On Super-Resolution *versus* Image Interpolation

Guilherme H. Costa
Centro de Ciências Exatas e Tecnologia
University of Caxias do Sul
Email: holsbach@ieee.org

José C. M. Bermudez
Electrical Engineering Department
Federal University of Santa Catarina
Email: j.bermudez@ieee.org

***Abstract*—This work presents a discussion about multi-frame Super-Resolution Reconstruction (SRR) and single-frame interpolation. SRR has been extensively studied in the last decades but it is not yet as popular as image interpolation. Since both methods are directed to improve image quality, comparisons are inevitable and desirable. However, fundamental misunderstandings can still be found in the literature on the choice between these two approaches. Recently, some mistaken conclusions have been reached regarding the SRR performance. This paper clarifies important SRR issues in image reconstruction, in contrast to interpolation. For reasons further discussed, two algorithms are considered: bicubic interpolation and LMS-SRR.**

I. Introduction

Image interpolation and Super-Resolution Reconstruction (SRR) are two techniques frequently employed to increase the resolution of a digital image. In interpolation, the number of image pixels is increased based on the statistics of the whole image, on a neighborhood of each pixel, or on *a priori* information about the image [1]. This *a priori* information is usually the same for all images processed by the algorithm. Thus, both the low-resolution (LR) and the interpolated high-resolution (HR) images contain the same information, except for the *a priori* knowledge included in the algorithm.

SRR combines multiple different LR images of the same scene or object to form a higher resolution image. It requires two steps: registration and fusion. Registration aligns the images, i.e., estimates the motion of pixels from one LR image to the others. The second step fuses the multiple (aligned) LR images into the HR one. Reference [2] reviews several important results on SRR available in the literature.

The major issues in SRR algorithms are: (i) dependence on an accurate registration [3], [4], [5]; (ii) dependence on outliers [6]; (iii) computational cost. Under inaccurate registration, SRR may lead to image degradation instead of image improvement. This degradation is usually called registration error noise and depends on the characteristics of both the registration algorithm and the image being processed [7]. Outliers are defined as data points whose distributions do not follow the assumed model. In the context of motion, outliers are often regions that have suddenly been occluded or appeared in the image, i.e., the innovations from one scene to the next. Computational cost is important for real-time applications and is usually traded off for performance.

This work was partially supported by Transportes Bertolini LTDA.

Some unclear concepts have been frequently published in this subject. For example, in [8] an interpolation algorithm is proposed as being a single-frame SRR. In [1], several interpolation methods have been tested for video reconstruction and compared among themselves and with the least mean square (LMS-SRR) algorithm proposed in [9], [10]. The LMS-SRR was considered in this comparison since its simplicity to solve the SRR problem makes it an interesting solution for real-time applications such as SRR of video sequences. Properly designed, the LMS-SRR has desirable properties when in presence of registration errors and outliers, both unavoidable in real applications [11]. Objective and subjective evaluations were presented. The main conclusion was that the Bicubic interpolation is the best candidate to be applied in the video reconstruction. When compared with LMS-SRR, Bicubic interpolation presented the same Peak Signal to Noise Ratio (PSNR). However, important SRR issues, such as registration errors and regularization were not considered in [1].

In this work we clarify important aspects of SRR and interpolation. We show that Bicubic interpolation is not competitive with LMS-SRR, and therefore with SRR, unless under severe registration errors. Contrary to what is suggested in [1], we also show that their computational costs are similar.

Notation and the signal models are defined in Section II. In sections III and IV the Bicubic interpolation and LMS-SRR algorithms are briefly presented. In Section V, interpolation and super-resolution are discussed. In Section VI, image reconstruction results obtained using Bicubic interpolation and LMS-SRR algorithms are presented. Finally, Section VII concludes this work.

II. Notation and Signal Models

Hereafter, bold lowercase letters denote column vectors and bold uppercase letters denote matrices. The variable t is integer and indexes discrete-time samples of images and operators. We refer to the observed (low-resolution) images as LR images, and to both the original (desired) and the reconstructed high-resolution images as HR images.

Given the $N \times N$ matrix representation of an LR (observed) digital image $\mathbf{Y}(t)$ and an $M \times M$ ($M > N$) matrix representation of the original HR digital image $\mathbf{X}(t)$, the acquisition process can be modeled as [2]

$$\mathbf{y}(t) = \mathbf{D}\mathbf{H}(t)\mathbf{x}(t) + \mathbf{e}(t)\,, \tag{1}$$

where vectors $\mathbf{y}(t)$ ($N^2 \times 1$) and $\mathbf{x}(t)$ ($M^2 \times 1$) are the lexicographic representations of the degraded and original images, respectively, at discrete time instant t. $\mathbf{D}(t)$ is an $N^2 \times M^2$ decimation matrix and models the subsampling taking place in the sensor. $\mathbf{H}(t)$ is an $M^2 \times M^2$ time-variant matrix that models the blurring. Here, it is assumed known. The $N^2 \times 1$ vector $\mathbf{e}(t)$ models the observation (electronic) noise, whose properties are assumed to be determined from camera tests.

The dynamics of the input signal is modeled by

$$\mathbf{x}(t) = \mathbf{G}(t)\mathbf{x}(t-1) + \mathbf{s}(t)\,, \tag{2}$$

where $\mathbf{G}(t)$ is the warp matrix that describes the relative displacement from $\mathbf{x}(t-1)$ to $\mathbf{x}(t)$. Vector $\mathbf{s}(t)$ models the innovations in $\mathbf{x}(t)$.

III. Bicubic Interpolation

Interpolation is usually performed by up-sampling followed by low-pass filtering. Thus, interpolated pixel values are based on their neighbors'. For computational complexity reasons, interpolation is usually limited to a $4{\times}4$ pixel neighborhood. The generalized Bicubic interpolation function can be expressed as [12]:

$$\tilde{x}_{r',c'}(t) = \sum_{m=-1}^{2}\sum_{n=-1}^{2} \tilde{y}_{r+m,c+n}(t) h(m-r'+r) h(c'-c-n)\,, \tag{3}$$

where $\tilde{y}_{r,c}(t)$ is the pixel in position (r, c) of the image represented by $\tilde{\mathbf{y}}(t) = \mathbf{D}^{\mathrm{T}}\mathbf{y}(t)$. It corresponds to the nearest neighbor to the interpolated pixel $\tilde{x}_{r',c'}(t)$. $h(\cdot)$ denotes the bicubic interpolation function.

IV. The LMS-SRR

The LMS-SRR algorithm attempts to minimize the mean-square error (MSE) $\mathrm{E}\{\|\boldsymbol{\epsilon}(t)\|^2\}$ [9], where $\boldsymbol{\epsilon}(t) = \mathbf{y}(t) - \mathbf{DH}(t)\hat{\mathbf{x}}(t)$, $\hat{\mathbf{x}}(t)$ is the estimate of $\mathbf{x}(t)$ and $\mathrm{E}\{\cdot\}$ denotes statistical expectation. The cost function is $\mathbf{J}_{\mathrm{MS}}(t) = \mathrm{E}\{\|\boldsymbol{\epsilon}(t)\|^2 \,|\, \hat{\mathbf{x}}(t)\}$. The steepest descent update of $\hat{\mathbf{x}}(t)$ is in the negative direction of the gradient

$$\nabla\mathbf{J}_{\mathrm{MS}}(t) = \frac{\partial \mathbf{J}_{\mathrm{MS}}(t)}{\partial \hat{\mathbf{x}}(t)} = -2\mathbf{H}^{\mathrm{T}}(t)\mathbf{D}^{\mathrm{T}}\{\mathrm{E}[\mathbf{y}(t)] - \mathbf{DH}(t)\hat{\mathbf{x}}(t)\} \tag{4}$$

and thus $\hat{\mathbf{x}}_{k+1}(t) = \hat{\mathbf{x}}_k(t) - (\mu/2)\nabla\mathbf{J}_{\mathrm{MS}}(t)$.

The LMS-SRR algorithm is the stochastic version of the steepest descent algorithm. Using the instantaneous estimate of (4) yields

$$\hat{\mathbf{x}}_{k+1}(t) = \hat{\mathbf{x}}_k(t) + \mu\mathbf{H}^{\mathrm{T}}(t)\mathbf{D}^{\mathrm{T}}[\mathbf{y}(t) - \mathbf{DH}(t)\hat{\mathbf{x}}_k(t)]\,, \tag{5}$$

which is the LMS-SRR update equation for a fixed t and for $k = 1, \ldots, K$. The time update of (5) is based on the signal dynamics (2), and performed by $\hat{\mathbf{x}}_0(t+1) = \mathbf{G}(t+1)\hat{\mathbf{x}}_K(t)$.

Using the latter expression in (5), solving for a time recursion in $\hat{\mathbf{x}}_K(t)$, and dropping the subscript K for simplicity, yields the LMS-SRR recursion

$$\hat{\mathbf{x}}(t) = \mathbf{A}^K(t)\mathbf{G}(t)\hat{\mathbf{x}}(t-1) + \mu\sum_{n=0}^{K-1}\mathbf{A}^n(t)\mathbf{H}^{\mathrm{T}}(t)\mathbf{D}^{\mathrm{T}}\mathbf{y}(t)\,, \tag{6}$$

where $\mathbf{A}(t) = [\mathbf{I} - \mu\mathbf{H}^{\mathrm{T}}(t)\mathbf{D}^{\mathrm{T}}\mathbf{DH}(t)]$.

V. Super-Resolution *versus* Interpolation

SRR performs data fusion from more than one LR sources into one output HR image. These sources may be distinct views of an object, acquired with the same image sensor in the presence of relative motion between sensor and object. In the absence of motion, acquisitions made by distinct sensors (with distinct degradation systems [2]) may be used. Differently from image interpolation, an HR reconstructed image obtained via SRR contains more information than available in one single LR image. The HR image is then perceptually much superior to that obtained by interpolation. Thus, the term (single-frame) *super-resolution* [1], [8] seems inadequate to be applied to image interpolation.

The drawbacks of traditional SRR techniques in comparison to image interpolation are: (i) computational cost; (ii) need for motion estimation (registration); (iii) sensitivity to registration errors and outliers. The high computational costs of SRR algorithms usually render them useless for real-time video applications. Fast algorithms such as LMS-SRR are competitive with interpolation in cost, except for the required pre-processing. The required image registration step may be as computationally expensive (or even more) than the algorithm itself. SRR algorithms are very sensitive to registration errors and to outliers. Faster registration algorithms usually lead to larger registration errors and worse reconstruction results. In the presence of outliers, interpolation may be preferable to SRR, which may result in corruption by artifacts in certain applications.

Adaptive algorithms such as LMS-SRR make SRR feasible for real-time applications at the cost of somehow worse reconstruction results. Nevertheless, the LMS-SRR results tend to be much better than those obtained by Bicubic interpolation. The choice between image interpolation and SRR is a commitment to the application. The characteristics required from the results, motion complexity and computational cost must be considered. For the video industry [1], subjective aspects of the image are very important. In this case, a blurred image (with serious frequency spectrum restrictions) may be preferred to an image with artifacts (not natural aspect) but containing more visible details. On the other hand, SRR would be preferred to image interpolation for reading a small text in a natural scene background.

The results presented in [1] seem biased by a bad choice of registration algorithm. The full-search block matching algorithm (with block size 16×16) used in [1] is adequate for a block-translational motion. However, the video sequences assessed also present zooming, affine and other types of motion. Thus, the SRR error images shown were probably mostly due to registration error noise, instead of reconstruction errors from the LMS-SRR algorithm. With a proper design, the LMS-SRR will converge to better reconstruction results [6] than Bicubic interpolation. In fact, it is usual to initialize the algorithm ($\hat{\mathbf{x}}_0(1)$) with the Bicubic interpolation of the first observed LR frame ($\mathbf{y}(1)$). Thus, only a sub-optimal implementation of the LMS-SRR algorithm can explain the performance evaluation results reported in [1]. In the next section we present an evaluation of both Bicubic and LMS-SRR algorithms where the effects of the reconstruction (LMS-SRR) and the registration are isolated.

Regarding computational complexity, LMS-SRR requires $K(2p^2+1)N^2/M^2$ multiplications per output pixel [9], where p is the size of the blur kernel used in $\mathbf{H}(t)$. From (3), Bicubic interpolation requires 16 multiplications per pixel. Assuming a practical implementation of the LMS-SRR with $K = 1$ [6] and a 4×4 blur kernel, this algorithm requires $33N^2/M^2$ multiplications. Thus, for an output image with $M^2 = 1920 \times 1080$ pixels (Full HDTV format) and a decimation factor of 2, 8.25 multiplications per output pixel are required. This is about half the Bicubic interpolation computational complexity. Even considering that a detailed comparison should include the contributions of all floating point operations and of the registration algorithm required by LMS-SRR, this simple comparison shows that the LMS-SRR algorithm is competitive with Bicubic interpolation in terms of computational burden. The complexity of the pre-processing (registration step) required by SRR is $\mathcal{O}[(2s+1)^2M^2]$, where s is the assumed maximum displacement of the pixels in any direction [13], [7]. Considering $s = 4$, it results in approximately 81 operations per pixel. However, some applications allow global image displacement approximation [11] and, in these cases, it is not necessary to consider the whole image in the registration step. When this approximation is allowed, using an $M/4 \times M/4$ sub-image in the registration step it can decreases to approximately 5 operations per pixel.

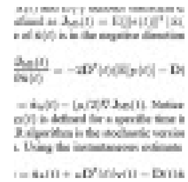
VI. Experimental Results

This section presents four simulation examples designed to compare the Bicubic interpolation and the LMS-SRR reconstruction results. The level of registration errors is progressively increased to clearly show its role in the overall reconstruction performance. The first example is for a known motion. Thus, no registration errors are present in SRR. For the following three examples, a real registration algorithm [7] is used. Thus, registration errors become present. The level of registration errors increases for each new simulation.

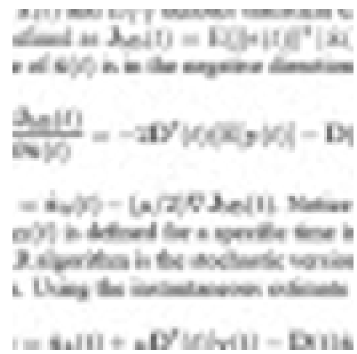
In all simulations, the additive noise vector $\mathbf{e}(t)$ was modeled as a zero mean and gaussian process WGN$(0, 10)$. Neumann boundary conditions were considered in the implementation of the warp matrix for the LMS-SRR algorithm. The LMS-SRR parameters were designed according to [6]. In the first two examples $\mathbf{DH}(t)$ modeled blurring through a 2×2 mean filter performed over an impulsive subsampling. In the last two examples the filter mask becomes 4×4. Error images (absolute difference between original HR image and reconstructed HR image) are shown multiplied by a factor of two for better visualization.

A. Example 1: the known motion case

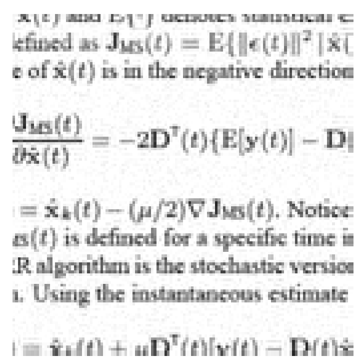
The only way to implement a known motion video is through synthetic sequences. A pure global and translational motion sequence with 50 frames was created by cropping a larger still image. The motion was created by random independent and identically-distributed displacements of sizes zero or one at each time instant t and in each direction (vertical and horizontal) in the HR space. This movement simulates a camera shaking. 240×240 HR and 120×120 LR images were used.



(a)



(b)



(c)

Fig. 1. Reconstruction results: (a) 50^{th} observed LR frame; (b) Bicubic interpolation of (a); (c) LMS-SRR result.

Figure 1 shows the 50^{th}: (a) observed LR frame; (b) interpolated HR frame; (c) LMS-SRR result. The 50^{th} frame error images are presented in Figure 2. The LMS-SRR results are clearly superior to the interpolation results. The evolution of the spatial mean square reconstruction error is presented in Figure 3, where $\mathbf{v}(t) = \mathbf{x}(t) - \hat{\mathbf{x}}(t)$. The PSNR achieved for this frame was 25.47dB for LMS-SRR and 18.58dB for Bicubic interpolation.

B. Example 2: low levels of registration errors

This simulation used the same sequence of Example 1, but with a registration algorithm [7] performing the motion estimation. This algorithm has been proposed for translational global motion, as synthetically implemented in the sequence. Thus, it probably leads to low registration error levels in this

(a)

(b)

Fig. 2. Error images for the 50^{th} frame: (a) Bicubic interpolation (b) LMS-SRR.

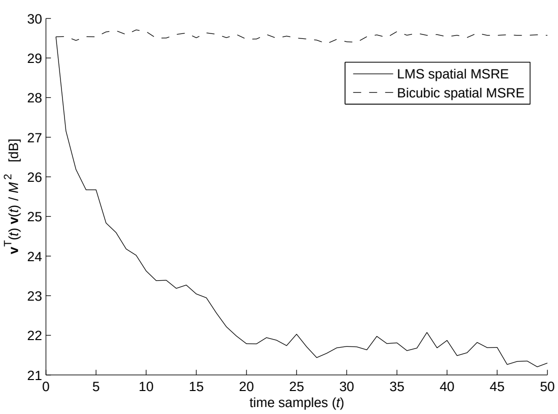


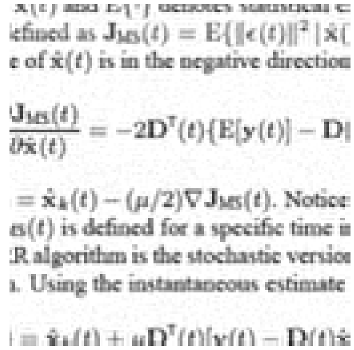
Fig. 3. Spacial mean square reconstruction error evolution.

case. Although it was possible, hereafter the registration was willfully not performed by blocks, but in the hole image.

Figure 4(a) shows the reconstruction result for the 50^{th} frame using the LMS-SRR algorithm. Figure 4(b) shows the respective error image. The results achieved by the LMS-SRR are once again clearly superior to the interpolation ones (figs 1(b) and 2(a)). The PSNR achieved by the LMS-SRR was 24.64dB (against 18.58dB for Bicubic interpolation).

C. Example 3: moderate levels of registration errors

A real sequence was used for this simulation. This sequence presents an approximately horizontal displacement of a toy over a still background. The actual motion can be approx-



(a)

(b)

Fig. 4. LMS-SRR results for the 50^{th} frame and low level of registration errors: (a) reconstruction result (b) error image.

imated as being global in this case, since the background is smooth. However, the used registration algorithm [7] will probably lead to higher registration errors than in Example 2, where the motion was truly global and translational.

Here a decimation factor of 4 (240×240 HR and 60×60 LR images) has been used to amplify some characteristics of the algorithms and to yield a better visualization. Figure 5 shows the 50^{th}: (a) observed LR frame; (b) interpolated HR frame; (c) LMS-SRR result. The result achieved by the LMS-SRR algorithm is still superior to that obtained by interpolation. The PSNR achieved for this frame was 33.13dB for LMS-SRR and 29.79dB for Bicubic interpolation.

D. Example 4: high levels of registration errors

The first 50 frames from *Mobile* sequence were used in this example. This sequence presents a non-global translational motion. Hence, the applied registration algorithm [7] is not capable of correctly estimate the motion, leading to high levels of registration errors.

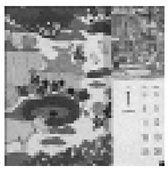
As in Example 3, a decimation factor of 4 was used: 240×240 (top-left pixels of each frame) HR and 60×60 LR images. Figure 6 shows the 50^{th}: (a) observed LR frame; (b) interpolated HR frame; (c) LMS-SRR result. The result achieved by the LMS-SRR algorithm is apparently worse than that obtained by interpolation. The PSNR achieved for this frame was 17.05dB for LMS-SRR and 16.70dB for Bicubic interpolation. Note that the aspect of the LMS-SRR result could be improved using regularization (smoothing the

(a)

(b)

(c)

Fig. 5. Reconstruction results under moderate levels of registration errors: (a) 50^{th} observed LR frame; (b) Bicubic interpolation of (a); (c) LMS-SRR result. Figures (b) and (c) are scaled by a factor of 0.5 for space purposes.



(a)

(b)

(c)

Fig. 6. Reconstruction results under high levels of registration errors: (a) 50^{th} observed LR frame; (b) Bicubic interpolation of (a); (c) LMS-SRR result. Figures (b) and (c) are scaled by a factor of 0.5 for space purposes.

solution). This would lead to an image quality more similar to the Bicubic interpolation result.

VII. Conclusions

This work discussed the comparative evaluation of interpolation and super-resolution reconstruction (SRR). These are the two most employed techniques to increase the resolution of digital images. More specifically, the performances of the Bicubic interpolation and the LMS-SRR algorithms were compared. The main conclusions of this comparison are: (i) interpolation is not competitive with SRR in general; (ii) the LMS-SRR algorithm is computationally competitive with Bicubic interpolation; (iii) under small or average registration errors, reconstruction results obtained using LMS-SRR are far superior to those obtained using Bicubic interpolation; (iv) under severe registration errors and considering objective metrics, the results obtained using either LMS-SRR or Bicubic interpolation tend to be similar.

References

[1] H. Özdemir and B. Sankur, "Subjective Evaluation of Single-Frame Superresolution Algorithms," in *Proc. EUSIPCO 2009*, Aug. 2009, pp. 1102–1106.

[2] S. C. Park, M. K. P. and M. G. Kang, "Super-Resolution Image Reconstruction: A Technical Overview," *IEEE Signal Processing Magazine*, vol. 20, pp. 21–36, May 2003.

[3] D. Capel and A. Zisserman, "Computer vision applied to super resolution," *IEEE Signal Processing Magazine*, vol. 20, pp. 75–86, May 2003.

[4] Z. Wang and F. Qi, "Super-Resolution Video Restoration With Model Uncertainties," in *Proc. IEEE ICIP 2002*, July 2002, pp. 853–856.

[5] E. S. Lee and M. G. Kang, "Regularized Adaptive High-Resolution Image Reconstruction Considering Inaccurate Subpixel Registration," *IEEE Trans. Image Processing*, vol. 12, pp. 826–837, July 2003.

[6] G. H. Costa and J. C. M. Bermudez, "Informed Choice of the LMS Parameters in Super-Resolution Video Reconstruction Applications," *IEEE Trans. Signal Processing*, vol. 56, pp. 555–564, Feb. 2008.

[7] R. Mester and M. Hotter, "Robust Displacement Vector Estimation Including a Statistical Error Analysis," in *Proc. IEE ICIPA 1995*, July 1995, pp. 168–172.

[8] A. Lukin, A. S. Krylov and A. Nasonov, "Image Interpolation by Super-Resolution," in *Proc. Graphicon 2006*, July 2006, pp.239–242.

[9] M. Elad and A. Feuer, "Superresolution Restoration of an Image Sequence: Adaptive Filtering Approach," *IEEE Trans. Image Processing*, vol. 8, pp. 387–395, Mar. 1999.

[10] M. Elad and A. Feuer, "Superresolution Reconstruction of an Image Sequence," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, pp. 817–834, Sept. 1999.

[11] G. H. Costa and J. C. M. Bermudez, "Registration Errors: Are They Always Bad for Super-Resolution?," *IEEE Trans. Signal Processing*, vol. 57, pp. 3815–3826, Oct. 2009.

[12] W. K. Pratt, *Digital Image Processing*. New York:Wiley, 2001.

[13] M. Trimeche, J. Lainema and M. Gabbouj, "Hierarchical Motion Estimation Using Recursive LMS Filters," in *Proc. PCS 2007*, Nov. 2007.