Video Super-Resolution Using Motion Compensation and Classification-Aided Fusion

Karen Simonyan[*], Sergey Grishin[†],
Moscow State University, Graphics & Media Lab

Dmitriy Vatolin[‡], and Dmitriy Popov[**]
YUVsoft Corp.

Abstract

In this paper, we propose a super-resolution algorithm based on image fusion via pixel classification. Two high-resolution images are constructed, the first by means of motion compensation and the second by means of image interpolation. The AdaBoost classifier is then used in the fusion of these images, resulting in an high-resolution frame. Experimental results show that the proposed method outperforms well-known video resolution enhancement methods while maintaining moderate computational complexity.

CR Categories: I.4.5 [Image Processing And Computer Vision]: Reconstruction; G.3 [Probability And Statistics]: Correlation and Regression Analysis

Keywords: super-resolution, dynamic super-resolution, video resolution enhancement, pixel classification, AdaBoost

1 Introduction

In the last years high-definition television (HDTV) is becoming more and more popular. Vendors have developed the manufacturing of displays complying recent HDTV standards. Various types of medium have been adapted to HDTV. However, there is still a lack of an high-definition video content. As most video content is still lower resolution, special algorithms are needed to convert it to higher resolution. Video resolution enhancement algorithms can be divided into two groups according to the type of information used.

The first group is composed of algorithms that use the information only from the current low-resolution (LR) video frame. All image interpolation methods can be considered belonging to this group, as video resolution enhancement can be performed by applying an image interpolation to each video frame. Methods from this group (such as bilinear and bicubic interpolation) are widely used due to their low computational complexity. The second group consists of the so-called super-resolution (SR) algorithms. In this case the information from neighboring frames is also used. Since video streams are highly redundant, such a processing leads to higher enlargement quality. At the same time, the analysis of neighboring frames together with the current frame significantly increases computational complexity of the algorithm.

In this paper, we propose a fast SR algorithm based on high-resolution (HR) image fusion via pixel classification. Two HR images are constructed by means of motion compensation and image interpolation. Next, the AdaBoost [Friedman et al. 1998] classifier is used in the fusion of these images, resulting in an HR frame. The low computational complexity of the proposed method makes it suitable for fast video processing. Performance superiority over standard video resolution enhancement methods is shown.

The rest of the paper is organized as follows. In Section 2 a review of the related work is given. In Section 3 the proposed SR algorithm is described. Experimental results are presented in Section 4. In Section 5 we conclude our results and discuss possible directions of future research.

2 Related Work

Most contemporary SR algorithms are aimed at producing one HR frame from a set of LR frames. According to [Farsiu et al. 2006], we call them static SR algorithms. Various approaches have been taken recently in this field. Farsiu et al. [2006] considered an image acquisition process as a sequence of operators, transforming a real world scene into a LR outcome. They treated SR as an energy minimization problem and used robust regularization based on bilateral total variation to construct an HR frame. It was assumed, that the camera point spread function (PSF) was known, and LR image sequence was obtained from spatially close points. Jiang et al. [2003] improved the classical iterative backward projection algorithm [Irani and Peleg 2001] by making it more robust to optical flow estimation errors and by modeling the PSF via an elliptical weighted area filter. Freeman et al. [2002] proposed an example-based image magnification method. They argued that the relationship between medium- and high-frequency image patches can be exploited to add high-frequency details to the enlarged image; a special large-size database was introduced to store such correspondences between patches. A similar technique combined with the reconstruction constraint was applied by Wang et al. [2005] for SR. The method, however, exhibits enormous computational complexity. Li [2006] proposed an SR algorithm for the special case where a set of LR frames acquired at different focal lengths is available. Static SR methods, if directly applied to each video sequence frame, can not ensure the temporal consistency of the enlarged video, which impedes their direct application to video resolution enhancement.

The problem of producing HR video from LR video (dynamic SR) has also been addressed. Bishop et al. [2003] extended the approach of Freeman et al. [2002] to video resolution enhancement by introducing special priors maintaining the temporal consistency of the enlarged video. As any example-based method, it is quite dependent on the training set. Moreover, patch search operations in the database are computationally expensive. Kong et al. [2006] proposed a method of video SR intended for cases where some HR photographs are available in addition to LR video,

[*] e-mail: simonyan@graphics.cs.msu.ru
[†] e-mail: sgrishin@graphics.cs.msu.ru
[‡] e-mail: dmitriy@yuvsoft.com
[**] e-mail: dpopov@yuvsoft.com

which forms a strong restriction on the application field of the method. Cheung et al. [2005] applied epitomic analysis (analysis of the probability distributions in 3D patches sampled from the video) to video resolution enhancement. This technique, however, is suitable only for processing video that contains scenes acquired at different focal lengths. Farsiu et al. [2006] used an approximation of the Kalman filter to bind the frame under enlargement with the previously processed frame. A translational motion model was assumed.

Therefore, unacceptable computational complexity, utilization of simple motion models, and the necessity of a priori knowledge about camera characteristics make modern SR algorithms unsuitable for real-time or near-real-time resolution enhancement of video streams.

3 Proposed Super-Resolution Algorithm

3.1 Algorithm Outline

We consider the case of two-times magnification of spatial resolution; the other cases can be processed similarly. It is also assumed that a pixel of LR frame is the mean of four corresponding pixels of this frame in high resolution.

Our approach can be considered as an extension of that described in Farsiu et al. [2006]. An unknown HR frame H^n (the upper index represents the frame number) is modeled by an output frame $\widetilde{H}^n$, which is constructed in three steps:

- **Motion compensation**. The current LR frame L^n is compensated from the already constructed previous HR frame $\widetilde{H}^{n-1}$ to form an HR image M (Section 3.2).
- **Spatial interpolation.** L^n is spatially interpolated using the Lanczos interpolation filter with radius 4 (Lanczos4) to form an HR image U.
- **Fusion**. The resultant frame $\widetilde{H}^n$ is constructed via pixel-wise fusion of two HR images, M and U. Pixel fusion coefficients are calculated via the AdaBoost classifier (Section 3.3).

The flowchart of the proposed algorithm is presented in Figure 1.

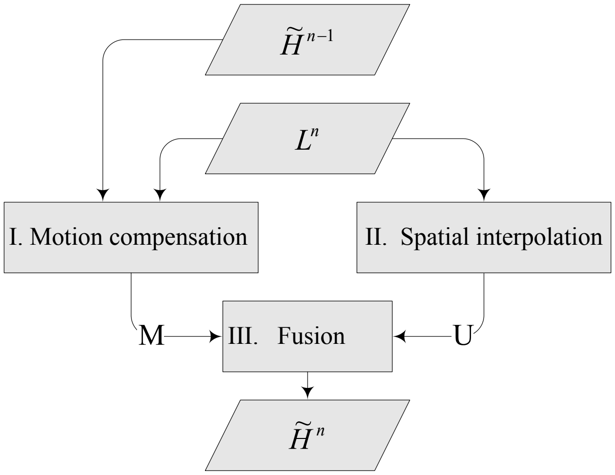


Figure 1: Flowchart of the proposed algorithm.

3.2 Motion Compensation

Our motion estimation (ME) algorithm is based on the block matching algorithm described in Ahmad et al. [2006]. We made two major improvements to the original algorithm. The first is the adaptive block size. Such a technique allows capturing complex motion in a video sequence. In uniform areas and in the areas with low motion 16×16 blocks are used while in other cases (areas of fine texture and areas with intense motion) 8×8 block partition is applied. A block is considered uniform, if the variance of its luminance does not exceed a certain threshold. The motion complexity is determined by thresholding the variance of neighboring blocks' motion vectors. Near the edges of moving objects the values of the variance are big, thus leading to smaller block size and more precise motion vector field.

The second improvement affects the robustness of ME in uniform areas. The problem becomes especially crucial when using 8×8 block partition, as the robustness of the block matching ME decreases when using such a block size. Thus, some smoothness constraints are to be applied to motion vector field in uniform areas. In our case the frequencies of candidate motion vectors appearance in a candidate list are taken into account. Moreover, the range of pattern search refinement is adaptively reduced in such areas.

Motion vectors are found with quarter-pixel accuracy. ME is performed in luminance plane only.

The architecture of SR ME differs from that of conventional ME, as the resolution of the current frame is two times smaller in each direction than the resolution of the reference frame. The same is true for the current block and its corresponding reference block. But under the assumption of dependency between LR pixel (pixel of LR frame) and corresponding HR pixels (pixels of HR frame) the adaptation of motion compensation to SR is straightforward. The following metric function ρ is used to compare an LR block B with an HR reference block pointed by a motion vector $\vec{d}$:

$$\rho\left(B,\vec{d}\right)=\sum_{(x,y)\in B}\left|L^n(x,y)-\widetilde{L}\left(x,y,\vec{d}\right)\right|, \tag{1}$$

where

$$\widetilde{L}\left(x,y,\vec{d}\right)= \frac{1}{4}\sum_{i,j=0}^{1}\widetilde{H}^{n-1}\left(2\cdot x+\vec{d}_x+i,\ 2\cdot y+\vec{d}_y+j\right) \tag{2}$$

is the pixel of a reference block converted to low resolution using bilinear downsampling. Thus, the metric is an adaptation of the Sum of Absolute Differences (SAD) metric to the SR framework.

After a motion vector is estimated for each block, the HR motion-compensated frame M is built from the HR reference blocks.

3.3 Images Fusion via Classification

After M and U HR images are built, a per-pixel fusion process is performed to construct an output HR frame. Pixel-wise fusion allows the masking of motion compensation artifacts (such as blockiness), since the pixels of U image will be used in the areas of bad compensation. We introduce a probabilistic framework for the fusion process, modeling an unknown HR pixel value H_j^n (the lower index represents the pixel number) by a random variate $\hat{H}_j^n$ for which two possible cases are considered:

$$\hat{H}_j^n \in \left\{\hat{M}_j = M_j + \Delta_i^M{}_j, \hat{U}_j = U_j + \Delta_i^U\right\},\quad j \in P(i), i = 1,\dots,\left|L^n\right|. \tag{3}$$

Here $P(i) = \{j_k(i)\}_{k=1}^4$ is the preimage of an LR pixel i (i.e., the set of numbers of four HR pixels corresponding to i, as illustrated in Figure 2). $\hat{M}_j$ and $\hat{U}_j$ are the pixels of M and U, corrected according to the upscaling errors Δ^M and Δ^U, which are defined as

$$\Delta_i^M = L_i^n - \underset{j \in P(i)}{mean} M_j,\ \Delta_i^U = L_i^n - \underset{j \in P(i)}{mean} U_j. \tag{4}$$

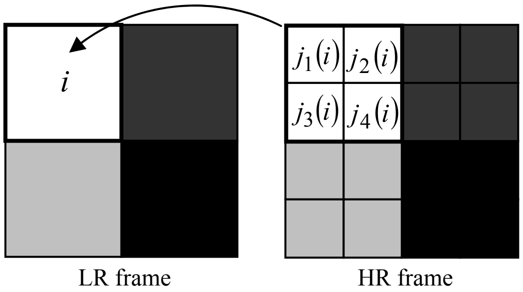


Figure 2: Correspondence between LR and HR pixels.

It is assumed that, for each LR pixel i, all four pixels from $P(i)$ are simultaneously taken either from $\hat{M}$ or from $\hat{U}$.

Thus, a classifier can be employed to estimate the class c_i for each LR pixel i, where $c_i = 1$ if pixels from $P(i)$ are taken from $\hat{M}$, and $c_i = -1$ otherwise. We apply the AdaBoost boosting algorithm which iteratively constructs a strong classifier as a linear combination of weak classifiers. The main advantage of AdaBoost is that at each iteration the training is focused on hard training samples. Various weak learners can be used; in our case we employed Classification And Regression Tree (CART) of depth 5.

The training set was constructed as follows: 30000 LR pixels were selected from various video sequences. For each LR pixel t from the training set the class was determined in the least squares sense, i.e., the following sums of squared errors were calculated:

$$SSE^{\hat{M}} = \sum_{j \in P(t)} \left(H_j^n - \hat{M}_j\right)^2,\ SSE^{\hat{U}} = \sum_{j \in P(t)} \left(H_j^n - \hat{U}_j\right)^2. \tag{5}$$

The class that corresponds to the lower sum was chosen, thus forming a ground truth training sample.

The feature vector X consists of five features. The first two are the upscaling errors (4). The third is the luminance variance of neighboring LR pixels. It is useful to determine uniform areas where the Lanczos interpolation usually produces better results. The fourth is the sum of variances of the motion vector coordinates, calculated for neighboring blocks. It helps to determine the areas of intense motion. The fifth feature is derived as follows: the difference between the sums of squared errors (5) is approximated by substituting L_i^n instead of H_j^n for each j from $P(i)$:

$$\begin{aligned} SSE^{\hat{M}} - SSE^{\hat{U}} &= \sum_{j \in P(i)} \left(\hat{M}_j - \hat{U}_j\right)\left(\hat{M}_j + \hat{U}_j - 2H_j^n\right) \approx \\ &\approx \sum_{j \in P(i)} \left(\hat{M}_j - \hat{U}_j\right)\left(\hat{M}_j + \hat{U}_j - 2L_i^n\right) = X_5. \end{aligned} \tag{6}$$

Four iterations of AdaBoost were performed to train a strong classifier with sufficient performance. A small number of weak learners in the committee ensured the low computational complexity of the classifier.

From our experiments, better results can be achieved, if H_j^n is modeled by the expectation of $\hat{H}_j^n$, i.e., by

$$\widetilde{H}_j^n = E\hat{H}_j^n = p_i \hat{M}_j + (1 - p_i)\hat{U}_j,\ j \in P(i), i = 1,\dots,\left|L^n\right|. \tag{7}$$

Here, $p_i = P(c_i = 1 \mid X)$ is the class conditional probability, which, according to Friedman et al. [1998], can be written as

$$p_i = \frac{e^{2F(X)}}{1 + e^{2F(X)}}, \tag{8}$$

where $F(X)$ is the classifier output. Such an approach can be somewhat argued by the fact that in cases where $F(X)$ is close to 0, it is difficult to determine the correct class. And in such cases, (7) leads to the averaging of $\hat{M}_j$ and $\hat{U}_j$, which is slightly better than using one of them. Equation (8) can be presented in a more general form:

$$p_i = \frac{e^{\alpha F(X)}}{1 + e^{\alpha F(X)}}, \tag{9}$$

where $\alpha > 0$ is the "aggressivity" parameter. Plots of p_i for $\alpha = \{2, 19\}$ are presented in Figure 3.

As can be seen, varying α we tune the shape of dependence of p_i on $F(X)$. Our experiments show that $\alpha = 19$ provides significantly higher quality compared to $\alpha = 2$. Therefore, all experimental results presented later were obtained using $\alpha = 19$.

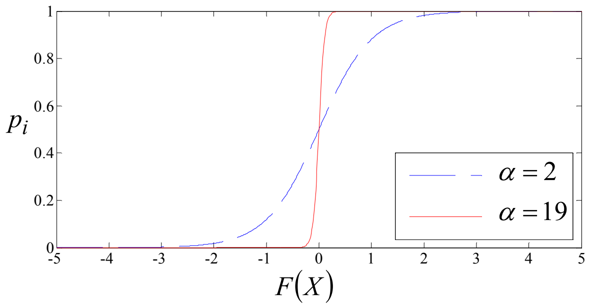


Figure 3: Dependence of p_i on $F(X)$.

For the resolution enhancement of chrominance planes the fusion process (7) is applied with the same p_i values as are used for the luminance plane enlargement.

4 Experimental Results

Our test set consists of four HR video sequences in 1280×720 resolution based on the material recorded by SVT Sveriges Television AB and available for free at *ftp://ftp.ldv.e-technik.tu-muenchen.de/dist/test_sequences/720p/* FTP site. Notably, we used Mobcal, Parkrun (604 frames version), Stockholm, and Shields (604 frames version) videos. Since the frames {44, 89, 135, 227, 273, 320, 367} of Mobcal video expose artifacts, they were removed. Grey frames were removed from all these videos.

LR video was derived from the test set applying bilinear downsampling by a factor of $1/2$. For comparison, video resolution enhancement was performed by the proposed method, Lanczos4 and bicubic interpolation. Next, the Y-PSNR metric was calculated for each of the compared methods using ground truth HR video as a reference. The results of the Y-PSNR comparison are presented in Table I.

TABLE I
Comparison of Y-PSNR (dB) for various methods

Method	Video Sequence			
	Mobcal	Parkrun	Shields	Stockholm
Bicubic	29.05	23.18	32.13	30.73
Lanczos4	29.84	23.82	33.14	31.57
Proposed	33.35	25.17	35.42	32.66

ΔY-PSNR metric was also calculated by subtracting algorithms' Y-PSNR values from those of bicubic interpolation, used as a reference. Plots of the ΔY-PSNR metric are presented in Figure 4. Our method outperforms Lanczos4 and bicubic interpolation while maintaining an acceptable processing speed of 2 frames per second, achieved on a single-core Athlon64 3600+ computer using a non-optimized C implementation. Visual performance of the proposed SR algorithm is demonstrated in Figure 5 – Figure 7 together with the results of the Lanczos4 and bicubic interpolation filters. The images were taken from Mobcal video sequence. For clarity, the presented images are enlarged two times by a nearest neighbor filter.

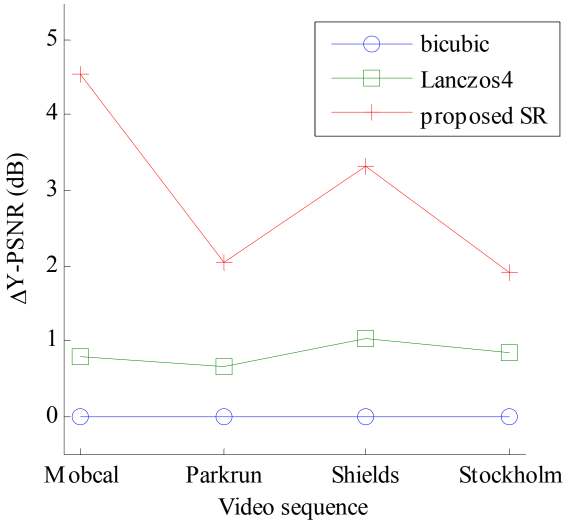


Figure 4: Comparison of ΔY-PSNR (relative to bicubic) for various methods.

As can be seen, the proposed SR algorithm provides a sharper and more detailed picture in comparison with other resolution enhancement methods.

The proposed SR algorithm can be easily adapted for the processing of video containing scene changes by employing any scene detection technique, so that the first frame after a scene change would be reconstructed using spatial interpolation only. However, even without such a detection the result of the fusion process after a scene change is slightly acceptable. We constructed a five-frame video sequence, which consists of a blank frame followed by the first frames of four videos from our test set, thus leading to four scene changes. The PSNR loss in comparison with Lanczos4 was 0.68 dB. While processing a real video, all the frames between scene changes will not be affected.

5 Conclusion

In this paper, we presented a new super-resolution method intended for fast video resolution enhancement. Our method provided better quality than did frequently used single-frame enlargement methods, as proven by both objective and subjective comparisons.

The performance and the processing speed of our method can be further improved by using cascade classifiers and by employing edge-directed interpolation. Moreover, motion compensation can be applied to just highly textured areas, thus increasing the speed even further. The research in these directions is ongoing.

The work was partially supported by the Russian Foundation for Basic Research, grant 07-01-00759-a.

References

Ahmad, I., Zheng, W.G., Luo, J.C., Liou, M. 2006. A Fast Adaptive Motion Estimation Algorithm. *IEEE Transactions on Image Processing*, vol. 16, issue 3, pp. 420–438.

Bishop C., Blake A., and Marthi B. 2003. Super-Resolution Enhancement of Video. In C. M. Bishop and B. Frey (Eds.), *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*.

Cheung V., Frey B. J., and Jojic N. 2005. Video Epitomes. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 42-49.

Farsiu S., Elad M., and Milanfar P. 2006. A Practical Approach to Super-Resolution, Invited paper. In *Proceedings of the SPIE Conference on Visual Communications and Image Processing,* vol. 6077.

Freeman W. T., Jones T. R., and Pasztor E. C.. 2002. Example-Based Super-Resolution. In *IEEE Computer Graphics and Applications*, vol. 22(2), pp. 56–65.

Friedman J. H., Hastie T., and Tibshirani R. 1998. *Additive Logistic Regression: a Statistical View of Boosting*. Dept. of Statistics, Stanford University Technical Report.

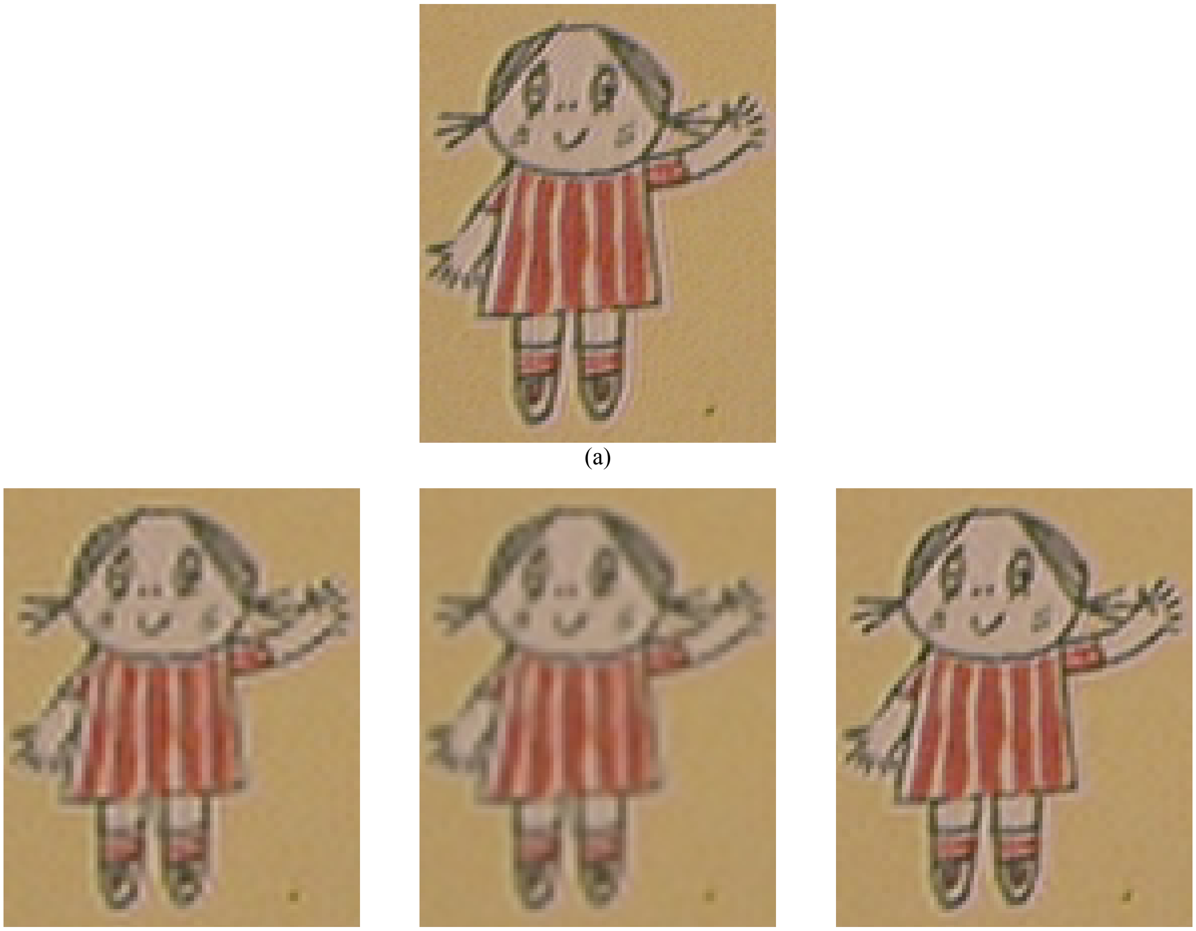
Irani M., and Peleg S. 1991. Improving Resolution by Image Registration, *Journal of Computer Vision, Graphics, and Image Processing*, vol. 53(3), pp. 231–239.

Jiang Z., Wong T.T., and Bao H. 2003. Practical Super-Resolution from Dynamic Video Sequences. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 549-554.

Kong D., Han M., Xu W., Tao H., and Gong Y.H. 2006. Video Super-Resolution with Scene-Specific Priors. In *Proc. British Machine Vision Conference*, pp. 549-558.

Li X. 2006. Super-Resolution for Synthetic Zooming. *EURASIP Journal on Applied Signal Processing*, No. Article ID 58195, pp. 1-12.

Wang Q., Tang X., and Shum H. 2005. Patch Based Blind Image Super Resolution. In *Proceedings of IEEE Conference on Computer Vision*, vol. 1, pp. 709-716.



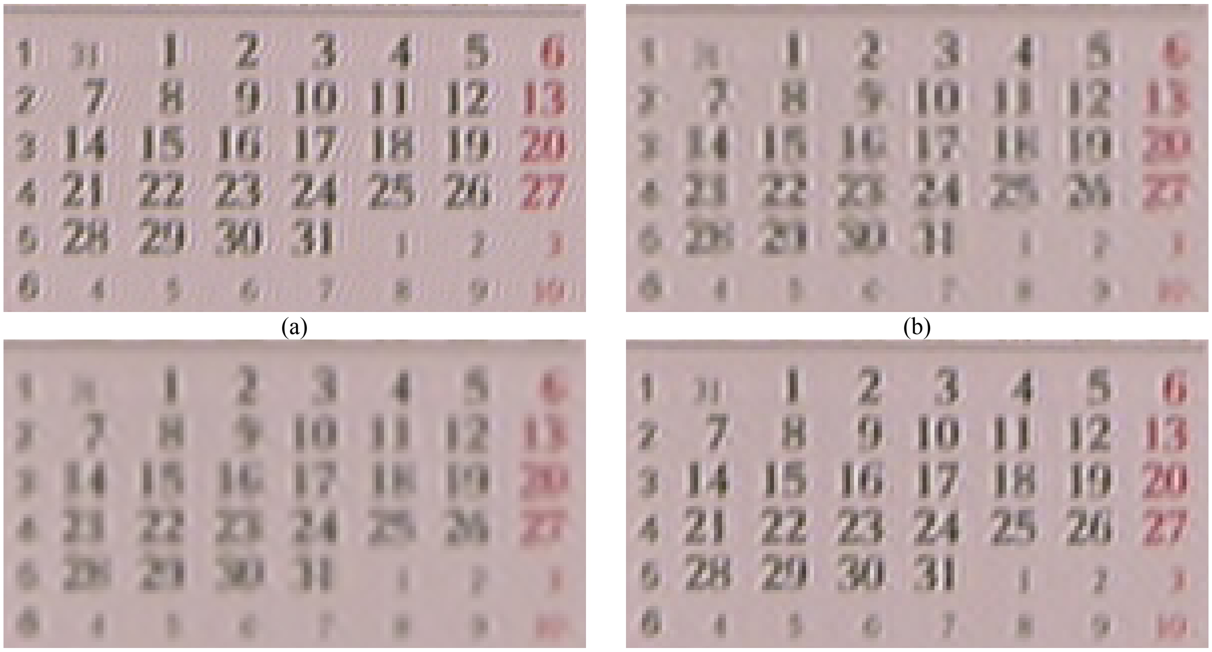
(a)

(b)

(c)

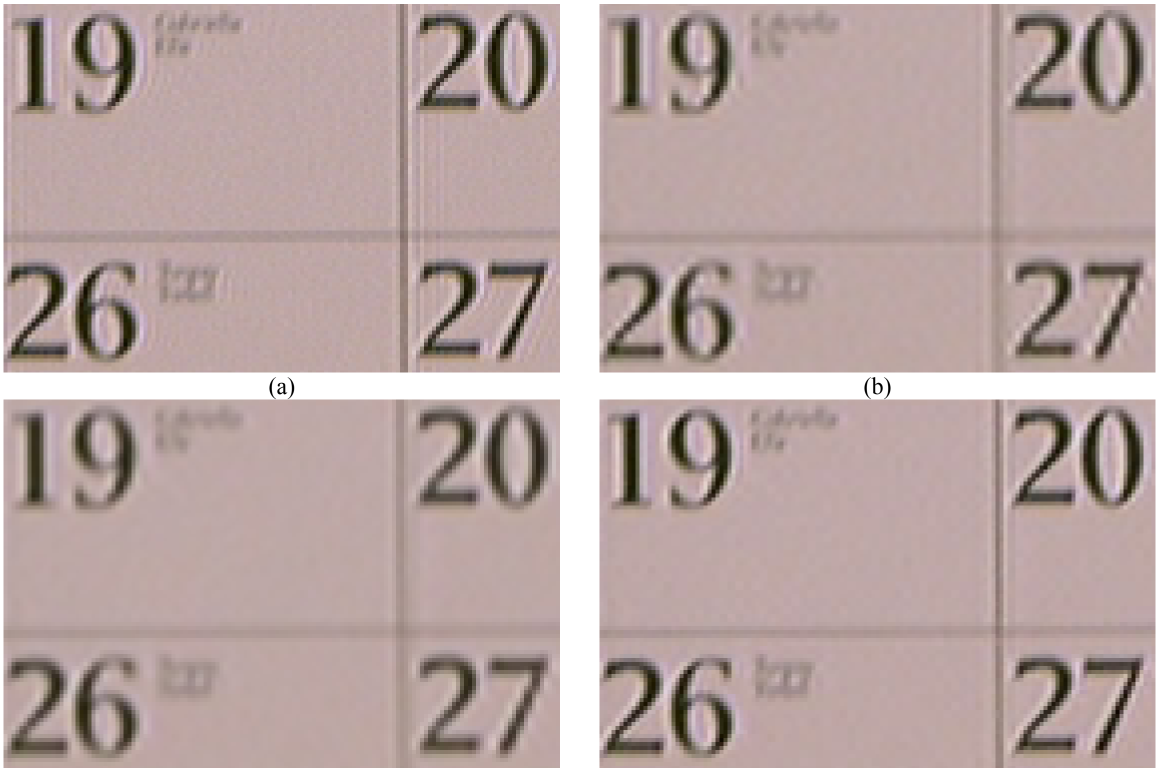
(d)

Figure 5: Visual quality comparison of Mobcal frame. (a) Ground truth; (b) Lanczos4 filter; (c) bicubic filter; (d) proposed SR.



(a) (b) (c) (d)

Figure 6: Visual quality comparison of Mobcal frame. (a) Ground truth; (b) lanczos4 filter; (c) bicubic filter; (d) proposed SR.



(a) (b) (c) (d)

Figure 7: Visual quality comparison of Mobcal frame. (a) Ground truth; (b) lanczos4 filter; (c) bicubic filter; (d) proposed SR.