Blind image superresolution using subspace methods

Superrésolution aveugle d'images par la méthode des sous-espaces

Muriel Gastaud
Saïd Ladjal
Henri Maître

2008D003

Mars 2008

Département Traitement du Signal et des Images
Groupe TII : Traitement et Interprétation des Images

Ce document a servi de rapport d'activité semestriel, pour la période de septembre 2006 à février 2008, pour le contrat N°040-Type PG de bourse post-doctorale de la fondation EADS.

BLIND IMAGE SUPERRESOLUTION USING SUBSPACE METHODS

Superrésolution aveugle d'images par la méthode des sous-espaces

Muriel Gastaud, Saïd Ladjal et Henri Maître

prénom.nom@telecom-paristech.fr

Institut TELECOM, TELECOM-ParisTech, LTCI CNRS

46 rue Barrault, 75013 Paris, France

Abstract :

Subspace methods are a powerful tool to recover unknown filters by looking at the second order statistics of various signals originating from the same source (also called a SIMO problem). An extension to the multiple source case is also possible and has been investigated in the literature.

In this report, we show how the blind superresolution problem can be solved by this tool. We first present the problem of superresolution as a multiple input multiple output (MIMO) one. We show that the subspace method can not be used, as is, to recover the filters affecting each image, and we present two possible solutions, based on the statistical characteristics of the images to solve this problem. Experiments are shown which validate these ideas.

Résumé:

Les méthodes de sous-espaces sont un outil puissant d'identification aveugle des filtres par l'étude de statistiques du second ordre de plusieurs sorties issues d'une même source (problème dit SIMO). L'extension de ce problème au cas de sources multiples peut être envisagée, et a été développée dans la littérature.

Dans ce rapport, nous montrons comment ces méthodes permettent de résoudre le problème de superrésolution aveugle. Nous présentons tout d'abord le problème de superrésolution comme un problème à entrées multiples et à sorties multiples (MIMO). Nous montrons que la méthode de sous-espace ne peut être utilisée seule pour retrouver les filtres affectant chaque image, et nous proposons deux solutions possibles utilisant les propriétés statistiques des images pour résoudre le problème. Nous présentons des résultats expérimentaux qui valident notre approche.

Contents

Chapter 1

Introduction

Blind superresolution is the task of estimating a high-resolution restored image from a set of low-resolution images, blurred by unknown filters and noisy. We propose a blind superresolution algorithm in three steps: first a registration step, then a filter identification step using subspace methods, and finally a restoration step : a regularized inversion of the observed images.
The superresolution algorithm first requires a registration step when the set of frames comes from a video with camera motion, or from several sensors with different perspectives. The registration step consists in mapping the observed frames into the coordinate system of a reference one.
Then the filters are identfied using subspace methods. The subspace method has been introduced by [1] and further investigate in a multitude of papers [2, 3, 4, 5]. The idea of the method is to observe multiple outputs of various unknown filters having all the same input (namely SIMO system). In this case, the second order statistics of the received signals carry enough information to allow the recovery of the filters and furthermore the recovery of the original signal.
The main application that authors had in mind was to conceive wireless protocols in a varying environment, in which no training sequence have to be transmitted. Indeed, in such an environment, the filters that affect the signal can change and have to be re-learned very often. Being able to learn them without the use of a training signal would be a great asset and could save an important amount of bandwidth.
Extensions to the case where a multitude of signals are transmitted through the same channel (namely MIMO systems) have been investigated [2, 3]. In this late approach, a crucial step is the use of a source separation technique.

We investigate the possibility of using the subspace method in the context of image superresolution. We extend subspace methods to image superresolution : the observed low-resolution images are the outputs of a high-resolution original image acquired through various filters, and then subsampled. The problem can be modeled as a multiple-input multiple-output system (MIMO).

Subspace methods applied to MIMO systems need an additional source separation step. Source separation methods previously investigated [2, 3] are not suitable in our case, as the inputs of the system are strongly correlated. We propose to minimize a functional to evaluate the mixing matrix.

This report is divided as follows:
Chapter 2 presents the registration step.
Chapter 3 presents the subspace method for 1D signal and images, in order to provide the reader with a self contained overview.
Chapter 4 states the problem of superresolution as a multiple input multiple output (MIMO) one, in which the multiple inputs are the various subsampled versions of the image (they differ by a translation). This presentation allows us to understand that:

- The separation of sources is impossible in the case of superresolution because the sources are very correlated with each other and have exactly the same statistics (Section 4.2).

- The subspace method provides us with a mixture of the actual filters. Therefore, we have to implement a method to unmix and recover the actual filters. In the same time, the subspace method allows us to restrain the search for the filters to a relatively small affine space. In Section 4.3, we introduce our method to disambiguate the results of the subspace method and recover the actual filters.

In Chapter 5, we provide the restoration step based on the minimization of a functional including a data driven term and a regularization one.
Finally, Chapter 6 presents experimental results for both filters recovery and image restoration based on this recovery.

Chapter 2

Frame registration

The superresolution algorithm first requires a registration step when the different frames comes from a video with camera motion, or from several sensors with different perspectives. The registration step consists in mapping the observed frames into the coordinate system of a reference one, usually the first or the middle one. In order to register the frames, we need to estimate the motion between each frame and a frame of reference, more precisely, the camera motion between these frames, that is to say the dominant motion (small moving objects are excluded). We choose to model the camera motion by an affine parametric model, as this is the best trade-off between accuracy and complexity. This model is robustly estimated from a pre-estimated motion vector field.

2.1 Camera motion estimation

The camera motion is defined as the main motion between two frames. When a video contains moving objects, these objects are assumed to be small enough so that the definition is still valid.

Motion estimation between two frames depends mainly on two choices :

- the motion model : dense or parametric,
- the motion estimation method : feature-based or direct.

A parametric motion model is well adapted for estimating a camera motion. We choose a 6 parameter motion model, as it is a good tradeoff between complexity and accuracy [6].

We propose a camera motion estimation between two frames in two steps:

- first a block-matching algorithm provides a field of motion vectors between the frames,
- and then a minimization algorithm searches for the parametric model fitting at best the set of motion vectors.

2.1.1 Motion estimation : block-matching

A block-matching algorithm proceeds as follows :

- the frames are first partitioned into blocks,
- then, for each block of one frame, we look for the matching block in the other frame, i.e the block which fits at best, following a correlation criterion (see figure 2.1).

Among all the existing block-matching methods, let us cite the exhaustive block-matching, which compares the block in the first frame with all the other blocks in the research window of the second frame. This exhaustive research ensures to find the global minimum of the correlation criterion in the research window, but for a high computational cost.
Other block-matching methods try to reach the best tradeoff between computational cost and accuracy of the estimation. The precursor three-step algorithm [7] looks for a local minimum in a 8-neighborhood far from 4 pixels, then looks for the local minimum around this first local minimum in a 8-neighborhood far from 2 pixels, and finally in a 8-neighborhood far from 1 pixel.
Zhu *et al.* [8] propose two iterative algorithms using different research patterns, for example diamond patterns, instead of squared ones as in the three-step search. These algorithms try to improve the accuracy of the estimation over the three-step search, and to reduce the computational cost in the same time.
However, although these methods are very attractive due to their low computational cost, they can not ensure to find the

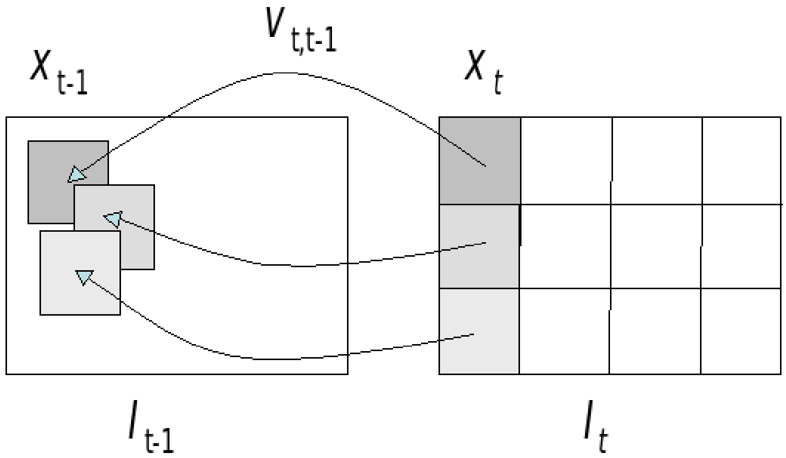


Figure 2.1: Backward motion estimation

global minimum and are therefore less accurate than an exhaustive search.

The accuracy of a block-matching algorithm depends also on several parameters :

- the correlation criterion : we use the zero-mean normalized sum of squared differences (ZNSSD), as this criterion is robust to affine variations of intensity;

$$znssd(\mathbf{v}_{t,t-1}) = \frac{\sum_{\mathbf{x}_t \in B_t}(I_{t-1}(\mathbf{x}_t + \mathbf{v}_{t,t-1}) - \bar{I}_{t-1}(\mathbf{x}_t + \mathbf{v}_{t,t-1}) - I_t(\mathbf{x}_t) + \bar{I}_t(\mathbf{x}_t))^2}{\sqrt{\sum_{\mathbf{x}_t \in B_t}(I_{t-1}(\mathbf{x}_t + \mathbf{v}_{t,t-1}) - \bar{I}_{t-1}(\mathbf{x}_t + \mathbf{v}_{t,t-1}))^2}\sqrt{\sum_{\mathbf{x}_t \in B_t}(I_t(\mathbf{x}_t) - \bar{I}_t(\mathbf{x}_t))^2}} \tag{2.1}$$

  where $\bar{I}(\mathbf{x})$ is the mean value of I over block B.

- the size of the search window : all block-matching methods look for the matching block in a limited window of research, defined by (or defining) a maximal displacement. The search window has to be largest than the camera displacement, but smaller as possible, as an increase of the search window size increases dramatically the computational cost;

- the size of the blocks : usually blocks are 8×8 or 16×16;

- the interpolation : to provide a sub-pixel accuracy, the images are interpolated.

The block-matching algorithm provides a set of vectors, one for each block of the image, and the next step consists in estimating the parameters of the motion model which best fit these motion vectors.

2.1.2 Robust camera motion model

The camera motion between the current frame, denoted I_t, and the previous one, denoted I_{t-1}, is represented by an affine model :

$$A_{t,t-1} = \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \quad \text{and} \quad T_{t,t-1} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \tag{2.2}$$

Let $\mathbf{x}_t$ denotes a pixel in frame I_t and $\mathbf{x}_{t-1}$ its matching pixel in frame I_{t-1}, provided by the block-matching algorithm, $\mathbf{x}_t$ is, for example, the center of a block of I_t.
The motion vector between these two pixels is denoted $\mathbf{v}_{t,t-1}$ and we have :

$$\mathbf{x}_{t-1} = \mathbf{x}_t + \mathbf{v}_{t,t-1} \tag{2.3}$$

The camera motion model must fit at best all the different displacements of each block belonging to frame I_t. We aim to minimize the difference between the image of the pixel $\mathbf{x}_t$ following the motion model, i.e. $A_{t,t-1}\mathbf{x}_t + T_{t,t-1}$, and its matching pixel $\mathbf{x}_{t-1}$ estimated by block-matching, for each $\mathbf{x}_t$ center of a block of I_t.
Let us note $\mathbf{x}_t^n$ the center of the n^{th} block of I_t, where $n = 1 : N$ if I_t is partitioned in N blocks.

The estimation of the camera motion model can be modeled as the minimization of the following criterion:

$$\sum_{n=1}^{N} \|\mathbf{x}_{t-1}^{n} - A_{t,t-1}\mathbf{x}_{t}^{n} - T_{t,t-1}\|^2 \tag{2.4}$$

As we use the quadratic norm, the solution of this problem is well-known .

Indeed, if the difference is written:

$$\mathbf{x}_{t-1}^{n} - A_{t,t-1}\mathbf{x}_{t}^{n} - T_{t,t-1} = \mathbf{x}_{t-1}^{n} - M_t^n \mathbf{p} \tag{2.5}$$

where

$$\mathbf{p} = (a_1\ a_2\ a_3\ a_4\ b_1\ b_2)^T \tag{2.6}$$

is the vector of the motion model parameters, and

$$M_t^n = \begin{pmatrix} (\mathbf{x}_t^n)_i & (\mathbf{x}_t^n)_j & 0 & 0 & 1 & 0 \\ 0 & 0 & (\mathbf{x}_t^n)_i & (\mathbf{x}_t^n)_j & 0 & 1 \end{pmatrix} \tag{2.7}$$

where $\mathbf{x}_t^n = ((\mathbf{x}_t^n)_i, (\mathbf{x}_t^n)_j)^T$, for $n = 1 : N$,

$$\sum_{n=1}^{N} \|\mathbf{x}_{t-1}^{n} - A_{t,t-1}\mathbf{x}_{t}^{n} - T_{t,t-1}\|^2 = \sum_{n=1}^{N} \|\mathbf{x}_{t-1}^{n} - M_t^n \mathbf{p}\|^2 \tag{2.8}$$

Minimization of equation (2.4) is equivalent to minimization of:

$$\|X - M\mathbf{p}\|^2 \tag{2.9}$$

where

$$X = \begin{pmatrix} \mathbf{x}_t^1 \\ \vdots \\ \mathbf{x}_t^N \end{pmatrix} \quad \text{and} \quad M = \begin{pmatrix} M_t^1 \\ \vdots \\ M_t^N \end{pmatrix} \tag{2.10}$$

the solution of equation (2.4) is given by :

$$p = (M^T M)^{-1} M^T X \tag{2.11}$$

However, such an estimator is not robust to errors due to moving objects, or to matching outliers.

To improve the accuracy of the motion model estimation, we use an M-estimator φ. The camera motion model minimizes the following criterion :

$$\sum_{\mathbf{x}_t \in I_t} \varphi(\|\mathbf{x}_{t-1} - A_{t,t-1}\mathbf{x}_t - T_{t,t-1}\|) \tag{2.12}$$

Thanks to M-estimators, the effects of an high dissimilarity $\|\mathbf{x}_{t-1} - A_{t,t-1}\mathbf{x}_t - T_{t,t-1}\|$ on the motion model estimation are less important. A high dissimilarity is mainly due to a a moving object or a wrong matching, these pixels are so-called outliers. The less the outliers are taken into account, the more robust is the estimation of the motion model. By allocating small weights to the outliers, M-estimators improve the model motion estimation and make it more robust to errors.
The main drawback of such a weighting function is that the differentiation of criterion (2.12) is no more straightforward. We use the following theorem, demonstrated by Charbonnier *et al.*[9], to provide the minimization of criterion (2.12) :

Theorem 2.1.1 *Let φ be a potential function that satisfies :*

1. *φ continuously differentiable.*
2. *$\varphi(r) \geq 0\ \ \forall r$ with $\varphi(0) = 0$.*
3. *φ increasing on $\Re^+$.*
4. *$\varphi(r) = \varphi(-r)$.*
5. *$\varphi'(r)/2r$ continuous and strictly decreasing on $[0, +\infty)$.*

6. $\lim_{r\to+\infty} \frac{\varphi'(r)}{2r} = 0$.

7. $\lim_{r\to 0^+} \frac{\varphi'(r)}{2r} = M, \quad$ *where* $\quad 0 < M < +\infty$.

Then :

1. *there exists a strictly convex and decreasing function* ψ, $\psi : (0, M] \to [0, \beta)$ *where :*

$$\beta = \lim_{r\to+\infty} \left(\varphi(r) - r^2 \frac{\varphi'(r)}{2r} \right)$$

such that :

$$\varphi(r) = \inf_{0<w\leq M} (wr^2 + \psi(w))$$

2. *For every fixed* r*, the value* w_r *for which the minimum is reached, i.e. such that :*

$$\inf_{0<w\leq M} (wr^2 + \psi(w)) = (w_r r^2 + \psi(w_r)),$$

is unique and given by :

$$w_r = \frac{\varphi'(r)}{2r}$$

You can find a demonstration of this theorem in [9, 10].

Thanks to theorem 2.1.1 we know that minimize criterion (2.12) is equivalent to minimize the semi-quadratic criterion :

$$\sum_{n=1:N} (w_n \|\mathbf{x}_{t-1}^n - A_{t,t-1}\mathbf{x}_t^n - T_{t,t-1}\|^2 + \psi(w_n)) \tag{2.13}$$

where

$$w_n = \frac{\varphi'(\|\mathbf{x}_{t-1}^n - A_{t,t-1}\mathbf{x}_t^n - T_{t,t-1}\|)}{2 * \|\mathbf{x}_{t-1}^n - A_{t,t-1}\mathbf{x}_t^n - T_{t,t-1}\|} \tag{2.14}$$

The minimization algorithm consists of iteratively estimating :

1. Motion model parameters $A_{t,t-1}$ and $T_{t,t-1}$ by minimizing equation (2.13), for a given w_n
   The optimal parameter vector is given by:

$$p = (M^T W M)^{-1} M^T W X \quad \text{where} \quad W = diag(w_n)_{n=1:N} \tag{2.15}$$

2. Weights w_n following (2.14) for given model parameters

We choose Geman and McClure's [11] potential function :

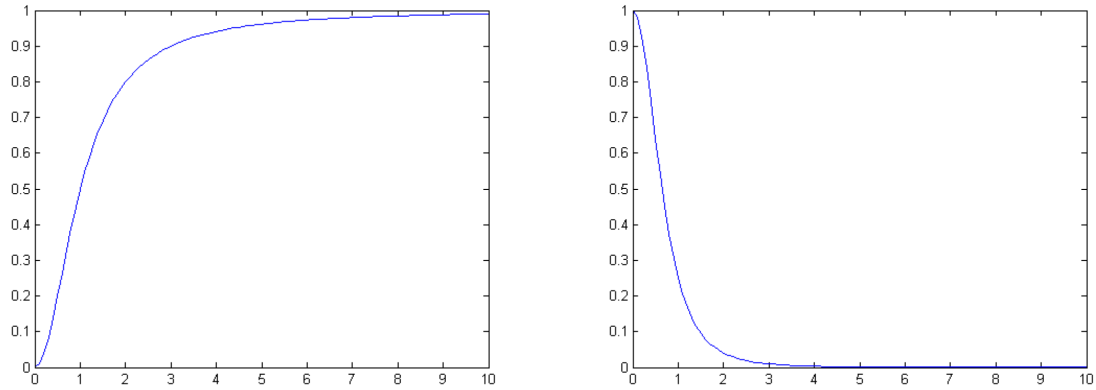
$$\varphi(r) = \frac{r^2}{\sigma^2 + r^2}$$

The weighting function associated to Geman and McClure's potential function is

$$w(r) = \frac{\varphi'(r)}{2r} = \frac{\sigma^2}{(\sigma^2 + r^2)^2}$$

See figure 2.2 for an illustration of Geman and McClure's potential and weighting functions.
Let us note that Geman and McClure's function satisfies the conditions of theorem 2.1.1.

The camera motion estimation has been developed between frames I_t and I_{t-1}, but it can be applied as well between the current frame and a reference one.



(a) $potential function$ (b) $weighting function$

Figure 2.2: Geman and McClure's weighting function for $\sigma^2 = 1$

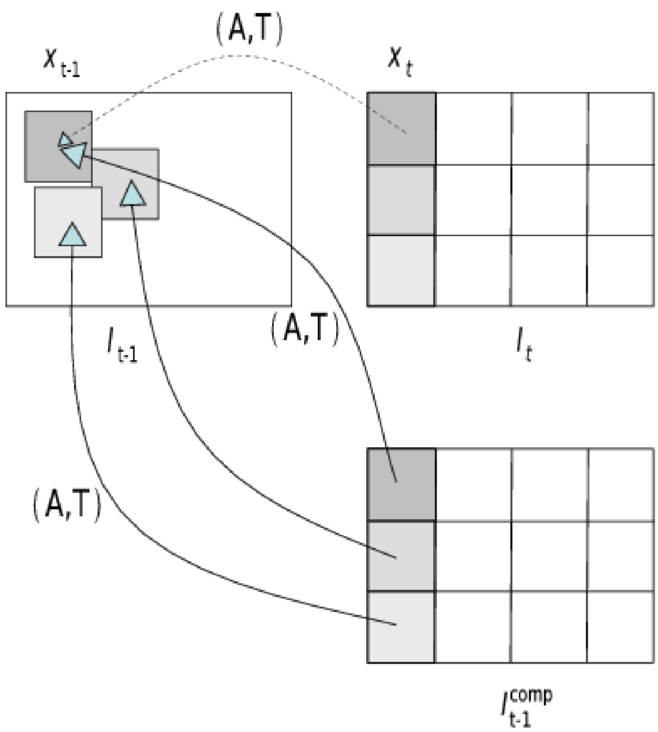


Figure 2.3: *backward* registration between frames I_t and I_{t-1}

2.2 Camera motion compensation

The registration of the current frame consists in compensating the camera motion i.e. in mapping the current frame into the coordinate system of a reference one. This can be done in a *forward* or *backward* way.

In a forward compensation, we estimate, for each pixel $\mathbf{x}_{t-1}$ in frame I_{t-1} its location $\mathbf{x}_{comp}$ in registered frame I_{t-1}^{comp} :

$$\mathbf{x}_{comp} = A_{t-1,t}\mathbf{x}_{t-1} + T_{t-1,n} \tag{2.16}$$

where $A_{t-1,t}$ and $T_{t-1,t}$ denote the camera motion model between frames I_{t-1} and I_t.

We assign to pixel $\mathbf{x}_{comp}$ the value of its matching pixel $\mathbf{x}_{t-1}$:

$$I_{t-1}^{comp}(\mathbf{x}_{comp} = A_{t-1,t}\mathbf{x}_{t-1} + T_{t-1,t}) = I_{t-1}(\mathbf{x}_{t-1}) \tag{2.17}$$

The main drawback of the forward compensation, is that it does not ensure that we assign a value at each pixel of the registered frame, and the registered frame may contain holes. On the other hand, several pixels of frame I_{t-1} may be registered in the same location of I_{t-1}^{comp}, leading to overlapping areas.

In a backward compensation, we assign, at each pixel $\mathbf{x}_{comp}$ in frame I_{t-1}^{comp} the value of its matching pixel $\mathbf{x}_{t-1}$ in frame I_{t-1}, pixel whose coordinates are given by :

$$\mathbf{x}_{t-1} = A_{t,t-1}\mathbf{x}_{comp} + T_{t,t-1} \tag{2.18}$$

where $A_{t,t-1}$ and $T_{t,t-1}$ denote the camera motion model between I_t and I_{t-1}.

We deduce that:

$$I_{t-1}^{comp}(\mathbf{x}_{comp}) = I_{t-1}(\mathbf{x}_{t-1} = A_{t,t-1}\mathbf{x}_{comp} + T_{t,t-1}) \tag{2.19}$$

In a backward compensation, we ensure that each pixel of the registered frame has a value, an this value is unique. However, two pixels of the registered frame may have the same matching pixel in I_{t-1}, resulting in redundancies.
Figure 2.3 shows a backward registration, as implemented in our algorithm.

The registration algorithm has been developed between the frames I_t and I_{t-1}, but it can be implemented as well between frames I_t and I_{ref}.

Chapter 3

Filter identification : the subspace method

In this chapter, we present the subspace method such as developed by [4] for 1D signals. This method, first introduced by [1], considers multi-output systems. This allows the use of second order statistics of the outputs, instead of higher order statistics, to identify blindly the filters. This method, under some mild assumptions, estimates the noise and signal subspaces from the eigenvalue decomposition of the autocorrelation matrix of the outputs, and exploits the orthogonality between these subspaces to identify the filter coefficients.
First, we state the problem for 1D signals and extend the formulation to images, then, we develop the subspace method for images. We refer the reader to [4] for a presentation of the subspace method for 1D signals.

3.1 Problem statement

3.1.1 1D signals

Let us consider L signals X^l as noisy outputs of an unknown system driven by an unknown input D. We aimed to identify the system function H blindly, i.e. using only the outputs of the system.
The outputs are described by a convolution model:

$$X^l(.) = H^l(.) * D(.) + B(.) \tag{3.1}$$

where $*$ denotes the convolution operator, and B is a white zero-mean noise.

Using a matrix formulation, we can write :

$$X = \mathcal{H}D + B \tag{3.2}$$

where

- X stacks the L output signals of size $(N, 1)$:

$$X = \begin{pmatrix} X^1 \\ \vdots \\ X^L \end{pmatrix} = \begin{pmatrix} x_0^1 & \dots & x_{N-1}^1 & \dots & x_0^L & \dots & x_{N-1}^L \end{pmatrix}^T \tag{3.3}$$

- $\mathcal{H}$ stacks the L Toeplitz matrices $\mathcal{H}^l$:

$$\mathcal{H} = \begin{pmatrix} \mathcal{H}^1 \\ \vdots \\ \mathcal{H}^L \end{pmatrix} \qquad (LN, N+M-1) \tag{3.4}$$

  where $\mathcal{H}^l$, for $l = 1$ to L, is the filtering matrix associated to filter H^l of size $(M, 1)$:

$$H^l = [h_0^l \dots h_{M-1}^l]^T \tag{3.5}$$

  i.e. :

$$\mathcal{H}^l = \begin{pmatrix} h_0^l & \dots & h_{M-1}^l & 0 & \\ & \ddots & & \ddots & \\ 0 & & h_0^l & \dots & h_{M-1}^l \end{pmatrix} \qquad (N, N+M-1) \tag{3.6}$$

- D is the unknown original signal of size $(N + M - 1, 1)$,
- and B is a white zero-mean noise, assumed to be uncorrelated with D.

3.1.2 Images

Let us consider L blurred images acquired by multiple cameras, or by a single camera through different conditions (focus changes, camera motion...).

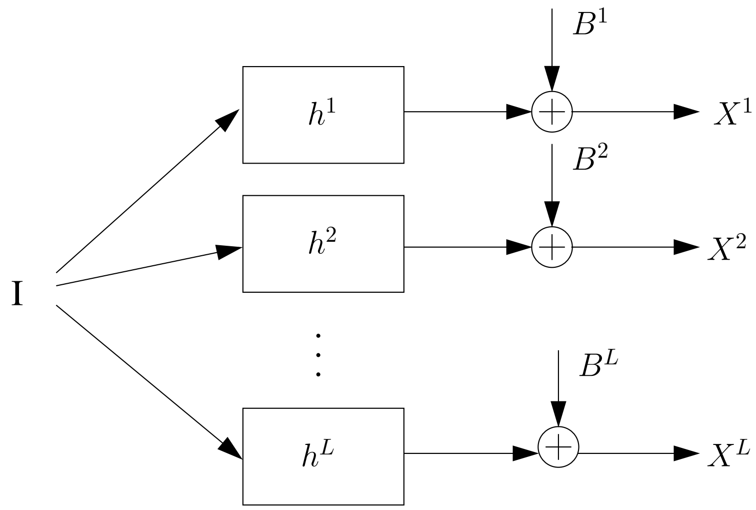


Figure 3.1: Model of acquisition for single-input multiple-output system

The L observed images are modeled as noisy outputs of a FIR system $\mathcal{H}$ driven by an input original image D :

$$X = \mathcal{H}D + B \tag{3.7}$$

where :

- X stacks L observed images X^l, for $l = 1$ to L, more precisely, a vectorized formulation of a processing windowed area, of size $(N_y\ ,\ N_x)$, extracted from the observed images :

$$X^l = [x^l(N_y - 1, N_x - 1)\ x^l(N_y - 2, N_x - 1)\ \cdots\ x^l(0,0)]^T \tag{3.8}$$

- D is a vectorized formulation of the related windowed area of the original image of size $(N_y + M_y - 1, N_x - M_x - 1)$,

$$D = [d(N_y + M_y - 2, N_x + M_x - 2) \cdots\ d(0,0)]^T \tag{3.9}$$

- $\mathcal{H}$ stacks L block-Toeplitz filtering matrices $\mathcal{H}^l$ associated with each filters H^l of size (M_y, M_x)

$$H^l = \begin{pmatrix} h^l(0,0) & \dots & h^l(0, M_x - 1) \\ \vdots & & \vdots \\ h^l(M_y - 1, 0) & \dots & h^l(M_y - 1, M_x - 1) \end{pmatrix} \tag{3.10}$$

i.e. :

$$\mathcal{H}^l = \begin{pmatrix} \mathcal{H}^l_0 & \cdots & \mathcal{H}^l_{M_x - 1} & 0 \\ \ddots & & \ddots & \\ 0 & \mathcal{H}^l_0 & \cdots & \mathcal{H}^l_{M_x - 1} \end{pmatrix} \tag{3.11}$$

where $\mathcal{H}^l_j$ is a Toeplitz matrix of size $(N_y\ ,\ N_y + M_y - 1)$ associated to the j^{th} column of H^l :

$$\mathcal{H}^l_j = \begin{pmatrix} h^l(0,j) & \dots & h^l(M_y - 1, j) & & 0 \\ \ddots & & & \ddots & \\ 0 & h^l(0,j) & & \dots & h^l(M_y - 1, j) \end{pmatrix} \tag{3.12}$$

$\mathcal{H}^l$ contains N_x rows of blocks and $N_x + M_x - 1$ columns of blocks of size $(N_y\ ,\ N_y + M_y - 1)$. $\mathcal{H}$ is of size $(LN_yN_x\ ,\ (N_y + M_y - 1)(N_x + M_x - 1))$,

- and B is a white zero-mean noise, assumed to be uncorrelated with D.

3.2 The subspace method for SIMO systems

Let $\mathbb{R}_X$ denotes the autocorrelation matrix of outputs X:

$$\mathbb{R}_X = E(XX^T) \tag{3.13}$$

where E denotes the expectation operator. $\mathbb{R}_X$ is of size $(LN_xN_y\ ,\ LN_xN_y)$.

From equation (3.2) we deduce that:

$$\mathbb{R}_X = \mathcal{H}\mathbb{R}_D\mathcal{H}^T + \mathbb{R}_B \tag{3.14}$$

where $\mathbb{R}_D$ and $\mathbb{R}_B$ denote respectively the autocorrelation matrices of the input D and of the noise B. We recall that the noise is assumed to be uncorrelated with the input.

From now on, we make two assumptions:

1. $\mathcal{H}$ is full column rank, a necessary condition is :

$$LN_yN_x > (N_x + M_x - 1)(N_y + M_y - 1) \tag{3.15}$$

2. and $\mathbb{R}_D$ is full rank.

We deduce from equation (3.14) and thanks to these assumptions, that the signal part of autocorrelation matrix $\mathbb{R}_X$, i.e. $\mathcal{H}\mathbb{R}_D\mathcal{H}^T$, has rank

$$d_H = (N_x + M_x - 1)(N_y + M_y - 1) \tag{3.16}$$

Through an eigenvalue decomposition of $\mathbb{R}_X$, we obtain a subspace decomposition between the signal and noise subspaces:

- The signal subspace is spanned by the eigenvectors associated with the d_H largest eigenvalues of $\mathbb{R}_X$
- The noise subspace, its orthogonal complement, is spanned by the eigenvectors associated with the $LN_xN_y - d_H$ smallest eigenvalues of $\mathbb{R}_X$.

The signal subspace is also the subspace spanned by the columns of the filtering matrix $\mathcal{H}$.

By orthogonality between signal and noise subspaces, we deduce that each vector of the noise subspace is orthogonal to each column of the filtering matrix.
Let G_i denotes an eigenvector associated with one of the $LN_xN_y - d_H$ smallest eigenvalues of the matrix $\mathbb{R}_X$. The orthogonality condition can be formulated as, for $i = 0 : LN_xN_y - d_H - 1$:

$$\underset{(1,LN_yN_x)(LN_yN_x,d_H)}{G_i^T\mathcal{H}} = \mathbf{0}_{(1,d_H)} \tag{3.17}$$

where $\mathbf{0}_{(1,d_H)}$ is a null vector of size $(1, d_H)$.

Since we have only an estimate of the autocorrelation matrix, the orthogonality condition is solved using a least square method. This leads to the minimization of the quadratic form:

$$q(\mathcal{H}) = \sum_{i=0}^{LN_xN_y-d_H-1} |G_i^T\mathcal{H}|^2 \tag{3.18}$$

Thanks to the following structural lemma, we provide an expression of the quadratic form in terms of the filter coefficients instead of the filtering matrix :

Lemma 1 :

$$G_i^T\mathcal{H} = H^T\mathcal{G}_i \tag{3.19}$$

You can find a proof of this lemma in [4].

In this expression, matrix $\mathcal{G}_i$, for $i = 0 : LN_xN_y - d_H - 1$, denotes a matrix of size $(LM_yM_x\ ,\ d_H)$. This matrix is constructed as follows:

- Each eigenvector G_i, for $i = 0$ to $LN_xN_y - d_H - 1$ is partitioned into L vectors G_i^l of size $(N_yN_x\ ,\ 1)$.

- Each part G_i^l can be considered as a vectorized formulation of the matrix:

$$G_i^l = \begin{pmatrix} g_i^l(0,0) & \dots & g_i^l(0,N_x-1) \\ \vdots & & \vdots \\ g_i^l(N_y-1,0) & \dots & g_i^l(N_y-1,N_x-1) \end{pmatrix} \tag{3.20}$$

  Note that we use the same notation for both the matrix form or the vectorized form of G_i^l, as the reader can easily differentiate them by looking at the size of G_i^l in the given expression.

- Let us define block-Toeplitz matrix $\mathcal{G}_i^l$ as the "filtering" matrix associated to G_i^l. The term "filtering" points out that we obtain $\mathcal{G}_i^l$ from G_i^l in the same way we obtain $\mathcal{H}^l$ from H^l (see eq. (3.11) and (3.12)).

$$\mathcal{G}_i^l = \begin{pmatrix} \mathcal{G}_{i,0}^l & \cdots & \mathcal{G}_{i,N_x-1}^l & & 0 \\ & \ddots & & \ddots & \\ 0 & & \mathcal{G}_{i,0}^l & \cdots & \mathcal{G}_{i,N_x-1}^l \end{pmatrix} \tag{3.21}$$

  where $\mathcal{G}_{i,j}^l$ is a Toeplitz matrix of size $(M_y\ ,\ M_y+N_y-1)$ associated to the j^{th} column of G_i^l in the matrix form (see eq. (3.20)):

$$\mathcal{G}_{i,j}^l = \begin{pmatrix} g_i^l(0,j) & \dots & g_i^l(N_y-1,j) & & 0 \\ & \ddots & & \ddots & \\ 0 & & g_i^l(0,j) & \dots & g_i^l(N_y-1,j) \end{pmatrix} \tag{3.22}$$

  $\mathcal{G}_i^l$ contains M_x rows of blocks and M_x+N_x-1 columns of blocks of size $(M_y\ ,\ My+N_y-1)$.

- Finally, $\mathcal{G}_i$ stacks the L matrices $\mathcal{G}_i^l$ and is of size (LM_yM_x, d_H).

The quadratic form is now expressed in terms of the filter coefficients:

$$q(H) = H^T \mathbb{Q} H \text{ where } \mathbb{Q} = \sum_{i=0}^{LN_xN_y-d_H-1} \mathcal{G}_i \mathcal{G}_i^T \tag{3.23}$$

The filter coefficients are identified, up to a constant, by the minimal eigenvector of $\mathbb{Q}$.

Modelization

$X = \mathcal{H}D + B$	X : L observed images each of size (N_y, N_x)
	D : the unknown original image
	$\mathcal{H}$: L unknown filtering matrices
	$(LN_yN_x, d_H = (N_y + M_y - 1)(N_x + M_x - 1))$
	B : white noise uncorrelated with D

Autocorrelation matrices

$\mathbb{R}_X = \mathcal{H}\mathbb{R}_D\mathcal{H}^T + \mathbb{R}_B$ $\qquad$ $\mathbb{R}_*$: autocorrelation matrix of $*$

Decomposition between signal and noise subspaces

Eigenvalue decomposition of $\mathbb{R}_X$: $\{\lambda_i, G_i\}$

▷ signal subspace : d_H eigenvectors (largest eigenvalues)

▷ noise subspace : $LN_xN_y - d_H$ eigenvect. (smallest eigenvalues)

Orthogonality condition

$G_i^T\mathcal{H} = \mathbf{0}_{(1,d_H)}$ $\qquad$ $\forall G_i$ in the noise subspace

Structural lemma

$G_i^T\mathcal{H} = H^T\mathcal{G}_i$ $\qquad$ proof of this lemma in [4]

Minimization of the quadratic form

$q(H) = H^T\mathbb{Q}H$ where $\mathbb{Q} = \sum_{i=0}^{LN_xN_y-d_H-1} \mathcal{G}_i\mathcal{G}_i^T$

The filter coefficients are identified, up to a constant, by the minimal eigenvector of $\mathbb{Q}$

Figure 3.2: Subspace methods for filter identification

Chapter 4

Filter identification : extension to superresolution

We now extend the subspace method to the case of subsampled observed images. The subsampling accounts for the aliasing that occurs in every image acquisition process. The purpose is to estimate, from the low-resolution observed images, a deconvolved image at a higher resolution: this problem is called superresolution. To this end, we assume that the original image is filtered by L high-resolution filters, and the L output images are then subsampled by a factor P. The estimation is blind, that is to say, we do not know the filters.
In this chapter, we first state the problem of superresolution as a multiple-input multiple-output (MIMO) one, for 1D signals and for images (section 4.1). Then, in section 4.2, we focus on the limits of the subspace method for MIMO systems. For MIMO systems, the subspace method provides only a mixture of the filters, and no more the actual filters, such as in the single-input multiple-output (SIMO) case. Source separation methods have been used to unmix the result of the subspace method and retrieve the actual filters, but these methods assume that the input signals are not correlated. In our case, the inputs are strongly correlated as they are the various subsampled versions of the same image. We present in section 4.3 our method to disambiguate the mixture, and provide the actual filter, for subsampled input signals.

4.1 Problem statement

4.1.1 1D signals

Each observed signal X^l is modeled as a noisy output of a FIR system driven by an input D (see equation (3.2) page 13) :

$$\underset{(N,1)}{X^l} = \underset{(N,N+M-1)}{\mathcal{H}^l} \underset{(N+M-1,1)}{D} + \underset{(N,1)}{B^l} \tag{4.1}$$

where $\mathcal{H}^l$ is the filtering matrix associated to filter H^l, for $l = 1$ to L.

After the convolution step, the output signals are subsampled by a factor P. These subsampled signals, denoted X^l_{LR}, can be deduced from D following :

$$\underset{(n,1)}{X^l_{LR}} = \underset{(n,P(n+m-1))}{\mathcal{H}^l_{LR}} \underset{(P(n+m-1),1)}{D} + \underset{(n,1)}{B^l_{LR}} \tag{4.2}$$

where $n = \frac{N}{P}$, $m = \frac{M}{P}$ and D is the original signal cut by its $P-1$ last samples. Note that we assume that N and M are multiples of P.

We obtain filtering matrix $\mathcal{H}^l_{LR}$ by extracting one row every P from filtering matrix $\mathcal{H}^l$. If $\mathcal{H}^l$ is written by :

$$\mathcal{H}^l = \begin{pmatrix} h_0^l & \dots & h_{M-1}^l & 0 & \dots & \dots & 0 \\ 0 & h_0^l & \dots & h_{M-1}^l & 0 & \dots & 0 \\ \vdots & & \ddots & & \ddots & & \vdots \\ 0 & \dots & 0 & h_0^l & \dots & h_{M-1}^l & 0 \\ 0 & \dots & \dots & 0 & h_0^l & \dots & h_{M-1}^l \end{pmatrix} \quad (N, M+N-1) \tag{4.3}$$

filtering matrix $\mathcal{H}_{LR}^l$ is written by :

$$\mathcal{H}_{LR}^l = \begin{pmatrix} h_0^l & \dots & h_{M-1}^l & 0 & \dots & \dots & 0 \\ \underbrace{0\dots0}_{P \text{ zeros}} & h_0^l & \dots & h_{M-1}^l & 0 & \dots & 0 \\ \vdots & & \ddots & & \ddots & & \vdots \\ 0 & \dots & 0 & h_0^l & \dots & h_{M-1}^l & \underbrace{0\dots0}_{P \text{ zeros}} \\ 0 & \dots & \dots & 0 & h_0^l & \dots & h_{M-1}^l \end{pmatrix} \quad (n, P(m+n-1)) \tag{4.4}$$

Let us denote H_p^l the subsampled component of filter H^l, also called a polyphase component of H^l :

$$H_p^l = \left[\, h_p \; h_{p+P} \; \dots \; h_{p+(m-1)P} \, \right] \tag{4.5}$$

The filtering matrix associated to H_p^l is denoted by $\mathcal{H}_p^l$ and is of the form :

$$\mathcal{H}_p^l = \begin{pmatrix} h_p^l & h_{p+P}^l & \dots & h_{p+(m-1)P}^l & 0 \\ \ddots & & & \ddots & \\ 0 & h_p^l & h_{p+P}^l & \dots & h_{p+(m-1)P}^l \end{pmatrix} \quad (n, n+m-1) \tag{4.6}$$

By switching on purpose the columns of $\mathcal{H}_{LR}^l$, we obtain an expression of $\mathcal{H}_{LR}^l$ in terms of the filtering matrices associated to the polyphase components of filter H^l :

$$\mathcal{H}_{LR}^l = \left(\underbrace{\begin{matrix} h_0^l & h_P^l & \dots & h_{(m-1)P}^l & 0 \\ \ddots & & & \ddots & \\ 0 & h_0^l & h_P^l & \dots & h_{(m-1)P}^l \end{matrix}}_{\mathcal{H}_0^l} \quad \dots \quad \underbrace{\begin{matrix} h_{P-1}^l & h_{2P-1}^l & \dots & h_{mP-1}^l & 0 \\ \ddots & & & \ddots & \\ 0 & h_{P-1}^l & h_{2P-1}^l & \dots & h_{mP-1}^l \end{matrix}}_{\mathcal{H}_{P-1}^l} \right) \tag{4.7}$$

By switching at the same time the relating rows of D, we obtain an expression of D in terms of its subsampled components :

$$D = \begin{pmatrix} D_0 \\ \vdots \\ D_{P-1} \end{pmatrix} \tag{4.8}$$

where D_p, for $p = 0$ to $P-1$, denotes a subsampled component of D :

$$D_p = [d_p \; d_{p+P} \dots d_{p+(n+m-2)P}]^T \qquad (n+m-1, 1) \tag{4.9}$$

Equation (4.2) can be written as :

$$\begin{matrix} X_{LR}^l & = & \begin{pmatrix} \mathcal{H}_0^l & \dots & \mathcal{H}_{P-1}^l \end{pmatrix} & \begin{pmatrix} D_0 \\ \vdots \\ D_{P-1} \end{pmatrix} & + & B_{LR}^l \\ (n,1) & & (n, P(n+m-1)) & (P(n+m-1), 1) & & (n,1) \end{matrix} \tag{4.10}$$

Let us illustrate this step by an example :

$D = [d_0 \; d_1 \; d_2 \; d_3 \; d_4 \; d_5 \; d_6 \; d_7 \; d_8 \; d_9 \; d_{10}]^T$ denotes a signal of size $M + N - 1 = 11$, filtered by $H^l = [h_0 \; h_1 \; h_2 \; h_3 \; h_4 \; h_5]$ of size $M = 6$, and noisy.
The resulting observed signal, denoted by $X^l = [x_0 \; x_1 \; x_2 \; x_3 \; x_4 \; x_5]$, is of size $N = 6$.
Output signal X^l is related to D following :

$$\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 & 0 & 0 & 0 \\ 0 & 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{pmatrix} \begin{pmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \\ d_9 \\ d_{10} \end{pmatrix} + B^l \tag{4.11}$$

Then the output signal is subsampled by a factor $P = 3$ and subsampled output signal X^l_{LR} is expressed by :

$$\begin{pmatrix} x_0 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 & h_4 & h_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & h_4 & h_5 \end{pmatrix} \begin{pmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \\ d_6 \\ d_7 \\ d_8 \end{pmatrix} + B^{LR}_l \tag{4.12}$$

By switching the columns of the filtering matrix, and at the same time the rows of the original signal, we obtain :

$$\begin{pmatrix} x_0 \\ x_3 \end{pmatrix} = \begin{pmatrix} h_0 & h_3 & 0 & h_1 & h_4 & 0 & h_2 & h_5 & 0 \\ 0 & h_0 & h_3 & 0 & h_1 & h_4 & 0 & h_2 & h_5 \end{pmatrix} \begin{pmatrix} d_0 \\ d_3 \\ d_6 \\ d_1 \\ d_4 \\ d_7 \\ d_2 \\ d_5 \\ d_8 \end{pmatrix} + B^{LR}_l \tag{4.13}$$

The filtering matrix can be partitioned into three sub-matrices :

$$\mathcal{H}^l_0 = \begin{pmatrix} h_0 & h_3 & 0 \\ 0 & h_0 & h_3 \end{pmatrix} \quad \mathcal{H}^l_1 = \begin{pmatrix} h_1 & h_4 & 0 \\ 0 & h_1 & h_4 \end{pmatrix} \quad \mathcal{H}^l_2 = \begin{pmatrix} h_2 & h_5 & 0 \\ 0 & h_2 & h_5 \end{pmatrix} \tag{4.14}$$

which can be seen as the filtering matrices of the polyphase components $H^l_0 = [h_0\ h_3]^T$, $H^l_1 = [h_1\ h_4]^T$, and $H^l_2 = [h_2\ h_5]^T$.
Let us denotes $D_0 = [d_0\ d_3\ d_6]$, $D_1 = [d_1\ d_4\ d_7]$ and $D_2 = [d_2\ d_5\ d_8]$. D_p, for $p = 0$ to $P-1$, is a subsampled component of the original signal.

Subsampled output signal X^l_{LR} can be expressed in terms of the subsampled components of the original signal D :

$$X^{LR}_l = \begin{pmatrix} \mathcal{H}^l_0 & \mathcal{H}^l_1 & \mathcal{H}^l_2 \end{pmatrix} \begin{pmatrix} D_0 \\ D_1 \\ D_2 \end{pmatrix} + B^{LR} \tag{4.15}$$

$$\tag{4.16}$$

Through this example, and in a more general framework, we show that the superresolution problem can be stated as a multiple-input multiple-output (MIMO) system, more precisely a P-input and L-output system, where the inputs are the subsampled components of the original signal.

$$\begin{array}{ccccccc} \begin{pmatrix} X^1_{LR} \\ \vdots \\ X^L_{LR} \end{pmatrix} & = & \begin{pmatrix} \mathcal{H}^1_0 & \dots & \mathcal{H}^1_{P-1} \\ \vdots & & \vdots \\ \mathcal{H}^L_0 & \dots & \mathcal{H}^L_{P-1} \end{pmatrix} & & \begin{pmatrix} D_0 \\ \vdots \\ D_{P-1} \end{pmatrix} & + & B_{LR} \\ (Ln, 1) & & (Ln, P(n+m-1)) & & (P(n+m-1), 1) & & (Ln, 1) \end{array} \tag{4.17}$$

4.1.2 Images

Each observed image X^l, $l = 1 : L$, is modeled as a noisy output of a FIR system $\mathcal{H}^l$ driven by an input image D (see section 3.1.2):

$$X^l = \mathcal{H}^l D + B^l \tag{4.18}$$

Then, the outputs are subsampled by a factor P:

$$X^l_{LR} = \mathcal{H}^l_{LR} D + B^l_{LR} \tag{4.19}$$

In this expression :

- X^l_{LR} is a subsampled component of X^l, of size $(n_x n_y, 1)$,
  where $n_x = \frac{N_x}{P}$ and $n_y = \frac{N_y}{P}$,
- D is the same as in equation (4.18), apart from the last $P-1$ rows and columns which are truncated,
- $\mathcal{H}^l_{LR}$ is defined by extracting one row every P from matrix $\mathcal{H}^l$
  and is of size $(n_x n_y, d_h)$, as we discard all the null columns,
  where $d_h = P^2(n_x + m_x - 1)(n_y + m_y - 1)$, $m_x = \frac{M_x}{P}$ and $m_y = \frac{M_y}{P}$.

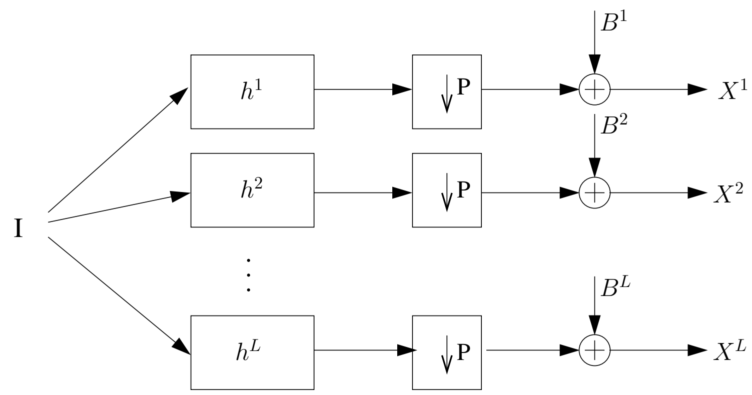


Figure 4.1: Model of acquisition for multiple-input multiple-output system

By switching on purpose the columns of $\mathcal{H}^l_{LR}$ (and at the same time the rows of D) in equation (4.19), the subsampled output images can be related to the subsampled components of the original image:

$$X^l_{LR} = \begin{pmatrix} \mathcal{H}^l_{0,0} & \mathcal{H}^l_{0,1} & \dots & \mathcal{H}^l_{P-1,P-1} \end{pmatrix} \begin{pmatrix} D_{0,0} \\ D_{0,1} \\ \vdots \\ D_{P-1,P-1} \end{pmatrix} + B^l_{LR} \tag{4.20}$$

where :

- D_{p_1,p_2} is a vectorized subsampled component of the input image D, i.e., if

$$D = \begin{pmatrix} d_{0,0} & \dots & d_{0,S_x-1} \\ \vdots & & \vdots \\ d_{S_y-1,0} & \dots & d_{S_y-1,S_x-1} \end{pmatrix} \tag{4.21}$$

  where $S_y = N_y + M_y - 1$ and $S_x = N_x + M_x - 1$,
  thus, for all $p_1, p_2 = 0 : P-1$, then D_{p_1,p_2} is a vectorized form of

$$D_{p_1,p_2} = \begin{pmatrix} d_{p_1,p_2} & d_{p_1,p_2+P} & \dots & d_{p_1,p_2+P(s_x-1)} \\ \vdots & \vdots & & \vdots \\ d_{p_1+P(s_y-1),0} & d_{p_1+P(s_y-1),P} & \dots & d_{p_1+P(s_y-1),p_2+P(s_x-1)} \end{pmatrix} \tag{4.22}$$

  where $s_y = n_y + m_y - 1$ and $s_x = n_x + m_x - 1$,
  note that we use the same notation for the vectorized and the matrix forms of D_{p_1,p_2},
- and $\mathcal{H}^l_{p_1,p_2}$ is the block-Toeplitz matrix of size $(n_y n_x, s_y s_x)$ associated to filter

$$H^l_{p_1,p_2} = \begin{pmatrix} h^l_{p_1,p_2} & \dots & h^l_{p_1,p_2+(m_x-1)P} \\ h^l_{p_1+P,p_2} & \dots & h^l_{p_1+P,p_2+(m_x-1)P} \\ \vdots & & \vdots \\ h^l_{p_1+(m_y-1)P,p_2} & \dots & h^l_{p_1+(m_y-1)P,p_2+(m_x-1)P} \end{pmatrix} \tag{4.23}$$

  one of the P^2 polyphase components of high resolution filter H^l (see eq. (3.10)).

By stacking all vectors and matrices coming from equation (4.20) for all $l = 1 : L$, we obtain the following model:

$$\begin{pmatrix} X_{LR}^1 \\ \vdots \\ X_{LR}^L \end{pmatrix} = \begin{pmatrix} \mathcal{H}_{0,0}^1 & \dots & \mathcal{H}_{P-1,P-1}^1 \\ \vdots & & \vdots \\ \mathcal{H}_{0,0}^L & \dots & \mathcal{H}_{P-1,P-1}^L \end{pmatrix} \begin{pmatrix} D_{0,0} \\ \vdots \\ D_{P-1,P-1} \end{pmatrix} + B_{LR} \tag{4.24}$$

The superresolution problem is now expressed like a multiple-input multiple-output problem. In multiple-input systems, the inputs usually come from different sources, and are considered as independent from each other [3]. In our case, the inputs are the different subsampled components of the same source image and are therefore strongly correlated.

4.2 Limits of the subspace method

In this section, we show that, for subsampled images, the subspace method is not sufficient to determine the filters, but provide an identification up to a (P^2, P^2) mixing matrix.

Let us call $\mathbb{R}_X^{LR}$ the autocorrelation matrix of L subsampled images X_{LR}^l.
If we apply the subspace method, we find that the eigenvectors, denoted G_i, associated to the $d_h = P^2(n_y + m_y - 1)(n_x + m_x - 1)$ largest eigenvalues of $\mathbb{R}_X^{LR}$ span the signal subspace, and that the eigenvectors associated to the $Ln_yn_x - d_h$ smallest eigenvalues of $\mathbb{R}_X^{LR}$ span the noise subspace.

The orthogonality condition between noise and signal subspaces is expressed by:

$$\underset{(1,Ln_yn_x)}{G_i^T} \quad \underset{(Ln_yn_x,\, s_ys_x)}{\mathcal{H}_{p_1,p_2}} = \underset{(1,\, s_ys_x)}{\mathbf{0}_{(1,s_ys_x)}} \tag{4.25}$$

where $i = 0 : Ln_yn_x - d_h - 1$, $\mathbf{0}_{(1,s_ys_x)}$ is a null vector of size $(1, s_ys_x)$,
and $\mathcal{H}_{p_1,p_2}$ a block column of the filtering matrix in equation (4.24).

$$\mathcal{H}_{p_1,p_2} = \begin{pmatrix} \mathcal{H}_{p_1,p_2}^1 \\ \vdots \\ \mathcal{H}_{p_1,p_2}^L \end{pmatrix} \tag{4.26}$$

The structural lemma (see equation (3.19)) provides an expression of the orthogonality condition in terms of the filter polyphase components, instead of in terms of the columns of the filtering matrix:

$$\mathbb{H}_{p_1,p_2}^T \mathcal{G}_i = \mathbf{0}_{(1,s_ys_x)} \quad \text{where} \quad \mathbb{H}_{p_1,p_2} = \begin{pmatrix} H_{p_1,p_2}^1 \\ \vdots \\ H_{p_1,p_2}^L \end{pmatrix} \tag{4.27}$$

where $\mathcal{G}_i$ is a (Lm_ym_x, s_ys_x) filtering matrix defined from eigenvectors G_i, and $\mathbb{H}_{p_1,p_2}$, for $p_1, p_2 = 0 : P - 1$ is of size $(Lm_ym_x, 1)$.

By stacking the contributions of all the polyphase components of the filters, we obtain:

$$\mathbb{H}^T \mathcal{G}_i = \mathbf{0}_{(P^2,s_ys_x)} \quad \text{where} \quad \mathbb{H} = \begin{pmatrix} \mathbb{H}_{0,0} & \dots & \mathbb{H}_{P-1,P-1} \end{pmatrix} \tag{4.28}$$

The minimization of the quadratic form associated to the orthogonality condition provides a set of P^2 vectors, denoted $\mathbb{V}$. We can not distinguish these eigenvectors using only the orthogonality condition. Indeed, each column of $\mathbb{V}$ is in the nullspace of the quadratic form, therefore $\mathbb{H}$ is a combination of the P^2 columns of $\mathbb{V}$. We can identify filters $\mathbb{H}$ only up to a reversible (P^2, P^2) mixing matrix denoted $\mathbb{M}_X$, such as:

$$\mathbb{H} = \mathbb{V}\mathbb{M}_X \tag{4.29}$$

Source separation methods have been used to estimate such a matrix [2, 3], but these methods usually state the assumption that the input signals are uncorrelated. This is not our case, as the inputs are the different subsampled components of the same source image.

4.3 Evaluation of the Mixing Matrix

The determination of matrix $\mathbb{M}_X$ is, as we showed theoretically, impossible in the case where the mixed sources (here the polyphase components of an image) have the same distribution.

Despite this fact, we try to estimate the mixing matrix by introducing some prior knowledge on the statistics of the image or the filters. Indeed, natural images have a spectrum which is far from constant (as in the case of a white noise or a compressed signal). On the other hand, filters that are encountered in image processing are often very smooth with a single local (and global) maximum at the origin, whereas a multi-reflection filter, that affects wireless communications, can be irregular and display a multitude of local maxima. The subspace method was designed to deal with such irregular filters, with the counterpart that the sources are of different statistical nature, allowing an efficient separation of sources.

In this section we will use a continuous notation, and the Fourier transform of a sampled signal at rate 1 will live in $[-1/2, 1/2]$ whereas the Fourier transform of a subsampled version at rate P, will live in $[-1/2P, 1/2P]$. $\tilde{H}^l$ will refer to the estimated filters we are trying to define.

4.3.1 Imposing Regularity of the Filters

First, let us see what happens when some regularity is imposed to the filters. We do so by minimizing a certain regularity measure of the filters under the constraint that the integral of each filter is one[1].

Two principal choices have been proposed for the measure of filters regularity.
The first one (which presents the advantage of a low computational cost) is the integral of the squared norm of the gradient (the $\mathbb{H}_1$ norm [12]) :0

$$J_1(\tilde{H}^1, ..., \tilde{H}^L) = \sum_l \int \left\|\nabla \tilde{H}^l\right\|_2^2 . \tag{4.30}$$

The other one is the integral of the gradient (the total variation norm [13]).

The first choice may lead to smooth solutions and disadvantages the non continuous filters (such as motion blur). Nevertheless, we use this $\mathbb{H}_1$ criterion, for two reasons:

- We search for the best solution in a small-dimensional affine space (namely the vector space in which $\mathbb{M}_X$ lives intersected with the affine space represented by constraint $\int H^l(x)dx = 1$). In such a case, the smoothing effect of $\mathbb{H}_1$ norm compared to TV norm could be ignored.
- The computational cost of such a minimization is much smaller than TV one (see for example [14] for the numerical intricacy of TV minimization, although recent advances have been made [15] but are not, as is, applicable to our problem).

4.3.2 Imposing Similarity of the Double-Filtered Images

In the following, we take advantage of the fact that we have multiple views of the same original scene to recover the filters (which implies the estimation of $\mathbb{M}_X$). Let's assume that we have two versions I_1 and I_2 of the same image, formed after being filtered by F_1 and F_2, and that we have two candidates $\tilde{F}_1$ and $\tilde{F}_2$: we can check easily if these candidates are reasonable or not. Indeed filtering I_2 using $\tilde{F}_1$ should yield the same result as filtering I_1 using $\tilde{F}_2$.

Based on this simple observation, we define a functional which should be minimized by our computed filters:

$$J_2(\tilde{H}^1, ..., \tilde{H}^L) = \sum_{k,l=1}^{k,l=L} \left\|\tilde{H}^l * X^k - \tilde{H}^k * X^l\right\|_2^2 . \tag{4.31}$$

where X^k are the observed images and $\tilde{H}^k$ are the estimated filters.
Note that we don't have access to a fully sampled version of the X^k, thus we interpret the convolutions that occurs in (4.31) as the product of the low frequencies of the filter $\tilde{H}$ with the Fourier transform of X, squaring the result and summing over the low-frequency domain.

We define[2]

$$\left\|\tilde{H}^l * X^k - \tilde{H}^k * X^l\right\|_2^2 = \int_{-\frac{1}{2P}}^{\frac{1}{2P}} \left|\hat{\tilde{H}}^l(u)\hat{X}^k(u) - \hat{\tilde{H}}^k(u)\hat{X}^l(u)\right|^2 du \tag{4.32}$$

This last functional could be the perfect criterion if no subsampling were present. Indeed, J_2 is null in a noise-free, well-sampled setting only if the filters are the real filters (after checking that J_2 is a positive definite quadratic form).

[1] This is a physical requirement for imaging filters. It may not be true if different images have been acquired under different illumination conditions. In this case, the mean of each image gives a very accurate estimation of the integral of the filter that generated it.

[2] We use a one dimensional notation to simplify the equations, we consider an infinite-size discrete signal subsampled at rate P. The hat denotes the time-discrete Fourier transform of a signal

Unfortunately the subsampling that affects our images is expressed by :

$$\begin{aligned}
& \left|\hat{H}^l(u)\hat{X}^k(u) - \hat{H}^k(u)\hat{X}^l(u)\right|^2 \\
= & \left|\hat{H}^l(u)\sum_{n=0}^{P-1}\hat{X}^0(u+\frac{n}{P})\hat{H}^k(u+\frac{n}{P}) - \hat{H}^k(u)\sum_{n=0}^{P-1}\hat{X}^0(u+\frac{n}{P})\hat{H}^l(u+\frac{n}{P})\right|^2 \\
= & \left|\sum_{n=1}^{P-1}\hat{X}^0(u+\frac{n}{P})\left(\hat{H}^k(u)\hat{H}^l(u+\frac{n}{P}) - \hat{H}^l(u)\hat{H}^k(u+\frac{n}{P})\right)\right|^2
\end{aligned} \tag{4.33}$$

for $u \in [-\frac{1}{2P}, \frac{1}{2P}]$,
where H^k are the actual filters and X^0 is the original image.

Criterion J_2 being not null when applied to the actual filters prevents us from concluding that its minimum is obtained for those filters. Nevertheless, images have a strong low-frequency component. This means that the minimizing filters for J_2 must reduce as much as possible the terms of the form $|\hat{\tilde{H}}^k(u)\hat{X}^0(u) - \hat{\tilde{H}}^l(u)\hat{X}^0(u)|^2$, because these terms dominate the others (see [16] for a review of proposed statistical models of images). As the experiments will show it, the error introduced by the aliasing is negligible and does not lead to a noticeable error in the recovery of the filters.

One can also say that the high frequency components of the filters are not taken into account. Although this point is correct, the filters, thanks to the subspace method, are constrained to live in a small-dimensional affine space, thus controlling the low frequency part of them is sufficient to yield a positive definite quadratic form on the subspace the filters live in.
In the next section we see how these two ideas can be applied to the disambiguation of mixing matrix $\mathbb{M}_X$.

4.4 Effects of noise

In this section, we try to answer to the question: what is the effects of noise on the filter identification? To this end, we study the impact of noise on the main steps of the subspace method.

The first main step is the partition of the eigenvectors of the autocorrelation matrix between noise subspace and signal subspace. Without noise, we have seen, in section 3.2 page 15, that the signal subspace is spanned by the eigenvectors associated with the d_H largest eigenvalues of $\mathbb{R}_X$, and the noise subspace is spanned by the eigenvectors associated with the $LN_xN_y - d_H$ smallest eigenvalues of $\mathbb{R}_X$.

For example, we simulate the acquisition process of a scene, represented by the image Lena, through a set of $L = 6$ sensors. Let us apply the subspace method to this set of images, with the following parameters: $P = 1$, $N_y = N_x = 5$, $M_y = M_x = 4$ and several values for the noise standard deviation $std = 0, 0.2, 1, 5$ (see figure 4.2 page 25 for an equivalence of this standard deviation in term of psnr between the noisy blurred images and the blurred ones).

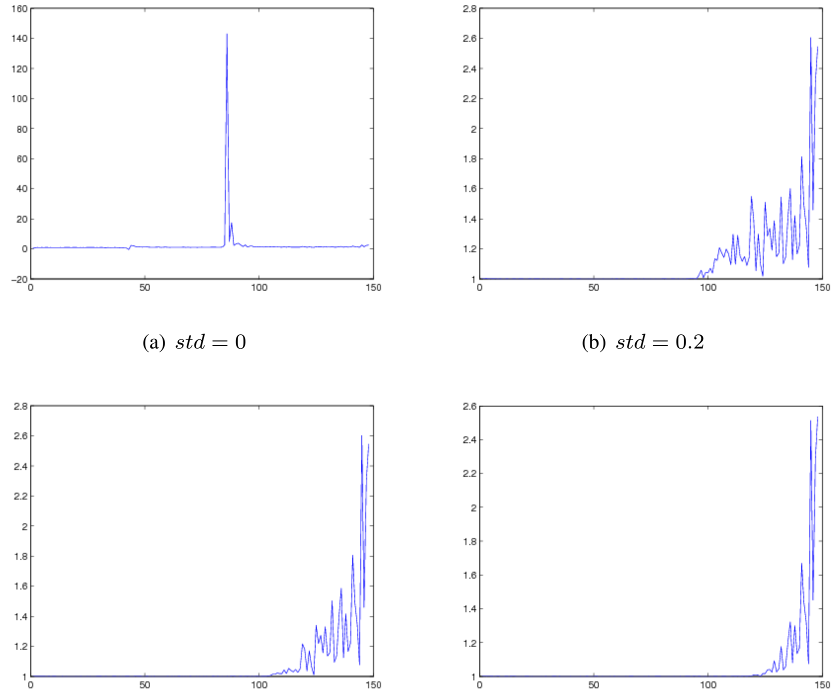
std	0	0.2	1	5
psnr (dB)	∞	60.43	46.46	32.48

Figure 4.2: Link between the noise standard deviation and the psnr of the acquisition

For $std = 0$, the signal subspace is of size $d_H = (N_x + M_x - 1)(N_y + M_y - 1) = 64$. If the autocorrelation matrix eigenvalues are sorted by increasing order, the first 86 eigenvalues span the noise subspace, and their value is zero, and the last 64 eigenvalues span the signal subspace. This partition is illustrated in figure 4.3.(a) by a peak in the ratio between the eigenvalues $\frac{\lambda_{i+1}}{\lambda_i}, \forall i = 1 : LN_yN_x$. This peak illustrates the gap between the eigenvalues of the noise subspace and those of the signal subspace and materializes the separation between signal and noise subspace.
As the noise increases, the separation between noise and signal subspace is less and less obvious (see figure 4.3), as noise affects the autocorrelation matrix eigenvalues.

Another main step of the subspace method is the computation of the quadratic form associated to the orthogonality condition between signal and noise subspaces. Without noise, the minimal eigenvector of the quadratic form Q provides the filter coefficients.

But in noisy conditions, how many eigenvectors of Q are necessary to estimate the filter coefficients? To answer this question, we study, for each noise standard deviation, the decomposition of the filter coefficients in the basis of eigenvectors of



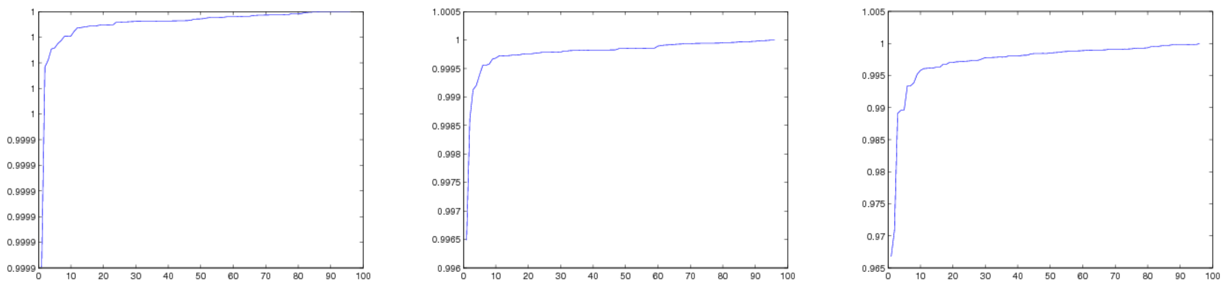
(a) $std = 0$ (b) $std = 0.2$

(c) $std = 1$ (d) $std = 5$

Figure 4.3: Ratio of the autocorrelation matrix eigenvalues for $\sigma = 0, 0.2, 1, 5$

Q.

The results are shown in figure 4.4 page 26. In this figure, we can see that, for $std = 5$, the minimal eigenvector contains 95% of the energy of the projection of the filter coefficients in the eigenvectors basis. But we need at least 6 eigenvectors to reach 99% of the energy.



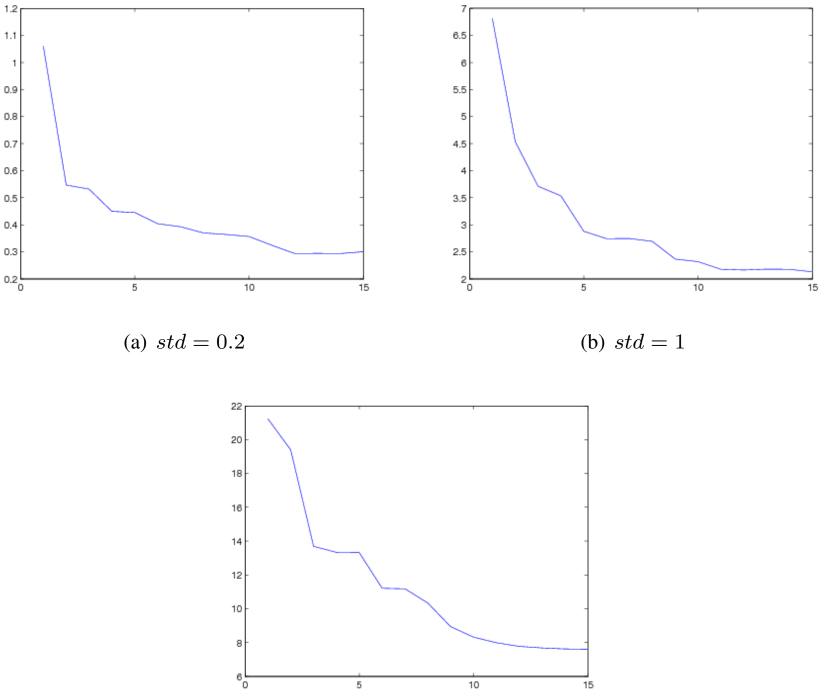
(a) $std = 0.2$ (b) $std = 1$ (c) $std = 5$

Figure 4.4: Decomposition of the filter coefficients in the eigenvectors of Q basis.

From these results, we ask if using several eigenvectors can improve the filters identification in noisy conditions.

Figure 4.5 page 27 presents the relative minimal error expected (in $\%$) relative to the number of eigenvectors implied in the filters identification. As several eigenvectors are taken into account, we can identify the filters up to a mixing matrix. We estimate the minimal error which can be expected using only a given number of eigenvectors, this error is given in percentage relative to the real known filter coefficients.

Figure 4.5 page 27 shows that, for a noise standard deviation of 5, if we use only one eigenvector, the error is of 21.24%, but it can be reduced to 13.69% by using only 3 eigenvectors.



(a) $std = 0.2$ (b) $std = 1$

(c) $std = 5$

Figure 4.5: Minimal error expected vs the number of eigenvectors taken into account

We conclude that, if noise affects the subspace method, the use of a few more additional eigenvectors than those expected can improve significantly the results, by it requires the estimation of a mixing matrix, even in the case where $P = 1$.

Chapter 5

Restoration

Once the filters are estimated, the recovery of the original image can take place. Recovered image D_{opt} must satisfy some straightforward conditions, namely :

- The image filtered by the estimated filters H^l and subsampled must be close to the observed images, which yields the first data-driven functional :

$$A(D_{opt}) = \sum_{l=1}^{L} \left\| S_P(H^l * D_{opt}) - X^l \right\|_2^2, \tag{5.1}$$

  where S_P is the subsampling operator at rate P.

- Since the observed images are affected by noise and, most importantly, the filters we computed are estimates of the actual ones, a regularization functional must also be minimized:

$$R(D_{opt}) = \int \|\nabla D_{opt}\|_2^2 \tag{5.2}$$

These two criteria sum up to the minimization of a single functional given by:

$$J_3(D_{opt}) = A(D_{opt}) + \lambda R(D_{opt}), \tag{5.3}$$

Let us denote A_l the operator including the convolution by filter H^l followed by the subsampling operator at rate P:

$$A_l(D) = S_P(H^l * D), \tag{5.4}$$

for $l = 1$ to L.

Minimum D_{opt} is obtained by solving the following system:

$$\left(\sum_{l=1}^{L} A_l^T \; A_l + \lambda \; C^T \; C \right) \; D_{opt} = \sum_{l=1}^{L} A_l^T \; X^l \tag{5.5}$$

where C denotes the gradient operator.
The system is solved in the frequency domain, as the operations of convolution and subsampling benefit from an easy and fast implementation.

First let us study operator A_l: this operator provides the convolution of input signal D with filter H^l and subsamples the result at a rate P. In the frequency domain, this operation is equivalent to multiply the Discrete Fourier Transform (DFT) of the input signal, denoted $\widehat{D}$, by the DFT of filter H^l, denoted $\widehat{H}^l$, and then, to take into account the aliasing resulting from the subsampling :

$$\begin{aligned} \widehat{A_l D}(w) &= \widehat{D}(w) \; \widehat{H^l}(w) + \widehat{D}(w+N) \; \widehat{H^l}(w+N) + \cdots \\ &+ \widehat{D}(w+(P-1)) \; \widehat{H^l}(w+(P-1)N) \end{aligned} \tag{5.6}$$

where the DFT of each low-resolution signal is computed on N uniformly-spaced samples.

Now we focus on operator A_l^T: this operator provides the up-sampling of the input signal at a rate P, followed by the convolution of the result by filter $(H^l)^H$, the conjugate transpose of H^l.

$$A_l^T(D) = (H^l)^T * S_P(D) \tag{5.7}$$

In the frequency domain, this operation is equivalent to duplicate P times the spectrum of D, weighted by $\frac{1}{P}$, then to multiply the result by $(\widehat{H}^l)^H$:

$$\begin{pmatrix} \widehat{A_l^T D}(w) \\ \widehat{A_l^T D}(w+N) \\ \vdots \\ \widehat{A_l^T D}(w+(P-1)N) \end{pmatrix} = \frac{1}{P} \begin{pmatrix} \widehat{H^l}^*(w) \\ \widehat{H^l}^*(w+N) \\ \vdots \\ \widehat{H^l}^*(w+(P-1)N) \end{pmatrix} \widehat{D}(w) \tag{5.8}$$

where $w = 1 : N$ and $\widehat{H^l}^*(w)$ is the conjugate of $\widehat{H^l}(w)$.

We deduce the formulation of equation (5.5) in the Fourier domain :

$$\left(\frac{1}{P} \sum_{l=0}^{L-1} \left((\widehat{\mathbf{H^l}})^H(w)\, \widehat{\mathbf{H^l}}(w) \right) + \lambda\, \widehat{\mathbf{C_2}}(w) \right) \widehat{\mathbf{D}}_{\mathbf{opt}}(w) = \frac{1}{P} \sum_{l=0}^{L-1} (\widehat{\mathbf{H^l}})^H(w)\, \widehat{\mathbf{X}}_l^{LR}(w) \tag{5.9}$$

where

$$\widehat{\mathbf{H^l}}(w) = \left[\widehat{H^l}(w)\ \widehat{H^l}(w+N)\ \cdots\ \widehat{H^l}(w+(P-1)N) \right] \tag{5.10}$$

$$\widehat{\mathbf{D}}_{\mathbf{opt}}(w) = \left[\widehat{D}_{opt}(w)\ \widehat{D}_{opt}(w+N)\ \cdots\ \widehat{D}_{opt}(w+(P-1)N) \right] \tag{5.11}$$

$$\widehat{\mathbf{C_2}}(w) = \left[|\widehat{C}(w)|^2\ |\widehat{C}(w+N)|^2\ \cdots\ |\widehat{C}(w+(P-1)N)|^2 \right] \tag{5.12}$$

and $w = 1 : N$.

Resolution of equation (5.9) requires the inversion of a $P^2 \times P^2$ matrix for each point of the Fourier transform of the image.

Chapter 6

Experimental results

6.1 Simulations for objective evaluation

We want to estimate a deconvolved image, at a resolution increased by a factor $P = 2$, from a set of $L = 6$ low-resolution images of the same scene, filtered by 6 different unknown filters. This can be expressed as a 4-input 6-output system.

To evaluate the results with an objective criterion, the psnr (see equation (6.1)), we have to simulate this case: we filter a known original image D with 6 known filters H, and then subsample the outputs by a factor $P = 2$ in each directions.

$$PSNR(D, D_{est}) = 10 \log_{10} \frac{(\max(D) - \min(D))^2}{MSE(D, D_{est})} \tag{6.1}$$

where MSE is the mean squared error between the images D and D_{est}.

The original image D is of size $(576, 720)$, and the windowed area of study is of size $(10, 10)$. The filters are $(6, 6)$ 2D-Gaussian centered at a random point with standard deviations: $0.7, 0.9, 1, 1.1, 1.3, 1.5$.
To recover the filters, we use a weighted sum of the two criteria on the filters, defined in equation (4.30) and equation (4.31) page 24: $\alpha J1 + (1 - \alpha)J2$. Note that J_1 and J_2 are normalized so their minimal eigenvalue is 1.
We obtain a psnr of 22.12 dB for $\alpha = 1$, and a psnr of 21.46 dB for $\alpha = 0$. The results are better when the two criteria are mixed, in our case for $\alpha = 0.04$, the filters are recovered with a psnr of 26.17 dB.

We present experimental results obtained with the observed output subsampled images and the filters estimated below.
Figure 6.1 shows 3 of the 6 output images, from the less blurred on the left, to the the more blurred on the right.
Figure 6.2 shows the restored image ($psnr = 35.98$ dB with $\lambda = 10^{-3}$ in equation (5.3)) versus the original image.
To better display the results, we focus on a window area of the less blurred output image, and the related window area in the superresolved image, and display them at their exact size (figure 6.3). For comparison purposes, a bilinear interpolation of the output image area and the related window in the original image are also given.

Figure 6.1: 3 of the 6 output images: frames 1,3 and 6 (scale = 0.35)

Figure 6.2: upper : the original image, down : the restored image (scale = 0.35)

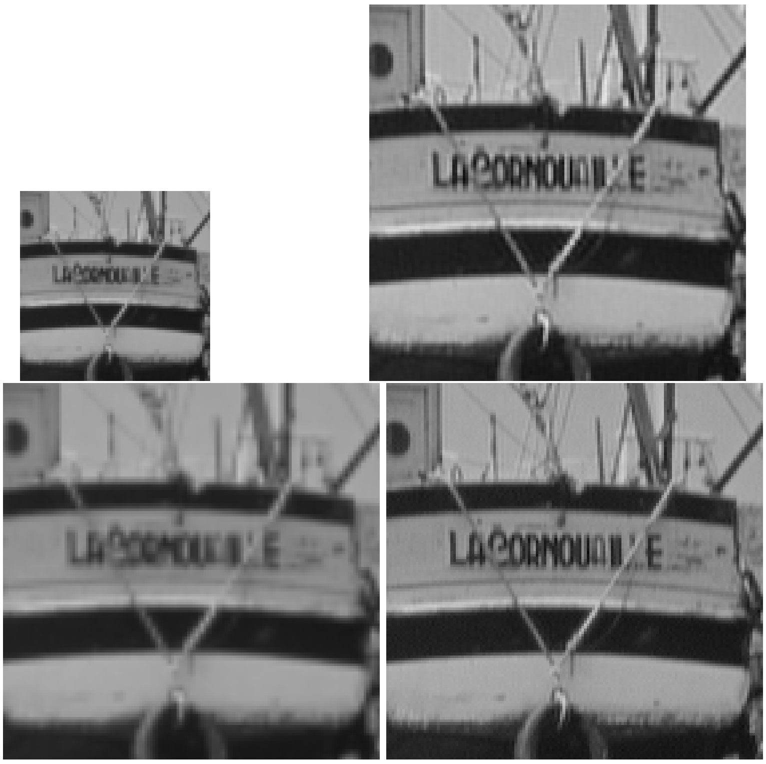


Figure 6.3: upper left : the 1st observed image; upper right : the restored image; down left : the bilinear interpolation; down right : the original image (scale=1)

6.2 Real images

We present experimental results obtained on real images, for blind restoration on sequence C1V2P32_14h00_16b.oti in section 6.2.1 and for blind superresolution on sequence Enst-09.ptw (section 6.2.2) and on images of the moon (section 6.2.3).

6.2.1 Sequence C1V2P32_14h00_16b

We apply our blind restoration algorithm on 5 frames of sequence C1V2P32_14h00_16b.oti, starting on the 530^{th} of the sequence. Figure 6.4 shows the observed frames.

As we aim to evaluate a restored frame, and not a superresolved one, we use the filter identification method as presented in section 3 page 13, and the restoration method presented in 5 page 29 with P=1.

But first, as the frames comes from a sequence with in-motion camera, we have to register all the frames relatively to a reference one : the middle one (the 532^{th}).
The images are registered using the following parameters:

- The motion is estimated with an accuracy up to an half pixel, using a spline interpolation method.
- The exhaustive block-matching is performed over a $(10, 10)$ research window, i.e. the largest displacement between the current frame and the frame of reference is assumed to be lesser than $+/-10$ pixels in horizontal and vertical directions. The frames are partitioned into blocks of size $(8, 8)$, and the matching criterion is evaluated onto blocks of the same size.
- We choose for parameter σ of the weighting function (2.16) the value 1.

The estimated motion model are :
between the first frame and the frame of reference :

$$A = \begin{pmatrix} 0.98268 & -0.00040 \\ -0.00078 & 0.97961 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} -4.00266 \\ 0.45556 \end{pmatrix} \tag{6.2}$$

between the second frame and the frame of reference :

$$A = \begin{pmatrix} 0.99131 & -0.00004 \\ -0.00025 & 0.98897 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} -2.06137 \\ 0.14216 \end{pmatrix} \tag{6.3}$$

between the fourth frame and the frame of reference :

$$A = \begin{pmatrix} 1.00839 & -0.00016 \\ 0.00005 & 1.01040 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 2.13106 \\ -0.02691 \end{pmatrix} \tag{6.4}$$

between the fifth frame and the frame of reference :

$$A = \begin{pmatrix} 1.01783 & -0.00019 \\ 0.00029 & 1.02126 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 4.04494 \\ -0.25224 \end{pmatrix} \tag{6.5}$$

Figure 6.5 shows the registered images (the third image is the image of reference).

The filter identification is provided following the method developed in section 3 page 13, assuming that the filters are of size $(4, 4)$. The resulting estimated filters are shown in Figure 6.6.

Finally, we estimate the restored image as proposed in section 5 page 29 with $P = 1$. For comparison purposes, we estimate the median of the registered images. In figure 6.7 we show several restored images associated to different regularization parameters, in order to illustrate the influence this parameter have on the accuracy of the restoration : too low and the restored image is still noisy, too high and the restored image is blurred. However the restored image with a parameter of regularization of 0.1 is sharpen that the median of the registered images, and than the observed frames, of course.

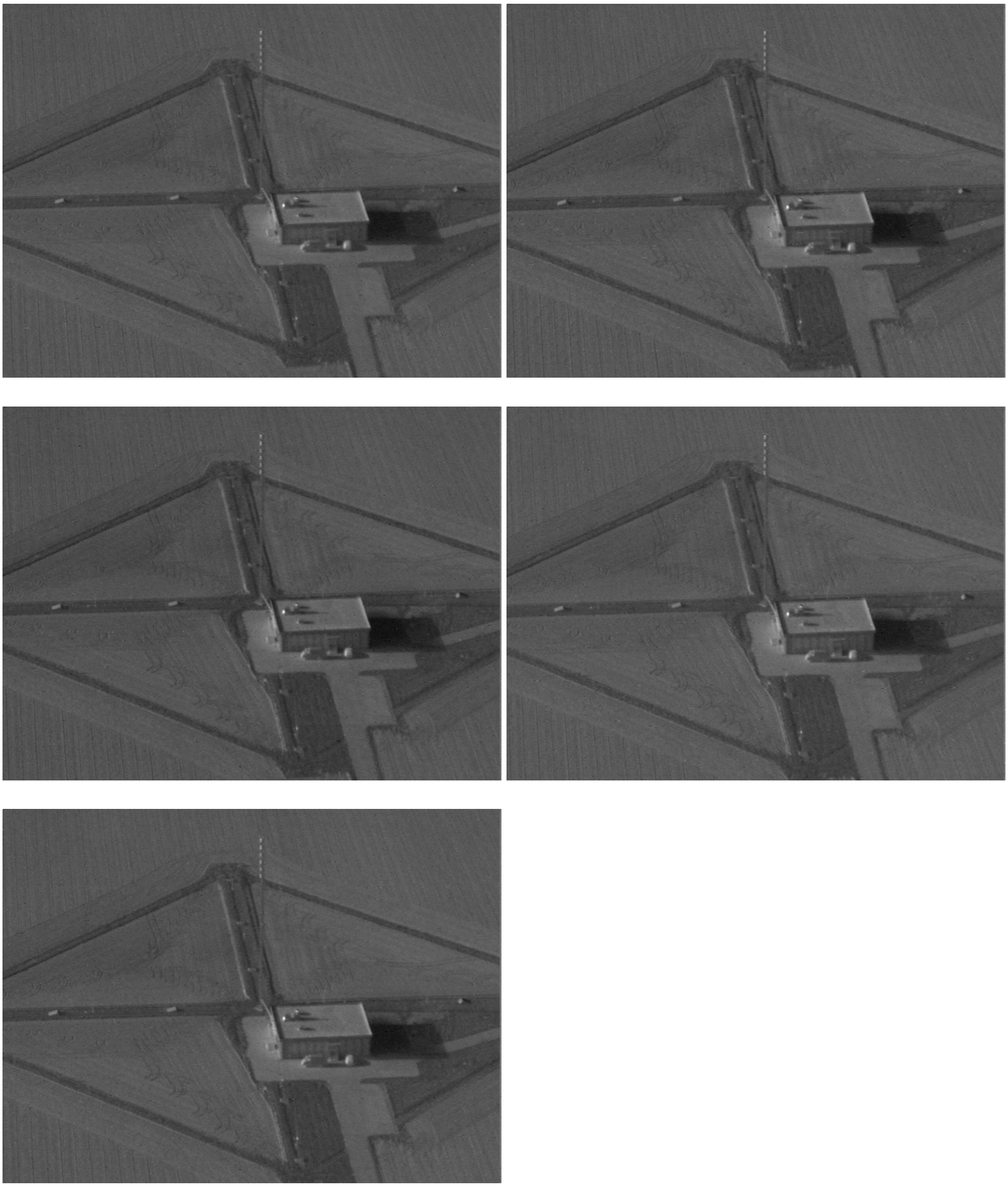


Figure 6.4: Frames 530, 531, 532, 533 and 534 of sequence C1V2P32_14h00_16b.oti

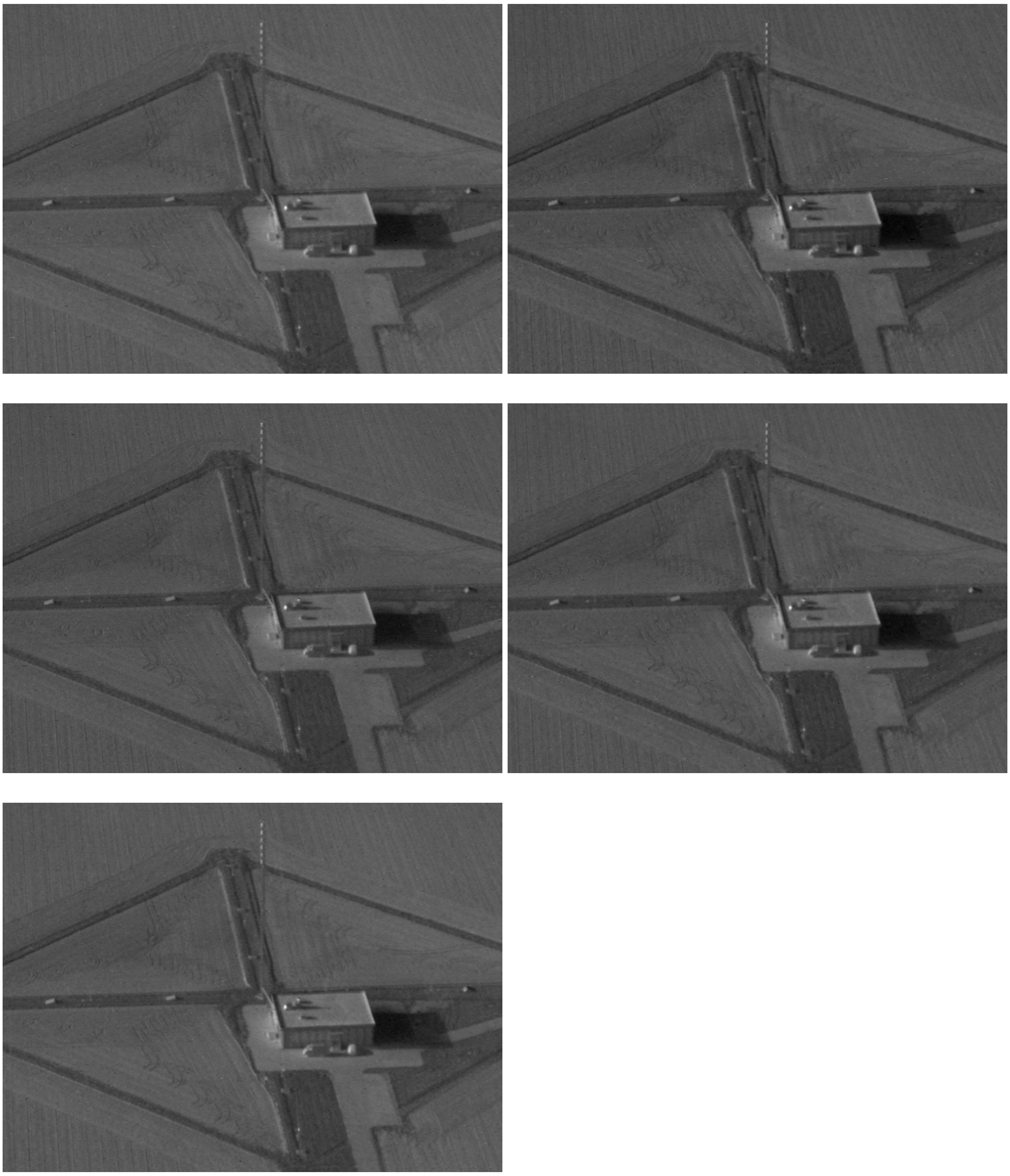


Figure 6.5: Registered images : 530, 531, 532 (image of reference), 533 and 534 of sequence C1V2P32_14h00_16b.oti

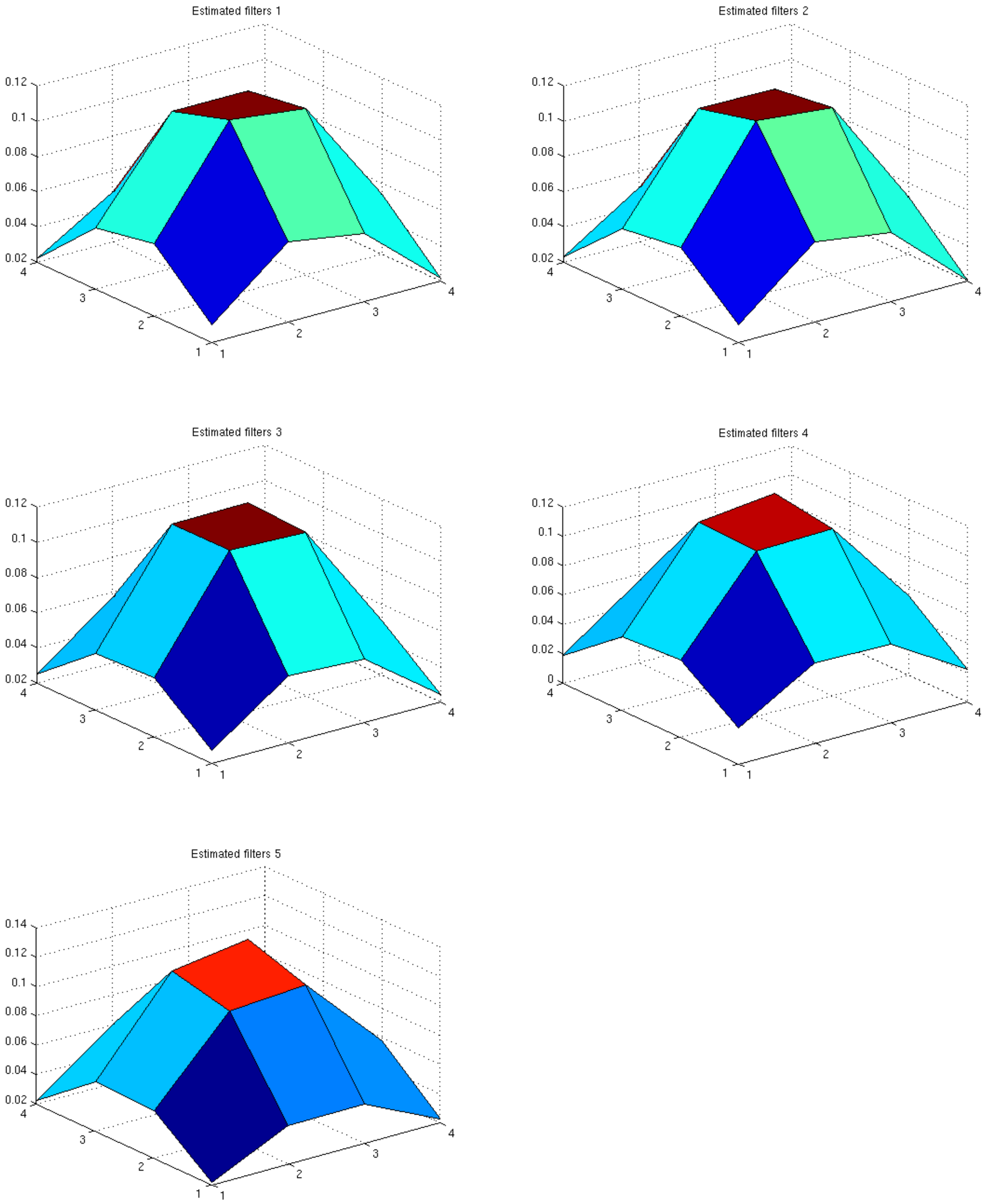


Figure 6.6: Estimated filters

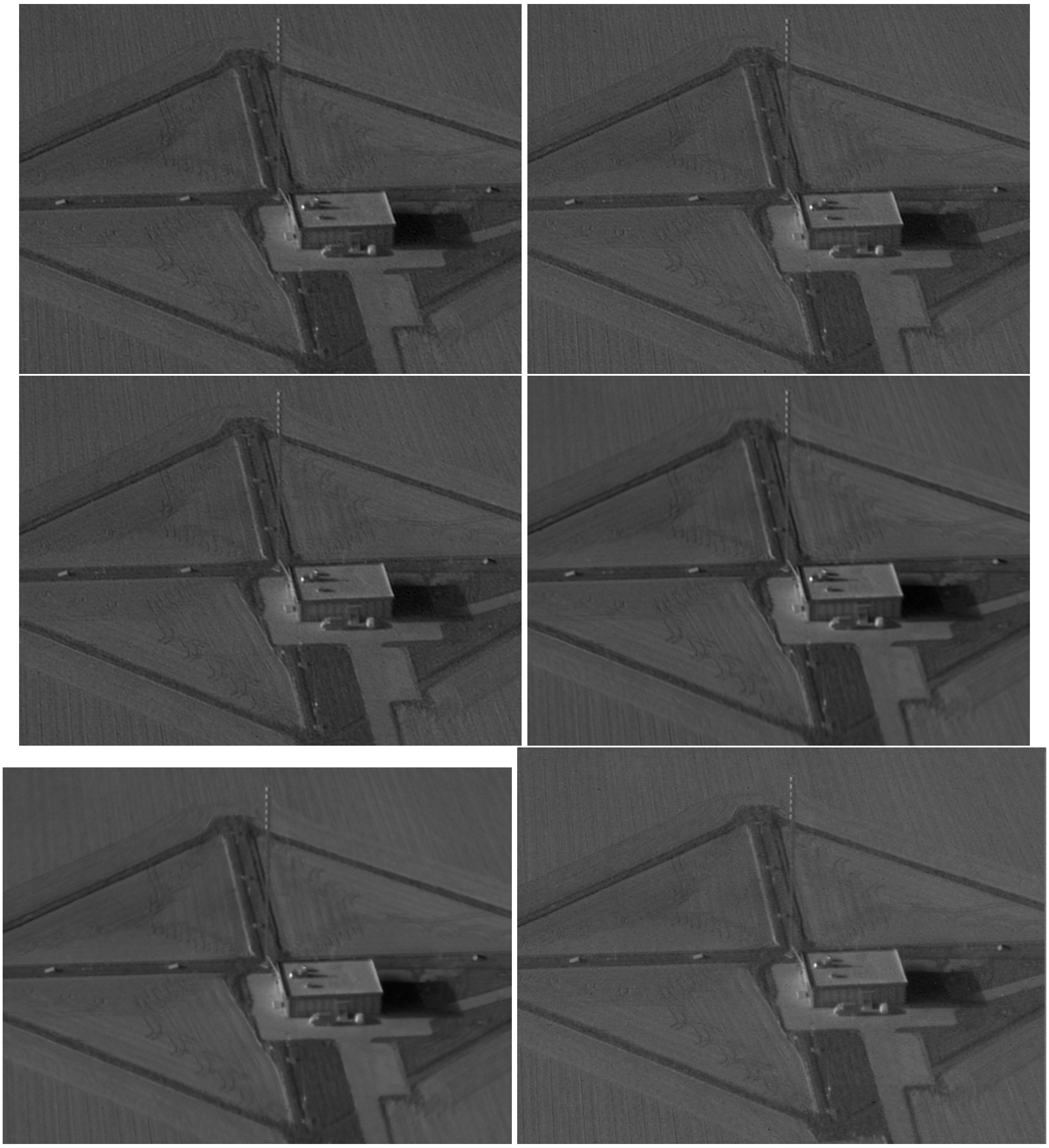


Figure 6.7: In lexicographic order : restored images for a regularization parameter of $0.001, 0.05, 0.1$ and 0.5, the median of the registered images, and the third observed image.

6.2.2 Sequence Enst09

In this section, we apply our blind superresolution algorithm on the 7 frames of sequence Enst-09.ptw, first frame is the 2^{nd} one of the sequence. Figure 6.8 shows the observed frames.

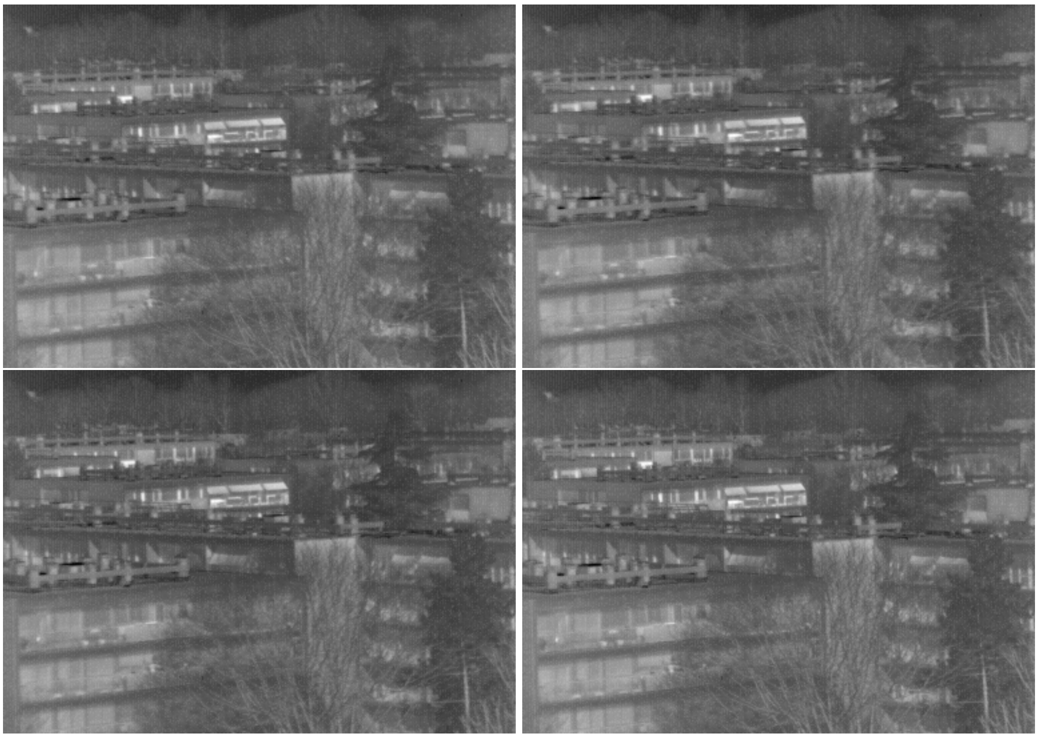


Figure 6.8: Frames 2, 4, 6 and 8 of sequence Enst-09.ptw

As we aim to evaluate a superresolved image, we use the filter identification method as presented in section 4 page 19, and the restoration method presented in 5 page 29 with a resolution increased by a factor 2 ($P = 2$).

But first, as the frames comes from a sequence with in-motion camera, we have to register all the frames relatively to a reference one : the middle one (the 4^{th}).
The images are registered using the following parameters:

- The motion is estimated with an accuracy up to an half pixel, using a spline interpolation method.
- The exhaustive block-matching is performed over a $(5, 5)$ research window. The frames are partitioned into blocks of size $(8, 8)$, and the matching criterion is evaluated onto blocks of the same size.
- We choose for parameter σ of the weighting function (2.16) the value 1.

The estimated motion model are :
between the first frame and the frame of reference :

$$A = \begin{pmatrix} 0.99919 & -0.00515 \\ -0.00072 & 0.99788 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.86813 \\ -2.53547 \end{pmatrix} \tag{6.6}$$

between the second frame and the frame of reference :

$$A = \begin{pmatrix} 0.99939 & 0.00198 \\ 0.00048 & 0.99805 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.30594 \\ -3.20452 \end{pmatrix} \tag{6.7}$$

between the third frame and the frame of reference :

$$A = \begin{pmatrix} 0.99988 & -0.00062 \\ -0.00000 & 0.99999 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.00221 \\ -0.13839 \end{pmatrix} \tag{6.8}$$

between the fifth frame and the frame of reference :

$$A = \begin{pmatrix} 0.99999 & -0.00088 \\ -0.00001 & 1.00032 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.04463 \\ -0.11082 \end{pmatrix} \tag{6.9}$$

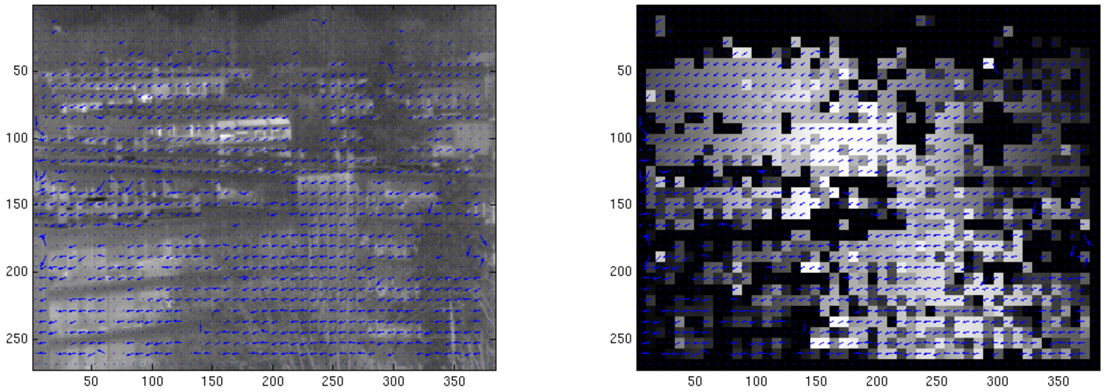


Figure 6.9: On left : the motion vectors estimated by the block-matching; On right : the influence of each block on the motion model estimation

between the sixth frame and the frame of reference :

$$A = \begin{pmatrix} 1.00118 & -0.00313 \\ -0.00031 & 1.00181 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.81031 \\ -1.70650 \end{pmatrix} \tag{6.10}$$

between the seventh frame and the frame of reference :

$$A = \begin{pmatrix} 1.00018 & 0.00107 \\ 0.00031 & 1.00014 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 0.98933 \\ -1.67958 \end{pmatrix} \tag{6.11}$$

Figure 6.9 shows, on the left, the motion vectors estimated by the block-matching, one for each block, and on the right, the influence of each vector on the estimation of the motion model : the darkest blocks are not taken into account in the estimation of the motion model; the lightest the blocks are, the most they influence the estimation. This image is provided by the last value of the weighting map w in the robust motion model estimation algorithm (see equation (2.14)).
Figure 6.10 shows the registered images.

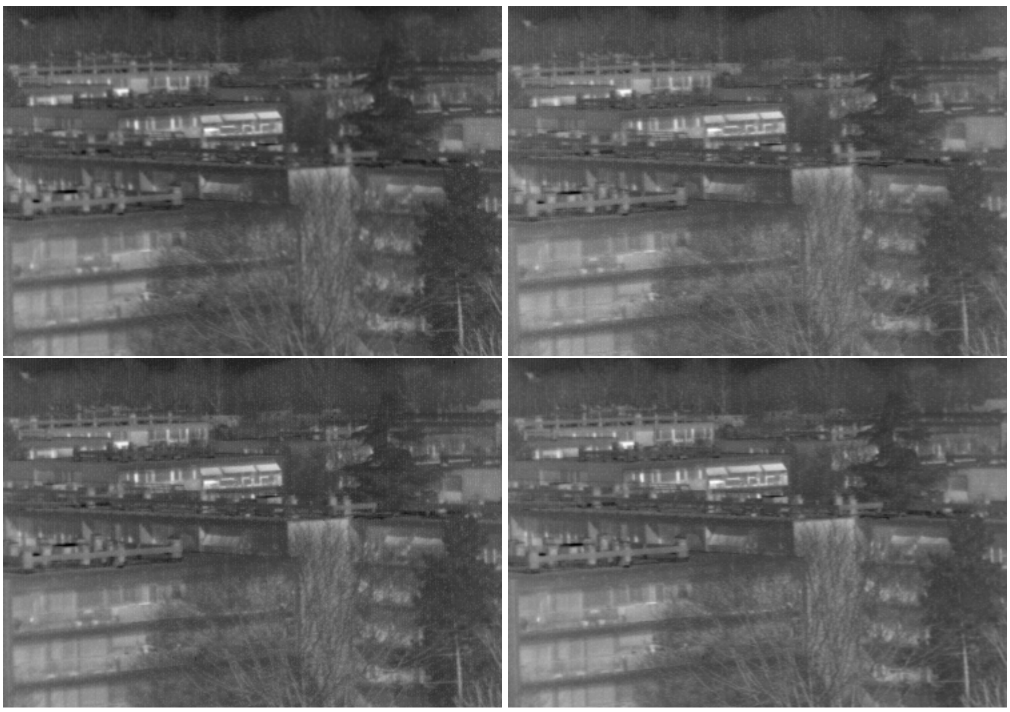


Figure 6.10: The registered images

The filter identification is provided following the method developed in section 4 page 19, assuming that the high-resolution filters are of size $(8, 8)$. We choose the criterion of regularity of the filters presented in section 4.3.1 page 24 to evaluate the mixing matrix. The resulting estimated filters are shown in Figure 6.11.

Finally, we estimate the superresolved image using the method detailed in section 5 page 29 with a resolution improved by a factor 2. For comparison purposes, we provide the median of the registered and then linearly interpolated observed images, and a linearly interpolated version of the image of reference (see Figure 6.12). Figure 6.13 shows the (visually) best superresolved image, for a regularization parameter of 0.7.

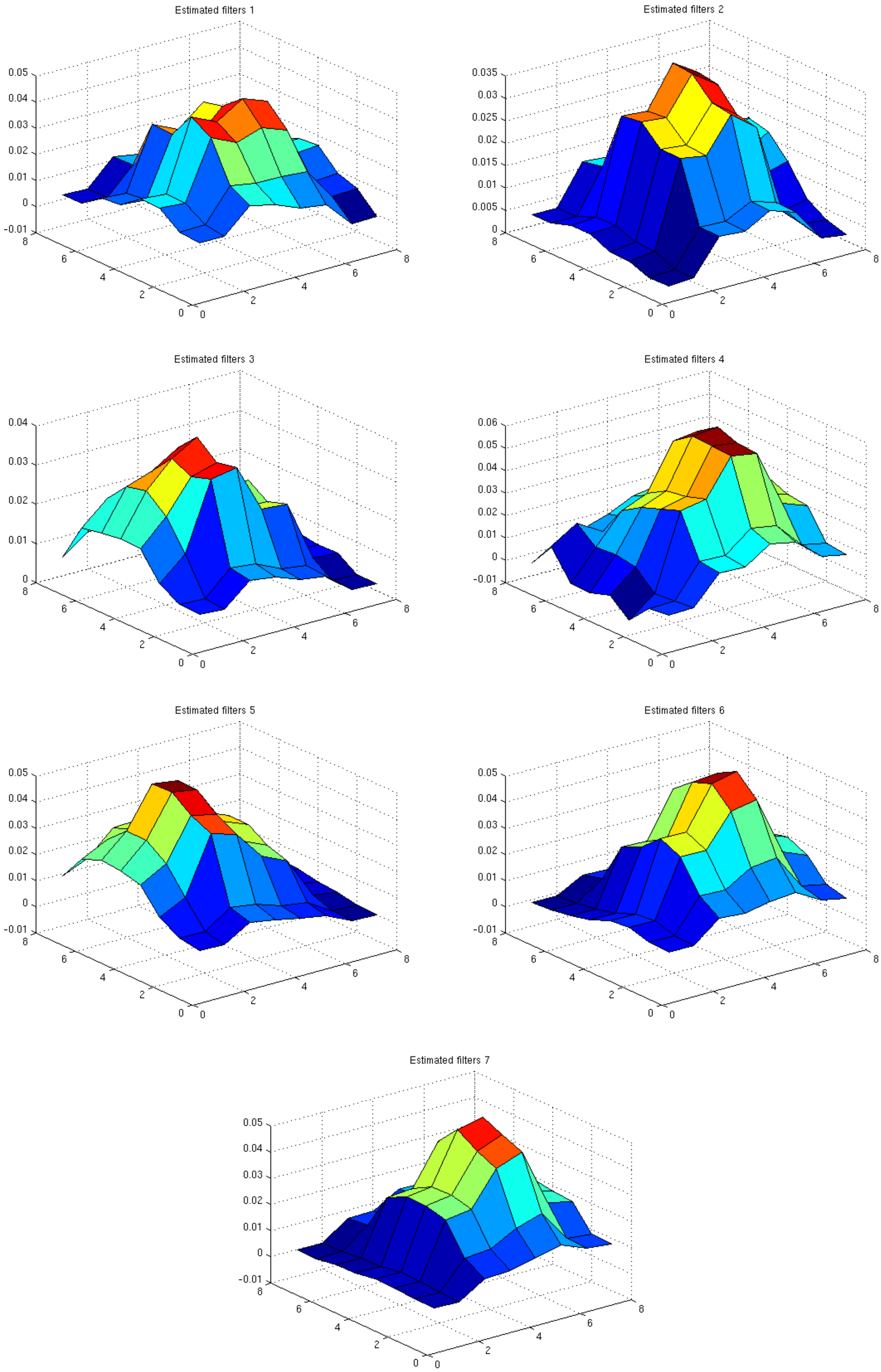


Figure 6.11: The estimated filters

Figure 6.12: Up : the median image of the registered and interpolated observed images, down : an interpolation of the image of reference

6.2.3 Sequence **Moon**

We apply our blind superresolution algorithm on 32 images of the moon of size $(150, 150)$. Figure 6.14 shows some of the observed frames.

As we aim to evaluate a superresolved image, we use the filter identification method such as presented in section 4 page 19, and the restoration method presented in 5 page 29 with a resolution increased by a factor 4 ($P = 4$).

But first, as the moon moved between each acquisition, we have to register all the frames relatively to a reference one : the middle one (the 16^{th}).
The images are registered using the following parameters:

Figure 6.13: Our restored image

Figure 6.14: Some of the observed images (images 1, 6, 11, 16, 21, 26, 31 and 32)

- The motion is estimated with an accuracy up to a quarter pixel, using a bilinear interpolation method.
- The exhaustive block-matching is performed over a $(15, 15)$ research window. The frames are partitioned into blocks of size $(8, 8)$, and the matching criterion is evaluated onto blocks of the same size.
- We choose for parameter σ of the weighting function (2.16) the value 1.

The block-matching algorithm runs in 231 s. As there are two many images, we provide results of the motion estimation model only for the first and the last images of the set:
between the first frame and the reference one:

$$A = \begin{pmatrix} 1.00063 & -0.00100 \\ 0.00126 & 1.00016 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} 8.96587 & -5.10574 \end{pmatrix} \tag{6.12}$$

between the last frame and the reference one:

$$A = \begin{pmatrix} 1.00106 & 0.00337 \\ -0.00220 & 1.00057 \end{pmatrix} \text{ and } \quad T = \begin{pmatrix} -0.73827 & 11.90123 \end{pmatrix} \tag{6.13}$$

The filter identification is provided following the method developed in section 4 page 19, assuming that the high-resolution filters are of size $(12, 12)$. We choose the criterion of regularity of the filters presented in section 4.3.1 page 24 to evaluate the mixing matrix. The resulting estimated filters are shown in Figure 6.15.

Finally, we estimate the superresolved image using the method detailed in section 5 page 29 with a resolution improved by a factor 4. For comparison purposes, we provide the median of the registered and then linearly interpolated observed images (see Figure 6.16). Figure 6.17 shows the (visually) best superresolved image, for a regularization parameter of 0.06.

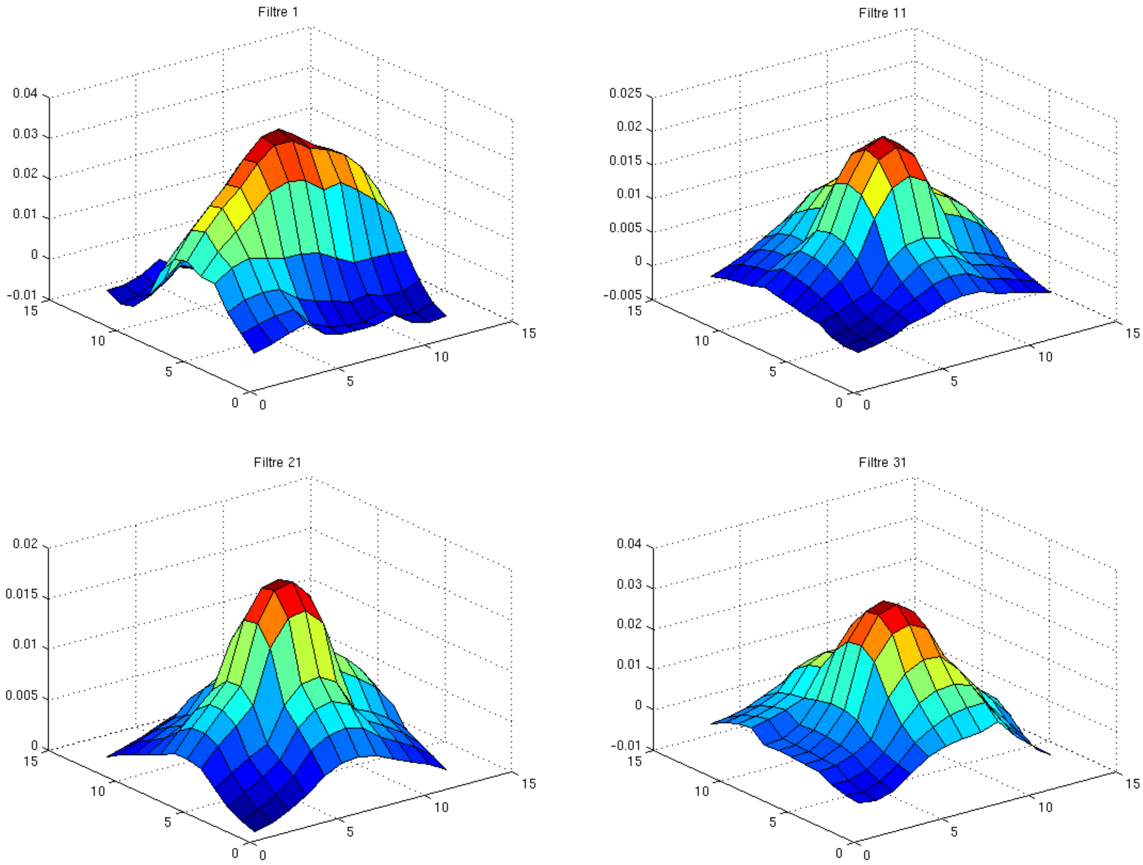


Figure 6.15: Some of the estimated filters (images 1, 11, 21 and 31)

Figure 6.16: The median of the registered and then bilinearly interpolated observed images

Figure 6.17: The restored image (P=4)

6.3 Pre-denoising

6.3.1 Estimation of the fixed pattern noise

Noise contained in infrared images can be harmful to filter identification and restoration. We propose to coarsely denoise the infrared images beforehand. This noise changes during the acquisition of the video, but it can be seen as a fixed pattern noise on several following images. To estimate the fixed pattern noise, we assume that this kind of noise is additive, and take advantage of its temporal redundancy. We start from the simple idea that this noise is the only low-level information shared by several following images taken by an in motion camera.

The algorithm is the following:

1. First, we estimate smoothed images from the original images, by applying a mean filter for example.
2. Then we estimate the difference between the original images and the smoothed ones (see figure 6.18 page 47). These difference images contain informations related to high-frequency components of the original images, such as contours or noise. If the original images are dissimilar enough each others, contours are not in the same place from one image to another, random noises neither, and the only redundant information at a given place is the fixed pattern noise one.
3. The median of the difference images provides a good estimation of the fixed pattern noise, as it removes random noise and contours, even if several images are quite similar, provided that the total number of images is large enough (see figure 6.19 page 47).
4. The images are denoised by subtracted the fixed pattern noise, i.e. the median of the difference (see figure 6.20 page 48

Figure 6.18 page 47 shows two examples of original infrared images. We can easily see the fixed pattern noise on the sky part of the second image (ENST04). This figure illustrates the second step of the algorithm and presents two original images, their smoothed version, and the difference between the smoothed image and the original one.

Figure 6.19 page 47 illustrates the third step of the algorithm and shows the fixed pattern noise estimated from 8 images extracted from 8 different sequences acquired with the same camera from different point of view. As the acquired scene is dissimilar enough from one image to another, 8 frames are enough to compute a satisfying estimation of the fixed pattern noise.

Figure 6.20 page 48 shows two results of denoising using this method, and compares the original images with the denoised ones. We can see that this method is quite simple but provides a good estimation of the fixed pattern noise, and preserves the contours of the images.

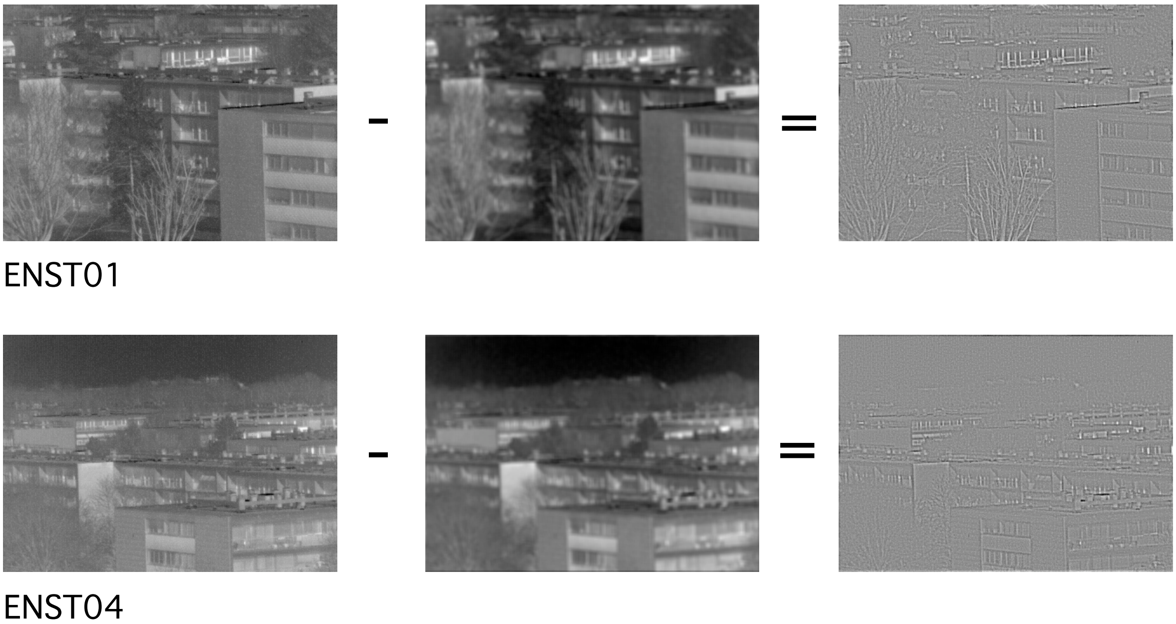


Figure 6.18: Two original images, their smoothed version and the difference between them

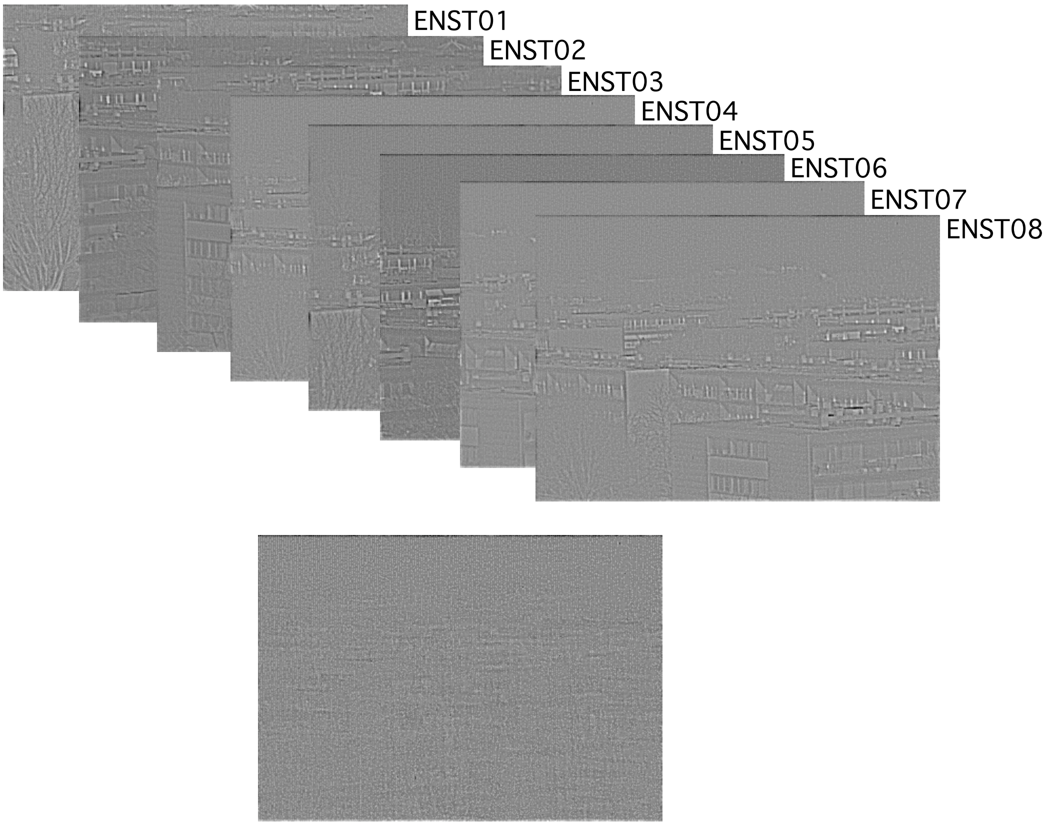


Figure 6.19: The median of the difference images

6.3.2 Sequence enst10

We apply our blind superresolution algorithm on 6 images of sequence ENST10 from the 992^{th} image of the sequence. Figure 6.21 shows the observed images.

These images were acquired by an infrared bolometer sensor and contain a fixed pattern noise. We perform an additional denoising step beforehand the filter identification step, following the method described in section 6.3.1 page 46.
We estimate the fixed pattern noise on 96 images of the sequence, from the 902^{th} image to the 997^{th} one, and denoise the

Figure 6.20: Comparison between original images and the denoised ones

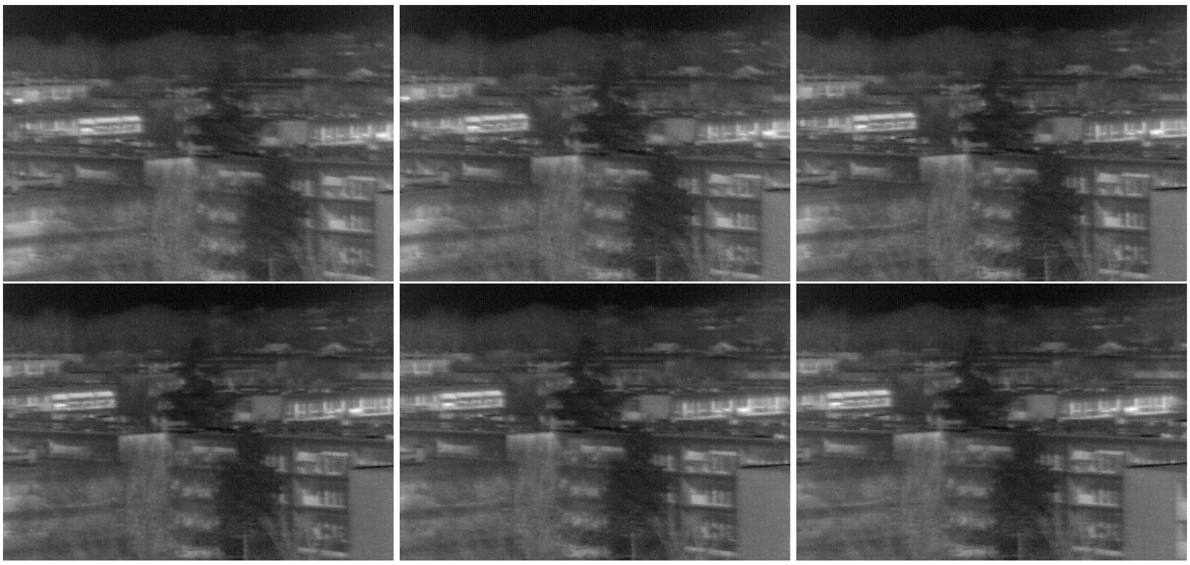


Figure 6.21: Observed images

observed images. Figure 6.22 page 49 shows the denoised observed images.

As we aim to evaluate a superresolved image, we use the filter identification method such as presented in section 4 page 19, and the restoration method presented in 5 page 29 with a resolution increased by a factor 2 ($P = 2$).

But first, as the camera is in motion, we have to register all the frames relatively to a reference one : the middle one (the 4^{th}).
The images are registered using the following parameters:

- The motion is estimated with an accuracy up to a quarter pixel, using a bilinear interpolation method.
- The exhaustive block-matching is performed over a $(30, 30)$ research window. The frames are partitioned into blocks

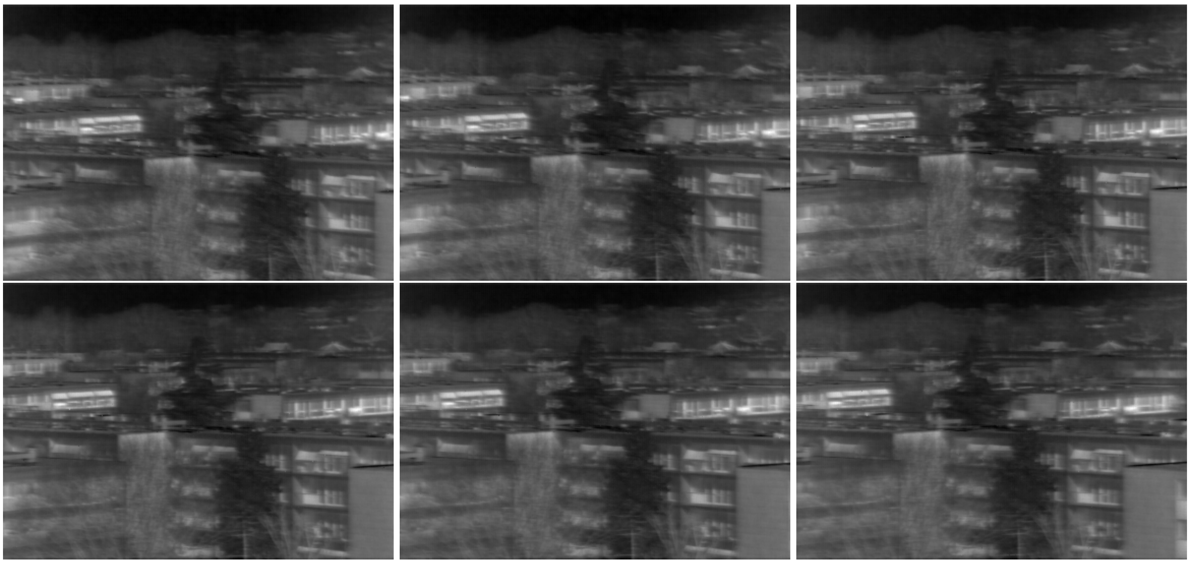


Figure 6.22: Denoised observed images

of size $(16, 16)$, and the matching criterion is evaluated onto blocks of the same size.

- We choose for parameter σ of the weighting function (2.16) the value 0.1.

The motion models are the following:
between the first frame and the reference one:

$$A = \begin{pmatrix} 0.99540 & 0.00044 \\ -0.00274 & 0.99938 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 3.23538 & 26.01583 \end{pmatrix} \tag{6.14}$$

between the second frame and the reference one:

$$A = \begin{pmatrix} 0.99627 & 0.00022 \\ -0.00326 & 0.99939 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 1.05917 & 16.04571 \end{pmatrix} \tag{6.15}$$

between the third frame and the reference one:

$$A = \begin{pmatrix} 1.00029 & -0.00008 \\ -0.00127 & 0.99995 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} 0.21819 & 7.66263 \end{pmatrix} \tag{6.16}$$

between the fifth frame and the reference one:

$$A = \begin{pmatrix} 0.99839 & -0.00001 \\ -0.00326 & 0.99996 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} -0.70913 & -8.27551 \end{pmatrix} \tag{6.17}$$

between the sixth frame and the reference one:

$$A = \begin{pmatrix} 1.00057 & -0.00017 \\ -0.00519 & 1.00047 \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} -1.23341 & -17.85070 \end{pmatrix} \tag{6.18}$$

The filter identification is provided following the method developed in section 4 page 19, assuming that the high-resolution filters are of size $(8, 8)$. We choose the criterion of regularity of the filters presented in section 4.3.1 page 24 to evaluate the mixing matrix.

Finally, we estimate the superresolved image using the method detailed in section 5 page 29 with a resolution improved by a factor 2. For comparison purposes, we provide the mean of the registered and then linearly interpolated observed images (see Figure 6.23). Figure 6.24 shows the (visually) best superresolved image, for a regularization parameter of 0.06.

Figure 6.23: The mean of the registered and then bilinearly interpolated observed images

Figure 6.24: The restored image (P=2)

6.4 Panorama

Sequences like ENST09 or ENST10 are in-motion camera sequences and contain mainly blurred images. These sequences pan across the same general scene, essentially buildings.
We have shown results of the superresolution algorithm obtained on these sequences, and proposed to compare these results to the median of the interpolated and registered observed images, or to an interpolated image, as we had no other reference to compare our results.
To overcome this problem, we acquired 8 sequences overlapping the same scene, with a still camera. Then, for comparison purposes, we estimate a panorama from 6 of the 8 sequences.

The algorithm estimating the panorama proceeds as follows:

- first, we denoise the observed images using the algorithm described in section 6.3.1 page 46;
- then, we register each frame into the coordinate of a chosen reference one; the registration algorithm is detailed in chapter 2 page 7

- finally, we merge all the information coming from the different images, we called this step: image blending or fusion.

We refer the reader to the relevant sections for more details on the first and second steps of the algorithm, and we focus here on the third step.

Once the images are registered in the same reference coordinates, we have several different informations at our disposal for each pixel of the scene, as the images were acquired at different times and present changes in illumination from one to another (see figure 6.25 page 51), or could be slightly misregistered. The quality of the panorama depends highly on the way these different informations are merged.

Figure 6.25: Denoised images used for the estimation of the panorama

The first ideas are to assign to a pixel the mean or the median of the intensities of each image containing this pixel. But these methods are not robust to changes in illumination between images, and the transition between overlapping areas are rough. Figure 6.26 page 52 shows a panorama obtained using a mean blending function, and figure 6.27 page 52 a panorama obtained using a median blending function. These panoramas are not visually satisfying but they allow us to know if the images are well registered, as we can see the overlapping areas.

In order to have a smoother transition between overlapping areas, we assign a weight function to each image before merging their contribution. This weight function is separable and varies linearly from 1 to 0, from the center of the image towards the edges. Figure 6.28 page 53 shows a panorama obtained using this merging method. This panorama is obviously better than the previous ones, as we do not see any more the transition between the frames.

Figure 6.26: Panorama using a mean merging function

Figure 6.27: Panorama using a median merging function

Figure 6.28: Panorama using a weight merging function

Chapter 7

Conclusion

In this work, we showed how the subspace method may be applied to image superresolution. We showed that this method is intrinsically ambiguous when presented with multiple sources which are, in fact, subsamples of the one same image. We showed how statistical properties of images can be used to disambiguate the problem and achieve a satisfactory recovery of the filters and of the original image. The advantage of using this method is that it can be applied to a wide range of filters without further assumption than their smoothness. In future work, one may want to apply other types of regularization to the image or the filters. The most promising lead is the TV regularization [15] which would be available as a usable technology very soon. The other possibility of improvement is the extension to the case where the made algebraic assumptions fail to be true, in such cases subspace method happens to be very unstable. We may apply the ideas presented here to stabilize the problem.

Bibliography

[1] L. Tong, G. Xu, and T. Kailath, "A new approach to blind identification and equalization of multipath channels," in *Asilomar Conference*, (Pacific Grove, CA.), pp. 856–860, 1991.

[2] K. Abed-Meraim and Y. Hua, "Blind identification of multi-input multi-output system using minimum noise subspace," *IEEE Transactions on Signal Processing*, vol. 45, no. 1, pp. 254–258, 1997.

[3] A. Gorokhov and P. Loubaton, "Subspace-based techniques for blind separation of convolutive mixtures with temporally correlated sources," *IEEE Transactions on Circuits and Systems-I: fundamental theory and applications*, vol. 44, no. 9, pp. 813–820, 1997.

[4] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel fir filters," *IEEE Transactions on Signal Processing*, vol. 43, no. 2, pp. 516–525, 1995.

[5] Wirawan, K. Abed-Meraim, H. Maître, and P. Duhamel, "Blind multichannel image restoration using subspace based method," in *International conference on Acoustics, Speech, and Signal Processing*, vol. 5, (Hong Kong.), pp. 9–12, Avril 6-10 2003.

[6] F. Dufaux and F. Moscheni, *Segmentation-based motion estimation for second generation video coding techniques*, ch. Video coding: a second generation approach. Kluwer Academic, 1996.

[7] R. Li, B. Zeng, and M. Liou, "A new threestep search algorithm for block estimation," *IEEE Transactions on Circuits Systems and Video Technology*, vol. 4, pp. 438–442, 1994.

[8] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block matching motion estimation," *IEEE Transactions on Image Processing*, vol. 9, no. 2, pp. 287–290, 2000.

[9] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 2, pp. 298–311, 1997.

[10] G. Aubert and P. Kornprobst, *Mathematical problems in image processing. Partial differential equations and the calculus of variations*. Applied Mathematical Sciences, Springer Verlag, 2001.

[11] S. Geman and D. Mc Clure, "Bayesian image analysis: an application to single photon emission tomography," in *Statistical Computing Section*, pp. 12–18, 1985.

[12] Y.-L. You and M. Kaveh, "A regularization approach to joint blur identification and image restoration," *IEEE Transactions on Image Processing*, vol. 5, no. 3, pp. 416–428, 1995.

[13] T. Chan and C.-K. Wong, "Total variation blind deconvolution," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998.

[14] R. H. Chan, F. Chan, and C.-K. Wong, "Cosine transform based precondioners for total variation deblurring," *IEEE Transactions on Image Processing*, vol. 8, no. 10, pp. 1472–1478, 1999.

[15] J. Darbon and M. Sigelle, "Image restoration with discrete constrained total variation," *Journal of Mathematical Imaging and Vision*, vol. 26, pp. 261–291, 2006.

[16] A. Srivastava, A. Lee, E. Simoncelli, and S.-C. Zhu, "On advances in statistical modeling of natural images," *Journal of Mathematical Imaging and Vision*, vol. 18, pp. 17–33, 2003.

Appendix A

Notations

Signals or images :

- X :
  - signal : stacks the L output signals of size N and is of size $(LN, 1)$:

$$X = \begin{pmatrix} X^1 \\ \vdots \\ X^L \end{pmatrix} \text{ where } X^l = \begin{pmatrix} x_0^l & \dots & x_{N-1}^l \end{pmatrix}^T \tag{A.1}$$

  - image : stacks the L observed images X^l, for $l = 1$ to L, more precisely, a vectorized formulation of a processing windowed area, of size $(N_y\ ,\ N_x)$, extracted from the observed images :

$$X^l = [x^l(N_y - 1, N_x - 1)\ x^l(N_y - 2, N_x - 1)\ \cdots\ x^l(0, 0)]^T \tag{A.2}$$

- D :
  - signal : is the unknown original signal of size $(N + M - 1, 1)$
  - image : is a vectorized formulation of the related windowed area of the original image of size $(N_y + M_y - 1, N_x - M_x - 1)$,

$$D = [d(N_y + M_y - 2, N_x + M_x - 2) \cdots\ d(0, 0)]^T \tag{A.3}$$

Subsampled signal or images :

- D_p : (signal) for $p = 0$ to $P - 1$, denotes a subsampled component of D

$$D_p = [d_p\ d_{p+P} \dots d_{p+(n+m-2)P}]^T \qquad (n + m - 1, 1) \tag{A.4}$$

- D_{p_1,p_2} : (images) is a vectorized form of

$$D_{p_1,p_2} = \begin{pmatrix} d_{p_1,p_2} & d_{p_1,p_2+P} & \cdots & d_{p_1,p_2+P(s_x-1)} \\ \vdots & \vdots & & \vdots \\ d_{p_1+P(s_y-1),0} & d_{p_1+P(s_y-1),P} & \cdots & d_{p_1+P(s_y-1),p_2+P(s_x-1)} \end{pmatrix} \tag{A.5}$$

  for all $p_1, p_2 = 0 : P - 1$, where $s_y = n_y + m_y - 1$ and $s_x = n_x + m_x - 1$.
  Note that we use the same notation for the vectorized and the matrix forms of D_{p_1,p_2}

Filters:

- H : the filters coefficients

$$H = \begin{pmatrix} H^1 \\ \vdots \\ H^L \end{pmatrix} \tag{A.6}$$

- H^l :

- 1D : is a filter of size $(M, 1)$:

$$H^l = [h_0^l \dots h_{M-1}^l]^T \tag{A.7}$$

- 2D : is a filter of size (M_y, M_x)

$$H^l = \begin{pmatrix} h^l(0,0) & \dots & h^l(0, M_x - 1) \\ \vdots & & \vdots \\ h^l(M_y - 1, 0) & \dots & h^l(M_y - 1, M_x - 1) \end{pmatrix} \tag{A.8}$$

- H_p^l : (1D) one of the P subsampled components of the filter H^l

$$H_p^l = \left[\, h_p \; h_{p+P} \; \dots \; h_{p+(m-1)P} \,\right] \tag{A.9}$$

- H_{p_1,p_2}^l : (2D) one of the P^2 polyphase components of the filter H^l

$$H_{p_1,p_2}^l = \begin{pmatrix} h_{p_1,p_2}^l & \dots & h_{p_1,p_2+(m_x-1)P}^l \\ h_{p_1+P,p_2}^l & \dots & h_{p_1+P,p_2+(m_x-1)P}^l \\ \vdots & & \vdots \\ h_{p_1+(m_y-1)P,p_2}^l & \dots & h_{p_1+(m_y-1)P,p_2+(m_x-1)P}^l \end{pmatrix} \tag{A.10}$$

- $\mathbb{H}$: (2D) a formulation of H in terms of its polyphase components

$$\mathbb{H} = \begin{pmatrix} \mathbb{H}_{0,0} & \dots & \mathbb{H}_{P-1,P-1} \end{pmatrix} \tag{A.11}$$

where

$$\mathbb{H}_{p_1,p_2} = \begin{pmatrix} H_{p_1,p_2}^1 \\ \vdots \\ H_{p_1,p_2}^L \end{pmatrix} \tag{A.12}$$

Filtering matrices:

- $\mathcal{H}$:
  - signal : stacks the L Toeplitz matrices $\mathcal{H}^l$:

$$\mathcal{H} = \begin{pmatrix} \mathcal{H}^1 \\ \vdots \\ \mathcal{H}^L \end{pmatrix} \qquad (LN, N + M - 1) \tag{A.13}$$

where $\mathcal{H}^l$, for $l = 1$ to L, is the filtering matrix associated to the filter H^l :

$$\mathcal{H}^l = \begin{pmatrix} h_0^l & \dots & h_{M-1}^l & 0 \\ \ddots & & \ddots & \\ 0 & h_0^l & \dots & h_{M-1}^l \end{pmatrix} \qquad (N, N + M - 1) \tag{A.14}$$

  - image : stacks the L block-Toeplitz filtering matrices $\mathcal{H}^l$ associated with each filters H^l :

$$\mathcal{H}^l = \begin{pmatrix} \mathcal{H}_0^l & \cdots & \mathcal{H}_{M_x-1}^l & 0 \\ \ddots & & \ddots & \\ 0 & \mathcal{H}_0^l & \cdots & \mathcal{H}_{M_x-1}^l \end{pmatrix} \tag{A.15}$$

where $\mathcal{H}_j^l$ is a Toeplitz matrix of size $(N_y \ , \ N_y + M_y - 1)$ associated to the j^{th} column of H^l :

$$\mathcal{H}_j^l = \begin{pmatrix} h^l(0,j) & \dots & h^l(M_y - 1, j) & 0 \\ \ddots & & \ddots & \\ 0 & h^l(0,j) & \dots & h^l(M_y - 1, j) \end{pmatrix} \tag{A.16}$$

$\mathcal{H}^l$ contains N_x rows of blocks and $N_x + M_x - 1$ columns of blocks of size $(N_y \ , \ N_y + M_y - 1)$. $\mathcal{H}$ is of size $(LN_yN_x \ , \ (N_y + M_y - 1)(N_x + M_x - 1))$

- $\mathcal{H}_p^l$: (signal) the filtering matrix associated to H_p^l

$$\mathcal{H}_p^l = \begin{pmatrix} h_p^l & h_{p+P}^l & \dots & h_{p+(m-1)P}^l & & 0 \\ & \ddots & & & \ddots & \\ 0 & & h_p^l & h_{p+P}^l & \dots & h_{p+(m-1)P}^l \end{pmatrix} \qquad (n, n+m-1) \tag{A.17}$$

- $\mathcal{H}_{p_1,p_2}^l$: (image) is the block-Toeplitz matrix of size $(n_y n_x, s_y s_x)$ associated to the filter H_{p_1,p_2}^l
- $\mathcal{H}_{LR}$: the filtering matrix of the MIMO system
  - signal :

$$\begin{pmatrix} \mathcal{H}_0^1 & \dots & \mathcal{H}_{P-1}^1 \\ \vdots & & \vdots \\ \mathcal{H}_0^L & \dots & \mathcal{H}_{P-1}^L \end{pmatrix} \tag{A.18}$$

  - image :

$$\begin{pmatrix} \mathcal{H}_{0,0}^1 & \dots & \mathcal{H}_{P-1,P-1}^1 \\ \vdots & & \vdots \\ \mathcal{H}_{0,0}^L & \dots & \mathcal{H}_{P-1,P-1}^L \end{pmatrix} \tag{A.19}$$

- $\mathcal{H}_{p_1,p_2}$: (image) a block column of the filtering matrix $\mathcal{H}_{LR}$

$$\mathcal{H}_{p_1,p_2} = \begin{pmatrix} \mathcal{H}_{p_1,p_2}^1 \\ \vdots \\ \mathcal{H}_{p_1,p_2}^L \end{pmatrix} \tag{A.20}$$

Dépôt légal : 2008 – 1er trimestre
Imprimé à l'Ecole Nationale Supérieure des Télécommunications – Paris
ISSN 0751-1345 ENST D (Paris) (France 1983-9999)

Ecole Nationale Supérieure des Télécommunications

Groupe des Ecoles des Télécommunications - membre de ParisTech

46, rue Barrault - 75634 Paris Cedex 13 - Tél. + 33 (0)1 45 81 77 77 - www.enst.fr

Département TSI