

**BLIND SUPER-RESOLUTION FROM MULTIPLE UNDERSAMPLED IMAGES
USING SAMPLING DIVERSITY**

by

Faisal M. Al-Salem

A dissertation submitted in partial fulfillment
of the requirements of the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2010

Doctoral Committee:

Professor Andrew E. Yagle, Chair
Professor Jeffrey A. Fessler
Professor Mahta Moghaddam
Professor Douglas C. Noll

The ability to ask the right question is more than half the battle of finding the answer.

—Thomas J. Watson.

© Faisal M. Al-Salem 2010
All Rights Reserved

To my family

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my advisor, Prof. Andrew Yagle, for recommending this exciting research topic, for his guidance, and for his patience and support.

I would also like to thank my thesis committee members, Prof. Fessler, Prof. Noll, and Prof. Moghaddam for their valuable time, input and suggestions.

Finally, I am grateful to Becky Turanski, the EE: Systems Graduate Program Coordinator, for her outstanding administrative assistance.

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES	vi
GLOSSARY	viii
ABSTRACT	xi
CHAPTER	
I. Introduction	1
1.1 Overview of Super-resolution Methods	2
1.2 Contribution	4
1.3 Applications	5
1.4 Thesis Outline	6
II. A Novel Approach to Multiframe Super-resolution	7
2.1 Introduction	7
2.2 Low-Resolution Images as Basis Signals.....	10
2.2.1 LSI Transforms	10
2.2.2 LSV Transforms.....	13
2.3 Sampling Diversity	15
2.3.1 An Illustration of the Property of Sampling Diversity	16
2.3.2 The Mapping Functions T_n and T_m	17
2.3.3 The Hardware Requirements	18
III. Solving for the Expansion Coefficients of the Polyphase Components	22
3.1 Introduction	22
3.1.1 The LS Solution	22
3.1.2 Regularized LS Solution	26
3.2 The TLS Solution	28
3.2.1 Tikhonov Regularized TLS	31
3.2.2 L1-Regularized TLS	32

3.3	Mean and Covariance of an Estimated PPC	36
3.4	Pre-denoising the LR Images using PCA.....	39
3.4.1	The Sample Mean and Sample Covariance Matrix.....	41
3.4.2	Outlier LR Images and their Effect on Denoising.....	44
3.5	Color Images	45
3.6	Post-Processing the SR Image.....	47
3.7	Summary	48
3.8	Future Work	49
IV.	Estimation of the Reference Polyphase Component	52
4.1	Introduction	52
4.2	Minimizing the Euclidean Distance in the Pixel Domain	53
4.2.1	Incomplete, Noisy Basis.....	54
4.2.2	Noise Magnification.....	55
4.3	Minimizing the Euclidean Distance in a Decorrelated Subspace	57
4.4	Which Reference PPC to Estimate?	58
4.5	An Intuitive Alternative to Estimating the Reference PPC	62
V.	Applications and Experimental Results	64
5.1	Applications	64
5.1.1	Introduction	64
5.1.2	The Case of Approximately Pure Translations	65
5.1.3	Super-resolution from Vibrations	65
5.1.4	Atmospheric Turbulence	65
5.2	Experimental Results	67
5.2.1	Synthetic Data Experiments.....	71
5.2.2	Real Data Experiments	75
VI.	Summary and Future Work.....	92
6.1	Summary	92
6.2	Future Work	93
BIBLIOGRAPHY	95

LIST OF FIGURES

Figure		
2.1	The observation model.	9
2.2	LR images obtained from a HR image via LSV transformation and downsampling.	13
2.3	In the LSV case, a LR image is a linear combination of separate parts of the PPCs of the HR image (a LR image can be viewed as a linear <i>local</i> mixing of subregions of the PPCs).	14
2.4	A two-CCD sensor camera configuration, using a beam splitter.	20
2.5	An illustration of the property of sampling diversity.	21
3.1	Reconstruction of the PPCs using the LR images as a basis.	24
4.1	For $I = 4$, $J = 5$, $m^* = 13$ and $n = 1, \dots, 16$, the highlighted (white) blocks represent all the pairs (n, m) that give the same pairs of sub data matrices given by (n, m^*)	61
5.1	An illustration of the integration effect of the primary LR CCD array corresponding to $\downarrow 4 \times 4$	69
5.2	An illustration of the integration effect of the secondary LR CCD array corresponding to $\downarrow 5 \times 5$	70
5.3	LSI PSF. (# of LRs = 16).	73
5.4	LSV PSF. (# of LRs = 100).	74
5.5	Approximately pure translations. (# of LRs = 35).	77
5.6	Approximately pure translations. Details: dog's face. (# of LRs = 35).	78
5.7	Approximately pure translations. Details: text. (# of LRs = 35).	79
5.8	Approximately pure translations—video. (# of LRs = 30).	80
5.9	Approximately pure translations—video: Iterative L1. (# of LRs = 30).	81

5.10	Random vibrations: estimating the ref. PPC in the pixel domain. No denoising.	83
5.11	Random vibrations: using a single sec. LR image vs. estimating the ref. PPC. (# LRs = 35).	84
5.12	Random vibrations: Iterative L1 + sharpened (UM). (# LRs = 35).	85
5.13	Rhythmic vibrations. Details part I. (# of LRs = 70).	86
5.14	Rhythmic vibrations. Details part II. (# of LRs = 70).	87
5.15	Rhythmic vibrations. Details part III. (# of LRs = 70).	88
5.16	Atmospheric turbulence. (# of LRs = 100).	90
5.17	Atmospheric turbulence: Blind SR vs. Iterative L1. (# of LRs = 100).	91

GLOSSARY

Symbol	Description
$\underline{\mu}$	mean of the primary LR images.
$\hat{\underline{\mu}}$	<i>sample</i> mean of the <i>sub</i> LR images.
$\underline{\mu}_d^P$	downsized mean of the primary LR images.
$\underline{\mu}_w$	bias of the estimated expansion coefficients of a primary PPC.
σ_v^2	variance of noise in the data matrix.
Φ	differentiating kernel.
C	covariance matrix of the noiseless version of the primary LR images.
C_y	covariance matrix of the primary LR images.
\hat{C}_y	<i>sample</i> covariance matrix of the <i>sub</i> LR images.
$\text{Cov}(\cdot)$	covariance.
d	number of pixels in a primary LR image/primary PPC.
D	reduced PCA matrix: retaining significant eigenvectors of \hat{C}_y .
D_i	$\downarrow I \times I$ downsampling matrix corresponding to the <i>i</i> -th <i>sub</i> PPC.
D_j	$\downarrow J \times J$ downsampling matrix corresponding to the <i>j</i> -th <i>sub</i> PPC.
$\mathbf{E}[\cdot]$	expectation.
E_r	reduced PCA matrix: retaining significant eigenvectors of C_y .
i	index to identify <i>sub</i> PPCs of a secondary PPC.
I	primary downsampling factor.
j	index to identify <i>sub</i> PPCs of a primary PPC.
J	secondary downsampling factor.

Symbol	Description		
K	number of primary LR images.	$\underline{U}_{m,i}$	i -th <i>sub</i> PPC of the m -th secondary PPC.
K^S	number of secondary LR images.	\underline{U}_n	n -th primary PPC.
m	index to identify secondary PPCs.	$\underline{U}_{n,j}$	j -th <i>sub</i> PPC of the n -th primary PPC.
$M_1 \times M_2$	size of the HR image.	\mathcal{V}	noise (in the data) matrix.
n	index to identify primary PPCs.	\underline{w}	estimation error of the expansion coefficients of a primary PPC.
N	total number of primary and secondary LR images.	$\ \underline{x}\ ^2$	squared norm of the <i>error-free</i> expansion coefficients of a primary PPC.
p	number of pixels in a <i>sub</i> LR image/ <i>sub</i> PPC.	X	matrix containing all the expansion coefficients of all the primary PPCs.
R_v	covariance matrix of noise in a primary LR image.	Y	primary data matrix (primary LR basis).
R_w	covariance matrix of the estimated expansion coefficients of a primary PPC.	Y_0	noise-free primary data matrix.
$T_m(n)$	mapping function: finds the index of the <i>sub</i> PPC of the m -th secondary PPC, that is shared with the n -th primary PPC.	Y^S	secondary data matrix (secondary LR basis).
$T_n(m)$	mapping function: finds the index of the <i>sub</i> PPC of the n -th primary PPC, that is shared with the m -th secondary PPC.	\underline{y}_k	k -th primary LR image.
$\text{Tr}(\cdot)$	trace of a matrix.	\underline{y}_k^S	k -th secondary LR image.
u	HR image.	\underline{y}_k^{sub}	k -th <i>sub</i> LR image.
\underline{U}_m	m -th secondary PPC.		

Acronym	Description
HR	high resolution.
LR	low resolution.
LS	least squares.
LSI; LSV	linear shift-invariant; linear shift-variant.
MD	median filter.
PCA	principal component analysis.
PCs	principle components.
PPC	polyphase component.
PSF	point spread function.
SR	super-resolution.
TSVD	truncated singular value decomposition.
TLS; TRTLS; TTLS	total least squares; Tikhonov regularized total least squares; truncated total least squares.
TV	total variation
UM	unsharp masking.

ABSTRACT

Multiframe super-resolution is the problem of reconstructing a single high-resolution (HR) image from several low-resolution (LR) versions of it. We assume that the original HR image undergoes different linear transforms that can be approximated as a set of linear shift-invariant transforms over different subregions of the HR image. The linearly transformed HR image is then downsampled, resulting in different LR images. Under the assumption of linearity, these LR images can form a basis that spans the set of the polyphase components (PPCs) of the HR image. We propose sampling diversity, where a reference PPC, of different sampling, is used to make known portions (*subpolyphase components*) of the PPCs of the HR image. To estimate the reference PPC, LR images are acquired using two imaging sensors with different sensor densities. This setup allows for blind reconstruction of the polyphase components of the HR image by solving a few small linear systems of equations where the number of unknowns is equal to the number of available LR images. The parameters we estimate are the expansion coefficients of the PPCs in terms of the LR basis, using the *subpolyphase components*. Both synthetic and real data sets are used to test the algorithm. The major features of our approach are: (1) it is blind, so that unknown motion and blurs can both be incorporated; (2) it is fast, in that

only small linear systems of equations need to be solved; and (3) it is robust, in that it avoids the problem of system model errors by treating the LR images as basis for reconstructing the polyphase components of the HR image.

CHAPTER I

Introduction

Image resolution is determined by two main factors. Blurring, due to optical limits and various other processes (like the effect of the atmosphere and motion blur, for example), results in soft images, while low-sensor density of the imaging device causes aliasing. Signal processing based super-resolution (SR) methods are typically concerned with overcoming the resolution limitation resulting in aliasing (although such techniques do take blur into consideration). In this context, ‘resolution’ refers to the sampling interval, or pixel size. Coarse sampling (pixels of relatively large size) results in ‘low resolution’ images, while ‘high resolution’ images correspond to fine sampling (pixels of relatively small size)¹. This is in contrast to optical super-resolution where the aim is to beat the diffraction limit² [40]. Optical SR methods are expensive and are usually developed to enhance the resolution of an already expensive imaging system [41] that is capable of producing very high resolution images (up to the diffraction limit). Henceforth, the term ‘super-resolution’ shall be used exclusively to refer to the process of overcoming the sensor density limitation using signal processing methods³.

Multiframe super-resolution is a technique that provides a cheap alternative to increasing the sensor density of an imaging chip, by combining multiple low-resolution (LR) images into a high-resolution (HR) image [1]. In particular, for more pixels, one

¹ For example, an image of an actual width of 0.5 meter, and height of 0.5 meter, can be sampled at 1000 samples per meter, in each direction, to obtain an image of 500x500 pixels. At a lower sampling rate of 200 samples per meter, we obtain a lower resolution image of 100x100 pixels.

² Diffraction of light results in blurring. It defines the maximum limit on resolution (acutance) of the optical system.

³ A diffraction-limited imaging system can still benefit from signal processing-based super-resolution techniques when imaging a larger field of view (zooming out). See §5.1 for details.

could either use a larger imaging chip, and consequently a larger lens will be needed, or decrease the pixel size which requires very high quality photo sensors that can perform well under deprived light conditions. Both options result in a substantial increase in cost. A third, much cheaper option, is to use super-resolution techniques.

Beyond cost reduction considerations, there are optimal physical limits on pixel density (and chip/lens size). For example, particularly large pixel spacing is required in some applications (in infrared imaging, for example [73]). Therefore, super-resolution is the only option when the optimal physical limits of sensor manufacturing (or the imaging system) are met.

The classical solution of the multiframe super-resolution problem is based on the following premise: given relative scene motions, we get different LR frames that can be combined into a HR image. In order for the scene motion to be useful in conventional multiframe SR techniques it must be different from frame to frame and modeled as a linear transformation. For example, the motion could be global (pure translations), local (general *linear* warping) or due to rotation. For many motion-based SR methods, the estimation of motion information (registration) is needed as a preliminary step. Typically, these methods assume available motion information or implement one of the available registration techniques [18, 19]. The extra computational load, required by the registration process, can be significant for cases more complex than the global motion model.

In order to reduce the effect of registration error on the super-resolved image, some methods, e.g. [43, 44], jointly estimate the motion parameters and the HR resolution image [1].

Also, these classical methods incorporate in their models the presence of both blur and noise as unwanted terms. Most of these techniques either assume that the blurring kernel(s) is known or could be identified via one of the blind blur identification methods [20]. Also, additive white Gaussian noise is usually assumed.

1.1 Overview of Super-resolution Methods

As our proposed method adopts a novel and completely different approach, in this thesis, we provide a very brief review of SR methods.

Super-resolution reconstruction started as a frequency-domain technique. The original idea of dealiasing in the frequency domain dates to [3] and was improved by others, for example [4-6]. These methods are theoretically simple and computationally efficient. However, their use is restricted to the case of pure translational motion and more importantly they are sensitive to errors [1, 14].

A more robust approach is solving the problem in the spatial domain. In fact, all modern techniques adopt the spatial (pixel) domain approach where the solution of a very large scale, ill-posed system of linear equations is sought. Different spatial domain methods use different assumptions and different approaches to the solution of the same matrix formulation and they are, in general, computationally expensive. This is especially true for projection type methods [16, 17]. Refer to [1, 2] for a comprehensive review of these and other techniques.

Elad and Hel-Or [7] provide a spatial domain solution to the special case of pure translation problem treated in [3-6]. They take advantage of this special case to develop a fast algorithm and optimality of their solution is shown to be in the maximum likelihood (ML) sense.

In [12] the authors adopt a completely deterministic approach to the solution of the large system of equations. Blurring is assumed to be known and the same for all acquired LR images and as is the case with typical motion-based SR techniques, the authors assume that the registration information is either available or estimated using one of the available image registration methods. They implement Tikhonov regularization to stabilize the solution with the regularization parameter automatically determined using the generalized cross-validation method (GCV). They provide a proof for the GCV formula for underdetermined systems and conjugate gradient (CG) algorithm is then used to iteratively solve the large system of linear equations. To accelerate convergence they derive and implement preconditioners. Later in [13] the authors improve on their previous work by developing a parametric estimation of the blur.

Other researchers, for example [8, 9, 15], have considered implementing *stochastic* regularization where *a priori* knowledge of the distribution of the HR image is used to constrain or stabilize the solution. In [8] the authors show that using a maximum *a posteriori* (MAP) estimator reduces the problem to solving the same huge system of

equations with the regularization term being *stochastically* determined. Stochastic regularization can have the advantage of edge-preserving reconstruction when the image prior's distribution model is accurate [1].

For its edge-preserving properties, the authors in [14] advocate using bilateral total variation method rather than Tikhonov regularization. Inspired by [42], the authors in [14] use the L1-norm for the data-fitting term, which gives solutions that are robust to outliers and registration errors. Their algorithm is relatively fast and when specialized to the case of pure translations it becomes even faster.

Unlike the conventional motion-based SR techniques, multiframe *motionless* SR does not require relative motion to estimate the HR image. This class of multiframe SR methods seeks HR image reconstruction using different blurs, zoom or photometric cues, and whole publications are devoted to this special class of SR techniques, for example [34, 9-11]. In fact, it was first shown in [8] that motionless SR is possible from differently blurred images. In contrast to motion-based SR, which treats the blurring process as a nuisance, in motionless SR the blurs are taken advantage of to produce a HR image. Blur-based motionless SR techniques usually assume that the blurs are known, but there are some attempts (for example [34]) at blindly de-mixing the *polyphase components* of the HR resolution image by treating the problem as a multiple input multiple output (MIMO) system with the input being the polyphase components. The authors in [34], however, reported that their blind method is very sensitive to error.

A recently active area in the field is single-frame super-resolution, where a HR image is obtained from a single LR frame using a training set of images of similar statistical nature [37]. The performance is dependent on the size and choice of the set of *example* images. Such *learning-based* methods are expected to perform well when specialized to super-resolving images with specific structure like face images [35, 36].

1.2 Contribution

The characteristics of our work can be summarized in the following points.

- **Motion or blur, both are useful:** The original, high-resolution image is assumed to undergo different *unknown* linear transforms and thus different undersampled versions of it are available. These different linear transformations of the original

image could be different distortion (e.g. blurring) processes or due to motion (global or local). Therefore, our work is different in the sense that we can make use of either motion or blur. This is different from motion-based methods in that they only make use of motion and incorporate blur in their model as a nuisance term. It is also different from the blur-based motionless algorithms, as these do not incorporate motion at all in their model.

- **LR images as basis:** Instead of reconstructing the HR image directly, we solve for the expansion coefficients of its polyphase components (PPCs) in terms of the available LR images under the assumption that the LR images can form a basis to reconstruct the polyphase components.
- **Blind reconstruction via sampling diversity:** Since we solve for the expansion coefficients of the PPCs in terms of the LR images, our proposed method is *blind* in the sense that, unlike other multiframe SR algorithms, our method requires no registration or blur estimation. These coefficients are estimated using only a tiny portion (a *subpolyphase* component) of each PPC. These *subpolyphase* components are determined via the property of *sampling diversity* (chapter II) by using a single PPC, corresponding to a *different* downsampling factor, as a reference. This reference PPC can be estimated using two sets of LR images, captured with two different imaging chips with different sensor densities (chapter IV).
- **Speed:** Our method involves the solution of a few small linear systems of equations where the number of unknowns is equal to the number of available LR images. This implies that the implementation of the method is inherently fast.

1.3 Applications

We list here a few examples of practical cases on which our algorithm could be used:

- Just like every motion-based SR technique, our method can handle the classical problem of achieving SR using (approximately) pure translational sub-pixel shifts. However, unlike previous work, our fast blind reconstruction method does not require registration as a preliminary step.

- Because of the *random* nature of the motion blur associated with vibrating imaging systems, conventional registration methods perform poorly, and as a result, the performance of conventional motion-based SR methods suffers. In our case, the randomness of the motion blur is actually a desired quality and no estimation of the motion blur or image registration is needed, and images are super-resolved fast, and all for the simple hardware requirement of adding another (secondary) lower resolution CCD sensor.
- When the imaging medium is the turbulent atmosphere, the effect can be modeled as a time-variant, shift-variant point spread function (PSF). In §5.1 we discuss the applicability of our method in this scenario.

1.4 Thesis Outline

This thesis is organized as follows. In chapter II, we introduce a novel approach to the problem of multiframe super-resolution where the set of LR images is viewed as a basis, in terms of which, the PPCs of the HR image can be represented. In addition, we introduce the property of sampling diversity which reveals a tiny portion (a *subpolyphase* component) of each one of the PPCs, using a *reference* PPC of *different* sampling. In chapter III, we investigate different classical methods to solve for the expansion coefficients of the PPCs in terms of the LR basis, using the *subpolyphase* components. In chapter IV, we address the problem of estimating the reference PPC, which can only be achieved using two sets of LR images captured by two different imaging sensors with different sensor densities. Applications and experimental results are discussed in chapter V, and the thesis is concluded in chapter VI.

CHAPTER II

A Novel Approach to Multiframe Super-resolution

2.1 Introduction

The general setup for the motion-based multiframe super-resolution problem is as follows [14]. Assuming that the original scene remains constant during the acquisition of K low-resolution images, each measured LR image is the result of different *relative* scene motion, blurring effects, and usually with a common downsampling factor that is the same in the horizontal and vertical directions, and additive noise corruption. In matrix formulation this translates to

$$(2.1) \quad \underline{y}_k = \mathcal{D} H_k^{cam} F_k H_k^{atm} \underline{u} + \underline{v}_k \quad \text{for } k = 1, \dots, K,$$

where \underline{y}_k is the lexicographical column-vector representation of the k -th $m_1 \times m_2$ LR image, $\underline{y}_k, \underline{u}$ is the lexicographical representation of the $M_1 \times M_2$ HR image, u, F_k is the motion matrix of size $M_1 M_2 \times M_1 M_2$, H_k^{atm} is the $M_1 M_2 \times M_1 M_2$ matrix representation of the k -th atmospheric blurring effect, H_k^{cam} is the $M_1 M_2 \times M_1 M_2$ matrix representation of the k -th camera blur, \mathcal{D} is of size $m_1 m_2 \times M_1 M_2$ and represents the decimation operation, and \underline{v}_k is the noise vector. The term ‘atmospheric blur’ shall refer to blurring due to atmospheric, and other types of blur (e.g. motion blur), that are not the direct result of the limitations of the imaging system. The camera’s optical blur and CCD integrating effect are represented by H_k^{cam} . Because H_k^{atm} and F_k can be represented with block circulant matrices, they commute [14] and (2.1) can be re-written as

$$\begin{aligned}
\underline{y}_k &= \mathcal{D}H_k^{cam} H_k^{atm} F_k \underline{u} + \underline{v}_k \\
(2.2) \quad &= \mathcal{D}H_k F_k \underline{u} + \underline{v}_k \quad \text{for } k = 1, \dots, K,
\end{aligned}$$

where $H_k = H_k^{cam} H_k^{atm}$ merges the blur effect in one matrix representation. See Figure 2.1 (b), for a graphical depiction of (2.2).

As mentioned in chapter I, typical classical SR reconstruction techniques assume F_k to be known and usually assume the blurring process to be *known* and it is viewed as an unwanted term. On the other hand, blur-based motionless SR takes advantage of the known blurring process if it is different for each measured image, and it assumes F_k to be the identity matrix [8]. The additive noise is usually assumed to be white Gaussian noise. Combining the equations in (2.2), we get

$$\begin{aligned}
\begin{bmatrix} \underline{y}_1 \\ \vdots \\ \underline{y}_K \end{bmatrix} &= \begin{bmatrix} \mathcal{D}H_1 F_1 \\ \vdots \\ \mathcal{D}H_K F_K \end{bmatrix} \underline{u} + \begin{bmatrix} \underline{v}_1 \\ \vdots \\ \underline{v}_K \end{bmatrix} = \begin{bmatrix} S_1 \\ \vdots \\ S_K \end{bmatrix} \underline{u} + \underline{\mathcal{V}} \\
(2.3) \quad &\Leftrightarrow \underline{Y} = S \underline{u} + \underline{\mathcal{V}}.
\end{aligned}$$

Note that the system of equations (2.3) is overdetermined if $K > I^2$, where I is the downsampling factor in the vertical and horizontal directions.

The size of the system matrix S in (2.3) is $Km_1m_2 \times M_1M_2$, which is so huge that storing it (let alone trying to directly compute its inverse) is impractical. For example, if the size of the HR image is 500x500, then an (over)determined system matrix will have (at least) $250,000 \times 250,000 = 62,500,000,000$ elements. In addition, the ill-posedness¹ of the problem (the system matrix is near singular [1, 14]) means that solving this problem without regularization will magnify the noise effect.

Conventional spatial domain SR methods are different from each other mainly in terms of how to deal with this huge size, ill-posed, inverse problem². They mainly differ in defining a regularization term that stabilizes the solution and subsequently deriving a

¹ A problem is said to be well-posed when a solution exists, is unique and stable [25].

² Estimating the system parameters from the data is the first step of solving an inverse problem. Assuming the system matrix is known, an inverse problem entails reversing the process that produced the observed data (e.g. by inverting the system matrix). The majority of spatial SR methods are formulated as an inverse problem with the assumption that the system parameters are known.

numerical algorithm to solve the problem efficiently. However, the speed of even the fastest of these algorithms is limited by the fact that the number of the unknowns in (2.3) is equal to the number of pixels in the HR image itself (e.g., 250,000 unknowns, for a HR image of size 500x500).

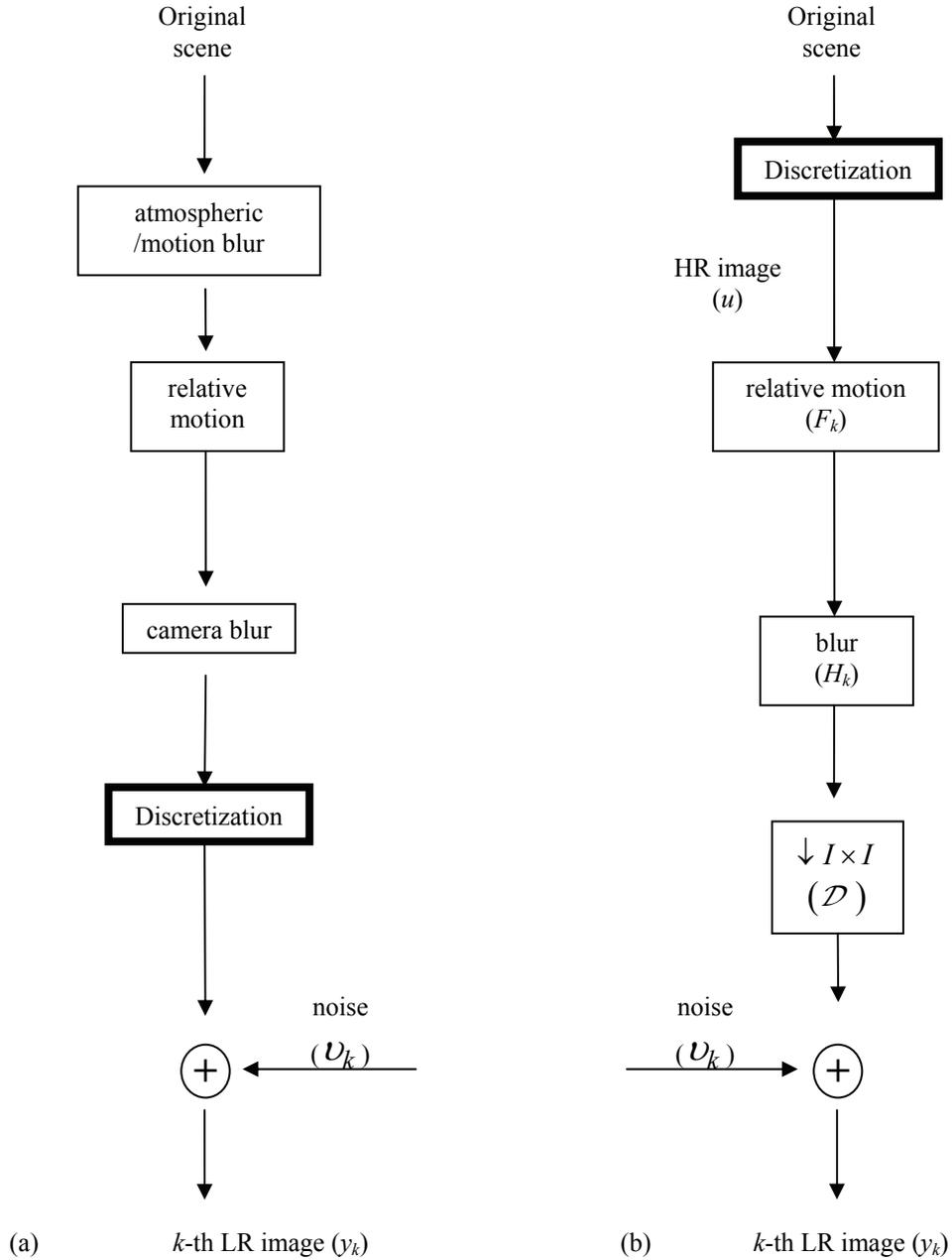


Figure 2.1: The observation model. (a) the actual physical process of image acquisition. (b) equivalent discrete observation model.

2.2 Low-Resolution Images as Basis Signals

A polyphase component (PPC) of a HR image is a shifted and downsampled version of it. Given that the downsampling factor, I , is the same in the vertical and horizontal direction, a HR image can be decomposed into I^2 PPCs. The first PPC is obtained by starting with the first pixel in the first row of the HR image, and downsampling by $\downarrow I \times I$. Downsampling, starting with the second pixel in the first row, we get the second PPC. The I -th PPC corresponds to downsampling beginning with the I -th pixel in the first row of the HR image. For the $(I+1)$ -th PPC, we move to the second row and downsample beginning with the first pixel in that row. The I^2 -th PPC is obtained by downsampling starting with the I -th pixel in the I -th row.

Since a low-resolution (LR) image has the same number of pixels of a PPC of the HR image, it is rather intuitive to expect that, under some conditions, a PPC can be written as a linear combination of the LR images. One extreme case where this is always guaranteed is obviously when the number of (linearly independent) LR images (or any other signals with the same dimensions) is equal to the number of pixels in a PPC. On the other hand, if only one LR image is available, the reconstructed PPCs, in terms of this single LR image, will be merely scaled versions of it. In order for this idea to be useful, a reasonable number of LR images should be enough to build a basis for representing the PPCs. But why should we be interested in any of this when the PPCs (of the HR image) are not known and our goal is to estimate them? The answer to this question shall be revealed in §2.3, chapter III and chapter IV.

2.2.1 LSI Transforms

The conventional matrix formulation (2.3) used for spatial domain SR methods can be replaced with a much more efficient formulation if each LR image is a decimated version of the HR image after going through a finite support linear shift invariant (LSI) transform (e.g., a finite impulse response (FIR) filter or a point spread function (PSF))

$$(2.4) \quad \begin{aligned} \begin{bmatrix} \underline{y}_1 & \cdots & \underline{y}_K \end{bmatrix} &= \begin{bmatrix} \underline{u}_{1,1} & \cdots & \underline{u}_{L_1, L_2} \end{bmatrix} \begin{bmatrix} \underline{h}_1 & \cdots & \underline{h}_K \end{bmatrix} + \begin{bmatrix} \underline{v}_1 & \cdots & \underline{v}_K \end{bmatrix} \\ \Leftrightarrow Y &= UH + \mathcal{V}, \end{aligned}$$

where \underline{h}_k is the lexicographical unwrapping of the k -th FIR filter coefficients of size $L_1 \times L_2$ and if I is the downsampling factor, then $L_1 \geq I$ and $L_2 \geq I$ must be satisfied. The vector $\underline{u}_{\ell_1, \ell_2}$ is also the unwrapping (by column) of the (ℓ_1, ℓ_2) -th submatrix of the original image u , defined below

$$(2.5) \quad \begin{aligned} u_{\ell_1, \ell_2}(k_1, k_2) &\triangleq u(k_1 I + L_2 - \ell_1 + 1, k_2 I + L_1 - \ell_2 + 1) \\ \text{for } \ell_1 &= 1, \dots, L_2, \quad \ell_2 = 1, \dots, L_1. \end{aligned}$$

If $L_1 = I$ and $L_2 = I$, then all these submatrices $\{u_{\ell_1, \ell_2}\}$ are the polyphase components of the HR image u . If, however, $L_1 > I$ and $L_2 > I$ then only I^2 of these are the polyphase components. Let c be the column index of the image matrix, U . Then \underline{U}_c (c -th column in U) is one of the I^2 unwrapped polyphase components if

$$(2.6) \quad c \in \bigcup_{q=0}^{I-1} \{qL_1 + 1, qL_1 + 2, \dots, qL_1 + I\}.$$

Equation (2.4) is therefore a convenient reformulation of multiple 2-D convolution operations followed by decimation. In addition, each LR image, \underline{y}_k is assumed to be obtained by *cropping* the convolved (*transformed*) HR image and then decimation, so there are no convolution terms in which the shifted kernel overflows the image support. This is known as the more practical “partial data” case [21].

If all the kernels (PSFs or FIR filters) are known and are linearly independent, and if the number of available LR images, $K \geq L_1 L_2$, then U could be estimated using least squares (since the noise is assumed zero-mean white Gaussian) with a trivial computational cost, $\hat{U} = YH^T (HH^T)^{-1}$. Note that HH^T has the small size of $L_1 L_2 \times L_1 L_2$, and thus the computational cost depends mainly on the size of the kernels.

Of course, the kernels are not always known and using any algorithm to estimate them means substantial additional computations and according to simulations, even when the system matrix H is known and well-conditioned, adding small perturbations to it can result in large errors. This means that solving (2.4) is sensitive to estimation errors of the system matrix, H .

The idea that a LR image can be written as a linear combination of the PPCs is not new (although, the matrix formulation (2.4) is novel). In fact, in [34], the authors developed a motionless blur-based SR algorithm with computational complexity that is mainly dependent on the size of the blurs rather than the size of the HR image (unlike in [8] where the formulation (2.3) was still used, with the motion matrix set to identity). Their contribution was to *blindly* estimate *restoration filters* to recover the PPCs, but their algorithm is very sensitive to error. Similarly, solving (2.4) is sensitive to errors in the system matrix, and therefore there is little motivation to try to estimate the kernels. Moreover, even if we somehow could estimate the kernels quite accurately, the assumption that the different kernels must be of the *same* finite size is quite restrictive.

Nevertheless, equations (2.4-2.6) are useful in answering the question as to when the LR images can span a subspace for the PPCs. Specifically, these equations tell us that, for the case of the same size LSI kernels, the LR images are linear mixtures of the PPCs ‘and’ *other* image sub-matrices (rearrangements of elements of the PPCs). Therefore, when the LR images are mixtures of K' submatrices (including I^2 PPCs) then we need K' mixtures (LR images) in order to be able to write the PPCs as linear combinations of LR images.

Now suppose we have available the PPCs and we calculate their expansion coefficients in terms of a set of different LR images that do not satisfy the assumptions exactly (LSI, same finite support kernels and sufficient number of LR images) and then using these expansion coefficients we reconstruct the PPCs. In another scenario, where we have exact knowledge of the transform kernels, suppose we ‘approximate’ them (the kernels) to *fit* our model (2.4) and then solve the problem. Which one of the two scenarios is expected to give better results? Noting that in the first case there is no wrong solution but rather a possibly incomplete one, we can easily expect the reconstructed PPCs of the first scenario to be much better.

Essentially, equations (2.4-2.6) give insight (under the LSI assumption) as to *how many* LR images might be enough to fully represent the PPCs but this does not mean that the PPCs cannot be represented, at least partially, by any number of available LR images. While formulations like (2.3, 2.4) are inverse problems, and as such are sensitive to model errors, finding the expansion coefficients of the PPCs is simply a change of basis.

2.2.2 LSV Transforms

When the HR image undergoes a linear shift-variant (LSV) transformation that can be approximated as a set of *local* LSI transforms³ (over different subregions of the HR image) then the previous discussion can be readily extended to the case of LSV transforms.

To be more precise, suppose the LSV transform can be approximated as r LSI kernels over r different subregions of the HR image. One option is to treat these subregions as r different HR images where we can reconstruct the PPCs of each one of them separately⁴. Alternatively, we can reconstruct the PPCs of the *whole* HR image but with r times more LR images⁵. This is because in the case of a LSV transform, a LR image can be viewed as a linear *local* mixing of subregions of the PPCs and therefore to reconstruct each PPC as a whole, we need r times more LR images than it is required in the LSI case.

For example, suppose a *square* HR image undergoes a LSV transform that can be approximated as 4 LSI kernels over the 4 quadrants of the HR image, each with approximately equal finite support of size 3×3 . The linearly transformed HR image is then downsampled by 3×3 to produce the LR images as shown below.

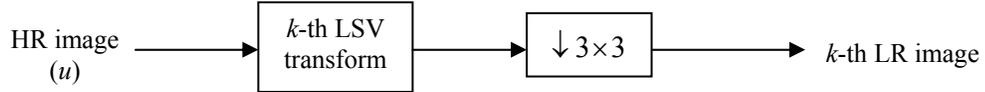


Figure 2.2: LR images obtained from a HR image via LSV transformation and downsampling.

In light of (2.4-2.6), we know that the ℓ -th quadrant of the k -th LR image can be written as a linear combination of the ℓ -th quadrants of the 9 PPCs of the HR. This means that the *whole* of the k -th LR image can be written as

$$(2.7) \quad y_k = \sum_{\ell=1}^r \sum_{n=1}^{I^2} \alpha_n^\ell (U_n \odot Z^\ell),$$

³ Rotation of an image is an example of a linear transform that cannot be approximated as a set of local LSI transforms.

⁴ Reconstruction of subregions of the HR image separately has a downside as will be discussed in §2.3.

⁵ Although the LSV case does require more LR images, according to simulations, good results can be achieved with a smaller number than recommended.

where \odot denotes the element-wise multiplication operator, U_n is the n -th PPC of the HR image, Z^ℓ is an all-zero matrix except for the elements corresponding to the ℓ -th quadrant, which are all equal to 1, and $\{\alpha_n^\ell\}_{n=1}^{I^2}$ is the ℓ -th set of linear combination coefficients. These are the elements of the ℓ -th LSI kernel. See Figure 2.3 for an illustration of equation (2.7). Naturally, since a LR image is composed of $rI^2 = 4 \times 9 = 36$ separate parts of the PPCs, then in order to be able to write the PPCs as linear combinations of the LR images, in the LSV case, we will need $K \geq rI^2$ LR images. Note that if the size of an LSI kernel is $L_1 \times L_2$, $L_1 > I$ and $L_2 > I$, then we need $K \geq rL_1L_2$, for a complete basis.

$$\begin{aligned}
& \alpha_1^1 \times \begin{array}{|c|c|} \hline 1^{\text{st}} & 0 \\ \hline \text{quad. of} & \\ \hline 1^{\text{st}} \text{ PPC.} & \\ \hline 0 & 0 \\ \hline \end{array} + \alpha_2^1 \times \begin{array}{|c|c|} \hline 1^{\text{st}} & 0 \\ \hline \text{quad. of} & \\ \hline 2^{\text{nd}} \text{ PPC.} & \\ \hline 0 & 0 \\ \hline \end{array} + \dots + \alpha_9^1 \times \begin{array}{|c|c|} \hline 1^{\text{st}} & 0 \\ \hline \text{quad. of} & \\ \hline 9^{\text{th}} \text{ PPC.} & \\ \hline 0 & 0 \\ \hline \end{array} \\
& + \\
& \alpha_1^2 \times \begin{array}{|c|c|} \hline 0 & 2^{\text{nd}} \\ \hline & \text{quad. of} \\ \hline & 1^{\text{st}} \text{ PPC.} \\ \hline 0 & 0 \\ \hline \end{array} + \alpha_2^2 \times \begin{array}{|c|c|} \hline 0 & 2^{\text{nd}} \\ \hline & \text{quad. of} \\ \hline & 2^{\text{nd}} \text{ PPC.} \\ \hline 0 & 0 \\ \hline \end{array} + \dots + \alpha_9^2 \times \begin{array}{|c|c|} \hline 0 & 2^{\text{nd}} \\ \hline & \text{quad. of} \\ \hline & 9^{\text{th}} \text{ PPC.} \\ \hline 0 & 0 \\ \hline \end{array} \\
& + \\
& \vdots \\
& + \\
& \alpha_1^4 \times \begin{array}{|c|c|} \hline 0 & 0 \\ \hline & \\ \hline & 4^{\text{th}} \\ \hline 0 & \text{quad. of} \\ \hline & 1^{\text{st}} \text{ PPC.} \\ \hline \end{array} + \alpha_2^4 \times \begin{array}{|c|c|} \hline 0 & 0 \\ \hline & \\ \hline & 4^{\text{th}} \\ \hline 0 & \text{quad. of} \\ \hline & 2^{\text{nd}} \text{ PPC.} \\ \hline \end{array} + \dots + \alpha_9^4 \times \begin{array}{|c|c|} \hline 0 & 0 \\ \hline & \\ \hline & 4^{\text{th}} \\ \hline 0 & \text{quad. of} \\ \hline & 9^{\text{th}} \text{ PPC.} \\ \hline \end{array}
\end{aligned}$$

Figure 2.3: In the LSV case, a LR image is a linear combination of separate parts of the PPCs of the HR image (a LR image can be viewed as a linear *local* mixing of subregions of the PPCs).

2.3 Sampling Diversity

In the previous section we explained that, under the assumption of linearity of the transformations, the I^2 polyphase components (PPCs) of the HR image can be written as linear combinations of the LR images, i.e.

$$(2.8) \quad \{\underline{U}_n\}_{n=1}^{I^2} \in R(Y),$$

where $R(Y)$ denotes the range (column space) of Y . Throughout the discussion in this section, we make the assumption that we have available only one of the J^2 PPCs of the HR image (corresponding to $\downarrow J \times J$), where I and J are two relatively prime integers. In other words, we assume that we know the m -th PPC, \underline{U}_m for some m between 1 and J^2 . Henceforth, we refer to this known PPC (of different sampling) as the *reference* PPC.

When I and J are relatively prime, the following property holds: any two PPCs corresponding to $\downarrow I \times I$ and $\downarrow J \times J$ respectively, share exactly⁶

$$(2.9) \quad \frac{M_1 M_2}{I^2 J^2}$$

pixels between them. These are the elements of a PPC corresponding to $\downarrow IJ \times IJ$. Said in a different way, if \underline{U}_n is one of the I^2 PPCs corresponding to $\downarrow I \times I$ and \underline{U}_m is one of the J^2 PPCs corresponding to $\downarrow J \times J$ and I and J are relatively prime, then \underline{U}_q , one of the $I^2 J^2$ PPCs of the HR image corresponding to $\downarrow IJ \times IJ$, is a *subpolyphase* component, $\underline{U}_{n,j}$, of \underline{U}_n corresponding to downsampling \underline{U}_n by $J \times J$, as well as a *subpolyphase* component, $\underline{U}_{m,i}$, of \underline{U}_m corresponding to downsampling \underline{U}_m by $I \times I$, for $q = T(m, n)$, $j = T_n(m)$, and $i = T_m(n)$, where T , T_n , and T_m , are 1-1 mappings between (m, n) , m , n and q, j, i , respectively. We refer to this property as the *sampling diversity* property. See Table 2.1 for a more concise definition of this property.

⁶ The number of common elements is *exactly* $(M_1 M_2 / I^2 J^2)$ when the dimensions of the HR image are integer multiples of IJ .

Therefore, if we know one of the J^2 PPCs of the HR image, then we already know (M_1M_2/I^2J^2) pixels in *each* one of the I^2 PPCs of the HR image⁷. In other words, knowing a single $\downarrow J \times J$ PPC of the HR image, means that we know a single *subpolyphase* component of each one of the I^2 PPCs. This property enables us to solve for the expansion coefficients of the polyphase components (chapter III), in terms of the LR images, without any knowledge of the distortion model that produced the LR images. We only make the assumption that (2.8) is valid and that a single $\downarrow J \times J$ PPC (the reference PPC) is known. In chapter IV we discuss how to estimate the reference PPC.

	HR image		
polyphase components	$\downarrow I \times I$ $\{\underline{U}_n\}_{n=1}^{I^2}$	$\downarrow IJ \times IJ$ $\{\underline{U}_q\}_{q=1}^{I^2J^2}$	$\downarrow J \times J$ $\{\underline{U}_m\}_{m=1}^{J^2}$
<i>subpolyphase</i> components	$\downarrow J \times J$ $\{\underline{U}_{n,j}\}_{j=1}^{J^2}$		$\downarrow I \times I$ $\{\underline{U}_{m,i}\}_{i=1}^{I^2}$

Table 2.1: Sampling diversity: when I and J are relatively prime, there exist 1-1 mappings T , T_n , and T_m , such that $\underline{U}_q = \underline{U}_{n,j} = \underline{U}_{m,i}$ for $q = T(m,n)$, $j = T_n(m)$, and $i = T_m(n)$.

2.3.1 An Illustration of the Property of Sampling Diversity

Suppose we have a HR image of size $M_1 = M_2 = 24$ and from it we obtained its 3rd $\downarrow 2 \times 2$ PPC. Suppose also that we downsampled the HR image by $\downarrow 3 \times 3$ to obtain the 9th $\downarrow 3 \times 3$ PPC. The sampling diversity property says that any two PPCs corresponding

⁷ In §2.2.2 we discussed that one option to deal with the LSV case is to super-resolve subregions of the HR image separately. The disadvantage of this approach is that the number of shared elements will be smaller since M_1 and M_2 in (2.9) will become smaller (the dimensions of *subregions* of the HR image).

to relatively prime downsampling factors must share exactly a *subpolyphase* component. So in this example the question is: which one of the *sub* PPCs of the 3rd $\downarrow 2 \times 2$ PPC ($U_{n=3}$) is equal to which one of the *sub* PPCs of the 9th $\downarrow 3 \times 3$ PPC ($U_{m=9}$)? By examining Figure 2.5, it is easy to see that the answer is the 8th and the 3rd, respectively (i.e. $j = T_n(m) = 8$, and $i = T_m(n) = 3$).

2.3.2 The Mapping Functions T_n and T_m

By examining many configurations, such as the one shown in Figure 2.5, for different⁸ $J = I + 1$, m and n , we derived the mapping functions T_n and T_m . Unfortunately, these do not seem to have a simple analytical form. We provide a description of these functions, below.

Function $T_n(m)$:

$$A_J \triangleq \begin{bmatrix} 1 & 2 & \cdots & J \\ J+1 & J+2 & \cdots & 2J \\ \vdots & \vdots & \ddots & \vdots \\ J^2 - J + 1 & J^2 - J + 2 & \cdots & J^2 \end{bmatrix}$$

$$T_n^1 = n + \left\lfloor \frac{n}{I} \right\rfloor - 1$$

$$r_n^1 = \left\lfloor \frac{T_n^1}{J} \right\rfloor$$

$$c_n^1 = J - (r_n^1 J - T_n^1)$$

$$B_J = \text{circshift}(A_J, [-r_n^1, -c_n^1])^T$$

$$\underline{T}_n = B_J(\cdot)$$

$$T_n(m) = \underline{T}_n(J^2 - m + 1).$$

⁸ Instead of the more general case of I and J being relatively prime, we restrict our discussion to the case of I and J being two consecutive integers (larger than 1) since this gives the largest possible number of common elements between any two PPCs corresponding to $\downarrow I \times I$ and $\downarrow J \times J$.

Function $T_m(n)$:

$$\begin{aligned}
A_I &\triangleq \begin{bmatrix} 1 & 2 & \dots & I \\ I+1 & I+2 & \dots & 2I \\ \vdots & \vdots & \ddots & \vdots \\ I^2-I+1 & I^2-I+2 & \dots & I^2 \end{bmatrix} \\
\underline{v} &= [1 \quad I \quad I-1 \quad I-2 \quad \dots \quad 1]^T \\
d &= \text{mod}(m, J) \\
\text{if } d &= 0 \\
d &\leftarrow -1 \\
\text{end} \\
r &= \underline{v} \left(\left\lceil \frac{m}{J} \right\rceil \right) \\
c &= \underline{v}(d) \\
T_m^1 &= A_I(r, c) \\
r_m^1 &= \left\lceil \frac{T_m^1}{I} \right\rceil \\
c_m^1 &= I - (r_m^1 I - T_m^1) \\
B_I &= \text{circshift} \left(A_I, \left[\text{mod}(I - r_m^1 + 1, I), \text{mod}(I - c_m^1 + 1, I) \right] \right)^T \\
\underline{T}_m &= B_I(\cdot) \\
T_m(n) &= \underline{T}_m(n).
\end{aligned}$$

Note: $\text{circshift}(A, [r, c])$ is a function that circularly shifts down the rows in matrix A by r , and it circularly shifts its columns to the right by c . If r is negative the rows are shifted upwards. If c is negative, the columns are shifted to the left.

2.3.3 The Hardware Requirements

In the previous sections we explained how the property of sampling diversity gives us a small part (a *subpolyphase* component) of each one of the $\downarrow I \times I$ PPCs, when we know a single $\downarrow J \times J$ PPC, and I and J are relatively prime. In chapter III we investigate how

to use these *subpolyphase* components to find the expansion coefficients (in terms of the available LR images) of all the I^2 PPCs of the HR image. In chapter IV, we address the problem of estimating a single $\downarrow J \times J$ PPC, which we refer to as the reference PPC.

As we will see in chapter IV, the estimation of the reference PPC is possible if we have two imaging sensors (e.g. two CCD arrays) with different sensor densities corresponding to $\downarrow I \times I$ and $\downarrow J \times J$, respectively. We shall refer to the CCD array with the higher sensor density, as the primary CCD sensor; the secondary CCD sensor is the one with the lower density⁹.

These sensors must therefore be designed to satisfy the requirement of relatively prime downsampling. In particular, if we want to reconstruct HR images of size $M_1 \times M_2$, where M_1 and M_2 are integer multiples of IJ , and $J > I$, then the primary CCD array must have

$$m_1 \times m_2 = \left(\frac{M_1}{I} \times \frac{M_2}{I} \right)$$

pixels and the secondary CCD array, must have

$$m_1^S \times m_2^S = \left(\frac{I}{J} m_1 \times \frac{I}{J} m_2 \right)$$

pixels. For example, if we want to get super-resolved images of size 3000x3000, corresponding to 4x4 resolution enhancement, then we should use a primary CCD array of size 750x750 and a secondary CCD array of size 600x600.

Working with two CCD arrays, means that we could either use two cameras, or install both sensors in the same camera. Aside from the extra cost associated with the first option, two cameras cannot capture the same scene except when imaging at a long range. For close-up images, we should take into account the framing errors due to parallax¹⁰.

The other option of using two sensors in one camera is a lot cheaper and much simpler without the need to correct for framing errors. For example, we could use a beam splitter

⁹ The idea of using a secondary CCD to help with solving an *entirely different problem* was suggested in [51], where *phase diversity* is achieved by placing the secondary sensor intentionally out of focus. This helps to jointly estimate the image and aberrations.

¹⁰ Parallax is the apparent displacement of an object viewed along two different lines of sight.

which is an optical device (a half-silvered mirror or a cube prism) that splits a beam of light in two, where half of the light is transmitted through (to the primary CCD array) while the other half is reflected, at a right angle (towards the secondary CCD array). The only disadvantage of using a beam splitter is that the signal-to-noise ratio (SNR) will decrease by 6 dB since only half the amount of light reaches the sensors. Using a larger aperture allows more light in, at the expense of loss of depth of field¹¹. Another solution is using a *non-stationary* 100% reflective mirror that moves in the optical path, for only half of the imaging time, reflecting all the light towards the secondary sensor.

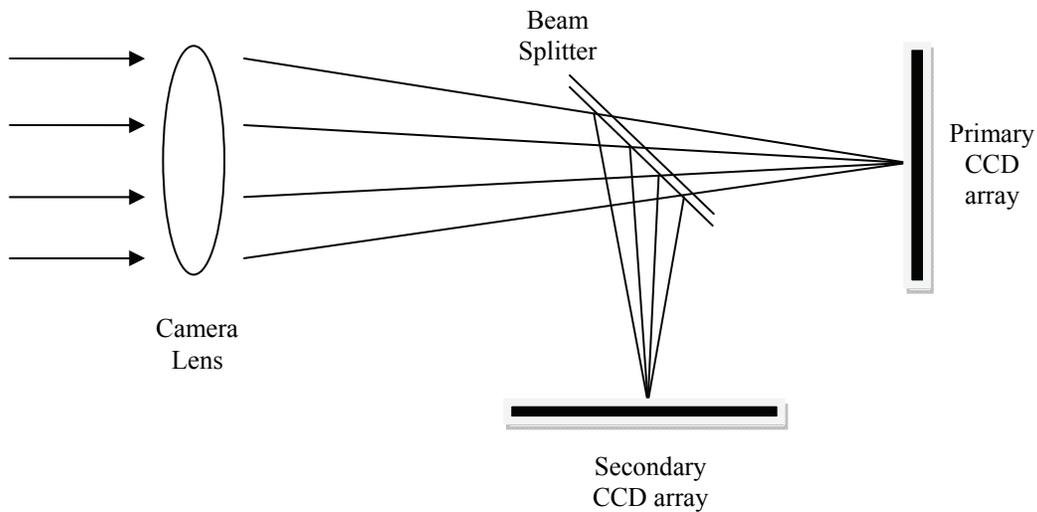


Figure 2.4: A two-CCD sensor camera configuration, using a beam splitter.

¹¹ Depth of field is the portion of an image that appears sharp due to focusing at only one distance. The loss of sharpness as we move away from the focus point is gradual and is proportional to the aperture size.

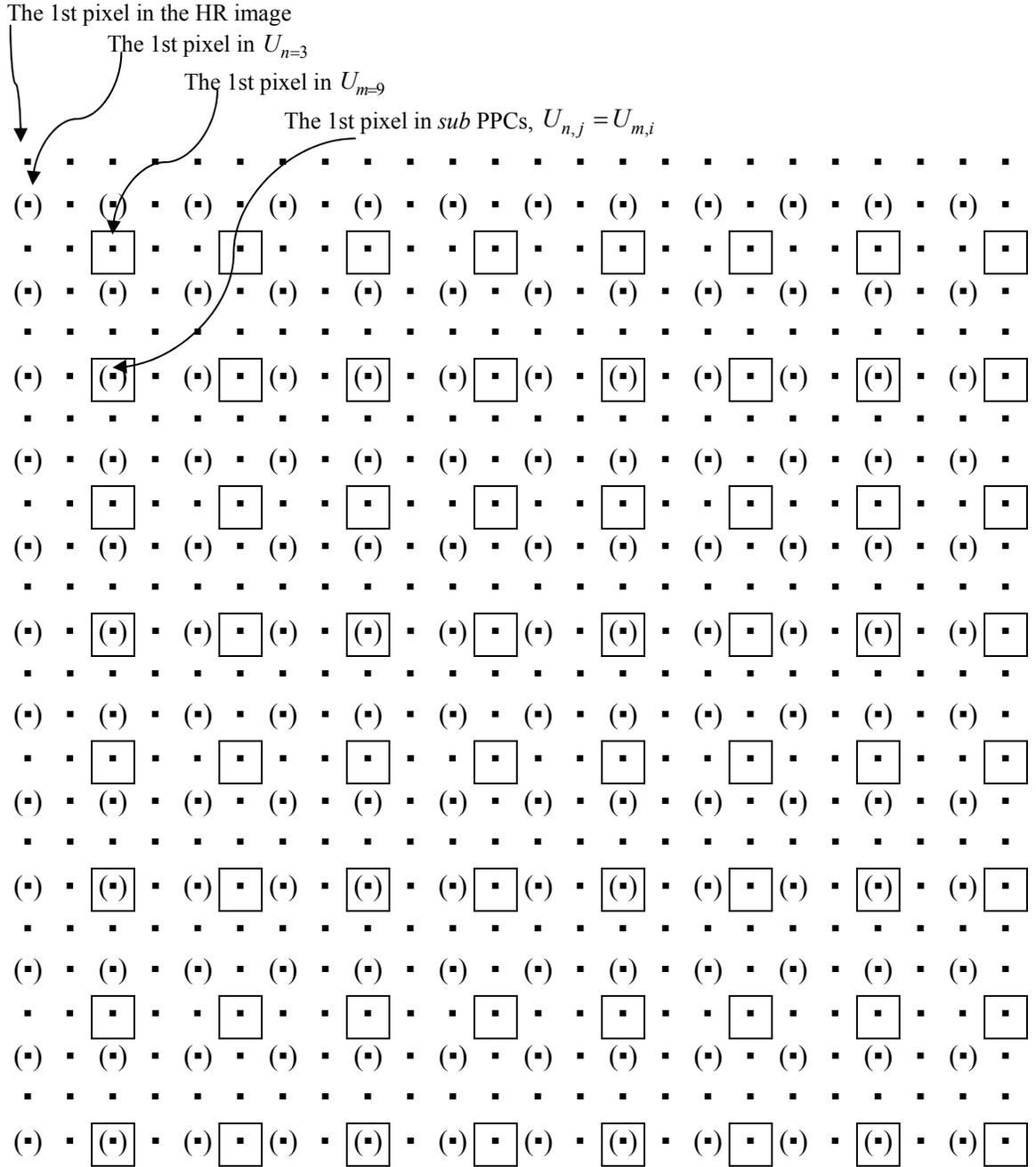


Figure 2.5: An illustration of the property of sampling diversity. For $I = 2$, $J = 3$, $n = 3$ (the 3rd out of 4 $\downarrow 2 \times 2$ PPCs) and $m = 9$ (the last of the 9 $\downarrow 3 \times 3$ PPCs), the polyphase components U_n and U_m , have subpolyphase components $\{U_{n,j}\}_{j=1}^9$ and $\{U_{m,i}\}_{i=1}^4$, respectively and $U_{n,j} = U_{m,i}$ for $j = 8$ and $i = 3$.

CHAPTER III

Solving for the Expansion Coefficients of the Polyphase Components

3.1 Introduction

In chapter II, we explained how the property of sampling diversity can be used to find portions (*sub* PPCs) of all the $\downarrow I \times I$ PPCs of the HR image, with the help of a *reference* PPC of *different* sampling. In addition, we noted that under the assumption of the linearity of the transforms, the LR images can be viewed as a basis spanning a subspace where the PPCs exist.

Our goal, in this chapter, is to find the expansion coefficients of the PPCs in terms of the LR basis, using their *sub* PPCs. The diagram in Figure 3.1 gives a pictorial summary of how the PPCs are reconstructed.

3.1.1 The LS Solution

Suppose we have a *perturbed* version of one of the $\downarrow J \times J$ PPCs, \underline{U}_m , for some m between 1 and J^2 , and using it as our reference PPC, we obtain a *sub* PPC of each one of the $\downarrow I \times I$ PPCs, $\{\underline{U}_n\}_{n=1}^{I^2}$. In other words, using the reference PPC, we obtain the I^2 *sub* PPCs, $\{\underline{U}_{n,j}\}_{n=1}^{I^2}$, for $j = T_n(m)$. Because the reference PPC contains error, all the *sub* PPCs will be noisy as well. Namely, the j -th *sub* PPC, $\underline{U}_{n,j}$ is related to the n -th PPC, \underline{U}_n via

$$(3.1) \quad \underline{U}_{n,j} = D_j \underline{U}_n + \underline{e},$$

where D_j is a $\downarrow J \times J$ matrix (performing shifting and decimation) that gives us the j -th *sub* PPC from the n -th PPC, and \underline{e} is assumed to be zero-mean, white Gaussian noise, i.e.

$$\underline{e} \sim \mathcal{N}(\underline{0}, R),$$

with $R = \sigma_e^2 I_p$, where I_p is the identity matrix of size $p \times p$ and $p = M_1 M_2 / I^2 J^2$ is the number of pixels in a *sub* PPC.

Now assume that the available LR images are noiseless and can span the PPCs, i.e.

$$(3.2) \quad \underline{U}_n = Y \underline{x}_n \quad \text{for } n = 1, \dots, I^2,$$

where \underline{x}_n are the expansion coefficients of \underline{U}_n in terms of the LR basis, Y . Substituting (3.2) in (3.1) we see that a *sub* PPC, $\underline{U}_{n,j}$ is Gaussian distributed,

$$p(\underline{U}_{n,j}; \underline{x}_n) = \frac{1}{(2\pi\sigma_e^2)^{p/2}} \exp\left(-\frac{1}{2\sigma_e^2} \|\underline{U}_{n,j} - D_j Y \underline{x}_n\|^2\right).$$

The maximum likelihood (ML) estimator of the expansion coefficients is therefore given by solving the minimization

$$\min_{\underline{x}_n} \|\underline{U}_{n,j} - D_j Y \underline{x}_n\|^2.$$

To simplify notation, let

$$(3.3) \quad \begin{aligned} A &\triangleq D_j Y \\ \underline{b} &\triangleq \underline{U}_{n,j} \\ \underline{x} &\triangleq \underline{x}_n. \end{aligned}$$

That is, we need to solve

$$(3.4) \quad \min_{\underline{x}} \|A \underline{x} - \underline{b}\|^2,$$

which has the LS solution

$$(3.5) \quad \hat{\underline{x}} = (A^T A)^{-1} A^T \underline{b}.$$

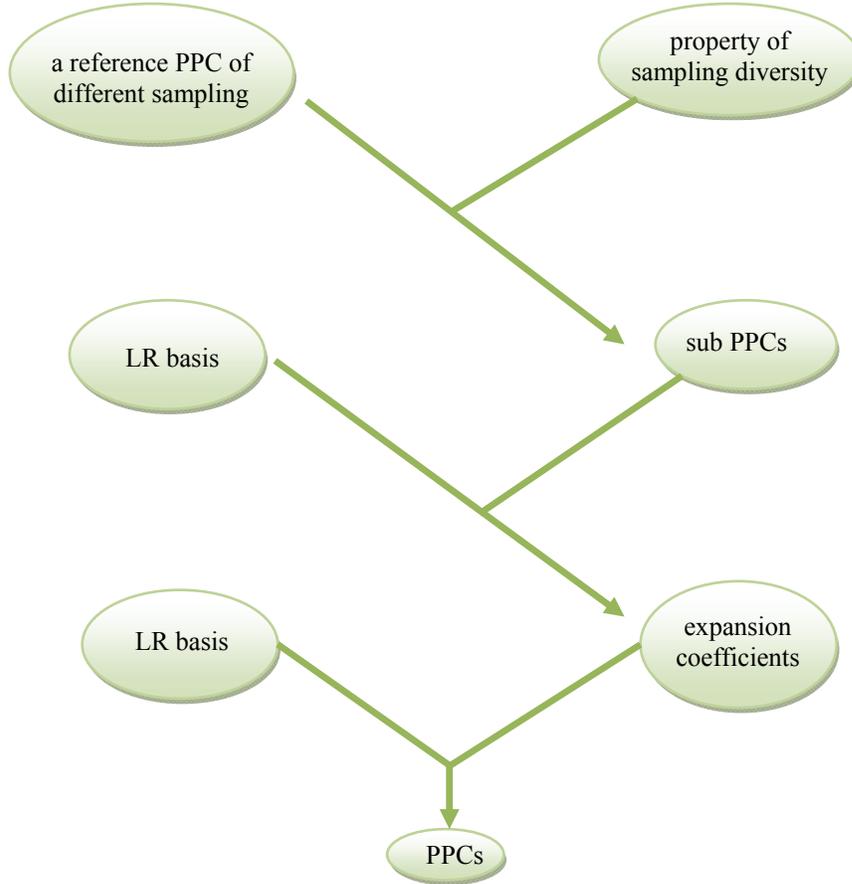


Figure 3.1: Reconstruction of the PPCs using the LR images as a basis.

Note that

$$\hat{\underline{x}} \sim \mathcal{N}\left(\underline{x}, \sigma_e^2 (A^T A)^{-1}\right),$$

and since it attains the Cramer-Rao lower bound (CRLB), then it is the minimum variance unbiased estimator (MVUE). Another way to prove this classical result is via the use of Gauss-Markov theorem which states that when the error model is linear (3.1) and the noise is zero-mean, uncorrelated and with the same variance, then the LS solution is the best (minimum variance) unbiased estimator (BLUE). If the noise is also assumed to

be Gaussian then the BLUE is the MVUE because a linear estimator requires only first and second order statistics and these are sufficient statistics in the Gaussian case [30].

Any ML estimator is asymptotically Gaussian¹, asymptotically unbiased, and asymptotically efficient, i.e. it attains the CRLB with more samples (larger LR images, see below). And with the assumptions made at the beginning of this section, the LS solution is the ML estimator and it is unbiased and efficient with Gaussian distribution (since it is a linear function of $\underline{b} = \underline{U}_{n,j}$), and it is unique when A has full column rank. Therefore, p , the size of the vector, \underline{b} must satisfy

$$(3.6) \quad p = M_1 M_2 / I^2 J^2 > K,$$

where K is the number of LR images. In other words, in order for the problem to be overdetermined, p , which is the number of the pixels in a *sub* PPC (which is the same number of pixels in sub LR images reordered as columns in the sub data matrix A), must be larger than the number of LR images. This means that the systems of equations we solve become more overdetermined by super-resolving larger LR images which can lead to an even lower CRLB bound to be asymptotically (or exactly, with our assumptions) attained by the ML estimator. For example, obtaining a HR image that is 4x4 times larger than LR images of size 200x200 can give a lower variance estimate, than does super-resolving (by the same factor of 4x4) smaller LR images of size 100x100. In short, it is preferable to super-resolve the HR image in its entirety rather than working on subregions of it. Of course, if the LR images are too large then we might need to super-resolve subregions of the HR image to lower memory requirements (and the computational cost).

Finally, we note that by the invariance property of the MLE,

$$\hat{\underline{U}}_n = Y\hat{\underline{x}}$$

is also the ML estimator of the n -th PPC, \underline{U}_n . It is also unbiased and efficient with Gaussian distribution

¹ Knowledge of the (asymptotic) distribution of an estimator is useful for purposes of statistical inference.

$$\hat{\underline{U}}_n \sim \mathcal{N}\left(\underline{U}_n, \sigma_e^2 Y (A^T A)^{-1} Y^T\right).$$

3.1.2 Regularized LS Solution

Given the fact that the columns of the data matrix, Y , are assumed to be ‘noise-free’ LR images, we would expect the data submatrix, A to be ill-conditioned. This is due to the fact that the LR images are highly correlated and thus columns of Y are hardly linearly independent. Also, if Y has singular values $\mu_1 \geq \dots \geq \mu_K$ and A has singular values $\sigma_1 \geq \dots \geq \sigma_K$ then by the interlacing theorem for singular values [26] we have

$$\mu_k \geq \sigma_k \text{ for } k=1, \dots, K,$$

and therefore if Y is ill-conditioned, then so is A . If that is the case, the solution (3.5) is numerically unstable. In order to see this, let $\{\underline{w}_k, \sigma_k, \underline{v}_k\}_{k=1}^K$ denote the singular triplets (left singular vectors, singular values and right singular vectors) of A , then equation (3.5) can be re-written as

$$\hat{\underline{x}} = \sum_{k=1}^K \left(\frac{\underline{w}_k^T \underline{b}}{\sigma_k} \right) \underline{v}_k.$$

Therefore, when the last few singular values are very small (A is ill-conditioned), the LS solution will be unstable, resulting in noise magnification as $1/\sigma_k \rightarrow \infty$ for the small singular values and the components of noisy \underline{b} in the direction of \underline{w}_k represent the most significant component of the solution in the direction of \underline{v}_k .

In other words, if A is (numerically) rank-deficient with rank $r < K$ then there exist an infinite number of solutions that minimize (3.4), for if \underline{x} is a minimizer and $\underline{x}' \in \text{null}(A)$ then $\underline{x} + \underline{x}'$ is also a minimizer. Of all these solutions, a minimal norm solution is usually preferred to control noise magnification which is synonymous with the non-uniqueness of the solution (the problem is said to be ill-posed). The *minimal norm* LS solution that avoids this problem is known as the truncated singular value decomposition (TSVD) and is given by [23]

$$\hat{\underline{x}}_{TSVD} = \sum_{k=1}^{r < K} \left(\frac{\underline{w}_k^T \underline{b}}{\sigma_k} \right) \underline{v}_k .$$

A most commonly used alternative to TSVD is the Tikhonov regularized LS solution which *smoothly* filters out the solution components corresponding to the smallest singular values [25]

$$\hat{\underline{x}}_{Tik} = \left(A^T A + \lambda^2 I \right)^{-1} A^T \underline{b} = \sum_{k=1}^K \left(\frac{\sigma_k}{\sigma_k^2 + \lambda^2} \underline{w}_k^T \underline{b} \right) \underline{v}_k ,$$

where λ is the regularization parameter. This is the solution to the minimization problem

$$\min_{\underline{x}} \|A\underline{x} - \underline{b}\|^2 + \lambda^2 \|\underline{x}\|^2 .$$

In Bayesian terms, the Tikhonov regularized LS solution is the maximum *a posteriori* (MAP) estimator of the expansion coefficients, \underline{x} , under the assumption that the expansion coefficients are uncorrelated with zero-mean and the same variance, $\sigma_x^2 = \sigma_e^2 / \lambda^2$, and are Gaussian distributed. In other words, it is the solution to the problem:

$$\begin{aligned} \max_{\underline{x}} p(\underline{b} | \underline{x}) p(\underline{x}) &= \frac{1}{(2\pi\sigma_e^2)^{p/2}} \exp\left(-\frac{1}{2\sigma_e^2} \|\underline{b} - A\underline{x}\|^2\right) \cdot \frac{1}{(2\pi\sigma_x^2)^{K/2}} \exp\left(-\frac{1}{2\sigma_x^2} \|\underline{x}\|^2\right) \\ &= \frac{1}{c} \exp\left\{-\frac{1}{2\sigma_e^2} \left(\|\underline{b} - A\underline{x}\|^2 + \frac{\sigma_e^2}{\sigma_x^2} \|\underline{x}\|^2 \right)\right\} . \end{aligned}$$

In addition, if we further assume that the expansion coefficients and the noise are independent, and thus \underline{x} and \underline{b} are jointly Gaussian ($p(\underline{b}, \underline{x})$ is Gaussian), then the Tikhonov regularized LS solution (with $\lambda^2 = \sigma_e^2 / \sigma_x^2$) is the minimum mean square error (MMSE) estimator (posterior mean) as well as the minimum absolute error (MAE) estimator (posterior median) [30].

Tikhonov regularized LS solution can also be viewed as a penalized likelihood estimator with a regularization term that penalizes the energy of the expansion

coefficients. Unlike Bayesian methods, penalized likelihood does not assume prior knowledge of the distribution of the parameters (expansion coefficients).

3.2 The TLS Solution

In the previous section, we made the assumption that the data matrix, Y , and thus the data submatrix A (3.3), are noiseless which is rarely ever the case. This means that the LS solution is not the ML estimator. Nevertheless, if we ignore the fact that A is noisy and apply the LS solution then we do not need any regularization as A is already well-conditioned (the smallest singular values will never be zero due to presence of noise). However, by opting to ignore the fact that A contains error then we will have a biased solution corresponding to the projection of \underline{b} on the wrong space (columns of A are noisy).

The total least squares (TLS) generalizes the original least squares solution by accounting for presence of noise in A . Specifically, the LS solution, which minimizes $\|A\underline{x} - \underline{b}\|^2$, is equivalent to solving the problem

$$\min_{\hat{\underline{b}}, \underline{x}} \left\| \hat{\underline{b}} - \underline{b} \right\|^2 \text{ subject to } A\underline{x} = \hat{\underline{b}}.$$

That is, $\hat{\underline{b}}$ is the smallest possible perturbation of \underline{b} which lies in the range of A . In other words, we perturb \underline{b} just enough to ensure that the perturbed equation has a solution, and then solve this system of equations. Now, if A is also subject to noise, then why not perturb A as well as \underline{b} ? That is, seek \hat{A} and $\hat{\underline{b}}$ such that² $\left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{\underline{b}}] \right\|_F^2$ is as small as possible subject to $\hat{\underline{b}} \in R(\hat{A})$. Then $\hat{A}\underline{x} = \hat{\underline{b}}$ has a solution, and any such solution is the TLS solution to the problem $A\underline{x} \approx \underline{b}$ [23].

Now, let the two equations below denote the *reduced*³ singular value decomposition (SVD) of A and the augmented matrix $[A \mid \underline{b}]$, respectively

² The notation $\left\| \right\|_F^2$ denotes the squared Frobenius norm of a matrix, which is the sum of the square of all its elements.

³ For a matrix $A \in \mathbb{R}^{p \times K}$ with $p > K$, it is sufficient (and more economical) to compute the left singular vectors corresponding to the non-zero singular values only.

$$(3.7) \quad A = W\Sigma V^T = \sum_{k=1}^K \sigma_k \underline{w}_k \underline{v}_k^T,$$

$$(3.8) \quad [A \mid \underline{b}] = \tilde{W}\tilde{\Sigma}\tilde{V}^T = \sum_{k=1}^{K+1} \tilde{\sigma}_k \tilde{\underline{w}}_k \tilde{\underline{v}}_k^T.$$

As discussed above, we seek to find a solution to the constrained minimization problem

$$(3.9) \quad \min_{\hat{A}, \hat{\underline{b}}, \underline{x}} \left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{\underline{b}}] \right\|_F^2 \quad \text{subject to} \quad \hat{A}\underline{x} = \hat{\underline{b}}.$$

The TLS problem (3.9) is non-convex. Nevertheless, an analytical solution does exist. We start by rewriting $A\underline{x} \approx \underline{b}$ as

$$[A \mid \underline{b}][\underline{x}^T, -1]^T \approx \underline{0}.$$

If $[A \mid \underline{b}]$ has full rank $K+1$ ($\tilde{\sigma}_{K+1} \neq 0$) then the best rank- K approximation $[\hat{A} \mid \hat{\underline{b}}]$ of $[A \mid \underline{b}]$ in the Frobenius norm sense is given by

$$(3.10) \quad [\hat{A} \mid \hat{\underline{b}}] = \sum_{k=1}^K \tilde{\sigma}_k \tilde{\underline{w}}_k \tilde{\underline{v}}_k^T,$$

and (3.9) is solved by solving

$$(3.11) \quad [\hat{A} \mid \hat{\underline{b}}][\hat{\underline{x}}^T, -1]^T = \underline{0}.$$

Therefore

$$\begin{aligned} [\hat{\underline{x}}^T, -1]^T &= \frac{-1}{\tilde{V}(K+1, K+1)} \tilde{\underline{v}}_{K+1} \\ \Rightarrow \hat{\underline{x}} &= \frac{-1}{\tilde{V}(K+1, K+1)} [\tilde{V}(1, K+1) \quad \dots \quad \tilde{V}(K, K+1)]^T, \end{aligned}$$

where \tilde{V} is the right singular matrix of the augmented matrix (3.8).

Statistical Properties of the TLS Solution

When the errors in the observations $[A \mid \underline{b}]$ are zero-mean, independent and identically distributed (i.i.d), the TLS is a strongly consistent, asymptotically unbiased and asymptotically Gaussian distributed estimator. If, in addition, the distribution of errors is Gaussian, then the TLS is the ML estimator (and thus it is asymptotically efficient, as well). In fact, regardless of distribution of errors, the TLS is at least weakly consistent, if the errors are zero mean, uncorrelated and with the same variance. On the other hand, the LS is asymptotically biased (and thus inconsistent). Nonetheless, the total variance of TLS is larger than that of LS. For more details on the statistical properties of the LS and TLS solutions refer to [23, 33].

Essentially, the advantage of the TLS solution over the LS solution is that as we increase the *overdeterminedness* of the systems of equations we solve, its bias becomes much lower compared to that of the LS solution. This is especially manifest at high levels of noise. In our case, this means that as we super-resolve *larger* LR images⁴, the TLS solution would be noticeably less biased than the LS solution, when working at relatively low signal-to-noise ratio (SNR).

Numerical Instability of the TLS Solution

A potential problem with the TLS solution, in our case, is due to the fact that the LR images are highly correlated causing the gaps between the last few singular values of $[A \mid \underline{b}]$ to be very narrow⁵. This means the solution of the TLS problem (3.11) is not unique. This is because when

$$\tilde{\sigma}_{K+1} \approx \dots \approx \tilde{\sigma}_{\ell+1} < \tilde{\sigma}_{\ell} \quad (\ell < K),$$

⁴ While, in terms of bias, the LS solution does not benefit much, especially at higher levels of noise, from increasing the overdeterminedness of the systems of equations (super-resolving *larger* LR images), its bias is significantly reduced by increasing the *number* of LR images. Indeed, adding noise to a *complete* basis renders it incomplete, and this is precisely why the solution becomes more biased with higher noise levels in the data matrix. In other words, adding noise to the available LR images makes their number effectively lower. See the beginning of §2.2 and also the end of §2.2.1 regarding using LR images as a basis set.

⁵ The smallest singular values correspond mostly to noise, and in the case of same variance, *uncorrelated* noise, they tend to be equal in size.

then *any* linear combination of $\tilde{\underline{v}}_{K+1}, \tilde{\underline{v}}_K, \dots, \tilde{\underline{v}}_{\ell+1}$ solves the TLS problem provided it results in a vector of the form $[\hat{\underline{x}}^T, -1]^T$ [23, section 3.3.1].

Another problem is that noise magnification is associated with the non-uniqueness of the TLS solution. In fact, it can be easily proven [23] that the TLS solution has the closed-form

$$(3.12) \quad \hat{\underline{x}} = \left(A^T A - \tilde{\sigma}_{K+1}^2 I \right)^{-1} A^T \underline{b}.$$

We review the simple proof here for convenience. First note that

$$[A \mid \underline{b}]^T [A \mid \underline{b}] [\hat{\underline{x}}^T, -1]^T = \tilde{\sigma}_{K+1}^2 [\hat{\underline{x}}^T, -1]^T,$$

also,

$$[A \mid \underline{b}]^T [A \mid \underline{b}] [\hat{\underline{x}}^T, -1]^T = \begin{bmatrix} A^T A & A^T \underline{b} \\ \underline{b}^T A & \underline{b}^T \underline{b} \end{bmatrix} [\hat{\underline{x}}^T, -1]^T.$$

Equating the top row of the right-hand-side of the last two equations we get

$$A^T A \hat{\underline{x}} - A^T \underline{b} = \tilde{\sigma}_{K+1}^2 \hat{\underline{x}},$$

which gives (3.12). Now, the interlacing theorem [26] implies that

$$\tilde{\sigma}_1 \geq \sigma_1 \geq \dots \geq \tilde{\sigma}_K \geq \sigma_K \geq \tilde{\sigma}_{K+1},$$

and realizing that the matrix $A^T A - \tilde{\sigma}_{K+1}^2 I$ has singular values $\{\sigma_k^2 - \tilde{\sigma}_{K+1}^2\}_{k=1}^K$, we notice that the TLS solution can be numerically unstable when the smallest singular values of $[A \mid \underline{b}]$ are close to each other. In fact, TLS can be seen as an attempt to reverse the process that made A and \underline{b} noisy, and compared to LS, it can be viewed as a de-regularization procedure [33].

3.2.1 Tikhonov Regularized TLS

The simplest solution to control noise magnification, due to non-uniqueness, is regularization by truncated total least squares (TTLS) [24]. Another alternative is to pick

the solution with the minimum norm, i.e. Tikhonov regularize the TLS solution (TRTLS). First note that problem (3.9) is equivalent to⁶

$$(3.13) \quad \min_{\hat{A}, \hat{\underline{x}}} \left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{\underline{x}}] \right\|_F^2.$$

The TRTLS problem is

$$(3.14) \quad \min_{\hat{A}, \hat{\underline{x}}} \left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{\underline{x}}] \right\|_F^2 + \lambda^2 \|\underline{x}\|^2.$$

Using Lagrange multiplier formulation [31], the authors in [27] proved that (3.14) has the solution

$$(3.15) \quad \hat{\underline{x}}_{TRTLS} = \left(A^T A + (\lambda^2 - \tilde{\sigma}_{K+1}^2) I \right)^{-1} A^T \underline{b}.$$

Note that for $\lambda^2 = \tilde{\sigma}_{K+1}^2$, we get the LS solution. In our case, A is rarely ill-conditioned because it is a submatrix of the data matrix Y which is always contaminated with noise. This precludes the need for increasing the regularization parameter beyond $\tilde{\sigma}_{K+1}^2$. In fact, the notion that a certain amount of error in the coefficient matrix might actually be beneficial is discussed, even within the context of super-resolution, in [28]. Therefore our choice of the regularization parameter should lie within

$$0 < \lambda^2 < \tilde{\sigma}_{K+1}^2,$$

where the lower limit achieves the TLS solution while the upper limit gives us the LS solution.

3.2.2 L1-Regularized TLS

The idea of using the L1-norm to penalize the least squares solution was first presented in the context of Linear Regression [29] under the name *Least Absolute Selection and Shrinkage Operator* (LASSO). The use of the L1-norm was motivated by the desire to get rid of irrelevant features for easier interpretability. An L1-norm penalty function has the property of concentrating on minimizing small residuals as opposed to large ones. Therefore, when the residuals are the elements of \underline{x} , this gives us a sparse set

⁶ One way to prove the result (3.12) is using Lagrange multipliers for (3.13). See [27].

of expansion coefficients. This is in contrast to the L2-norm penalty (Tikhonov) which forces the coefficients to be rather more similar to each other.

Typically, L1-norm minimization is used for robustness against outliers. In addition to noise, outliers represent an important source of error. For our problem, outliers are irrelevant LR images⁷ reordered as columns in the data matrix. Ideally, the expansion coefficient corresponding to an outlier LR image should be zero. Fortunately, as our problem is typically highly overdetermined (3.6), outliers, if present, should not affect the solution. Now, \underline{x} being the expansion coefficients in terms of the set of LR images, adding an L1 penalty nonlinearly denoises the solution, partly by *shrinking* it and partly by *discarding* the least significant components. These small components likely correspond to noise so discarding them is desirable.

Adding an L1 regularization term to the data fitting term (3.13) we get

$$(3.16) \quad \min_{\hat{A}, \underline{x}} \left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{A}\underline{x}] \right\|_F^2 + \gamma \|\underline{x}\|_1.$$

Like (3.13, 3.14), problem (3.16) is non-convex. Unlike (3.13, 3.14), however, problem (3.16) does not happen to have an analytical solution. Consequently, we replace (3.16) with a convex surrogate problem. First note that (3.13) is equivalent to

$$\min_{\underline{x}} \left\| \hat{A}\underline{x} - \hat{\underline{b}} \right\|^2,$$

where \hat{A} and $\hat{\underline{b}}$ are as defined in (3.10). Now, consider the (convex) cost function

$$(3.17) \quad \min_{\underline{x}} \left\| \hat{A}\underline{x} - \hat{\underline{b}} \right\|^2 + \gamma \|\underline{x}\|_1,$$

and note that for $\gamma = 0$, we get the unregularized TLS solution while for $\gamma > 0$, we get what we refer to as the L1-norm regularized TLS solution. Of course, (3.17) is not equivalent to (3.16), and we do not know how well it approximates it. Nevertheless, according to all our simulations, for the same data fitting error, solving (3.17) gives better denoising performance compared to the TRTLS (3.14).

⁷ In our case, an outlier image is one that is either too distorted, too noisy or simply does not belong to the LR basis.

Problem (3.17) can be reformulated as

$$(3.18) \quad \min \|\underline{x}\|_1 \quad \text{subject to} \quad \|\hat{A}\underline{x} - \hat{\underline{b}}\|^2 < \varepsilon^2,$$

where $\varepsilon^2 = \|\hat{A}\hat{\underline{x}}_{TRTLS} - \hat{\underline{b}}\|^2$. This, of course, requires evaluating (3.15) which takes only a fraction of the time needed to solve (3.18). By solving (3.18) we find the L1-regularized TLS solution, to within the *same* error (data misfit) corresponding to the TRTLS solution. This is the easiest way to highlight the denoising performance of the L1-norm compared to the linear filtering effect of the L2-norm (Tikhonov) penalty.

A Note on Convex Optimization

Generally, for mathematical optimization problems, an *analytical* solution exists only when the optimization problem is unconstrained (or with affine equality constraints) with a quadratic objective function. These conditions are of course extremely limiting and one should try instead to formulate problems that are convex and seek a numerical solution. In particular, if the problem can be recast as a linear programming (LP), quadratic programming (QP), second order cone programming (SOCP) or semidefinite programming (SDP), then it is considered essentially solved. Efficient solvers are freely and commercially available for these types of problems. Problem (3.18) can be recast as SOCP using epigraph form⁸ [31]

$$\min_{\underline{t}, \underline{x}} \quad \underline{1}^T \underline{t} \quad \text{subject to} \quad \|\hat{A}\underline{x} - \hat{\underline{b}}\|^2 < \varepsilon^2, \quad -\underline{t} < \underline{x} < \underline{t}.$$

Trimmed TLS

The closed-form TLS solution is given by (3.12) and it is equivalent to

$$\hat{\underline{x}} = \sum_{k=1}^K \left(\frac{\sigma_k}{\sigma_k^2 - \tilde{\sigma}_{K+1}^2} \underline{w}_k^T \underline{b} \right) \underline{v}_k.$$

⁸ We used the solver SDPT3 [60, 61], along with the interface CVX [58, 59], to obtain an *exact* solution to (3.18) reformulated in the SOCP form. Of course, for larger problems, iterative methods become essential.

Obviously, the last few components of the solution are responsible for the numerical instability and noise magnification associated with the TLS solution. It is therefore rather intuitive to simply discard the highest order components of the solution. This is not to be confused with truncated TLS (TTLS) where regularization is reached by finding the optimal linear combination of the last few right singular vectors of the *augmented* matrix, $[A \mid \underline{b}]$ [24]. This is also different from Tikhonov regularized TLS (TRTLS) in that, unlike TRTLS, the weights of the lower order components of the solution are not changed.

To the best of our knowledge, there is no reference in the literature to this type of regularization of the TLS solution. Also, it appears there is no easy way to assess the optimality of this method as the cost function it minimizes is unknown. The simulations, however, point to the superiority of trimmed TLS (better bias-variance tradeoff) compared to Tikhonov regularized TLS.

A Different Regularization Term?

Tikhonov regularized TLS solution should be appreciated at least for its simplicity and providing numerical stability. However, in Bayesian terms, using a minimum energy penalty entails the assumption that the expansion coefficients we solve for are a zero-mean, uncorrelated, with the same variance and jointly Gaussian distributed. On the other hand, using an L1-norm minimization corresponds to the assumption of a Laplacian distribution. Naturally, since the LR basis is highly correlated, the assumption that the expansion coefficients are uncorrelated is unrealistic. In addition, the assumption that the joint distribution of the expansion coefficients is Gaussian (or Laplacian) cannot be accurate but it is somewhat more acceptable compared to some other methods where a minimum energy penalty is used to stabilize the solution for the *pixels* of the HR image itself [12, 13].

Two popular regularization methods are based on the assumption that natural signals are smooth. These are the Markov random field (MRF) prior [52] and the total variation (TV) norm minimization. TV is commonly used as a regularizer for denoising/deblurring of images [53, 54]. It penalizes the total amount of change in the image as measured by the L1-norm of the magnitude of the gradient. In our case, however, what we solve for

are the *expansion coefficients*, hence using MRF or TV to regularize the solution is inappropriate. In addition, even if we reformulate the regularization to be a function of the PPC, for example,

$$\min_{\hat{A}, \underline{x}} \left\| [A \mid \underline{b}] - [\hat{A} \mid \hat{A}\underline{x}] \right\|_F^2 + \lambda \Gamma(Y\underline{x}),$$

where $\Gamma(Y\underline{x})$ is the regularization term, and even if we could solve this non-convex problem exactly, it is counter-intuitive to try to penalize the roughness of a non-smooth signal. In particular, polyphase components are expected to be rough, since they contain large high frequency components due to aliasing. Moreover, in §3.3, it becomes evident that a part of the variance of error of an estimated PPC is independent of the bias-variance tradeoff provided by any penalty term, and therefore formulating the penalty term as a function of the PPC should be avoided.

In §3.4, we propose using principle component analysis (PCA) to *optimally*⁹ pre-denoise the data, which is an essential pre-processing step when the noise is relatively high as shall be seen in §3.3. This pre-processing of the data also reduces the bias, and renders the TLS solution, and the search for an optimal regularization thereof, superfluous.

3.3 Mean and Covariance of an Estimated PPC

In this section we show that an estimated PPC will always be *noisier* than the LR images, even if the estimated expansion coefficients have zero variance.

First, we assume that the data matrix is corrupted with additive noise,

$$Y = Y_0 + \mathcal{V},$$

where Y_0 is the noise-free data matrix (the signal component of the data) and \mathcal{V} is a noise matrix with entries that are uncorrelated, zero-mean and with the same variance σ_v^2 .

Let $\underline{\mu}_w$ and R_w denote the mean and covariance, respectively, of the error, \underline{w} , in the estimated expansion coefficients, $\hat{\underline{x}} = \underline{x} + \underline{w}$, where \underline{x} is the error-free expansion coefficients. For tractability, we further assume that \mathcal{V} and \underline{w} are independent.

⁹As a denoiser, PCA is optimal when the noise's covariance matrix is a scaled identity matrix and the covariance matrix of the data is known.

The corresponding estimated n -th PPC component is thus,

$$\begin{aligned}\hat{U}_n &= Y\hat{x} \\ &= Y_0\underline{x} + Y_0\underline{w} + \mathcal{V}\hat{x}.\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbf{E}[\hat{U}_n] &= Y_0\underline{x} + Y_0\underline{\mu}_w \\ \hat{U}_n - \mathbf{E}[\hat{U}_n] &= Y_0(\underline{w} - \underline{\mu}_w) + \mathcal{V}\hat{x},\end{aligned}$$

where $\mathbf{E}[\cdot]$ denotes the expectation operator.

The covariance matrix of error is

$$\begin{aligned}\text{Cov}(\hat{U}_n - U_n) &= \text{Cov}(\hat{U}_n) \\ &= \mathbf{E}\left[(\hat{U}_n - \mathbf{E}[\hat{U}_n])(\hat{U}_n - \mathbf{E}[\hat{U}_n])^T\right] \\ &= Y_0 R_w Y_0^T + \mathbf{E}[\mathcal{V}\hat{x}(\mathcal{V}\hat{x})^T] + 2\mathbf{E}\left[Y_0(\underline{w} - \underline{\mu}_w)(\mathcal{V}\hat{x})^T\right].\end{aligned}$$

It can be easily verified that

$$\begin{aligned}\mathbf{E}[\mathcal{V}\hat{x}(\mathcal{V}\hat{x})^T] &= \mathbf{E}[\hat{x}^T \hat{x}] \sigma_v^2 I_d \\ &= \sigma_v^2 \left(\|\underline{x}\|^2 + \|\underline{\mu}_w\|^2 + \text{Tr}(R_w) + 2\underline{\mu}_w^T \underline{x} \right) I_d,\end{aligned}$$

and

$$\mathbf{E}\left[Y_0(\underline{w} - \underline{\mu}_w)(\mathcal{V}\hat{x})^T\right] = [0],$$

where $\text{Tr}(\cdot)$ denotes the trace of a matrix, I_d is the identity matrix of size $d \times d$ and $d = M_1 M_2 / I^2$ is the number of pixels in a PPC. The covariance matrix of error is thus

$$\begin{aligned}\text{Cov}(\hat{U}_n) &= Y_0 R_w Y_0^T + \mathbf{E}[\hat{x}^T \hat{x}] \sigma_v^2 I_d \\ &= Y_0 R_w Y_0^T + \sigma_v^2 \left(\|\underline{x}\|^2 + \|\underline{\mu}_w\|^2 + \text{Tr}(R_w) + 2\underline{\mu}_w^T \underline{x} \right) I_d,\end{aligned}\tag{3.19}$$

and the mean square error (MSE) of $\hat{\underline{U}}_n$ is given by

$$\begin{aligned}
 \text{MSE}(\hat{\underline{U}}_n) &= \text{Total variance}(\hat{\underline{U}}_n) + \|\text{Bias}(\hat{\underline{U}}_n)\|^2 \\
 (3.20) \quad &= \text{Tr}(Y_o R_w Y_o^T) + d\sigma_v^2 \left(\|\underline{x}\|^2 + \|\underline{\mu}_w\|^2 + \text{Tr}(R_w) + 2\underline{\mu}_w^T \underline{x} \right) + \|Y_o \underline{\mu}_w\|^2
 \end{aligned}$$

Equation (3.19) tells us that even if we knew the error-free expansion coefficients, \underline{x} , in terms of the noiseless version of the data matrix, Y_o , then a reconstructed PPC will be noisier than a LR image by a factor of $\|\underline{x}\|^2$. In other words, if we could somehow obtain a perfect estimate of the expansion coefficients, the covariance of error will be

$$(3.21) \quad \text{Cov}(\hat{\underline{U}}_n) = \|\underline{x}\|^2 \sigma_v^2 I_d.$$

Consequently, it is obvious that even in the absence of error in estimating the expansion coefficients, pre-denoising of the data matrix (§3.4) or post-denoising of the reconstructed HR image (§3.6), or both, is a necessity when the noise in the data matrix is moderately high. Also, equation (3.21) reveals that $\hat{\underline{U}}_n$ is inconsistent, regardless of the estimation of the expansion coefficients, and therefore, given that the expansion coefficients are known, the only way to benefit from an increased overdeterminedness of the problem (super-resolving *larger* LR images) is if the pre-denoiser of the data does benefit from super-resolving large LR images. As will be explained in the next section, PCA denoising, which denoises by maximizing the SNR of the low order principal components and discarding the ones with small SNR, performs better, at least theoretically, when dealing with larger LR images.

The MSE formula (3.20) contains three error parameters:

- σ_v^2 which is, as defined previously, the variance of noise in LR images.
- R_w , the covariance matrix of the estimated expansion coefficients, is dependent on the amount of noise in the estimated *reference* PPC and the bias-variance tradeoff, if any, of the estimation (regularization).
- Assuming the noiseless version of the LR images (the signal part of the data matrix) spans the PPCs, the bias of the estimated expansion coefficients, $\underline{\mu}_w$, is dependent on the noise level in the sub data matrix A (i.e. σ_v^2), the bias caused by

regularization (if any), and the bias of the estimated reference PPC. According to experiments, at moderate values of σ_v^2 (e.g. at 30dB SNR), the bias due to noisy A is normally marginal, even using the LS estimator.

Although, it might not be easily discernible from examining equation (3.20), according to our experiments, the bias of $\hat{\underline{U}}_n$ can *overshadow* its variance (the reconstructed HR image appears much less noisy when it is blurred or aliased). As mentioned above, this can only be *partly* owing to the bias-variance *tradeoff* associated with estimating the expansion coefficients (regularization). In other words, a blurred reference PPC has the advantage of submerging the noisy appearance of the reconstructed HR image. However, the best way to control the *enhanced noise manifestation* (3.21) is to directly control the effect of the parameter σ_v^2 (3.19 - 3.21) by pre-denoising the data matrix. This, incidentally, also strips the TLS of its advantage of low bias compared to the LS solution, even at relatively low SNRs.

3.4 Pre-Denoising the LR Images using PCA

In light of the last two sections, the goal of pre-denoising the LR images is clear: reducing the noise enhancement effect associated with multiplying the LR images with the expansion coefficients and obtaining less biased estimates of the expansion coefficients.

Using first and second order statistics of a data set, principal component analysis (PCA) provides an orthonormal optimal basis (in the mean squared error (MSE) sense) for a reduced representation of the data [32], where the first few principal axes can capture, on average, a significant portion of a data point's energy while the last few principle axes correspond mainly to *insignificant features*. In other words, it is the optimal *linear* minimum MSE (MMSE) *compressor* of the data, regardless of the distribution¹⁰. This property of the PCA makes it also the optimal *linear* denoiser when the data is contaminated with additive zero-mean, same variance, uncorrelated noise. Specifically, if we assume that the *noisy* LR images are realizations of a random vector

¹⁰ If the mean and covariance matrix are known, the distribution of the data is irrelevant to the performance of PCA as a *linear* MMSE compressor.

$$\underline{y} = \underline{y}_0 + \underline{v},$$

where \underline{v} is a zero-mean noise vector with covariance matrix, $\sigma_v^2 I_d$, and is statistically independent of \underline{y}_0 , which is the underlying random vector generating the *noiseless* part of the LR images (the signal part) with mean, $\underline{\mu}$, and covariance matrix, C , with eigen-decomposition

$$C = E\Lambda E^T,$$

where the columns of E are the orthonormal eigenvectors of C , and the diagonal matrix, Λ , contains eigenvalues $\lambda_1 > \lambda_2 > \dots > \lambda_d$, then the covariance matrix of the random vector \underline{y} , is

$$\begin{aligned} C_y &= C + \sigma_v^2 I_d \\ &= E\tilde{\Lambda}E^T, \end{aligned}$$

where

$$\tilde{\Lambda} = \Lambda + \sigma_v^2 I_d.$$

The PCA basis vectors (the principal axes) are the columns of E , and the transformation

$$\underline{z}_k = E^T \bar{y}_k$$

where \bar{y}_k is the k -th centered LR image, decorrelates the centered LR image and maximizes the variance of the lower order principal components (expansion coefficients in terms of the PCA basis) of the k -th centered LR image:

$$\begin{aligned} \mathbf{E}[\underline{z}_k \underline{z}_k^T] &= \mathbf{E}[E^T \bar{y}_k \bar{y}_k^T E] \\ &= E^T C_y E \\ &= \tilde{\Lambda}. \end{aligned}$$

Noting that the principal components (PCs), i.e. the elements of the *feature* vector, \underline{z}_k , have variances

$$(3.22) \quad \lambda_\ell + \sigma_v^2 \quad \text{for } \ell = 1, \dots, d,$$

it becomes evident that the PCA also maximizes the SNR along the low order principal axes. Consequently, if we replace the q highest order PCs (the last q elements of \underline{z}_k) with zeros, resulting in the vector, $\hat{\underline{z}}_k$, the (reconstruction) MSE,

$$\mathbf{E} \left[(\underline{z}_k - \hat{\underline{z}}_k)^T (\underline{z}_k - \hat{\underline{z}}_k) \right] = \sum_{\ell=d-q+1}^d (\lambda_\ell + \sigma_v^2),$$

would correspond mostly to noise. Therefore, we can denoise the LR images by centering them, PCA transforming them and then discarding the high order PCs, or we could simply *retain* only the low order principal axes (corresponding to the largest eigenvalues) and use them for denoising:

$$\hat{\underline{y}}_k = E_r E_r^T (\underline{y}_k - \underline{\mu}) + \underline{\mu},$$

where E_r is the *reduced* PCA basis, and $\hat{\underline{y}}_k$ is the denoised k -th LR image.

3.4.1 The Sample Mean and the Sample Covariance Matrix

Since we have no knowledge of the true mean and true covariance matrix, we can only empirically estimate them from the data. The most commonly used estimators are the sample mean and the sample covariance matrix, which are unbiased under the assumption that the observations are i.i.d. If the data is also Gaussian distributed, the sample mean and (a slightly differently scaled) sample covariance matrix are also the ML estimates of the true mean and the true covariance matrix, respectively. The assumption of independence of observations is unrealistic. Moreover, the distribution of the data is hardly Gaussian and thus taking the eigenvectors of the sample covariance matrix as our PCA basis is not optimal (the *empirically* derived PCA basis is not the linear MMSE compressor, and thus it cannot be the optimal linear denoiser). For the scope of this thesis, however, the sample mean and sample covariance shall suffice.

In our problem, the number of observations (LR images) is far smaller than their dimensionality¹¹. Under such circumstances, the sample covariance matrix provides a poor estimate. A better strategy is to denoise *sub* LR images. This, not only reduces the

¹¹ Typically, the number of LR images is less than 1% of the number of variables (pixels within a LR image).

number of parameters to be estimated (smaller covariance matrix), but it also provides more samples (observations), allowing for a larger denoising space, where it is possible to discard a lot more high order PCs¹². In particular, we use both the primary and secondary LR images¹³ (corresponding to the primary and secondary sensors, respectively) and downsample them by $J \times J$ and $I \times I$, respectively, obtaining $KJ^2 + K^S I^2$ highly correlated *sub* LR images of the same size, where K^S is the number of secondary LR images. From these sub LR images we compute the sample mean and sample covariance, and then PCA denoise them using the eigenvectors of the empirically estimated covariance matrix. The sample mean of the sub LR images is given by

$$\underline{\hat{\mu}} = \frac{1}{KJ^2 + K^S I^2} \left(\sum_{k=1}^{KJ^2 + K^S I^2} \underline{y}_k^{sub} \right),$$

where \underline{y}_k^{sub} is the k -th sub LR image, reordered as a column vector. The sample covariance is defined as

$$\hat{C}_y = \frac{1}{KJ^2 + K^S I^2 - 1} \left(\sum_{k=1}^{KJ^2 + K^S I^2} (\underline{y}_k^{sub} - \underline{\hat{\mu}})(\underline{y}_k^{sub} - \underline{\hat{\mu}})^T \right) \in \mathbb{R}^{p \times p}.$$

Now, let D denote the matrix of the orthonormal eigenvectors of \hat{C}_y , corresponding to the largest r_0 eigenvalues¹⁴. D is, therefore, the reduced PCA matrix which we use to denoise the sub LR images as follows

$$(3.23) \quad \underline{\hat{y}}_k^{sub} = DD^T (\underline{y}_k^{sub} - \underline{\hat{\mu}}) + \underline{\hat{\mu}},$$

where $\underline{\hat{y}}_k^{sub}$ is the denoised k -th sub LR image.

Now we list the reasons for our choice of the *sub* LR images to be obtained by downsampling the primary and secondary LR sets by $J \times J$ and $I \times I$, respectively:

¹² The more computationally expensive Kernel PCA (nonlinear PCA), is known in the literature to be a much more superior denoiser than the empirically derived linear PCA when the number of samples *far* exceeds their dimensionality [55]. However, this is not applicable in our case.

¹³ The primary LR images are normalized to have the same L2-norm, and the secondary LR images are normalized to have the L2-norm of a primary LR image scaled by I/J . This step is useful to ensure that no single LR image can dominate the analysis.

¹⁴ According to synthetic and real data experiments, at $r_0 = 0.3 p$, there is virtually no loss of detail associated with denoising. In fact, even at $r_0 = 0.1 p$ there is slightly noticeable loss of detail. The default value we use for r_0 is $0.2 p$.

1. By denoising the sub LR images as described above, we also *directly* denoise the sub data matrices used for estimating the expansion coefficients.
2. The reason for choosing sub LR images to be downsampled versions of the LR images, rather than subregions of them, is that subregions across the LR images are *not as highly correlated* and thus more PCs would need to be retained to avoid significant loss of detail, which translates to less denoising capability.
3. Of course, to lower the computational¹⁵ cost of finding the eigen-decomposition (or SVD) of the sample covariance matrix, we could use even smaller sub LR images by downsampling further. This also makes the corresponding sample covariance matrix a better estimate since even more samples will be used to compute it. But on the other hand, the *denoising space will get smaller* (a smaller covariance matrix means a smaller number of eigenvectors, hence fewer can be discarded). Moreover, this will result in *smaller SNR along the lower order axes* since, theoretically, the noise level is *constant* along all axes (3.22) and of course it *does not* get lower when dealing with smaller sub LR images, while the signal's variance is maximized along the low order axes *and* is proportional to its total energy. Hence, working with smaller sub LR images results in smaller denoising space *and* lower SNR in the retained PCs. We digress slightly here to note that PCA denoising, at least theoretically¹⁶, can circumvent the inconsistency of the PPC estimator (3.21), since working with larger LR images translates to a larger denoising space and higher SNR in the retained PCs. Practically, however, and given that the number of LR images is fixed, working with larger LR images means that the sample covariance matrix estimate of the true covariance matrix of the sub LR images becomes poorer, not to mention the higher computational cost of finding the eigen-decomposition of the increased size sample covariance matrix (although when the number of samples is smaller than their dimensionality, the computational cost is primarily determined by the number of samples where the reduced SVD of the matrix of samples, rather than the covariance matrix itself, is computed [32], which

¹⁵ For faster computation of the first r_0 singular vectors of the covariance matrix, we use the Matlab code prepared by Mark Tygert, which is an implementation of the algorithm described in [62].

¹⁶ Assuming the true covariance matrix is known.

is expected to be the case if the problem involves super-resolving larger LR images, given that the number of LR images is fixed).

4. Finally, as will be explained in chapter IV, the same reduced PCA matrix D (3.23) will be also used in estimating the reference PPC, saving us the trouble of calculating the eigen-decomposition of another covariance matrix.

3.4.2 Outlier LR Images and their Effect on Denoising

Outlier LR images are those images irrelevant to the reconstruction of the PPCs. Since we use the LR images as basis signals, given that the estimated reference PPC does not have any components corresponding to outliers, the expansion coefficients in terms of the outlier images should be exactly zero, and thus outliers should be of no concern to us. However, since we pre-denoise the LR images using PCA, which is dependent on the sample covariance matrix, the presence of outliers in the samples will make high order PCs more representative of the signal's energy [32] and thus we will have to retain more PCs or risk significant loss of detail. Of course, more PCs to be retained means more noise too and therefore getting rid of outlier LR images becomes essential *for better denoising*.

Depending on the application, there is more than one suitable method for detection and removal of outliers in the data. For example, *trimming* the data involves finding the Mahalanobis distance of each LR image from the mean, and iteratively calculating a new covariance matrix (and mean) [56]. Of course, the Mahalanobis distance involves finding the inverse of the sample covariance matrix of the LR images, which is decidedly singular since number of LR images is far much lower than their dimensionality. Alternatively, and since our goal is to find a robust estimation of the covariance matrix of the *sub* LR images, we could implement the minimum covariance determinant (MCD) method. It works by finding the subset of samples whose covariance matrix has the lowest determinant [57]. However, and regardless of the computational cost, this method requires that the number of samples be *much* higher than their dimensionality which is hardly the case in our problem, even when the samples are *sub* LR images.

Fortunately, while our problem is short on samples (relative to their dimensionality), it is advantaged by the fact that the LR images are *highly correlated*. Therefore, outliers can

be defined as those images that are farthest from the mean. In order to identify outliers in the *secondary* LR set, the mean of the primary LR set is lowered in size (via nearest neighbor interpolation) to the same size of a secondary LR image, and outlier secondary LR images are thus those that are farthest from the resized mean. There are two reasons we did not use the mean of the secondary LR set to identify outliers within this set:

- The (same size) sub LR images, from both sets, are assumed to have the same mean and same covariance matrix and therefore, using two means to identify the outliers to the computation of the sample mean and sample covariance is meaningless.
- Ultimately, the secondary set of LR images is there only so we can estimate the reference PPC in order to compute the expansion coefficients of the primary PPCs in terms of the primary LR images. As a result, the relevance of an estimated reference PPC, and by extension the secondary LR images used to construct it, is determined by the available primary LR set. Namely, the ‘outlyingness’ of a secondary LR image can only be measured in terms of the ensemble of the primary LR set.

Clearly, this simple method of rejecting outlier images assumes that the *number* of outliers in the primary and secondary sets of LR images is already known. In chapter IV, where the estimation of a reference PPC is highly affected by the presence of outliers, we describe a simple intuitive way to obtain an approximate estimate of the number of outliers.

3.5 Color Images

The typical approach to processing color images is to simply super-resolve each of the three color-band images *separately* (thus *tripling* the computational cost) while ignoring the color artifacts present in the *demosaiced*¹⁷ LR images [46, 47]. Although none of the authors of [35-37], who addressed the problem of single-frame super-resolution using subspace learning methods, explained how they dealt with the case of color, we believe they too ignored the color artifacts and assumed that the LR images are captured by a 3-

¹⁷ Single CCD color cameras use the Bayer (color) filter to obtain all 3 color band images using one CCD sensor, where each pixel senses only one of the 3 colors, according to the Bayer pattern, and then the three raw color band images are demosaiced to interpolate the missing pixels. This results in color artifacts that are normally negligible at high resolutions but easily noticeable in LR images.

CCD¹⁸ camera (one sensor per color-band), where there would be no color artifacts at all. On the other hand, Farsiu et al. [48] considered joint *demosaicing* and super-resolution of color images to reduce the color artifacts associated with single CCD color cameras.

In our case, we also assume that the *primary* set of LR images is obtained by 3 primary CCD sensors. For the *secondary* set of *lower resolution* LR images, *only one* sensor for the green (luminance) band¹⁹ is required since we need to estimate the set of expansion coefficients only once. Recall that a LR image is assumed to be a linear mixing of the PPCs, and since each one of the three HR color-band images, undergoes the *same* transform resulting in the corresponding LR color-band image within the *same* LR frame, the same set of expansion coefficients can be used to un-mix the PPCs of each HR color-band image. In other words, if we let X denote the matrix containing all the expansion coefficients computed using only the green primary and green secondary LR images, then

$$\begin{aligned} U^R &= Y^R X \\ U^G &= Y^G X \\ U^B &= Y^B X, \end{aligned}$$

where Y^R , Y^G and Y^B are the red, green and blue data matrices, containing the unwrapped by column K red, K green and K blue LR images, respectively, and U^R , U^G and U^B are the red, green and blue image matrices containing the I^2 red, I^2 green and I^2 blue PPCs, respectively.

Although we are using only the green primary and secondary LR images to estimate the expansion coefficients, we might still want to pre-denoise the primary red and blue LR images since multiplying noisy LR images with the expansion coefficients enhances the noise (3.21) as we explained in §3.3. Of course, in this case, the sample covariance matrix will be derived from the primary red and blue LR sets only, as there are no secondary sets of red and blue LR images (we require only one lower resolution green sensor for the secondary set of LR images).

¹⁸ A beam splitter is used to split the image into its red, green and blue components to be separately detected on 3 CCD sensors.

¹⁹ The green (luminance) band of a color image is approximately equivalent to its grayscale version.

3.6 Post-Processing the SR Image

TV Denoising

Post-denoising the super-resolved image is an option to reduce the noise further when the PCA pre-denoising, on its own, is not sufficient.

Total variation (TV) is a well-known edge-preserving denoising method. The denoiser solves the minimization [53]

$$(3.24) \quad \min_{u_d} \|\nabla u_d\|_1 + \frac{\lambda}{2} \|u_d - u\|^2$$

where u_d is the denoised version of the original image, u , and λ is the parameter that controls the fidelity to data (the original noisy image). We use the code written by Pascal Getreuer which is an implementation of the algorithm described in [63] for iteratively solving the minimization problem (3.24). The code also handles color images by jointly denoising using the vectorial generalization of the TV, implementing the algorithm in [64] which is a generalization of the algorithm in [63].

Unsharp Masking

The super-resolved image can be blurred, mainly because the estimation of the reference PPC is biased to some degree (chapter IV). Also, as we shall explain in chapter V, the CCD sensor causes additional blur as well. Unsharp masking (UM) is a generic and a very simple sharpening technique [72]. In UM, a blurred version of the original image is subtracted from it and the result is scaled and then added to the original image. We use MATLAB's unsharp masking with default settings.

The Median Filter

After deblurring using the unsharp masking, the processed image usually contains what looks like impulsive noise around the edges. This could probably be due to the fact that we estimate the HR image by estimating its PPCs *separately* and then interlacing, which might cause some subtle irregularities in pixel intensity levels, especially around

the edges, that become more pronounced after sharpening. This problem is easily dealt with by using a simple 2×2 median filter.

3.7 Summary

In this chapter we examined the applicability of classical solutions to the problem of finding the expansion coefficients of the PPCs in terms of the LR basis, using knowledge of their *sub* PPCs. Specifically, under the assumption that the *sub* PPCs are contaminated with zero-mean, white Gaussian noise, the LS solution gives us a stable but biased solution because the LR images are normally noisy. The TLS solution takes into account the noise in the LR basis, but it is very unstable due to the high correlation between LR images. Penalizing the TLS solution using Tikhonov regularization, numerically stabilizes the solution, but it roughly translates to the unrealistic *a priori* assumption that the expansion coefficients are uncorrelated. Using a (surrogate) L1-norm regularization of the TLS solution, we obtained better results but with slower performance and without correcting for the unrealistic assumption of no correlation between the expansion coefficients. We also explained why popular regularization techniques such as MRF and TV cannot be applicable in our case. Moreover, in §3.3, it became evident that part of the error in a reconstructed PPC is independent of any penalty term that might be used to regularize the TLS.

Using PCA to pre-denoise the LR images, lowers the bias by reducing the noise in the sub data matrices and thus it revokes the TLS solution's advantage over the LS solution. Also, independently of any expansion coefficients' estimation error, multiplying the expansion coefficients with the data matrix, to estimate the PPCs, augments the noise. Therefore, PCA pre-denoising provides a remedy to this problem, as well.

The presence of outlier LR images can diminish PCA's denoising capability as it is based on the sample covariance matrix which is sensitive to outliers. Luckily, since the LR images are highly correlated, the outliers are easily identifiable as those images farthest from the mean.

For color images, other than pre-denoising each set of color-band LR images separately, our SR method estimates the color HR image at virtually no additional computational cost.

In practice, the estimated reference PPC is usually blurred, and it is therefore the main source of bias in the super-resolved image. Also, some additional edge-preserving denoising might be desired. For these reasons, we use TV denoising, followed by unsharp masking and median filtering.

3.8 Future Work

Different Types of Noise

The following list shows some of the errors, images captured by digital cameras are usually corrupted with.

- Camera sensor readout noise (zero-mean, white Gaussian, independent of signal). Cause: electronics.
- Shot noise (Poisson distribution, signal dependent). Cause: fluctuation of photon counts. It becomes *negligible* and more Gaussian-like distributed with more photons (good light conditions, and larger pixels).
- Impulsive noise (Laplacian or heavy-tailed distribution). Cause: long exposure time, A/D errors, and transmission errors (*rare*).
- Compression artifacts. This *depends* on the user defined compression level.

Throughout this chapter, we assumed the errors are uncorrelated and Gaussian distributed, which is generally a reasonable assumption. Depending on the application, however, other types of noise might dominate and must be addressed accordingly. In particular, since we use PCA as a pre-denoiser of the LR images, it is essential for us to consider other forms of PCA in accordance with the application at hand. In addition, we might need to consider data-fitting terms other than the L2-norm (LS solution). For example, we could use weighted LS if the reference PPC contains colored noise or an L1-norm data-fitting term for impulsive noise.

Variants of PCA

Assuming that the error and the signal parts of the data are independent, if the covariance matrix of error, R_v , is known and the data's covariance matrix, C_y , is known

as well, then the PCA basis, $\{\underline{e}_\ell\}_{\ell=1}^d$, that maximizes the SNR of the PCs, subject to their being uncorrelated²⁰ with respect to the error's covariance matrix, is given by [32].

$$(3.25) \quad \max_{\underline{e}_\ell} \frac{\underline{e}_\ell^T C_y \underline{e}_\ell}{\underline{e}_\ell^T R_v \underline{e}_\ell} \quad \text{subject to} \quad \underline{e}_q^T R_v \underline{e}_\ell = 0 \quad \text{for } 1 \leq q < \ell, \quad \ell \neq 1,$$

which is equivalent to

$$\begin{aligned} & \max_{\underline{e}_\ell} \underline{e}_\ell^T C_y \underline{e}_\ell \quad \text{subject to} \\ & \underline{e}_\ell^T R_v \underline{e}_\ell = 1 \\ & \underline{e}_q^T R_v \underline{e}_\ell = 0 \quad \text{for } 1 \leq q < \ell, \quad \ell \neq 1. \end{aligned}$$

Clearly, if $R_v = \sigma_v^2 I_d$, then $\{\underline{e}_\ell\}_{\ell=1}^d$ are the eigenvectors of C_y , which is the conventional PCA basis. Otherwise, the non-convex problem (3.25) can be solved by solving the eigen problem

$$C_y \underline{e}_\ell = \lambda_\ell R_v \underline{e}_\ell$$

subject to the constraints $\underline{e}_q^T R_v \underline{e}_\ell = 0$, $1 \leq q < \ell$, $\ell \neq 1$.

In practice, the data covariance matrix is unknown and an estimate of it becomes necessary. The sample covariance matrix can be used in the above system of equations to find the PCA basis that maximizes the SNR. However, when the errors are correlated and/or are not Gaussian²¹ distributed, using the sample covariance matrix might give unsatisfactory denoising results. For example, for heavy-tailed distributed errors, a *robust* form of PCA is preferred [69]. And for errors that are Gaussian distributed but *correlated*, Wentzell et al. [70] advocate ML-PCA, which is a PCA *estimator* that is optimal in the ML sense and is tightly related to extended-weighted TLS²² [71, 33].

In the future, we would like to investigate variants of PCA to better denoise the LR images when contaminated with errors that hardly follow the i.i.d Gaussian model.

²⁰ Imposing the condition that the PCs must be uncorrelated simplifies the expression for the (reconstruction) MSE, which simplifies deriving the optimization problem that defines the PCA basis.

²¹ Note that the *theoretical* PCA performance as an optimum linear denoiser is independent of the distribution of either the signal or the noise as it depends on first and second order statistics only. The quality of the *estimation* of the covariance matrix, however, is dependent on the distribution of the data.

²² Extended-weighted TLS also addresses the problem of parameter estimation when the Gaussian noise is correlated.

Other Post-Processing Options

The post processing techniques mentioned in §3.6 are admittedly very generic and therefore we might want to consider more sophisticated options. For example, if the leftover noise is a bit significant, using TV with a low enough data fidelity parameter would smooth out textured areas of the image and hence using an adaptive TV method [65] would be a better option. Also, we might get better results by jointly deblurring and denoising [66]. In addition, there are other alternatives for the data fitting term in the minimization problem (3.24), for handling non-Gaussian error, like the impulsive noise [67] or poisson noise [68]. Of course, the literature on denoising and deblurring is huge, but these examples are particularly attractive since they involve edge-preserving processing.

CHAPTER IV

Estimation of the Reference Polyphase Component

4.1 Introduction

At the end of chapter II, we mentioned that in order to be able to estimate the reference PPC, two sets of LR images must be obtained from two image sensors with different sensor densities. We refer to the set of the LR images acquired by the primary sensor (Figure 2.4) as the primary LR set (corresponding to the primary downsampling factor, I). The LR images acquired by the secondary sensor are referred to as the secondary LR set (corresponding to the secondary downsampling factor, J). The $\downarrow I \times I$ PPCs, $\{\underline{U}_n\}_{n=1}^{I^2}$, and the $\downarrow J \times J$ PPCs, $\{\underline{U}_m\}_{m=1}^{J^2}$ are referred to as the primary and secondary PPCs, respectively. We assume that $J = I + 1$ for the maximum possible overdeterminedness of the systems of equations we solve (3.6). The reference PPC we need to estimate is one of the secondary PPCs, i.e. we need to estimate \underline{U}_m for some m between 1 and J^2 .

As explained in chapter II, under the assumption of linearity, a set of LR images can span PPCs of the same resolution level (corresponding to the same downsampling factor). Therefore we assume that

$$(4.1) \quad \begin{aligned} \{\underline{U}_n\}_{n=1}^{I^2} &\in R(Y) \\ \{\underline{U}_m\}_{m=1}^{J^2} &\in R(Y^S), \end{aligned}$$

where Y and Y^S contain the primary and secondary LR images, respectively. According to the property of sampling diversity we have

$$\begin{aligned}
& j = T_n(m) \\
& i = T_m(n) \\
(4.2) \quad & \Leftrightarrow \underline{U}_{n,j} = \underline{U}_{m,i}
\end{aligned}$$

for any n and m . Recall that $\underline{U}_{n,j}$ and $\underline{U}_{m,i}$ are the j -th and i -th *sub* PPCs of \underline{U}_n and \underline{U}_m , respectively, and they are equal for $j = T_n(m)$ and $i = T_m(n)$. Refer to §2.3 for details.

Since $\underline{U}_{n,j} = D_j \underline{U}_n$ and $\underline{U}_{m,i} = D_i \underline{U}_m$, where D_j and D_i are the $\downarrow J \times J$ and $\downarrow I \times I$ downsampling matrices corresponding to the j -th and i -th *sub* PPCs, respectively, and in light of (4.1), equation (4.2) can be re-written as

$$(4.3) \quad D_j Y \underline{x}_n = D_i Y^S \underline{x}_m,$$

where \underline{x}_n and \underline{x}_m are the expansion coefficients of \underline{U}_n and \underline{U}_m in terms of Y and Y^S , respectively. Equation (4.3) enables us to estimate the (m -th) reference PPC¹, by solving for its expansion coefficients in terms of the secondary set of LR images. To simplify notation, let

$$\begin{aligned}
& A_1 \triangleq D_j Y \\
& A_2 \triangleq D_i Y^S \\
& \underline{x}_1 \triangleq \underline{x}_n \\
(4.4) \quad & \underline{x}_2 \triangleq \underline{x}_m,
\end{aligned}$$

and therefore (4.3) is rewritten as

$$(4.5) \quad A_1 \underline{x}_1 = A_2 \underline{x}_2.$$

4.2 Minimizing the Euclidean Distance in the Pixel Domain

First, we start by reformulating (4.5) as a homogeneous system of equations

$$(4.6) \quad A \underline{x} = \underline{0},$$

where

$$\begin{aligned}
A &= [A_1 \mid -A_2] \\
\underline{x} &= [\underline{x}_1^T, \underline{x}_2^T]^T.
\end{aligned}$$

¹ In §4.4 we explain that equation (4.3) is not unique for any *arbitrary* choice of m and n and we describe how this fact should be dealt with.

An obvious approach to solving equation (4.6) is to minimize the L2-norm of $A\underline{x}$, subject to avoiding the trivial zero solution,

$$(4.7) \quad \min_{\underline{x}} \|A\underline{x}\|^2 = \underline{x}^T A^T A \underline{x} \quad \text{subject to} \quad \|\underline{x}\|^2 = 1.$$

Problem (4.7) is non-convex (because of the quadratic equality constraint) but it has a well-known analytical solution. First, let

$$(4.8) \quad A = W \Sigma V^T = \sum_{k=1}^N \sigma_k \underline{w}_k \underline{v}_k^T$$

denote the (reduced) SVD of A , where $N = K + K^S$. The solution of (4.7) is

$$(4.9) \quad \hat{\underline{x}} = \underline{v}_N,$$

which is the last right singular vector² of A .

Note that problem (4.7) is equivalent to

$$(4.10) \quad \min_{\underline{x}_1, \underline{x}_2} \|A_1 \underline{x}_1 - A_2 \underline{x}_2\|^2 \quad \text{subject to} \quad \|\underline{x}_1\|^2 + \|\underline{x}_2\|^2 = 1,$$

which simply finds the two vectors in $R(A_1)$ and $R(A_2)$, with the minimum Euclidean distance between them.

4.2.1 Incomplete, Noisy Basis

Solving (4.5) by solving (4.10) is based on the assumption that the two vectors in $R(A_1)$ and $R(A_2)$, that best approximate $\underline{U}_{n,j}$ and $\underline{U}_{m,i}$, respectively, have the minimum Euclidean distance between them. But how accurate is this assumption? Note that (4.3), and thus (4.5), implicitly assume noise-free and complete primary and secondary LR basis, in which case the minimum Euclidean distance (4.10) is equal to zero and thus solving (4.10) solves (4.5) exactly. Of course, the LR images are always noisy and they do not exactly fully represent the PPCs, and therefore solving (4.10) is not necessarily the

² This result can be easily derived using Lagrange multipliers [22]. Note that $p = M_1 M_2 / I^2 J^2 \geq N - 1$ is a necessary condition for a unique solution. This supersedes (3.6).

best option. In fact, the Euclidean distance, as a *dissimilarity* measure, is known to be sensitive to errors (noise, outlier LR images and the incompleteness of the LR basis, in our case). The (squared) Euclidean distance is simply the sum of the square of differences between pixels, and since the pixels are highly correlated, errors will greatly *bias* the decision as to which two vectors in $R(A_1)$ and $R(A_2)$ are closest to each other. In §4.3, we suggest a better alternative to solving (4.5). Moreover, besides bias, the problem setup of (4.10) can be numerically unstable as we shall see next.

4.2.2 Noise Magnification

Small gaps between the last few of the N singular values of matrix A in problem (4.7), which is exactly equivalent to (4.10), result in a similar numerical instability as that of the TLS solution we discussed in chapter III. Since the columns of A are *sub* LR images (unwrapped by column) obtained from the primary and secondary LR sets, these columns can be highly correlated causing the gaps between the last few of the N singular values to be small³. Specifically, if we partition the matrix A as follows

$$A = [Z \mid \underline{z}_N],$$

where Z is a submatrix of A containing all the columns of A except the last column which we denote \underline{z}_N , then in light of §3.2, the solution (4.9) can be rewritten as

$$(4.11) \quad \begin{aligned} \hat{\underline{x}} &= \underline{y}_N \\ &= \left[-c \left((Z^T Z - \sigma_N^2 I)^{-1} Z^T \underline{z}_N \right)^T \mid c \right]^T, \end{aligned}$$

where c is the last element in \underline{y}_N , and σ_N is the smallest singular value of A (4.8). Therefore, if the last singular values of A are close to each other, then, by the interlacing theorem, its submatrix Z will have its last singular values close to each other *and* to σ_N as well.

³ Recall that the last few of the N singular values cannot be zero due to presence of (white) noise.

Denoising

Equation (4.11) reveals that the components of the solution $\hat{\underline{x}}$ can be *large*. In order to regularize, one might consider adding a regularization constraint to the non-convex problem (4.7). For example, we could limit the L1-norm of the solution to a certain threshold, but in this case, an analytical solution to the new non-convex problem does not exist and we would have to solve it approximately (using the *convex-concave procedure*, for example).

A simple and effective method to *denoise* $\hat{\underline{x}}_2$, which contains the last K^S elements of $\hat{\underline{x}}$, is inspired by the TSVD discussed in §3.1. First, let B_2 denote the matrix containing the left singular vectors of A_2 corresponding to the K^S (non-zero) singular values, then

$$A_2 \hat{\underline{x}}_2 \in R(B_2).$$

Equation (4.11) suggests that the highest order components of $A_2 \hat{\underline{x}}$ and thus, $A_2 \hat{\underline{x}}_2$, could be very noisy. Therefore, we could represent $A_2 \hat{\underline{x}}_2$ in terms of a reduced basis matrix, \tilde{B}_2 , which excludes the left singular vectors corresponding to the smallest q singular values. This is equivalent to removing the highest order components of $A_2 \hat{\underline{x}}_2$. We then perform a change of coordinates to get back a denoised version of $\hat{\underline{x}}_2$, which we denote $\hat{\underline{x}}_2^d$

$$(4.12) \quad \hat{\underline{x}}_2^d = \left(A_2^T A_2 \right)^{-1} A_2^T \tilde{B}_2 \tilde{B}_2^T A_2 \hat{\underline{x}}_2.$$

The estimated⁴ reference PPC component is thus

$$(4.13) \quad \hat{U}_m = Y^S \hat{\underline{x}}_2^d.$$

However, as previously explained in §4.2.1, the reason we seek a different approach to solving (4.5) is the *decision bias* caused by the sensitivity of the Euclidean distance, as a dissimilarity measure, to errors (such as noise and incompleteness of the LR basis).

Finally, we would like to mention that PCA pre-denoising the primary and secondary LR images, as described in §3.4, greatly reduces both the noise magnification⁵ and

⁴ The reference PPC is estimated up to a scale factor. Nevertheless, since we assume the reference PPC has the energy of a secondary LR image, we scale it accordingly.

decision bias. Nevertheless, we still get better results by solving the problem as described next.

4.3 Minimizing the Euclidean Distance in a Decorrelated Subspace

Our goal is to find the two vectors $\underline{f} \in R(A_1)$ and $\underline{g} \in R(A_2)$ with minimal dissimilarities. However, columns of A_1 are highly correlated and therefore a vector, written as a linear combination of these columns, is highly correlated with these columns, and thus the correlation among the vector's elements (pixels) is high. The same can be said regarding vectors in $R(A_2)$. This means that the choice of the pair of vectors with minimal Euclidean distance can be greatly biased by any kind of perturbations. Therefore, removing the dependencies among pixels in \underline{f} and pixels in \underline{g} , before deciding which \underline{f} and \underline{g} are with minimal dissimilarity, gives a less biased decision. Therefore, we could minimize the Euclidean distance in a lower-dimensional decorrelated subspace, using PCA since it decorrelates by removing first and second order dependencies between the variables (pixels). It gives us a basis, in terms of which, the expansion coefficients (PCs) of (centered) $\underline{f} \in R(A_1)$ and $\underline{g} \in R(A_2)$ are uncorrelated, and with the lowest order PCs having the highest variances, which gives them the greatest weight in the choice of the pair \underline{f} and \underline{g} with minimal dissimilarity. The underlying assumption here is that the PCs with high variance represent significant *features*. Moreover, the fact that the SNR of the low order PCs is maximized means smaller decision bias (although maximizing the SNR does not address error due to incompleteness of the basis). Hence, (4.10) is replaced with

$$\min_{\underline{x}_1, \underline{x}_2} \left\| D^T (A_1 \underline{x}_1 - A_2 \underline{x}_2) \right\|^2 \quad \text{subject to} \quad \|\underline{x}_1\|^2 + \|\underline{x}_2\|^2 = 1,$$

which is equivalent to

$$(4.14) \quad \min_{\underline{x}} \left\| D^T A \underline{x} \right\|^2 = \underline{x}^T A^T D D^T A \underline{x} \quad \text{subject to} \quad \|\underline{x}\|^2 = 1,$$

where

⁵ Smaller amount of (white) noise means larger gaps between the last singular values and less noise to be magnified.

$$A = [A_1 \parallel -A_2]$$

$$\underline{x} = \begin{bmatrix} \underline{x}_1^T & \underline{x}_2^T \end{bmatrix}^T,$$

A_1 and A_2 are obtained from the PCA pre-denoised data, and D is the reduced PCA matrix used to denoise the data as described in §3.4.1. Hence, the same matrix D used to denoise the LR images (by denoising sub LR images) is also used to decorrelate $\underline{f} \in R(A_1)$ and $\underline{g} \in R(A_2)$. The solution of problem (4.14) is the last right singular vector of $D^T A$.

4.4 Which Reference PPC to Estimate?

In chapter II, and at the beginning of this chapter, we explained that any secondary PPC shares a *sub* PPC with any primary PPC (4.2). In chapter III, we have seen how this fact is used to estimate the expansion coefficients of the primary PPCs given their sub PPCs which are derived from a reference (secondary) PPC. In this chapter, we also use the property of sampling diversity to estimate the expansion coefficients of the reference PPC as demonstrated by (4.3). However, equation (4.3) is not unique for any arbitrary choice of m and n . For example, suppose the primary downsampling factor, $I = 4$, and the secondary downsampling factor, $J = 5$, and we want to estimate the 13-th (out of 25) secondary PPC, $\underline{U}_{m=13}$, as our reference PPC, using its sub PPC shared with the first (out of 16) primary PPC, $\underline{U}_{n=1}$. According to the sampling diversity property (see §2.3.2)

$$j = T_{n=1}(13) = 19$$

$$i = T_{m=13}(1) = 11$$

$$\Leftrightarrow \underline{U}_{n=1, j=19} = \underline{U}_{m=13, i=11}.$$

In other words, the **19-th** (out of 25) *sub* PPC of the first primary PPC is equal to the **11-th** (out of 16) *sub* PPC of the 13-th secondary PPC. However, the **19-th** *sub* PPC of the second primary PPC is equal to the **11-th** *sub* PPC of the 14-th secondary PPC. Also, the **19-th** *sub* PPC of the 11-th primary PPC is equal to the **11-th** *sub* PPC of the 25-th secondary PPC. In fact, we have

$$\begin{aligned}
\underline{U}_{n=1,j=19} &= \underline{U}_{m=13,i=11} \\
\underline{U}_{n=2,j=19} &= \underline{U}_{m=14,i=11} \\
\underline{U}_{n=3,j=19} &= \underline{U}_{m=15,i=11} \\
\underline{U}_{n=5,j=19} &= \underline{U}_{m=18,i=11} \\
\underline{U}_{n=6,j=19} &= \underline{U}_{m=19,i=11} \\
\underline{U}_{n=7,j=19} &= \underline{U}_{m=20,i=11} \\
\underline{U}_{n=9,j=19} &= \underline{U}_{m=23,i=11} \\
\underline{U}_{n=10,j=19} &= \underline{U}_{m=24,i=11} \\
\underline{U}_{n=11,j=19} &= \underline{U}_{m=25,i=11}.
\end{aligned}$$

This has one consequence: equation (4.3) is not unique. Because, for example, while the 11-th sub PPC component of the 13-th secondary PPC is not equal to the 11-th sub PPC of the 20-th secondary PPC, any 11-th sub PPC is spanned by the same set of sub LR images (of the secondary set). Similarly, while the 19-th sub PPC component of the first primary PPC is not equal to the 19-th sub PPC of the 7-th primary PPC, any 19-th sub PPC is spanned by the same set of sub LR images (of the primary set). Namely, we have to solve the *same* equation

$$(4.15) \quad D_{19}Y\underline{x}_n = D_{11}Y^S\underline{x}_m,$$

regardless of whether our goal is to estimate the 13-th, 14-th, 15-th, 18-th, 19-th, 20-th, 23-th, 24-th, or the 25-th secondary PPC, as our reference PPC. In other words, solving (4.15) will give us the expansion coefficients, \underline{x}_m , of a reference PPC, without knowing which one (which m) it is from among the list above. In fact, (4.3) is unique only for the following choices of n and m

$$(4.16) \quad (n, m) \in \left\{ (1, J^2), (I, J^2 - J + 1), (I^2 - I + 1, J), (I^2, 1) \right\}.$$

For example, for $I = 4$, and $J = 5$, only the 7-th *sub* PPC of the first ($n = 1$) primary PPC is equal to the first *sub* PPC of the 25-th ($m = 25$) secondary PPC.

So which secondary PPC component (out of J^2) should we estimate as our reference PPC? And using which sub PPC (out of I^2)? Should we only limit ourselves to the four possible choices (4.16) for which equation (4.3) is unique? The fact is these four choices

do not necessarily give the best estimation of a reference PPC. The procedure for estimating the reference PPC, and determining which secondary PPC it is, is as follows.

- Pick $m = m^*$, which is the middle value between 1 and J^2 . For example, initially assume that we are estimating the 13-th ($m^* = 13$) secondary PPC out of 25 ($J = 5$).
- Find I^2 different estimates of the reference PPC (the m^* -th secondary PPC) based on all I^2 possible sub PPCs ($n = 1, \dots, I^2$). In other words, solve (4.3) I^2 times for a fixed m^* .
- Since a PPC is expected to have large high frequency components, due to aliasing, we pick $n = n^*$, for which the estimated reference PPC has significant energy content in the high frequency band, relative to its total energy, thus discarding smooth estimates of the reference PPC. Namely,

$$n^* = \max_n \frac{\left\| \hat{U}_m \Big|_n ** \Phi \right\|_F^2}{\left\| \hat{U}_m \Big|_n \right\|_F^2} \quad \text{subject to} \quad \frac{\left\| \hat{U}_m \Big|_n ** \Phi \right\|_F^2}{\left\| \hat{U}_m \Big|_n \right\|_F^2} < ub,$$

where ub is the upper bound⁶ ($\sim 1\%$) on the energy of the high frequency components of the reference PPC, relative to its the total energy, $**$ denotes the 2-D convolution, and

$$(4.17) \quad \Phi = \frac{1}{16} \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

is a unity-gain differentiator (or a high pass filter).

- Now that we have an estimated reference PPC, we find the set of values of m (and n) for which $T_n(m) = T_{n^*}(m^*)$ and, at the same time, $T_m(n) = T_{m^*}(n^*)$, i.e.

$$(4.18) \quad (n, m) = \left\{ (q, \ell) \in \{1, \dots, I^2\} \times \{1, \dots, J^2\} : T_q(\ell) = T_{n^*}(m^*) \text{ and } T_\ell(q) = T_{m^*}(n^*) \right\}.$$

This gives the set of candidate values of m , from which we find the most suitable one to be assigned to the estimated reference PPC as described below.

⁶ Recall that we solve (4.3) by solving (4.14) where the decision, as to which pair of vectors in the feature subspace are closest to each other, is mainly determined by the low order PCs, and that the decision is independent of the *mean*. Therefore, depending on the sub data matrices, these low order PCs can bias the decision towards a solution with greatly emphasized high frequency contents.

- Using the estimated reference PPC, we estimate the HR image (by estimating its I^2 primary PPCs) for each value of m from the set defined in (4.18). Since misassignment of the estimated reference PPC (assigning the wrong value of m to the estimated reference PPC) results in a rough HR image, we pick the value of m from the set (4.18) for which the reconstructed HR image has the smallest high frequency components

$$\min_m \left\| \hat{u}|_m ** \Phi \right\|_F^2,$$

where $\hat{u}|_m$ is the estimated HR image using the estimated reference PPC as being the m -th secondary PPC, and Φ is the differentiator defined in (4.17).

Because the same *sub* PPCs share the same *sub* LR basis, resulting in the non-uniqueness of equation (4.3), except for choosing (n, m) according to (4.16), choosing m^* to be in the middle, and then estimating the reference PPC I^2 times, covers approximately half of all the possible $I^2 J^2$ choices of the pair (n, m) . For example, for $I = 4, J = 5$, and $m^* = 13$, estimating the reference PPC 16 times (for $n = 1, \dots, 16$), covers 196 (out of 400 possible) choices of the pair (n, m) . See Figure 4.1 for a visualization of the area covered by choosing $m^* = 13$ and estimating the reference PPC 16 times.

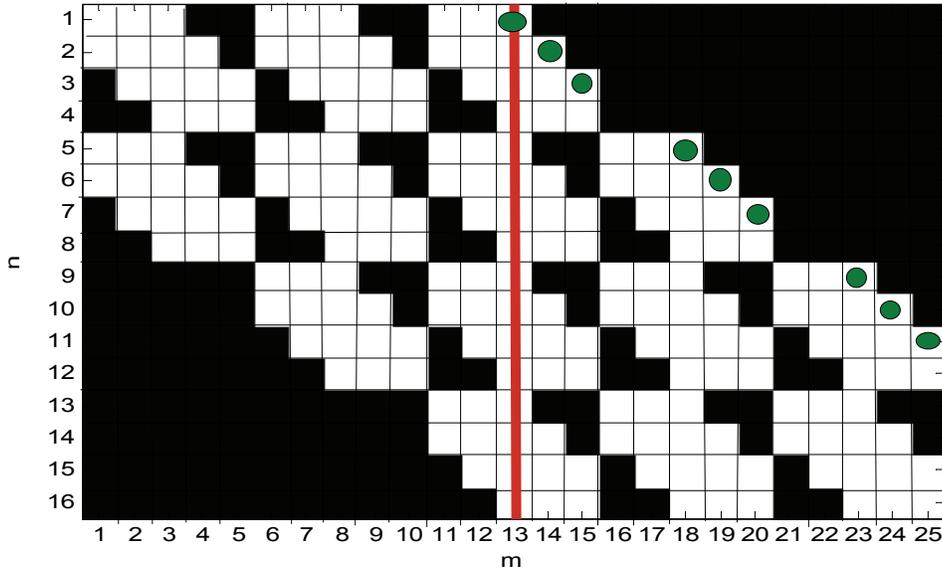


Figure 4.1: For $I = 4, J = 5, m^* = 13$ and $n = 1, \dots, 16$, the highlighted (white) blocks represent all the pairs (n, m) that give the same pairs of sub data matrices given by (n, m^*) . For example, the green dotted blocks represent the pairs (n, m) that share the same equation (4.3) corresponding to $(n = 1, m^* = 13)$.

4.5 An Intuitive Alternative to Estimating the Reference PPC

Instead of estimating the expansion coefficients of a reference PPC, in terms of the secondary LR basis, by solving an equation of the form, $A_1 \underline{x}_1 = A_2 \underline{x}_2$, we describe below how to choose a single secondary LR image as our reference PPC. In other words, since the LR images and the PPCs are highly correlated, then why not pretend that one of the available secondary LR images can pass for one of the secondary PPCs? There is one limitation to this idea: except for the case of perfectly pure translational motion, a LR image is normally a mixture of PPCs⁷. However, loosely speaking, a LR image can be viewed as a blurred version of one of the PPCs. Thus the super-resolved image will be (at least) as biased (blurred) as the secondary LR image, we pick as our reference PPC. We describe below a simple two-step procedure for choosing a reference PPC from among the secondary LR set, and for determining which secondary PPC it is.

- Since a PPC is expected to have large high frequency components, after normalizing the LR images to have the same energy, we pick the secondary LR image that has the largest high frequency components. In addition, since secondary LR images that are farthest from the (downsized) mean of the primary LR set, $\underline{\mu}_d^P$, are the least relevant to the reconstruction of the primary PPCs of the HR image (refer to §3.4.2), we make sure that we do not pick an ‘outlier’ secondary LR image. For choosing the ‘best’ secondary LR image as our reference PPC, we use

$$(4.19) \quad \max_k \frac{\left\| y_k^S ** \Phi \right\|_F^2}{\left\| \underline{y}_k^S - \underline{\mu}_d^P \right\|},$$

where Φ is the differentiator defined in (4.17), and y_k^S is the k -th secondary LR image.

- Using the chosen secondary LR image (4.19), we determine which of the secondary PPCs it best represents (determine the most suitable m) by estimating the HR image for $m = 1, \dots, J^2$. Then we assign, to the chosen LR image, the value of m , for which the reconstructed HR image is the smoothest, i.e.

⁷ In fact, even for the case of pure translational motions, a LR image is blurred because of the CCD averaging effect as shall be explained in chapter V.

$$\min_m \|\hat{u}|_m - \Phi\|_F^2,$$

where $\hat{u}|_m$ is the estimated HR image using the chosen secondary LR image as being the m -th secondary PPC.

This simple method is expected to outperform estimating the reference PPC if there is at least one secondary LR frame that is sharper than an estimated reference PPC. This is almost always guaranteed for the case of pure translational motion.

The Effect of Outliers

When there are outlier images, estimating the reference PPC is greatly affected as it involves solving an equation of the form, $A_1 x_1 = A_2 x_2$, and thus outlier elements can be on both sides of the equation. In §3.4.2 we described a simple method to get rid of outliers from both the secondary and the primary sets of LR images, for better PCA pre-denoising. The same method can be used for a better estimation of the reference PPC in presence of outliers. Of course, if we choose a secondary LR image as our reference PPC, as described above, outliers will have no effect as their corresponding expansion coefficients should be zero, since the chosen secondary LR cannot be an outlier. This fact can be used to estimate the *number* of outlier primary LR images. Specifically, using a secondary LR image as a reference PPC, if we average the squared expansion coefficients of all primary PPCs in terms of the primary LR set, we can estimate the number of irrelevant (outlier) primary LR images by counting the number of averaged squared coefficients that are close to zero. The number of outlier secondary LR images will be also the same if the primary and secondary sensors see the same scene at the same time, by using a beam splitter. Otherwise, the number of outlier secondary LR images has to be guessed.

Finally, we would like to reiterate that removing outlier LR images should be considered only for better PCA pre-denoising performance and if we are going to *estimate* the reference PPC (rather than simply *choosing the best* secondary LR image as our reference PPC).

CHAPTER V

Applications and Experimental Results

5.1 Applications

5.1.1 Introduction

Although the primary goal of multiframe super-resolution (SR) is to provide a cheap alternative to building expensive high density imaging sensors, even the priciest diffraction-limited systems can still benefit from SR techniques when imaging larger areas. On one hand, to capture wider field-of-view (FOV) images (with the same resolution level) we need higher pixel density. On the other hand, larger FOV requires zooming out (decreasing the focal length). This results in a smaller Airy radius¹ and the imaging system is thus no longer diffraction-limited. The empirical formula below explains the parameters affecting the Airy radius, δ

$$\delta = 1.22 \lambda \frac{f}{a},$$

where λ is the wavelength of light, f is the focal length, and a is the diameter of the aperture. This means that any imaging system can benefit from the resolution enhancement² via (signal processing) SR methods, at least when imaging wide areas.

In the following sections, we discuss some of the applications where our proposed SR method can be implemented.

¹ The Airy radius is the smallest resolvable distance between two point objects. The larger the diffraction of light, the larger the radius.

² When the sensor has a pixel density of 2 pixels per Airy radius, the sensor is said to be diffraction-limited which means that higher pixel densities cannot enhance the resolution.

5.1.2 The Case of Approximately Pure Translations

In some applications, the relative scene motion can be modeled as pure translations. For example, a video camera recording a video sequence of a static scene while moving with slight translations, or a scanner scanning the same document several times with slightly different initial points [7]. Several papers were completely devoted to treat this classical SR problem, for example [3-7]. Unlike previous work, our fast blind reconstruction method does not require registration.

5.1.3 Super-resolution from Vibrations

In applications such as airborne and ground reconnaissance, robotics and machine vision systems, vibrations are inevitable during imaging, and despite the best mechanical stabilization systems, images still come out distorted by motion blur [38, 39]. Because of the *random* nature of the blur associated with vibrations, conventional motion-based SR methods, which are dependent on the accuracy of motion estimation, might not be a viable option. In particular, conventional image registration methods perform poorly when the blur is random (different from frame to frame). In order to mitigate the effect of the randomness of the motion blurs, the authors in [38] adopt the particularly computationally expensive method of projection onto convex sets (POCS) for image registration, blur estimation and SR reconstruction. Other work [45] proposes avoiding motion blur altogether by building a specialized *jitter* camera. This is done by shifting the video detector instantaneously and timing the shifts to occur between pixel integration periods. In the case of our method, the randomness of the motion blur is actually a desired quality and no estimation of the blur or image registration is needed, and images are super-resolved fast, and all for the simple hardware requirement of adding a lower resolution (secondary) CCD sensor.

5.1.4 Atmospheric Turbulence

Ground-based astronomical imaging and satellite imaging of the Earth are two applications that require imaging through the atmosphere. Unfortunately, the turbulent nature of the imaging medium (the atmosphere), distorts the images. The distortion can

be modeled as convolving the image with a speckle³ PSF. The size, shape and location of the PSF are time-variant (different from frame to frame). In addition, in the case of wide-area-imaging, the distortion is space-variant as well, which means that different regions, within the same frame, are distorted differently. This is known as anisoplanatic distortion, as opposed to isoplanatic distortion which is associated with a space-invariant PSF. Typically, all imaging through the atmosphere is subject to the anisoplanatic type of distortion unless the FOV is very narrow [50].

In short, imaging through the atmosphere can be modeled as a linear shift-variant (LSV) transform that is different from frame to frame. This means that our method can benefit from these randomly transformed frames to achieve super-resolution. However, it is well known that atmospheric distortion can be severe for long-exposure imaging (few frames per second). In addition, far-field imaging increases the severity of distortions. In our case, a certain amount of (time-variant) distortions is useful or in fact, necessary to achieve SR but according to the discussion in §2.2, large size PSFs (corresponding to severe blurring) require too many LR frames, and we cannot use *too many* LR images, even if we had a lot of them, since we need to keep our systems of equations overdetermined. Namely, only a moderate amount of atmospheric distortion can be useful for our method to give reasonable results. This means that the method is best suited for near-field, short-exposure imaging under reasonable atmospheric conditions. There are two applications that fit these requirements:

- Lunar imaging⁴.
- Satellite imaging of the earth.

In the case of lunar (and planetary) imaging at high rates of frames per second, while it reduces the severity of the distortions, it also lowers the SNR which makes it difficult to deblur these images as deblurring magnifies the noise. Stacking is a method aimed at preparing the images in such a way that they can be added together without increasing the blur while enhancing the SNR. The stacked image is then deblurred using one of the sharpening tools. Typically, hundreds of frames are used for stacking and the process is a lengthy one. While the purpose of stacking is deblurring, our goal is primarily removing

³ Speckle PSFs have very irregular shapes.

⁴ Obviously, the moon is a lot closer to Earth than any planet or star (near-field imaging) and it is a lot brighter which allows for much shorter exposure without the images getting too dim.

aliasing by increasing the pixel density. It is rather interesting to note that in the absence of atmospheric distortions, stacking is needless while in our case, SR is impossible.

When it comes to satellite imaging of objects on Earth, the distortions due to atmospheric turbulence are much smaller because the Earth's surface is in contact with the turbulent imaging medium (the atmosphere). This is similar to when an object behind a diffuse glass is observed. When the object is very close to the diffuse glass it appears much clearer than when it is far from it. Therefore, even when the conditions of the atmosphere are somewhat bad, satellite imaging of objects on Earth is still expected to be reasonably distorted, which makes our SR method particularly well-suited and potentially useful for this type of application⁵.

To the best of our knowledge, no one tried to super-resolve images distorted by the atmosphere⁶. This could probably be due to the fact that the atmospheric distortion contains both warps and blurring elements. Blur-based methods⁷ are not designed to work with warps and motion-based techniques might fail due to the fact that the prior step of image registration is sensitive to the randomness of the blur from frame to frame. And while there are attempts to handle the case of random motion blur [38-39, 45], the case of super-resolution of atmospherically distorted images is not addressed before.

5.2 Experimental Results

In this section we present the results we obtained from working with both synthetic and real data. Before we proceed, we would like to discuss the integrating effect of the CCD sensor. In particular, the LR images are related to the transformed (warped/distorted) HR images via downsampling by integration of pixels of the HR image. This can be modeled as an averaging PSF convolved with the transformed HR images followed by decimation.

Except for two of our experiments, we used primary LR images corresponding to $\downarrow 4 \times 4$ and secondary LR images corresponding to $\downarrow 5 \times 5$. For $\downarrow 4 \times 4$, the CCD PSF was

⁵ Although satellite surveillance usually uses high resolution imaging systems, for this type of application, being able to zoom out to cover larger areas, without aliasing, is an extremely useful feature that can be delivered using super-resolution.

⁶ By super-resolve, we primarily mean removal of *aliasing*.

⁷ Blur-based SR is very sensitive to model errors (for example, due to inaccurate estimates of the PSFs, when not known).

assumed to be a 4x4 Gaussian with variance equal to one [7, 12-14] and we used a 5x5 Gaussian PSF with the same variance for $\downarrow 5 \times 5$. This is reasonable since only a portion of the LR CCD pixel is active which means that the HR pixels (within a LR pixel) should not have the same integration weights. See Figure 5.1 and 5.2 for an illustration of the integration effect of the LR CCD arrays for $\downarrow 4 \times 4$ and $\downarrow 5 \times 5$, respectively.

For the remaining two experiments (Experiment 5 and 7), to obtain easily appreciable aliasing effect, the primary and secondary LR images correspond to downsampling by $\downarrow 8 \times 8$ and $\downarrow 10 \times 10$, respectively. For $\downarrow 8 \times 8$ and $\downarrow 10 \times 10$, downsampling, the CCD PSFs we used were (scaled) and resized versions of the 4x4 and 5x5 Gaussian PSFs mentioned above, respectively.

Note that the CCD PSF introduces the *same* additional distortion to all the frames, and thus its effect cannot be alleviated with more LR images. Specifically, if the HR image is distorted by different PSFs and then by the *same* averaging blur, the *overall* effect is that *what we solve for is a blurred version of the HR image*. This is another reason why post-processing (via unsharp masking, for example) is necessary since our method is non-parametric and the solution cannot account for the *common* CCD averaging effect. In short, the CCD PSF is an additional source of bias, over which we have no control and cannot address except via post-processing.

Bias Due the Incompleteness of the LR Basis

In chapter III, we discussed the bias of the super-resolved image under the assumption that (the noiseless version of) the LR images form a complete basis. However, the incompleteness of the (noiseless version) of the LR basis adds more bias to the solution. According to our experiments, this additional bias takes the *form of both aliasing and blur* when

$$(5.1) \quad K < rI^2,$$

where r is the number of LSI kernels that approximate the LSV transform, undergone by the HR. ($r = 1$ in the LSI case. Refer to §2.2.2). However, if

$$(5.2) \quad rI^2 < K < rL_1L_2,$$

where $L_1 \times L_2$ is the size of an LSI kernel and $L_1 \geq I$ and $L_2 \geq I$ (§2.2.1), then the bias due to the incompleteness of (the noiseless version of) the LR images takes the *form of blur only*.

In short, if the (noiseless version of) the LR set is incomplete only with respect to the *extent* of the distortions, then this will add bias in the form of *blur* only, which is far more tolerable than aliasing. The same can be said regarding estimating the reference PPC, which is more sensitive to the incompleteness of the basis (and errors, in general) since it involves solving an equation of the form, $A_1 \underline{x}_1 = A_2 \underline{x}_2$.

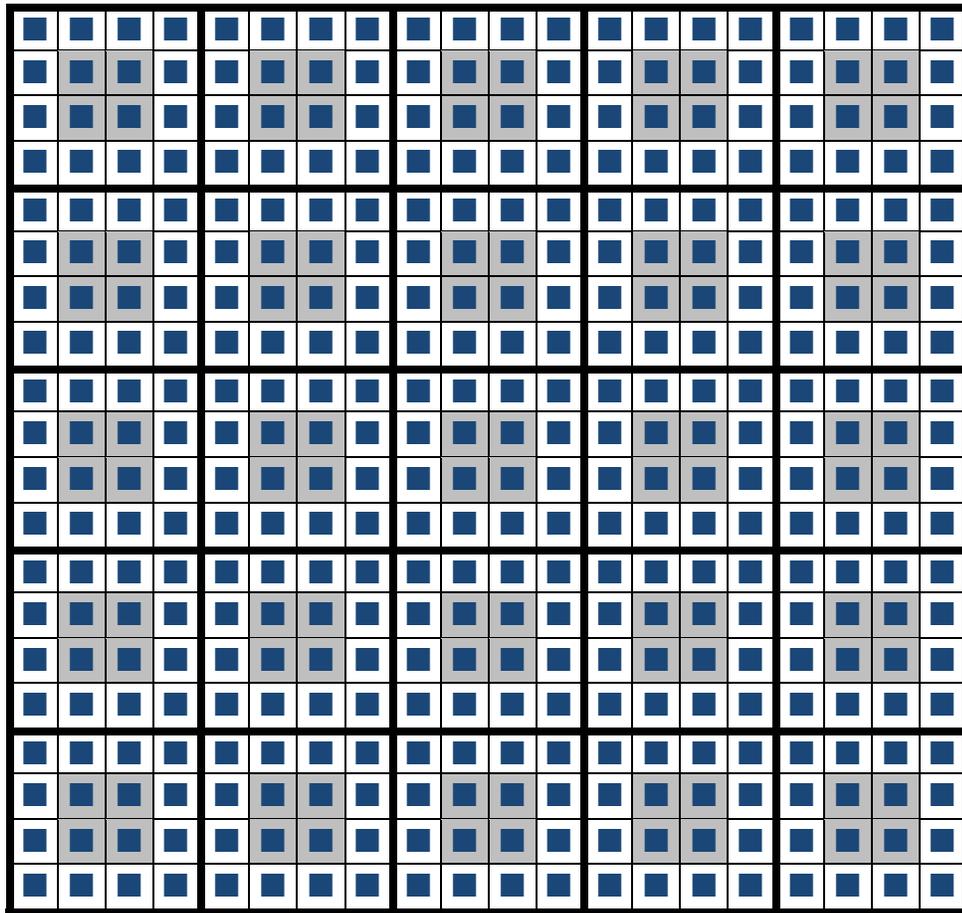


Figure 5.1: An illustration of the integration effect of the primary LR CCD array corresponding to $\downarrow 4 \times 4$. The gray shaded areas represent the active portions of the LR pixels. The small blue squares represent the active portions of the pixels of the HR CCD array.

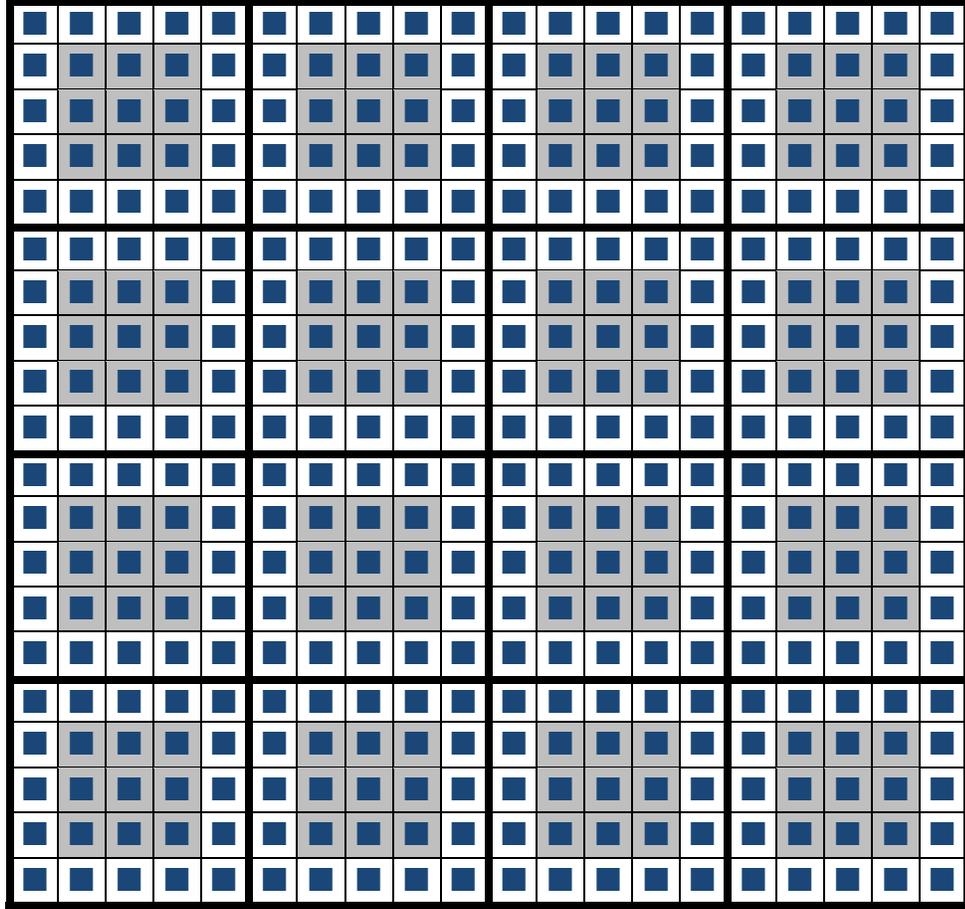


Figure 5.2: An illustration of the integration effect of the secondary LR CCD array corresponding to $\downarrow 5 \times 5$. The gray shaded areas represent the active portions of the LR pixels. The small blue squares represent the active portions of the pixels of the HR CCD array.

Miscellaneous

In all (but one of) these experiments, we PCA pre-denoised the data matrices, using a PCA matrix containing 10-30% of the eigenvectors⁸ of the sample covariance matrix of the *sub* LR images (§3.4). As mentioned previously, our method involves the solution of a few systems of linear equations where the number of unknowns is equal to the number of LR images. However, the PCA pre-denoising step considerably slows down⁹ the overall solution as it involves finding the eigenvectors of the sample covariance matrix.

⁸ Larger number of eigenvectors must be retained when using a lot of LR images, since the number of retained eigenvectors must exceed the total number of LR images or else (4.14) will not have a unique solution.

⁹ All computations were performed using MATLAB running on a 1.5 GHz Intel Core Duo CPU with 2GB RAM.

We compare some of our results to those obtained by the “iterative L1” solution, which is an implementation of equation (22) in [14], using the software in [49]. To be more specific, the authors in [14] propose solving equation (2.3) with an L1-norm data-fitting term and bilateral total variation for regularization. The two main advantages of their method are robustness to error (e.g. registration errors) and relative speed.

In the following experiments, our method proved to be *at most* ~ 10 times slower than bicubic interpolation and *at least* ~ 20 times faster than the iterative L1 algorithm. Moreover, our method works without motion/blur/distortion estimation and therefore it has an advantage over any model-based solution.

5.2.1 Synthetic Data Experiments

Experiment 1: LSI PSF

In this experiment we used the HR image ‘Building’ and obtained a synthetic sequence of differently blurred HR images, of size 460×620 , as follows. 16 random 5×5 PSFs were generated using MATLAB’s `rand` function. These were used to distort the same original image resulting in 16 blurred HR images. These images were downsampled by $\downarrow 4 \times 4$ and $\downarrow 5 \times 5$ to obtain the primary and secondary sets of LR images, respectively, which simulates the case where the primary and secondary sensors are placed in the same camera and a beam splitter is used so that both sensors *see* the same image at the same time (refer to §2.3.3 for details). Zero-mean white Gaussian noise was added at 30 dB SNR.

Recall that for this size of PSFs, in order for the primary (and secondary) LR basis to be complete, more than 25 frames are needed¹⁰. However, only 16 were available, which adds bias (blur) to the estimation of the reference PPC and the primary PPCs (5.2).

Figure 5.3 (a) shows the first primary LR image, resized ($\uparrow 4 \times 4$) using bicubic interpolation. Figure 5.3 (b) shows the super-resolved image (using an *estimated* reference PPC) after post-processing using TV, unsharp masking (UM) and median filtering (MD).

¹⁰ Since we downsample by averaging according to the CCD PSF, even having more than 25 LR images cannot get rid of the blur due to the sensor’s integrating effect and thus post-deblurring is always required.

The overall computation time (including pre-denoising the red and blue LR images) was 14.5 seconds (of which 4.66 seconds was for post-processing). Bicubic interpolation took 2.89 seconds.

Experiment 2: LSV PSF

In order to simulate a LSV PSF, we divided the HR ‘Building’ image into 8 subregions, and applied a randomly generated LSI PSF of size 4×4 in each one of these subregions. We repeated this process 100 times, to obtain 100 HR images, each distorted with a randomly generated LSV PSF. These images were downsampled by $\downarrow 4 \times 4$ and $\downarrow 5 \times 5$ to obtain the primary and secondary sets of LR images, respectively, and noise was added at SNR of 30 dB.

Since each LSV PSF is a set of 8 LSI PSFs, in order for the LR basis to be complete, we need *at least*¹¹ $8 \times 16 = 128$ LR images, of which we only have 100.

The first primary LR image was resized ($\uparrow 4 \times 4$) using bicubic interpolation and is shown in Figure 5.4 (a). Figure 5.4 (b) shows the super-resolved image, after post-processing (TV+UM+MD), which was computed in 28.1 seconds. Note that the computation time is greater in this case because the number of LR images (and thus the number of expansion coefficients we solve for) has increased from 16, in the previous example, to 100. More importantly, pre-denoising so many LR images (including the red and blue images), adds significantly to the computational load.

When we used smaller number of LR images (e.g. 50), the reconstructed image (not shown) had some regions that were super-resolved, while other regions were blocky (aliased). This, again, emphasizes the chief strength of our method in that it is not model-based and therefore despite violating the assumption of completeness of the LR basis, the HR image was, nevertheless, partially reconstructed.

¹¹ Again, since we downsample by averaging according to the CCD PSF, the super-resolved image will always be blurred and post-processing is needed at least to address the CCD blurring effect.



(a) Bicubic interpolation. Comp. time = 2.89 sec.



(b) Blind SR + post-processed (TV+UM+MD). Comp. time = 14.5 sec.

Figure 5.3: LSI PSF. (# of LRs = 16).



(a) Bicubic interpolation. Comp. time = 2.83 sec.



(b) Blind SR + post-processed (TV+UM+MD). Comp. time = 28.1 sec.

Figure 5.4: LSV PSF. (# of LRs = 100).

5.2.2 Real Data Experiments

Since we do not have cameras with two different¹² density sensors, we used real-world distorted HR image sequences and then downsampled them (by integrating the HR pixels) to get the two sets of primary and secondary LR images. In other words, in these experiments, the only simulated part of the degradation process is the downsampling.

For Experiments 3, 4, and 6, all the images were captured using the same camera, SONY Cyber-shot DSC-L1. For Experiment 5, Canon EOS DIGITAL REBEL XT was used.

Experiment 3: Approximately Pure Translations

The HR test sequence of images used for this experiment was obtained using a hand-held camera taking multiple monochromatic shots, of size 480×640 , of the same scene¹³, “Outdoors”. However, the camera moved slightly every time a picture was taken, thus approximating the pure translations case. A total of 108 shots were taken. The first half of these images were downsampled by $\downarrow 5 \times 5$ and the other half was downsampled by $\downarrow 4 \times 4$, producing the secondary and primary sets of LR images, respectively. This simulates the case where the two sensors are either placed in two different cameras or in the same camera, using a fully reflective mirror positioned in the optical path for half of the imaging time (refer to the discussion in §2.3.3).

We used only 35 primary LR images that are closest to the mean. Similarly, only 35 secondary LR images that are closest to the (resized) mean of the primary set were kept (§3.4.2). Then we pre-denoised these images using PCA. The HR image was reconstructed using the 35 primary LR set as a basis for its primary PPCs, and for a reference PPC, we used a single secondary LR image, chosen according to the procedure described in §4.5.

As noted in chapter IV, choosing a single secondary LR image for our reference PPC is expected to give better results, in the case of approximately pure translations, than estimating the reference PPC. This is because the translational motion does not cause any

¹² The different densities should correspond to downsampling factors that are relatively prime (or more usefully, consecutive integers.)

¹³ This was a page from the AAA Living magazine, May/June 2005 issue.

blur. We used UM for post-processing mainly to reduce the blur due to CCD averaging effect.

Figure 5.5 (a) shows the main portion of the first primary LR image, resized ($\uparrow 4\times 4$) using bicubic interpolation. Figure 5.5 (b) shows the main portion of the super-resolved image after post-processing (UM+MD). It took 1.03 seconds to perform the bicubic interpolation while the super-resolved image was computed in only 10.88 seconds¹⁴.

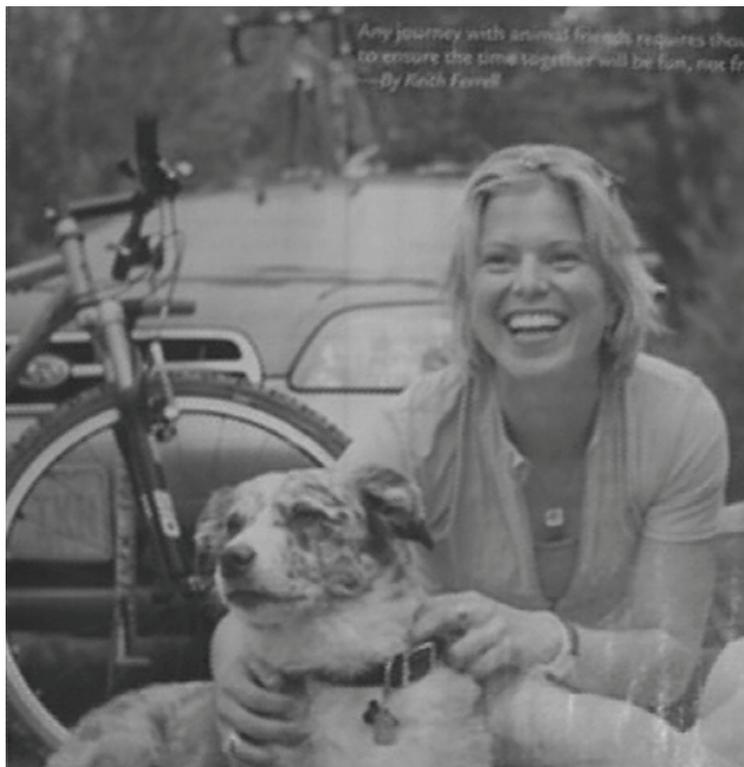
Figure 5.6 (a), (c) and Figure 5.7 (a), (c) show two different detail areas of the images shown in Figure 5.5.

Finally, for comparison, we reconstructed the HR image using the iterative L1 method [14, 49]. This took about 4 minutes (using 40 iterations, 0.001 regularization factor, and the shift & add image for the initial guess). The same two detail areas (of the dog's face and text) are shown in Figure 5.6 (b) and Figure 5.7 (b), respectively. Comparing Figure 5.6 (b) to Figure 5.6 (c), we notice that our method outperforms the iterative L1 method. However, by examining Figure 5.7 (b) and Figure 5.7 (c), we observe that the iterative L1's result is better. In other words, the two methods have an overall comparable performance when it comes to this experiment (although the blind SR method is much faster).

¹⁴ We note here that there was virtually no need for pre-denoising, but we pre-denoised to learn how much time this would cost for this experiment.



(a) Bicubic interpolation. Comp. time = 1.03 sec.



(b) Blind SR + post-processed (UM+MD). Comp. time = 10.88 sec.

Figure 5.5: Approximately pure translations. (# of LRs = 35).



(a) Bicubic interpolation.

(b) Iterative L1.



(c) Blind SR + post-processed (UM+MD).

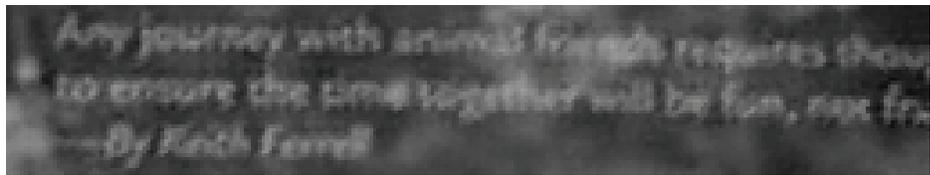
Figure 5.6: Approximately pure translations. Details: dog's face. (# of LRs = 35).



(a) Bicubic interpolation.



(b) Iterative L1.



(c) Blind SR + post-processed (UM+MD).

Figure 5.7: Approximately pure translations. Details: text. (# of LRs = 35).

Experiment 4: Approximately Pure Translations—Video

In this experiment we used a video of a HR static scene, “Watch”, of size 480×640 , displayed on a laptop screen. The video’s temporal resolution was 30 frames/second. The video contained periodic streaks which normally result from very close-range shooting of an LCD screen. The camera was slightly moving while recording. This approximately corresponds to the pure translational motion case.

We downsampled the first frame by $\downarrow 5 \times 5$ and used it as our reference PPC. Then, we downsampled every other frame in the next 100 frames by $\downarrow 4 \times 4$, of which we kept only 30 frames that are closest to the mean. In other words, we used only 30 frames, which we pre-denoised using PCA and then used as our primary LR basis set. The super-resolved image was then post-processed using TV, UM and MD.

Figure 5.8 shows the main portion of the super-resolved image compared to the corresponding area of the bicubic interpolated ($\uparrow 4 \times 4$) first primary LR frame. The iterative L1 result is shown in Figure 5.9, for comparison (number of iterations was 20, the regularization factor was 0.001 and the shift & add image was used as an initial guess).



(a) Bicubic interpolation. Comp. time = 3 sec.



(b) Blind SR + post-processed (TV+UM+MD). Comp. time = 21.3 sec.

Figure 5.8: Approximately pure translations—video. (# of LRs = 30).



Figure 5.9: Approximately pure translations—video: Iterative L1. (# of LRs = 30).

Experiment 5: Random Vibrations

A digital camera was mounted on a tripod and placed on a vibrating table. The captured images, of the black and white “Michigan Seal”, were thus randomly motion-blurred¹⁵. We used only the first 35 images. These motion-blurred images were of very high resolution (large number of pixels). We cropped¹⁶ them to size 960x960 and then downsampled them by $\downarrow 8 \times 8$ and $\downarrow 10 \times 10$ to obtain the primary and secondary sets of LR images of easily noticeable aliasing, respectively. Then we super-resolved to size¹⁷ 480x480.

Figure 5.10 (a) shows the first primary LR image, resized ($\uparrow 4 \times 4$) using bicubic interpolation. The reference PPC was first estimated in the *pixel domain*, by solving problem (4.7) without pre-denoising. Also, we ignored denoising the expansion

¹⁵ The vibrations were produced by continuously pounding on the table in different random locations while the camera was taking separate shots with a lowered shutter’s speed (exposure time = 1 second).

¹⁶ We cropped the blank ‘wall’ space in the images.

¹⁷ Note that given the dimensions of the primary and secondary LR images we can only super-resolve with a resolution gain of $\uparrow 4 \times 4$, since the ratio of their dimensions is $5/4$. Refer to §2.3.3.

coefficients as per (4.12). Due to noise magnification associated with estimating the expansion coefficients of the reference PPC, in the pixel domain (§4.2.2), the estimated reference PPC was extremely noisy which resulted in the noisy super-resolved image shown in Figure 5.10 (b).

The LR images were pre-denoised using PCA and then one of the secondary LR images was chosen as the reference PPC, according to the procedure in §4.5. The corresponding super-resolved image is shown in Figure 5.11 (a). Note that since the frames are motion blurred, even the best secondary LR image (that is closest to the mean) is slightly blurred and thus the corresponding super-resolved image is blurred as well.

Figure 5.11 (b) shows the super-resolved image based on an estimation of the reference PPC in the feature space (4.14) as described in §4.4. The result is clearly sharper than the super-resolved image based on a chosen secondary LR image.

In this experiment, there is some translational motion but most of the distortion is random blur. Moreover, motion estimation, because of the randomness of the blur, is inaccurate and thus the iterative L1 solution¹⁸ performed poorly as shown in Figure 5.12. This experiment serves to prove the advantage of our *non-parametric* approach to the solution of the problem of SR.

Experiment 6: Rhythmic Vibrations

We obtained a color video sequence of size $480 \times 640 \times 70$ (with temporal resolution of 30 frames/second) of the image “Life” (a page from National Geographic magazine, featuring life’s diversity and DNA, May 2010 issue). The camera was placed at approximately 1.5 feet from the page and the zoom-in function was used so as to avoid empty wall space. Vibrations were produced mechanically by attaching a vibrating device to the table on which we placed the camera. The vibrations were rhythmic in nature resulting in both global motion and motion blur.

The 70 frames were downsampled by $\downarrow 4 \times 4$ and $\downarrow 5 \times 5$ to produce the primary and secondary sets of LR images, respectively. These LR images were not pre-denoised¹⁹ and the reference PPC was taken to be one of the secondary LR images.

¹⁸ Number of iterations was 20, regularization factor was 0.001, and the initial guess was the shift & add image.

¹⁹ The TV post-processing could take care of the noise augmentation on its own.



(a) Bicubic interpolation. Comp. time = 0.83 sec.



(b) Blind SR + post-processed (TV+UM+MD). Ref. PPC was estimated in the pixel domain.

Figure 5.10: Random vibrations: estimating the ref. PPC in the pixel domain. No denoising. (# LRs = 35).



(a) Blind SR + post-processed (TV+UM+MD). A single sec. LR image was used as a ref. PPC. Comp. time = 7.22 sec.



(b) Blind SR + post-processed (TV+UM+MD). Ref. PPC was estimated (in the feature subspace). Comp. time = 6.9 sec.

Figure 5.11: Random vibrations: using a single sec. LR image vs. estimating the ref. PPC. (# LRs = 35).



Figure 5.12: Random vibrations: Iterative L1 + sharpened (UM). (# LRs = 35).

The super-resolved image was then post-processed using TV and UM²⁰. The total processing time was 13.57 seconds (of which 8.6 seconds were for TV post-processing!).

The reason we needed more images for this experiment, despite its being representative of the LSI case, is the fact that the rhythmic distortions did not allow for much change in the captured images within a small time frame. In fact, because the associated blur was not very random and that there was more global motion shifts, compared to the previous experiment, the iterative L1 method did relatively well, although there were still noticeable artifacts around the edges due to registration errors caused by the presence of (less random) motion blur.

Figures 5.13-5.15 show portions of the bicubic interpolated (and sharpened) first primary LR image and the corresponding portions of the (sharpened) iterative L1 SR image²¹ along with the matching parts of the SR image according to our method.

²⁰ For this experiment we used Photoshop's unsharp masking, as MATLAB does not provide much freedom with its unsharp masking tool.

²¹ Number of iterations was 50, regularization factor was 0.0015, and the initial guess was the shift & add image.

(a) Bicubic interpolation + sharpened (UM).
Comp. time = 2.65 sec.



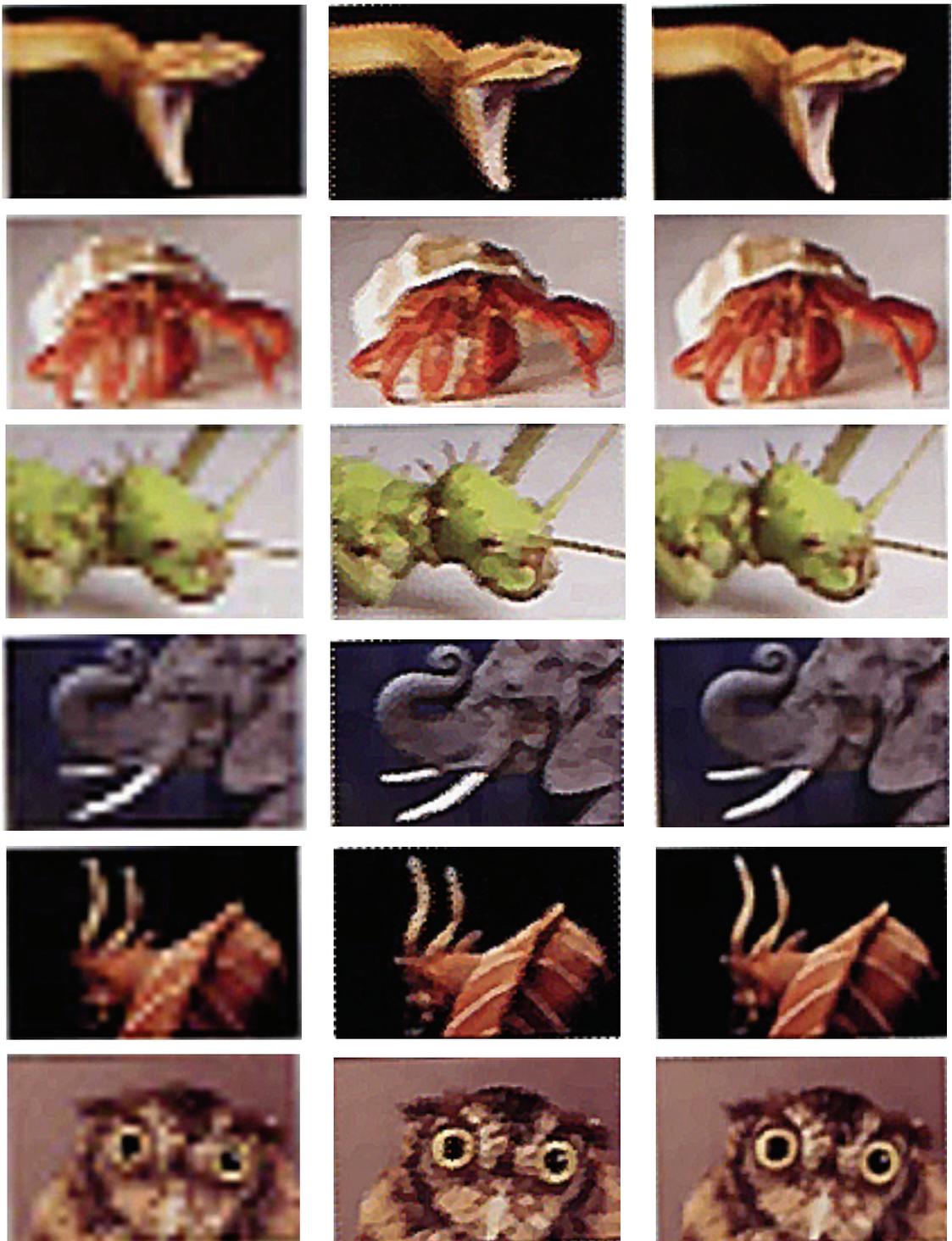
(b) Iterative L1 + sharpened (UM).
Comp. time = 15+ minutes.



(c) Blind SR + post-processed (TV+UM).
Comp. time = 13.57 sec.

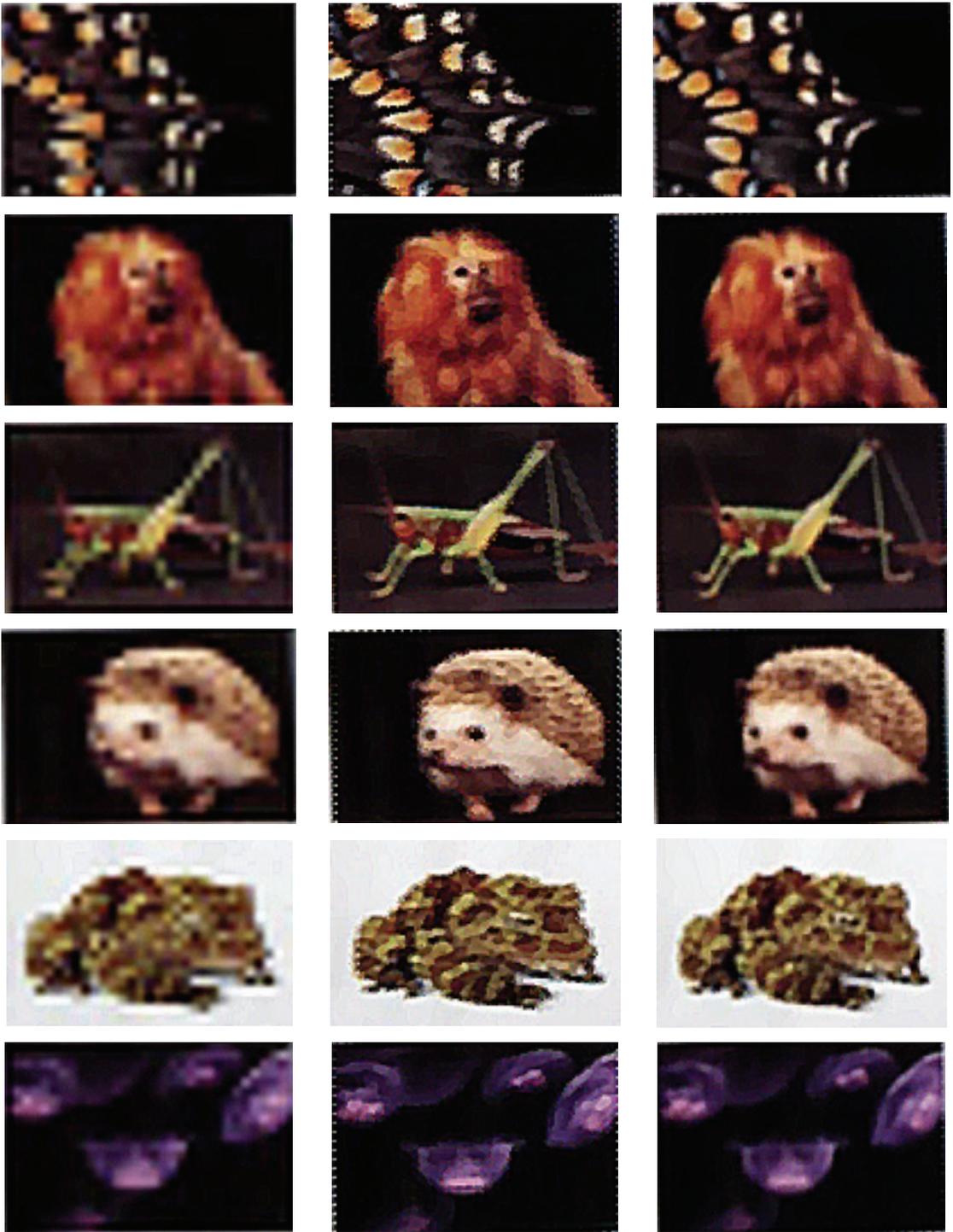


Figure 5.13: Rhythmic vibrations. Details part I. (# of LRs = 70).



(a) Bicubic interpolation + sharpened (UM). (b) Iterative L1 + sharpened (UM). (c) Blind SR + post-processed (TV+UM).

Figure 5.14: Rhythmic vibrations. Details part II. (# of LRs = 70).



(a) Bicubic interpolation + sharpened (UM).

(b) Iterative L1 + sharpened (UM).

(c) Blind SR + post-processed (TV+UM).

Figure 5.15: Rhythmic vibrations. Details part III. (# of LRs = 70).

Experiment 7: Atmospheric Turbulence

The original high resolution AVI sequence of the Moon, used for this experiment, is courtesy of Dr. Joseph M. Zawodny, NASA Langley Center. It was shot in coastal²² Virginia at angular (spatial) sampling of 0.34 arcsecond/pixel. The resolution (in terms of pixel density) almost met the diffraction limit at 1.7 pixels/Airy radius. The temporal resolution was 30 frames/second.

The sequence contains 1300 frames of size 768x1024, of which, we only used 100 frames²³. To obtain easily noticeable aliasing we downsampled them by $\downarrow 8 \times 8$ and $\downarrow 10 \times 10$ to obtain the primary and secondary sets of LR images, respectively. These were pre-denoised using PCA.

The first LR image from the primary set was resized ($\uparrow 4 \times 4$) using bicubic interpolation and then sharpened as shown in Figure 5.16 (a). Figure 5.16 (b) shows the sharpened²⁴ and median filtered super-resolved image corresponding to choosing one of the secondary LR images as a reference PPC, which is slightly better than the super-resolved image corresponding to estimating the reference PPC, shown in Figure 5.17 (b). This suggests that we should always obtain two estimates of the HR image corresponding to estimating the reference PPC and choosing a secondary LR image as reference PPC as well. Of course, the PCA pre-denoising step need not be repeated.

Finally, Figure 5.17 (a) shows the reconstructed HR image using the iterative L1 method²⁵, after sharpening. The aliasing and other artifacts are due to the fact that the warping effect is LSV and the motion estimation methods included in the software [49] can only handle the global motion case, not to mention that the randomness of the blur negatively affects the performance of motion estimation.

²² The effect of the atmospheric turbulence is larger at lower altitudes.

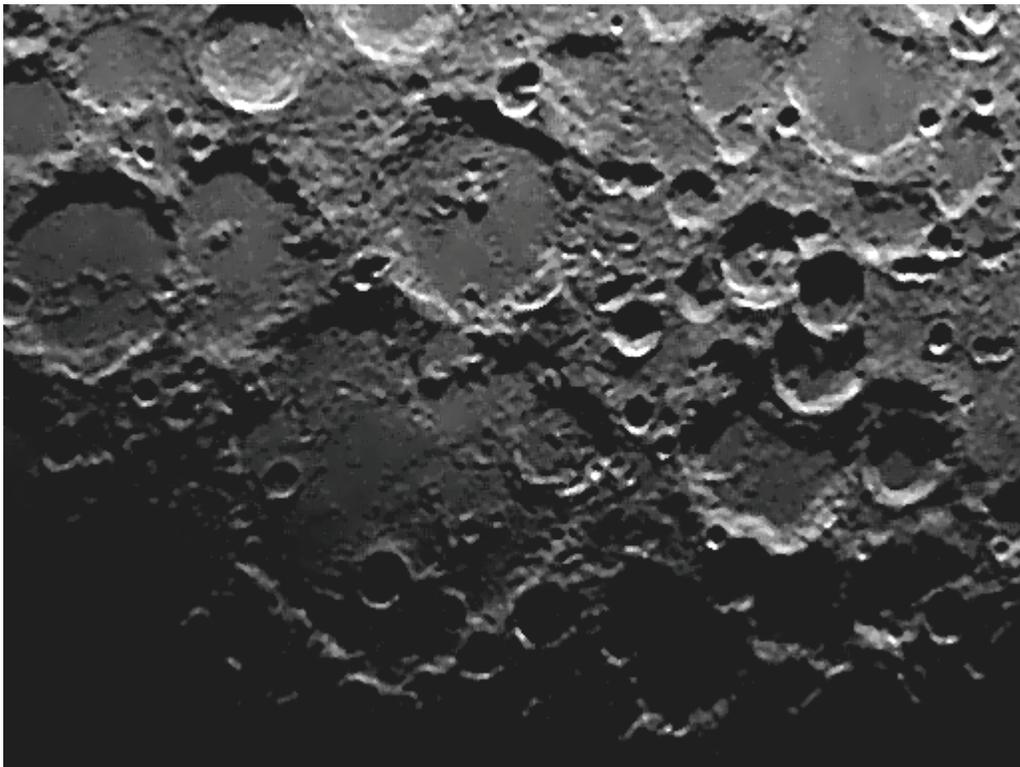
²³ We appended zeros to the HR frames to have dimensions of 800x1040, which are integer multiples of 80. Refer to the discussion related to equation (2.9).

²⁴ We used Photoshop's unsharp masking, instead of MATLAB's, for more deblurring freedom.

²⁵ Number of iterations was 20, regularization factor was 0.001, and the initial guess was the shift & add image.

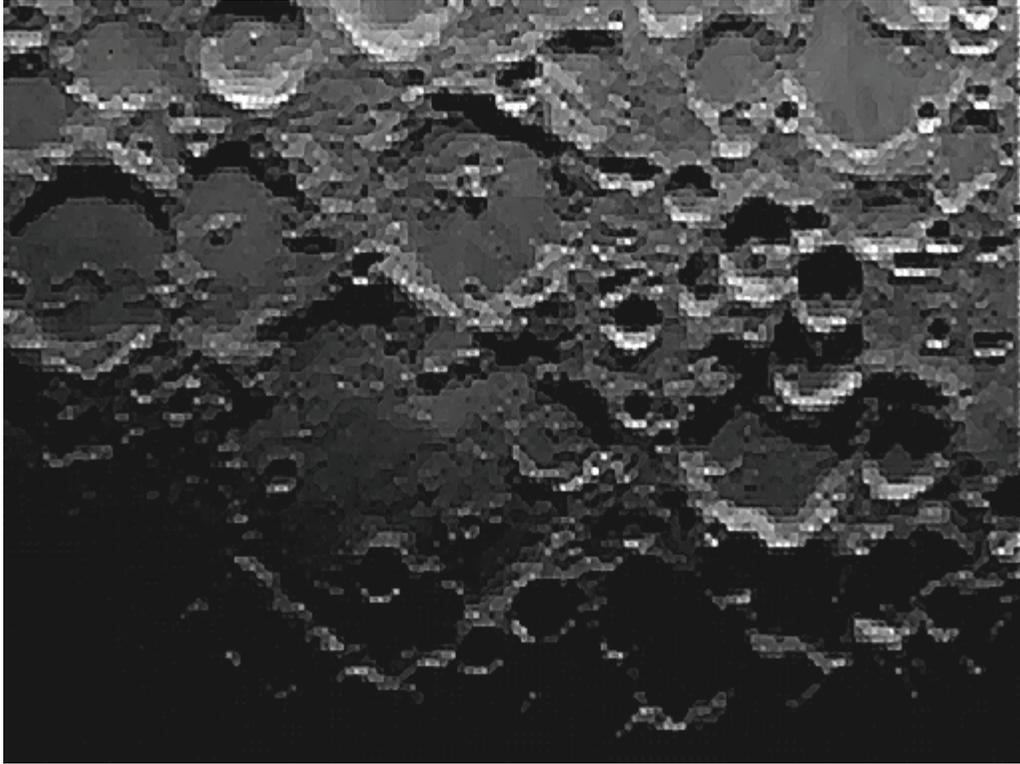


(a) Bicubic interpolation + sharpened. Comp. time = 0.83 sec.



(a) Blind SR + post-processed (UM+MD). A single sec. LR image was used as a ref. PPC. Comp. time = 9.31 sec.

Figure 5.16: Atmospheric turbulence. (# of LRs = 100).



(a) Iterative L1 + sharpened (UM).



(b) Blind SR + post-processed (UM+MD). The ref. PPC was estimated. Comp. time = 10.9 sec.

Figure 5.17: Atmospheric turbulence: Blind SR vs. Iterative L1. (# of LRs = 100).

CHAPTER VI

Summary and Future Work

6.1 Summary

Multiframe super-resolution is normally formulated as a large inverse problem where the degradation model parameters are assumed to be either known or reliably estimated. Hence, the primary objective of typical SR methods is to develop efficient and stable algorithms to tackle the huge size and ill-posedness of the problem. Consequently, robustness to model errors is characteristically a major concern and efficiency is always limited by the huge number of variables.

Instead of trying to parameterize, and then inverse the process that produced the degraded LR images, our SR method essentially reformulates the problem as a change of basis, where we postulate that the available set of LR images form a basis that can represent the polyphase components (PPCs) of the HR image.

Given the fact that the LR images and PPCs are both of the same resolution level and are both derived from the same signal (the HR image), this idea of the LR images forming a ‘LR basis’ for the PPCs seems rather intuitive. The *completeness* of the LR basis is dependent on the type (LSI vs. LSV) and extent (severity) of the distortion process.

Therefore, instead of solving for the pixels of the HR image, we estimate the expansion coefficients of the PPCs in terms of the LR basis, using portions (*sub* PPCs) of the PPCs. These *sub* PPCs are estimated using the property of sampling diversity with a simple hardware requirement of adding a secondary (lower resolution) sensor.

In effect, our proposed method veers away from the major limitations associated with typical model-based solution of the SR problem. Specifically, our SR method is fast, does not require any estimation of the degradation process and is robust in the sense that the only ‘model’ we use is in fact completely accurate: portraying *sub* PPCs as shifted and decimated versions of the PPCs. And besides the trivial hardware requirement, completeness of the LR basis is the only key assumption we make; the invalidity of which has only one consequence: the PPCs will be *partially* reconstructed.

Finally, in certain applications where typical multiframe SR performs poorly (e.g. in the case of random vibrations), our method not only provides a much faster solution, it actually benefits from the random nature of distortions.

6.2 Future Work

- Throughout this thesis, the noise was assumed to be uncorrelated (Gaussian). Although the *theoretical* PCA performance is independent of the data distribution, the empirical PCA is dependent on it. We would like to investigate other forms of PCA that can handle cases of correlated error and Laplacian noise. In addition, the post-processing step can benefit from more complex techniques, other than the simple unsharp masking, for example.
- In the case of color images, we assumed that the *primary* LR images are obtained using 3 CCD sensors. The *Bayer filter* is needed when only a single CCD sensor is used to capture the primary LR images, causing color artifacts. The effects (of using a Bayer filter) on the performance of our blind SR method, are yet to be addressed in future research.
- Using real-world distorted HR sequences of images, we obtained LR images by averaging the HR pixels to simulate the CCD integrating effect. Although very accurate, we would still like to avoid simulating the downsampling process. In the future, we would like to have a prototype camera built with a secondary imaging sensor as described in chapter II. Having a successful SR method with results based on 100% real data degradation process, including downsampling, can sway the industry towards building cameras with an additional (lower resolution) sensor since this will be beneficial even beyond the cost reduction resulting from avoiding using

larger (or denser) imaging chips, as there are always physical limits that can only be beaten using super-resolution techniques.

- A special application of interest is satellite imaging of the Earth. Driven by the success of our experiment involving super-resolving lunar images corrupted with random atmospheric distortion, we would like to pay special attention to super-resolving satellite images, which are also affected (to a lesser degree) by the atmosphere.
- Can we extend our method to handle the case of dynamic SR? We believe the answer could be yes, depending on the temporal resolution of the video sequence. To be more specific, we could use each secondary frame as a reference PPC, thus obtaining a sequence of SR images that are, in essence, HR versions of the secondary LR images. This, however, would probably require a temporal resolution high enough for a valid assumption of the rigidity of the scene within reasonably short time windows.
- An active field of research is learning-based super-resolution, where SR methods are designed to reconstruct a HR image from a single LR frame. The success of the reconstruction process is heavily dependent on the *training* set of images carefully chosen to be within the same class of the HR image. The best example of this is face hallucination [35, 36], where a HR face image can be reconstructed from its LR version, given a database of HR face images. As a future research direction, it is interesting to investigate whether such methods might benefit from the idea of applying the property of sampling diversity, where the single (distortion-free) LR frame plays the role of the reference PPC. In other words, instead of estimating the HR image directly from the LR image, estimating the PPCs using the LR image as a reference PPC, might be advantageous since signals at lower resolutions have more in common. That is to say, it might be easier to train a basis to reconstruct low resolution signals (PPCs) and as such, the sampling diversity idea could be extended to single frame SR and *without* the additional requirement of a secondary sensor.

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] S. C. Park, M. K. Park, and M. G. Kang, "Super-resolution image reconstruction: A technical overview," *IEEE SP Magazine*, pp. 21-36, 2003.
- [2] S. Chaudhuri, Ed., *Super-Resolution Imaging*, Kluwer, Norwell, MA, 2001.
- [3] T. S. Huang and R. Y. Tsai, "Multi-frame image restoration and registration," *Advances in Computer Vision and Image Process.*, vol. 1, pp. 317-339, 1984.
- [4] S. P. Kim, N. K. Bose, and H. M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframe," *IEEE Trans. ASSP*, vol. 38, pp. 1013-1027, 1990.
- [5] S. P. Kim and W. Y. Su, "Recursive high-resolution reconstruction of blurred multiframe images," *IEEE Trans. IP*, vol. 2, pp. 534-539, 1993.
- [6] S. H. Rhee and M. G. Kang, "Discrete cosine transform based regularized high-resolution image reconstruction algorithm," *Opt. Eng.*, vol. 38, no.8, pp. 1348-1356, 1999.
- [7] M. Elad and Y. Hel-Or, "A fast super-resolution reconstruction algorithm for pure translational motion and common space-invariant blur," *IEEE Trans. IP*, vol. 10, pp. 1187-1193, 2001.
- [8] M. Elad and A. Feuer, "Reconstruction of a single super-resolution image from several blurred, noisy, and undersampled measured images," *IEEE Trans. IP*, vol. 6, pp. 1646-1658, 1997.
- [9] D. Rajan and S. Chaudhuri, "Generation of super-resolution images from blurred observations using an MRF model," *J. Math. Imaging Vision*, vol. 16, pp. 5-15, 2002.
- [10] D. Rajan and S. Chaudhuri, "Simultaneous estimation of super-resolved intensity and depth maps from low resolution defocused observations of a scene," in *Proc. IEEE Int. Conf. Computer Vision*, Vancouver, Canada, 2001, pp. 113-118.

- [11] S. Chaudhuri and J. Manjunath, *Motion-free super-resolution*, Springer-Verlag, New York, 2005.
- [12] N. Nguyen, P. Milanfar, and G. Golub, "A computationally efficient superresolution image reconstruction algorithm," *IEEE Trans. IP*, vol. 10, pp. 573-583, 2001.
- [13] N. Nguyen, P. Milanfar, and G. Golub, "Efficient generalized cross-validation with applications to parametric image restoration and resolution enhancement," *IEEE Trans. IP*, vol. 10, pp. 1299-1308, 2001.
- [14] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. IP*, vol. 13, pp. 1327-1344, 2004.
- [15] R. R. Shultz, and R. L. Stevenson, "A Bayesian approach to image expansion for improved definition," *IEEE Trans. IP*, vol. 3, pp. 233-242, 1994.
- [16] H. Stark and P. Oskoui, "High resolution image recovery from image plane arrays, using convex projections," *J. Opt. Soc. Amer. A*, vol. 6, pp. 1715-1726, 1989.
- [17] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Proc.*, vol 53, pp. 231-239, 1991.
- [18] J. L. Barron, D. J. Fleet, and S. Beauchemin, "performance of optical flow techniques," *Int. J. Comput. Vision*, vol. 12, pp. 43-77, 1994.
- [19] L. Brown, "A survey of image registration techniques," *ACM Comput. Surv.*, vol. 24, pp. 325-376, 1992.
- [20] R. L. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*, Kluwer, New York, NY, 1991.
- [21] G. Harikumar, and Y. Bresler, "Perfect blind restoration of images blurred by multiple filters: Theory and effecient algorithms," *IEEE Trans. IP*, vol. 8, pp. 202-219, 1999.
- [22] G. H. Golub and C. F. Van Loan, *Matrix Computations: Third Edition*, Johns Hopkins University Press, Baltimore, MD, 1996.
- [23] S. Van Huffel and J. Vanderwalle, *The Total Least Squares Problem—Computational Aspects and Analysis*, SIAM, Philadelphia, PA, 1991.
- [24] R. D. Fierro, G. H. Golub, P. C. Hansen and D. P. O’Leary, "Regularization by truncated total least squares," *SIAM J. Sci. Comput.*, vol. 18, pp. 1223-1241, 1997.
- [25] C. R. Vogel, *Computational Methods for Inverse Problems*, SIAM, Philadelphia, PA, 2002.

- [26] R. C. Thompson, "Principal submatrices IX: interlacing inequalities for singular values of submatrices," *Linear Algebra Appl.*, vol. 5, pp. 1-12, 1972.
- [27] G. H. Golub, P. C. Hansen and D. P. O'Leary, "Tikhonov regularization and total least squares," *SIAM J. Matrix Anal. Appl.*, vol. 21, pp. 185-194, 1999.
- [28] G. H. Costa and J. C. M. Bermudez, "Are registration errors always bad for super-resolution?" *ICASSP*, vol.1, pp. 569-572, 2007.
- [29] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of Royal Statistical Society*, vol. 58, pp. 267-288, 1996.
- [30] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall PTR, Upper Saddle River, NJ, 1993.
- [31] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, NY, 2004.
- [32] I. T. Jolliffe, *Principal Component Analysis, Second Edition*, Springer-Verlag, New York, NY, 2002.
- [33] I. Markovsky and S. Van Huffel, "Overview of total least-squares methods," *Signal Processing*, vol. 87, pp. 2283-2302, 2007.
- [34] P. D. Wirawan and H. Maitre, "Multi-channel high resolution blind image restoration," *ICASSP*, vol. 6, pp. 3229-3232, 1999.
- [35] J. Yang, H. Tang, Y. Ma, and T. Huang, "Face hallucination via sparse coding," *ICIP*, pp. 1264-1267, 2008.
- [36] B. G. V. Kumar and R. Aravind, "A 2D model for face superresolution," *ICPR*, pp. 1-4, 2008.
- [37] J. Yang, J. Wright, Y. Ma, and T. Huang, "Image super-resolution as sparse representation of raw image patches," *CVPR*, pp. 1-8, 2008.
- [38] A. Stern, Y. Porat, A. Ben-Dor, and N. S. Kopeika, "Enhanced-resolution image restoration from a sequence of low-frequency vibrated images by use of convex projections," *Applied Optics*, vol. 40, pp. 4706-4715, 2001.
- [39] A. Stern, E. Kempner, A. Shukrun, and N. S. Kopeika, "Restoration and resolution enhancement of a single image from a vibration-distorted image sequence," *Opt. Eng.*, vol. 39, pp. 2451-2457, 2000.
- [40] Z. Zalevsky and D. Mendlovic, *Optical Superresolution*, Springer-Verlag, New York, 2005.

- [41] K. A. Parulski, L. J. D’Luna, B. L. Benamati, and P. R. Shelley, “High performance digital color video camera,” *J. Electron. Imaging*, vol. 1, pp. 35–45, 1992.
- [42] A. Zomet, A. Rav-Acha, and S. Peleg, “Robust super resolution,” *CVPR*, vol. 1, pp. 645–650, 2001.
- [43] B. C. Tom and A. K. Katsaggelos, “Reconstruction of a high-resolution image by simultaneous registration, restoration, and interpolation of low-resolution images,” *ICIP*, vol. 2, pp. 539-542, 1995.
- [44] R. C. Hardie, K. J. Barnard, and E. E. Armstrong, “Joint MAP registration and high-resolution image estimation using a sequence of undersampled images,” *IEEE Trans. IP*, vol. 6, pp. 1621-1633, 1997.
- [45] M. Ben-Ezra, A. Zomet, and S. Nayar, “Video super-resolution using controlled subpixel detector shifts,” *IEEE Trans. PAMI*, vol. 27, pp. 977–987, 2005.
- [46] N. R. Shah and A. Zakhor, “Resolution enhancement of color video sequences,” *IEEE Trans. IP*, vol. 8, pp. 879–885, June 1999.
- [47] B. C. Tom and A. Katsaggelos, “Resolution enhancement of monochrome and color video using motion compensation,” *IEEE Trans. IP*, vol. 10, pp. 278–287, 2001.
- [48] S. Farsiu, M. Elad, and P. Milanfar, “Multiframe demosaicing and super-resolution of color images,” *IEEE Trans. IP.*, vol. 15, pp. 141–159, 2006.
- [49] S. Farsiu, D. Robinson, and P. Milanfar, Resolution Enhancement Software. <http://users.soe.ucsc.edu/~milanfar/software/superresolution.html>
- [50] M. C. Roggemann, and B. Welsh, *Imaging Through Turbulence*, CRC Press, Boca Raton, Florida, 1996.
- [51] R. Paxman, T. Schulz, and J. Fienup, “Joint estimation of object and aberrations by using phase diversity,” *J. Opt. Soc. Amer. A*, vol. 9, pp. 1072–1085, 1992.
- [52] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*, American Math. Soc., Providence, RI, 1980.
- [53] L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D*, vol. 60, pp. 259–268, 1992.
- [54] C. Vogel and M. Oman, “Fast, robust total variation based reconstruction of noisy, blurred images,” *IEEE Trans. IP*, vol. 7, pp. 813–824, 1998.

- [55] S. Mika, B. Schölkopf, A.J. Smola, K. R. Müller, M. Scholz, and G. Rätsch, “Kernel PCA and De-Noising in Feature Spaces,” *Advances in Neural Information Processing Systems II*, M. S. Kearns, S. A. Solla, and D. A. Cohn, eds., pp. 536-542, MIT Press, Cambridge, MA, 1999.
- [56] N. A. Campbell, “Robust procedure in multivariate analysis 1: Robust covariance estimation,” *Applied Statistics*, vol. 29, pp. 231-237, 1980.
- [57] P. Rousseeuw and K. Van Driessen, “A fast algorithm for the Minimum Covariance Determinant estimator,” *Technometrics*, vol. 41, pp. 212–223, 1999.
- [58] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming (web page and software). <http://stanford.edu/~boyd/cvx>, June 2009.
- [59] M. Grant and S. Boyd, “Graph implementations for nonsmooth convex programs,” *Recent Advances in Learning and Control (a tribute to M. Vidyasagar)*, V. Blondel, S. Boyd, and H. Kimura, editors, pp. 95-110, *Lecture Notes in Control and Information Sciences*, Springer, 2008. http://stanford.edu/~boyd/graph_dcp.html.
- [60] K. C. Toh, M. J. Todd, and R. H. Tutuncu, “SDPT3 -- a Matlab software package for semidefinite programming,” *Optimization Methods and Software*, vol. 11, pp. 545-581, 1999.
- [61] R. H. Tutuncu, K. C. Toh, and M.J. Todd, “Solving semidefinite-quadratic-linear programs using SDPT3,” *Mathematical Programming Ser. B*, vol. 95, pp. 189-217, 2003.
- [62] V. Rokhlin, A. Szlam, and M. Tygert, “A randomized algorithm for principal component analysis,” *SIAM J. Matrix Anal. Appl.*, vol. 31, pp. 1100-1124, 2009.
- [63] A. Chambolle, “An algorithm for total variation minimization and applications,” *J. Math. Imaging and Vision*, vol. 20, pp. 89-97, 2004.
- [64] X. Bresson and T. F. Chan, “Fast minimization of the vectorial total variation norm and applications to color image processing,” CAM Report 07-25.
- [65] G. Gilboa, N. Sochen, and Y. Y. Zeevi, “Variational denoising of partly textured images by spatially varying constraints,” *IEEE Trans. IP*, vol. 15, pp. 2281-2289, 2006.
- [66] A. Chambolle and P. L. Lions, “Image recovery via total variation-based restoration,” *SIAM J. Sci. Comput.*, Vol. 20, pp. 1964-1977, 1999.
- [67] T. F. Chan and S. Esedoglu, “Aspects of total variation regularized L1 function approximation,” UCLA CAM Report 04-07, 2004.

- [68] T. Le, R. Chartrand and T. Asaki, "A variational approach to constructing images corrupted by poisson noise," *J. Math. Imaging and Vision*, vol. 27, pp. 257-263, 2007.
- [69] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. PAMI*, vol. 30, pp. 1672-1680, 2008.
- [70] P. D. Wentzell, D. T. Andrews, D. C. Hamilton, K. Faber, and B. R. Kowalski, "Maximum likelihood principal component analysis," *J. Chemometrics*, vol. 11, pp. 339-366, 1997.
- [71] M. Schuermans, I. Markovsky, P. Wentzell, and S. Van Huffel, "On the equivalence between total least squares and maximum likelihood PCA," *Anal. Chim. Acta*, vol. 544, pp. 254-267, 2005.
- [72] J. S. Lim, *Two-Dimensional Signal and Image Processing*, Prentice Hall, Englewood Cliffs, NJ, 1990.
- [73] M. S. Alam, J. G. Bognar, R. C. Hardie, and B. J. Yasuda, "Infrared image registration and high-resolution reconstruction using multiple translationally shifted aliased video frames," *IEEE Trans. IM*, vol. 49, pp. 915-923, 2000.