are generally non-orthogonal and the subspace spanned by binary box base vectors is called a *non-orthogonal binary subspace* (NBS) [19]. Tao *et al.* [19] propose to use an optimized orthogonal matching pursuit (OOMP) approach to find the set of binary base vectors to represent an image (details about OOMP will be addressed in section III). They show that an image can be approximated using NBS with arbitrary precision. As a result, it can be used to accelerate a wide variety of applications such as fast normalized cross correlation, fast object recognition. This has motivated us to investigate whether it is possible to use binary features to construct a subspace that has similar discriminative power as LDA but with significantly reduced computation as NBS. In [18], a similar approach has been use to accelerate the principal component analysis using NBS.

1.3 Our approach

This paper presents a novel subspace representation that has similar discriminative power as LDA, and at the same time, the classification process can be computed very efficiently using NBS. The idea is to represent each LDA base vector as a linear combination of Haar-like box functions. As the result, the dot product between the data vector and LDA bases can be computed efficiently using integral image. Main contributions of this paper include:

- A novel efficient discriminative subspace representation called binary LDA which has comparable classification performance as LDA but with much reduced computation.
- An LDA guided NBS method to obtain the binary LDA bases each of which is a linear combination of binary box functions.
- Theoretical analysis of the properties of B-LDA bases and the associated subspace projection.
- The application of the binary LDA method to face recognition.

The rest of the paper is organized as follows: in section 2, we formulate the B-LDA problem as an optimization problem. The proposed solution - LDA guided NBS method to find the B-LDA base vectors is discussed in section 3. In section 4, we conduct theoretical analysis of the properties of B-LDA bases. The speed improvement is shown in section 5. Experimental results are demonstrated in section 6. Section 7 concludes the paper.

2 The problem formulation

In the original LDA, the problem is formulated to find the linear subspace that can best discriminate the data within class from other classes. While in the proposed binary LDA, we aim at finding a subspace representation that can preserve the discrimination power of the traditional LDA and at the same time, reduce the computation cost involved in the floating point dot product. We then formulate the binary LDA as follows:

$$\mathbf{W}_{opt} = \underset{\mathbf{W}=[\mathbf{w}_1,\ldots,\mathbf{w}_K]}{\arg\max} \frac{|\mathbf{W}^T\mathbf{S}_b\mathbf{W}|}{|\mathbf{W}^T\mathbf{S}_w\mathbf{W}|} - \beta\sum_i c(\mathbf{w}_i) \tag{8}$$

$$\textit{subject to}: \ \mathbf{w}_i = \sum_j \alpha_j \mathbf{b}_j$$

where $\mathbf{b}_j$ is a binary box function from the dictionary D, examples of the base vectors in D are shown in Figure 1. α_j is the coefficient and $c(\mathbf{w}_i)$ is the computational cost of the projection of a data vector to the base vector $\mathbf{w}_i$. Basically, the objective function consists of two terms: the first term is the discriminative power term which is the ratio of the between- and within-class scatters; the second term is the computation cost term, with β as a positive weight to control the relative importance of the two terms. Since the B-LDA base vectors are represented as a linear combination of a small number of box functions, there is no guarantee that they are orthogonal to each other, so the B-LDA subspace is a non-orthogonal subspace. The relation between LDA and B-LDA subspaces is illustrated in Figure 2.

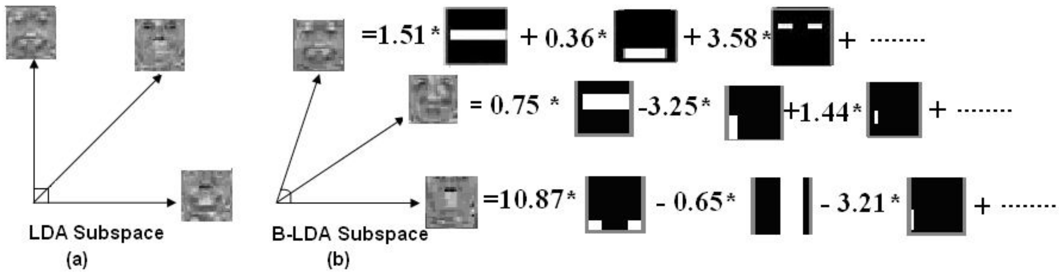

Fig. 2. Relation of LDA subspace (orthogonal) and B-LDA subspace (non-orthogonal).

3 Basis pursuit

In this section, we will first show the basis pursuit method used in NBS and why it cannot be directly applied to solve our problem. Then the LDA guided NBS method will be presented as the approximate solution to our problem of Eq. 6.

3.1 OOMP for basis pursuit

Base vectors for most orthogonal subspaces can be obtained in a principled way with mathematical decompositions or factorizations. But the problem of searching for the best subspace representation in a set of predefined non-orthogonal base vectors is known to be NP-hard [4]. Two of the popular greedy solutions to this problem include: the matching pursuit (MP) [12] and the optimized orthogonal matching pursuit (OOMP) method [14, 15]. The authors of [19] use OOMP to select the binary base vectors for NBS because it can provide a more accurate approximation of the input image than MP with the same number of base vectors.

Optimized orthogonal matching pursuit (OOMP) used in NBS [19] to find the base vectors is a technique for computing adaptive signal expansion by iterative selection of base vectors from a dictionary. Such a dictionary $D = \{\mathbf{b}_i\}_{i \in I}$ is usually non-orthogonal (binary box functions in our paper). Suppose Λ denotes the set of indices of the selected bases, the OOMP algorithm iteratively selects base vectors $\mathbf{B}_\Lambda = [\mathbf{b}_{l_1}, ..., \mathbf{b}_{l_{|\Lambda|}}]$ from D according to the following procedure: Suppose that at iteration k the already selected k base vectors are defined by the index set $\Lambda_k = (l_i)_{i=1}^k$. To find the next base vector in iteration $k+1$, the OOMP prescribes to select the index l_{k+1} that minimizes the new approximation error:

$$\varepsilon_{k+1} = \min_i \frac{|\langle \gamma_i, \varepsilon_k \rangle|}{\|\gamma_i\|}, \ \|\gamma_i\| \neq 0, \ i \in \bar{\Lambda}_k \tag{9}$$

where $\varepsilon_k = \mathbf{x} - R_{\mathbf{B}_{\Lambda_k}}(\mathbf{x})$ is the approximation error using $\mathbf{B}_{\Lambda_k}$ and $\gamma_i = \mathbf{b}_i - R_{\mathbf{B}_{\Lambda_k}}(\mathbf{b}_i)$. $R_{\mathbf{B}_\Lambda}(\mathbf{x}) = \mathbf{B}_\Lambda(\mathbf{B}_\Lambda^T \mathbf{B}_\Lambda)^{-1}\mathbf{B}_\Lambda^T \mathbf{x}$ is the reconstruction of the signal $\mathbf{x}$ using the non-orthogonal base vectors Λ_k. $\bar{\Lambda}_k$ is the subset of indices that are not selected in the previous iteration k, i.e. $\bar{\Lambda}_k = I - \Lambda_k$. An efficient implementation of this optimization can be achieved by the forward adaptive bi-orthogonalization [1]. In essence, OOMP is a greedy algorithm that finds a sub-optimal decomposition of data vector using minimum number of base vectors in D.

3.2 LDA guided NBS

The search space for the optimization problem in Eq.6 is extremely large because the solution can be any base vector that is a linear combination of any box functions $\mathbf{b}$ from the binary feature dictionary D. Even for a small image of size 24×24 used in our experiments, there are 134998 box functions in D. Suppose each B-LDA base vector is represented by 10 box functions, the number of possible choices of box functions for a single B-LDA base vectors is C_{134998}^{10}. This makes it impractical to find the global optimal solution.

One possible solution is to apply the LDA on the training data to obtain k LDA base vectors $[\mathbf{e}_1, ..., \mathbf{e}_k]$, then employ NBS to approximate each of these LDA base vectors with a given precision, and use the approximated vectors as the B-LDA base vectors $[\mathbf{w}_1, ..., \mathbf{w}_k]$. But the problem with this solution is that the approximation errors $(\mathbf{e}_i - \mathbf{w}_i)$ are generally not represented by any of the B-LDA base vectors, this leads to an inaccurate subspace.

To overcome this problem, we propose a LDA guided NBS method to find a sub-optimal solution efficiently. In the LDA guided NBS, we denote the selected B-LDA base vectors up to iteration k as $\mathbf{W}_k = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_k]$. This set is empty at the beginning. We start from the original LDA procedure to obtain the first principal component that captures the majority of the data variance. We call the first principal component the *Pre-LDA* vector, denoted as $\mathbf{w}_1^-$. NBS is then applied to approximate this vector as $\mathbf{w}_1 = \sum_{j=0}^{N_1} c_{j,1}\mathbf{b}_{j,1}$. Then, in iteration k, the data $\mathbf{X}$ is projected to the subspace spanned by the already selected B-LDA bases $\mathbf{W}_{k-1}$, and LDA is applied on the residual of the data $\mathbf{X} - R_{\mathbf{W}_{k-1}}(\mathbf{X})$ to obtain the next *Pre-LDA* $\mathbf{w}_k^-$ which is again approximated using NBS. The approximation of *Pre-LDA* at iteration k is called the *k-th B-LDA base vector.* This procedure iterates until the desired number of B-LDA bases have been obtained The flow of LDA guided NBS method is shown in Fig. 3

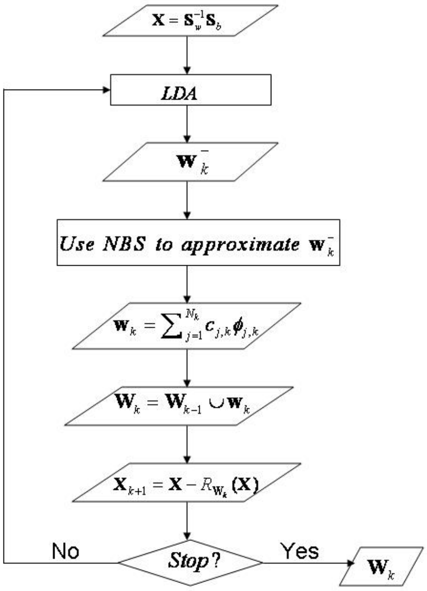
$\mathbf{X} = \mathbf{S}_w^{-1}\mathbf{S}_b$
LDA
$\mathbf{w}_k^-$
Use NBS to approximate $\mathbf{w}_k^-$
$\mathbf{w}_k = \sum_{j=1}^{N_k} c_{j,k}\phi_{j,k}$
$\mathbf{W}_k = \mathbf{W}_{k-1} \cup \mathbf{w}_k$
$\mathbf{X}_{k+1} = \mathbf{X} - R_{\mathbf{W}_k}(\mathbf{X})$
No
Stop?
Yes
$\mathbf{W}_k$

Fig. 3. The LDA guided NBS algorithm.

Generally, it takes a large number of box functions to represent each *Pre-BLDA* perfectly. However, the computational cost term in the objective function prefers a solution with fewer box functions. To make the optimization simpler, we enforce a computational cost constraint by finding the minimum number of box functions that satisfy:

$$(1-\tau)||\mathbf{w}||^2 \leq ||\overline{\mathbf{w}^-}||^2 \leq ||\mathbf{w}||^2 \tag{10}$$

where $\overline{\mathbf{w}_1^-}$ is the reconstruction of $\mathbf{w}_1^-$ using binary box functions. $\tau \in [0, 1]$ is the approximation error threshold that controls the precision. A smaller value of τ tends to produce a more accurate approximation. N is the dimension of the base vector. Fig. 4 demonstrates the selected box functions used to approximate the first iteratively selected LDA base vectors. The comparison of LDA, pre-BLDA and B-LDA base vectors are shown in Fig. 5.

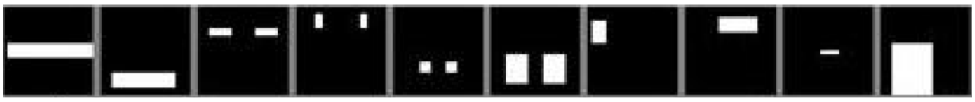

Fig. 4. Some of LDA-guided OOMP selected box functions to approximate the first binary LDA base vector.

Since these bases are linear combinations of binary box functions, there is no guarantee that they are orthogonal to each other. As a result, the reconstruction process becomes $P_{\mathbf{W}}(\mathbf{x}) = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{x}$. This pseudo-inverse projection can be approximated using direct dot product (DNP): $P_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$. Experiments show that this approximation does not cause much performance deduction. In the next section, we will denote that the error between the direct dot product signal representation in B-LDA subspace and that in LDA subspace has an upper bound.

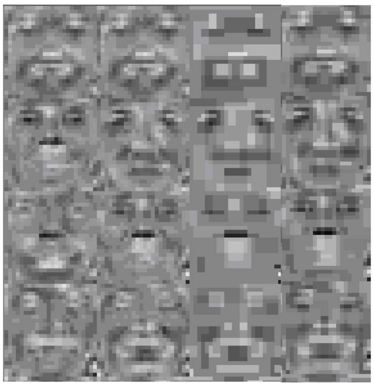

Fig. 5. Comparison of the original LDA bases, pre-LDA bases, binary LDA bases ($\tau = 0.8$), binary LDA bases ($\tau = 0.5$), from left to right.

4 Theoretical analysis of the B-LDA bases

As mentioned in the previous section, when the approximation error threshold τ is 0, the B-LDA base vector is identical to the LDA base vector. When τ increases, B-LDA base vectors deviate from the LDA bases and also become more non-orthogonal. Non-orthogonality, which is often measured using *coherence*, will be defined in this section. We will prove that by approximating the original projection process $P_{\mathbf{W}}(\mathbf{x}) = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{x}$ with the *direct dot non-orthogonal projection process* (DNP): $P_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$, the resultant distance error of $P_{\mathbf{W}}(\mathbf{x})$ is related to coherence and therefore τ. Based on this property, we conclude that when τ is small, the information loss by using B-LDA and DNP is also small, while the computational complexity is reduced significantly. This was verified by our experiments on real datasets.

Definition 1. *A* μ-coherent *base vector set* $\mathbf{W}$ *has coherence* μ *for* $0 \leq \mu \leq 1$,

if $|\langle \mathbf{w}_i, \mathbf{w}_j \rangle| \leq \mu$ *for all distinct* $\mathbf{w}_i, \mathbf{w}_j \in W$. *Intuitively, for a* μ-coherent *dictionary the angle between any pair of base vectors or the negation of the vectors has to be larger than* $|\cos^{-1}\mu|$. *A* 0-coherent *base vector set is orthogonal.*
Lemma 1. *If we denote* $B = |\mathbf{W}|$ *and* $\mu B \leq 0.5$, *then there exists a set of vectors* $\mathbf{e}_i, i = 1, ..., B$, *such that*

- *The* $\mathbf{e}_i$*'s form an orthonormal system;*
- *Span* $(\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_B) =$ *span* $(\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_B)$*;*
- $||\mathbf{e}_i - \mathbf{w}_i||^2 \leq 8\mu^2 B$.

This lemma states that when the coherence satisfies the above conditions, we can find an orthonormal system that has the same span as the non-orthogonal base vectors. In addition, these orthonormal base vectors are very close to the original non-orthogonal ones. The distance between corresponding base vectors is a function of coherence. Proof can be found in [5].
Lemma 2. *The angle* θ_i *between each non-orthogonal base vector* $\mathbf{w}_i$ *and it's corresponding orthogonal base vector* $\mathbf{e}_i$ *is smaller than* $\theta_{max} = 2\sin^{-1}(2\mu^2 B)^{1/2}$, *where* $B = |W|$.

Proof. See the Appendix.

□

Theorem 1. *By approximating the original projection process* $P_{\mathbf{W}}(\mathbf{x}) = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{x}$ *with the direct dot non-orthogonal projection process (DNP)* $\widehat{P}_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$, *the resultant distance error of* $P_{\mathbf{W}}(\mathbf{x})$ *is bounded by a function of* μ, *i.e.,* $||\widehat{P}_{\mathbf{W}}(\mathbf{x})|| - ||P_{\mathbf{W}}(\mathbf{x})|| \leq g(\mu) = (\sqrt{\sum_i c_i'^2} - 1)||\mathbf{x}|| \leq (\sqrt{1 + 2(B-1)H + BH^2} - 1)||\mathbf{x}||$ *where* $H = \sqrt{8\mu^2 B(1 - 2\mu^2 B)}$.

Proof. See the Appendix. □

5 Speed Improvement

Suppose the image size is $m \times n$, T_{LDA} denotes the time for computing the LDA subspace projection coefficients and K denotes the number of LDA base vectors. It will take $m \times n \times K$ floating point multiplications and $K \times (m \times n - 1)$ floating point additions to perform the projection operation, or

$$T_{LDA} = K \times m \times n \times T_{fm} + K \times (m \times n - 1) \times T_{fa} \tag{11}$$

where T_{fm} is the time for a single floating point multiplication and T_{fa} is the time for a single floating point addition.

For B-LDA, the time for a single projection is denoted as T_{BLDA} which consists of two parts. One part is T_{ii}, the time to construct the integral image. For an $m \times n$ image, it will take $m \times n \times 2$ integer additions with recursive implementation. This is performed only once for each image. The other part is the time for the projection operation $P_{\mathbf{W}}(\mathbf{x}) = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{x}$. When the bases are nearly orthogonal to each other, we can approximate the projection coefficient using the direct dot-product $\mathbf{W}^T\mathbf{x}$. The B-LDA base vector $\mathbf{w}_i$ $(1 \leq i \leq K)$ is represented as a linear combination of N_i box functions, $\mathbf{w}_i = \sum_{j=1}^{N_i} c_j \mathbf{b}_j$. The projection of $\mathbf{x}$ to $\mathbf{w}_i$ can be written as $\langle \mathbf{w}_i, \mathbf{x} \rangle = \sum_{j=1}^{N_i} c_{i,j} \langle \mathbf{b}_j, \mathbf{x} \rangle$. Each box function $\mathbf{b}_j$ has n_j boxes, where n_j can be one or two. The $\langle \mathbf{b}_j, \mathbf{x} \rangle$ can be performed using $3 \times n_j$ integer additions. Since c_j is floating point, $\langle \mathbf{w}_i, \mathbf{x} \rangle$ needs N_i floating point multiplications and $N_i - 1$ floating point additions.

$$T_{BLDA} = T_{ii} + \sum_{i=1}^{K} \sum_{j=1}^{N_i} (3 \times n_j \times T_{ia} + N_i \times T_{fm} + (N_i - 1) \times T_{fa}) \qquad (12)$$

where T_{ia} is the time for one integer addition. As we can observe, T_{BLDA} is only dependent on the number of binary box functions which is often much less than the dimension of the image. Note the dot product between the image and box functions $\mathbf{w}_j^T\mathbf{x}$ can be computed using 3 or 7 integer additions using integral image trick. For LDA, however, the time is proportional to the image dimension. Since the number of operations in B-LDA is much smaller than LDA, T_{BLDA} is much less than T_{LDA}, and the speed up is more dramatic with higher dimensional data. Using B-LDA, the computation is reduced from $O(N)$ (N is the data dimension) to constant, which is only related to the number of box functions used to approximate each LDA base vector.

Suppose $m = n = 24$, $K = 15$, then T_{LDA} needs $24 \times 24 \times 15 = 8640$ floating point multiplications to compute the projection coefficients. Suppose the total number of NBS base vectors used to represent all the B-LDA base vectors is 200, that is, $\sum_{i=1}^{K} N_i = 200$, then the B-LDA projection only needs between $\sum_{i=1}^{K} N_i = 200$ and $2 \times \sum_{i=1}^{K} N_i = 400$ floating point operations. The speed up is significant.

6 Experiments

We tested the proposed B-LDA method for face recognition. B-LDA is applied on the training data to find the bases, then the testing images are projected onto these bases to obtain the feature vector, the classification is achieved using nearest neighbor. Extensive experiments are carried out on two popular dataset ORL and FERET. Promising results have been obtained.

6.1 ORL dataset

The ORL (Olivetti Research Laboratory) face database is used in our first experiment. It contains 400 images of 40 individuals. Each image is originally 64×64, but is down-sampled to 24×24 in our application. Some images were captured at different times and have different variations including expression (open or closed eyes, smiling or non-smiling) and facial details (glasses or no glasses). The images were taken with a tolerance for some tilting and rotation of the face up to 20 degrees. We use 320 images (8 for each person) for training to build a 15 dimensional B-LDA subspace and the other 80 for testing. Note we handle the small sample size problem by first performing a PCA process before LDA, and we approximate the product of PCA base vector and LDA base vector as B-LDA base vector.

The B-LDA bases coherence μ and recognition performance are directly influenced by the approximation threshold τ in the LDA guided NBS. With a higher threshold, which implies a less accurate approximation, the coherence will increase and the bases become less orthogonal. When the base vectors are more orthogonal (smaller τ), B-LDA base vectors become more similar to LDA base vectors. We have listed the coherence of the B-LDA base vectors with different τ in Table-1. To make the coherence easier to understand, we also show the angle between the original LDA base vector and the corresponding B-LDA base vector (denoted as θ in degrees). From section 4, we can easily see $\theta = cos^{-1}\mu$.

As can be seen from section 5, the key factor in determining the speed improvement of B-LDA over LDA is the number of box functions used to approximate the LDA base vectors. We denote $\sum_{i=1}^{15} N_i$ as the total number of box functions to represent the B-LDA bases, and denote $\overline{K}$ as the average number of box functions used to approximate a single B-LDA base vector. As can be observed from Table-2, the larger the approximation threshold, the less number of box functions is needed, which means a less accurate approximation, but more efficient.

In order to show the effectiveness of the B-LDA approach, we compare the recognition performance of B-LDA and LDA in Fig. 6, 7, 8, 9, 10, 11. Fig. 6 shows the performance of LDA and B-LDA under different approximation thresholds (τ). Note in this curve, the projection process is done using original pseudo-inverse projection. As can be observed, roughly when τ is small, the recognition performance is close to the LDA performance. This is because the B-LDA subspace is more similar to the LDA subspace. Fig. 7 shows the performance comparison of LDA and B-LDA using DNP, which is less accurate but more efficient. To make the easier to see, we also show the performance

Table 1
B-LDA base vectors properties with different approximation thresholds.

τ	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
μ	0.0238	0.0311	0.0489	0.0828	0.1323	0.1488	0.3280	0.2940
θ	0.9976	2.2417	2.8609	4.8743	5.2661	7.2187	14.1648	19.2949
$\overline{K}$	81	50	33	22	15	10	7	3
$\sum_{i=1}^{15} N_i$	1225	749	493	331	230	147	96	41

comparison between LDA, B-LDA pseudo-inverse projection and B-LDA DNP with fixed τ in Fig. 8, 9, 10, 11.

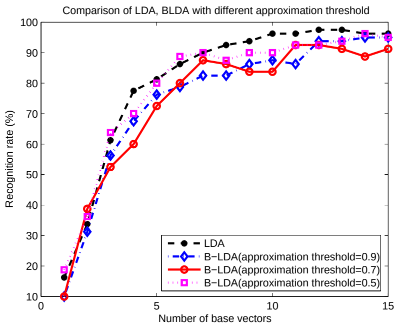

Fig. 6. Performance comparison between LDA, B-LDA(using pseudo-inverse projection) with different number of base vectors.

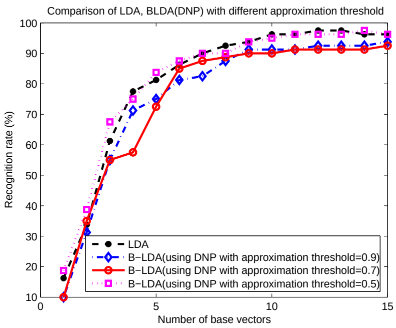

Fig. 7. Performance comparison between LDA, B-LDA(using DNP) with different number of base vectors.

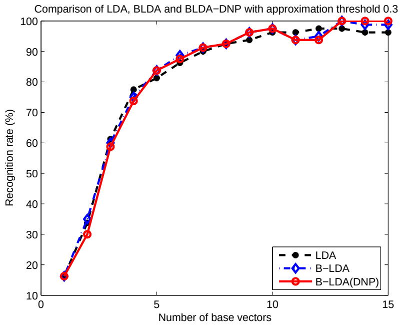

Fig. 8. Performance comparison between LDA, B-LDA and B-LDA(DNP) using different number of base vectors. The approximation threshold for B-LDA is set to 0.3.

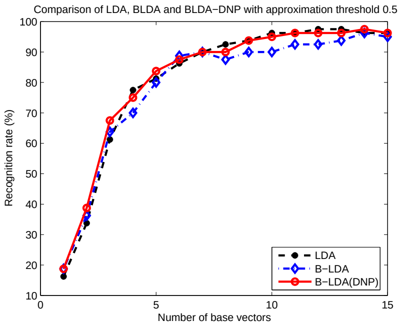

Fig. 9. Performance comparison between LDA, B-LDA and B-LDA(DNP) using different number of base vectors. The approximation threshold for B-LDA is set to 0.5.

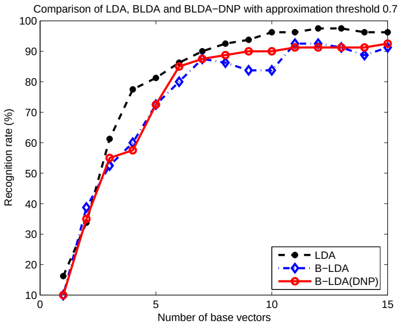

Fig. 10. Performance comparison between LDA, B-LDA and B-LDA(DNP) using different number of base vectors. The approximation threshold for B-LDA is set to 0.7.

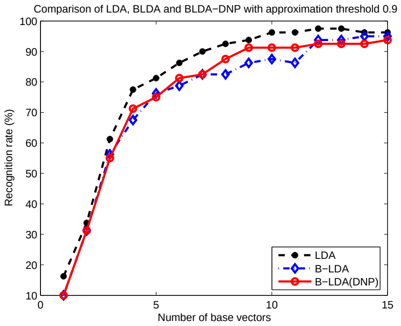

Fig. 11. Performance comparison between LDA, B-LDA and B-LDA(DNP) using different number of base vectors. The approximation threshold for B-LDA is set to 0.9.

6.2 FERET dataset

The second experiment is on a selected set of 500 frontal view images from the the FERET dataset. These images were spatially aligned and scaled to 24×24 pixels. Using 342 training samples of 64 different persons, the B-LDA base vectors are computed. The first 15 of these vectors are computed and the first four of them are shown in Fig. 5. It can be observed that, like LDA, B-LDA base vectors can capture the face structure. Each individual base vector resembles some face shape. However, the B-LDA base vectors appear to be blocky due to the approximation using box functions. Fig. 4 shows the features used to approximate the first pre-LDA base vector. Fig. 12 shows the classification performance using pseudo-inverse projection. As can be observed, in general, the performance increases with the number of base vectors being used. Even with the approximation error τ to be 0.8 (very coarse approximation), the classification performance of B-LDA is comparable to LDA.

To show the effectiveness of DNP, the coherence for B-LDA bases under different approximation thresholds are computed, they are listed in Table 2. As can be observed, the smaller the approximation thresholds (more accurate approximation), the smaller the bases coherence, which means that the B-LDA base vectors are more orthogonal to each other. As a result, the difference between DNP and pseudo-inverse projection is small. Fig. 13 shows the recognition result by setting the approximation threshold $\tau = 0.5$. As can be observed, the difference is slight. Fig. 14 is the comparison of the recognition performance using direct dot product (DNP) with different approximation thresholds. As can be observed, the smaller the τ, the smaller difference between B-LDA DNP and LDA.

Table 2

B-LDA base vectors coherence μ with different approximation thresholds τ.

τ	0.2	0.5	0.8
μ	0.0085	0.0836	0.1731

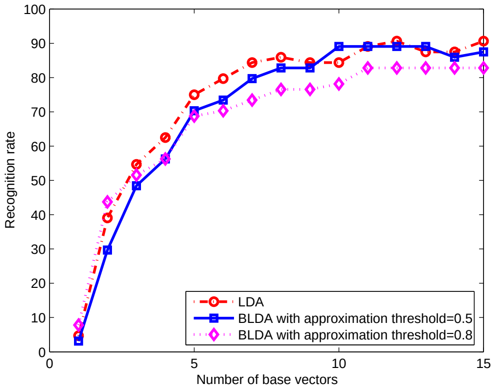

Fig. 12. Comparison of the LDA and B-LDA performance.

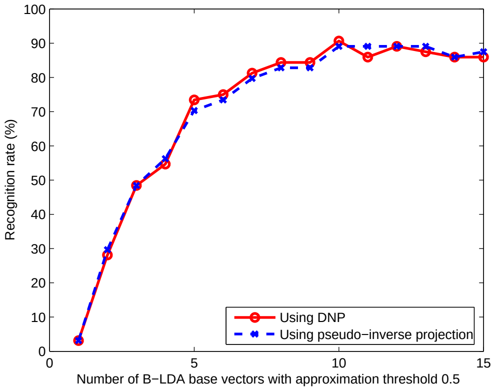

Fig. 13. Comparison of the DNP with pseudo-inverse projection for face recognition with $\tau = 0.5$

The experiment for speed improvement is carried out on a Pentium IV, 3.2 GHz, 1G RAM machine, using C++ code. Fifteen base vectors are computed, for both LDA and B-LDA, and the time to project images onto each subspace is observed. The B-LDA used direct dot product projection. We tested 500 samples and use the average time of a single projection operation and the results are listed in Table 2.

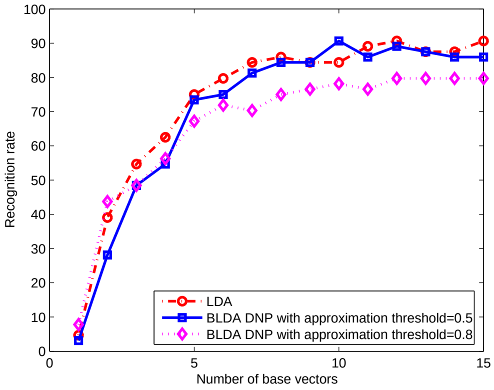

Fig. 14. The recognition rate of B-LDA using the direct non-orthogonal projection (DNP).

7 Conclusion

A novel efficient discriminative method called B-LDA is presented in this paper. It inherits the properties of LDA in terms of discriminating data from different class while take advantages of the computational efficiency of non-orthogonal binary bases. We proposed an LDA guided NBS method to obtain the B-LDA base vectors and applied the B-LDA to the face recognition. Experiments show that the discriminative power of LDA is preserved and the computation is significantly reduced.

References

[1] M. Andrle and L. Rebollo-Neira. A swapping-based refinement of orthogonal matching pursuit strategies. *Signal Processing, special issue on Sparse Approximations in Signal and Image Processing*, 86(3):480–495, March 2006.

[2] P.N. Belhumeur, J. Hespanda, and D. Kiregeman. Eigenfaces

Table 3
Comparison of the computational cost between LDA and B-LDA projection operation.

$Threshold : \tau$	0.5	0.8
#box functions	424	99
T_{LDA} (ms)	0.315	
T_{ii} (integral image) (ms)	4.72×10^{-3}	
T_{BLDA}(ms)	8.13×10^{-3}	2.10×10^{-3}
Speedup($\frac{T_{LDA}}{T_{BLDA}+T_{ii}}$)	24.51	46.18

vs.fisherfaces: Recognition using class specific linear projection. *PAMI*, 19(7):711–720, July 1997.

[3] S. Chen and D. Li. Modified linear discriminant analysis. *Pattern Recognition*, 38(3):441–443, 2005.

[4] G. Davis, S. Mallat, and M. Avellaneda. Greedy adaptive approximation. *Journal of Constructive Approximation*, 13(1):57–98, March 1997.

[5] A.C. Gilbert, S. Muthukrishnan, and M. Strauss. Approximation of functions over redundant dictionaries using coherence. In *Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms 2003*, pages 243–252, 2003.

[6] R. Huang, Q. Liu, H. Lu, and S. Ma. Solving the small sample size problem of ldaf. In *ICPR02*, pages III: 29–32, 2002.

[7] X.Y. Jing, D. Zhang, and Y.F. Yao. Improvements on the linear discrimination technique with application to face recognition. 24(15), November 2003.

[8] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *ICCV*, pages I 166–173, 2005.

[9] M. Li and B. Yuan. 2d-lda: A statistical linear discriminant analysis for image matrix. *Pattern Recognition Letters*, 26(5):527–532, 2005.

[10] Yang Lin, R., M.H., and S.E. Levinson. Object tracking using incremental fisher discriminant analysis. In *ICPR*, pages 23–26, 2004.

[11] J. Lu, K. N. Plataniotis, and A. Venetsanopoulos. Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition. *Pattern Recognition Letters*, 26(2):181–191, 2005.

[12] S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Trans on Signal Processing*, 41(12):3397–3415, December 1993.

[13] T. Mita, T. Kaneko, and O. Hori. Joint haar-like features for face detection. In *ICCV*, pages II: 1619–1626, 2005.

[14] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. Orthogonal matching pursuits: Recursive function approximation with applications to wavelet decomposition. In *Proc. 27th Asilomar Conf. Signals, Systems, Computers*, pages 40–44, 1993.

[15] L. Rebollo-Neira and D. Lowe. Optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9(4):137–140, April 2002.

[16] H. Schweitzer, J.W. Bell, and F. Wu. Very fast template matching. In *ECCV*, pages 358–372, 2002.

[17] D.L. Swets and J. Weng. Hierarchical discriminant analysis for image retrieval. *PAMI*, 21(5):386–401, May 1999.

[18] F. Tang and H. Tao. Binary principal component analysis. In *British Machine Vision Conference*, page I:377, 2006.

[19] H. Tao, R. Crabb, and F. Tang. Non-orthogonal binary subspace and its applications in computer vision. In *ICCV*, pages 864–870, 2005.

[20] O. Veksler. Fast variable window for stereo correspondence using integral

images. In *CVPR*, pages I: 556–561, 2003.

[21] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, pages I: 511–518, 2001.

[22] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, pages II: 734–741, 2003.

[23] X. Zhuang and D. Dai. Inverse fisher discriminate criteria for small sample size problem and its application to face recognition. *Pattern Recognition*, 38(11):2192–2194, 2005.

8 Appendix

Lemma 1. *If we denote $B = |\mathbf{W}|$ and $\mu B \leq 0.5$, then there exists a set of vectors $\mathbf{e}_i, i = 1, ..., B$, such that*

- *The $\mathbf{e}_i$'s form an orthonormal system;*
- *Span $(\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_B)$ = span $(\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_B)$;*
- *$||\mathbf{e}_i - \mathbf{w}_i||^2 \leq 8\mu^2 B$.*

Lemma 2. *The angle θ_i between the non-orthogonal base vector $\mathbf{w}_i$ and its corresponding orthogonal base vector $\mathbf{e}_i$ is smaller than $\theta_{max} = 2\sin^{-1}(2\mu^2 B)^{1/2}$, where $B = |W|$.*

Proof. From the lemma-1 in section IV-D, we have $||\mathbf{e}_i - \mathbf{w}_i|| \leq 2(2\mu^2 B)^{1/2}$, since $\sin(\theta/2) = \frac{1}{2}||\mathbf{e}_i - \mathbf{w}_i||/||\mathbf{w}_i|| = \frac{1}{2}||\mathbf{e}_i - \mathbf{w}_i|| \leq (2\mu^2 B)^{1/2}$, so $\theta \leq \theta_{max} = 2\sin^{-1}(2\mu^2 B)^{1/2}$, as shown in Fig. 15. □

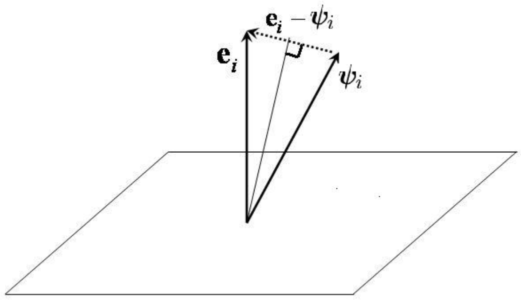

Fig. 15. Relation between non-orthogonal base vector and its corresponding orthogonal base vector.

Theorem 1. *By approximating the original projection process $P_{\mathbf{W}}(\mathbf{x}) = (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{x}$ with the direct dot non-orthogonal projection process (DNP) $\widehat{P}_{\mathbf{W}}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$, the resultant distance error of $P_{\mathbf{W}}(\mathbf{x})$ is bounded by a function of μ, i.e., $||\widehat{P}_{\mathbf{W}}(\mathbf{x})|| - ||P_{\mathbf{W}}(\mathbf{x})|| \leq g(\mu) = (\sqrt{\sum_i c_i'^2} - 1)||\mathbf{x}|| \leq (\sqrt{1 + 2(B-1)H + BH^2 -}$*

1)$||\mathbf{x}||$ *where* $H = \sqrt{8\mu^2 B(1-2\mu^2 B)}$.

Proof. Suppose a unit data vector $\mathbf{x}$ is presented by $\mathbf{e}_i$'s as $\mathbf{x} = c_1\mathbf{e}_1 + c_2\mathbf{e}_2 + ... + c_B\mathbf{e}_B$, the DNP coefficients of $\mathbf{x}$ are $c_i' = \langle c_1\mathbf{e}_1 + c_2\mathbf{e}_2 + ... + c_B\mathbf{e}_B, \mathbf{w}_i\rangle$. The distance error of DNP is $\sqrt{\sum_i^B c_i'^2} - 1$. It's obvious that:

$$\begin{cases} \cos\theta_{max} \leq \langle \mathbf{e}_i, \mathbf{w}_i\rangle \ \leq \ 1 \ \ i \in [1, ..., B] \\ |\langle \mathbf{e}_i, \mathbf{w}_j\rangle| \ \leq \ \sin\theta_{max} \ \ i, j \in [1, ..., B], i \neq j \end{cases}$$

where θ_{max} is the angle defined in Lemma. Then, we have

$$\begin{aligned} & c_i'^2 = \sum_{m,n} c_m\langle \mathbf{e}_m, \mathbf{w}_i\rangle c_n\langle \mathbf{e}_n, \mathbf{w}_i\rangle \\ & = \sum_{m=n=i} c_i^2\langle \mathbf{e}_i, \mathbf{w}_i\rangle^2 + 2\sum_{m\neq i} c_m c_i\langle \mathbf{e}_m, \mathbf{w}_i\rangle\langle \mathbf{e}_i, \mathbf{w}_i\rangle + \sum_{m\neq i, n\neq i} c_m c_n\langle \mathbf{e}_m, \mathbf{w}_i\rangle\langle \mathbf{e}_n, \mathbf{w}_i\rangle \\ & \leq c_i^2 + 2\sum_{m\neq i} c_m c_i \sin\theta_{max} + \sum_{m\neq i, n\neq i} c_m c_n \sin^2\theta_{max} \end{aligned}$$

and

$$\begin{aligned} \sum_i c_i'^2 \leq \sum_i (c_i^2 + 2\sum_{m\neq i} c_m c_i \sin\theta_{max} + \sum_{m\neq i, n\neq i} c_m c_n \sin^2\theta_{max}) = \\ \sum_i (c_i^2 + 2(\sum_{m,n} c_m c_n - 1)\sin\theta_{max} + \sum_{m\neq i, n\neq i} c_m c_n \sin^2\theta_{max}) \end{aligned}$$

It's easy to see, when $\sum_i c_i^2 = 1$, $\sum_{m,n} c_m c_n$ is maximized if and only if $c_1 = c_2 = ... = c_B = \frac{1}{\sqrt{B}}$. Then, we have

$$\sum_i c_i'^2 \leq \sum_i c_i^2 + 2[B^2(\frac{1}{\sqrt{B}})^2 - 1]\sin\theta + B\sin^2\theta = 1 + 2(B-1)\sin\theta + B\sin^2\theta$$

According to the Lemma, $\theta_{max} \leq 2\sin^{-1}(2\mu^2 B)^{1/2}$,

$$\begin{aligned} & \sin^2\theta_{max} \leq [\sin(2\sin^{-1}(2\mu^2 B)^{1/2})]^2 \\ & = [2\sin(\sin^{-1}(2\mu^2 B)^{1/2})\cos(\sin^{-1}(2\mu^2 B)^{1/2})]^2 \\ & = 8\mu^2 B[1 - \sin^2(\sin^{-1}(2\mu^2 B)^{1/2})] \\ & = 8\mu^2 B(1 - 2\mu^2 B) \end{aligned}$$

so,

$$\sqrt{\sum_i c_i'^2} - 1 \leq \sqrt{1 + 2(B-1)H + BH^2} - 1$$

where $H = \sqrt{8\mu^2 B(1 - 2\mu^2 B)}$.

so the squared distance error of DNP is bounded by: $\sqrt{1 + 2(B-1)H + BH^2} - 1$ where $H = \sqrt{8\mu^2 B(1 - 2\mu^2 B)}$. □