

A simple super-resolution algorithm

Paolo D'Apice, Lawrence Iviani

June 19, 2007

Abstract

Super-resolution reconstruction produces one high-resolution image from a set of low-resolution degraded images. The objective of our project is develop an algorithm that uses the basic concepts of super-resolution. Different variants of a main algorithm are proposed and tested. Finally, possible optimizations and suggestions for a future development are presented.

Copyright © 2007 Paolo D'Apice and Lawrence Iviani. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

Super-resolution (SR) are techniques that in some way enhance the resolution of an imaging system. SR is typically used for breaking the diffraction-limit of systems or the limit of digital imaging sensors, for scientific data visualization (spatial or medical images) and also for video surveillance applications.

The main idea behind SR is to exploit several slightly different low-resolution (LR) images in order to obtain one enhanced-resolution image. In the general case, LR images are obtained applying some transformations, such as warp, rotation or translation, or using different points of view of the same scene. Informations contained in the applied transformations are hence exploited in order to construct the SR image.

There exist both single-frame and multiple-frame variants of SR, based either on frequency or space domain approach (see [2] for a review about SR). Spatial domain methods are based on single image restoration techniques, that basically try to use some a priori knowledge about the to-be-restored image. Such techniques aim to reduce the set of the possible solutions. In fact the SR problem is an ill-posed problem and has no unique solution (see [1], [8]), hence the importance of reducing solution space. Frequency domain approaches have the big advantage of low computing complexity, and aim to enhance the Signal-to-Noise-Ratio (see [9]).

Our objective is the application of basical concepts of SR on simple problems where the chain of applied tranformations consists in only a translation.

2 Problem and proposed solution

In the most general case, a scene obtained with an imaging device seems to be warped at the camera lens, blurred and discretized at the sensor, resulting in an outcome frame as shown in the model in Figure 1 (see [1], [4], [5]).

Since usually a set of acquired images is used, it is possible modelize the problem as depicted in the block diagram in Figure 2.

The model can be also described by the set of equations

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} = \begin{bmatrix} D_1 C_1 F_1 \\ \vdots \\ D_N C_N F_N \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_N \end{bmatrix} = \begin{bmatrix} H_1 \\ \vdots \\ H_N \end{bmatrix} \mathbf{x} + \mathbf{e} = H\mathbf{x} + \mathbf{e}$$

where F_k represents the geometric warp, C_k the blur and D_k the decimation process. \mathbf{y}_k , \mathbf{x} and \mathbf{e}_k represent respectively the LR images, the HR image and the additive noise due to the acquiring system, expressed in lexicographic order.

A SR algorithm must take into account the sequence of transformations and exploit them in order to combine the LR images and obtain the SR image. For our purposes, we will consider only the case in which the transformation sequence is made up of only a translation.

Here is, in summary, the description of our algorithm

1. creation of starting set of LR images
2. images registration
3. image reconstruction
4. optional image enhancement

In the following paragraphs a detailed description of each step is provided.

2.1 Starting set of images

The set of LR images is obtained decimating a HR image. Decimation filter is designed in such a way that the filtered image contains some alias. Alias is needed for two reasons: because informations gained in the SR-image are embedded in the LR images as alias, and because it must be taken into account the fact that the acquiring system introduces some alias. In practice, it is possible obtain an alias-like effect by simply filtering out high frequencies.

Filtered image is then downsampled by taking pixels in different grid sets, as shown in Figure 3. This generation process simulates the situation in which a sequence of pictures are taken with a very slowly moving camera. In fact each LR image seems to be a translated version of the others.

Translation vectors are randomly generated among all the possible $(D^2 - 1)$ combinations, where D is the downsampling factor. For example, with reference to Figure 3, downsampleig by a factor 2 it is possible obtain 2^2 LR images with $(2^2 - 1)$ translation vectors $[0 1]$, $[1 0]$ and $[1 1]$ in addition to the trivial $[0 0]$.

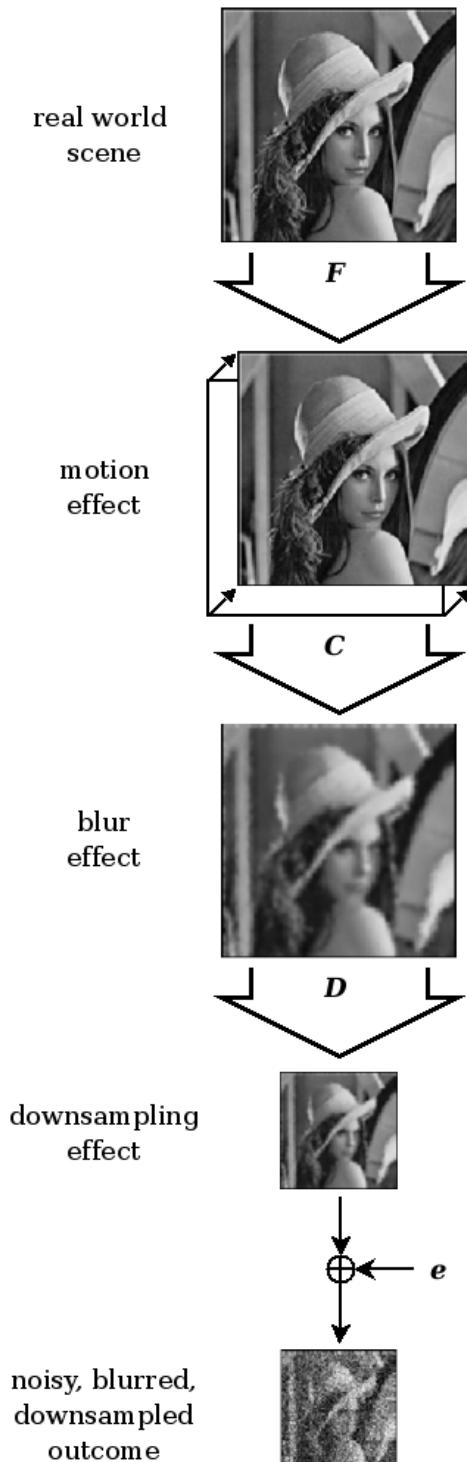


Figure 1: Problem modelization

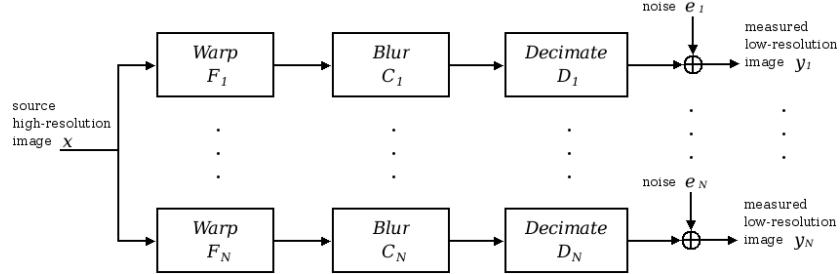


Figure 2: Degradation model for the SR restoration problem

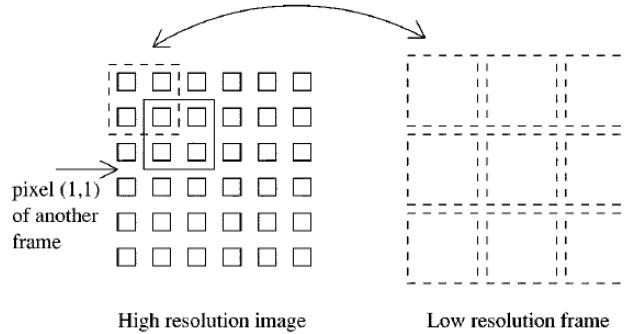


Figure 3: Generation of LR images

2.2 Image Registration

Once the set of LR images is created, a registration process is required in order to align the images. The process is made up of two stages:

interpolation LR images are upsampled to the desired HR size

alignment translation vectors are estimated and images are aligned to a common resolution grid.

The execution order of the two stages affects in a significative way the final result. A detailed discussion about this topic is provided in the Section 3.3.

Interpolation

Since simply upsampling LR images is not enough because no extra information is added (Fig. 4), an interpolation is performed.

Most of existing SR techniques perform alignment after the interpolation stage. In this way it is possible take into account original sub-pixel shifts. In fact a half pixel shift can be transformed into a two pixels shift just interpolating by a factor $U = 4$. On the other hand, performing interpolation after translation estimation lead to a better perceived quality in some cases, but to a less accurate registration. This is because registering LR image keep some alias that reduces some artifacts depending on the SR algorithm, even introducing some other artifacts due the less accurate registration.

A	B	C
D	E	F
G	H	I

A	0	0	B	0	0	C	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
D	0	0	E	0	0	F	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
G	0	0	H	0	0	I	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Figure 4: Image upsampling with factor $U = 3$

So, depending on the execution order, two different interpolation methods are used:

- *nearest-neighbor*, simple method that introduces needed alias
- *bicubic*, a more precise method that does not introduce alias as much as nearest-neighbor.

Translation estimation

This step is performed in the frequency domain and makes explicit use of aliasing that exists in each LR image (also because of sub-pixel shifts) to reconstruct a HR image.

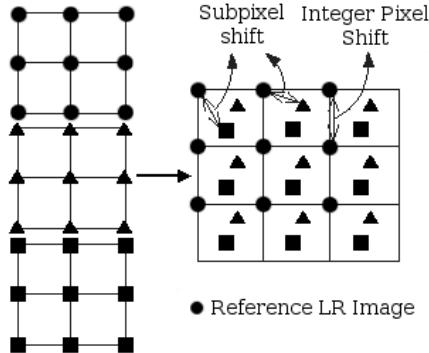


Figure 5: Image registration

The frequency domain approach used in this stage is based on the following three properties:

- the shifting property of the Fourier transform
- the aliasing relationship between the continuous Fourier transform (CFT) of the original HR image and the discrete Fourier transform (DFT) of observed LR images
- the assumption that the original HR image is bandlimited

By mean of these properties it is possible formulate an equation system that relates the aliased DFT coefficients of the observed LR images to a sample of the CFT of an unknown HR image.

For example, let us consider two monodimensional LR signals sampled below the Nyquist sampling rate. From the above three principles, the aliased LR signals can be decomposed into the unaliased HR signal as shown in Figure 6 (refer to [4] for a detailed explanation).

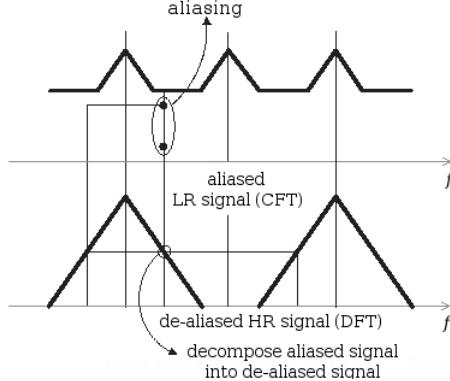


Figure 6: Aliasing relationship between LR image and HR image

Moreover, according to [2] and [9], since a global translation is assumed, it is possible to estimate shifts between images using the phase difference of spectra at low frequency bins.

Spectra of each LR image $\mathbf{x}_k (k = 2 \dots N)$ is compared with the spectrum of a reference LR image \mathbf{x}_1 (the first image of the set) using the phase difference defined as $\angle(\mathbf{X}_k / \mathbf{X}_1)$. Since most of the energy is packed at low frequencies, the comparison is made only on the low-frequency bins between DC and the cutoff frequency $\mathbf{u}_c = (u_c, v_c)$. So, for each frequency \mathbf{u} in $(-\mathbf{u}_c, \mathbf{u}_c)$ it is possible write a linear equation describing the plane through the computed phase difference with unknown slope Δq_k .

Shift vectors are hence the parameters Δq_k computed as the least squares solution of the obtained equations (refer to [9] for further details). Once translation vector are computed, the images are aligned by applying a reverse transformation.

2.3 Super-resolution image reconstruction

In our proposed SR algorithm, each SR pixel is obtained combining the corresponding pixels of the set of registered LR images.

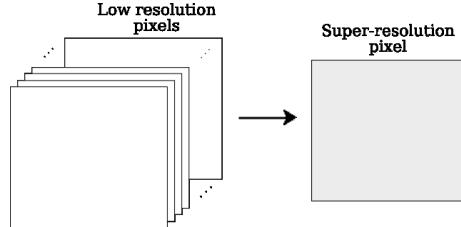


Figure 7: SR pixel obtained by combination of LR pixels

Basically, one can choose among different possible ways for combining pixels. We propose three simple methods, explained below.

Mean

In order to produce the final SR pixel, a mean filter is used:

$$\mathbf{s}_{mean}(x, y) = \frac{1}{N} \sum_{k=1}^N \tilde{\mathbf{y}}_k(x, y) \quad \forall (x, y).$$

So each pixel of the SR image \mathbf{s} at the position (x, y) is the average value of all the corresponding pixels of the registered images $\tilde{\mathbf{y}}_k$.

Median

The second proposed method is just a variation of the first one. In fact, instead of using a mean filter, a median filter is used:

$$\mathbf{s}_{median}(x, y) = \text{median} [\tilde{\mathbf{y}}_k(x, y)]_{k=1}^N \quad \forall (x, y).$$

As a result, the SR image obtained with these two methods are perceived as similar (see Section 3).

Spectral mean

This method is based on a frequency domain approach. Instead of applying an inverse transformation the set of LR images, the estimated translation vectors are used to compute a phase correction

$$\varphi_k(u, v) = \varphi_k(\mathbf{u}) = \langle \mathbf{u}, \hat{\mathbf{t}}_k \rangle = [u \ v] \cdot \begin{bmatrix} \hat{t}_{u,k} \\ \hat{t}_{v,k} \end{bmatrix} \quad \forall (u, v)$$

defined as the inner product between the bidimensional frequency bin \mathbf{u} and the estimated translation vector $\hat{\mathbf{t}}_k$ for the image \mathbf{y}_k obtained with estimation described in the Section 2.2. LR image spectra are then adjusted

$$\check{\mathbf{Y}}_k(u, v) = \mathbf{Y}_k(u, v) e^{-j2\pi\varphi(u, v)} \quad \forall (u, v)$$

and the SR image spectra is computed as the average of the adjusted spectra

$$\mathbf{s}_{spectral} = DFT^{-1} \left(\frac{1}{N} \sum_{k=1}^N \check{\mathbf{Y}}_k \right).$$

2.4 Image enhancing

The image enhancement process is needed only in the case in which interpolation is performed before images alignment. In fact in this case blocking artifacts are too much evident. Further details about this issue are explained in the Section 3.3.

The enhancement process consists in a cascade of two filters. The first one is a circular averaging filter of variable size, whose aim is to smooth the image, that is reducing the blocking artifacts. The second filter is an unsharpening filter defined as a negative of Laplacian, used to restore edges previously smoothed. Parameters of both filters were fixed during test session.

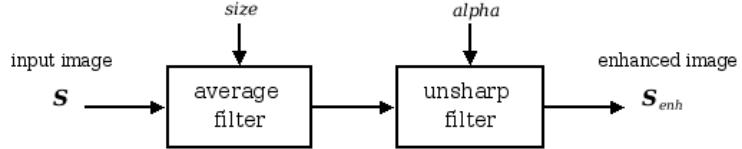


Figure 8: Image enhancing process

3 Experiments

Several tests were performed on two levels in order to check the effectiveness of our proposed SR algorithm:

- horizontal, comparing different SR methods
- vertical, comparing the output of each SR method using different parameters

During a first test session the different proposed method are tested. Moreover, in order to take into account the issue about the execution order of interpolation and registration, during a second test session results obtained with swapped execution order of interpolation and registration with respect to the first session are compared. The objective error measures used to evaluate results are the Mean Squared Error (MSE) defined as

$$MSE = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} [\mathbf{y}(i,j) - \mathbf{y}_{SR}(i,j)]^2$$

and its equivalent Peak Signal-to-Noise Ratio (PSNR)

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \text{ (dB)}$$

where \mathbf{y} and \mathbf{y}_{SR} are respectively the original image and the super-resolved image with size $(N \times M)$ pixels. Execution time of each step of the algorithm is used as a rough performance index¹. Finally, a subjective test was performed in order to check the overall perceived quality of our algorithm.

Parameters used during tests are:

1. input image

- a *small* image: **konqi** (PNG) 284x261 (24 bit truecolor)
- a *big* image: **lena** (PNG) 512x512 (24 bit truecolor)

2. algorithm parameters

D reduction factor

N number of generated LR images

U magnification factor

In the following paragraphs tests are explained.

¹test machine equipped with Pentium® 4 @2.4 GHz (756 MB RAM memory) running Matlab® 7.3.0 on Kubuntu 7.04 (kernel 2.6.20-16)

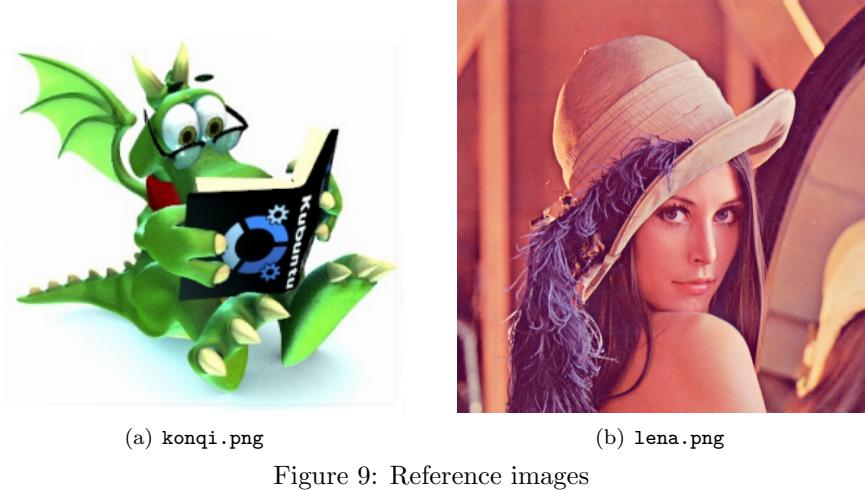


Figure 9: Reference images

3.1 Methods comparison

As a first step, proposed methods are tested using different input images, and results are in Table 1. During this tests interpolation was performed after registration.

	konqi			lena		
	Mean	Median	Spectral	Mean	Median	Spectral
Exec. time (s)	14.9896	20.3554	5.1608	59.5011	67.5618	18.7415
MSE	12.5223	12.5223	25.0009	34.5098	39.8677	59.1032
PSNR (dB)	37.1436	37.1436	34.1512	32.7514	32.1246	30.4147

Table 1: Methods test ($D = 2$, $N = 4$, $U = 2$)

3.2 Parameters variation

In the second step, each method was tested varying algorithm parameters. Results are shown in Tables 2, 3, 4 and in Figures 12, 13, 14.

3.3 Swapped execution order of interpolation and translation estimation

As announced in the Section 2.2, the execution order of the interpolation and registration affects in a significative way the final image. In fact, as a result of the testing process, it turns out that performing interpolation after alignment leads to better result only for *big enough* images (more than about 512×512 pixels).

On the other hand, if alignment follows interpolation then it is possible obtain good results if the image enhancement process is performed on the super-resolved images.

Let us define for clarity reasons two variants of the algorithm:

A interpolation is performed before aligning images



(a) Original



(b) Downsampled



(c) Upsampled



(d) Mean SR



(e) Spectral Mean SR



(f) Median SR

Figure 10: Methods test: super-resolved images for `konqi` ($D = 2$, $N = 4$, $U = 2$)



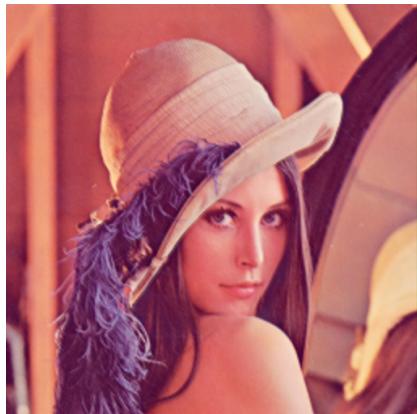
(a) Original



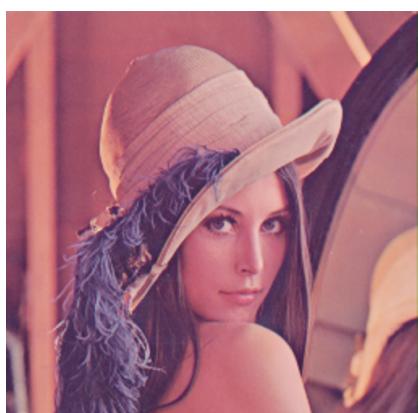
(b) Downsampled



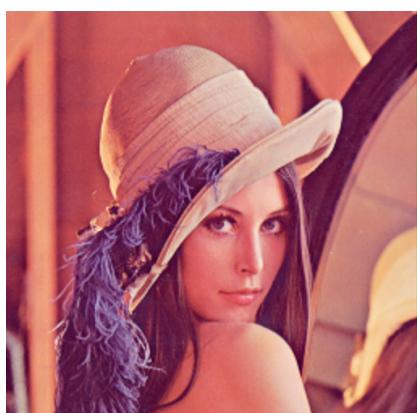
(c) Upsampled



(d) Mean SR



(e) Spectral Mean SR



(f) Median SR

Figure 11: Methods test: super-resolved images for **lena** ($D = 2$, $N = 4$, $U = 2$)

(a) konqi						
(D N U)	algorithm parameters					
	(2 4 2)		(3 4 2)		(4 4 2)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	37.1436	14.9896	33.8064	7.3909	33.5182	4.5296
Median	37.1436	20.3554	34.2054	8.9694	33.9544	5.5179
Spectral	34.1512	5.1608	32.7515	2.3452	32.7808	1.3079

(b) lena						
(D N U)	algorithm parameters					
	(2 4 2)		(3 4 2)		(4 4 2)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	32.7514	59.5011	34.9339	25.5647	35.4577	14.1916
Median	32.1246	67.5618	34.5953	28.8123	35.4424	18.2537
Spectral	30.4147	18.7415	31.0405	7.5336	31.1379	4.5020

Table 2: Algorithm test with varying parameter D

(a) konqi						
(D N U)	algorithm parameters					
	(3 4 2)		(3 6 2)		(3 8 2)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	33.8064	7.3909	33.5142	7.7311	33.7181	8.2032
Median	34.2054	8.9694	33.4336	9.2482	34.1177	9.6046
Spectral	32.7515	2.3452	32.9538	3.6818	33.4037	5.0883

(b) lena						
(D N U)	algorithm parameters					
	(3 4 2)		(3 6 2)		(3 8 2)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	34.9339	25.5647	34.8468	27.2297	35.3372	29.1988
Median	34.5953	28.8123	35.2611	32.5240	35.3507	35.3307
Spectral	31.0405	7.5336	32.1850	12.5987	33.3170	17.0265

Table 3: Algorithm test with varying parameter N

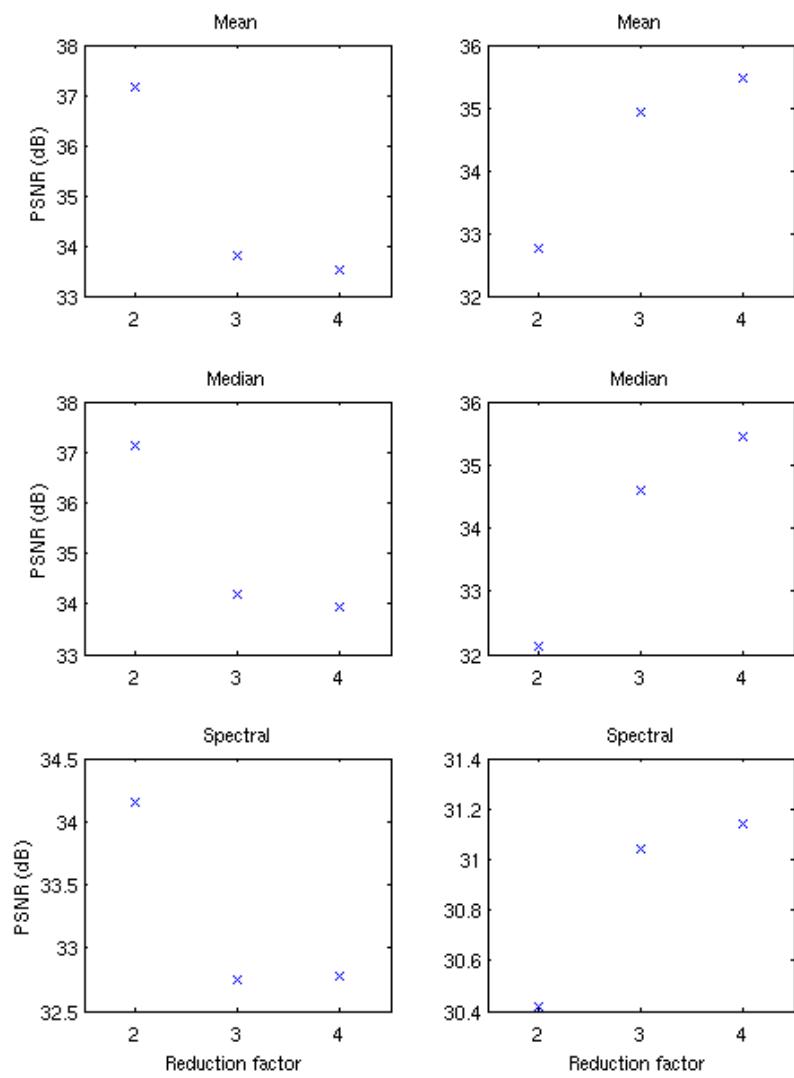


Figure 12: Algorithm test for **konqi** (left) and **lena** (right) with varing parameter D

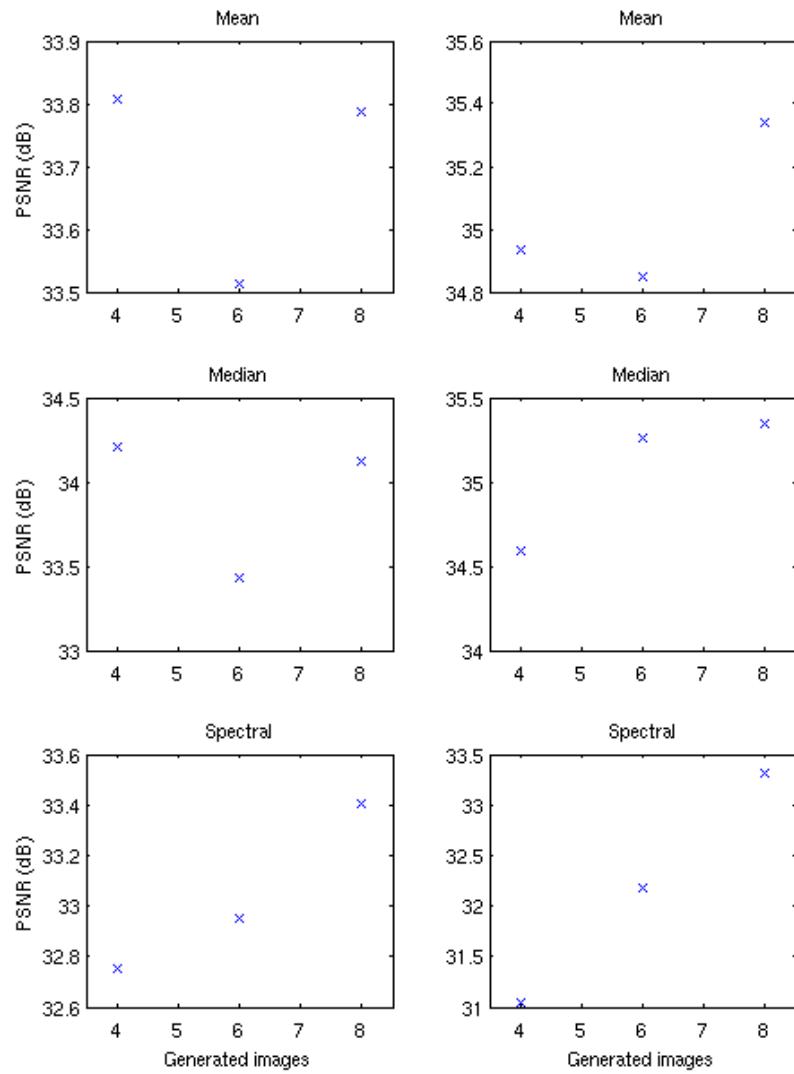


Figure 13: Algorithm test for **konqi** (left) and **lena** (right) with varing parameter N

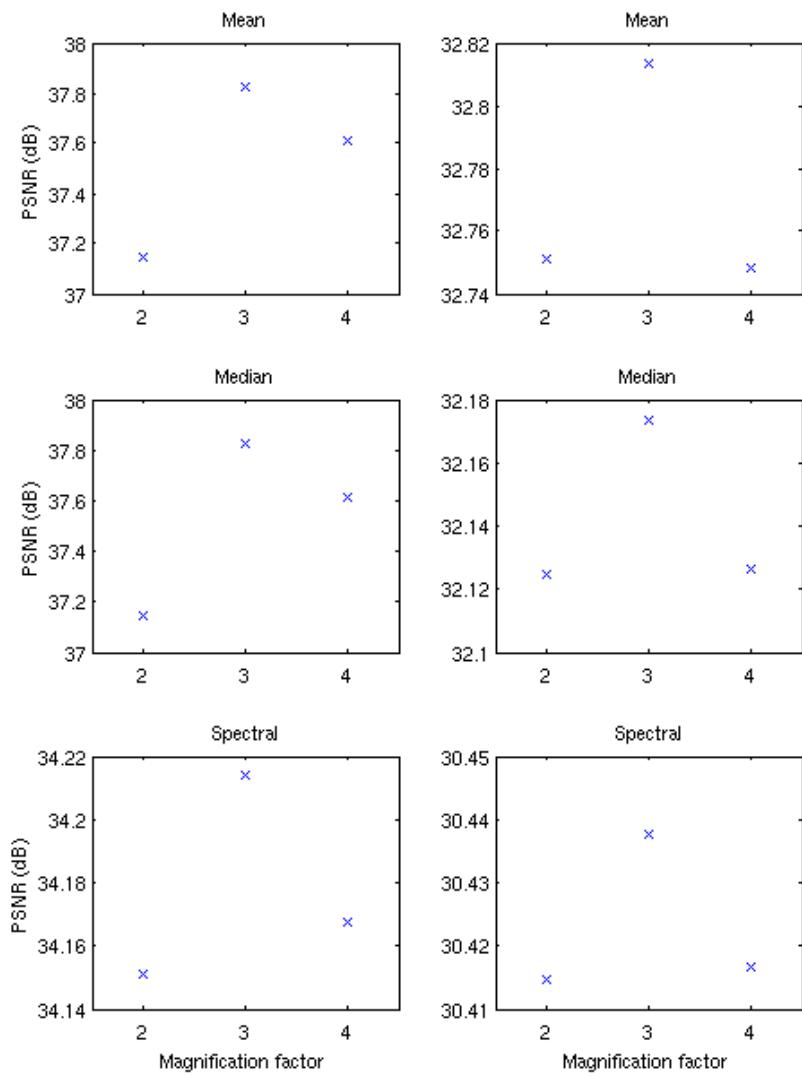


Figure 14: Algorithm test for **konqi** (left) and **lena** (right) with varing parameter U

(a) konqi						
(D N U)	algorithm parameters					
	(2 4 2)		(2 4 3)		(2 4 4)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	37.1436	14.9896	37.8232	33.2917	37.6124	59.1242
Median	37.1436	20.3554	37.8232	41.8511	37.6124	79.3275
Spectral	34.1512	5.1608	34.2143	11.1184	34.1677	20.3879

(b) lena						
(D N U)	algorithm parameters					
	(2 4 2)		(2 4 3)		(2 4 4)	
	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)	PSNR (dB)	Execution time (s)
Mean	32.7514	59.5011	32.8136	151.6995	32.7483	224.1230
Median	32.1246	67.5618	32.1736	156.9219	32.1263	271.1968
Spectral	30.4147	18.7415	30.4378	42.9803	30.4167	75.8435

Table 4: Algorithm test with varying parameter U

B interpolation is performed after alignment stage and the image enhancement is applied to the output images.

In the variant *B* the filtering stage is applied mainly in order to reduce blocking artifacts and enhance perceived quality, especially for *big enough* images. In fact, in Table 5 it is possible notice that for big images the error is reduced, while for small images it become bigger.

	konqi.png		lena.png	
	PSNR _{nofil} (dB)	PSNR _{filt} (dB)	PSNR _{nofil} (dB)	PSNR _{filt} (dB)
Mean	36.7368	35.7707	34.2783	34.3780
Median	36.3569	35.4823	33.7576	33.9012
Spectral	33.7909	33.8328	30.8618	30.9287

Table 5: Error measure on the variant *B* with and without the filtering stage ($D = 2$, $N = 4$, $U = 4$)

In Table 6 differences using variant *A* and *B* are pointed out.

3.4 Subjective tests

Finally, all the images obtained from previous tests were used in some subjective tests. Five subjects were asked to grade the images by their perceived quality. For each set of images, the original one was shown followed by others in random order. Subjects were instructed to assign a score from one to five, representing the range from the worst to the best quality.

3.5 Results

The crucial point of the algorithm is in the registration process. In fact a relative big amount of error is introduced in this stage. Error is due to artifacts present



Figure 15: Variants test. Super-resolved images for `konqi.png` with parameters $D = 2$, $U = 2$, $N = 4$ using variant A and variant B



Figure 16: Variants test. Super-resolved image details for `lena.png` with parameters $D = 2$, $U = 2$, $N = 4$ using variant A and variant B

(a) konqi.png					
	variant A			variant B	
	Mean	Median	Spectral	Mean	Median
Exec. time (s)	14.9396	20.3554	5.1608	14.8597	18.5227
MSE	12.5223	12.5223	25.0009	17.2190	18.4013
PSNR (dB)	37.1436	37.1436	34.1512	35.7707	35.4823
(b) lena.png					
	variant A			variant B	
	Mean	Median	Spectral	Mean	Median
Exec. time (s)	59.5010	67.5618	18.7415	55.5419	70.0805
MSE	34.5098	39.8677	59.1032	23.7291	26.4826
PSNR (dB)	32.7514	32.1246	30.4147	34.3780	33.9012

Table 6: Variants test ($D = 2$, $N = 4$, $U = 2$)

at borders because of the padding and because the precision of the process.

The proposed methods give satisfactory results, but it is worth to point out some remarks. Basically, the Mean and Median methods are equivalents since values of PSNR are quite similar. Spectral Mean method lower values of PSNR and lower subjective grades than the ones of other methods, mainly because of an alteration of the spectral content of the images, perceivable in a luminance change.

Algorithm parameters have, of course, a key role. From graphs it is possible underline dependency of the obtained reconstruction quality from parameters. Generally increasing D or U quality become lower for small images, while for big images it becomes higher, while increasing the nuber of LR imaged gives almost better quality.

As regards performance, it is possible note that higher values of the reduction factor D speed up the algorithm, while higher number of images slow down it. About algorithm, Spectral Mean method is faster than other two method, and variant B is a little slower than variant A .

Moreover, tests point out that the algorithm variants A and B are effective for different classes of images. In particular, variant A works better for *big* images, since the output images are *smooth* enough. Variant B , on the other hand, works better for *small* images. In fact performing interpolation before alignment it is possible restore more details, even if blocking artifacts are present. Finally, the filtering stage reduces the effect of blocking artifacts, with an overall acceptable quality.

An important result is the difference between objective quality measures, based on MSE and PSNR, and perceived quality. This is a well known issue in the audio-video signal processing. In fact, in some cases perceived quality results to be acceptable even for lower values of PSNR.

4 Conclusions and future development

In this project a simple super-resolution algorithm is proposed and some variation of it have been analyzed and tested. The effectiveness of such an algorithm

depends on wide number of parameters, which depend both on the algorithm itself (e.g. reduction or magnification factors, filters coefficients, execution order of stages, and more) and on external factors not (e.g. image size, spectral content, image generation process, ...). Acting on these parameters it is possible improve the final result, both in terms of quality, perceived or objective, and in performance (mainly execution time).

Here are some suggestion for some possible future developments:

- choose a different interpolation technique (e.g. linear or spline)
- use adaptive algorithm based on image features
- use different image enhancing process after the super-resolution stage, for instance adaptive filtering based on spacial or frequency domain or using perceptual models
- improve registration process considering padding techniques on image borders different from replication
- improve or extend super-resolution reconstruction process.

References

- [1] M. Elad, A. Feuer, *Restoration of a Single Superresolution Image from Several Blurred, Noisy, and Undersampled Measured Images*, IEEE Trans. on Image Processing, vol. 6, no. 12, December 1997.
- [2] S. Borman, R. Stevenson, *Spatial Resolution Enhancement of Low-Resolution Image Sequences - A Comprehensive Review with Directions for Future Research*, July 1998, available on line at <http://www.seanborman.com/publications/SRreview.pdf>.
- [3] N. Nguyen, P. Milanfar, *A Computationally Efficient Superresolution Image Reconstruction Algorithm*, IEEE Trans. on Image Processing, vol. 10, no. 4, April 2001.
- [4] S. C. Park, M. K. Park, M. G. Kang, *Super-Resolution Image Reconstruction: A Technical Overview*, IEEE Signal Processing Magazine, May 2003.
- [5] S. Farsiu, M. Elad, P. Milanfar, *Multi-Frame Demosaicing and Super-Resolution of Color Images*, IEEE Trans. on Image Processing, vol. 15, no. 1, January 2006.
- [6] S. Farsiu, D. Robinson, M. Elad, P. Milanfar, *Advances and Challenges in Super-Resolution*, IEEE Trans. on Image Processing, vol. 13, no 10, October 2004.
- [7] S. Farsiu, D. Robinson, M. Elad, P. Milanfar, *Fast and Robust Multiframe Super Resolution*, IEEE Trans. on Image Processing, vol. 13, no. 10, October 2004.
- [8] H. Chang, D. Y. Yeung, Y. Xiong, *Super-Resolution Through Neighbor Embedding*, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, June 2004.
- [9] P. Vandewalle, S. Susstrunk, M. Vetterli, *A Frequency Domain Approach to Registration of Aliased Images with Application to Super-resolution*, EURASIP Journal on Applied Signal Processing, vol. 2006, May 2005.
- [10] M. Tagliasacchi, *Lecture Notes on Digital Audio-Video Signal Processing*, Politecnico di Milano, December 2006, available on line at <http://corsi.metid.polimi.it>