# LGE-KSVD: Flexible Dictionary Learning for Optimized Sparse Representation Classification

Raymond Ptucha
Rochester Institute of Technology
Rochester, NY, USA
rwpeec@rit.edu

Andreas Savakis
Rochester Institute of Technology
Rochester, NY, USA
andreas.savakis@rit.edu

## Abstract

*Sparse representations have successfully been exploited for the development of highly accurate classifiers. Unfortunately, these classifiers are computationally intensive and subject to the adverse effects of coefficient contamination, where for example variations in pose may affect identity and expression recognition. We propose a technique, called LGE-KSVD, that addresses both problems and attains state-of-the-art results for face and gesture classification problems. Specifically, LGE-KSVD utilizes variants of Linear extension of Graph Embedding to optimize K-SVD, an iterative technique for small yet overcomplete dictionary learning. The dimensionality reduction matrix, sparse representation dictionary, sparse coefficients, and sparsity-based linear classifier are jointly learned through LGE-KSVD. The atom optimization process is redefined to have variable support using graph embedding techniques to produce a more flexible and elegant dictionary learning algorithm. Results are obtained for a wide variety of facial and activity recognition problems to demonstrate the robustness of the proposed method.*

## 1. Introduction

The notion of Sparse Representations (SRs), or finding sparse solutions to underdetermined systems, has found applications in a variety of scientific fields including computer vision. An image $x_i$ is efficiently represented by sparse linear coefficients from a dictionary $\Phi$ of overcomplete basis functions, where $\Phi \in \mathbf{R}^{Dxn}$. SR solves for coefficients $a \in \mathbf{R}^n$ that satisfy the $\ell^1$ minimization problem $\hat{x}=\Phi a$. It has been shown that under typical conditions, the minimal solution is the sparsest one [1, 2]. There have been several studies optimizing both the $\ell^1$ minimization [3, 4] as well as the selection of dictionary elements [5, 6].

Although the SR framework is designed for reconstruction purposes, it has been adapted successfully for classification problems. In the influential facial recognition work of Wright *et al*. [7, 8], the *a* coefficients are passed into a minimum reconstruction error classifier. In this framework, the dominant signal always prevails, but it could produce some unintended effects. For example, when trying to extract facial identity, pose variation may contaminate or even dominate the sparse coefficients. This coefficient contamination is unfortunate yet important, as it has been shown that images of a single person under multiple poses exhibit greater variation than images of different people at a single pose [9].

Wright *et al*. [7, 8] used random projections to make the coefficient learning computationally tractable. Tzimiropoulos *et al*. [10] and Zafeiriou and Petrou [11] used Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) respectively, along with SR techniques based on [8] to demonstrate further computational and accuracy improvements. The work in [11] struggled with coefficient contamination, noting that applying Wright's framework is not a straightforward process because the facial identity of the person is often confused with facial expression. Ptucha *et al*. [12] addressed the coefficient contamination problem by preprocessing the data with supervised manifold learning. Similarly to subspace clustering [13], supervision in manifold learning encourages clustering of sample images in accordance with their classification labels.

Given *n* data samples, $x_1, x_2, \ldots, x_n$, each sample $x_i \in \mathbf{R}^D$, stored in matrix $X$, $X \in \mathbf{R}^{Dxn}$ and $D < n$, PCA and LDA are effective techniques for obtaining a lower dimensional representation of $X$. During PCA or LDA, the top $d$ eigenvectors are used in projection matrix $U$ such that the low dimensional representation of $X$ is $Y^T=X^TU$, $Y \in \mathbf{R}^{dxn}$. Although these linear dimensionality reduction techniques produce meaningful results, we wish to find an alternate representation in a low dimension $d$, such that $d<<D$. Further, the underlying linearity assumption of PCA and LDA may be limiting when modeling the behavior of complex imagery such as face representations.

Manifold learning techniques reduce the dimensionality of input data by identifying a non-linear lower dimensional space where the data resides [14, 15]. In order to support the extension of the manifold model to new examples, linearized techniques called Linear extension of Graph Embedding (LGE) [16], solve a linear approximation of the non-linear object.

Research in manifold learning has influenced the SR

community and vice-versa. The Sparsity Preserving Projections [17] approach replaces the adjacency matrix used in LGE techniques with SR sparse coefficients. Mairal *et al*. [18] injects a multiclass logistic regression term to the sparse energy function to make dictionary learning have both reconstructive and discriminative properties. Discriminative Sparse Coding [19] uses sparse coefficients in an LDA framework. Graph Regularized Sparse Coding [20] adds the LGE objective function on sparse coefficients to the traditional $\ell^1$ sparse objective function as it jointly learns the sparse coefficients and dictionary terms. What lacks is a single method that optimizes dimensionality reduction, SRs, and classification learning concepts into a single framework.

This paper solidifies the relationship between manifold learning and SRs by proposing an elegant solution to jointly optimize dimensionality reduction, sparse dictionary learning, and sparsity-based classification. Distinguishing features of our novel framework include:

- Utilization of a semi-supervised Linear extension of Graph Embedding to minimize coefficient contamination and reduce compute intensity.
- Iterative procedure for optimizing dimensionality reduction matrix in conjunction with dictionary atoms and coefficients.
- Modification of K-SVD algorithm to remove the fixed atom support in the iterative atom selection process.
- Simultaneous creation of a sparse classifier.

We contrast our technique, which we call LGE-KSVD, to other recently introduced techniques across a wide variety of facial and activity classification problems.

The rest of this paper is organized as follows. Sections 2 and 3 introduce the necessary principles of manifold learning and sparse signal representation. Section 4 describes how to combine the two concepts into the LGE-KSVD framework. Section 5 presents experimental results. Section 6 summarizes with conclusions.

## 2. Linear Extension of Graph Embedding

High dimensional feature spaces used in computer vision are not only inefficient and computationally intensive, but the sheer number of dimensions often masks the discriminative signal embedded in the data. For samples $x_i \in \mathbf{R}^D$ we seek a low dimensional representation yielding $y_i \in \mathbf{R}^d$, where $d << D$. For linear models, e.g. PCA or LDA, $y_i^T = x_i^T U$, where $U$ is a $D \times d$ projection matrix. Alternatively, the high dimensional feature space can be parameterized by a lower dimensional embedded manifold discovered using manifold learning [14, 15].

During manifold learning a fully connected graph of the input space is constructed, where each of the $n$ input samples or nodes is connected to all other ($n$-1) input samples with a weight, $0 \le w_{ij} \le 1$, $i,j = 1 \ldots n$. The resulting connection matrix $W$ is called the adjacency matrix and the connections or weights $w_{ij}$ can be solved several ways. For example, $w_{ij}$ is set to 1 if $x_i$ is amongst the $z$ nearest neighbors of $x_j$, 0 otherwise. Alternatively, $w_{ij}$ is set to 1 if $||x_i - x_j|| < \varepsilon$, and 0 otherwise.

The goal of graph embedding is to preserve the similarities amongst neighbors in both high and low dimensional spaces. The optimal $Y$ is found by minimizing:

$$\sum_{i,j} (y_i - y_j)^2 w_{ij} \tag{1}$$

As such, if neighbors $y_i$ and $y_j$ have a strong connection $w_{ij}$, their Euclidean distance should be minimal. $W$ is defined similarly for $X$ and $Y$, such that if neighbors $x_i$ and $x_j$ are close, $y_i$ and $y_j$ are also close. LGE seeks a linear approximation to this nonlinear concept of the form $y_i^T = x^T{}_i U$ or $Y^T = X^T U$. We define $D$ as a diagonal matrix of the column sums of $W$, $D_{ii} = \Sigma_j w_{ij}$; and $L$ is the Laplacian matrix, $L = D - W$. After simplification, the optimal $U$ is given by the minimum eigenvalue of the generalized eigenvector problem:

$$XLX^T U = \lambda XDX^T U \tag{2}$$

where $U$ is the resulting projection matrix.

Different choices of $W$ yield a multitude of dimensionality reduction techniques such as LDA, Locality Preserving Projections (LPP) [21], and Neighborhood Preserving Embedding (NPE) [22]. For each approach, $W$ is initialized to all zeros, and then connected $w_{ij}$ entries are determined by similarity.

For LDA, nodes $w_{ij}$ from the same class are set to $1/k_n$, where $k_n$ is the number of samples per their shared class:

$$w_{ij} = 1/k_n \tag{3}$$

For LPP, if nodes $i$ and $j$ are connected, then:

$$w_{ij} = e^{-\frac{\|x_i - x_j\|^2}{t}} \tag{4}$$

LPP can be used in supervised mode by defining connected neighbors as those which share similar class labels.

## 3. Sparse Signal Representation

### 3.1. Sparse Representations

A natural representation of a low dimensional sample $y \in \mathbf{R}^d$ from a training dictionary $\Phi \in \mathbf{R}^{dxn}$ is obtained by solving $\hat{y} = \Phi a$, where $a \in \mathbf{R}^n$ is the weight of each training exemplar in the dictionary $\Phi$. The objective of SRs is to identify the smallest number of nonzero coefficients $a$, such that $\hat{y} = \Phi a$. Donoho *et al*. [1] and Candes *et al*. [2] introduced a convex relaxation approach called Basis Pursuit Denoising (BPDN):

$$\hat{a} = \min\|a\|_1 \quad s.t. \|\hat{y} - \Phi a\|_2 \leq \epsilon \tag{5}$$

Often (5) is approximated by loosening the error constraints and reconfigured to specifically include a regularization term, λ:

$$\hat{a} = \min\{\|\hat{y} - \Phi a\|_2^2 + \lambda\|a\|_1\} \tag{6}$$

Perhaps the most widely used method to solve the $\ell^1$ minimization of (5) or (6) is Orthogonal Matching Pursuit (OMP) [23]. Given the SR coefficients $\hat{a}$ of a test image using the dictionary Φ, a reconstruction error method estimates the class $k^*$ of a query sample $y$. Given $k$ classes, the reconstructed sample using sparse coefficients $a$ from all classes is compared to the reconstructed sample using coefficients $a^i$ from each respective class:

$$k^* = \min_{i=1:k} \left\| y - \Phi a^i \right\|_2 \tag{7}$$

When constructing Φ the goal is to generate an over-complete dictionary with $n>d$. This allows the necessary degrees of freedom for choosing the sparsest solution and produces smooth and graceful coefficient activity across diverse test samples [24].

To avoid dictionary redundancy, K-SVD [25] was introduced as a means to learn an over-complete but small dictionary. K-SVD is an iterative technique, where at each iteration, training samples are first sparsely coded using the current dictionary estimate, and then dictionary elements are updated one at a time while keeping others fixed. Each new dictionary element is a linear combination of training samples. Rubinstein [26] implemented an efficient implementation of K-SVD using Batch Orthogonal Matching Pursuit.

The works of [18, 27, 28] jointly optimize dictionary learning and classifier training to select exemplars that minimize both reconstructive and discriminative errors. Jiang *et al*. [5] devised efficient methods for choosing Φ from a set of training exemplars by minimizing both reconstruction and classification errors in an optimal fashion. The work in [5] encourages input samples from the same class to have similar sparse codes.

## 4. Formulation of LGE-KSVD

### 4.1. Dimensionality Reduction and Sparse Representations

Although methods for populating the adjacency matrix $W$ vary, sparseness is one common characteristic across all techniques. Sparsity Preserving Projections (SPP) [17] is similar to NPE, but uses sparse coefficients instead of local topology when solving for $W$. Global Sparse Representation Projections [29] modifies the dimensionality reduction function in SPP to simultaneously maximize supervised class separability and minimize sparse representation error. [19] uses the sparse coefficients to populate matrix $W$, then adds supervised similarity and dissimilarity matrices akin to LDA. [20] replaces the $y$ terms in (1) with coefficients $\hat{a}$, claiming that nearby samples should have similar coefficients.

We wish to combine the dimensionality reduction matrix $U$ from (2) with a method to learn a dictionary Φ and sparse coefficients $a$. K-SVD solves:

$$\{\hat{\Phi}, \hat{a}\} = \min\|x - \Phi a\|_2^2 \quad s.t. \|a\|_0 \leq T \tag{8}$$

where $\hat{}$ denotes estimate. Combining (2) with (8), we get:

$$\{\hat{U}, \hat{\Phi}, \hat{a}\} = \min\|X^T U - \Phi a\|_2^2 + \frac{U^T X L X^T U}{U^T X D X^T U} \quad s.t. \|a\|_0 \leq T \tag{9}$$

The first term performs K-SVD optimization in low dimensional space, and the second term is the LGE dimensionality reduction objective function. LGE is used as subspace clustering during dimensionality reduction minimizes SR coefficient contamination by enforcing class separation.

Equation (9) is neither directly solvable nor convex. A discriminative dictionary was utilized in [5, 17-19, 29]. We find better results if the SR energy function minimizes reconstruction errors and the LGE energy function encourages class discrimination. Not only does this offer superior classification results, but because we are operating in a low dimensional space, the resulting framework minimizes compute intensity.

After an initial dimensionality reduction matrix $U$ is obtained via semi-supervised LGE, we propose a double nested iterative training procedure. The outer loop updates $U$ based upon the best estimates of Φ and $a$, and the inner loop uses K-SVD to iteratively update $a$, then Φ.

To get an updated estimate of $U$, coefficients $a$ from each training example are stored into matrix $A$, $A \in \mathbf{R}^{mxn}$. The update problem is then formulated as:

$$\hat{U} = \min\|X^T U - A^T \Phi^T\|_2^2 \tag{10}$$

and is solved directly:

$$U = (XX^T)^{-1} X A^T \Phi^T \tag{11}$$

Classification is performed with coefficient transformation matrix $C$, $C \in \mathbf{R}^{mxk}$, where $k$ is the number of classes and $m$ is the number of dictionary elements. We define $H$ as a sparse ground truth matrix, $H \in \mathbf{R}^{kxn}$. Each column of $H$ corresponds to a training sample, where the $k^{th}$ element is set to 1 if $y_i$ belongs to class $k$, 0 otherwise. This problem is formulated as:

$$\hat{C} = \min\|H - C^T A\|_2^2 \tag{12}$$

which can be solved directly:

$$C = (AA^T)^{-1} A H^T \tag{13}$$

With training complete, given a test sample $x$, along with $U$, $\Phi$, and $C$, we first calculate low dimensional sample $y^T = x^T U$, then calculate sparse coefficients $a$ using (6), and finally use $C$ along with $a$ to estimate class label vector $l \in \mathbf{R}^k$, where the maximum value of $l$ is used as a class identifier:

$$\hat{l} = \max_{i=1:k}(l = C^T a_i) \tag{14}$$

The choice of LGE technique needs to be a discriminative embedding which maintains input topology. An optimal approach uses a convex combination of supervised and unsupervised adjacency matrices $W_{LDA}$ and $W_{Gaussian}$ corresponding to (3) and (4) respectively. The two are combined into a single $W$:

$$W = \alpha W_{LDA} + (1 - \alpha) W_{Gaussian} \tag{15}$$

For posed datasets which are linearly separated, $W_{LDA}$ is weighted higher. For natural datasets or classification problems in which the number of classes is small, we emphasize the addition of $W_{Gaussian}$.

We call this method LGE-KSVD for Linear extension of Graph Embedding for optimized K-SVD dictionary learning. The next section demonstrates how to improve the K-SVD learning by injecting LGE concepts directly into the K-SVD atom definition.

### 4.2. Updating the Atom Optimization in K-SVD

The K-SVD penalty term of (8) can be rewritten as:

$$\begin{aligned} \|X - \Phi A\|_F^2 &= \left\| X - \sum_{z=1}^{m} \Phi_z A_T^z \right\|_F^2 \\ &= \left\| \left( X - \sum_{z \neq j} \Phi_z A_T^z \right) - \Phi_j A_T^j \right\|_F^2 \\ &= \left\| E_j - \Phi_j A_T^j \right\|_F^2 \end{aligned} \tag{16}$$

Where $\Phi A$ is decomposed into the sum of $m$ rank-1 matrices $\in \mathbf{R}^{dxn}$, each representing one dictionary element, with $\Phi \in \mathbf{R}^{dxm}$ and $A \in \mathbf{R}^{mxn}$, and $\|\cdot\|_F$ is the Frobenius norm. The error, $E_j$ is the total error for all $n$ training samples with the $j^{th}$ dictionary element removed. The step of updating dictionary elements sequentially updates one element at a time. While updating element $j$, K-SVD assumes that dictionary matrix $\Phi$ and sparse coefficient matrix $A$ are fixed except for (column) element $j$ of $\Phi$, $\Phi_j \in \mathbf{R}^{dx1}$, and the corresponding coefficients for $\Phi_j$, which comprise row $j$ of coefficient matrix $A$, $A^j{}_T \in \mathbf{R}^{1xn}$.

SVD solves the closest rank-1 matrix that approximates $E_j$, yielding $\Phi_j$ and $A^j{}_T$ directly; our new best estimates for dictionary element $j$ and its corresponding coefficients. Unfortunately this would tend to result in a non-sparse $A^j{}_T$. K-SVD enforces sparsity by fixing the support of $\Phi_j$ to only those training entries with non-zero coefficients of element $j$; as such, at initialization, each dictionary element is paired up with a list of training samples that can never change.

An improvement is to let the training samples that contribute to each dictionary element be governed by sample-to-sample similarity and class labels. Further, as long as sparsity is maintained, it is desired for the support of each element to change at each iteration. We propose to use semi-supervised LGE adjacency matrix $W$ as per (15) to regulate the support of each dictionary element. In particular, the support of dictionary element $j$ may:

- Expand: Modify the support of element $j$ by adding (union) all training entries *similar* to element $j$.
- Contract: Modify the support of element $j$ by removing (intersection) training entries not *similar* to element $j$.
- Redefine: Set the support of element $j$ to be only training samples *similar* to element $j$.

In LGE-KSVD, *similar* is defined to be training samples that respect LGE adjacency matrix $W$ (i.e., all samples of same class or nearest neighbors). During the updating of dictionary element $j$, LGE-KSVD modifies the support of $E_j$ using the expand, contract, and redefine operations, creating a different $E^R{}_j$ for each condition; then SVD decomposes $E^R{}_j = U \Delta V^T$. $\Phi_j$ is the first column of U, and $A^j{}_T$ is the first column of V multiplied by $\Delta(1,1)$. Given three sets of $E^R{}_j$, $\Phi_j$, and $A^j{}_T$ we choose the $\Phi_j$ and $A^j{}_T$ that minimize our penalty term (16) as:

$$\{\widehat{\Phi_j}, \widehat{A_T^j}\} = \min_{i=1:3} \left( \left\| E_j^R{}_i - \Phi_j{}_i A^j{}_{T\,i} \right\|_F^2 \right) \tag{17}$$

Rather than assigning a single class to each element $j$, LGE-KSVD uses the top coefficients from $A^j{}_T$. Each of those top coefficients is used as a look-up into adjacency matrix $W$. All training samples similar to each of those top coefficients (as defined by $W$) are used to expand, contract, or redefine $E_j$ before the SVD decomposition. The top coefficients are solved by keeping the top percentile of total energy.

Although it may seem desirable to use the same $W$ for dimensionality reduction as well as element neighbor similarity determination, there are advantages in maintaining some degree of flexibility. For example, it is often desirable to decrease $\alpha$ in (15) or increase $\tau$ in (4) for element neighbor similarity determination. Both modifications make $W$ less sparse, and perhaps less discriminate, but simultaneously make $W$ more open to finding relationships between diverse training samples.

### 4.3. Putting it all together

The LGE-KSVD algorithm is summarized in Figure 1. To calculate $U$ in step 1a, we use LPP in a semi-supervised mode via (15). Regarding the selection of $\alpha$ in

(15), $\alpha$ values should be in the range $0.1 \le \alpha \le 0.9$, and we recommend use $\alpha=0.5$.

WHILE1 $\varepsilon$ has not converged
  IF firstIteration
    1a. Calculate $U$ using LGE.
  ELSE
    1b. Calculate $U$ using (11).
  ENDIF
  2. Calculate low dimensional samples $Y^T = X^T U$.
  3. Initialize the $m$ samples of $\Phi$ randomly from the $n$ low dimensional training samples.
  4. Calculate $\{A, \Phi\}$ using modified K-SVD:
  WHILE2 $\xi$ has not converged
    4.1. Calculate coefficients $A$ using (6).
    4.2. Update dictionary $\Phi$:
    FOREACH element $j$ in dictionary
      4.2a. Calculate $\Phi_j$, $A^j_T$, and $E^R_j$ in expand.
      4.2b. Calculate $\Phi_j$, $A^j_T$, and $E^R_j$ in contract.
      4.2c. Calculate $\Phi_j$, $A^j_T$, and $E^R_j$ in redefine.
      4.2d. Calculate $\Phi_j$, $A^j_T$, and $E^R_j$ in fixed.
      4.2e. Select the $E^R_j$ and corresponding $\Phi_j$ and $A^j_R$ that minimize (17).
    END FOREACH
  END WHILE2
  5. Calculate $C$ using (13).
  6. Calculate verification set error, $\varepsilon = \|H - C^T A\|_2^2$.
END WHILE1

Figure 1. The LGE-KSVD Algorithm.

The introduction of K-SVD not only shrinks the size of the dictionary, but also results in higher classification accuracy. The size of the dictionary $m$ varies with the dataset size $n$, and we typically set $m=n/2$, and can often shrink $m$ such that $m<<n$.

To update the variable support of each dictionary element in Step 4.2, the selection of top coefficients is done by picking training samples with >50% of the total coefficient energy. Similar neighbors are those elements whose Euclidean distance in the $W$ matrix includes 99% of all related training samples. This accounts for 100% of samples of the same class, as well as other samples that are deemed to be similar by the unsupervised LPP adjacency matrix

## 5. Experiments

We evaluate the proposed LGE-KSVD approach on four public databases: the extended Cohn-Kanade (CK+) facial expression dataset [30], the extended Yale B facial recognition database [31], the Facial Expression Recognition and Analysis Challenge (FERA2011) GEMEP-FERA [32] dataset, and the i3DPost multi-view activity recognition dataset [33]. We test each dataset across three categories of (i) dimensionality reduction; (ii) sparse representation; and (iii) combined techniques. The dimensionality reduction techniques include PCA, LDA, LPP [21], NPE [22], and Sparsity Preserving Projections (SPP) [17]. The sparse representation methods include K-SVD [25], LC-KSVD1 and LC-KSVD2 [5]. The combined methods include Sparse Representation-based Classification (SRC) [8] and the LGE-KSVD method.

### 5.1. Testing Datasets

The CK+ [30] expression dataset contains 118 subjects in 327 sequences exhibiting the expressions of anger, disgust, fear, happiness, sadness, surprise, and contempt. An Active Appearance Model (AAM) automatically localizes 68 points on the face. The AAM eye and mouth corner points are used to define an affine warp to a canonical face of 60x51 pixels. As such, from this dataset we compare two variants: D=68x2=136 (AAM point based), and D=60x51=3060 (pixel based). Each has 164 training and 163 testing faces (chosen randomly), and the K-SVD methods use a dictionary size of 63 elements.

The Extended YaleB facial recognition dataset contains 2,414 frontal images of 38 people under varying illumination and facial expression. Each face is 192x168 pixels which are reduced to $D$=504 via random projections following [8]. The test set contains 1216 training faces and 1198 testing faces. The K-SVD methods use a dictionary size of 570 elements.

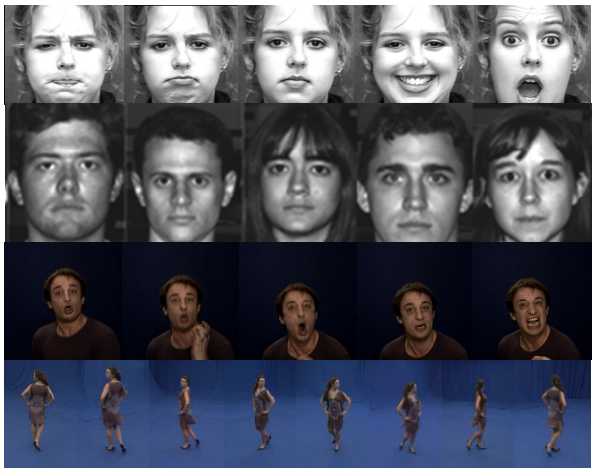

Figure 2. Sample subjects exhibiting (from top to bottom) static facial expressions, facial identity, temporal emotion, and multi-view activity from the CK+, YaleB, GEMP-FERA, and i3DPost datasets respectively.

The GEMEP-FERA temporal expression dataset contains 155 training and 134 testing videos. Each video sequence varies from 20-150 frames of 10 actors exhibiting the five emotions of anger, fear, joy, relief, and sadness. Automatically localized eye and mouth corner points define an affine warp to a canonical face of 60x51 pixels per each frame. A sequence of 16 frames at the $1/3^{rd}$ and $2/3^{rd}$ mark of each video is fed into Motion History Image (MHI) [34] analysis yielding a 24x20 dense optical flow per sequence. The X and Y coordinates at each 24x20 grid point for each of the two sequences formed the $D$=1920 input dimensions per sample. The K-SVD methods use a dictionary size of 75 elements.

The i3DPost multi-view [33] activity recognition dataset contains 768 videos of 8 people performing 12 actions from 8 views. The 12 activities are walk, run, jump, bend, hand-wave, jump in place, sit-stand, run-fall, walk-sit, run-jump-walk, handshake, and pull. Each video is MHI processed, giving 125 MHI sequences, each sequence containing 1500 motion vector points. PCA yielded 767 dimensions per video. The dataset contains 512 training videos and 256 testing videos. The K-SVD methods use a dictionary size of 450 elements.

### 5.2. Testing Methodologies

The dimensionality reduction techniques capture 99.9% of the data variance, and all use multi-class linear SVM as a classifier. LDA uses equation (3) and LPP uses (4). NPE and SPP are adopted from [22] and [17] respectively.

The sparse representation techniques all use K-SVD to define a training dictionary of size $m$, where $m<n$. Coefficient transformation matrix $C$ is generated from the training set as per (13). Test samples use the $m$ element dictionary to generate sparse coefficients using (6), setting $\lambda$=0.25. These sparse coefficients are converted to a class estimate using (14). LC-KSVD1 modifies the K-SVD objective function to favor clustering of coefficients by class and LC-KSVD2 further modifies the K-SVD objective function to include the solution of coefficient transformation matrix $C$.

The SRC method uses random projection matrices for dimensionality reduction. The low dimensional projection of all training samples forms the training dictionary. The corresponding sparse coefficients of test samples use (7) to make a final classification estimate. All LPP methods use $\alpha$=0.5 in creation of $W$ using (15). The LGE-KSVD method uses a $\tau$=1 for dimensionality reduction $W$ and $\tau$=100 for element neighbor similarity $W$. The LGE-KSVD method keeps the top coefficients which make up 50% of the total energy from $A^j_T$.

### 5.3. Experimental Results

Table 1 demonstrates the performance of the five dimensionality reduction methods, the three sparsity based methods, and the SRC combined method against LGE-KSVD on the 7-class CK+ dataset using the 68 AAM points. Because the data is only 136 dimensions, no dimensionality reduction is used for K-SVD, LC-KSVD1, LC-KSVD2, or SRC. This is a posed dataset, and as such LDA performs the best out of the dimensionality reduction techniques. The LGE-KSVD method has two numbers in the accuracy entry for Tables 1-5. The first is with iterative convergence turned off (1 iteration), and the second is the accuracy after convergence. The value in (·) after the second accuracy entry is the number of iterations required for convergence.

Table 2 uses the same CK+ dataset from Table 1, but uses 60x51 images as input. This higher dimensional space is not as discriminative as the 68 AAM points, but all methods do well because of the large separation of facial expression in each class.

Table 3 uses the 38-class YaleB facial recognition dataset. The 504 random projection input for all methods was further reduced in dimensionality as indicated by the $d$ column, where $d$ is the dimension where classification is performed. The SR methods are advantaged over the dimensionality reduction methods, while the combined LGE-KSVD method performs the best.

| *Method* | *d* | *m* | *% Accuracy* |
|---|---|---|---|
| PCA | 62 | - | 82.2 |
| LDA | 6 | - | 89.6 |
| LPP | 62 | - | 83.4 |
| NPE | 24 | - | 80.4 |
| SPP | 48 | - | 87.7 |
| K-SVD | 136 | 63 | 79.1 |
| LC-KSVD1 | 136 | 63 | 79.1 |
| LC-KSVD2 | 136 | 63 | 75.5 |
| SRC | 136 | 164 | 43.6 |
| LGE-KSVD (this paper) | 62 | 63 | 90.2 / **92.0** (2) |

Table 1. Classification results on the 7-Class CK+ Expression Dataset, using 68 AAM Points. 164 training and 163 testing samples.

| *Method* | *d* | *m* | *% Accuracy* |
|---|---|---|---|
| PCA | 162 | - | 82.8 |
| LDA | 6 | - | 86.5 |
| LPP | 163 | - | 84.7 |
| NPE | 71 | - | 84.0 |
| SPP | 80 | - | 77.9 |
| K-SVD | 3060 | 63 | 84.0 |
| LC-KSVD1 | 3060 | 63 | 85.9 |
| LC-KSVD2 | 3060 | 63 | 84.7 |
| SRC | 500 | 164 | 71.8 |
| LGE-KSVD (this paper) | 163 | 63 | 86.5 / **87.1** (5) |

Table 2. Classification results on the 7-Class CK+ Expression Dataset, using 60x51 Images. 164 training and 163 testing samples.

Table 4 uses the 5-class GEMEP-FERA emotion dataset. Two MHI optical flow sequences per video were used as input. The dimensionality reduction methods are advantaged over the SR methods, and the combined

methods perform better than the dimensionality reduction methods.

| *Method* | *d* | *m* | *% Accuracy* |
|---|---|---|---|
| PCA | 477 | - | 89.1 |
| LDA | 37 | - | 90.3 |
| LPP | 477 | - | 89.3 |
| NPE | 271 | - | 91.2 |
| SPP | 288 | - | 88.7 |
| K-SVD | 504 | 570 | 93.2 |
| LC-KSVD1 | 504 | 570 | 93.7 |
| LC-KSVD2 | 504 | 570 | 93.4 |
| SRC | 504 | 1216 | 86.1 |
| LGE-KSVD (this paper) | 477 | 570 | **95.7** / **95.7** (1) |

Table 3. Classification results on the 38-Class YaleB Recognition Dataset. 192x168 pixel images reduced to 504 dimensions via random projections. 1216 training images, 1198 testing images.

| *Method* | *d* | *m* | *% Accuracy* |
|---|---|---|---|
| PCA | 154 | - | 55.2 |
| LDA | 4 | - | 55.2 |
| LPP | 154 | - | 55.2 |
| NPE | 66 | - | 56.7 |
| SPP | 75 | - | 52.2 |
| K-SVD | 1920 | 75 | 51.5 |
| LC-KSVD1 | 1920 | 75 | 53.7 |
| LC-KSVD2 | 1920 | 75 | 51.5 |
| SRC | 500 | 155 | 57.5 |
| LGE-KSVD (this paper) | 154 | 75 | 58.2 / **61.2** (9) |

Table 4. Classification results on the 5-Class GEMEP-FERA Emotion Dataset. MHI motion vectors. 155 training videos, 134 testing videos.

| *Method* | *d* | *m* | *% Accuracy* |
|---|---|---|---|
| PCA | 510 | - | 94.9 |
| LDA | 510 | - | 94.5 |
| LPP | 510 | - | 96.1 |
| NPE | 224 | - | 94.9 |
| SPP | 241 | - | 91.0 |
| K-SVD | 767 | 450 | 94.1 |
| LC-KSVD1 | 767 | 450 | 95.3 |
| LC-KSVD2 | 767 | 450 | 93.8 |
| SRC | 767 | 512 | 88.7 |
| LGE-KSVD (this paper) | 510 | 450 | **96.9** / **96.9** (1) |

Table 5. Classification results on the 12-Class i3DPost Multi-view Activity Recognition Dataset. 512 training videos, 256 testing videos.

Table 5 uses the 12-class i3DPost multi-view activity recognition dataset. The 767 PCA projection input for all methods was further reduced in dimensionality as indicated by the *d* column. While there is no clear winner on this dataset, the semi-supervised LPP methods performed the best.

The results in Tables 1-5 show impressive accuracy performance of LGE-KSVD across a wide variety of problem sets. We attribute this to the discriminative strengths of dimensionality reduction, the classification power of SR methods, along with the integration of LGE into the K-SVD dictionary learning architecture.

When SR methods have insufficient training exemplars in Φ, their performance lags behind SVM classification methods. When datasets are posed, LDA dimensionality reduction is preferred; when datasets are natural, semi-supervised LPP or NPE methods are preferred. The LGE-KSVD representation offers the discriminative properties of LDA while maintaining the local topology of complex data representations in the low dimensional manifold representations. As such, LGE-KSVD has been shown to be robust over datasets with few vs. many classes, high vs. low dimensionality, posed vs. spontaneous faces, static vs. temporal features, and across the classification problems of facial expression, facial recognition, and human activity recognition.

## 6. Conclusions

This paper presents LGE-KSVD, a new method that integrates manifold-based dimensionality reduction and sparse representations within a single framework. We leverage LGE dimensionality reduction concepts to optimize K-SVD dictionary learning such that the support of dictionary elements remains sparse, but is no longer fixed. Semi-supervised dimensionality reduction produces sufficient discrimination between input classes to minimize coefficient contamination while preserving enough geometric detail from the high dimensional space for real world imagery. The modification of the dictionary element update step in K-SVD improves classification accuracy while making K-SVD more generally applicable to a broader spectrum of problems and less reliant on a good initialization. Our results show that our proposed LGE-KSVD framework provides significant advantages over other techniques across a wide variety of facial and activity classification problems.

## Acknowledgements

This paper was supported in part by Cisco and a National Science Foundation Graduate Research Fellowship.

## References

[1] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory,* vol. 52, pp. 6-18, 2006.

[2] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory,* vol. 52, pp. 489-509, 2006.

[3] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient Sparse Coding Algorithms," Advances in Neural Information Processing Systems, 2006.

[4] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least Angle Regression," *Ann. Statist.,* vol. 32, pp. 407-499, 2004.

[5] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[6] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher Discrimination Dictionary Learning for Sparse Representation," *International Conference on Computer Vision*, 2011.

[7] A. Yang, J. Wright, Y. Ma, and S. Sastry, "Feature Selection in Face Recognition: A Sparse Representation Perspective," University of California at Berkely, 2007.

[8] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and M. Yi, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 31, pp. 210-27, 2009.

[9] J. Sherrah, S. Gong, and E. J. Ong, "Face distributions in similarity space under varying head pose," *Image and Vision Computing,* vol. 19, pp. 807-819, 2001.

[10] G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "Sparse representations of image gradient orientations for visual recognition and tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops,* 2011.

[11] S. Zafeiriou and M. Petrou, "Sparse representations for facial expressions recognition via l1 optimization," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops,* 2010.

[12] R. Ptucha, G. Tsagkatakis, and A. Savakis, "Manifold Based Sparse Representation for Robust Expression Recognition without Neutral Subtraction," BeFIT Workshop, International Conference on Computer Vision, 2011.

[13] E. Elhamifar and R. Vidal, "Sparse subspace clustering," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops,* 2009.

[14] A. Ghodsi., "Dimensionality Reduction A Short Tutorial," University of Waterloo, Ontario,m CA, 2006.

[15] L. Cayton., "Algorithms for manifold learning," University of California, San Diego, Tech Rep. CS2008-0923, 2005.

[16] C. Deng, H. Xiaofei, H. Yuxiao, H. Jiawei, and T. Huang, "Learning a spatially smooth subspace for face recognition," *IEEE Conference on Computer Vision and Pattern Recognition,* 2007.

[17] L. Qiao, S. Chen, and X. Tan, "Sparsity preserving projections with applications to face recognition," *Pattern Recognition,* vol. 43, pp. 331-341, 2010.

[18] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Discriminative learned dictionaries for local image analysis," *IEEE Conference on Computer Vision and Pattern Recognition,* 2008.

[19] F. Zang and J. Zhang, "Discriminative learning by sparse representation for classification," *Neurocomputing,* vol. 74, pp. 2176-2183, 2011.

[20] M. Zheng*, et al.*, "Graph regularized sparse coding for image representation," *IEEE Transactions on Image Processing,* vol. 20, pp. 1327-1336, 2011.

[21] X. He and P. Niyogi, "Locality Preserving Projections," *Advances in Neural Information Processing Systems*, Vancouver, Canada, 2003.

[22] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *IEEE International Conference on Computer Vision,* 2005.

[23] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," *Asilomar Conference on Signals, Systems and Computers,*1993.

[24] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multiscale transforms," *IEEE Transactions on Information Theory,* vol. 38, pp. 587-607, 1992.

[25] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing,* vol. 54, pp. 4311-22, 2006.

[26] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient Implementationof the K-SVD Algorithm using Batch Orthogonal Matching Pursuit," Technion, Computer Science Dept., Haifa, Israel, 2008.

[27] Z. Qiang and L. Baoxin, "Discriminative K-SVD for Dictionary Learning in Face Recognition," *IEEE Conference on Computer Vision and Pattern Recognition,* 2010.

[28] W. Jinjun, Y. Jianchao, Y. Kai, L. Fengjun, T. Huang, and G. Yihong, "Locality-constrained linear coding for image classification," *IEEE Conference on Computer Vision and Pattern Recognition,* 2010.

[29] L. Zhihui, J. Zhong, and Y. Jian, "Global sparse representation projections for feature extraction and classification," in *Chinese Conference on Pattern Recognition. and the First CJK Joint Workshop on Pattern Recognition,* 2009.

[30] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops,* 2010.

[31] 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 Transactions on Pattern Analysis and Machine Intelligence,* vol. 23, pp. 643-60, 06/ 2001.

[32] M. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. R. Scherer, "The First Facial Expression Recognition and Analysis Challenge," *Face and Gesture Recognition*, 2011.

[33] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis, and I. Pitas, "The i3DPost multi-view and 3D human action / interaction," in *CVMP*, 2009.

[34] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 23, pp. 257-67, 2001.