Computation Time Optimization in Super-Resolution Applications

Eduardo Quevedo[1,2], David Horat[1], Gustavo M. Callicó[2], and Félix Tobajas[2]

[1] Oceanic Platform of the Canary Islands, Carretera de Taliarte s/n, 35200, Telde, Spain
[2] Research Institute for Applied Microelectronics (IUMA), University of Las Palmas de Gran Canaria (ULPGC), Campus Universitario de Tafira, Las Palmas de Gran Canaria, Spain
{eduardo.quevedo,david.horat}@plocan.eu,
{gustavo,tobajas}@iuma.ulpgc.es

Abstract. The necessity to improve image resolution is of great concern in multiple diverse fields such as: medicine, communications, or satellite and underwater applications. A high variety of techniques for image enhancement has been proposed in the literature, being a trade-off the relation between the computation time and the quality of the obtained results. This work is focused on a test environment that permits to objectively compare the quality enhancement of images processed by two different improvement methods: bilinear interpolation and Super-Resolution (SR), presenting how these results relate to the computation time. The objective comparison is based on the PSNR (Peak Signal-to-Noise Ratio) and the SSIM (Structural SIMilarity).

Keywords: Image enhancement, Super-Resolution, computation time.

1 Introduction

In some scenarios, such as satellite or underwater imagery, it is difficult to use high resolution sensors due to physical constraints. A way to address this problem is to accept the image degradations and use signal processing techniques to post-process the captured images; one of these techniques is referred to as Super-Resolution (SR) reconstruction [1]. The used approach for SR in this paper is to construct High-Resolution (HR) images from several observed Low-Resolution (LR) images. In this approach every LR frame is considered to be an undersampled, aliased observation of the true scene. In the imaging process, the camera captures several LR frames, which are downsampled from the HR scene with subpixel shifts between each other. SR reconstruction reverses this process by aligning the LR observations to subpixel accuracy and combining them into a HR image grid, thereby overcoming the imaging limitation of the camera. SR arises in many areas such as surveillance video, remote sensing, medical imaging, or underwater applications [2]. The work presented in this paper is focused on the comparison of the results obtained from the application of a bilinear interpolation and an alternative method developed for image improvement to evaluate the quality and the consumed computation time in order to achieve this purpose.

R. Moreno-Díaz et al. (Eds.): EUROCAST 2013, Part II, LNCS 8112, pp. 101–108, 2013.

In general terms there is not a commonly accepted method for quality image evaluation in the current literature, and in most of the cases, the assessment is just visual. Some works [3], use the PSNR (*Peak Signal to Noise Ratio*) in order to compare the bilinear interpolated image with the super-resolved one, and the SSIM (*Structural SI-Milarity*) index is introduced in [4]. After evaluating several metrics, it was detected that PSNR and SSIM are extensively used. Consequently it was decided to use both of them as a basis for the image enhancement evaluation.

2 Super-Resolution Algorithm

The SR algorithm that has been used and tested in this work is based on [5] and belongs to the *fusion* category. Specifically, it is a non-iterative dynamic SR algorithm, which greatly reduces the computational cost and memory requirements. One of its particularities is that it provides both *static* and *dynamic* SR, depending only if one high resolution output frame, or a sequence of frames, is obtained as a result.

The algorithm execution can be divided into three different and somehow independent stages, as shown in Fig. 1. The first stage is known as *Motion Estimation* and it determines the motion between two or more views of the same scene with sub-pixel accuracy, which depends on the selected scale factor (i.e. obtaining an output sequence which size is twice the size of the input sequence would mean a scale factor of 2). For each frame (*current frame*) a *working window* is generated, that contains the adjacent frames (*reference frames*), from which the new information will be obtained. Afterwards, the current frame is divided into uniform square regions, called Macro-Blocks (MB), and the motion estimation is performed between the current frame and each reference frame, following a MB basis. The algorithm uses a block-matching method to obtain the motion vectors associated to each MB. Two search algorithms have been implemented on this stage: the Full Search Algorithm (FSA) and the New Three Step search (NTS), being both of them extremely reliable, but having the NTS less computational load.

Once all the motion vectors have been found, the second stage, called *Shift & Add*, is executed. A grid of Very High Resolution (VHR) size (its size depends on the precision of the motion estimation stage) is filled with the information pointed by the estimated motion vectors of each frame of the *working window*, superimposing the information from several different frames. However, it is possible that the reference frames do not contain information enough to fill all the locations on the grid for the current frame, so there could be some empty positions on it, denoted *holes* in the context on this work. During the next stage, called *Fill Holes*, each empty pixel is filled by using a bilinear surface interpolator. Lastly, the VHR image is decimated in order to obtain the super-resolved image with HR size. The whole process is repeated for each one of the frames contained in the input sequence. Hence, there will be an output sequence as a result.

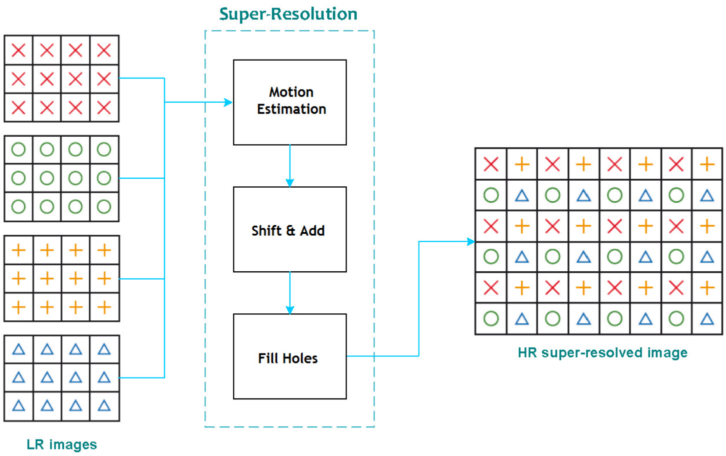


Fig. 1. General approach for SR

The main input parameters for the SR algorithm are the following:

- *Scale (SC)*: This is the scale factor that will be applied to the input sequence to generate the super-resolved image/sequence.
- *Search Area (SA):* Number of pixels around every MB that will be considered when searching the motion-vectors. For scenes containing high amounts of motion, high values of SA are required. For low-motion scenes, high values of SA are not self-defeating, to the extent that they will not decrease the output quality, but a lot amount of inefficient computation would be performed. Values from 8 to 16 are usually enough for the majority of sequences.
- *Macro-Block Size (MBS):* Size of the LR MBs that the images will be divided onto. Low values of MBS allow having more independent motion vectors, but it largely increases the computation time. High values of MBS increases the coherence of the motion vectors, but it degrades the quality when several objects with different motions are considered. MB sizes from 4 to 32 are usually enough for the majority of sequences.
- *Window Back Frames (BF):* Number of frames that will be used before the current reference frame to be combined with the pixels of the current reference frame. Large values of BF will increase the probability of having SR improvements but, at the same time, it also increases the probability of having artifacts if the frames are weakly correlated.
- *Window Forward Frames (FF):* Idem to BF, but applied to frames after the current reference frame.

In this paper, the used frames for the SR process are selected not only depending on the defined BF and FF values, but also on a dynamic threshold, which works depending on the similarity between the LR frames, using for this purpose the metric SSIM, which is also considered together with the PSNR to measure the image improvement when the SR process is applied.

3 Proposed Testbench

In order to set up a validation environment for the proposed image quality enhancement method, a testbench has been developed. It is based on the fact that a LR input sequence is necessary to perform the SR process. Therefore, if the process to be tested is the SR method, firstly a LR sequence is obtained from a HR input sequence. Then, the LR sequence is super-resolved, and, at the same time, a bilinear interpolation is applied for comparing their results with the super-resolved sequence. However, the testbench, which is shown in Fig. 2, is flexible, so its configuration can be modified to adapt to any other image enhancement method. While interpolation uses only the information available in the current frame, SR combines the information from several frames in the reference window, increasing the probability to enhance the resolution of the current frame. The proposed selective filter dynamically decides what frames should be used according to an objective metric decision maker.

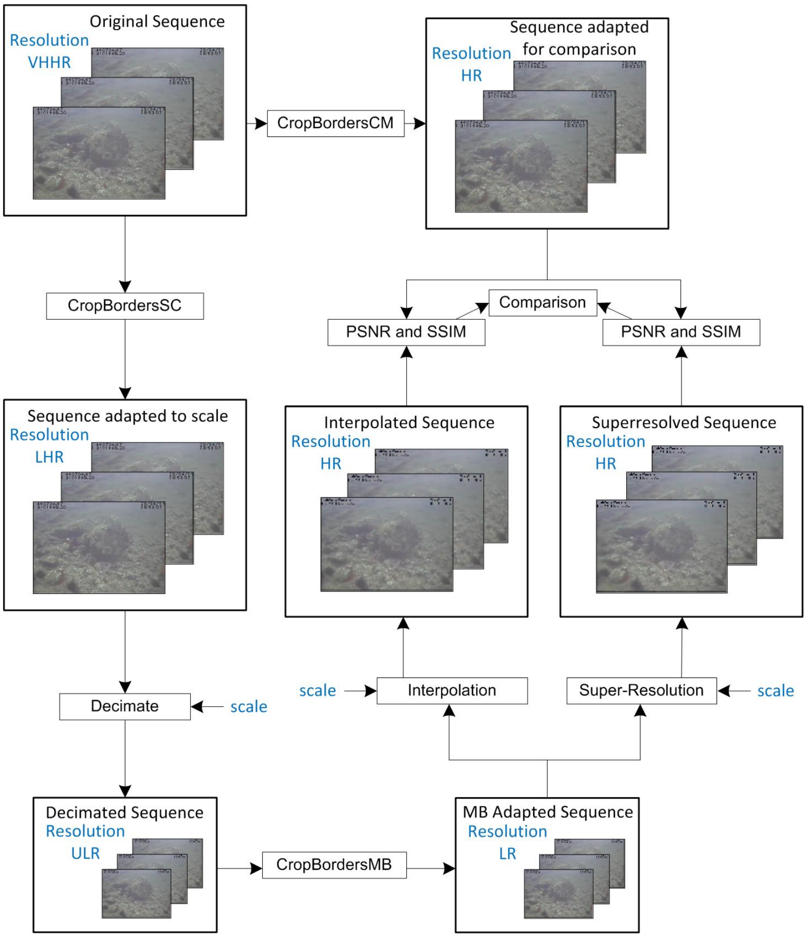


Fig. 2. Proposed testbench

4 Results

In this section, the original sequences and the superresolved sequences are showed to compare them, both objectively and subjectively against the interpolated image. The computation time is also analyzed in order to check how it is reduced when the frame selection is done depending on a dynamic threshold (selective filter). The compared sequences present different characteristics in order to show the algorithm behavior in different scenarios. In this paper two sequences with quite different characteristics are presented in order to show the correct behavior of the algorithm:

- The first sequence, *galdar*, presents null Local Motion (LM) and medium Global Motion (GM). This situation is quite adequate for SR, so a high value of MBS (16) and a low value of SA (2) will be used, as shown in Table 1.
- The second sequence, *foreman*, shows high Local Motion (LM) and Global Motion (GM). This situation is more difficult for SR, so an intermediate value of MBS (8) and a higher value of SA (4) will be used, as shown in Table 2.

Table 1. Characteristics and SR parameters of sequence "*galdar*"

T	*GM*	*LM*	*WxH*	*SC*	*MBS*	*SA*	*BF&FF*
NA	Medium	Null	352x288	2	16	2	+/- 20

Table 2. Characteristics and SR parameters of sequence "*foreman*"

T	*GM*	*LM*	*WxH*	*SC*	*MBS*	*SA*	*BF&FF*
NA	High	High	352x288	2	8	4	+/- 20

Table 3. Optimum thresholds depending on the considered sequence

sequence	*galdar*	*mobile*	*foreman*	*flower*	*mobcal*	*torock*
threshold	30%	20%	40%	30%	40%	60%

The selection of the dynamic threshold used to reduce the computation time and improve quality is decisive in order to optimize the results. Table 3 shows the optimum experimental thresholds for different sequences. From here it can be deduced that values around 30% work well with the majority of the sequences. Fig. 3 shows an example (for *mobile* sequence) of how behaves the average quality of a sequence (PSNR in this case) depending on the selected threshold (difference between using or not using selective filter), where 0% is identified with full SR (all frames of the sequence are used in the SR process) and 100% with interpolation (no frames are used in the SR process). The presented minimum in Fig. 3 (using a 20% threshold) maximizes the quality when a selective filter is used (the appropriate frames are selected and used in the SR process). Fig. 4 and Fig. 5 show how the execution time is reduced as the threshold increases its value. In this case the selected thresholds: 30% for *galdar* sequence and 40% for *foreman* sequence reduce the computation time in a 12% and a 36% respectively.

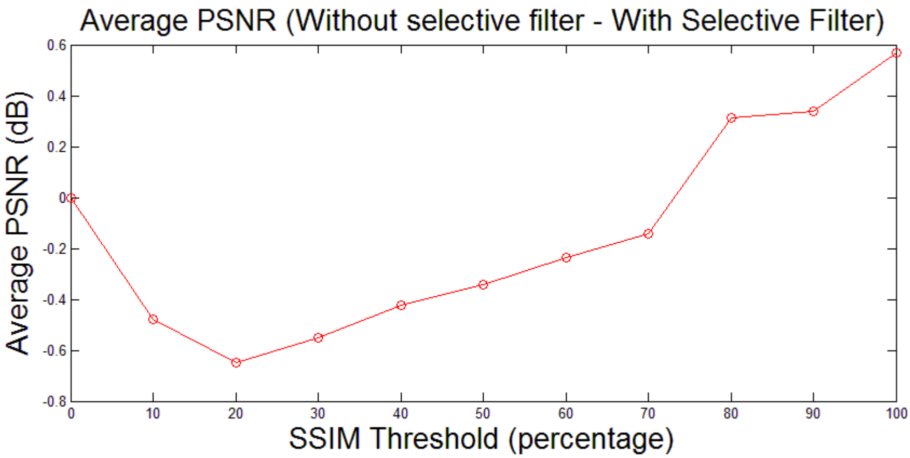
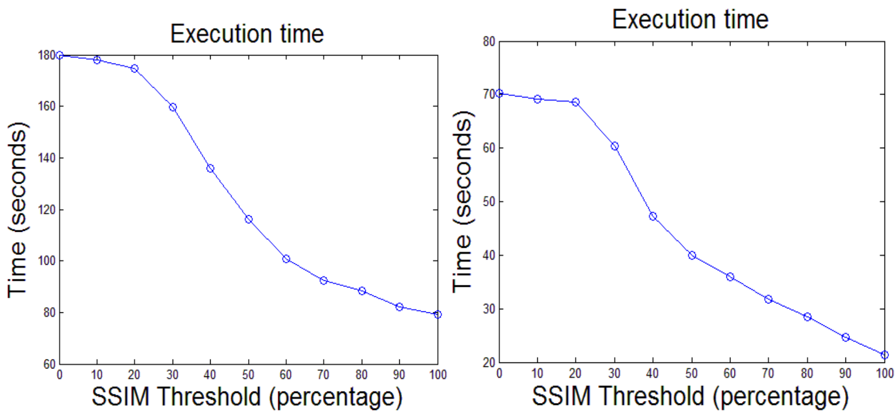


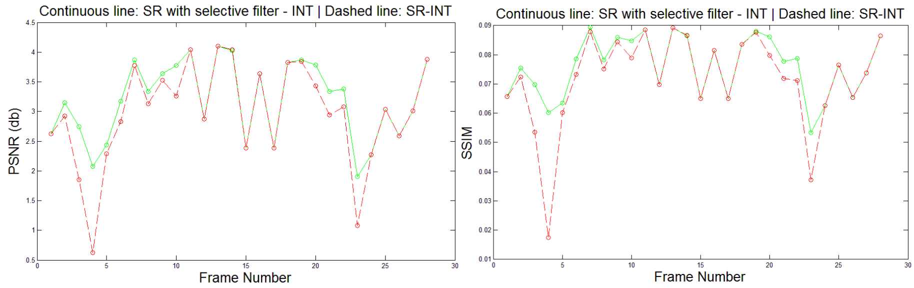
Fig. 3. Threshold behavior for *mobile* sequence – The minimum expresses the highest quality using a selective filter



a) b)

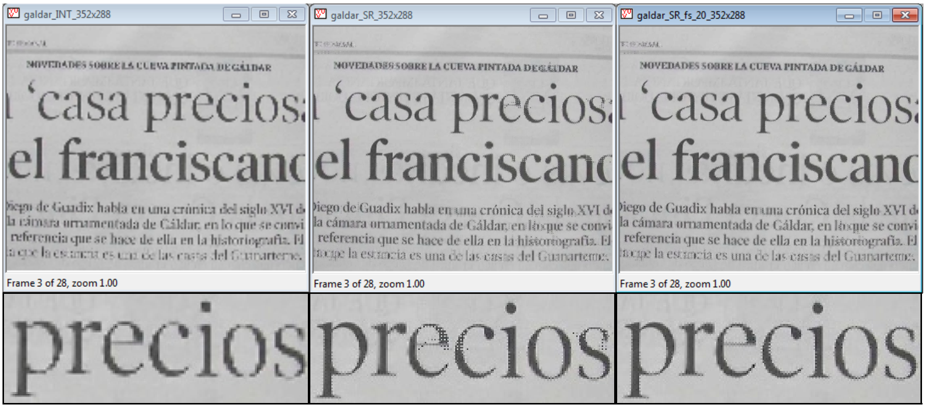
Fig. 4. Execution time – a) *galdar* sequence, b) *foreman* sequence

Next figures show how not only the computation time is reduced but also the quality is improved using a selective filter (INT means *interpolation*). Fig. 5 shows objective quality metric results for *galdar* sequence. An average improvement of 0.3 dB in PSNR and 0.01 in SSIM are achieved. The image results (Fig. 6) show how the artifacts have been reduced.



a) b)

Fig. 5. Objective quality metrics results (*galdar* sequence) – a) PSNR, b) SSIM



a) b) c)

Fig. 6. 3rd *galdar* frame – a) Interpolation, b) Original SR, c) Filtered SR

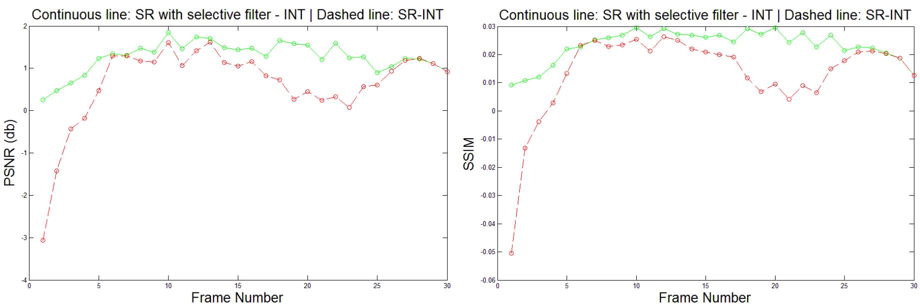


Fig. 7. Objective quality metrics results (*foreman* sequence)

Fig. 7 shows objective quality metric results for *foreman* sequence. An average improvement of 0.6 dB in PSNR and 0.01 in SSIM are achieved. The artifacts reduction and borders smoothness in Fig. 8 justifies the quality metrics improvement.

Fig. 8. 1^{st} *foreman* frame – a) Interpolation, b) Original SR, c) Filtered SR

5 Conclusions

In this paper, it has been demonstrated how the computation time can be reduced in SR applications, even with an increase of the quality. The considered frames in the process are determined with a previous selection based on a dynamic threshold which uses the SSIM objective quality metric.

The proposed threshold has been applied to a wide set of sequences and the results not only show computation time reduction but also quality improvement, which has been justified using objective quality metrics such as the PSNR (*Peak Signal to Noise Ratio*) and the SSIM (*Structural SIMilarity*) index. Currently, adaptive determination of the thresholds to the images features is under research.

References

1. Milanfar, P.: Super-Resolution Imaging. CRC Press (2011)
2. Quevedo, E., Rodríguez, O., Callicó, G.M., Tobajas, F., de Armas, V.: Image Resolution Enhancement in Underwater Applications. In: DCIS (2012)
3. Han, Y., Chae, T.B., Lee, S.: Parameter-estimation based single-image super-resolution. In: Proceedings of the IEEE Global Conference on Consumer Electronics (GCCE), pp. 560–563 (2012)
4. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: From Error Visibility to Structural Similarity. IEEE (2004)
5. Callicó, G.M., Peset Llopis, R., Núñez, A., Sethuraman, R., de Beeck, M.O.: A Low-Cost Implementation of Super-Resolution based on a Video Encoder. In: IECON (2002)