Single Image Super Resolution Based on Sparse Representation via Structurally Directional Dictionaries

Fahimeh Farhadifard

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Electrical and Electronic Engineering

Eastern Mediterranean University
December 2013
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Elvan Yılmaz
Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Aykut Hocanın
Chair, Department of Electrical and Electronic Engineering

We certify that we have read this thesis and that in our opinion, it is fully adequate, in scope and quality, as a thesis of the degree of Master of Science in Electrical and Electronic Engineering.

Prof. Dr. Hüseyin Özkaramanlı
Supervisor

Examining Committee

1. Prof. Dr. Hüseyin Özkaramanlı
2. Prof. Dr. Runyi Yu
3. Assoc. Prof. Dr. Hasan Demirel

ABSTRACT

In this thesis, we propose an algorithm of sparse representation using structurally directional dictionaries to super resolve a single low resolution input image. We have focused on the designing structured dictionaries for different clusters of patches instead of a global dictionary for all the patches. Due to highly directional nature of image content, designing structurally directional dictionaries promises to better capture the intrinsic image characteristics. Furthermore, designing multiple dictionaries with smaller sizes leads to less computational complexity.

The proposed algorithm is based on dictionary learning in the spatial domain. In order to design dictionaries the K-SVD algorithm is used and for this purpose for each of the structured dictionaries a structured training set is prepared. In order to classify the patches into different data sets, a set of templates are designed and each patch is clustered using template matching. Each and every of the templates is modeled according to a specific direction. Then using a similarity measurement, the HR patches and the corresponding features (LR patches) are clustered into directional clusters. Then structurally directional dictionaries are learned by employing the structured training clusters via the K-SVD algorithm. For every cluster two dictionaries are designed: one for the HR patches and the other one for the features.

In the reconstruction part, a LR input image comes in and all the features are coded sparsely with the most suitable directional LR dictionary; and the sparse coding coefficients are then used together with the corresponding HR dictionary to reconstruct the HR patch. In order to choose the best dictionary in sense of direction, a dictionary selection model is needed. Many approaches are tried to find the best dictionary selection method which are mostly error based. But it is not an easy issue while the LR patches (features) are the main criteria to select the most appropriate HR dictionary; it does not always yield to correct selection. However the core idea of the proposed method, designing structurally directional dictionaries, is demonstrated to have superior results compared to the state-of-the-art algorithm proposed by R. Zeyde et.al [23], both visually and quantitatively with an average of 0.2 dB improvements in PSNR over Kodak set and some bench mark images.

Keywords: super resolution, sparse representation**,** structurally directional dictionary.

ÖZ

Bu tez çalışmasında, düşük çözünürlüklü tek bir giriş görüntüsü mükemmel bir şeklilde dönüştürülmek üzere yapısal yönlü sözlükler kullanılarak bir seyrek tasarım algoritması tasarlanmıştır. Çalışmada, tüm parçalar için global bir sözlük yerine, farklı parça kümeleri için yapılandırılmış sözlüklerin tasarlanması üzerinde yoğunlaşılmıştır. Görüntü içeriğinin çok yönlü yapısı nedeniyle yapısal yönlü sözlüklerin tasarımı gerçek görüntü karakteristiklerinin daha iyi bir şekilde ele alınmasını sağlamaktadır. Ayrıca çok daha küçük boyutlara sahip çoklu sözlüklerin tasarımı daha düşük hesaplama karmaşıklığına yol açmaktadır.

Tasarlanan algoritma, uzaysal alanda sözlük öğrenimine dayanmaktadır. Sözlüklerin tasarlanması amacıyla K-SVD algoritması kullanılmış olup bu amaç ile her bir yapılandırılmış sözlük için yapılandırılmış bir çalışma seti hazırlanmıştır. Parçaların farklı veri grupları arasında sınıflandırılması amacıyla bir şablon seti tasarlanmış olup şablon eşleştirmesi kullanılarak her bir parça toparlanmıştır. Şablonların her biri belirli bir yön dikkate alınarak modellenmiştir. Daha sonra bir benzerlik ölçümü yardımıyla, yüksek çözünürlüklü parçalar ve bunlara karşılık gelen özellikler (düşük çözünürlüklü parçalar) yönlü kümelerde toplanmıştır. Daha sonra ise yapısal yönlü sözlükler K-SVD algoritması üzerinden yapılandırılmış çalışma kümelerinden yararlanılarak öğrenilmiştir. Her bir küme için, biri yüksek çözünürlüklü parçalar için ve diğeri özellikler için olmak üzere iki sözlük tasarlanmıştır.

Yeniden yapılandırma kısmında, düşük çözünürlüklü bir giriş görüntüsü giriş yapıp tüm özellikler en uygun yönlü düşük çözünürlüklü sözlük yardımıyla seyrek bir şekilde kodlanmakta ve daha sonra ise, yüksek çözünürlüklü parçanın yeniden yapılandırılması amacıyla, ilgili yüksek çözünürlüklü sözlük ile seyrek kodlama katsayıları birlikte kullanılmaktadır. Yön açısından en iyi sözlüğün seçilmesi amacıyla bir sözlük seçim modeline gereksinim duyulmaktadır. En iyi sözlük seçim yönteminin bulunması amacıyla çoğunlukla hata bazlı olan birçok yöntem denenmiştir. Ancak bu kolay bir konu olmayıp düşük çözünürlüklü parçalar (özellikler) en uyugn yüksek çözünürlüklü sözlüğün temel seçim kriteri olduğu sürece her zaman doğru seçim ile sonuçlanmamaktadır. Ancak yine de tanıtılan yöntemin temel fikri olan yapısal yönlü sözlüklerin tasarımının, model seçiminin her zaman en uygun sözlüğü seçmesi halinde, Kodak seti ve bazı kriter görüntüler üzerinden PSNR'da ortalama 0.2 dB iyileştirme ile hem görsel hem de nicel olmak üzere iki açıdan R. Zeyde et.al [23], tarafından öne sürülen benzer algortimaya üstünlük sağladığı gösterilmiştir.

Anahtar Kelimeler: süper çözünürlük, seyrek gösterim, yapısal yönlü sözlük

Dedication

This thesis is dedicated to my parents whose endless love and support is the most powerful encouragement to progress and overcome all difficulties in my life.

ACKNOWLEDGMENTS

First and foremost I would like to offer my sincerest gratitude to my supervisor, Prof. Dr. Hüseyin Özkaramanlı for the continuous help and support in all stages of my master thesis with his patience, motivation and immense knowledge. I would also like to thank him for being an open person to ideas.

Besides my supervisor, I would like to thank the rest of my committee members, Prof. Dr. Runyi Yu and Assoc. Prof. Dr. Hasan Demirel for their encouragement and insightful comments.

Last but not the least, I would like to thank my family, especially my parents who always believe in me and support me in my decisions. Without their endless love and encouragements, I could not have made it here.

TABLE OF CONTENTS

ABSTRACT iii

ÖZ v

DEDICATION vii

ACKNOWLEDGMENTS viii

LIST OF FIGURES xii

LIST OF TABLES xiii

LIST OF SYMBOLS AND ABBREVIATIONS xiv

1 INTRODUCTION 1

1.1 Sparse Representation 1

1.1.1 Problem Formulation 1

1.1.2 The Matching Pursuit Family 2

1.1.2.1 Matching Pursuit (MP) 3

1.1.2.2 Orthogonal Matching Pursuit (OMP) 3

1.1.2.3 Optimized Orthogonal Matching Pursuit (OOMP) 4

1.1.3 Choice of Dictionary 4

1.1.3.1 K-SVD Dictionary 7

1.1.3.2 Online Dictionary (ONLD) 7

1.2 Application of Sparse Representation Using Dictionary Learning 8

1.2.1 Inpainting 9

1.2.2 Denoising 9

1.2.3 Texture Separation and Classification 10

1.2.4 Image Compression 11

1.2.5 Image Super Resolution 11

2 SUPER RESOLUTION VIA SPARSE REPRESENTATION 14

2.1 Introduction 14

2.1.1 K-SVD Approach 16

2.1.2 Orthogonal Matching Pursuit (OMP) for Calculating the Representation Coefficients of LR Features 18

3 THE PROPOSED SUPER RESOLUTION APPROACH 20

3.1 Introduction 20

3.2 Training Phase 20

3.2.1 Directional Templates 23

3.2.2 Clustering 25

3.2.2.1 Clustering via Dummy Dictionaries 25

3.2.2.2 Euclidean Distance Between Patches and Templates 26

3.2.2.3 Correlation Between Patches and Templates 27

3.3 Reconstruction Phase 28

3.3.1. Dictionary Selection 29

4 SIMULATION AND RESULTS 31

4.1 Introduction 31

4.2 Effect of Patch Size and Number of Dictionary Atoms on the Representation Quality: 32

4.3 Learned Dictionaries 34
4.3.1 Designed Directional Dictionaries Based on Classification via Dummy Dictionaries 34
4.3.2 Designed Directional Dictionaries Based on Classification via Euclidean Distance ……………………………………………………………..36
4.3.3 Designed Directional Dictionaries Based on Classification via Correlation… 37
4.3.4 Performance Test of Designed Directional Dictionaries with Correct Model Selection 39
4.3.4.1 Quantitative Result 40
4.3.4.2 Qualitative Result 43
4.4 Simulation Results of the Reconstruction Phase 45
4.4.1.1 Simulation Results Using Designed Dictionaries Based on Classification via Dummy Dictionaries 46
4.4.1.2 Simulation Results Using Designed Dictionaries Based on Classification via Euclidean Distance 49
4.4.1.3 Simulation Results Using Designed Dictionaries Based on Classification via Correlation ………………………………………………………………49
5 CONCLUSIONS AND FUTURE WORK 53
5.1 Conclusions 53
5.2 Future Work 55
REFERENCES 56

LIST OF FIGURES

Figure 2.1. The K-SVD Algorithm [20] 18

Figure 2.2. The OMP Algorithm [3] 19

Figure 3.1. Directional 6×6 Templates with Shifts 24

Figure 3.2. Designed Dummy Dictionaries, from top left: $0°$ (horizontal), $22.5°$, $45°$, $67.5°$, $90°$ (vertical), $112.5°$, $135°$, $157.5°$. 26

Figure 3.3. Flowchart of Training Phase 28

Figure 3.4. Flowchart of Reconstruction Phase. 30

Figure 4.1. Average Kodak set PSNR vs. dictionary length-width 33

Figure 4.2. Designed HR dictionaries using classification via dummy dictionaries from top left: $0°$(horizontal), $90°$(vertical), 45°, 135°, 22.5°, 67.5°, $112.5°$, $157.5°$and non-directional. 35

Figure 4.3. Designed HR dictionaries using classification via Euclidean distance, from top left: $0°$(horizontal), $90°$(vertical), 45°, 135°, 22.5°, 67.5°, $112.5°$, $157.5°$and non-directional. 37

Figure 4.4. Designed HR dictionaries using classification via correlation; from top left: $0°$(horizontal), $90°$(vertical), $45°, 135°, 22.5°, 67.5°, 112.5°, 157.5°$ and non-directional. 39

Figure 4.5. Visual comparison for zone-plate, from top left insets of: the original, Bicubic, R. Zeyde and the proposed method with perfect model selection 44

Figure 4.6. Visual comparison for Barbara, from top left insets of: the original, Bicubic, R. Zeyde and the proposed method with perfect model selection 45

LIST OF TABLES

Table 4.1. PSNR results, corresponding to Bicubic,.. **Error! Bookmark not defined.**

Table 4.2. Size of Trained Dictionaries.. 46

Table 4.3. PSNR results using classification via dummy dictionaries and correlation based model selection, corresponding to Bicubic, R.Zeyde and proposed method. 48

Table 4.4. PSNR results using classification via dummy dictionaries and Euclidean dictance based model selection, corresponding to Bicubic, R. Zeyde and proposed method.. 48

Table 4.5. PSNR results using classification via Euclidean distance, and correlation based model selection, corresponding to Bicubic, R. Zeyde and proposed method. 50

Table 4.6. PSNR results using classification via Euclidean distance, and Euclidean dictance based model selection, corresponding to Bicubic, R. Zeyde and proposed method.. 50

Table 4.7. PSNR results using classification via correlation, and correlation based model selection, corresponding to Bicubic, R. Zeyde and proposed method. 51

Table 4.8. PSNR results using classification via correlation, and Euclidean distance based model selection, corresponding to Bicubic, R. Zeyde and proposed method. 51

LIST OF SYMBOLS AND ABBREVIATIONS

A	Sparse coefficient matrix
B	Blurring operator
D	Dictionary
E	Error matrix
r^i	Error in the i-th iteration
S	Down-sampling operator
T	Sparsity
u_k	k-th left singular value
v_k	k-th right singular value
X	High resolution training set
x	High resolution patch
x_a	Available pixels
x_m	Missing pixels
Y	Low resolution training set
y	Low resolution patch
α	Sparse coefficient vector
σ_k	k-th Singular value
$\sigma_{X\hat{X}}$	Covariance of X , $\hat{X}$
μ_X	Average of X
Δ	Diagonalized 1/0-value matrix
$\|.\|_0$	Norm zero
$\dagger$	Pseudo inverse

HR	High Resolution
ISD	Image Signature Dictionary
LR	Low Resolution
MAP	Maximum A Posterior
MP	Matching Pursuit
MPF	Matching Pursuit Family
MSE	Mean Squared Error
OMP	Orthogonal Matching Pursuit
ONLD	Online Learned Dictionary
OOMP	Optimized Orthogonal Matching Pursuit
SISR	Single Image Super Resolution
SR	Super Resolution

Chapter 1

INTRODUCTION

1.1 Sparse Representation

In recent years, the sparse representations have become one of the most active areas of research. Many new algorithms have been proposed in a wide range of image processing applications including inpainting, denoising, super resolution and compression. A lot of them are used the advantage of sparse representations and achieve the state-of-the-art results.

The problem of sparse representation consists of two main parts: one is a generally over-complete basis which is called the dictionary and the other one is an algorithm which selects basis vectors which are termed the atoms and the sparse coefficients that are produced in order to approximate an input signal. The algorithm uses only a small number of atoms to represent a signal which leads to the name of sparse representation. In order to use the sparse representation, the signal of interest has to be compressible. A signal is termed compressible when the signal vectors can be represented sparsely with an acceptable error.

1.1.1 Problem Formulation

According to the Sparseland model [1], a signal can be represented sparsely by a vector which has just a few nonzero components. Representation is done over an over-complete dictionary by employing a linear combination of atoms. To be more precise, consider a given over-complete dictionary $\mathrm{D} \in \mathrm{R}^{\mathrm{n} \times \mathrm{k}}$ which consists of k

bases, and a vectorized signal of interest $x \in R^n$. Thus signal can be represented over dictionary D as follows:

$$x = D\alpha \tag{1.1}$$

Such a system is ill-posed and has many solutions. Not only a unique solution is required but also it has to have the minimum number of nonzero components. So the signal x can be sparsely represented as $x \approx D\alpha_0$ where just a few elements in the vector $\alpha_0 \in R^k$ are nonzero ($\ll$ k). The sparse representation can be obtained by solving the following optimization problem:

$$\underset{\alpha}{\operatorname{argmin}} |x - D\alpha| \ \text{subject to} \quad \|\alpha\|_0 < T \tag{1.2}$$

where the l_0 norm $|.|_0$ counts the number of non-zero coefficients in a vector and T defines sparsity of the signal.

In order to solve the vector selection problem above, all possible combinations of the atoms should be tried to find the best one which is not an easy problem to solve. So the complexity of this approach is intractable. In order to find a solution which is more feasible in sense of complexity, various authors have developed different algorithms.

1.1.2 Matching Pursuit Family

One of the well-known family of algorithms which is employed to obtain an approximation solutions of (1.2) is the matching pursuit family (MPF).

The MPF algorithms select a new atom d_i in every iteration i. Considering the selected atoms from first up to iteration i is represented by S^i.

$$S^i \equiv [d_1 \dots d_i] \tag{1.3}$$

The MPF algorithms, iteratively, proceed to solve a simplification of the optimization problem in (1.2):

$$arg\min_{d_i,\alpha^i}\left|x-\left[S^{i-1}\middle|d_i\right]\alpha^i\right|, \tag{1.4}$$

where the coefficients are represented by $\alpha^i = [\alpha_1 \dots \alpha_i]^T$which correspond to each atom d_i. The resulting estimation of x at iteration i is:

$$\hat{x}^i = S^i\alpha^i \tag{1.5}$$

We will talk about three MPF methods which are named matching pursuit (MP), orthogonal matching pursuit (OMP) and optimized orthogonal matching pursuit (OOMP). They are proposed by [2], [3] and [4] respectively.

1.1.2.1 Matching Pursuit (MP)

For the case of the matching pursuit (MP) algorithm, the defined residue vector is in charge of choosing the updated atom-index/ coefficient pair:

$$r^{i-1} = x - \hat{x}^{i-1} \tag{1.6}$$

The MP atom/coefficient selection is done based on the following equations. Where there is the assumption of $r^0 = x$ and at iteration i a single atom representation $\alpha_i.d_i$ is used.

$$arg\max_{d_i}\left|d_i^T.r^{i-1}\right| \tag{1.7}$$

$$\alpha_i = d_i^T.r^{i-1} \tag{1.8}$$

Although the MP coefficient selection criteria enjoys of the low complexity but at the same time it suffers from the problem which says the residual norm never will be zero even for $i \geq d$ (where d is the dimension of x).

$$\hat{x}^i = x \quad \Leftrightarrow i \longrightarrow \infty \tag{1.9}$$

1.1.2.2 Orthogonal Matching Pursuit (OMP)

Orthogonal matching pursuit (OMP) overcomes the above difficulty. The OMP algorithm, at every iteration i uses all the atoms which are selected up from the first, and conduct a subspace of span of all those atoms and then updates all the coefficients at that iteration using the projection onto the subspace.

The OMP atom/coefficient pair selection criterion is as following:

$$arg \max_{d_i} \left| d_i^T . r^{i-1} \right| \tag{1.10}$$

$$\alpha^i = S^{i\dagger} x \tag{1.11}$$

where $S^{i\dagger}$ is pseudo-inverse of S^i. Actually the OMP at every iteration optimizes all the coefficients which are obtained for the previously selected atoms. But MP only considers the residual r^{i-1} in every iteration i and obtains the atom/coefficient pair of only that iteration.

Thus the OMP enjoys from the property that says:

$$\hat{x}^d = x \tag{1.12}$$

1.1.2.3 Optimized Orthogonal Matching Pursuit (OOMP)

An extension of the OMP algorithm idea is optimized orthogonal matching pursuit (OOMP). The OOMP not only modifies all coefficients by looking back to the previous iterations but also selects a better atom in each iteration i. The OOMP using the following atom selection rules, is able to solve (1.4) exactly:

$$arg \max_{d_i} \left| \left[S^{i-1} \left| d_i \right] \left[S^{i-1} | d_i \right]^\dagger x \right| \tag{1.13}$$

$$\alpha^i = S^{i\dagger} x \tag{1.14}$$

The same as OMP algorithm, OOMP again enjoys from the exact reconstruction property (1.12) for $i = d$, where the selection of coefficients α^i are the same as well.

1.1.3 Choice of Dictionary

The signal vectors in the sparse representation problem are considered to be compressible. This is a common assumption when the sparse representation is used. Compressibility of a signal means that the signal vectors can be well-represented using an over-complete dictionary while it uses only a small number of dictionary atoms. Signal compressibility depends on the dictionary which is used to obtain the representation; in order to have dictionaries which are adapted to a particular signal class; various authors have come up with new training algorithms. There are many proposed state-of-the-art algorithms that employ sparse representation using trained dictionaries.

The advantage of using learned dictionary instead of a fixed dictionary shows up when using trained dictionary D, gives low-error approximations $D\alpha$ of the training signal x with sufficiently sparse coefficient α. The representation error for all the training vectors can be obtained as following:

$$E = \|X - DA\|_F^2 \tag{1.15}$$

Let $\|.\|_F$ denote the Frobenius norm of a matrix. And is a set of given training vectors $X = \{x_1, x_2, x_3, \ldots, x_N\}$ And matrix A contains the sparse coefficients α, $A = \{\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_N\}$.

The coefficient matrix A depends on the dictionary D where given a dictionary D, the columns of A are calculated by employing one of the atom selection algorithms such as the matching pursuit family. Thus, dictionary training algorithms are iterative methods which in every iteration calculate one of the quantities D or A while the other one is considered to be fixed.

The approaches which have been proposed so far in order to design such dictionary are two-fold process: i) sparse coding step; given the dictionary, find the sparse coefficients and then ii) updating dictionary while the coefficients assume to be known and fixed. The difference between the algorithms is the method of finding those coefficients and procedure of modifying the dictionary.

Olshausen and Field proposed one of the earliest dictionary training algorithms [5]. In their method, the optimal dictionary is estimated using maximum-likelihood estimation. They assumed a Gaussian or Laplace prior on the sparse representation coefficients while optimal dictionary is estimated. In order to update both dictionary and the sparse coefficients, they employed the steepest descent method.

Method of Optimal Direction (MOD) [6] which is proposed by Engan et.al enjoys of simple dictionary update procedure while uses OMP or FOCUSS algorithms in sparse coding stage. They used overall representation error (1.15) and take the derivative of this error respect to D. Then update the dictionary by forcing the derivative to zero when sparse coefficients are assumed to be fixed ($(X - DA)A^T = 0$). Such a method update dictionary in one step and thus suffers from high complexity.

There are approaches which are proposed to simplify the training task and reduce the complexity. In such methods, dictionaries are considered to be the unions of orthonormal bases [7]. The coefficients of sparse representation A is decomposed to the same number as orthonormal bases and every of them correspond to a different bases.

$$D = [D_1 | \ldots | D_L] \tag{1.16}$$

$$Q = \left[{A_1}^T | \ldots | {A_L}^T\right]^T$$

Such methods enjoy of simplicity of pursuit algorithm needed for sparse coding stage. The proposed algorithms update every orthonormal matrices sequentially using singular value decomposition.

1.1.3.1 K-SVD Dictionary

The K-SVD algorithm is different from discussed approaches above where they freeze A and try to update the dictionary D while the K-SVD is update the dictionary sequentially and let the relevant coefficient change as well.

The joint dictionary learning and sparse representation of a signal can be defined by the following optimization problem:

$$\min_{D,\alpha} \|X - DA\|_F^2 \qquad subject\ to \qquad \forall i, \|\alpha_i\|_0 < T \tag{1.17}$$

where $\|.\|_F^2$ denotes the Frobenius norm. Frobenius norm of a matrix is defined as the square of every element in the matrix. The K-SVD algorithm first uses an initial estimation for dictionary and by employing a pursuit algorithm calculates the best atoms from the current dictionary for representing the data X. Then with the representation coefficients calculated, it updates both the dictionary and the representation coefficient as well. In every iteration, just one atom is replaced in the updated dictionary and it is selected such that reduces the error. In this iterative method, the error of the representation reduces or at the worst situation it remains the same as previous iteration. This approach will be disused in more details in the next chapter.

1.1.3.2 Online Dictionary (ONLD)

In order to have dictionaries which are more suitable to online tasks such as streaming data processing like video, there need to learning algorithms with lower complexity and consequently faster. Such approaches are generally based on

Stochastic Gradient Descent (SGD) methods. These methods instead of training over the entire set as batch, use a single or a small number of training example at a time to process. So they advantage of lower memory requirement and result in faster convergence rates [8].

The Image Signature Dictionary (ISD) proposed by Aharon et.al [9] is one of those methods. Based on this approach, in order to have more compact dictionary some of the dictionary redundancies are sacrificed. Thus such a method enjoys from reduced training complexity and consequently becomes an interesting approach for online tasks. Based on this approach, dictionary is represented as a small image of N pixels when size of dictionary assumed to be $d \times N$. And every one of N pixels of the ISD is surrounded by dictionary atoms $d_k \in R^d$ as a block. The same as before, dictionary update is a two-step process where either the sparse representations α or the dictionary D are getting update when the other one is fixed.

Based on the work of Mairal et.al [8] [10] a variation of this approach is proposed. According to this method, at the time t, x_t is used together with the D_{t-1} obtained from previous iteration and the sparse coefficient α_t is computed. Dictionary update is done based on the block-coordinate descent.

$$D_t \equiv arg \min_D \frac{1}{t} \sum_{i=1}^{t} \frac{1}{2} \|x_i - D_{t-1}\alpha_i\| + \lambda \|\alpha_i\|_1 \tag{1.18}$$

1.2 Application of Sparse Representation using Dictionary Learning

There are wide ranges of sparse representation applications and in this section we briefly describe some of them.

1.2.1 Inpainting

Image inpainting is a useful application in several scenarios of image processing. This application is used to fill in pixels which are missed in the image. It is used in the data transmission in order to provide an alternative for the channel codes [11], [12] and also in image manipulation to remove the superposed text, road-signs or publicity logos [13].

Considering an image patch $x = [x_a^T \ x_m^T]^T$ which is made of two sub-vectors, sub-vector x_a is defined for the available pixels in the patch and x_m contains the missing pixels where image inpainting is used to estimate. Guleryuz [14] proposed a method to estimate the missing sub-vector x_m. They approximate the missing data employing a concatenation of orthonormal bases that render y compressible. Compressibility of a signal x means that given D, there exists some sparse vector α which satisfies $x = D\alpha$. Consider the diagonal matrices Δ_n, Δ_a *and* Δ_m which are diagonalized with 1/0-value and are used to define the non-zero entries of α, the available entries and missing entries of x respectively. Thus the i-th estimate of x can be expressed as:

$$\hat{x}^i = \begin{pmatrix} x_a \\ \hat{x}_m^i \end{pmatrix} = \Delta_a \hat{x}^{i-1} + \Delta_m (D \Delta_n D^T) \hat{x}^{i-1} \tag{1.19}$$

1.2.2 Denoising

Sparse representations have also used to denoise the images and videos [1], [15]. The denoising problem is formulated as a MAP estimation problem when a sparse prior is considered on the data. By obtaining the sparse estimation of image blocks which are overlapped, the solution of MAP approximation is defined; and then the denoised data is specified by averaging over all blocks.

Consider a noisy image Y of size $R \times d$ all overlapping patch y_k are extracted and then reshaped to the vectors. Then by employing an over-complete dictionary D using K-SVD, all the patches are sparsely coded using OMP. The best atom d_k is represented the meaningful parts y_k so the noisy of these parts of y_k are discarded.

$$z_i = D.\alpha_i \tag{1.20}$$

At the end all the denoised patches are reshaped to the 2-D patches. Then by averaging pixel value in the overlapping patches and merging them, the denoised image is obtained. This overlapping also suppresses the noise.

1.2.3 Texture Separation and Classification

There are a series of works using the application of sparse representations to texture separation [16], [17], [18]. According to this application every image block x is assumed to consist of a mixture of overlapping component layers u_k:

$$x = \sum_k \alpha_k u_k \tag{1.21}$$

This problem can be adapted by sparse representation if assume an available dictionaries $D_k$which can render u_k compressible. In order to use this tool, two issues come up: (i) forming the D_k and then (ii) using them in order to separate the various layers of the image.

Peyre et.al [18] proposed such an approach. According to this approach, two kinds of dictionaries are used for edge/contours textures layers (named cartoon layers) and one for more complex texture layers (oscillatory texture layers). The cartoon layers are modeled using a combination of off-the-shelf dictionaries (fixed for all the image patches) and also the learned, adaptive dictionaries are employed in order to sample the more complex texture layers. Then solve the problem in three stages:

(i) Using linear programming methods to find sparse coefficient α.

(ii) Employing conjugate gradient descent in order to solve texture layers.

(iii) Solving the adaptive dictionary using gradient descent.

1.2.4 Image Compression

Recently in the case of image compression, using learned over-complete dictionaries which are adapted to a signal class result to successful result. The advantage of using a learned dictionary which is very beneficial for the image encoder is the grater compressibility of the consider signal class. An example of a such approach is the work which is introduced by Bryt and Elad [19] based on the learned K-SVD dictionary [20]. In their approach, they use some pre-specified face templates. Each of the face templates which are not overlapped with the others, are used in order to specify a class of the signals. And then they are represented employing the corresponding K-SVD dictionary. The discussed approach, results in a wide PSNR improvement over the state-of-the-art JPEG2000 [21] algorithm but at the same time, it suffers from the expense of a large storage for the dictionaries.

1.2.5 Image Super Resolution

The problem of super resolution is an important active area of research, due to wide demand for high resolution image in many applications. Obtaining a high-resolution (HR) image from single or multiple low-resolution (LR) images, that have lost higher frequency information during acquisition, transmission or storage, is known as "super-resolution". Conventional SR methods, require using several LR images to reconstruct the HR one. In situations where the number of available input images is small, these algorithms are not practical. In such cases Single Image Super Resolution (SISR) becomes more important.

The most recent and successful approach to this purpose uses sparse representation to enhance the quality of an image. This approach will be discussed in more detail in the next chapter.

Previously, J. Yang et.al [22] and R. Zeyde et.al [23], have proposed their approaches to super resolve a single LR image via sparse representation.

The method is proposed by J. Yang et.al [22], [24] uses a set of HR images, then by down sampling and blurring operators the corresponding LR images are obtained. They subtract mean value for each patch and then use them in order to learn. A pair of high and low resolution dictionary is trained. There is a main assumption which considers the same sparse representation for both HR and LR patches. At the end, HR patch is reconstructed by multiplying HR dictionary with the sparse representation of corresponding LR patch.

R. Zeyde et.al [23] use the basic idea of J. Yang. They also assume that HR and LR patches have the same sparse representation. They try to super resolve the high frequency components of an image which is most difficult part where Bicubic interpolation is not successful. They learnt both HR and LR dictionaries over high frequency components of the patches and features using KSVD algorithm. Then in the reconstruction part by employing OMP algorithm, the sparse representation of LR patches is found and then using sparse representation coefficient together with HR dictionary the HR patches are recovered.

In this thesis we use J. Yang's idea about the same sparse representation for both HR and LR patches and extend the concepts by designing structurally directional dictionaries and apply them to the process of SR. The core difference is that instead of learning a single dictionary for all the patches, we have trained eight pairs of structured dictionaries for different class of patches. It has been shown in [25] that designing multiple dictionaries is more beneficial than a single one; furthermore, in [26] it is pointed out that designing several dictionaries using clustering improves both quality and computational complexity.

The approach is a two-fold scheme, the dictionary training phase and reconstruction phase. In the first stage, in order to learn structurally directional dictionaries, we first form structurally directional training sets. The proposed method is template matching based; thus for this purpose, eight sets of directional templates are designed. For every direction, the corresponding templates including all shifted versions are modeled. Then both HR and LR training sets where the LR version is constructed using down sampling and blurring operators, together with the templates are employed to cluster the corresponding patches. The classification is done using a similarity measurement. We also consider a cluster for non-directional patches. Then for every cluster, two dictionaries are learned by employing the K-SVD algorithm; one for LR features and the other one for the HR patches. In the next step, reconstruction phase, an input LR image comes in and after extracting patches, the most proper dictionary pair is selected to reconstruct the corresponding HR patch. Dictionary selection criterion is based on the error in representation of LR features via LR dictionaries. The OMP algorithm is used in the reconstruction part to find the sparse representation coefficients and then super resolve the LR patches.

Chapter 2

SUPER RESOLUTION VIA SPARSE REPRESENTATION

2.1 Introduction

Obtaining a high-resolution (HR) image X from the single low-resolution (LR) image Y is known as "single image super-resolution (SISR)". The LR image is a version of the HR image which has lost its higher frequency information during acquisition, transmission or storage. In order to solve such a problem which has so many solutions, two constraints are assumed: (i) reconstruction constraint: Based on image observation model, reconstructed HR image X should be in agreement with the LR image; and (ii) sparsity prior: HR image can be sparsely represented over an over-complete dictionary and it can be recovered from the LR version.

To be more precise, consider LR image Y which is down sampled and blurred version of HR image X. And assume that there is a HR over-complete dictionary $D_h \in R^{n\times k}$ of k bases which is a large matrix learned using HR image patches. Then the vectorized patches of image X, $x \in R^n$ can be sparsely represented over dictionary D_h. So the high resolution patch x can be represented as $x = D_h\alpha_0$ where $\alpha_0 \in R^k$ is a vector with very few nonzero elements ($\ll k$). The relationship between a HR patch x and its LR counterpart y can be expressed as:

$$y = SBx = Lx = LD_h\alpha_0 \tag{2.1}$$

Note that S is representing a down sampling operator, B represents a blurring filter and L denotes their combined effect. Substituting the representation for the HR patch x into (2.1) and noting that $D_l = LD_h$, one gets:

$$y = LD_h\alpha_0 = D_l\alpha_0 \tag{2.2}$$

Equation (2.2) implies that the LR patch y will also has the same sparse representation coefficients α_0. Now given the LR patches, one can obtain the representation coefficients using a vector selection such as OMP.

After obtaining the sparse coefficients, one can reconstruct the high resolution patch x.

$$\hat{x} = D_h\hat{\alpha}_0 \tag{2.3}$$

The sparse representation problem (vector selection) has the formulation as an optimization problem which results in finding the sparse coefficient α using dictionary D_l. For obtaining the sparse representation coefficients for the LR patch y, one solves the following optimization problem:

$$\min_{\alpha_0} \|y - D_l\alpha_0\|_2 \quad subject\ to \quad \|\alpha_0\|_0 < T \tag{2.4}$$

where T is a threshold which is used to control the sparseness of the representation. The l_0 norm is used to identify the number of nonzero elements of the vector α_0. An error based formulation of the vector selection problem can also be employed.

In order to represent the signal of interest, a suitable dictionary and a sparse linear combination of the dictionary atoms is needed. The sparse representation problem subject to find the most proper selection of those linear combination vectors from an over-complete dictionary D_l. To find such a representation different pursuit algorithm can be used such as OMP and the over-complete dictionary can be formed using K-SVD.

2.1.1 K-SVD Approach

As it was mentioned previously, an over complete dictionary together with the sparse coefficients are needed to represent a signal. The joint dictionary learning and sparse representation of a signal can be defined by the following optimization problem:

$$\min_{D_l,\alpha} \|Y - D_l A\|_F^2 \quad \mathit{subject\ to} \quad \forall i, \|\alpha_i\|_0 < T \tag{2.5}$$

Consider a set of over-complete basis vectors Y, and an initial dictionary which is formed by choosing its elements from the set randomly, D_l. In order to find the sparse coefficients of such a set over the dictionary, one the dictionary is assumed to be fixed and then the sparse coefficients are calculated using OMP by solving following optimization problem for each and every input signal y_i

$$\min_{\alpha_i} \|y_i - D_l \alpha_i\|_2^2 \quad \mathit{subject\ to} \quad \|\alpha_i\|_0 \quad i = 1,2,\dots,N \tag{2.6}$$

Since the K-SVD algorithm attempts to update dictionary by replacing one atom at a time to reduce the error in representation, thus in every iteration, the dictionary and effective sparse coefficient vectors are considered to be fixed and just one atom in the dictionary is questioned to be replaced and the corresponding sparse coefficient is calculated.

For this purpose, the objective function (2.5) is written as following:

$$\|Y - D_l A\|_F^2 = \left\| Y - \sum_{j=1}^{K} d_l^j \alpha_j \right\|_F^2 = \left\| (Y - \sum_{j \neq k} d_l^j \alpha_j) - d_l^k \alpha_k \right\|_F^2 \tag{2.7}$$
$$= \left\| E_k - d_l^k \alpha_k \right\|_F^2$$

where the E_k is the overall error matrix for training signals when the k-th atom of the dictionary d_k and its corresponding coefficient α_k is not involved. Then all the signals which use atom d_k and coefficient α_k in their representation form a matrix $\{Y\}^k$ and the corresponding error matrix $\{E_k\}^k$ is then obtained. Now (2.7) is rewritten as (2.8) and dictionary can be updated by minimizing it:

$$\left\|\{E_k\}^k - d_l^k\{\alpha_k\}^k\right\|_F^2 \tag{2.8}$$

where $\{\alpha_k\}^k$ is the k-th row of matrix coefficients which its zero entries are discarded. Using singular value decomposition (SVD), $\{E_k\}^k$ is decomposed to $\{E_k\}^k = U\Delta V^T$. Solution is the vectors corresponding to the maximum value which modifies the updated atom $\hat{d}_l^k$ as normalized version of the first column of U ($u_1/\|u_1\|_2$) and $\{\alpha_k\}^k$ as the first column of V multiplied by $\Delta(1,1)$, ($s_1 v_1$).

The K-SVD algorithm is summarized in the following flowchart:

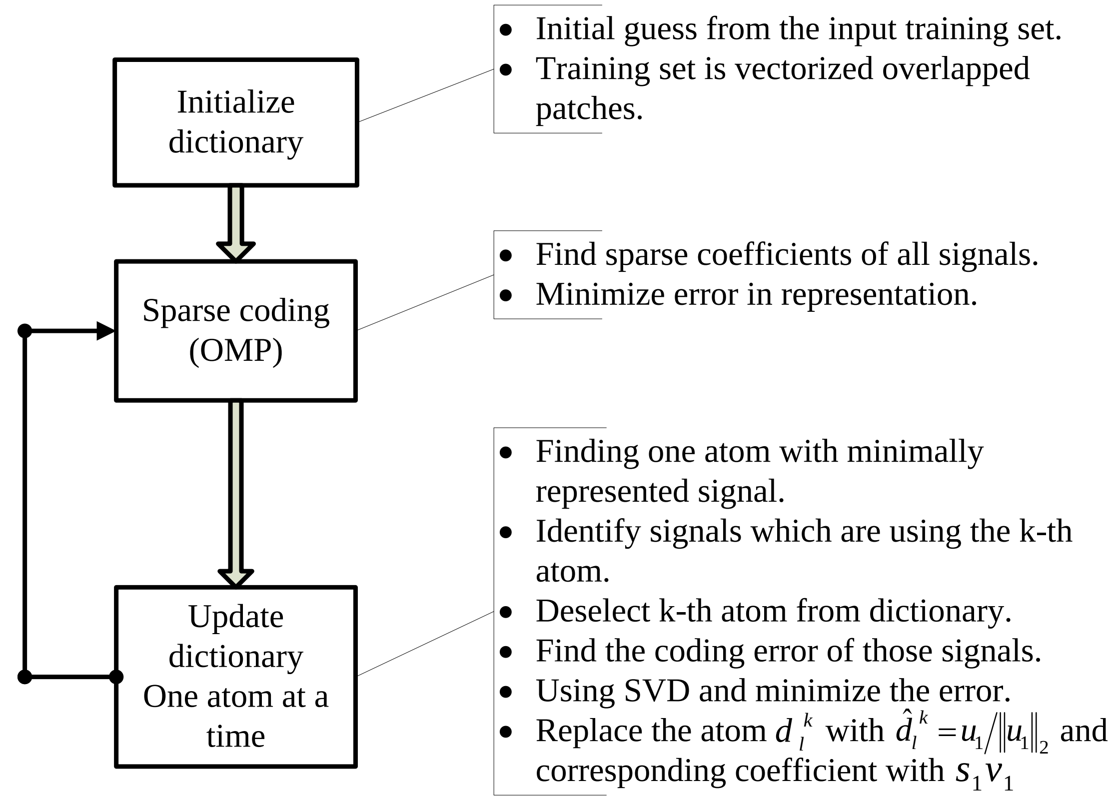


Figure 2.1. The K-SVD Algorithm [20]

2.1.2 Orthogonal Matching Pursuit (OMP) for Calculating the Representation Coefficients of LR Features

As it was mentioned before, finding an exact sparse representation of a signal is not easily achievable. As the result, many researchers have aimed to find the best approximate solution. Among all the methods Orthogonal Matching Pursuit has been the main choice. The OMP is a simple method which enjoys from the fast running time.

Given a dictionary, the OMP as a greedy algorithm, aims to find sparse representation of the signals of interest over that dictionary. It is an iteratively algorithm which updates the basis vector in every iteration and as the result reduces the error in the representation. According to this scheme, the dictionary atoms with the largest absolute projection on the error vector are selected. This results into selection of atoms which contain maximum information and consequently reduce the error in the reconstruction. According to the Figure (2.2), the OMP algorithm selects the code vector α in three steps, by given signal y and dictionary D_l:

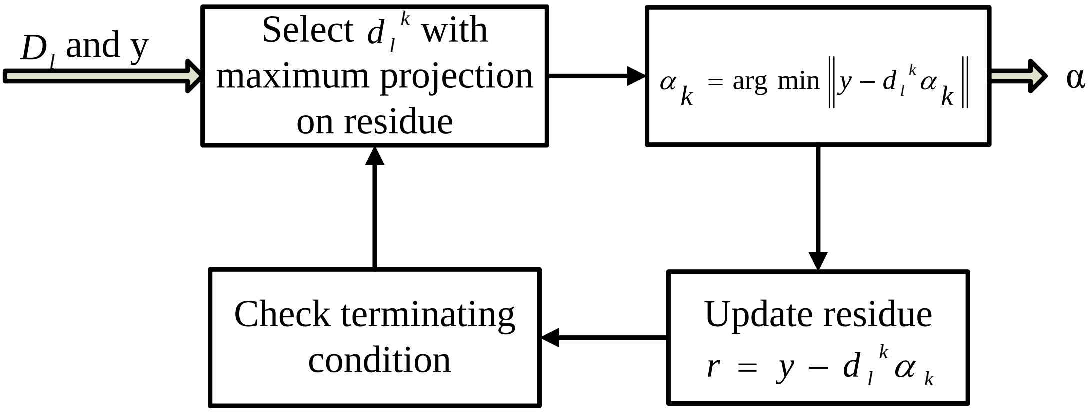


Figure 2.2. The OMP Algorithm [3]

Chapter 3

THE PROPOSED SUPER RESOLUTION APPROACH

3.1 Introduction

In order to have better capture of the intrinsic image characteristic we have focused on the designing structured dictionaries instead of designing a global dictionary for all kind of patches. As we know image content are highly directional. In order to have more proper dictionaries to reconstruct directional patches, structurally directional dictionaries are learnt. Another advantage of such an approach is less computational complexity due to the fact that structural dictionaries can be much smaller than a global dictionary.

Our proposed algorithm consists of two main parts, dictionary training and reconstruction HR image from the LR image. We have learnt several sets of structurally directional high and low resolution dictionaries in training part and then we use them in the reconstruction part to recover a HR image from the LR version.

3.2 Training Phase

This part starts by collecting a set of HR images. Then the LR version of all those images are constructed by using down-sampling and blurring operators. To reach destination size, LR images are scaled up to the size of HR image via Bicubic interpolation. This scaling serves mainly for rendering the coding part easy.

The main focus in this phase is learning the most appropriate dictionaries to reconstruct edges and texture content of an image accurately. The edges and texture regions constitute the high frequency components of an image. To follow this idea, in order to learn the HR dictionaries from the high components only, the HR images are subtracted from the mid-resolution one (LR image which are scaled up to the size of HR image termed mid-resolution image); Then local patches are extracted and vectorized to form the HR training set X.

In order to ensure that the reconstructed HR patch is the best estimation, the calculated coefficients must fit to the most relevant part of LR signal. Thus the Laplacian and Gradient high-pass filters are employed to remove low frequency content of LR images similar to the approach in [22], [27]. This choice is reasonable while people are sensitive to the high frequency component of an image.

Instead of applying high-pass filters on the image patches which results in boundary problems due to small patch sizes [28], we first filter whole the image using four 1-D filters (3.1) to extract first and second derivatives as the features for the low-resolution patch. Then local features corresponding to the gradient maps are extracted and reshaped to the vectors. Features in the same location are concatenated to form a big vector as a feature for LR training set Y.

$$f_1 = [-1, 0, 1] \qquad f_2 = f_1^T \tag{3.1}$$
$$f_3 = [1, 0, -2, 0, 1] \qquad f_4 = f_3^T$$

To have more detailed result, we decided to have more than one pair of dictionary; a pair of suitable dictionary for every specific direction. To cover 2D space we chose eight directions, one pair of dictionary for each and every 22.5 degree. In order to have an exact model for every direction, some templates are modeled to cover all possibilities of a single direction. All patches and features are clustered to eight different sets based on their similarity to those templates.

Before training dictionaries, a dimensionality reduction algorithm is applied over LR features. The basic idea of dimensionality reduction technique is projection; so using dimensionality reduction results into employing only the real dimensions in the data and in our case saves computations in the training phase and super resolution algorithm. Among all proposed methods for this purpose, the Principal Component Analysis (PCA) is used. This method preserves 99.9% of patches average energy.

At the end, For every cluster, the features from the LR training sets $Y_m \subseteq R^{\acute{N}}$ are given to the KSVD algorithm and then the corresponding LR dictionary is trained $\{D_l\}_m \in \mathbb{R}^{\acute{N}\times M}$:

$$\{D_l\}_m, \{\alpha^k\}_m = arg \min_{\{D_l\}_m, \{\alpha^k\}_m} \sum_k \|\{y^k\}_m - \{D_l\}_m \{\alpha^k\}_m\|^2 \tag{3.2}$$

$$s.t. \|\{\alpha^k\}_m\|_0 \leq T \ \ \forall k.$$

where α^k is the sparse representation coefficient vectors which belong to the feature y^k. And index k defines the k-th feature in each training set Y_m and index m is used to determine the corresponding cluster. After learning all directional and non-direction LR dictionaries, each and every HR dictionaries are calculated from the HR training sets X_m together with the sparse coefficients of corresponding LR dictionary for every cluster m.

Noting that:

$$\{x^k\}_m = \{D_h\}_m . \{\alpha_h^k\}_m \tag{3.3}$$

And using the fact that:

$$\{\alpha_h^k\}_m \approx \{\alpha_l^k\}_m \tag{3.4}$$

We obtain:

$$\{x^k\}_m \approx \{D_h\}_m . \{\alpha_l^k\}_m \tag{3.5}$$

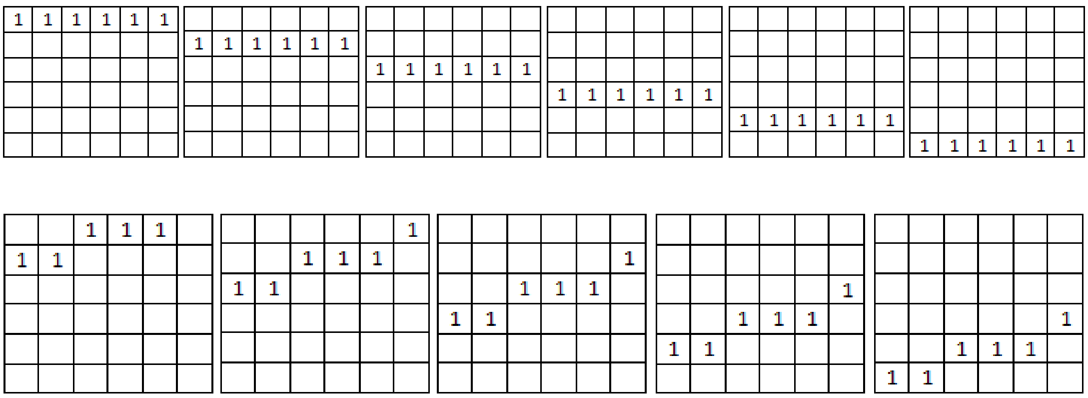
$$\{D_h\}_m = X_m (A_m)^T (A_m (A_m)^T)^{-1} \tag{3.7}$$

where A_m is the matrix coefficients contains all coefficients vector obtained from the LR dictionaries for each cluster m.

All the steps to classify training sets are explained in the following.

3.2.1 Directional Templates

In order to have an exact model of any direction to classify the patches, we have designed eight sets of templates. Each and every one corresponds to a direction and consists of all shifted versions of that direction. Using all those eight directions we covered two-dimensional space. Figure 3.1 shows the designed templates. Templates are at the size of 6 by 6 which are proper to compare with HR patches with the same size From top to the end templates are: 0°(horizontal), $22.5^\circ, 45^\circ, 67.5^\circ, 90^\circ$(vertical), 112.5°, $135^\circ, 157.5^\circ$.



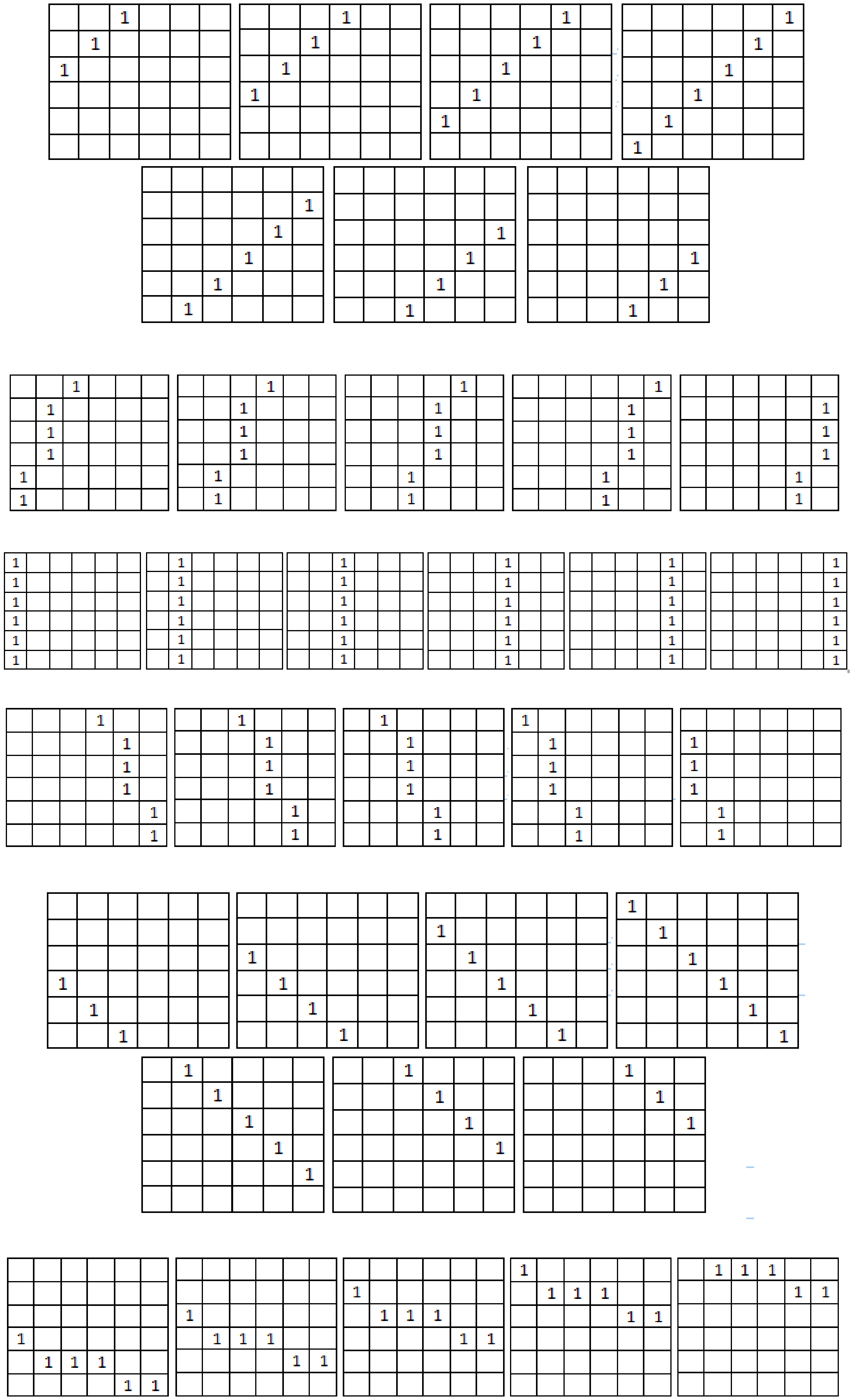


Figure 3.1. Directional 6 × 6 Templates with Shifts.

3.2.2 Clustering

According to the purpose of designing structurally directional dictionaries, it is needed to have directional training data set for each and every of dictionaries. To classify patches and features corresponding to HR and LR image patches respectively, we tried three different approaches to find the best one in order to have groups of patches and features with the same direction of image content: using dummy dictionaries, Euclidean distance and Correlation.

Using all training data, we collected 6 by 6 patches of high resolution images, and corresponding LR features from mid-resolution images which are the scaled up version of LR images.

3.2.2.1 Clustering via Dummy Dictionaries

Using templates of every direction and their shifted versions, we constructed non-repeated linear combination of them in order to have a big matrix of all possibilities of every direction as a dummy dictionary (Figure 3.2). Then by employing OMP method which chooses the best atoms in the dictionary iteratively, coefficients of every HR patch are found using all dummy dictionaries, then the corresponding patch is recovered by all those dictionaries.

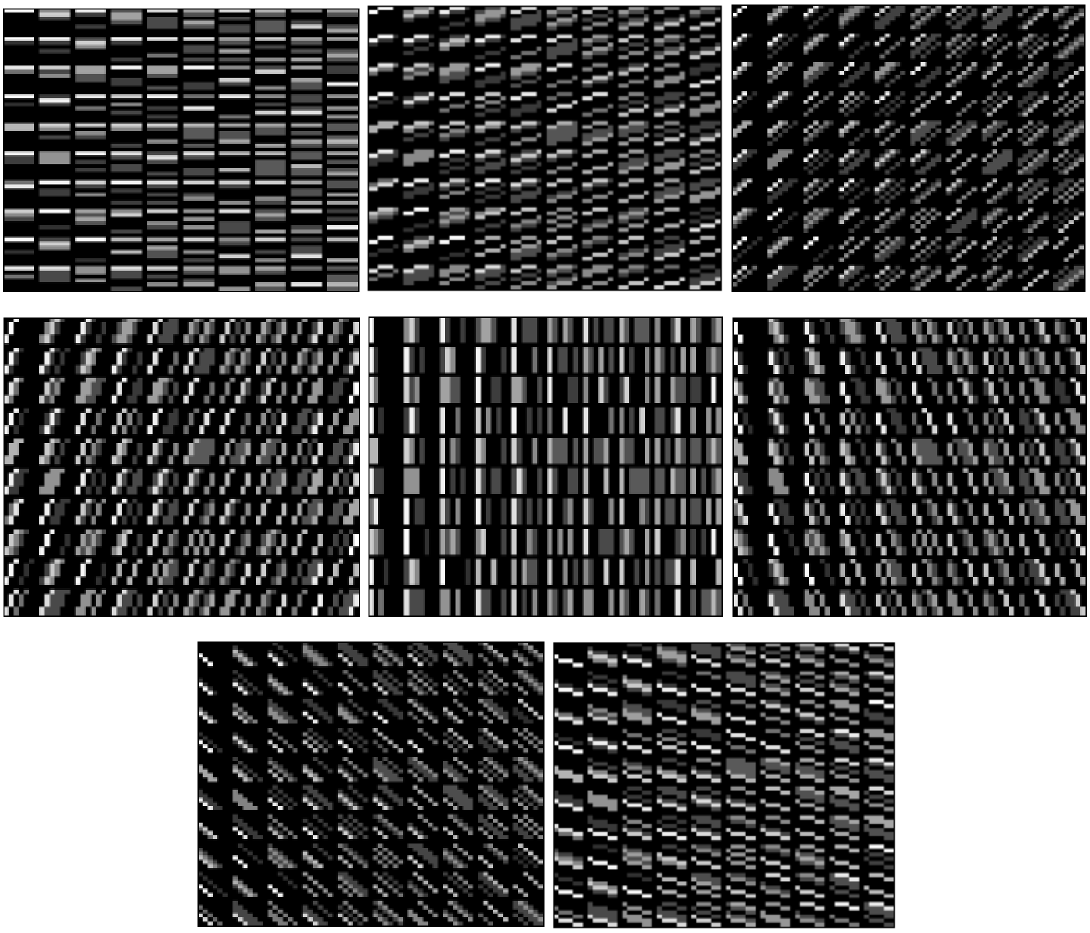


Figure 3.2. Designed dummy dictionaries, from top left: 0°(horizontal), $22.5^{\circ}, 45^{\circ}, 67.5^{\circ}, 90^{\circ}$(vertical), 112.5°, $135^{\circ}, 157.5^{\circ}$.

The amount of error between the original patch and the reconstructed ones is obtained and the minimum error is used to determine which group the patch in question belongs to. In order to have a non-directional dictionary for the patches which are less directional, we decided to have a threshold for every cluster which is chosen according to the error histograms. Those patches that did not belong to any of the directional clusters are designated to belong to the non-directional cluster.

3.2.2.2 Euclidean Distance Between Patches and Templates

According to this approach, reshaped HR patch to the vector of size 36 by 1 together with vectorized templates of the same size are used to find the most similar template to the patch. The criterion for clustering is Euclidean distance; thus the Euclidean distances between the HR patch and all templates are obtained and the minimum value defines the cluster which the patch in question and corresponding feature (LR patch) belongs to. A non-directional cluster is defined which contains those patches with the distance bigger than a specific threshold.

3.2.2.3 Correlation Between Patches and Templates

Based on correlation, the vectorized HR patch is correlated by the reshaped templates of the same size; and the first choice which has the biggest value concatenates the HR patch and corresponding feature to its own template cluster. We use correlation to show the similarity between the patches and templates and as our templates are directional this comparison gives us the most directional patches. The same as the previous method, a non-directional cluster is considered. A single threshold decides about the directional or non- directional nature of the patches; if a patch was directional then structurally clustering starts, if not it goes to non-directional category. Figure 3.3 summarizes the basic steps in the training phase.

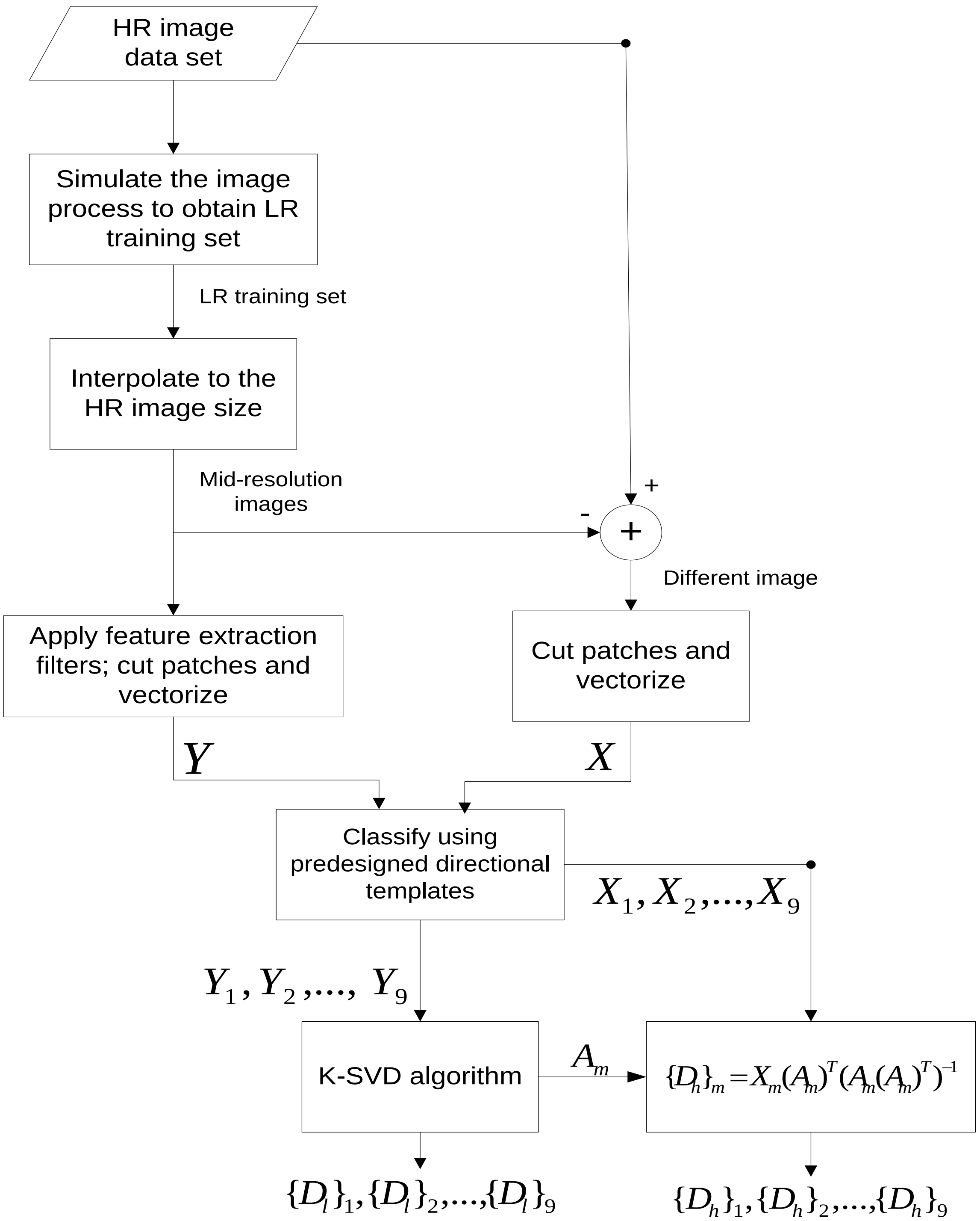


Figure 3.3. Flowchart of Training Phase.

3.3 Reconstruction Phase

Reconstruction phase should conform to the training phase. In that sense, steps reminiscent of the ones used in the learning steps is encountered. Given a LR image to be reconstructed, it first needs to be rescaled to the same size as the HR image. This is done by Bicubic interpolation. Using this so called mid-resolution image, we first filter it to extract the meaningful features in exactly the same way that the learning stage did. Then the features are extracted to be recovered using the most suitable dictionary.

In order to have the best reconstruction result, we need proper criteria for dictionary selection. For this purpose, we tried two out of three approaches which we used in the training phase for classification. These are correlation and Euclidean distance between features and templates. Then the chosen LR dictionary and OMP algorithm are used to sparsely represent the feature. Then the HR patch is recovered using corresponding HR dictionary together with the sparse representation coefficients of the feature.

3.3.1. Dictionary Selection

The most important issue in reconstruction part is dictionary selection criteria. The two most proper approaches are used to come up with the best result; the first one is correlation.

The correlation between the LR feature to be super resolved and the designed templates is first calculated. The template that gives the highest correlation determines the LR dictionary to be used. Using the OMP algorithm and the selected LR dictionary, the sparse representation coefficients are calculated. The same sparse representation coefficients are then used together with the HR dictionary to reconstruct the HR patch.

The second approach is Euclidean distance between LR features and templates; according to this approach, every LR feature will be reconstructed by all the LR dictionaries and the error between the reconstructed feature and the original one is found.

$$\{\hat{y}_l^k\}_m = \{D_l\}_m . \{\alpha_l^k\}_m \tag{3.8}$$

$$\{e_k\}_m = y_l^k - \{\hat{y}_l^k\}_m \tag{3.9}$$

The one with least error is used to determine which directional dictionary one needs to use for reconstructing the HR patch. Figure 3.4 shows a summary of the reconstruction stage.

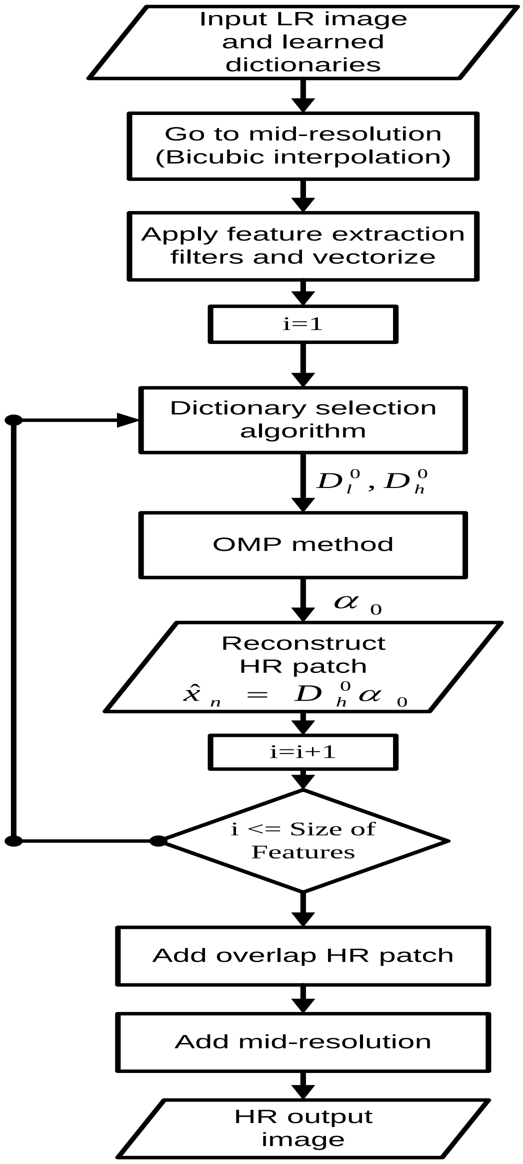


Figure 3.4. Flowchart of Reconstruction Phase.

Chapter 4

SIMULATION AND RESULTS

4.1 Introduction

In this chapter, the performance of the proposed method is evaluated by simulation. We show the result of applying our method on the Kodak set and some benchmark images in two quantitative and qualitative subsections. Our approach is compared to Bicubic interpolation and the state-of-the-art method proposed by R. Zeyde et.al [23]. The super resolution is done herein scales up test images from a 384×256 dimension to a 768×512 dimension mostly for both our proposed method and baseline algorithm [23].

We first conduct an experimental study which shows the ability of designed dictionaries to super-resolve images using different setting for patch size and dictionary redundancy. Then using the optimal patch size and redundancy, further tests are performed.

In order to show the quantitative performance, the Peak Signal-to-Noise Ratio (PSNR) is used.

$$PSNR(X,\hat{X}) = 10\ log_{10}\frac{255^2}{MSE(X,\hat{X})} \tag{4.1}$$

where X is true image and $\hat{X}$ is the estimated version of it and $MSE(X,\hat{X})$ denotes the mean square error between X and $\hat{X}$ and is defined by:

$$MSE(X,\hat{X}) = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{ij} - \hat{x}_{ij})^2 \tag{4.2}$$

Both X and $\hat{X}$ are 8-bit (gray level) with $M \times N$ pixels. To measure the perceptual image quality, the structural similarity index measure (SSIM) is used. The SSIM is believed to be more fit with human perception rather than PSNR.

$$SSIM(X,\hat{X}\,) = \frac{(2\mu_X\mu_{\hat{X}} + c_1)(2\sigma_{X\hat{X}} + c_2)}{(\mu_X^2 + \mu_{\hat{X}}^2 + c_1)(\sigma_X^2 + \sigma_{\hat{X}}^2 + c_2)} \tag{4.3}$$

where $\mu_X, \mu_{\hat{X}}$ are the average of X and $\hat{X}$ respectively and $\sigma_X^2,\ \sigma_{\hat{X}}^2$ are the variance of X and $\hat{X}$ respectively and $\sigma_{X\hat{X}}$ is the covariance of $\hat{X}, X$. Two variables c_1, c_2 are used to stabilize the devision.

4.2 Effect of Patch Size and Number of Dictionary Atoms on the Representation Quality:

According to [29] the dictionary redundancy is an important concept in sparse representation. The sparse representation problem is a patch-based method; the redundancy of the dictionary in the problem is defined by the ratio of the number of dictionary atoms to the patch size. It is expected that choosing a large patch size helps to better represent the image structures. But at the same time, larger patch sizes result in larger dictionary atoms. In addition, to learn dictionaries with larger patch sizes, one needs significantly bigger training set of images. Therefore, although representation can enjoy from larger patch sizes but it suffers from increased computation complexity as well.

Figure 4.1 illustrates the average PSNR of reconstructed Kodak set images based on different settings of the aforementioned two parameters. The horizontal axis represents the redundancy as the ratio of the number of dictionary columns (atoms) to the number of dictionary rows (patch size), and the vertical axis represents the

PSNR performance for four different HR patch sizes 4×4, 6×6, 8×8, 10×10 (which are correspond to the 2×2, 3×3, 4×4, 5×5 for LR image patches respectively). Dictionaries are trained using the K-SVD algorithm with sparsity parameter S=3, and 20 iterations. Then for each image in the Kodak set, the reconstruction algorithm is employed and the average PSNR is plotted for three different patch sizes. The images which are used in the reconstruction part are not the same as training part (Test image is not involved in training set). Thus the ratio between the number of atoms in the dictionary and the patch size is defined as the dictionary redundancy. The vectorized HR patch sizes are 16×1, 36×1, 64×1, 100×1.

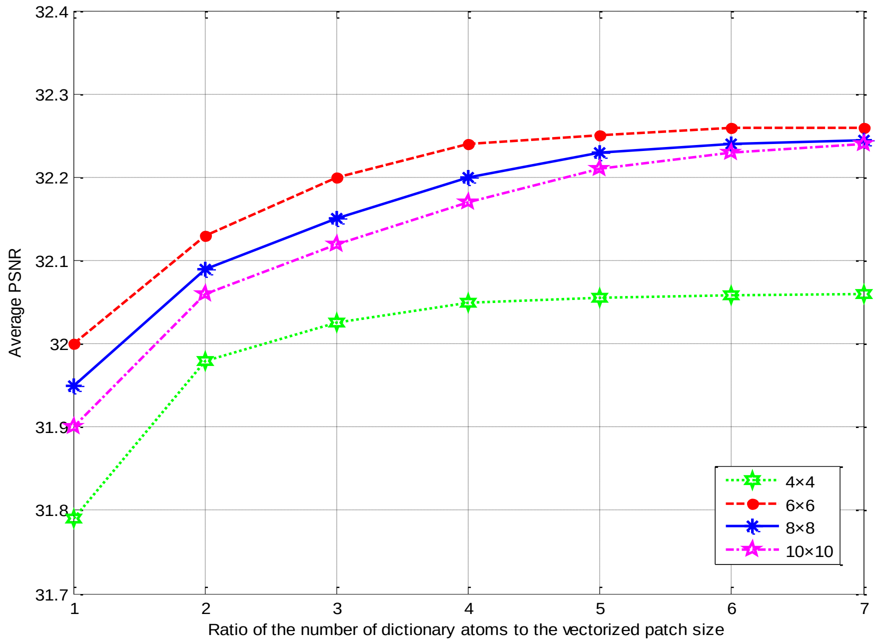


Figure 4.1. Average Kodak set PSNR vs. dictionary length-width using four different HR patch sizes, 4×4, 6×6, 8×8, 10×10.

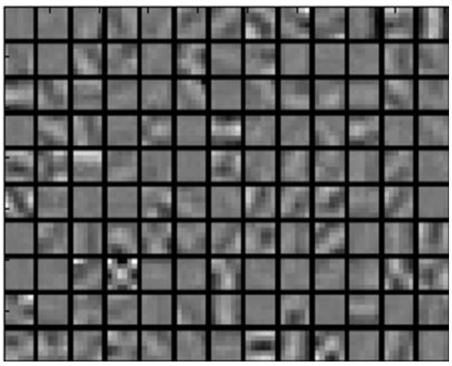
According to the Figure 4.1, for all patch sizes increasing the size of dictionary atoms improves the reconstruction performance rather than using a complete dictionary. It can be observed from the Figure that the patch size of 6×6 is the best choice in order to have good PSNR quality. For this patch size, it can be seen that by increasing the dictionary redundancy more than 4 the average PSNR does not change significantly and improvement getting slow after redundancy 4. Therefore, considering 6×6 patch size and dictionary redundancy of 4 is a good compromise in terms of performance and computational complexity.

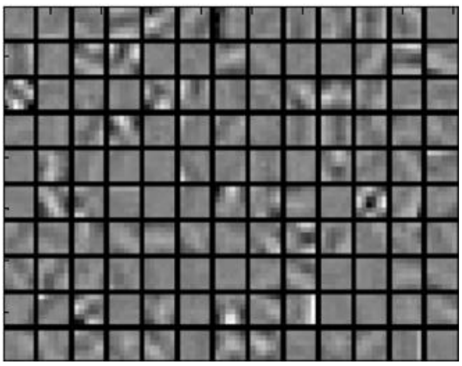
4.3 Learned Dictionaries

The proposed dictionary learning phase used three approaches to classify training data. Over which directional dictionaries are learned. These are: dummy dictionaries, Euclidean distance and correlation based classification.

4.3.1 Designed Directional Dictionaries Based on Classification via Dummy Dictionaries

To remind, we designed eight dummy dictionaries which are structurally directional, then training set is classified to eight structured sets and one non-directional set from the patches which are not directional. The criteria for defining a patch as one of those eight directions was OMP selection based on the least error in the reconstruction. The designed HR dictionaries are shown in Figure 4.2. They are ordered by: 0°(horizontal), 90°(vertical), $45^\circ, 135^\circ, 22.5^\circ, 67.5^\circ, 112.5^\circ, 157.5^\circ$ and non-directional one. All learned dictionaries are of size 130.





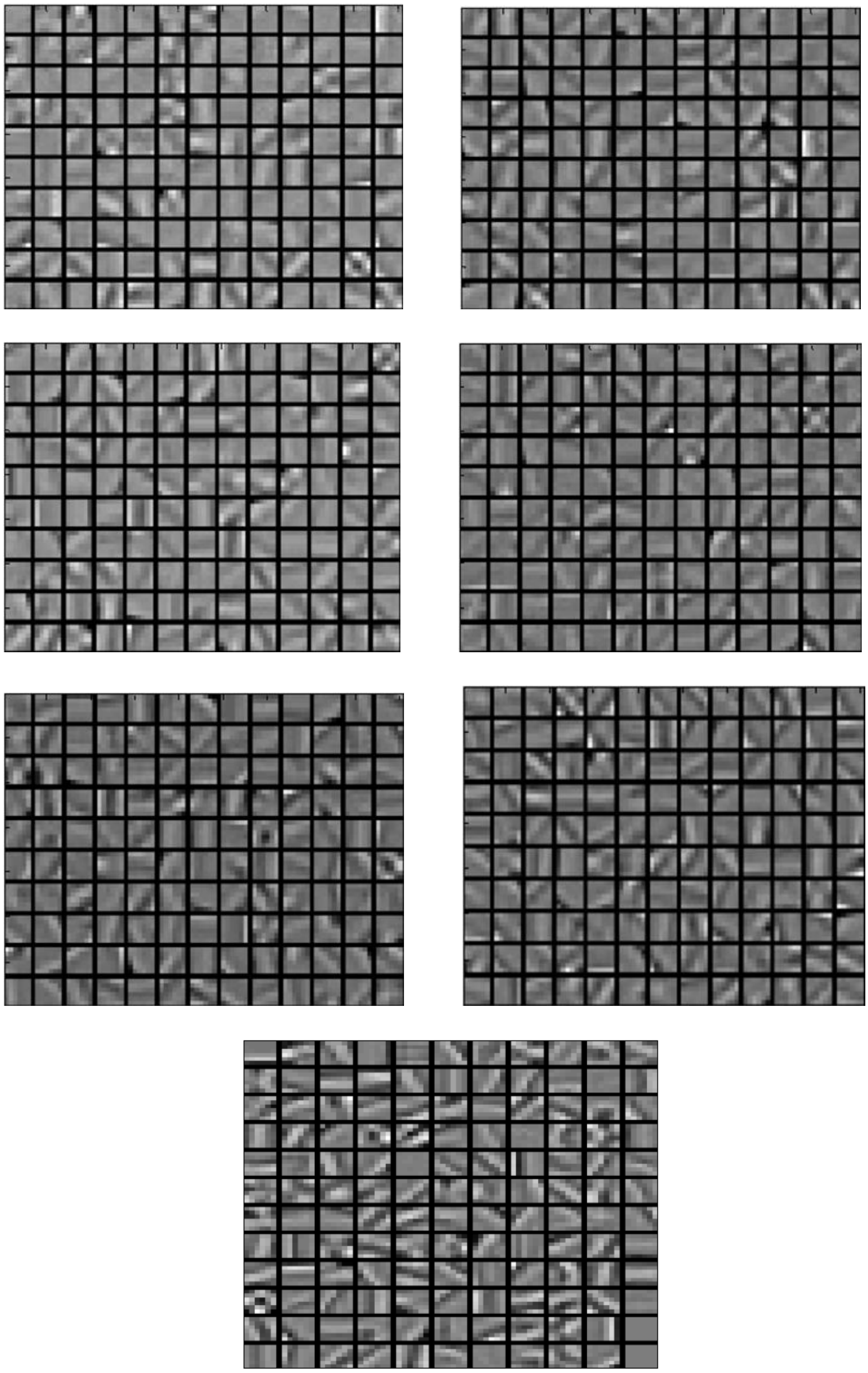
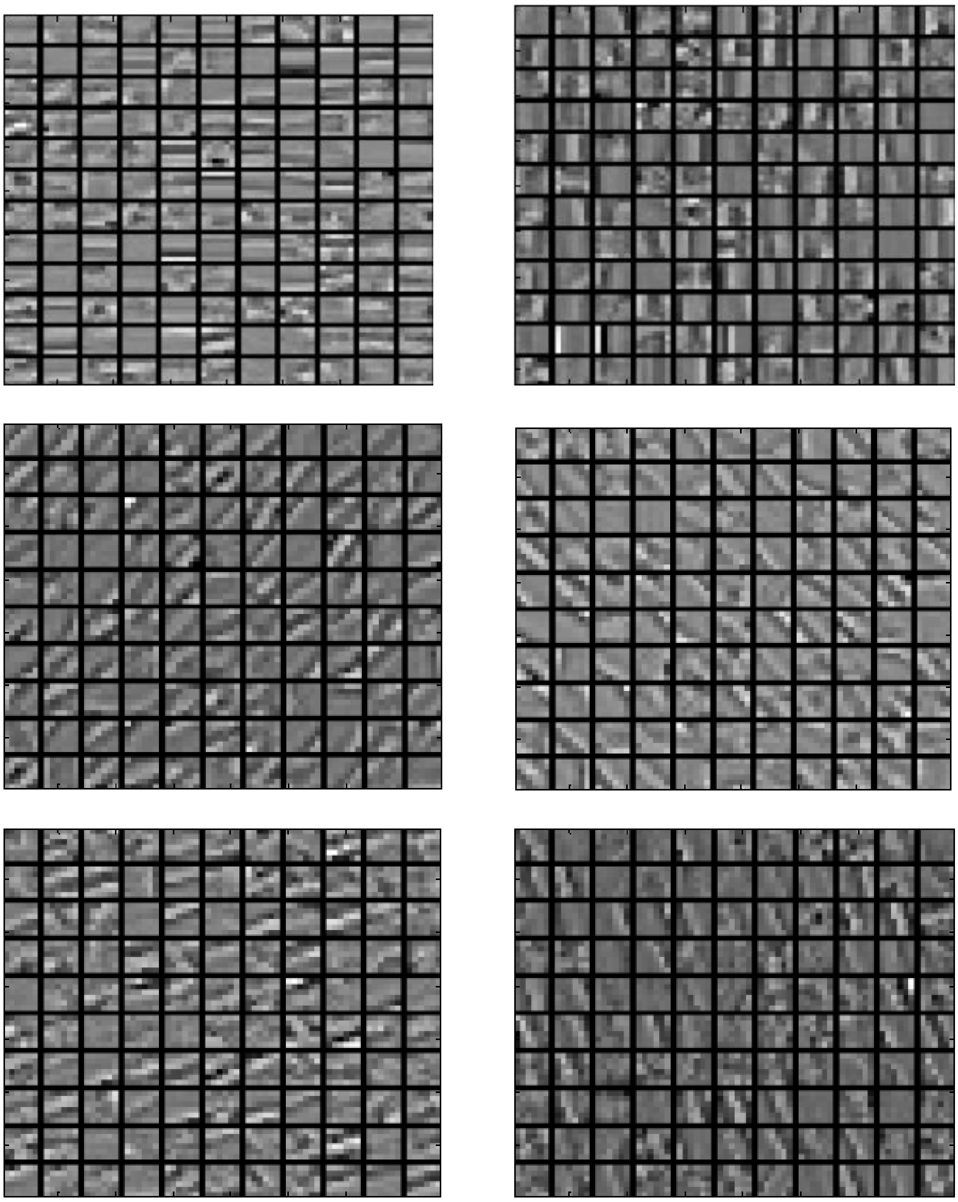


Figure 4.2. Designed HR dictionaries using classification via dummy dictionaries from top left: 0°(horizontal), 90°(vertical), $45^{\circ}, 135^{\circ}, 22.5^{\circ}, 67.5^{\circ}, 112.5^{\circ}, 157.5^{\circ}$ and non-directional.

According to Figure 4.2, some directional atoms can be seen in the dictionaries but none of dictionaries have the specific direction.

4.3.2 Designed Directional Dictionaries Based on Classification via Euclidean Distance

Based on this approach we designed structured dictionaries using structured training sets which are obtained by the gathering all patches which have the least error with the templates. The dictionaries which are designed using this approach are demonstrated in the Figure (4.3). Learned dictionaries all have 130 atoms.



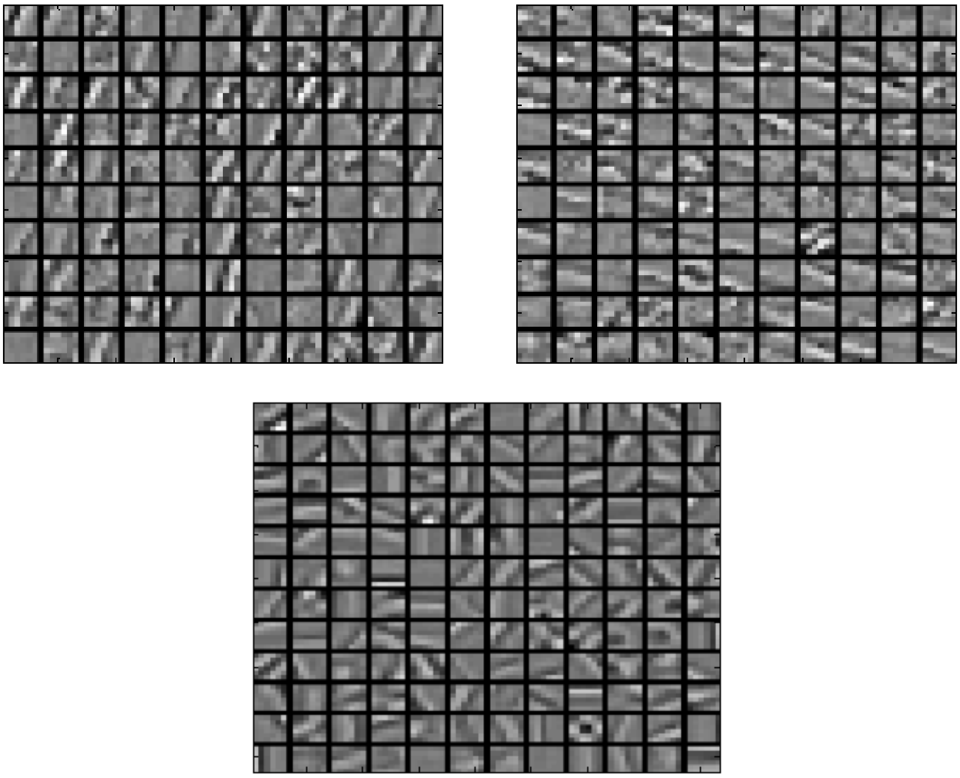


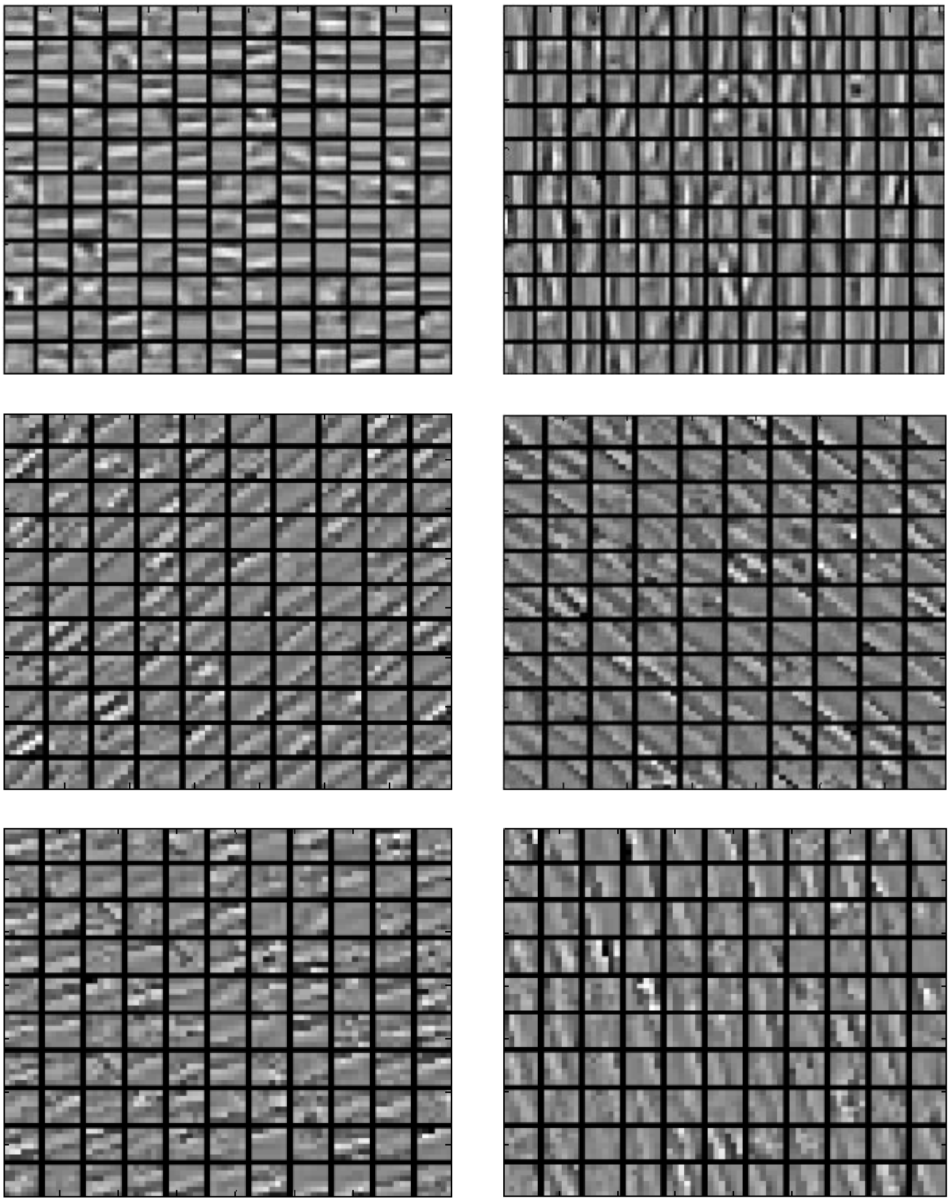
Figure 4.3. Designed HR dictionaries using classification via Euclidean distance, from top left: 0°(horizontal), 90°(vertical), $45^\circ, 135^\circ, 22.5^\circ, 67.5^\circ, 112.5^\circ, 157.5^\circ$and non-directional.

The learned dictionaries contain obviously more directional atoms rather than the previous approach. By looking at the dictionaries, the direction of them can be observed. Horizontal and vertical dictionaries are the richest one where many of their atoms have correct directions.

4.3.3 Designed Directional Dictionaries Based on Classification via Correlation

The third and the last approach is designing dictionaries using correlation to classify training data. According to this approach, every single patch is correlated with the templates and the largest similarity between the patch in question and any of those templates (the template that has the biggest value in correlation with the patch) defines the correct direction of that patch. Figure 4.4 illustrates the designed HR dictionaries using correlation method. All dictionaries are at size 130.

Based on designed dictionaries, it can be observed that classification using correlation is the most successful approach compared to the last two approaches. Dictionaries contain more directional atoms specially for horizontal and vertical dictionaries which almost all their atoms have correct direction.



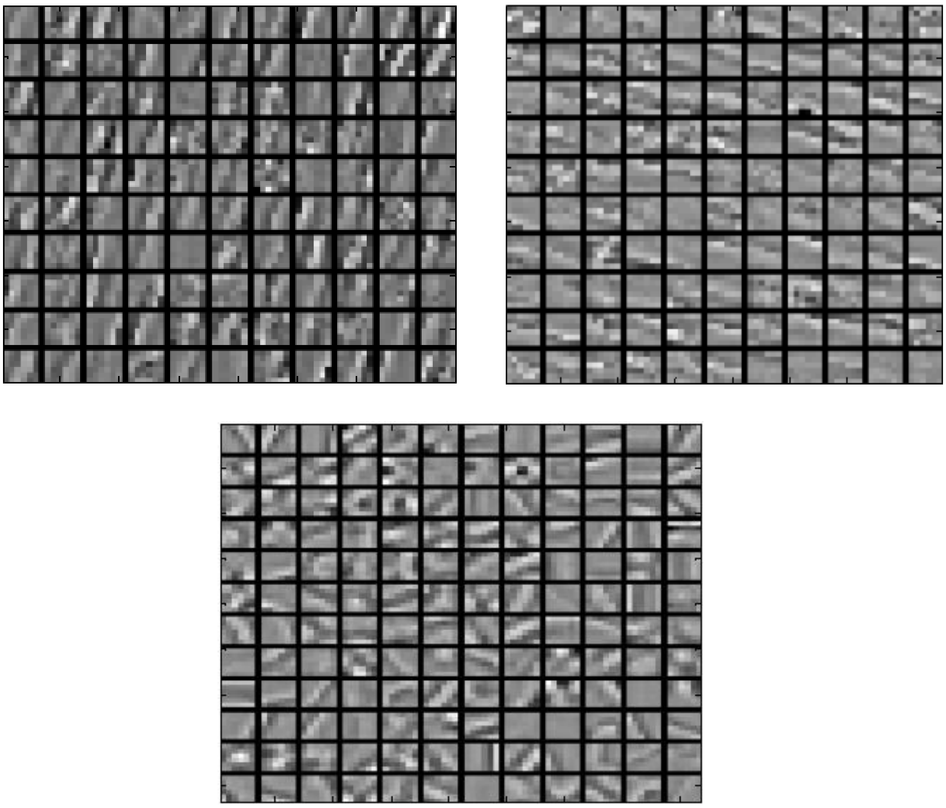


Figure 4.4. Designed HR dictionaries using classification via correlation; from top left: 0°(horizontal), 90°(vertical), $45^{\circ}, 135^{\circ}, 22.5^{\circ}, 67.5^{\circ}, 112.5^{\circ}, 157.5^{\circ}$ and non-directional.

4.3.4 Performance Test of Designed Directional Dictionaries with Correct Model Selection

According to the Figures of designed HR dictionaries especially for the last method, it can be observed that the purpose of designing structurally directional dictionaries is partially achieved. In order to demonstrate the efficiency of training such dictionaries, an experimental study is done. Based on this study, using all structured dictionaries and non-directional one a set of low resolution input image is reconstructed. The dictionary selection model in the reconstruction part is error based where the HR image patches of corresponding LR images are assumed to be known. Thus the LR features are reconstructed using the most appropriate chosen pair of dictionaries based on the minimum error of the representation the corresponding HR patches with HR dictionaries.

4.3.4.1 Quantitative Result

This experiment indicates that it is indeed possible to improve the performance of SISR with directionally structured dictionaries provided that the correct model is selected. The obtained PSNR results correct model selection, illustrate improvement over the state-of-the-art results proposed by R.Zeyde et.al [23].

Table 4.1 shows the PSNR results using Bicubic interpolation, state-of-the-art results [23], and result of our study. It is evident from Table that the test performance using designed structurally directional dictionaries shows better results in terms of PSNR with much better improvement of 1.48 dB on average over Bicubic interpolation and an improvement of 0.2 dB over the-state-of-the-art result.

From the result in this Table, it can be observed that for the images with directional nature, the PSNR improvements are noticeable. For example for the Barbara image which is a highly directional image, the PSNR is improved about 0.5 dB over the-state-of-the-art results while it is 1.18 dB over Bicubic interpolation. Also for the zone-plate image which is a curvy directional image, the PSNR result using the structurally directional dictionaries, shows about 1 dB improvement over the state-of-the-art result and also it gives much better result over Bicubic interpolation with about 1.54 dB improvements. The image set in the result Tables are shown from top left.

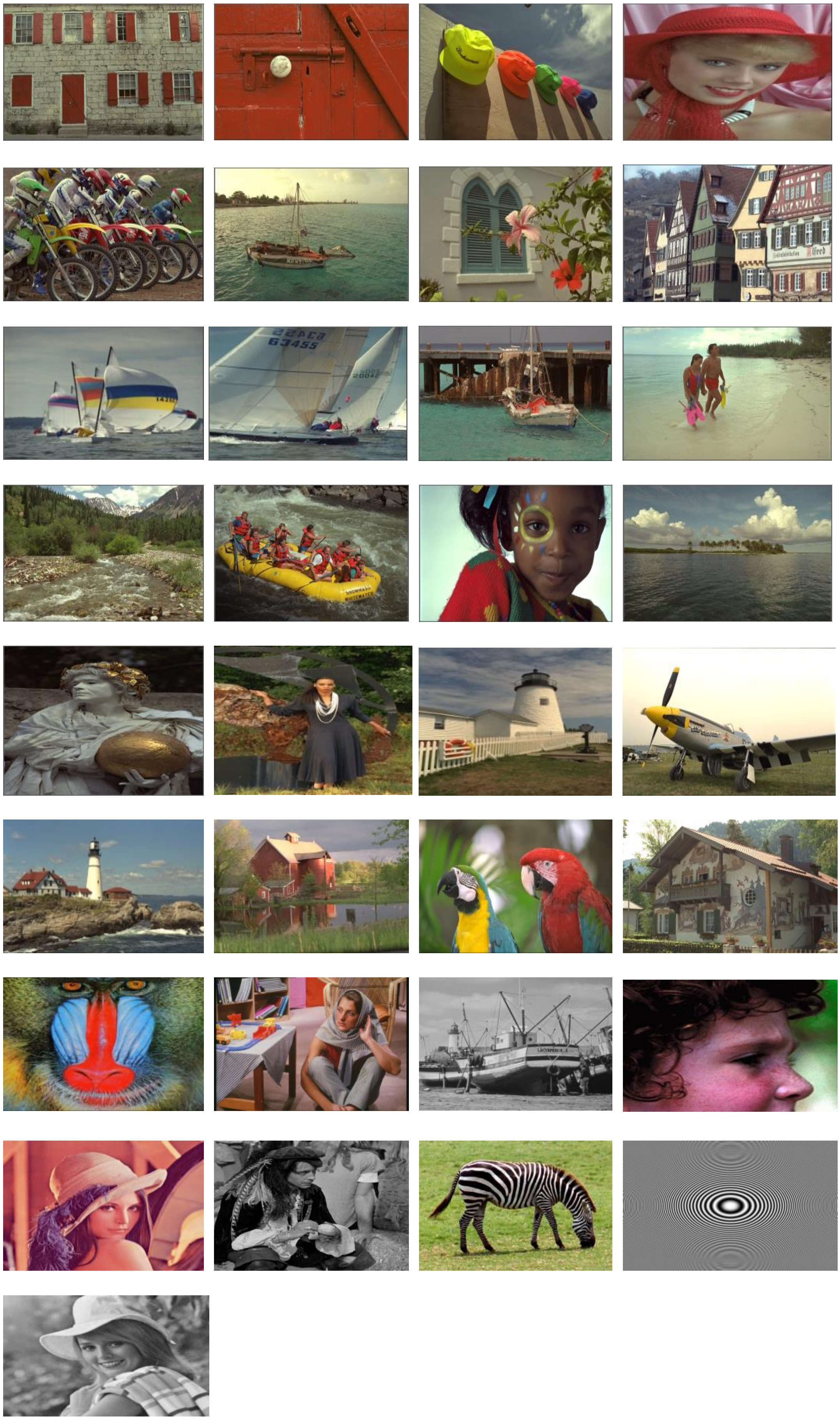


Table 4.1. PSNR results, corresponding to Bicubic, R. Zeyde and Proposed method

Name	Bicubic		R. Zeyde Method		Proposed Method	
	PSNR (dB)	SSIM	PSNR (dB)	SSIM	PSNR (dB)	SSIM
K.1	26.7	0.9597	27.85	0.9806	28.15	0.9783
K.2	34.0	0.9697	35.04	0.9763	35.14	0.9741
K.3	35.0	0.9842	36.66	0.9855	36.66	0.9843
K.4	34.6	0.9813	36.03	0.9847	36.03	0.9833
K.5	27.1	0.9775	28.95	0.9852	29.21	0.9842
K.6	28.3	0.9628	29.42	0.9809	29.66	0.9785
K.7	34.3	0.9923	36.33	0.9905	36.36	0.9900
K.8	24.3	0.9668	25.50	0.9829	25.94	0.9812
K.9	33.1	0.9856	35.04	0.9873	35.16	0.9860
K.10	32.9	0.9848	34.75	0.9874	34.79	0.9862
K.11	29.9	0.9717	31.14	0.9830	31.39	0.9813
K.12	33.6	0.9773	35.58	0.9872	35.58	0.9856
K.13	24.7	0.9576	25.54	0.9780	25.80	0.9758
K.14	29.9	0.9773	31.30	0.9810	31.47	0.9797
K.15	32.9	0.9824	34.90	0.9868	34.90	0.9858
K.16	32.1	0.9705	32.84	0.9770	33.10	0.9752
K.17	32.9	0.9880	34.38	0.9827	34.47	0.9818
K.18	28.8	0.9755	29.89	0.9841	30.07	0.9827
K.19	28.8	0.9719	30.04	0.9782	30.43	0.9766
K.20	32.4	0.9865	34.11	0.9864	34.18	0.9850
K.21	29.3	0.9816	30.36	0.9841	30.65	0.9831
K.22	31.4	0.9753	32.59	0.9821	32.72	0.9806
K.23	35.9	0.9924	37.90	0.9877	37.97	0.9873
K.24	27.6	0.9764	28.62	0.9818	28.82	0.9804
Baboon	24.9	0.9651	25.46	0.9756	25.77	0.9739
Barbara	28.0	0.9577	28.66	0.9813	29.14	0.9792
boat	34.1	0.9863	33.78	0.9817	33.97	0.9807
Face	34.8	0.8463	35.56	0.8485	35.64	0.8489
Lena	34.7	0.9893	36.23	0.9820	36.29	0.9813
Man	29.2	0.9820	30.51	0.9822	30.69	0.9813
Zebra	30.6	0.9877	33.21	0.9806	33.35	0.9800
Z-plate	12.7	0.7054	13.27	0.8680	13.97	0.8654
Elaine	31.1	0.9767	31.31	0.9718	31.32	0.9703
Average	**30.29**	**31.8456**	**31.59**	**32.1731**	**31.77**	**32.128**

The same improvement can be seen for the directional images in Kodak set as well. For the Kodak image number one which contains a lot of directional edges, the improvement is 0.33 dB and 1.48 dB over the state-of-the-art result and bicubic respectively. For the Kodak image number 8 and 24 which are both directional images, the improvement is 0.46 dB, 0.2 dB over the state-of-the-art results.

It is evident that, using structurally directional dictionaries to super-resolve LR images especially directional ones, provides superior results compared to employing only one global dictionary for all kind of images.

4.3.4.2 Qualitative Result

Figures 5 and 6 present visual comparisons of different reconstruction methods for zone-plate and Barbara respectively. Figures show insets of selected zoomed scenes to clarify the comparison. The visual comparison is between the Bicubic interpolation, the-state-of-the-art [23] and the proposed method with correct model selection respectively. The improvements over the-state-of-the-art can be seen visually as well.

Figure 4.5 shows original image and reconstructions from Bicubic, R. Zeyde and proposed method with correct model selection of Zone-plate image. It can be observed that the reconstructed Zone-plate image using proposed method with correct model selection contains more information compared to Bicubic and the-state-of-the-art result. While the proposed method with correct model selection solves the blurring problem in the Bicubic method, it recovered the edges better than the-state-of-the-art approach.

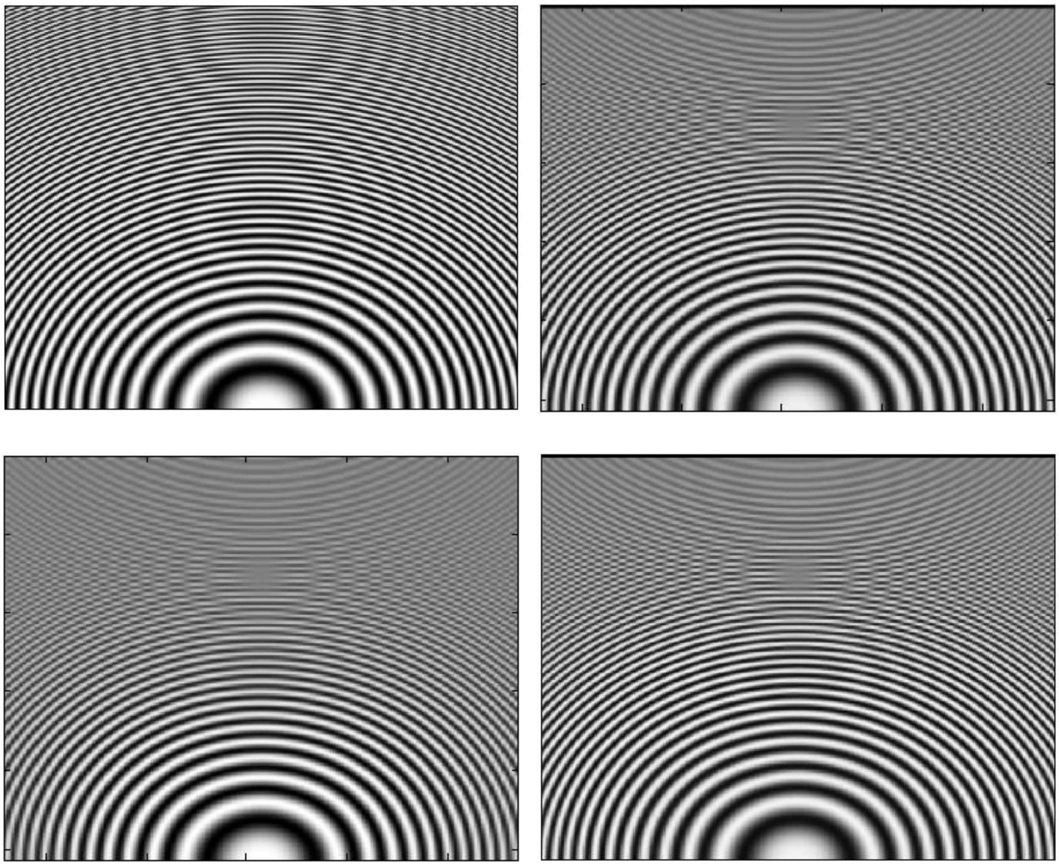


Figure 4.5. Visual comparison for zone-plate, from top left insets of: the original, Bicubic, R. Zeyde and the proposed method with perfect model selection.

The same as zone-plate image, Figure 4.6 illustrates original and the reconstructions from Bicubic, R. Zeyde and the proposed method with correct model selection of Barbara image respectively. Again the result of proposed method enjoys of less blurring rather than Bicubic interpolation while reconstructed directions more accurate than the-state-of-the-art [23].

Our qualitative results are in line with the PSNR results presented in the Table 4.1.

Figure 4.6. Visual comparison for Barbara, from top left insets of: the original, Bicubic, R. Zeyde and the proposed method with perfect model selection.

4.4 Simulation Results of the Reconstruction Phase

According to the proposed method, in order to reconstruct a HR image, two dictionary model selections are proposed: correlation and Euclidean distance based approaches. Using these models together with the three classification schemes, the simulation results are presented.

PSNR results are shown in Tables 4.3, 4.4, 4.5, 4.6, 4.7 and 4.8 for three classification methods and two dictionary model schemes. Results are obtained with the patch size of 6 by 6 and sparsity 3 for all dictionaries and image resolution is increased by factor of 2 in both directions for proposed method and baseline algorithm [23]. Two different settings are defined for the dictionary sizes (number of atoms in the dictionary). One of the configurations is defined according to the experimental study which is discussed earlier. According to this category, dictionary redundancy is 4 while the vectorized patch size is 36 by 1.

Besides the aforementioned setting, we consider another possibility for the number of dictionary atoms. In this category, the dictionary sizes are defined according to how often they are used to recover an image. To decide on the dictionary sizes, simple tests are conducted using the Kodak set. The test results demonstrate that non-directional dictionary and dictionaries corresponding to horizontal and vertical directions are used more than other dictionaries for reconstructing the HR patches respectively. The dictionaries which belong to 45 and -45 degree are at the third place and the rest fall in the bottom of the list.

The configurations of two categories are shown in the Table 4.2. The first category belongs to the dictionaries with the same size based on optimal redundancy. The second category shows the dictionary sizes according to our empirical study.

Table 4.2. Size of Trained Dictionaries

Category	Non-directional	Horizontal and vertical	45 and -45 degree	others
1	130	130	130	130
2	190	150	130	110

4.4.1.1 Simulation Results Using Designed Dictionaries Based on Classification via Dummy Dictionaries

Using designed dictionaries employing classification via dummy dictionaries for reconstruction a HR image, the PSNR performances in Tables 4.3 and 4.4 are obtained. The test images are Kodak set and some benchmark images. Two Tables are defined to show the results of using such dictionaries and reconstruct the LR images via two different dictionary model selections. The PSNR results using correlation and Euclidean distance as the dictionary selection methods are shown in Tables 4.3, 4.4 respectively.

According to the correlation model selection, the LR features are correlated to the all designed templates and the biggest value determines the direction of feature corresponding to that template. Then the corresponding dictionary pair is used to reconstruct the HR patch. Euclidean distance model selection, select the most suitable HR dictionary based on minimum error of the representation of each feature with all LR dictionaries.

Each Table represents PSNR for two categorizes. As it was mentioned before, the category 1 belongs to the reconstruction results using learned dictionaries with the same size and the second category is for the result of designing dictionaries using different sizes. It can be observed from the Tables that the Euclidean distance model selection chooses dictionaries more accurate than the correlation based model selection while it shows on average 1 dB improvement over Bicubic interpolation and 0.26 dB less than the state-of-the-art proposed by R. Zeyde [23] for the second category while results of correlation based model selection illustrates 0.55 dB over Bicubic and 0.75 dB less than R. Zeyde results.

Table 4.3. PSNR results using classification via dummy dictionaries and correlation based model selection, corresponding to Bicubic, R.Zeyde and proposed method.

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.16	27.85	27.22
K.2	34.0	35.04	34.43	35.04	34.46
K.3	35.0	36.66	35.67	36.66	35.67
K.4	34.6	36.03	35.24	36.03	35.24
K.5	27.1	28.95	27.78	28.95	27.79
K.6	28.3	29.42	28.70	29.42	28.71
K.7	34.3	36.33	35.12	36.33	35.16
K.8	24.3	25.50	24.84	25.50	24.86
K.9	33.1	35.04	33.96	35.04	33.98
K.10	32.9	34.75	33.65	34.75	33.66
K.11	29.9	31.14	30.41	31.14	30.42
K.12	33.6	35.58	34.17	35.58	34.17
K.13	24.7	25.54	25.08	25.54	25.07
K.14	29.9	31.30	30.46	31.30	30.47
K.15	32.9	34.90	33.64	34.90	33.65
K.16	32.1	32.84	32.45	32.84	32.45
K.17	32.9	34.38	33.43	34.38	33.44
K.18	28.8	29.89	29.26	29.89	29.26
K.19	28.8	30.04	29.50	30.04	29.51
K.20	32.4	34.11	33.03	34.11	33.04
K.21	29.3	30.36	29.77	30.36	29.76
K.22	31.4	32.59	31.90	32.59	31.90
K.23	35.9	37.90	36.80	37.90	36.81
K.24	27.6	28.62	28.07	28.62	28.07
Baboon	24.9	25.46	25.16	25.46	25.16
Barbara	28.0	28.66	28.34	28.66	28.34
boat	34.1	33.78	33.06	33.78	33.08
Face	34.8	35.56	35.19	35.56	35.20
Lena	34.7	36.23	35.40	36.23	35.41
Man	29.2	30.51	29.74	30.51	29.74
Zebra	30.6	33.21	31.74	33.21	31.73
Z-plate	12.7	13.27	13.11	13.27	13.11
Elaine	31.1	31.31	31.18	31.31	31.18
Average	**30.29**	**31.59**	**30.83**	**31.59**	**30.84**

Table 4.4. PSNR results using classification via dummy dictionaries and Euclidean dictance based model selection, corresponding to Bicubic, R. Zeyde and proposed method

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.52	27.85	27.62
K.2	34.0	35.04	34.80	35.04	34.84
K.3	35.0	36.66	36.18	36.66	36.28
K.4	34.6	36.03	35.67	36.03	35.78
K.5	27.1	28.95	28.42	28.95	28.54
K.6	28.3	29.42	29.10	29.42	29.17
K.7	34.3	36.33	35.82	36.33	35.96
K.8	24.3	25.50	25.18	25.50	25.24
K.9	33.1	35.04	34.50	35.04	34.69
K.10	32.9	34.75	34.20	34.75	34.36
K.11	29.9	31.14	30.80	31.14	30.90
K.12	33.6	35.58	34.72	35.58	34.91
K.13	24.7	25.54	25.34	25.54	25.38
K.14	29.9	31.30	30.90	31.30	30.99
K.15	32.9	34.90	34.54	34.90	34.73
K.16	32.1	32.84	32.68	32.84	32.72
K.17	32.9	34.38	33.93	34.38	34.09
K.18	28.8	29.89	29.60	29.89	29.66
K.19	28.8	30.04	29.79	30.04	29.81
K.20	32.4	34.11	33.53	34.11	33.67
K.21	29.3	30.36	30.11	30.36	30.17
K.22	31.4	32.59	32.25	32.59	32.32
K.23	35.9	37.90	37.43	37.90	37.54
K.24	27.6	28.62	28.37	28.62	28.42
Baboon	24.9	25.46	25.36	25.46	25.38
Barbara	28.0	28.66	28.53	28.66	28.57
boat	34.1	33.78	33.47	33.78	33.54
Face	34.8	35.56	35.43	35.56	35.47
Lena	34.7	36.23	35.87	36.23	35.97
Man	29.2	30.51	30.12	30.51	30.19
Zebra	30.6	33.21	32.49	33.21	32.64
Z-plate	12.7	13.27	13.23	13.27	13.**23**
Elaine	31.1	31.31	31.27	31.31	31.28
Average	**30.29**	**31.59**	**31.24**	**31.59**	**31.33**

The results of second category are in line with the first category with 0.95 dB and 0.54 dB improvements over Bicubic interpolation and 0.35 dB and 0.76 dB below R. Zeyde results for Euclidean distance and correlation based model selection respectively.

4.4.1.2 Simulation Results Using Designed Dictionaries Based on Classification via Euclidean Distance

Based on this approach we are designing structured dictionaries using structurally directional training sets which are obtained by the gathering all patches which have the least error with the templates. Such designed dictionaries were shown in Figure 4.3.

Tables 4.5, 4.6 illustrate the corresponding PSNR results of the super resolution of test images in two different categories with the same configuration as the previous approach. The same as previous method, results show the superior of using Euclidean distance rather than the correlation method in sense of dictionary selection. Although the result of category 2 for both Tables does not have big difference from category1 but it shows a slightly improvement.

4.4.1.3 Simulation Results Using Designed Dictionaries Based on Classification via Correlation

The PSNR performances listed in Tables 4.7 and 4.8 are obtained by running the proposed algorithm in the two mentioned categories. The same way of the previous approaches the test images are reconstructed using both dictionary selection models. The learned dictionaries have visually slightly sharper directions rather than the designed dictionaries using the Euclidean distance which lead to a slightly improvement in the PSNR result.

Table 4.5. PSNR results using classification via Euclidean distance, and correlation based model selection, corresponding to Bicubic, R. Zeyde and proposed method.

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.79	27.85	27.79
K.2	34.0	35.04	34.89	35.04	34.89
K.3	35.0	36.66	36.36	36.66	36.31
K.4	34.6	36.03	35.77	36.03	35.82
K.5	27.1	28.95	28.59	28.95	28.59
K.6	28.3	29.42	29.27	29.42	29.17
K.7	34.3	36.33	35.83	36.33	35.95
K.8	24.3	25.50	25.38	25.50	25.33
K.9	33.1	35.04	34.78	35.04	34.83
K.10	32.9	34.75	34.36	34.75	34.36
K.11	29.9	31.14	30.97	31.14	30.97
K.12	33.6	35.58	35.17	35.58	34.97
K.13	24.7	25.54	25.42	25.54	25.42
K.14	29.9	31.30	31.08	31.30	31.07
K.15	32.9	34.90	34.60	34.90	34.62
K.16	32.1	32.84	32.76	32.84	32.77
K.17	32.9	34.38	34.12	34.38	34.13
K.18	28.8	29.89	29.72	29.89	29.71
K.19	28.8	30.04	29.90	30.04	29.92
K.20	32.4	34.11	33.68	34.11	33.71
K.21	29.3	30.36	30.22	30.36	30.22
K.22	31.4	32.59	32.43	32.59	32.38
K.23	35.9	37.90	37.58	37.90	37.57
K.24	27.6	28.62	28.46	28.62	28.46
Baboon	24.9	25.46	25.38	25.46	25.39
Barbara	28.0	28.66	28.57	28.66	28.59
boat	34.1	33.78	33.58	33.78	33.67
Face	34.8	35.56	35.47	35.56	35.49
Lena	34.7	36.23	36.01	36.23	36.06
Man	29.2	30.51	30.28	30.51	30.28
Zebra	30.6	33.21	32.92	33.21	32.93
Z-plate	12.7	13.27	13.21	13.27	13.24
Elaine	31.1	31.31	31.28	31.31	31.29
Average	**30.29**	**31.59**	**31.38**	**31.59**	**31.41**

Table 4.6. PSNR results using classification via Euclidean distance, and Euclidean dictance based model selection, corresponding to Bicubic, R. Zeyde and proposed method.

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.81	27.85	27.81
K.2	34.0	35.04	34.99	35.04	35.00
K.3	35.0	36.66	36.61	36.66	36.63
K.4	34.6	36.03	35.96	36.03	35.96
K.5	27.1	28.95	28.91	28.95	28.91
K.6	28.3	29.42	29.39	29.42	29.39
K.7	34.3	36.33	36.19	36.33	36.21
K.8	24.3	25.50	25.46	25.50	25.48
K.9	33.1	35.04	34.97	35.04	34.95
K.10	32.9	34.75	34.71	34.75	34.73
K.11	29.9	31.14	31.09	31.14	31.10
K.12	33.6	35.58	35.52	35.58	35.47
K.13	24.7	25.54	25.49	25.54	25.49
K.14	29.9	31.30	31.25	31.30	31.25
K.15	32.9	34.90	34.88	34.90	34.96
K.16	32.1	32.84	32.81	32.84	32.80
K.17	32.9	34.38	34.32	34.38	34.29
K.18	28.8	29.89	29.83	29.89	29.84
K.19	28.8	30.04	30.02	30.04	29.97
K.20	32.4	34.11	33.98	34.11	34.01
K.21	29.3	30.36	30.30	30.36	30.29
K.22	31.4	32.59	32.52	32.59	32.53
K.23	35.9	37.90	37.86	37.90	37.88
K.24	27.6	28.62	28.58	28.62	28.57
Baboon	24.9	25.46	25.45	25.46	25.45
Barbara	28.0	28.66	28.58	28.66	28.57
boat	34.1	33.78	33.74	33.78	33.75
Face	34.8	35.56	35.57	35.56	35.55
Lena	34.7	36.23	36.22	36.23	36.22
Man	29.2	30.51	30.43	30.51	30.45
Zebra	30.6	33.21	33.13	33.21	33.13
Z-plate	12.7	13.27	13.21	13.27	13.20
Elaine	31.1	31.31	31.31	31.31	31.31
Average	**30.29**	**31.59**	**31.54**	**31.59**	**31.55**

Table 4.7. PSNR results using classification via correlation, and correlation based model selection, corresponding to Bicubic, R. Zeyde and proposed method.

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.77	27.85	27.81
K.2	34.0	35.04	34.91	35.04	34.91
K.3	35.0	36.66	36.37	36.66	36.38
K.4	34.6	36.03	35.90	36.03	35.81
K.5	27.1	28.95	28.61	28.95	28.63
K.6	28.3	29.42	29.26	29.42	29.26
K.7	34.3	36.33	35.90	36.33	35.90
K.8	24.3	25.50	25.37	25.50	25.37
K.9	33.1	35.04	34.84	35.04	34.83
K.10	32.9	34.75	34.36	34.75	34.40
K.11	29.9	31.14	30.98	31.14	31.01
K.12	33.6	35.58	35.24	35.58	35.27
K.13	24.7	25.54	25.42	25.54	25.43
K.14	29.9	31.30	31.07	31.30	31.10
K.15	32.9	34.90	34.68	34.90	34.65
K.16	32.1	32.84	32.75	32.84	32.77
K.17	32.9	34.38	34.12	34.38	34.17
K.18	28.8	29.89	29.72	29.89	29.73
K.19	28.8	30.04	29.98	30.04	29.90
K.20	32.4	34.11	33.71	34.11	33.75
K.21	29.3	30.36	30.22	30.36	30.23
K.22	31.4	32.59	32.42	32.59	32.43
K.23	35.9	37.90	37.58	37.90	37.57
K.24	27.6	28.62	28.45	28.62	28.47
Baboon	24.9	25.46	25.37	25.46	25.38
Barbara	28.0	28.66	28.58	28.66	28.58
boat	34.1	33.78	33.56	33.78	33.57
Face	34.8	35.56	35.46	35.56	35.47
Lena	34.7	36.23	36.01	36.23	36.03
Man	29.2	30.51	30.30	30.51	30.31
Zebra	30.6	33.21	32.91	33.21	32.95
Z-plate	12.7	13.27	13.22	13.27	13.22
Elaine	31.1	31.31	31.27	31.31	31.28
Average	**30.29**	**31.59**	**31.40**	**31.59**	**31.42**

Table 4.8. PSNR results using classification via correlation, and Euclidean distance based model selection, corresponding to Bicubic, R. Zeyde and proposed method.

Name	Bicubic PSNR (dB)	Category1		Category2	
		R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)	R.Zeyde PSNR (dB)	Proposed Method PSNR (dB)
K.1	26.7	27.85	27.86	27.85	27.93
K.2	34.0	35.04	35.03	35.04	35.02
K.3	35.0	36.66	36.61	36.66	36.58
K.4	34.6	36.03	35.93	36.03	35.99
K.5	27.1	28.95	28.95	28.95	28.95
K.6	28.3	29.42	29.28	29.42	29.35
K.7	34.3	36.33	36.23	36.33	36.22
K.8	24.3	25.50	25.45	25.50	25.46
K.9	33.1	35.04	35.03	35.04	35.03
K.10	32.9	34.75	34.75	34.75	34.75
K.11	29.9	31.14	31.11	31.14	31.13
K.12	33.6	35.58	35.30	35.58	35.38
K.13	24.7	25.54	25.49	25.54	25.50
K.14	29.9	31.30	31.24	31.30	31.25
K.15	32.9	34.90	34.93	34.90	34.72
K.16	32.1	32.84	32.82	32.84	32.82
K.17	32.9	34.38	34.33	34.38	34.38
K.18	28.8	29.89	29.85	29.89	29.85
K.19	28.8	30.04	30.07	30.04	30.01
K.20	32.4	34.11	34.03	34.11	34.04
K.21	29.3	30.36	30.29	30.36	30.32
K.22	31.4	32.59	32.52	32.59	32.56
K.23	35.9	37.90	37.95	37.90	37.98
K.24	27.6	28.62	28.57	28.62	28.57
Baboon	24.9	25.46	25.46	25.46	25.46
Barbara	28.0	28.66	28.55	28.66	28.55
boat	34.1	33.78	33.71	33.78	33.75
Face	34.8	35.56	35.54	35.56	35.54
Lena	34.7	36.23	36.24	36.23	36.23
Man	29.2	30.51	30.46	30.51	30.48
Zebra	30.6	33.21	33.10	33.21	33.12
Z-plate	12.7	13.27	13.20	13.27	13.20
Elaine	31.1	31.31	31.30	31.31	31.31
Average	**30.29**	**31.59**	**31.54**	**31.59**	**31.56**

According to the last two approaches, the obtained results are much better than Bicubic method by 1.12 dB And 1.13 dB improvement respectively when dictionaries are designed using Euclidean distance and correlation classification approaches and both are reconstructed using correlation based dictionary selection method. Also by 1.26 dB and 1.27 dB improvement over Bicubic interpolation when the Euclidean distance approach is used as the dictionary selection model.

Tables illustrate that selecting dictionaries by employing Euclidean distance approach as the dictionary selection model, gives us better result in comparison with the correlation model selection. Thus according to Tables 4.6, 4.8 our method results are comparable to the-state-of-the-art proposed by R. Zeyde with just a very small difference less, 0.04 dB and 0.03 dB for the classification using Euclidean distance and correlation respectively.

The PSNR and SSIM values listed in the Tables were obtained by running the proposed algorithm with sparsity 3 and 6×6 patch size such that the trained dictionaries have a dimension of 36×130. With this configuration on an Intel Pentium dual Core, 2200 MHz laptop PC under Matlab 2013a, the execute time in order to learn all nine dictionaries together are measured to be 170s, 280s and 140s for the first, second and the third approach respectively for 20 K-SVD iterations. The time required to train a single dictionary of size 36×1170 and the same number of iterations and sparsity for the method proposed by R. Zeyde is about 830s. For both method Bicubic interpolation of factor 2 were used.

Chapter 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this thesis we have proposed an algorithm for single image super resolution based on sparse representation, in terms of structurally directional dictionaries. The proposed algorithm is based on dictionary learning in the spatial domain. Structured dictionaries in eight directions and one non-directional one are trained employing KSVD algorithm.

Designing structurally directional dictionaries is template matching based where templates are designed to model eight directions which all together cover 2-D space. Training data is classified in nine clusters, eight directional and one non-directional. Classification is done using a similarity measurement together with templates and then corresponding HR and LR dictionaries are learned.

In the reconstruction part, LR input image is reconstructed using nine HR designed dictionaries together with the sparse coefficients obtained using LR dictionaries. Dictionary selection model is error based while the criteria to choose the most appropriate HR dictionary is LR feature.

The effect of dictionary redundancy is empirically studied and it is found that a patch of size 6×6 for the HR patches (3×3 for LR patches) and dictionary size of 130 for all the dictionaries is a good compromise between representation quality and computational complexity. After training structured dictionaries, an experimental study is done to show the effectiveness of designing such dictionaries compared to a single global one. Results indicate the effectiveness of directional dictionaries compared to Bicubic, the proposed method with correct model selection gives 1.5 dB for Kodak set and some benchmark images; and also 0.2 dB improvements over the-state-of-art result proposed by R. Zeyde et.al.

Although structurally directional dictionaries are shown via simulation to have superior performance over state-of-the-art, it should be mentioned choosing the best HR dictionary based on LR patches (features) is not a trivial task. Using two proposed dictionary model selections which choose the HR dictionaries based on LR features, it is observed that the correct HR dictionary is not always chosen. Thus the results are not the same as it is expected from the test performance. Results illustrate 1.3 dB improvements over Bicubic interpolation while they are almost the same as the-state-of-the-art result presented by R. Zeyde with 0.03 dB lower PSNR on average.

5.2 Future Work

According to the results which were discussed in the previous chapter, we found out that designing structurally directional dictionaries is an effective approach in order to improve the enhancement of an image. For this purpose, we need a strong dictionary selection model when the only criterion is LR input image patches.

Thus as the future work:

- Propose a powerful model selection to select the most suitable dictionary where using LR patches yield in the best HR dictionary selection.
- In order to have stronger result not only for directional patches but also for all different patch structures, design more complicated templates and learn corresponding dictionaries.

REFERENCES

[1] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions On Image Processing*, vol. 15, pp. 3736-3745, 2006.

[2] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3397-3415, 1993.

[3] Y. C. Pati, R. Rezaiifar and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet," in *Conference on Signals, Systems and computers Asilomar,* Pacific Grove, CA, 1993.

[4] L. Rebollo-Neira and D. Lowe, "Optimized orthogonal matching pursuit approach," *Signal Processing Letters, IEEE Journal*, vol. 9, pp. 137-140, 2002.

[5] B. A. Olshausen and B. J. Field, "Natural image statistics and efficient coding," *Computation in Neural Systems*, vol. 7, pp. 333-339, 1996.

[6] K. Engan, S. O. Aase and J. H. Hakon-Husoy, "Method of optimal directions for frame design," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, Phoenix, AZ, 1999, pp. 2443-2446.

[7] O. G. Sezer, O. Harmanci and O. G. Guleryuz, "Sparse orthonormal transforms

for image compression," in *IEEE International Conference on Image Processing*, San Diego, CA, 2008, pp. 149 - 152.

[8] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, pp. 10-60, 2010.

[9] M. Aharon and M. Elad, "Sparse and redundant modeling of image content using an image-signature-dictionary," *SIAM Journal*, vol. 1, pp. 228-247, 2008.

[10] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online dictionary learning for sparse coding," in *International Conference on Machine Learning(ICML)*, New York, USA, 2009.

[11] J. Zepeda and F. Labeau, "Tandem filter bank-DFT code for bursty erasure correction," in *Vehicular Technology Conference, IEEE,* Montreal, Quebec, 2006.

[12] G. Rath and C. Guillemot, "Subspace-based error and erasure correction with DFT codes for wireless channels," *IEEE Transactions on Signal Processing*, vol. 52, pp. 3241 - 3252, 2004.

[13] M. Elad, J. L. Starck, P. Querre and D. L. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA),"

Applied and Computational Harmonic Analysis, vol. 19, pp. 340-358, 2005.

[14] O. G.Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising-part I: theory," *IEEE Transactions on Image Processing*, vol. 15, pp. 539-554, 2006.

[15] M. Elad and M. Aharon, "Image denoising via learned dictionaries and Sparse Representation," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, New York, NY, USA, 2006, pp. 895 - 900.

[16] J. L. Starck, M. Elad and D. L. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing*, vol. 14, pp. 1570-1582, 2004.

[17] J. L. Starck, M. Elad and D. L. Donoho, "Redundant multiscale transforms and their application for morphological component analysis," *Advances in Imaging and Electron Physics*, vol. 132, pp. 287–348, 2004.

[18] Peyre, Gabriel, Fadili, Jalal, Starck and Jean Luc, "Learning the morphological diversity," *SIAM Journal on Image Sciences*, vol. 7, pp. 646–669, 2010.

[19] O. Bryt and M. Elad, "Compression of facial images using the K-SVD algorithm," *Journal of Visual Communication and Image Representation*, vol. 19, pp. 270-282, 2008.

[20] 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. 4311-4322, 2006.

[21] M. D. Adams, "The JPEG-2000 still image compression standard," *Signal Processing Magazine, IEEE*, vol. 18, no. 5, pp. 36 - 58, 2005.

[22] J. Yang, J. Wright, T. Huang and Yi. Ma, "Image super-resolution as sparse representation of raw image patches," in *IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, 2008, pp. 1 - 8.

[23] R. Zeyde, M. Elad and M. Protter, "On single image scale up using sparse representation," in *the 7th International conference on Curves and Surfaces*, vol. 6920, Avignon, France, 2012, pp. 711-730.

[24] J. Yang, J. Wright, T. Huang and Yi Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, pp. 2861-2873, 2010.

[25] M. Elad, I. Yavneh, "A Plurality of Sparse Representation is Better than the Sparsest One Alone," *IEEE Transaction on Information Theory*, vol. 55, pp. 4701-4714, 2009.

[26] L. Zhand, G. Shi and X. Wu W. Dong, "Image debluuring and super-resolution by adaptive sparse domaiin selection and adaptive regularization," *IEEE*

Transactions on Image Processing, vol. 20, pp. 1838-1857, 2011.

[27] W. T. Freeman and E. C. Pasztor, "Learning low-level vision," in *IEEE International Conference on Computer Vision*, vol. 40, Kerkyra, 1999, pp. 1182 - 1189.

[28] M. Irani and S. Peleg, "Improving resolution by image registration," *Graphical Model and Image Processing(CVGIP)* , vol. 53, pp. 231-239, 1991.

[29] A. M. Bruckstein, D. L. Donoho and M. Elad, "From Sparse solutions of systems of equations to sparse modeling of signals and images," *SAIM Review*, vol. 51, pp. 34-81, 2009.