



# Adaptive Hybrid Representation for Graph-Based Semi-supervised Classification

F. Dornaika<sup>1,2</sup>  and A. Bosaghzadeh<sup>3</sup>

<sup>1</sup> University of the Basque Country UPV/EHU, San Sebastian, Spain  
fadi.dornaika@ehu.es

<sup>2</sup> IKERBASQUE, Basque Foundation for Science, Bilbao, Spain

<sup>3</sup> Shahid Rajaei Teacher Training University, Tehran, Iran

**Abstract.** Building an informative graph over a collection of images or signals is one of the most important tasks in semi-supervised learning (SSL). Local Hybrid Coding (LHC) was recently proposed as an alternative to the sparse coding scheme that is used in Sparse Representation Classifier (SRC). The LHC blends sparsity and bases-locality criteria in a unified optimization problem. This paper introduces a data-driven graph construction method that exploits and extends the LHC scheme. We propose a new coding scheme coined Adaptive Local Hybrid Coding (ALHC). The main contributions are as follows. First, the proposed coding scheme automatically selects the local and non-local bases of LHC using data similarities calculated by Locality-constrained Linear code. Second, the estimated similarities are used in the regularization of the final solution. Third, the proposed ALHC scheme is used in order to construct graphs over image datasets. For SSL tasks adopting label propagation, we show that the proposed graph outperforms many state-of-the-art graphs on three public face datasets.

**Keywords:** Graph construction · Sparse coding · Local hybrid code · Label propagation

## 1 Introduction

Semi-supervised learning is one of the most important fields in machine learning. It is mainly used in the cases where there are a huge amount of labeled samples, but very few labeled ones.

It can be an interesting solution specially in the cases where acquiring unlabeled data is easy and cheap but obtaining labeled data is difficult which is the case in many real world problems such as: (i) image classification, (ii) webpage classification, (iii) speech recognition, (iv) person emotion recognition in videos [1], and (v) protein sequence classification.

Graph-based semi-supervised learning which adopts an affinity graph to represent the relation between the samples has gain a lot of attention in the last

decade (e.g., [4, 6]). Indeed, graph-based algorithms are widely used nowadays in a variety of machine learning tasks such as: (i) semi-supervised learning for label propagation and regression [7], (ii) feature selection, (iii) graph-based embedding [8], and (iv) spectral clustering [14]. Over the past decade, several graph construction techniques have been proposed. In this paper, we propose a new technique to construct a graph based on the Local Hybrid Coding [18]. Considering both the bases-locality and sparsity constraints in a unified framework, LHC obtains the advantages of both types of coding. Dense coding with  $\ell_2$  regularization can better represent the geometric structure of data manifold which can increase the accuracy of classification due to better discrimination power [15]. At the same time, the  $\ell_1$ -sparsity guarantees the correct representation of input data in case very few samples are available [12, 17].

The main differences between our approach and the LHC scheme of [18] are as follows. Firstly, in our work, we construct a data-driven graph using data self-representativeness whereas in [18], the authors propose a variant of the Sparse Representation Classifier that uses the hybrid coding instead of the sparse coding. Hence, in our work the adopted dictionary is obtained from the data whereas in [18] they use a pre-trained database. Secondly, in our work the similarity between the samples are derived from the coefficients that are obtained from a coding scheme namely, Locality-constrained Linear Coding (LLC) while in [18] the selection of local and non-local bases is based on Euclidean distance.

Thirdly, we adopt a biased weight for the coefficients of the local bases.

The remainder of this paper is organized as follows. Section 2 provides a brief review of graph construction and reviews the Local Hybrid Coding scheme. Our proposed method is introduced in Sect. 3. In Sect. 4, we present some experimental results obtained with three benchmark face image datasets. Section 5 concludes the paper. In this paper, capital bold letters denote matrices and small bold letters denote vectors.

## 2 Related Work

This section describes some existing methods for graph construction. Then, it will present a review of the recent Local Hybrid Coding scheme.  $k$ -nearest neighbor and  $\varepsilon$ -neighborhoods are two traditional graph construction methods. Let the original data set be denoted by  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ .

Locally Linear Embedding (LLE) focuses on preserving the local structure of data [11]. LLE formulates the manifold learning problem as a neighborhood-preserving embedding, which learns the global structure by exploiting the local linear reconstructions. It estimates the reconstruction coefficients by minimizing the reconstruction error of the set of all local neighborhoods in the dataset. It turned out that the linear coding used by LLE can be used for computing the graph weight matrix. Thus, LLE graph can be obtained by applying two stages: adjacency matrix computation followed by the linear reconstruction of samples from their neighbors. The adjacency matrix can be computed using the KNN or  $\varepsilon$ -Neighborhood method. In [5], the authors utilize LLC for graph construction. They propose a graph construction method that is based on a variant of LLC.

On the other hand, sparsity representation based graph is parameter-free. [10] and [19] proposed sparsity representation based graph construction methods in which every sample is represented as a sparse linear combination of the rest of input samples and the coefficients are considered as weights.

$$\min \|\mathbf{w}_i\|_1, s.t. \mathbf{x}_i = \mathbf{X} \mathbf{w}_i, \quad (1)$$

where  $\mathbf{w}_i = [w_{i1}, \dots, w_{i,i-1}, 0, w_{i,i+1}, \dots, w_{in}]^T$  is an  $n$ -dimensional vector with the  $i$ -th element being zero (implying that the  $\mathbf{x}_i$  is removed from  $\mathbf{X}$ ),  $\|\cdot\|_1$  is the  $\ell_1$  norm of a vector or matrix, and the elements  $w_{ij}, j \neq i$  denote the contribution of  $\mathbf{x}_j$  in the reconstruction of  $\mathbf{x}_i$ .

After the weight vector  $\mathbf{w}_i$  for each  $\mathbf{x}_i, i = 1, 2, \dots, n$  is obtained, the affinity matrix  $\mathbf{W} = (w_{ij})_{n \times n}$  is obtained as:

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n]^T, \quad (2)$$

where  $\mathbf{w}_i$  is the optimal solution of Eq. (1) problem: A robust version of the sparse graph can be obtained by solving the following problem:

$$\min \|\mathbf{w}_i\|_1 + \|\mathbf{e}\|_1, s.t. \mathbf{x}_i = \mathbf{X} \mathbf{w}_i + \mathbf{e}. \quad (3)$$

In this article, we call the graph that is constructed by the weights obtained from Eq. (1) as standard sparse graph ( $\ell_1$ -s) and the graph obtained by solving the Eq. 3 as robust sparse graph ( $\ell_1$ -r).

## 2.1 Review of Local Hybrid Coding

The authors in [18] propose a Local Hybrid Coding scheme to encode image descriptors by taking into account the bases-locality and  $\ell_1$ -sparsity. Hence their proposed method retains the advantages of Least Square coding scheme and  $\ell_1$ -sparsity.

Let  $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n] \in \mathbb{R}^{d \times n}$  denote a pre-trained dictionary which contains  $n$  samples each with dimensionality of  $d$ . Let  $\mathbf{x} \in \mathbb{R}^d$  denote a test sample. The objective is to project this sample onto the bases of  $\mathbf{B}$  via computing a code vector  $\mathbf{c}$  such that  $\mathbf{x} \approx \mathbf{B} \mathbf{c}$ . LHC ensembles the  $\ell_1$ -sparsity and bases-locality criteria into a unified optimization problem. The coding of sample  $\mathbf{x}$  with respect to the dictionary  $\mathbf{B}$  can be obtained by applying two steps.

In the first step, based on the distance between the sample  $\mathbf{x}$  and the atoms of the dictionary, the pre-trained dictionary is divided into two disjoint sets of  $\mathbf{B}^{(l)}$  that contain the  $k$ -nearest-neighbor (KNN) atoms ( $k_l$ ) and  $\mathbf{B}^{(s)}$  that contains the non- $k$ -nearest-neighbor ( $k_s$ ) atoms. The  $\mathbf{B}^{(l)}$  which contains the local samples are used for local coding and  $\mathbf{B}^{(s)}$  that contains non-local samples are used for sparse coding.

In the second step, based on the local codes  $\mathbf{c}^{(l)}$  that are obtained from the local bases  $\mathbf{B}^{(l)}$  and the sparse codes  $\mathbf{c}^{(s)}$  that are obtained from the  $\mathbf{B}^{(s)}$  basis, a hybrid code will be constructed by:

$$\min_{\mathbf{c}} \|\mathbf{x} - [\mathbf{B}^{(l)}, \mathbf{B}^{(s)}] [\mathbf{c}^{(l)T}, \mathbf{c}^{(s)T}]^T\|_2^2 + \gamma \|\mathbf{c}^{(l)}\|_2^2 + \lambda \|\mathbf{c}^{(s)}\|_1 \quad (4)$$

where  $\mathbf{c} = [\mathbf{c}^{(l)T}, \mathbf{c}^{(s)T}]^T$  is the hybrid code formed from two parts, local code  $\mathbf{c}^{(l)}$  and sparse code  $\mathbf{c}^{(s)}$ .  $\|\cdot\|_1$  and  $\|\cdot\|_2$  denote the  $\ell_1$ -norm and  $\ell_2$ -norm of a vector, respectively. Criterion 4 has three terms: the first term is the residual error of the sample reconstruction, the second term is the  $\ell_2$  norm of local basis coefficients and the third term is the  $\ell_1$  norm of the non-local bases coefficients.

Although the dictionary  $\mathbf{B}$  is partitioned into two disjoint subsets, their coefficients  $\mathbf{c}^{(l)}$  and  $\mathbf{c}^{(s)}$  are coupled, thus the convex optimization problem (4) is solved in an alternating optimization procedure. The two sets of unknown coefficients are then iteratively obtained by alternating regularized  $\ell_2$  coding and  $\ell_1$  coding over the local bases and the non-local bases, respectively. Note that when the sparse code  $\mathbf{c}^{(s)}$  is constant, the minimization problem in Eq. (4) reduces to a regularized Least Square problem that can be solved using a closed-form solution. Let  $\mathbf{x}^{(l)} = \mathbf{x} - \mathbf{B}^{(s)}\mathbf{c}^{(s)}$ . The optimal  $\mathbf{c}^{(l)}$  is then given by  $\mathbf{c}^{(l)} \leftarrow (\mathbf{B}^{(l)T}\mathbf{B}^{(l)} + \gamma\mathbf{I})^{-1}\mathbf{B}^{(l)T}\mathbf{x}^{(l)}$ . When the local part  $\mathbf{c}^{(l)}$  is constant, then the minimization problem in Eq. (4) reduces to a  $\ell_1$  regularized sparse coding problem that can be efficiently solved by the feature-sign search method [9].

Algorithm 1 describes the procedure of the proposed method. The *FeatureSign()* function is the algorithm described in [9] which computes the sparse code of a given sample w.r.t. a given dictionary. It should be noted that in each iteration only the local code (i.e.  $\mathbf{c}^{(l)}$ ) and sparse code (i.e.  $\mathbf{c}^{(s)}$ ) change. According to [18], convergence can be obtained in five iterations.

**Input:** Dictionary matrix  $\mathbf{B} \in \mathbb{R}^{d \times n}$ , sample  $\mathbf{x}$ ,  $\gamma$ ,  $\lambda$ ,  $k_l$ , and  $k_s$ .

**Output:** LHC Code vector  $\mathbf{c}$

$\mathbf{c}^{(s)} \leftarrow \mathbf{0}$  ;

Sort the vectors of  $\mathbf{B}$  in ascending order of their distances to  $\mathbf{x}$  and obtain basis vectors into the matrix  $\mathbf{B}$  ;

Split  $\mathbf{B}$  into KNN bases  $\mathbf{B}^{(l)}$  ( $k_l$  bases) and non-KNN bases  $\mathbf{B}^{(s)}$  ( $k_s$  bases) ;

**repeat**

$\mathbf{x}^{(l)} \leftarrow \mathbf{x} - \mathbf{B}^{(s)}\mathbf{c}^{(s)}$  ;  
     $\mathbf{c}^{(l)} \leftarrow (\mathbf{B}^{(l)T}\mathbf{B}^{(l)} + \gamma\mathbf{I})^{-1}\mathbf{B}^{(l)T}\mathbf{x}^{(l)}$  ;  
     $\mathbf{x}^{(s)} \leftarrow \mathbf{x} - \mathbf{B}^{(l)}\mathbf{c}^{(l)}$  ;  
     $\mathbf{c}^{(s)} \leftarrow \text{FeatureSign}(\mathbf{B}^{(s)}, \mathbf{x}^{(s)})$  ;

**until** *Convergence*;

$\mathbf{c} = [\mathbf{c}^{(l)T}, \mathbf{c}^{(s)T}]^T$

**Algorithm 1:** Local Hybrid Coding.

### 3 Proposed Approach: Adaptive LHC (ALHC) Graph

In this paper, we propose an adaptive graph construction method that is based on data self-representativeness and adopted a modified version of the LHC method. The proposed method is different from LHC is several aspects. First, in our work, we construct a data-driven graph using data self-representativeness whereas in [18], the authors target a coding scheme that can replace the sparse coding stage

in the Sparse Representation Classifier. Hence, the dictionary in the proposed method is constructed from the data themselves, compared to a pre-trained dictionary in [18]. Second, while [18] determined the similarity between the samples (and the selection of local and non-local bases) adopting Euclidean distance, in this article we use the similarity coefficients obtained by a Locality-constrained Linear Code (LLC) method. Third, our proposed scheme is able to adaptively select the local and non-local bases without any user-defined parameter. Fourth, our coding introduces weights for the local bases coefficients.

To construct the graph, for every sample, we estimate its code with respect to the rest of the samples in the database. Let  $\mathbf{X}_i \in \mathbb{R}^{d \times (n-1)}$  denote the data matrix associated with the set  $S_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_n\}$ . The whole process has two steps. In the first step, based on the similarity between a sample (i.e.  $\mathbf{x}_i$ ) and the rest of the samples (i.e.  $S_i$ ), the local and non-local bases are selected. In the second step, we obtain a hybrid code from the local and non-local sets. We proceed as follows.

**First Step.** We first estimate the coding of the sample  $\mathbf{x}_i$  with respect to the data matrix  $\mathbf{X}_i$  using LLC. Let  $\mathbf{a} \in \mathbb{R}^{n-1}$  denote this code. This vector is given by minimizing the LLC criterion:

$$\mathbf{a} = \arg \min_{\mathbf{a}} (\|\mathbf{x}_i - \mathbf{X}_i \mathbf{a}\|_2^2 + \sigma \sum_{j=1}^{n-1} p_j a_j^2) = \arg \min_{\mathbf{a}} \left( \|\mathbf{x}_i - \mathbf{X}_i \mathbf{a}\|_2^2 + \sigma \|\mathbf{P}^{1/2} \mathbf{a}\|_2^2 \right) \quad (5)$$

where  $\mathbf{P}$  is a diagonal matrix with elements  $P_{jj} = p_j$ . Any formula which forms a distance criterion between the sample  $\mathbf{x}_i$  and the sample  $\mathbf{x}_j$  can be used to calculate  $p_j$ . In our work, we use the following formula:

$$p_j = 1 - \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2) \quad (6)$$

By using simple linear algebra calculations, the solution to (5) has a closed-form solution:

$$\mathbf{a} = (\mathbf{X}_i^T \mathbf{X}_i + \sigma \mathbf{P})^{-1} \mathbf{X}_i^T \mathbf{x}_i \quad (7)$$

Since the score  $|a_j|$  encodes the similarity between the sample  $\mathbf{x}_i$  and the sample  $\mathbf{x}_j \in S_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_n\}$ , it is expected to be much better than the classic Euclidean distance  $\|\mathbf{x}_i - \mathbf{x}_j\|^2$ . Thus,  $|a_j|$  can be a good measure of locality between samples  $\mathbf{x}_i$  and  $\mathbf{x}_j$ .

We use the  $|a_j|, j = 1, \dots, n-1$  to split the data matrix  $\mathbf{X}_i$  (equivalently the set  $S_i$ ) into two disjoint sets of local  $\mathbf{X}_i^{(l)}$  and non-local  $\mathbf{X}_i^{(s)}$  bases.

The scores  $|a_j|$  are sorted in a descending order (i.e. decreasing the similarity) and correspondingly the samples in the set  $S_i$  are sorted into the set  $\hat{S}_i$ .

An adaptive threshold can be the result of applying any statistical function on the coefficients as:

$$t(\mathbf{x}_i) = f(|a_1|, \dots, |a_{n-1}|), \quad (8)$$

where  $f(|a_1|, \dots, |a_{n-1}|)$  is a statistical function that returns a scalar that depends on the set of  $|a_j|$ . One possible choice for this function can be the average of the obtained coefficients:

$$t(\mathbf{x}_i) = \frac{1}{n-1} \sum_{j=1}^{n-1} |a_j|. \quad (9)$$

Based the estimated threshold  $t(\mathbf{x}_i)$ , we can generate from the original set  $S_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \dots, \mathbf{x}_n\}$  the local set  $S_i^{(l)}$  and the non-local set  $S_i^{(s)}$ . The local set  $S_i^{(l)} = \{\mathbf{x}_j\}$  is determined by selecting the samples who coding coefficient satisfies  $|a_j| > t(\mathbf{x}_i)$ .

The non-local set is given by  $S_i^{(s)} = S_i - S_i^{(l)}$ . It should be noticed that the cardinality of both  $S_i^{(l)}$  and  $S_i^{(s)}$  depends on the current sample  $\mathbf{x}_i$ . However, in the case of LHC the cardinality of both local and non-local bases is fixed a priori for the whole dataset. Furthermore, the samples in  $S_i^{(s)}$  are ordered according to their scores  $|a_j|$ . Let  $k_l$  denote the size of the local bases (i.e., the size of  $S_i^{(l)}$ ), and  $k_s$  the size of the non-local bases (the size of  $S_i^{(s)}$ ).

**Second step.** In this step, we estimate the hybrid code  $\mathbf{c}_i$  for every sample  $\mathbf{x}_i$  using a modified LHC scheme.

For the sake of clarity, the subscript  $i$  is omitted in Eqs. (10) and (12). The hybrid code is obtained by:

$$\min_{\mathbf{c}} \|\mathbf{x} - [\mathbf{X}^{(l)}, \mathbf{X}^{(s)}][\mathbf{c}^{(l)T}, \mathbf{c}^{(s)T}]^T\|_2^2 + \gamma \|\mathbf{D}\mathbf{c}^{(l)}\|_2^2 + \lambda \|\mathbf{c}^{(s)}\|_1 \quad (10)$$

where  $\mathbf{D} \in \mathbb{R}^{k_l \times k_l}$  is a diagonal matrix containing the weights  $D_{jj}$  associated with each  $j^{th}$  component of the local code  $\mathbf{c}_i^{(l)}$ . In our work, we use the following expression for  $D_{jj}$ :

$$D_{jj} = 1/|a_j|, j = 1, \dots, k_l \quad (11)$$

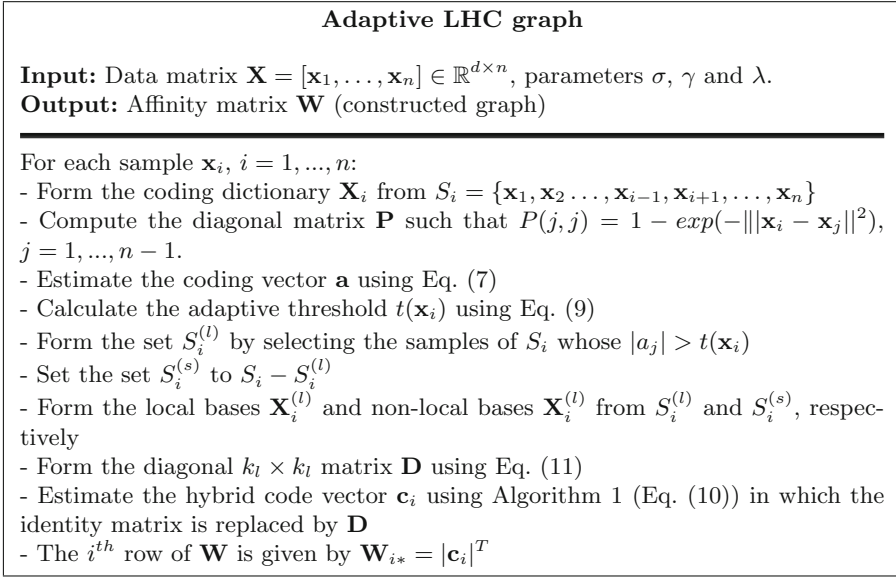
The solution to the above minimization can be obtained by Algorithm 1 where the solution for the local part is now given by:

$$\mathbf{c}^{(l)} = (\mathbf{B}^{(l)T}\mathbf{B}^{(l)} + \gamma\mathbf{D})^{-1}\mathbf{B}^{(l)T}\mathbf{x}^{(l)} \quad (12)$$

The ALHC is summarized in Fig. 1. We stress the fact that in the proposed method the size of local and non-local basis is sample dependent.

### 3.1 Kernel Variant of ALHC

The motivation behind using kernel representation relies on the fact that a linear model for data self-representation cannot be the best model. Therefore, by adopting non-linear models for data self-representation, it is expected that the estimated coding coefficients could better quantify the dependency and relation among samples and hence, better graph coefficients can be derived. Let  $\Phi: \mathbf{X} \rightarrow \Phi(\mathbf{X})$  be a non-linear mapping that projects original data samples onto a space of high dimension. Following the Kernel theory, it is not necessary to know the explicit function  $\Phi$  since what is really needed is the dot product among the projected samples. In this new space, the data samples are represented by the



**Fig. 1.** The proposed ALHC graph.

matrix  $\Phi = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_n)]$ . Let  $K_{ij} = \phi^T(\mathbf{x}_i) \phi(\mathbf{x}_j)$  be the dot product of the projection of two samples  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . This dot product quantifies a similarity measure between samples  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . The kernel matrix  $K(\cdot, \cdot)$  can be built using Gaussian, polynomial, or any other function that satisfies Mercer's conditions. It is easy to show that the matrix  $\mathbf{K}$  will be given by  $\Phi^T \Phi$ . By adopting the mapped data,  $\Phi$ , the kernel variant of the proposed method can be obtained by replacing the data with their non-linear projections. Thus, the code vector associated with each sample will be estimated by minimizing the following:

$$\min_{\mathbf{c}} \|\phi(\mathbf{x}) - [\Phi(\mathbf{X}^{(l)}), \Phi(\mathbf{X}^{(s)})][\mathbf{c}^{(l)T}, \mathbf{c}^{(s)T}]^T\|_2^2 + \gamma \|\mathbf{c}^{(l)}\|_2^2 + \lambda \|\mathbf{c}^{(s)}\|_1 \quad (13)$$

## 4 Performance Evaluation: Graph-Based Label Propagation for Image Classification

The graph-construction method is assessed by the performance of the post-graph construction task. The latter is given by label propagation over the graph. In the experiments, we will use the Gaussian Fields and Harmonic Functions (GFHF) method [21] since it is non-parametric.

We used the following three public face datasets:

1. **Extended Yale - part B<sup>1</sup>**: It contains images of 38 human subjects. Each subject has about 60 images. The images are resized to  $32 \times 32$  pixels.

<sup>1</sup> <http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html>.

**Table 1.** Recognition performance (Mean recognition accuracy %) on the Extended Yale, PF01 and FERET datasets over ten different random splits.

Ext. Yale				PF01			
Method \ Lab.	q = 9	q = 14	q = 20	Method \ Lab.	q = 5	q = 8	q = 12
KNN	80.55	82.06	83.25	KNN	44.38	49.44	52.17
GoLPP [20]	35.85	48.62	59.65	GoLPP [20]	42.81	61.32	73.91
LNP [13]	93.45	94.23	95.05	LNP [13]	64.07	72.61	74.47
$\ell_1 - s$ [3]	82.15	86.59	89.98	$\ell_1 - s$ [3]	53.81	59.33	62.39
$\ell_1 - r$ [19]	93.19	95.01	96.35	$\ell_1 - r$ [19]	72.06	79.53	84.00
$\ell_1 - c$ [2]	80.15	84.44	87.76	$\ell_1 - c$ [2]	53.80	60.65	64.50
LLC [5]	88.57	92.34	95.37	LLC [5]	72.87	79.57	83.25
LHC [18]	87.56	89.72	92.45	LHC [18]	66.85	75.14	80.39
SRLS [16]	91.44	93.80	95.22	SRLS [16]	71.55	76.61	78.17
<b>ALHC</b>	<b>94.79</b>	<b>96.06</b>	<b>97.15</b>	<b>ALHC</b>	<b>74.07</b>	<b>80.92</b>	<b>85.53</b>

FERET			
Method \ Labeled	q = 2	q = 3	q = 4
KNN	31.33	38.96	49.70
GoLPP [20]	12.05	17.31	25.00
LNP [13]	56.20	70.24	77.95
$\ell_1 - s$ [3]	55.93	66.59	74.78
$\ell_1 - r$ [19]	55.05	69.13	81.27
$\ell_1 - c$ [2]	51.18	61.01	72.50
LLC [5]	57.06	71.09	80.25
LHC [18]	56.81	70.59	82.00
SRLS [16]	57.36	68.62	73.55
<b>ALHC</b>	<b>61.65</b>	<b>74.72</b>	<b>83.93</b>

- PF01**<sup>2</sup>: It contains the true-color face images of 103 people, 53 men and 50 women, representing 17 different images (1 normal face, 4 illumination variations, 8 pose variations, 4 expression variations) per person.
- FERET**<sup>3</sup>: In our experiments, we use a subset of FERET. This subset consists of 1400 images for 200 different person (7 images per person).

#### 4.1 Method Comparison

For quantitative evaluation of the proposed method, we compare the performance of the classification of the graph obtained from the proposed method with the ones obtained from several state of the art graph construction techniques. We divide the database into two sets of labeled and unlabeled, and then construct the graph using the union of both sets.

<sup>2</sup> <https://sites.google.com/site/postechimlab2012/databases/face-database-2001>.

<sup>3</sup> <http://www.itl.nist.gov/iad/humanid/feret>.



For every database, we randomly select  $q$  samples in each class as labeled samples and leave the rest as unlabeled samples.

The adopted graph construction methods are: KNN graph, LNP graph [13], GoLPP graph [20], standard  $\ell_1$  Graph ( $\ell_1$ -s), Robust  $\ell_1$  Graph ( $\ell_1$ -r), constrained  $\ell_1$  graph ( $\ell_1$ -c) [2], LHC graph, SRLS [16], and our proposed construction method ALHC. In each database,  $q$  labeled samples are selected and the label of the rest of the nodes (samples) are estimated using (GFHF) [21] method adopting the constructed graph of every graph construction technique. The process is repeated ten times for ten different combinations of labeled/unlabeled samples and the average classification accuracy is reported. The above process is repeated for three different  $q$  values, corresponding to three numbers of labeled samples.

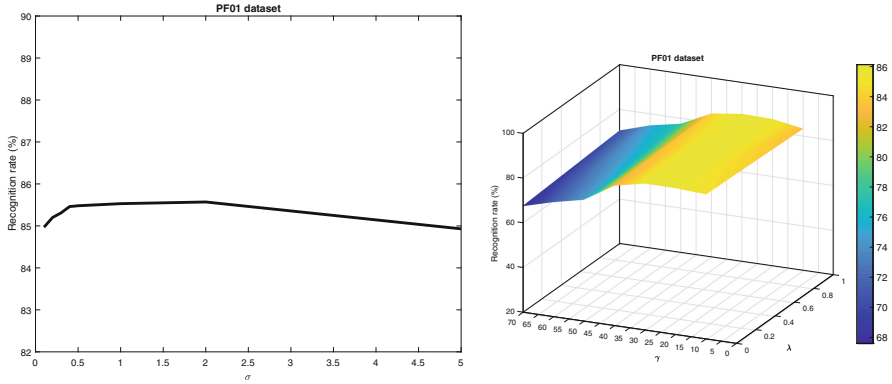
KNN and LNP methods have the neighborhood size parameter  $k$ . The standard and robust  $\ell_1$  graphs have  $\lambda$  ( $\ell_1$ -sparsity). The constrained sparse graph has  $\alpha$  and  $\beta$ . The LHC method has  $\gamma$  (local regularization),  $\lambda$  ( $\ell_1$ -sparsity),  $k_l$  and  $k_s$ . The LLC method has  $\sigma$ . The proposed ALHC method has  $\sigma$ ,  $\gamma$  (local regularization), and  $\lambda$  ( $\ell_1$ -sparsity). In our experiments,  $k$  is chosen from 5 to 60 with a step of 5 for  $k$ NN and LNP graph construction methods.  $\sigma$  is set to one. The  $\ell_1$ -sparsity parameter  $\lambda$  used in  $\ell_1$ -s and  $\ell_1$ -r is fixed to 0.1. For LHC and ALHC, this parameter is chosen from  $\{0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.1, 0.15, 0.2\}$ . The parameter  $\gamma$  is tuned from  $\{0.03, 1\}$ .  $k_l$  is chosen from  $\{10, 20, 30, 40, 50, 100\}$  and  $k_s$  is chosen from  $\{50, 100, 150, 200, 250, 300\}$ . We used the regularization parameters of the  $\ell_1$ -c and SRLS graphs as the ones suggested in [2] and [16].

For every graph construction method, several values for the parameter are used. We then report the best recognition accuracy of each method from the best parameter configuration. Table 1 illustrates the average classification rate in % of label propagation using different graph construction methods for Extended Yale, PF01, and FERET datasets.

We can observe that the proposed ALHC method outperformed other graph construction techniques and obtained the highest accuracy in all databases and different number of labeled samples. It demonstrates that the graph constructed by the proposed method is very informative. Moreover, we can see that the performance of the graph obtained by the proposed method is better than that of standard and constrained  $\ell_1$  graphs and outperforms the three types of sparse graphs.

## 4.2 Sensitivity to Parameters

In this section, we evaluate the sensitivity of the proposed method with respect to the variation of its parameters, namely  $\sigma$ ,  $\gamma$ , and  $\lambda$ . The goal is to study the performance of the proposed method when these parameters vary. The first parameter is a simple regularization parameter in Locality-constrained coding—the phase in which similarities are computed. The last two parameters are two regularization parameters that are used in the hybrid coding scheme where  $\gamma$  penalizes a weighted  $\ell_2$  norm and  $\lambda$  penalizes the  $\ell_1$  norm. Figure 2 (left) illustrates the variation of the recognition rates as a function of  $\sigma$  for the PF01



**Fig. 2.** Performance variation (recognition rate) as a function of the regularization parameter  $\sigma$  (Left) and of the two regularization parameters  $\gamma$  and  $\lambda$  (Right).

dataset. In this experiment, we use 12 labeled samples per class and fixed the other two parameters.

Figure 2 (right) illustrates the variation of the recognition rates as a function of  $\gamma$  and  $\lambda$  for the PF01 dataset. In this experiment, the parameter  $\sigma$  was kept fixed to one since this value seems to be a near optimal value. From the above observations, we can conclude that it is easy to define a near optimal domain for all parameters.

## 5 Conclusion

In this paper, we have proposed a new graph construction method that is based on data self-representativeness. The main contribution of this paper is the adaptive selection of local and non-local bases for the Local Hybrid Coding. The proposed method simultaneously takes into account the locality and sparsity in the graph construction. Thus, the adaptively constructed graph can be very informative.

Experimental results obtained on image databases, demonstrate that in the task of graph-based label propagation, the graph constructed by the proposed method can give better results compared to many state-of-the art graph construction techniques. Currently, we are quantifying the improvement of results through the use of the kernel variant.

## References

1. Araujo, R., Kamel, M.S.: Audio-visual emotion analysis using semi-supervised temporal clustering with constraint propagation. In: Campilho, A., Kamel, M. (eds.) ICIAR 2014. LNCS, vol. 8815, pp. 3–11. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-11755-3\\_1](https://doi.org/10.1007/978-3-319-11755-3_1)

2. Chen, S.B., Ding, C.H., Luo, B.: Similarity learning of manifold data. *IEEE Trans. Cybern.* **99**, 1–13 (2014)
3. Cheng, B., Yang, J., Yan, S., Fu, Y., Huang, T.S.: Learning with L1-graph for image analysis. *IEEE Trans. Image Process.* **19**(4), 858–866 (2010)
4. Dornaika, F., Bosaghzadeh, A.: Adaptive graph construction using data self-representativeness for pattern classification. *Inf. Sci.* **325**, 118–139 (2015)
5. Dornaika, F., Bosaghzadeh, A., Raducanu, B.: Efficient graph construction for label propagation based multi-observation face recognition. In: Salah, A.A., Hung, H., Aran, O., Gunes, H. (eds.) *HBU 2013. LNCS*, vol. 8212, pp. 124–135. Springer, Cham (2013). [https://doi.org/10.1007/978-3-319-02714-2\\_11](https://doi.org/10.1007/978-3-319-02714-2_11)
6. Dornaika, F., Weng, L., Jin, Z.: Structured sparse graphs using manifold constraints for visual data analysis. *Neurocomputing* **315**, 107–114 (2018)
7. Gong, C., Tao, D., Liu, W., Liu, L., Yang, J.: Label propagation via teaching-to-learn and learning-to-teach. *IEEE Trans. Neural Netw. Learn. Syst.* **28**(6), 1452–1465 (2016)
8. Gong, C., Tao, D., Yang, J., Fu, K.: Signed Laplacian embedding for supervised dimension reduction. In: *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence* (2014)
9. Lee, H., Battle, A., Raina, R., Ng, A.: Efficient sparse coding algorithms. In: *Advances in Neural Information Processing System* (2006)
10. Qiao, L., Chen, S., Tan, X.: Sparsity preserving projections with applications to face recognition. *Pattern Recogn.* **43**(1), 331–341 (2010)
11. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**(5500), 2323–2326 (2000)
12. Sprechmann, P., Bronstein, A.M., Sapiro, G.: Learning efficient sparse and low rank models. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(9), 1821–1833 (2015)
13. Wang, F., Zhang, C.: Label propagation through linear neighborhoods. *IEEE Trans. Knowl. Data Eng.* **20**(1), 55–67 (2008)
14. Wang, X., Qian, B., Davidson, I.: On constrained spectral clustering and its applications. *Data Min. Knowl. Disc.* **28**(1), 1–30 (2014)
15. Waqas, J., Yi, Z., Zhang, L.: Collaborative neighbor representation based classification using l2-minimization approach. *Pattern Recogn. Lett.* **34**(2), 201–208 (2013)
16. Weng, L., Dornaika, F., Jin, Z.: Graph construction based on data self-representativeness and Laplacian smoothness. *Neurocomputing* **207**, 476–487 (2016)
17. Wright, J., Yang, A., Ganesh, A., Sastry, S., Ma, Y.: Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(2), 210–227 (2009)
18. Xiang, W., Wang, J., Long, M.: Local hybrid coding for image classification. In: *IEEE International Conference on Pattern Recognition* (2014)
19. Yan, S., Wang, H.: Semi-supervised learning by sparse representation. In: *SIAM International Conference on Data Mining*, pp. 792–801. SIAM (2009)
20. Zhang, L., Qiao, L., Chen, S.: Graph-optimized locality preserving projections. *Pattern Recogn.* **43**, 1993–2002 (2010)
21. Zhu, X., Ghahramani, Z., Lafferty, J., et al.: Semi-supervised learning using Gaussian fields and harmonic functions. *ICML* **3**, 912–919 (2003)