Toward Sparse Coding on Cosine Distance

Jonghyun Choi Hyunjong Cho Jungsuk Kwac† Larry S. Davis

Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 USA
†Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA
Email: {jhchoi,cho,lsd}@umiacs.umd.edu, †kwjusu1@stanford.edu

***Abstract*—Sparse coding is a regularized least squares solution using the L_1 or L_0 constraint, based on the Euclidean distance between original and reconstructed signals with respect to a pre-defined dictionary. The Euclidean distance, however, is not a good metric for many feature descriptors, especially histogram features, *e.g.* many visual features including SIFT, HOG, LBP and Bag-of-visual-words. In contrast, cosine distance is a more appropriate metric for such features. To leverage the benefit of the cosine distance in sparse coding, we formulate a new sparse coding objective function based on approximate cosine distance by constraining a norm of the reconstructed signal to be close to the norm of the original signal. We evaluate our new formulation on three computer vision datasets (UCF101 Action dataset, AR dataset and Extended YaleB dataset) and show improvements over the Euclidean distance based objective.**

***Keywords*—Sparse coding, cosine distance.**

I. Introduction

Sparse coding has been widely used in solving many machine learning and computer vision problems to reduce noisy information of original signals [1]–[4]. An obtained sparse code (representation) is a real valued vector that has a few non-zero coefficients with respect to a set of over-complete bases, called the *dictionary*. Sparse coding usually means to solve a least squares problem (Euclidean distance), with L_0 or L_1 regularizer on the coefficients, as follows:

$$\hat{x} = \arg\min_x \|y - Dx\|_2^2 + \gamma\|x\|_p, \tag{1}$$

where y is the query signal, D is the over-complete dictionary and x is a vector of coefficients for dictionary atoms. $\hat{x}$ is the sparse code obtained by minimizing a loss function (the least square term) with a L_p-norm regularizer on x where p is either 0 or 1. The L_0 constrained problem is usually approximately solved by the various matching pursuit (MP) algorithms [5]. The L_1 constrained problem is a convex problem and can be solved by the LASSO method [6] or its variants with low computational complexity [7].

The Euclidean distance used in the least squares term in (1), however, is not always a good metric for many visual features used in computer vision applications [8], [9]. To be specific, for histogram features (*e.g.* Histograms of Oriented Gradients (HOG) [10], LBP [11], SIFT [12] and Bag-of-words features [13]), cosine (or angular) distance is known as a better metric [9], [14] together with χ^2 distance [15], histogram intersection kernel [8] and Earth Mover's Distance [16]. Cosine distance has two advantages over Euclidean distance; 1) It is scale invariant since it is an angular distance between two vectors. 2) It gives more weight to dimensions having high value (peaks in histogram) while the Euclidean distance weighs all dimensions equally.

In order to take advantage of a better metric for sparse coding with such features, we propose to replace the Euclidean distance with the cosine distance. A straightforward formulation for it can be written as following:

$$\hat{x} = \arg\min_x \left(1 - \frac{y^T Dx}{\|y\|_2\|Dx\|_2}\right) + \gamma\|x\|_p, \tag{2}$$

where $p \in \{0, 1\}$. Due to the $\|Dx\|$ term in the denominator, (2) is not straightforward to solve. Instead, we formulate an approximate cosine distance based sparse coding objective function by a simple modification of the Euclidean distance based L_1 sparse coding objective using a constraint to make Euclidean distance and cosine distance equivalent.

The rest of the paper is organized as follows. The next section reviews related work. Section III,IV,V introduce our new formulation of sparse coding on cosine metric. Section VI presents experimental results on various computer vision datasets and Section VII concludes the paper.

II. Related Work

Sparse coding has been widely used in many fields including signal processing, machine learning and computer vision. Among its computer vision applications, face identification is one of the most popular ones since the work of Wright *et al.* [1]. The goal of face identification is to find the nearest neighbor (NN) of a query image (called the *probe*) in a pre-defined face database (called the *gallery*). They viewed the NN search as finding the dictionary atom that has maximum-valued coefficient among the few non-zero coefficients in the obtained sparse code of a query face. The coefficient values serve as similarity scores between the query image and the dictionary atoms. Sparsity plays the role of making the maximum coefficient more prominent by suppressing others to zero. This approach, called sparse representation-based classification (SRC), is known to be robust to partial corruption of the signal due to occlusions or pixel noise. Due to the noise robustness of SRC, it is widely used in many other computer vision problems such as image classification [2], action recognition [3] and visual tracking [4].

Many previous work using SRC uses raw pixel as a feature. When they use raw pixel, the performance of algorithms is very sensitive to alignment of images. In order to mitigate the alignment problem, local edge information is pooled (*e.g.* histogram based features) or correlated with a Gaussian kernel (*e.g.* Gabor filter) to create a feature descriptor to replace the raw pixels. Recent work, including [17] and [18], showed that local feature descriptors (*e.g.*, Gabor filters and local binary patterns (LBP)) lead to better performance with sparse coding on locally misaligned face datasets.

In the image classification or object categorization literature, sparse coding is used for a better method of encoding

local features into codewords than vector quantization (VQ) on cluster centers (usually obtained by K-means). The main benefits of sparse coding are to assign a local feature the codeword by a weighted combination of a few meaningful candidates (soft assignment) or to construct a better codeword set [2], [19], [20].

The fact that different feature descriptors need to be paired with appropriate metrics for the best performance is illustrated in many previous work including [14], [21]–[23]. In particular, Ma *et al.* formulated a better discriminative analysis function by adopting a correlation based metric, which is a cosine distance [23]. The correlation based discriminative analysis, called CDA, performs better than the traditional Euclidean distance based linear discriminative analysis (LDA) on both the UCI machine learning dataset and the ORL face recognition dataset.

We formulate an objective function to leverage the benefit of the cosine metric in a sparse coding framework for various kinds of visual feature descriptors. The new formulation is a simple modification of the conventional Euclidean distance based sparse coding formulation and can be solved exactly and efficiently.

III. Formulation

The straightforward formulation of cosine distance based sparse coding (2) is not trivial to solve. To make the formulation easy to solve, we take advantage of the fact that the cosine distance is equivalent to the square of the Euclidean distance if the norms of both y and Dx are 1 as the following simple theorem [24]:

Theorem 1. *If the norms of two vectors are 1, the square of the Euclidean distance is proportional to the cosine distance (with a factor of 2).*

Proof: Suppose the original signal is y and the reconstructed signal against dictionary D is Dx, where x is the sparse code. If the norms of y and Dx are both 1, then the square of the Euclidean distance between the two vectors is proportional to the cosine distance:

$$\begin{aligned} ||y - Dx||_2^2 &= y^T y + (Dx)^T (Dx) - 2y^T Dx \\ &= 1 + 1 - 2\frac{y^T Dx}{1} = 2 - 2\frac{y^T Dx}{\|y\|_2 \cdot \|Dx\|_2} \\ &= 2 \cdot (1 - \text{cosine similarity}(y, Dx)) \\ &= 2 \cdot \text{cosine distance}(y, Dx) \end{aligned} \tag{3}$$

■

Based on the observation, we formulate an approximate cosine distance based sparse coding objective function by normalizing the original signal y by its L_2-norm and adding a term that forces the norm of the reconstructed signal Dx to be 1 to the Euclidean based sparse coding objective function. This can be written as:

$$\begin{aligned} \min_x ||\hat{y} - Dx||_2^2 + \alpha \left|1 - ||Dx||_2^2\right| + \gamma||x||_1, \\ s.t.\ 0 < \alpha < 1, \\ \|\hat{y}\|^2 = 1, \end{aligned} \tag{4}$$

where $\hat{y}$ is y normalized by its L_2-norm and α and γ are the hyper-parameters to balance the term forcing $\|Dx\|_2^2$ to be close to 1 and L_1 regularizer, respectively.

Normalizing the signal y, however, may reduce classification accuracy in the SRC framework when the magnitude of the signal itself contains important information [25]. (4), however, requires y to be normalized thus forces to lose such information. So, we extend Theorem 1 to make another equivalence between Euclidean and cosine distances without requiring normalizing y as following:

Theorem 2. *If the norms of two vectors are both n, then the square of the Euclidean distance is proportional to their cosine distance (with a factor of $2n^2$).*

Proof: The proof follows the proof of Theorem 1 with $\|y\|^2 = \|Dx\|^2 = n^2$. ■

Without normalizing y, we can formulate an objective function for approximate cosine distance based sparse coding by constraining the norm of the reconstructed signal Dx to be close to the norm of y. The formulation can be rewritten as follows:

$$\begin{aligned} \min_x ||y - Dx||_2^2 + \alpha \left|||y||_2^2 - ||Dx||_2^2\right| + \gamma||x||_1, \\ s.t.\ 0 < \alpha < 1. \end{aligned} \tag{5}$$

A. Relaxation

The (5), however, is not differentiable due to the absolute operator in the added term and make the gradient descent stuck at the peak point. We then relax it based on the inherent positivity of the term.

1) Relaxation by Positivity Constraint: We relax objective function by removing the modulus term by constraining the α to make $\|y\|_2^2 - \|Dx\|_2^2 > 0$ as follows:

$$\begin{aligned} \min_x ||y - Dx||_2^2 + \alpha(\|y\|_2^2 - \|Dx\|_2^2) + \gamma||x||_1, \\ s.t.\ 0 < \alpha < 1. \end{aligned} \tag{6}$$

The relaxation is justified by considering the two cases of whether or not D is orthogonal, which lead to range constraints of α:

*a) D **is orthogonal**:* If D is column-wise orthogonal, the relaxed objective function (6) can be solved in a closed form as:

$$\hat{x} = \frac{1}{1-\alpha}\text{sign}(D^T y)(D^T D)^{-1}(|D^T y| - \frac{\gamma}{2})^+. \tag{7}$$

Thus,

$$\|D\hat{x}\|^2 = \frac{1}{(1-\alpha)^2}(|D^T y| - \frac{\gamma}{2})^{+T}(D^T D)^{-1}(|D^T y| - \frac{\gamma}{2})^+. \tag{8}$$

where $(\cdot)^+ = \max(0, \cdot)$. Therefore, given D and y, we can obtain a range on α (upper bound of α, the lower bound is 0) for which $\|D\hat{x}\|^2 < \|y\|^2$ with (8). When we set α within the obtained range, this relaxation holds.

*b) D **is not orthogonal**:* If D is not column-wise orthogonal, the final solution cannot be obtained in closed form. But still we can obtain a tighter upper bound on α than the case when D is orthogonal because $\|Dx\|^2$ with the L_1 penalty is smaller than $\|Dx\|^2$ without that penalty.

Excluding the L_1 penalty in (6), we can obtain a solution of a modified ordinary least square (OLS) problem, $\hat{x}'$, as:

$$\hat{x}' = \frac{1}{1-\alpha}(D^T D)^{-1}(D^T y). \tag{9}$$

Thus, $\|D\hat{x}'\|^2 = \frac{1}{(1-\alpha)^2}(D^T y)^T (D^T D)^{-1}(D^T y)$. We can obtain the tighter range of α (tighter upper bound of α, the lower bound is 0) for which ($\|D\hat{x}\| <$) $\|D\hat{x}'\| < \|y\|^2$. Therefore, if we set α within the tighter range, this relaxation again holds.

2) Convexity: The solutions of the relaxed objective function can be easily obtained by gradient descent algorithms if it is convex. We can ensure the convexity of (6) by forcing the hyper-parameter α to be greater than zero and less than 1. The convexity of the new objective function (6) with respect to x is easily seen from the expanded equation with the constraint, $0 < \alpha < 1$:

$$\begin{aligned} &y^T y + x^T D^T D x - 2y^T D x + \alpha(y^T y - x^T D^T D x) + \gamma||x||_1 \\ &= (1-\alpha)x^T D^T D x - 2y^T D x + (1+\alpha)y^T y + \gamma\|x\|_1 \end{aligned} \tag{10}$$

In summary, if α is within the range that satisfies the convexity and positivity relaxation condition, (6) is theoretically justified and solved by a gradient descent procedure. Usually the bound for α is very tight ($\sim 10^{-3}$), which may limit the amount of changes of solution path and the improvement involved.

IV. Effect of the New Term

At first glance, the new term, $\alpha(\|y\|_2^2 - \|Dx\|_2^2)$, seems to be redudant; since the Euclidean distance based sparse coding objective function is already minimizing the error term, $\|y - Dx\|^2$, the distance between their norms is also minimized by the Reverse Triangle Inequality. In other words, simply forcing less penalty on the L_1 regularization might achieve the goal of reducing the gap between two norms.

However, enforcing the new term is different from having a lower L_1 penalty. Since the new term is a lower bound on the error term $\|y - Dx\|^2$ by the reverse triangle inequality, our objective function can reduce the gap between their norms without changing the value of the error term.

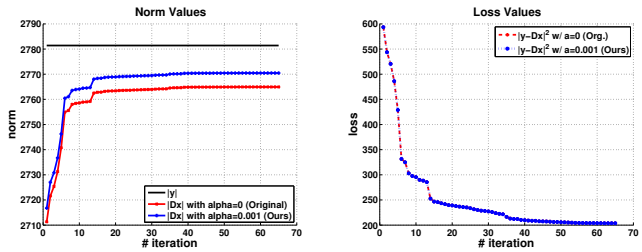

Fig. 1. An example of growing of $|Dx|$ with or without the new term that enforces the norm of $|Dx|$ to be closer to norm of $|y|$ (Left). 'Loss' in the right figure means $\|y - Dx\|^2$ (Right). The red curve is generative by the original Euclidean distance based sparse coding formulation ($\alpha = 0$). The blue curve is generated by our new objective function with $\alpha = 0.001$. (must viewed in color)

Fig. 1 shows an example of $|Dx|$ as a function of iteration with and without the new term. The figure clearly shows that the new term forces $|Dx|$ to be closer to $|y|$ than the Euclidean distance based sparse coding objective function, without changing the loss value. Eventually this behavior will change the solution path by not changing the error value itself but its lower bound.

The experimental validations also imply that the new term plays a role to change solution path to a favorable direction (towards cosine distance based sparse coding); leading better classification accuracy.

V. Modified Feature-Sign (FS) Algorithm

With the constraint on α, we can efficiently solve the optimization problem with an analytic solution of the quadratic program (10) at each iteration just as the original FS algorithm does. The intermediate solution at each iteration, $\hat{x}_{new}$, is as follows:

$$\hat{x}_{new} = \frac{1}{1-\alpha}(\hat{D}^T \hat{D})^{-1}(\hat{D}^T y - \gamma, \frac{\hat{\theta}}{2}), \tag{11}$$

where $\hat{D}$ is a sub-matrix of D with bases of the active set in the current iteration [7]. The procedure to obtain $\hat{x}_{new}$ is as follows. We define a sub-matrix of D, denoted by $\hat{D}$, which consists of bases of active sets only. On the assumption that $\hat{D}^T \hat{D}$ is invertible, the $\hat{x}_{new}$ can be obtained by taking derivative of (10) as following:

$$\begin{aligned} &\frac{\partial}{\partial x}\left(||\tilde{y} - \hat{D}x||_2^2 + \alpha(\|y\|_2^2 - ||\hat{D}x||_2^2) + \gamma||x||_1\right) \\ &= \frac{\partial}{\partial x}\left(||\tilde{y} - \hat{D}x||_2^2 + \alpha(\|y\|_2^2 - ||\hat{D}x||_2^2) + \gamma\hat{\theta}^T x\right) \\ &= -2\hat{D}^T\tilde{y} + 2\hat{D}^T\hat{D}\hat{x} + \gamma\hat{\theta} - 2\alpha\hat{D}^T\hat{D}\hat{x} \\ &\rightarrow \hat{x}_{new} = \arg_x\left(-2\hat{D}^T\tilde{y} + 2(1-\alpha)\hat{D}^T\hat{D}\hat{x} + \gamma\hat{\theta} = 0\right) \\ &\rightarrow 2(1-\alpha)\hat{D}^T\hat{D}\hat{x} = 2\hat{D}^T\tilde{y} - \gamma\hat{\theta} \\ &\therefore \hat{x}_{new} = \frac{1}{1-\alpha}(\hat{D}^T\hat{D})^{-1}(\hat{D}^T\tilde{y} - \gamma\frac{\hat{\theta}}{2}). \end{aligned} \tag{12}$$

The modified FS algorithm leads to a new solution path that is different from the one obtained by the original FS algorithm. The differences are two folds. First, the new term changes the criterion to select the active set, so that the initial active set is different from that of the original FS algorithm. Second, in the iterative feature-sign steps to update the active set, $\hat{x}_{new}$ is scaled-up by $\frac{1}{1-\alpha}$ ($\frac{1}{1-\alpha} > 1$, since $0 < \alpha < 1$). Thus the subsequent line search algorithm may find a new lowest value to update $\hat{x}$, which in general is different from the one found in the original FS algorithm. Therefore, the new objective function yields a different solution from the original solution, which is a closer solution to the one that would be obtained by the cosine distance based sparse coding formulation.

A. Convergence

The modified FS algorithm converges in a finite number of steps just as the original FS algorithm does. The convergence of the original FS algorithm is proved by Lemma 3.1, Lemma 3.2 and Theorem 3.3 in [7]. The modified objective function shown in Eq.(10) is identical to the original sparse coding formulation except for the additional factor of $(1-\alpha)$ in the term quadratic in x. Since the factor does not affect any of the properties required for the convergence proof of the original

FS algorithm (convexity of objective function and finite size of active set), the convergence proof of our modified FS algorithm simply follows that of the original FS algorithm.

VI. Experimental Results

We evaluate our formulation on three visual recognition datasets: UCF101 action dataset, Extended YaleB face dataset and AR face dataset. We compare the recognition accuracies obtained by our formulation to those by the Euclidean distance based sparse coding [7]. All hyper-parameters are empirically determined as the best performing values after grid-searches.

A. Action Recognition

Action recognition can be formulated as the NN search of action feature descriptors using the SRC approach [26], [27]. We use the SRC approach for action recognition to compare the accuracy by our formulation and the conventional one. For better performance, we use dictionaries learned by K-SVD [28] and LC-KSVD2 [29].

1) UCF101 Dataset: It contains 101 action classes with 13,320 clips from 27 hours of YouTube video footages. The clips are recorded in realistic environments with many variations including camera motion, cluttered background, various lighting, occlusion, low quality imaging and *etc.* It is currently the largest and one of the most challenging datasets of its kind [30].

2) Feature Details: For the feature descriptor, we use SIFT [12], Space-time interest points (STIP) [31] and dense trajectory features (DTF) [32], which are a standard feature-set provided in UCF101. In particular, the DTF feature represents a clip by computing Histogram of Oriented Gradient (HOG), Histograms of optical Flow (HOF), motion boundary histograms (MBH) and trajectory descriptors along the dense motion trajectories.

Each of the six descriptors (SIFT, STIP and four DTF descriptors) is aggregated by a bag-of-words model with 4,000 visual words obtained by K-means. We normalize each feature descriptor by its L_2-norm and concatenate them to form an action descriptor for each clip. The concatenated feature descriptor of a clip is a high dimensional vector (4,000 dim.$\sim$24,000 dim.), thus PCA is performed to reduce the feature dimension. Please refer to UCF101 dataset page for more details of feature descriptors[1].

3) Comparison: We compare recognition accuracies obtained by our formulation and Euclidean-sparse coding formulation on the UCF101 action dataset [30]. Table I summarize the recognition accuracies.

Note that the number of classes is large (101) and each class is challenging, overall 2.2% (Set2 on K-SVD) improvement is significant. For detailed analysis, we plot class-wise improvement on each dictionary learned by K-SVD and LC-KSVD2 summarized in Figure 2 and Figure 3, respectively. In many classes, the improvement is significant (up to 13%). However, there are also a few classes whose accuracies are decreased by our method.

The current best result on the dataset is 85.9%, which uses better and more features including MBH, HOG, HOF, dense trajectory and improved trajectory and encodes it with Fisher Vector [33]. Only with provided features (HOG, HOF, MBH, and trajectory descriptors along the dense motion trajectories with standard bag-of-words model), our result is the best reported one, outperforming previous result (65.9%[2]) by 3.0%.

[1] http://crcv.ucf.edu/data/UCF101.php

TABLE I. Average Recognition Accuracy (%) on UCF101 Dataset. We compare our method (Ours) to the conventional Euclidean distance based sparse coding (EUC-Sparse) on dictionaries learned by K-SVD and LC-KSVD2.

Methods	Set1	Set2	Set3	Avg
on a dictionary by K-SVD				
EUC-Sparse	66.1	66.1	69.2	67.1
Ours	**68.0**	**68.3**	**70.4**	**68.9**
on a dictionary by LC-KSVD2				
EUC-Sparse	65.3	65.9	67.7	66.3
Ours	**67.5**	**67.8**	**69.1**	**68.1**

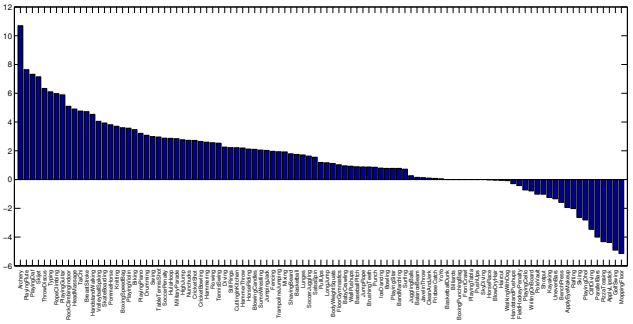

Fig. 2. Class-wise Improvement (%) by our formulation over an Euclidean-Sparse Coding (K-SVD-Avg)

The action categories can be partitioned by super-categories such as 'Human-Object Interaction', 'Body Only Action', 'Human-Human Interaction', 'Playing Instrument' and 'Sports'. Especially, in classes that belong to 'Human-Human interaction', our method outperforms the Euclidean-based formulation in 4 out of 5 classes. In classes of 'Body Only Action' and 'Playing Instrument', our method always outperforms or performs on par the conventional sparse coding objective. Qualitatively, classes of these two super-categories contain the actions that are less abrupt than 'Human Object Interaction' and 'Sports' that contain more abrupt motions. Since our objective may be less sensitive to low frequency repetitive patterns in feature than Euclidean objective when two features are not correlated in angle, we guess that the repetitive patterns of the feature dimensions in the classes of the two super-categories may improve the accuracy.

B. Face Identification

We also compare our formulation to the conventional one in the SRC framework for face identification [1]. We compare the rank-1 recognition accuracies of both formulations on two datasets, Extended YaleB and AR dataset.

1) Extended YaleB Dataset: It contains 2,414 frontal face images of 38 subjects (about 64 images/subject) [34]. Images are well aligned and cropped to 168(W)$\times$192(H) pixels. There are serious illumination and expression variations across the

[2] 12th ranked results that use the same setting in 2013 THUMOS Workshop http://crcv.ucf.edu/ICCV13-Action-Workshop/results.html

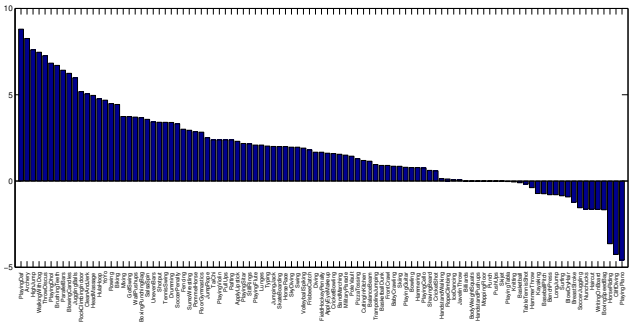

Fig. 3. Class-wise Improvement (%) by our formulation over an Euclidean-Sparse Coding (LC-KSVD2-Avg)

images. Following the standard experimental protocol suggested [34], we randomly select half of the images per subject as a training set and the other half as a testing set.[3]

a) Feature: Our formulation would work differently with different types of visual descriptors. In Table. II, we elaborate the results on various feature descriptors including Randomface [1], HOG [10], LBP [35], Gabor [36] and SIFT [12]. HOG, LBP and SIFT are histogram features while Gabor and Randomface are not. For the 'Randomface' feature, we generate and use a 504 dimensional random projection matrix (504×*Number of Pixels*) whose rows are randomly generated row-vectors from the standard Gaussian distribution (zero mean, unit variance) and normalized by their L_2-norm. We extract HOG features with five bins for each sub-window of size 10×10 pixels in a non-overlapping manner. For LBP, we use 30×30 pixel non-overlapping sub-windows for histogram pooling. For Gabor filters, we use eight orientations and five scales and extract the response every 30 pixels. We extract the SIFT descriptor within 20×20 pixel windows at every 55 pixels.

b) Comparison: We compare the rank-1 recognition accuracy (%) of the algorithms based on the SRC framework. Since the performance of SRC depends on the representativeness of the dictionary, we use both a plain dictionary that uses all training samples as dictionary atoms and a dictionary discriminatively learned by LC-KSVD2. For the parameter set of LC-KSVD2, we use the ones that the authors suggest: $\alpha = 16,\ \beta = 4,$ and $T = 30$.

TABLE II. FACE IDENTIFICATION ACCURACY ON THE EXTENDED YALEB DATASET BY SRC [1] APPROACH. WE USE A PLAIN DICTIONARY AND A DICTIONARY LEARNED BY LC-KSVD2. EUC-SC AND OURS STAND FOR EUCLIDEAN DISTANCE BASED SPARSE CODING OBJECTIVE FUNCTION AND OURS, RESPECTIVELY.

Feature	Accuracy (%)			
	Plain Dict.		LC-KSVD2	
	EUC-SC	**Ours**	EUC-SC	**Ours**
Randomface	92.6	**92.7**	94.3	94.3
HOG	95.4	**95.5**	98.0	**98.2**
LBP	87.1	**87.7**	97.8	**98.4**
Gabor	84.0	**84.3**	92.6	92.6
SIFT	96.2	**96.3**	99.8	**99.9**

Unlike the UCF101 action dataset [30], the accuracy on this dataset is already saturated. Even though the baseline is high, we notice that the proposed approach (Ours) further improves the accuracy. In many feature descriptors, 'Ours' improves the accuracy, although for Randomface and Gabor (not histogram features) with the dictionary learned by LC-KSVD2, the accuracy remains the same. Note that the SIFT descriptor with our formulation achieves the best performance of 99.9% accuracy. The best performing hyper-parameter pair for the SIFT descriptor is $\alpha = 0.004$ and $\gamma = 3$.

2) AR Dataset: It contains 2,600 frontal face images of 100 subjects (50 males and 50 females) so each subject has 26 images [37]. We use the standard cropped-AR dataset whose image size is 165(W)×120(H) pixels. There are illumination and expression variations and occlusion across the images. Following the same experimental protocol used in previous work, we randomly select half of the images per subject as a training set and the other half as a testing set and average the performance of several iterations.[4]

a) Feature: We use the same set of features used in the experiments on Extended YaleB dataset with different sub-window configurations due to the size of images. We extract HOG features with 5 bins for each 10×10 pixel non-overlapping sub-windows. For LBP, we use 25×25 pixel non-overlapping sub-windows for histogram pooling. For Gabor filters, we use eight orientations and five scales and extract the response every 25 pixels. The SIFT descriptors are extracted at every 35 pixels within 20×20 pixel windows. The 504 dimensional random matrix for Randomfaces is also generated from the standard Gaussian distribution.

b) Comparison: We also compare the rank-1 recognition accuracies (%) of the algorithms with different feature descriptors on both plain dictionary and dictionary learned by LC-KSVD2. The results are summarized in Table III. Similar to the experiments on Extended YaleB dataset, the accuracy on this dataset is even more saturated especially with the dictionary learned by LC-KSVD2. Nonetheless, our formulation (Ours) further improves face identification accuracy over the conventional formulation in many cases. Notably, the SIFT descriptor with our formulation achieves the best performance of 100% as shown in Table III. The best performing hyper-parameter pair for SIFT descriptor is $\alpha = 0.001$ and $\gamma = 3$.

TABLE III. FACE IDENTIFICATION ACCURACY ON THE AR DATASET BY SRC APPROACH [1]. WE USE 20 IMAGES PER SUBJECT AS A TRAINING SET (DICTIONARY ATOM), TESTING WITH 6 IMAGES PER SUBJECT.

Feature	Accuracy (%)			
	Plain Dict.		LC-KSVD2	
	EUC-SC	**Ours**	EUC-SC	**Ours**
Randomface	66.3	66.3	94.3	94.3
HOG	84.8	**86.0**	99.7	99.7
LBP	89.2	**89.5**	99.7	**99.8**
Gabor	**94.7**	94.2	99.8	**100**
SIFT	91.2	**92.2**	99.8	**100**

3) Comparison to the state of the art: We compare our results to the state-of-the-art results on both datasets in Table IV. In both datasets, the SIFT descriptor with our formulation achieves the state-of-the-art performance of 99.9% and 100% accuracy, respectively.

VII. DISCUSSION AND CONCLUSION

We proposed a new formulation of sparse coding based on the approximate cosine distance by modifying the Euclidean distance based sparse coding objective function. The new

[3]Our random split will be provided in our website for future comparison

[4]Our random split will be provided in our website for future comparison

TABLE IV. COMPARATIVE FACE IDENTIFICATION ACCURACY ON THE EXTENDED YALE B AND AR DATASET. **OURS** REFERS TO OUR BEST RESULT (WITH SIFT DESCRIPTOR).

Approach	eYaleB	AR
	Acc.(%)	Acc.(%)
SRC (Best) [1]	99.0	97.5
LLC (Best) [38]	96.7	88.7
LC-KSVD2 (Best) [29]	99.0	97.8
Ours	**99.9**	**100**

formulation is convex and easy to solve by modifying the efficient version of the LASSO method called the feature-sign (FS) algorithm [7]. The experimental results show that our formulation yields better performance than the Euclidean distance based sparse coding formulation in three datasets: UCF101, Extended YaleB and AR dataset.

The magnitude of α determines the amount of perturbation in the solution path of the original FS algorithm. With the theoretical bound of α, we can change the path of the least square solution without changing the value of the error term $\|y - Dx\|^2$ and this leads to better recognition accuracies.

As future work, we could formulate sparse coding objective functions based on different metrics such as histogram intersection kernel or χ^2-distance.

ACKNOWLEDGMENT

This work is partially supported by MURI from the Office of Naval Research under the Grant N00014-10-1-0934. The first author thanks to Dr. Zhe Lin at Adobe Research and Prof. Bohyung Han in POSTECH for initial discussions.

REFERENCES

[1] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust Face Recognition via Sparse Representation," *IEEE Trans. PAMI*, vol. 31, no. 2, pp. 210–227, 2009.

[2] A. Quattoni, M. Collins, and T. Darrell, "Transfer Learning for Image Classification with Sparse Prototype Representations," in *CVPR*, 2008.

[3] Q. Qiu, Z. Jiang, and R. Chellappa, "Sparse Dictionary-based Representation and Recognition of Action Attributes," in *ICCV*, 2011.

[4] B. Liu, J. Huang, L. Yang, and C. Kulikowski, "Robust tracking using local sparse appearance model and K-Selection," in *CVPR*, 2011.

[5] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal Matching Pursuit : recursive function approximation with application to wavelet decomposition," in *Asilomar Conf. on Signals, Systems and Computer*, 1993.

[6] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*, 2nd ed., 2003.

[7] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *NIPS*, 2007, pp. 801–808.

[8] J. Wu and J. M. Rehg, "Beyond the Euclidean distance: Creating effective visual codebooks using the histogram intersection kernel," in *ICCV*, 2009.

[9] S. Yan, H. Wang, X. Tang, and T. Huang, "Exploring Feature Descriptors for Face Recognition," in *ICASSP*, vol. 1, 2007.

[10] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *CVPR*, 2005.

[11] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *PAMI*, vol. 24, no. 7, pp. 971–987, 2002.

[12] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *IJCV*, vol. 2, no. 60, pp. 91–110, 2004.

[13] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," in *CVPR*, 2006.

[14] H. V. Nguyen and L. Bai, "Cosine Similarity Metric Learning for Face Verification," in *ACCV*, 2010.

[15] D. A. Kumar and J. Esther, "Comparative Study on CBIR based by Color Histogram, Gabor and Wavelet Transform," *International Journal of Computer Applications*, vol. 17, no. 3, 2011.

[16] Y. Rubner, C. Tomasi, and L. J. Guibas, "A Metric for Distributions with Applications to Image Databases," in *ICCV*, 1998.

[17] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with gabor occlusion dictionary," in *ECCV*, Berlin, Heidelberg, 2010, pp. 448–461.

[18] C. H. Chan and J. Kittler, "Sparse representation of (Multiscale) histograms for face recognition robust to registration and illumination problems," in *ICIP*, 2010, pp. 2441–2444.

[19] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *CVPR*, 2009, pp. 1794–1801.

[20] S. Gao, L.-T. Chia, and I. W. H. Tsang, "Multi-layer group sparse coding-For concurrent image classification and annotation," in *CVPR*, 2011, pp. 2809–2816.

[21] B. ji Zou and M. P. Umugwaneza, "Shape-Based Trademark Retrieval Using Cosine Distance Method," in *Intelligent Systems Design and Applications, 2008. ISDA '08. Eighth International Conference on*, vol. 2, 2008, pp. 498–504.

[22] L. Zhang, Y. Zhang, J. Tang, K. Lu, and Q. Tian, "Binary Code Ranking with Weighted Hamming Distance," in *CVPR*, 2013.

[23] Y. Ma, S. Lao, E. Takikawa, and M. Kawade, "Discriminant analysis in correlation similarity measure space," in *ICML*, New York, NY, USA, 2007, pp. 577–584.

[24] D. Sun, C. H. Q. Ding, B. Luo, and J. Tang, "Angular Decomposition," in *IJCAI*, T. Walsh, Ed., 2011, pp. 1505–1510.

[25] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation." 2003.

[26] T. Guha and R. K. Ward, "Learning Sparse Representations for Human Action Recognition," *IEEE Trans. on PAMI*, vol. 34, no. 8, pp. 1576–1588, 2012.

[27] C. Yuan, W. Hu, G. Tian, S. Yang, and H. Wang, "Multi-task Sparse Learning with Beta Process Prior for Action Recognition," in *CVPR*, 2013, pp. 423–429.

[28] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311–4322, 2006.

[29] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a Discriminative Dictionary for Sparse Coding via Label Consistent K-SVD," in *CVPR*, 2011.

[30] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild," Tech. Rep., 2012.

[31] H. Wang, M. M. Ullah, A. Kl"aser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *BMVC*, 2009.

[32] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *CVPR*, 2011.

[33] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification," in *ECCV*, ser. ., ., Ed., vol. 6314, ., 2010, pp. 143–156.

[34] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. on PAMI*, vol. 23, no. 6, pp. 643–660, 2001.

[35] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE T. PAMI*, vol. 28, no. 12, pp. 2037–2041, 2006.

[36] L. Shen and L. Bai, "A review on Gabor wavelets for face recognition," *Pattern Anal. Appl.*, vol. 9, pp. 273–292, 2006.

[37] A. M. Martinez and R. Benavente, "The AR Face Database," Tech. Rep., 1998.

[38] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained Linear Coding for Image Classification," in *CVPR*, 2010.