Non-orthogonal Binary Subspace and its Applications in Computer Vision

Hai Tao, Ryan Crabb and Feng Tang
Department of Computer Engineering, University of California, Santa Cruz
{tao,rcrabb,tang}@soe.ucsc.edu

Abstract

This paper presents a novel approach that represents an image or a set of images using a non-orthogonal binary subspace (NBS) spanned by box-like base vectors. These base vectors possess the property that the inner product operation with them can be computed very efficiently. We investigate the optimized orthogonal matching pursuit method for finding the best NBS base vectors. It is demonstrated in this paper how the NBS based expansion can be applied to speed up several common computer vision algorithms, including normalized cross correlation (NCC), sum of squared difference (SSD) matching, appearance subspace projection and subspace-based object recognition. Promising experimental results on facial and natural images are demonstrated in this paper.

1. Introduction

By exploiting the spatial correlation among pixels in two-dimensional images, subspace methods can represent images in compact forms. The principle behind these methods is that even though image vectors are described as high dimensional data points with the pixel intensity values as coordinates, they usually resides in a rather low dimensional subspace that preserves most of the signal energy. Such a subspace can be found using the principal component analysis (PCA) method. The subspace methods have been successfully applied in object appearance modeling with applications in object recognition, detection, and visual tracking.

This paper investigates an alternative class of subspace that is spanned by non-orthogonal binary base vectors. As shown in the top three rows of Figure 1, these base vectors consist of one or two 2D rectangular areas. In these areas, the base image values are 1. The rest image area has value 0. The main benefit of using this form of base vectors is that the projection process can be computed very efficiently. The inner product of an input vector with these base vectors can be computed as the image intensity sum in one or two rectangular image areas. As shown in [11,18], only three or six integer additions are needed by using the integral image. The bottom three rows in Figure 1 show the reconstructions of the input face image in the middle row using binary base vectors. With each additional base vector, more details are added to the reconstructed image.

This work is inspired by previous effort in fast image correlation and template matching algorithms [5,9,14], the box filter based object detection methods [18,19], and recent papers in highly nonlinear approximation [8]. The main contributions of this paper are

1. A novel non-orthogonal binary subspace representation for efficient image expansion.
2. An efficient incremental algorithm for finding the base vectors that preserve the most data energy.
3. Applications of the proposed non-orthogonal binary subspace method in computing normalized cross correlation, fast sum of squared difference matching, projection residual computation, and subspace based object recognition.

Figure 1. Top three rows: the first 30 non-orthogonal binary base vectors computed using 500 face images. Middle row: an input face image. Bottom three rows: the reconstructed face images, using one to thirty of the base vectors.

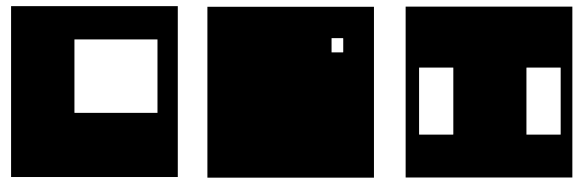
The paper is organized as follows. In Section 2, box-like base vectors, their properties and advantages are discussed. In Section 3, a method for obtaining the optimal box base vectors is presented. The applications of the NBS method are described in Section 4. Section 5 presents the implementation details and experimental results. Conclusions and future work can be found in Section 6.

2. Non-orthogonal binary subspace

The images of objects often lie in a low dimensional manifold, with the dimension roughly equal to the number of physical factors that affect the object appearance. These factors may include the viewing angles, the light source directions, and the rigid or nonrigid motion of the object. Linear subspace method is used extensively in computer vision algorithms to describe such a manifold in order to obtain compact representations. Some examples using this technique include the Eigenface method[17], the EigenTracking algorithm [4], Illumination subspaces [2,3], and the non-negative matrix factorization method [6,10]. Template matching method can also be considered as a special case of linear subspace projection where a one-dimensional subspace is used with the template as the base vector.

A subspace can be represented using either orthogonal or non-orthogonal base vectors. Orthogonal subspaces in DCT, Walsh-Hardmad transform, wavelet transform, and the one derived from PCA are often used. These subspace representations are compact and the projection of a data vector using these subspaces require computing inner dot product with each base vector.

As mentioned Section 1, our motivation for using non-orthogonal binary basis functions, more specifically the box-like base vectors, is that the subspace projection process can be computed very efficiently. As shown in Figure 2, a base vector with one rectangle is defined by two of its corners. A base vector with two vertically symmetric rectangles is defined similarly. For an image of $h \times w$ 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. For representing objects of symmetric appearance, symmetric two-box base vectors such as the one in Figure 2(c) are more effective. It is easy to prove that the set of these two types of basis functions are complete, redundant, and non-orthogonal.



(a) (b) (c)

Figure 2. Three typical box basis functions. (a) and (b) are one-box base vectors and (c) is a symmetric two-box base vector.

The main advantage of using the above basis functions is that the inner product of a data vector with them can be performed by three or six integer additions, instead of d floating point multiplications, where $d = h \times w$ is the dimension of the base vectors. This is achieved by first computing the integral image $f_{\text{int}}(i, j)$ of the original image $f(i, j)$. The integral image is defined as

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

The multiplication of an image patch with a one-box basis function is the summation of a rectangular area of the image, which can be computed efficiently using three additions based on the integral image. This process is formulated as

$$\begin{aligned} \sum_{i=top}^{bottom} \sum_{j=left}^{right} f(i, j) &= f_{\text{int}}(bottom, right) - \\ &- f_{\text{int}}(bottom, left-1) - f_{\text{int}}(top-1, right) + \\ &+ f_{\text{int}}(top-1, left-1) \end{aligned} \quad (2)$$

where top, $bottom$, $left$, $right$ are coordinates that define the rectangular area. This trick has been used in [11,19]. In the applications that will be investigated in this paper, we show that significant speedup can be achieved.

3. Optimal non-orthogonal binary basis

3.1. The problem statement

For a given signal $\mathbf{x} \in R^N$ and a dictionary $D = \{\phi_i\}_{i \in I}$ that consists of a set of base vectors, with the index set $I = \{1, ..., M\}$, $\mathbf{x}$ can be approximated using base vectors as $\hat{\mathbf{x}} = \sum_{i \in \Lambda} \alpha_i \phi_i$, where Λ is a set of indices of the vectors in D and $\Lambda \subseteq I$. Typically $|\Lambda| << N$, so the representation is compact. The subspace spanned by $D_\Lambda = \{\phi_i\}_{i\in\Lambda}$ is represented by a matrix formed by the base vectors $\Phi_\Lambda = [\phi_{l1},...,\phi_{l|\Lambda|}]$. The subspace projection operator is denoted as $P_{\Phi_\Lambda}(\cdot)$, therefore, $\hat{\mathbf{x}} = P_{\Phi_\Lambda}(\mathbf{x})$. For a non-orthogonal D, the project coefficients are computed as

$$[\alpha_1,...,\alpha_{|\Lambda|}]^T = (\Phi_\Lambda^T \Phi_\Lambda)^{-1} \Phi_\Lambda^T \mathbf{x}\,. \quad (3)$$

The project process is performed as

$$P_{\Phi_\Lambda}(\mathbf{x}) = \Phi_\Lambda (\Phi_\Lambda^T \Phi_\Lambda)^{-1} \Phi_\Lambda^T \mathbf{x}\,. \quad (4)$$

The goal of an optimal approximation algorithm is to find a set of base vectors indexed by Λ, so that the following objective function is minimized.

$$\arg\min_\Lambda \| \mathbf{x} - P_{\Phi_\Lambda}(\mathbf{x}) \| + \alpha \textstyle\sum_{i\in\Lambda} c(\phi_i) \quad (5)$$

where $c(\phi_i)$ is the cost function associated with computing inner product with base vector ϕ_i. α is a weighting factor.

When a set of signal $\mathbf{X} = [\mathbf{x}_1,...,\mathbf{x}_S]$ is approximated using non-orthogonal binary basis, the formulation is identical except that the notations represent matrices instead of vectors.

$$\begin{aligned}\arg\min_\Lambda trace[(\mathbf{X} - P_{\Phi_\Lambda}(\mathbf{X}))^T (\mathbf{X} - P_{\Phi_\Lambda}(\mathbf{X}))] + \\ + \alpha S \textstyle\sum_{i\in\Lambda} c(\phi_i)\end{aligned} \quad (6)$$

Eq.(5) and Eq.(6) are the core problem in the newly emerging research area of highly nonlinear approximation. The general case of this problem proved to be NP hard. It has been shown even the verification of the optimal solution is difficult. However, attempts have been made recently to derive performance bound under the assumption of low coherence [8]. Suboptimal greedy algorithms that converge to a local optimal solution exist and have been used extensively. A family of these algorithms is based on the celebrated matching pursuit method [12].

3.2. The solution: MP, OMP, and OOMP

The matching pursuit (MP) method [12] is a technique to compute adaptive signal representation by iterative selection of base vectors from *a* dictionary. Such a dictionary is usually non-orthogonal. In its original form, this technique does not yield, in each iteration, the linear span of the selected base vectors that best approximates the signal in a minimum distance sense. A later refinement called orthogonal matching pursuit (OMP) [7,13,15] was proposed. However, since OMP selects the base vectors according to the MP prescription, the selection criterion is not optimal in the sense of minimizing the residual of the new approximation.

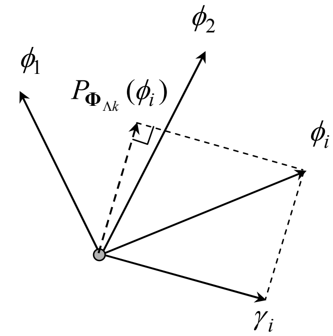


Figure 3. Projection of a base vector ϕ_i into the subspace formed by selected base vectors $\Phi_{\Lambda k}$. The residual vector is $\gamma_i = \phi_i - P_{\Phi_{\Lambda k}}(\phi_i)$.

The requirement of such minimization has led to the optimized orthogonal matching pursuit approach (OOMP) [16]. This algorithm iteratively selects base vectors according to the following procedure:

Suppose that at iteration k the already selected k base vectors are defined by the index set $\Lambda_k = (l_k)_{i=1}^k$. 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.

$$\varepsilon_{k+1} = \min_i \frac{|\langle \gamma_i, \varepsilon_k \rangle|}{\|\gamma_i\|}, \|\gamma_i\| \neq 0, i \in \overline{\Lambda}_k \quad (7)$$

where $\varepsilon_k = \mathbf{x} - P_{\Phi_{\Lambda k}}(\mathbf{x})$ is the approximation error using $\Phi_{\Lambda k}$, and $\gamma_i = \phi_i - P_{\Phi_{\Lambda k}}(\phi_i)$. $\overline{\Lambda}_k$ is the subset of indices that are not selected in the previous iteration k, i.e. $\overline{\Lambda}_k = I - \Lambda_k$. The geometric interpretation of γ_i is illustrated in Figure 3. An effective implementation of this optimization can be achieved by the forward adaptive biorthogonalization technique [1].

3.3. Fast OOMP using the box basis functions

OOMP can be made very efficient for box basis functions. The main computation in Eq. (7) is the evaluation of the projection residual γ_i of each available base vector. Therefore, the key question is how to project a box basis function into the subspace spanned by other box basis functions in $\Phi_{\Lambda k}$. This can be accomplished by computing $\Phi_{\Lambda k}(\Phi_{\Lambda k}^T \Phi_{\Lambda k})^{-1}$

once and for every base vector to be considered, $\Phi_{\Lambda k}^T \phi_i$ is computed very efficiently by counting the overlapping area of ϕ_i with each of the base vector in $\Phi_{\Lambda k}$. By doing so, kd multiplications in computing $\gamma_i = \phi_i - P_{\Phi_{\Lambda k}}(\phi_i)$ are eliminated.

3.4. Coherence

A μ-coherent dictionary D has coherence μ for $0 \le \mu \le 1$, if $|<\phi_1, \phi_2>| \le \mu$ for all distinct $\phi_1, \phi_2 \in D$. 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-coherence basis set is orthogonal. It is possible to revise the OOMP method to enforce the resultant basis set having coherence larger than a threshold τ. This implies that in every iteration of OOMP, if a vector in $\overline{\Lambda}_k$ is below this threshold, it will not be considered in the following iterations. This can dramatically improves the efficiency by avoiding considering basis that can be well approximated by the current basis set $\Phi_{\Lambda k}$ and with low risk of degrading the solution.

4. Applications

4.1. Fast normalized cross correlation

Normalized cross correlation is a popular method for finding 2D patterns in images. An $(2h+1)\times(2w+1)$ template $\mathbf{t}$ is correlated against an image $\mathbf{x}$. At the image location (u, v), the normalized cross correlation is computed as

$$\mathbf{c}(u,v) = \frac{\sum_{i=-h}^{h}\sum_{j=-w}^{w} X(i,j)T(i,j)}{\sqrt{\sum_{i=-h}^{h}\sum_{j=-w}^{w} X(i,j)^2}\sqrt{\sum_{i=-h}^{h}\sum_{j=-w}^{w} T(i,j)^2}} \quad (8)$$

with

$$X(i,j) = \mathbf{x}(u+i, v+j) - \bar{\mathbf{x}}$$
$$T(i,j) = \mathbf{t}(h+i, w+j) - \bar{\mathbf{t}} \quad (9)$$

In [11], a fast algorithm was developed to compute the denominator term

$$\sum_{i=-h}^{h}\sum_{j=-w}^{w}(\mathbf{x}(u+i, v+j) - \bar{\mathbf{x}})^2 \quad (10)$$

This is achieved by observing that this term can be decomposed into three parts: $\sum_{i=-h}^{h}\sum_{j=-w}^{w}\mathbf{x}(u+i,v+j)^2$, $\bar{\mathbf{x}}\sum_{i=-h}^{h}\sum_{j=-w}^{w}\mathbf{x}(u+i,v+j)$, and $(2h+1)(2w+1)\bar{\mathbf{x}}$. The first two terms can be computed efficiently using integral images of the original image and the squared image. To speed up the numerator computation, we can decomposed the template into box basis function so that

$$\mathbf{t} \approx \sum_{i\in\Lambda} \alpha_i \phi_i \ . \quad (11)$$

Then the numerator becomes

$$\mathbf{x}(u,v)\mathbf{t} = \sum_{i\in\Lambda} \alpha_i(\phi_i \mathbf{x}(u,v)) \quad (12)$$

This can be computed using $|\Lambda|$ multiplications and $4|\Lambda|-1$ additions.

4.2. Fast template matching using SSD

When the sum of squared difference (SSD) is used as the error measure for template matching, it is computed at each image location as

$$\begin{aligned}\mathbf{c}(u,v) &= \sum_{i=-h}^{h}\sum_{j=-w}^{w}(\mathbf{x}(u+i,v+j) - \mathbf{t}(h+i,w+j))^2 \\ &= \sum_{i=-h}^{h}\sum_{j=-w}^{w}\mathbf{x}(u+i,v+j)^2 + \sum_{i=-h}^{h}\sum_{j=-w}^{w}\mathbf{t}(h+i,w+j)^2 - \\ &\quad -2\sum_{i=-h}^{h}\sum_{j=-w}^{w}\mathbf{x}(u+i,v+j)\mathbf{t}(h+i,w+j)\end{aligned} \quad (13)$$

The first two terms are constants and the last term can be computed using $|\Lambda|$ multiplications and $4|\Lambda|-1$ additions, as discussed in the previous section.

4.3.Subspace expansion and residual computation

As mentioned in the previous section, the projection of a data vector $\mathbf{x}$ in the subspace Φ_Λ is $\Phi_\Lambda(\Phi_\Lambda^T\Phi_\Lambda)^{-1}\Phi_\Lambda^T\mathbf{x}$. To compute $\Phi_\Lambda^T\mathbf{x}$, $3|\Lambda|$ additions are needed. $(\Phi_\Lambda^T\Phi_\Lambda)^{-1}$ can be pre-computed and the product between $(\Phi_\Lambda^T\Phi_\Lambda)^{-1}$ and $\Phi_\Lambda^T\mathbf{x}$ requires $|\Lambda|^2$ multiplications. The product of Φ_Λ and $(\Phi_\Lambda^T\Phi_\Lambda)^{-1}\Phi_\Lambda^T\mathbf{x}$ can be efficiently computed using $N|\Lambda|$ additions. So total of $|\Lambda|^2$ multiplications and $(N+3)|\Lambda|$ additions are needed, compared to the original $2N|\Lambda|$ multiplications and $2N|\Lambda|$ additions. Similarly, to find the subspace projection residual, additional N additions are needed.

4.4. Recognition in the non-orthogonal space.

It is possible to perform object recognition directly using the projection $\Phi_{\Lambda}^{T}\mathbf{x}$. Therefore, instead of using $d|\Lambda|$ multiplication and $(d-1)|\Lambda|$ additions to compute the projection, only $3|\Lambda|$ additions are needed. The computational saving is significant. The difference between non-orthogonal feature values and the feature values in the same subspace but spanned by orthogonal coordinate axes is a linear transformation. Two separable clusters remain separable under a linear transformation.

5. Implementation and experimental results

5.1. Optimal orthogonal matching pursuit

The OOMP method described in Section 3 was implemented for obtaining optimal non-orthogonal base vectors using both single and multiple input data samples. For face images, 500 frontal view images from the FERET database were used. These images were spatially aligned using an affine transform based on the facial feature points. Each of these aligned images is then cropped and scaled down to 25×20 pixels. Using these 500 data samples, the optimal box base vectors are computed. The first 30 of these vectors are shown in the top image of Figure 1. It can be observed that many of these base vectors are symmetric. The asymmetric ones may be caused by the illumination bias in the relatively small data set. The reconstructions of a single face image using these 30 base vectors are shown in the bottom image of Figure 1. With each additional bases vector, more details are added to the reconstructed image.

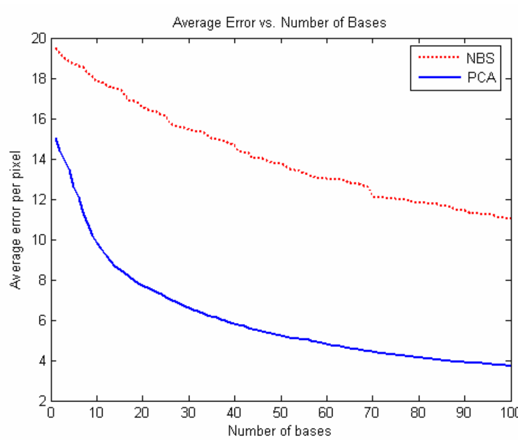
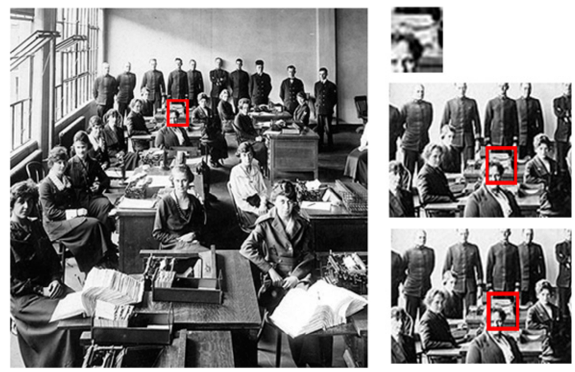


Figure 4. Approximation error as a function of the number of base vectors, both for the PCA and the NBS methods.

5.2. Effectiveness of NBS and PCA

Compared to the principal component analysis method, the proposed NBS method, though far more efficient, is not optimal in terms compactness. We analyzed the curves of approximation error vs. number of base vectors, both for the PCA method and the proposed NBS method. The same 500 face data samples as described in the previous subsection were used. The result is show in Figure 4. It can be observed that PCA uses 10 base vectors to reduce the average error to intensity level 10 per pixel, while the NBS method needs about 70 base vectors.



(a) (b)

Figure 5. (a) The original 480x480 image, (b)-top: randomly selected 25x20 template patch, (b)-middle: the position with the highest normalized correlation score, (b)-bottom: the result using NCC approximated by the proposed NBS method.

5.3. Normalized Cross Correlation using NBS

As described in Section 4.1, to compute the numerator of Eq. (8), the template is first approximated using NBS. To achieve this, the OOMP algorithm is performed to obtain the optimal base vectors. Then, the numerator is computed using Eq. (11-12), with $|\Lambda|$ multiplications and $4|\Lambda|-1$ additions. In a straightforward implementation, $(2h+1)\times(2w+1)$ multiplications and $(2h+1)\times(2w+1)$ additions are needed. To verify this result, experiments on real images were performed. Ten images of varying scenes (indoor, outdoor, from microscope, etc) were used, and ten patches were chosen at random from each to then be located using NCC. In Figure 5, an example is shown. Figure 5(a) is the original image with 480×480 pixels. Figure 5(b) is an image patch in the original image. The size of the image patch is 25×20 pixels. It is approximated using up to 20 NBS base vectors. Figure 5(c) and Figure 5(d) show the normalized correlation matching results using the original image patch and using the NBS approximated template. Both of them locate the correct object position. Figure 6 shows that no more than twelve binary bases were needed in any test case to accurately locate the patch. The run time using the original normalized cross correlation is 9.2 seconds. The run time using the NBS method is 0.81 second. A speedup of 11.4 times is achieved. It should be mentioned that the denominator computation is the same for both methods and is included in the calculation. Therefore the actual speedup of the numerator computation is more than 11.4 times. In addition, further speedup can be achieved by using fewer than 20 base vectors.

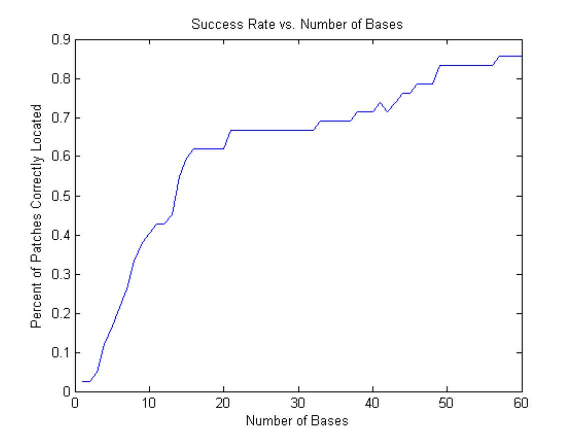


Figure 6. Performance of NCC patch matching using NBS.

5.4. Face recognition

To demonstrate the effectiveness of the proposed non-orthogonal binary basis method, it was implemented to perform face recognition and the result was compared to that of obtained using the Eigenface method [17]. The test database comprises 64 aligned face images in the FERET database. The size of these images is 24x24 pixels. There are total of 768 different people in the 1260 image database, though images of the 64 test individuals were not used in base construction. Some people have multiple images in the dataset. Each image is projected to the subspaces obtained using the PCA and the proposed NBS method. A simple nearest neighbor method is used for finding the best match as the recognition result. The cross validation method is used to obtain the recognition rate. In Figure 7, curves of the recognition rate vs. the number of base vectors are shown for the PCA and the proposed NBS methods. It can be observed that both methods achieve better recognition results with more base vectors. NBS representation is less compact than the PCA representation. Therefore, more NBS base vectors are needed to achieve the same recognition rate. However, the projection process using the NBS method is much more efficient. This is especially useful when the number of input image is large.

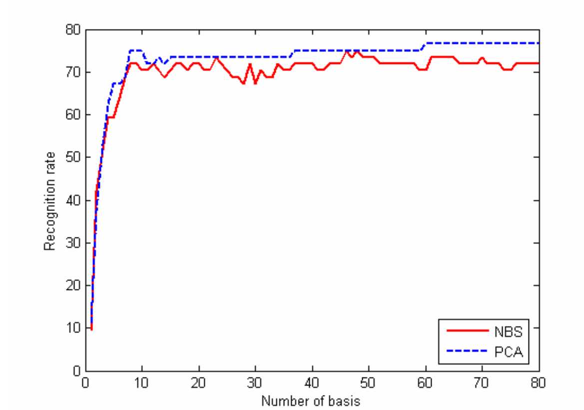


Figure 7. Face recognition using the PCA and the NBS methods.

6. Conclusions

A new method is developed to represent images as linear combinations of non-orthogonal binary base vectors. Using the integral image trick, inner product with these 2D box basis functions can be performed using several additions. This property makes the proposed NBS representation suitable for speed critical applications. This paper demonstrates how a suboptimal set of NBS vectors can be computed using the optimized orthogonal matching pursuit method. The application of the NBS representation in several computer vision algorithms including normalized cross correlation, SSD computation, subspace based object recognition is discussed. The experimental results demonstrated that the NBS subspace is effective in representing object appearance and significantly reduces the computational complexity.

Further investigation of the proposed new approach needs to answer some of the following questions:
(1) How does the coherence requirement affect the quality of the resultant NBS subspace? With very low threshold on coherence, the OOMP will find near-to-orthogonal base vectors very efficiently. However, preliminary experimental results have shown that a greedy algorithm such as OOMP may lead to an incomplete basis set. The relation between the

coherence threshold τ and the approximation quality needs to be carefully studied.

(2) Two types of box basis functions are investigate in this paper. For different applications, however, more possibilities may arise. For example, to handle object motion, a differential box filter can be developed to represent the spatial-temporal content of an image sequence [19].

(3) The effectiveness of the proposed NBS method on different type of objects needs to be studied. Human faces are tested in this paper. More experimental results on vehicles, pedestrians, and other objects will establish the proposed method as a valid alternative to the existing subspace method.

10. References

[1] M. Andrle and L. Rebollo-Neira, "A swapping-based refinement of orthogonal matching pursuit strategies", preprint, 2004.

[2] R. Basri and D. Jacobs, "Lambertian Reflectance and Linear Subspaces," *PAMI*, vol. 25, no. 2, pp. 218-233, 2003.

[3] P. Belhumeur, D. Kriegman, "What is the Set of Images of an Object Under All Possible Lighting Conditions?" *International Journal of Computer Vision*, 28(3), pp. 245-260, 1998.

[4] M. J. Black and A. D. Jepson, "EigenTracking: Robust Matching and Tracking of Articulated Objects Using a View-Based Representation," in *Proc. ECCV*, pp. 329-342, 1996.

[5] K, Briechle and U. D. Hanebeck, "Template matching using fast normalized cross correlation," in *Proceedings of SPIE, V. 4387*, *Optical Pattern Recognition XII*, Orlando, FL. pp. 95-102, 2001.

[6] X. Chen, L. Gu, S. Z. Li, Hong-Jiang Zhan, "Learning representative local features for face detection," *CVPR vol*. 1, pp. 1126-1131, 2001.

[7]G.M. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations", *Conts. Approx.*, Vol 13, 57-98 (1997).

[8] A. C. Gilbert, S. Muthukrishnan, M. Strauss, "Approximation of functions over redundant dictionaries using coherence," in *Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms*, pp. 243 - 252, 2003.

[9] Yacov Hel-Or and Hagit Hel-Or "Real Time Pattern Matching Using Projection Kernels," *ICCV*, *pp*. 1486-1493, 2003.

[10] D.D. Lee and H.S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature* 401 pp. 788—791, 1999.

[11] J.P.Lewis, "Fast Template Maching," *Vision Interface*, pp. 120-123, 1995.

[12] S. Mallat, and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, 41:3397-3415, 1993.

[13] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998.

[14] P. Nillius and J.-O. Eklundh, "Fast block matching with normalized cross-correlation using Walsh transforms," report, ISRN KTH/NA/P--02/11--SE, Sept. 2002.

[15] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad, "Orthogonal matching pursuits: recursive function approximation with applications to wavelet decomposition", in *Proceedings of the 27th Asilomar Conference in Signals, Systems and Computers*, pp. 40-44, 1993.

[16] L. Rebollo-Neira and D. Lowe, "Optimized Orthogonal Matching Pursuit Approach", *IEEE Signal Processing Letters*, Vol(9,4), 137-140 (2002).

[17] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, 1991.

[18] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Dec. 2001.

[19] P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," in *ICCV'03*, pp. 734–741, Nice, France, 2003.