Super Resolution For Subpixel-Based Downsampled Images

Ketan Tang†, Oscar C. Au†, Lu Fang‡, Yuanfang Guo†, Pengfei Wan†, Lingfeng Xu†

†Department of Electronic and Computer Engineering,
The Hong Kong University of Science and Technology

‡University of Science and Technology of China

Email: †{tkt, eeau, eeandyguo, leoman, lingfengxu}@ust.hk, ‡fanglu@ustc.edu.cn

***Abstract*—Subpixel-based downsampling is a new downsampling technique which utilizes the fact that each pixel in LCD is composed of three individually addressable subpixels. Subpixel-based downsampling can provide higher apparent resolution than pixel-based downsampling. In this paper we study the inverse problem of subpixel-based downsampling. We found that conventional pixel-based super resolution algorithms are not suitable for subpixel-based downsampled images due to the special downsampling pattern. In this paper we propose a super resolution algorithm specially for subpixel-based downsampled images, which use piecewise autoregressive model to model spatial correlation of neighboring pixels, and incorporate the special data degradation term corresponding to the subpixel downsampling pattern. We formulate the super resolution problem as a constrained least square problem and solve it using Gauss-Seidel iteration. Experiment results demonstrate the effectiveness of the proposed algorithm.**

I. Introduction

Downsampling is a common task in image processing. The simplest way to downsample an image by N times is to select one pixel out of every N pixels both in horizontal and vertical direction, named Direct Pixel-based Downsampling (DPD) (Fig. 1). Many sophisticated adaptive image super resolution (SR) methods have been proposed for DPD downsampled images. Soft-decision Adaptive Interpolation (SAI) method proposed by Zhang and Wu [10] is one of the most successful ones, in which they model pixel correlation using a piecewise 2-D autoregressive (PAR) model, and recover the HR image block by block using least square minimization.

An improved downsampling scheme is subpixel-based downsampling (subpixel-downsampling for short), which utilizes the fact that each pixel on a color LCD is actually composed of individual addressable red, green, and blue subpixel stripes (Fig. 1). Subpixel-downsampling has gained increasing attention these days. Existing subpixel-downsampling algorithms include Direct Subpixel Downsampling (DSD) [7] in which subpixel downsampling is applied only in horizontal direction, and Direct Diagonal Subpixel Downsampling (DDSD) [3], in which subpixel downsampling is applied diagonally in a 3×3 block (Fig. 1). The abbreviations DPD, DSD and DDSD are coined in [3]. More sophisticated algorithms include MMSE-SD [3] and DDSD-FA [4], in which anti-aliasing filters are applied before subpixel downsampling. Compared to conventional pixel based downsampling (pixel-downsampling for short), subpixel-downsampling is more capable of preserving fine details of the images, with a cost of creating color-fringing artifacts in downsampled images [3], [4].

With the development of subpixel downsampling, an interesting question rises: given a subpixel downsampled image, can we recover the original HR image? Potential application of this technology can be low-bitrate image/video compression, image demosaicing, etc. This technology is not a trivial extension of conventional pixel-based super resolution. Most of the existing SR methods cannot apply on subpixel downsampled images, due to the special downsampling pattern. Existing pixel-based SR algorithms are developed under the assumption that the input small image is obtained by pixel-downsampling, and they usually only consider super-resolving luminance component (Y channel) if the input image is a full color image while the chrominance component (U, V channels) are interpolated by simple interpolators such as bicubic. Thus pixel-based SR algorithms are not suitable for subpixel-downsampled images, because the Y component of the LR image is no longer a simple down-sized version of HR Y component, also significant amount of information is contained in U, V channels and those algorithms cannot effectively utilize the information.

To overcome the above mentioned problems, we propose a SR algorithm for subpixel-downsampled images utilizing the subpixel downsampling characteristic. Unlike conventional pixel-based SR methods, we super-resolve R, G, B channels simultaneously. For each channel the pixel correlation is modeled as a piecewise autoregressive (AR) process as in [10]. The optimization problem is solved using Gauss-Seidel iteration as in [8]. Since our method is based on autoregressive model, we name this method Subpixel based Autoregressive Super Resolution (SASR).

The rest of the paper is organized as follows. We briefly review the DDSD algorithm in Section II , and 2-D autoregressive model in Section III. We propose our subpixel-based SR algorithm in Section IV. The experiment results are shown in Section V. We conclude our work in Section VI.

II. Subpixel based image downsampling

This paper focuses on DDSD for its simplicity and capability of maintaining spatial details. Basically DDSD takes RGB components differently from the three pixels on the diagonal

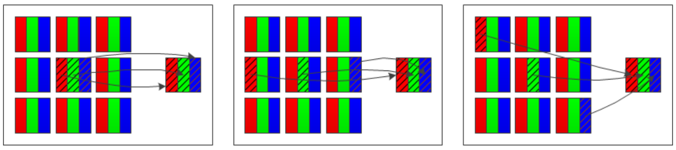


Fig. 1: Three main downsampling patterns. Left: DPD; Middle: DSD; Right: DDSD.

line of each 3×3 block (Fig. 1). Thus DDSD can increase the apparent resolution both in horizontal and vertical directions. By changing the data degradation term, our algorithm can easily adapt to any subpixel-downsampling algorithm.

Figure 2 shows the downsampled images of "tree". We can see that compared to DPD, DDSD can preserve more details, especially on sharp edges, e.g. the long grasses and the tree branches.

III. Image interpolation using autoregressive model

Similar as in [10], we model an image as a piecewise autoregressive (PAR) process

$$X_i = \sum_t \alpha_t X_{i \diamond t} + v_i \qquad (1)$$

where v_i is a random perturbation independent of spatial location i and the image signal X, $X_{i\diamond t}$ $(t = 1, 2, \ldots, 8)$ are the first-order neighbors of X_i. The PAR coefficient α_t is assumed the same for all pixels in a local window. Let I_h be the HR image to be estimated by interpolating the observed LR image I_l, which is a downsampled version of the HR image by a factor of two. Let $x_i \in I_l$ and $y_i \in I_h$ be the pixels of images I_l and I_h, $y_{i\diamond t}$ $(t = 1, 2, \ldots, 8)$ be the neighbors of pixel location i in the HR image. $y^{(8)}_{i\diamond t}$ $(t = 1, 2, 3, 4)$ are the 8-connected neighbors, and $y^{(4)}_{i\diamond t}$ $(t = 1, 2, 3, 4)$ are the 4-connected neighbors. Note that $x_i \in I_l$ implies $x_i \in I_h$, i.e. all LR pixels are naturally HR pixels.

We formulate the super resolution problem as a constrained least square problem as follows.

$$\begin{aligned} \min_{\{y,a,b\}} \quad & F(y,a,b) = \sum_{i\in W} \\ & \left(\|y_i - \sum_t a_t y^{(8)}_{i\diamond t}\|^2 + \lambda^2 \|y_i - \sum_t b_t y^{(4)}_{i\diamond t}\|^2 \right) \\ \text{s.t.} \quad & Sy = x \end{aligned} \qquad (2)$$

where W is a window of HR image I_h; λ^2 controls the importance of the horizontal and vertical correlation over diagonal correlation; Matrix S selects the HR pixels in W that are also in the LR image lattice.

This optimization problem can be easily solved using Gauss-Seidel algorithm as in [8]. Basically we solve $\{a, b\}$ and y iteratively until convergence (initial value of y can be obtained by bicubic interpolation):

$$\{a^{(n+1)}, b^{(n+1)}\} = \arg\min_{a,b} F(y^{(n)}, a, b), \qquad (3)$$

$$y^{(n+1)} = \arg\min_{y} F(y, a^{(n+1)}, b^{(n+1)}). \qquad (4)$$

IV. Super resolution for subpixel-downsampled images

The key factor that makes super resolution for subpixel downsampled image different from ordinary pixel-based super resolution, is that subpixel downsampled images are always full color images, and the R, G, B channels have slight subpixel shift compared to each other. Thus we can no longer convert the small images into YCbCr and then apply super resolution on Y channel only while the Cb, Cr channels are interpolated by simple convolution-based interpolators (such as bicubic), as done in most pixel-based super resolution algorithms [6], [9], [5], [2].

Therefore we propose to do the super resolution process in RGB space. For each separate color channel, we formulate the problem (2) with a small modification of matrix S. For example, the problem for R channel can be written as

$$\begin{aligned} \min_{\{y^r,a^r,b^r\}} \quad & F^r(y^r, a^r, b^r) \\ \text{s.t.} \quad & S^r y^r = x^r \end{aligned} \qquad (5)$$

where

$$\begin{aligned} & F^r(y^r, a^r, b^r) = \sum_{i\in W} \\ & \left(\|y^r_i - \sum_t a^r_t y^{r(8)}_{i\diamond t}\|^2 + \lambda^2 \|y^r_i - \sum_t b^r_t y^{r(4)}_{i\diamond t}\|^2 \right) \\ & = \|C^r(a^r, b^r) y^r\|^2, \end{aligned}$$

where $C^r(a^r, b^r) = [C^r_1(a^r)^T, \lambda C^r_2(b^r)^T]^T$ is a function of $\{a^r, b^r\}$ and

$$C^r_{1,i,j}(a^r) = \begin{cases} 1, & \text{if } y^r_j \text{ is the } i\text{th pixel in R channel} \\ -a^r_t, & \text{if } y^r_j \text{ is the } t\text{th diagonal neighbor of } y^r_i \\ 0, & \text{otherwise} \end{cases}$$

C_2 is similar to C_1 except that $C_{2,i,j}(b^r) = -b^r_t$ when y^r_j is the tth 4-connected neighbor of y^r_i. And

$$S^r(i,j) = \begin{cases} 1, & \text{if } y^r_j \text{ is } i\text{th LR pixel in R channel} \\ 0, & \text{otherwise} \end{cases}$$

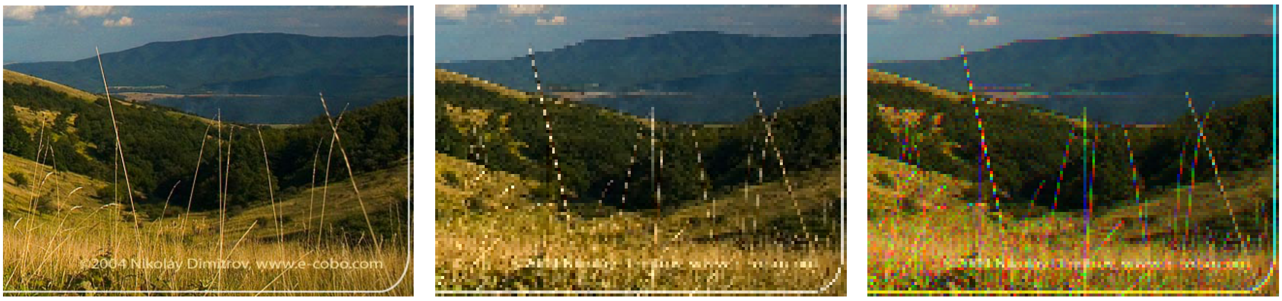
Put all RGB channels together we have

$$\begin{aligned} \min_{\{y,a,b\}} \quad & F(y,a,b) \\ \text{s.t.} \quad & \begin{bmatrix} S^r & & \\ & S^g & \\ & & S^b \end{bmatrix} \begin{bmatrix} y^r \\ y^g \\ y^b \end{bmatrix} = \begin{bmatrix} x^r \\ x^g \\ x^b \end{bmatrix} \end{aligned} \qquad (6)$$

where $a = (a^r, a^g, a^b)$, $b = (b^r, b^g, b^b)$, $y = (y^r, y^g, y^b)$, and

$$\begin{aligned} & F(y,a,b) = \sum_{i\in W} \sum_{c\in(R,G,B)} \\ & \left(\|y^c_i - \sum_t a^c_t y^{c(8)}_{i\diamond t}\|^2 + \lambda^2 \|y^c_i - \sum_t b^c_t y^{c(4)}_{i\diamond t}\|^2 \right) \\ & = \|C^r y^r\|^2 + \|C^g y^g\|^2 + \|C^b y^b\|^2 \end{aligned}$$

The optimization problem can be solved by iteratively solving coefficients $\{a, b\}$ and HR pixel values y as follows.



(a) original (b) DPD (c) DDSD

Fig. 2: Part of downsampled image "tree" by factor of 3.

A. Solving Coefficients

Let $u = (u^r, u^g, u^b)$ be the vector of HR RGB pixels inside window W, $v = (v^r, v^g, v^b)$ be the vector of HR RGB pixels outside window W. Then we have y is the concatenation of u and v, with necessary re-ordering of elements. To solve $\{a, b\}$ we have

$$\{a, b\} = \arg\min_{a,b} F(a, b, y^{(n)}) = \|u - Aa\|^2 + \lambda^2 \|u - Bb\|^2,$$

Since a and b are naturally uncoupled, we have

$$\begin{aligned} \hat{a} &= \arg\min_{a} \|u - Aa\|^2 = (A^T A)^{-1} A^T u \\ \hat{b} &= \arg\min_{b} \|u - Bb\|^2 = (B^T B)^{-1} B^T u \end{aligned}$$

where A, B are the matrices formed of 8-connected and 4-connected neighboring pixels of u. Basically the ith row of matrix A consists of the 8-connected neighbors $u^{(8)}_{i\diamond t}, t = 1, \ldots, 4$ of u_i. The ith row of matrix B consists of the 4-connected neighbors $u^{(4)}_{i\diamond t}, t = 1, \ldots, 4$ of u_i.

B. Solving HR pixels

Let $x = (x^r, x^g, x^b)$, $S = diag(S^r, S^g, S^b)$, $C = diag(C^r(\hat{a}, \hat{b}), C^g(\hat{a}, \hat{b}), C^b(\hat{a}, \hat{b}))$. Note at this step now C is determinant matrix rather than a function of $\{a, b\}$, since for each $C^c(\hat{a}, \hat{b}) = [C^c_1(\hat{a}), \lambda C^c_2(\hat{b})]$ we can calculate $C^c_1(\hat{a})$ and $C^c_2(\hat{b})$ and estimate λ using the fitting errors of a and b similar to [8], i.e. $\lambda^2 = \|u - Aa\|^2 / \|u - Aa\|^2$. We treat v as known variable and only solve u via the following problem

$$\begin{aligned} \hat{y} &= \arg\min_{y} \|C(\hat{a}, \hat{b}) y\|^2 = \arg\min_{u} \|D_u u + D_v v\|^2 \\ \text{s.t.} \quad & Sy = x \iff S_u u = x \end{aligned} \tag{7}$$

where $C = [D_u, D_v]$, $S = [S_u, S_v]$. D_u is a matrix composed of the columns of C corresponding to vector u, and D_v is a matrix composed of the other columns of C corresponding to vector v. So is S_u and S_v. The reason we can substitute S with S_u is that x is only related to u, i.e. the pixels inside window W. Then the problem can be solved using KKT system [1]:

$$\begin{bmatrix} u \\ \nu \end{bmatrix} = \begin{bmatrix} D_u^T D_u & S_u^T \\ S_u & 0 \end{bmatrix}^{-1} \begin{bmatrix} D_u^T D_v v \\ x \end{bmatrix} \tag{8}$$

where ν is the Lagrange multiplier. Note what we need to do is just updating the pixels in vector u and continue to next iteration.

V. Experiment results

To evaluate the performance of the proposed algorithm, we conduct experiments on several commonly used images. Most of them contain complex textures which are difficult for super resolution. The LR images are produced by downsampling HR images using DDSD method, thus LR images are of $1/3$ size of the HR images both horizontally and vertically. Note for grayscale images, DDSD method will first clone the intensity value to R, G, B channels and then downsample them.

Table I shows the PSNR comparison for grayscale images and Table II shows the comparison results for full color images. **Bi-sp** means downsampling images using **s**ubpixel-based method DDSD, and upsampling the small images using **p**ixel-based method bicubic. Similarly **SAI-pp** means **p**ixel-based downsampling method nearest neighbor downsampling and **p**ixel-based algorithm SAI, and so on. **SASR** is the Subpixel based Autoregressive Super Resolution that we propose. Note SAI is designed for upsampling an image with factor of 2, to apply SAI method with factor of 3 we first interpolate the LR image to 2 times of original size using SAI, then interpolate it 1.5 times using bicubic.

We see in Table I that SASR has averagely 4 dB gain over SAI-sp. SAI-sp results are achieved by applying SAI algorithm on Y channel only. The UV channels are interpolated by bicubic convolution. Since both SAI and SASR use same PAR model, we believe this result supports our argument that for subpixel-downsampled images, conventional pixel-based SR algorithms are not suitable. Besides, SASR has over 1 dB gain over SAI-pp. This result shows the superiority of subpixel-downsampling algorithms from the other aspect. Because SASR uses same PAR prior as SAI, the fact that SASR is better than SAI-pp implies subpixel-downsampled images contain more information, thus produces better super-resolved image.

Fig. 3 shows the super-resolved images of grayscale **baboon** image. We can see that both SAI-sp and SAI-pp generates unnatural textures and broken lines. SAI-sp result is slightly better than SAI-pp at the regions of beard. On the other hand, SASR method generates much better result. The textures of hair are much more natural and finer. And the lines of beard are more continuous.

Table II shows the PSNR results for full color images. We

TABLE I: PSNR (dB) comparison of grayscale images.

Image	Bi-sp	Bi-pp	SAI-sp	SAI-pp	SASR
baboon	22.19	21.18	19.93	21.25	**22.87**
bike	22.07	21.35	19.15	21.78	**23.33**
flower	20.41	19.55	17.74	19.58	**21.05**
lena	30.46	30.25	25.92	31.00	**32.52**
necklace	19.01	18.41	16.12	18.40	**19.90**
parrot	30.04	29.41	26.94	30.10	**31.58**
building	21.86	21.34	19.30	21.94	**23.43**
tree	24.78	24.28	21.77	24.48	**25.95**
Average	23.85	23.22	20.86	23.57	**25.08**

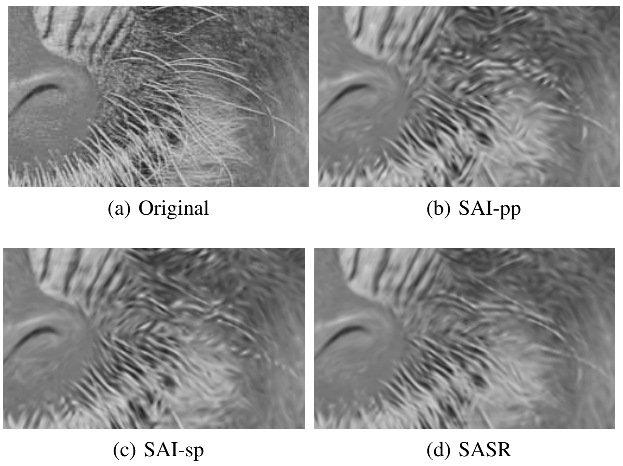
TABLE II: PSNR (dB) result comparison of full color images.

Image	Bi-sp	Bi-pp	SAI-sp	SAI-pp	SASR
baboon	19.98	21.20	19.98	21.26	**21.49**
bike	19.24	21.39	19.51	21.81	**21.91**
flower	16.71	18.83	16.70	18.87	**18.91**
lena	26.38	30.43	26.55	31.04	**31.16**
necklace	16.16	18.42	16.16	18.39	**18.55**
parrot	25.81	28.97	26.03	29.49	**29.87**
building	19.34	21.33	19.96	22.00	**22.44**
tree	21.58	24.15	21.63	24.31	**24.46**
Average	20.65	23.09	20.81	23.40	**23.60**

have similar results as grayscale case, except that SASR has only a slightly higher PSNR than SAI-pp. This is because the color-fringing artifact is much more severe for color images, thus SASR recovered image has a relatively lower PSNR. However if we look at the visual quality shown in Fig. 4, we can notice that SASR recovered image has better visual quality, especially at the beard region of baboon.

VI. Conclusion

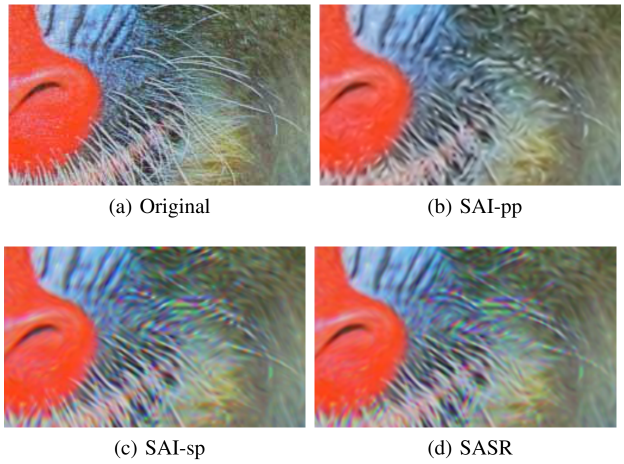
We propose a novel super resolution algorithm for subpixel-downsampled images. Because of the special characteristics of subpixel-based downsampling, traditional pixel-based SR algorithms cannot apply to subpixel-downsampled images. A special data degradation model has to be designed to fit the real image downsampling process. We design the data degradation term in this spirit and use the piecewise AR model as the pixel correlation model. Experiments show that the proposed algorithm is effective both in PSNR and visual quality.



(a) Original (b) SAI-pp

(c) SAI-sp (d) SASR

Fig. 3: Super-resolved grayscale **baboon** images.



(a) Original (b) SAI-pp

(c) SAI-sp (d) SASR

Fig. 4: Super-resolved color **baboon** images.

This algorithm is a very preliminary result. Our future effort will be put on utilizing the correlation between R, G, B channels and designing more sophisticated algorithms.

VII. Acknowledgement

This work has been supported in part by the Research Grants Council (RGC) of the Hong Kong Special Administrative Region, China. (GRF Project no. 610210)

References

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[2] W. Dong, L. Zhang, and G. Shi. Centralized sparse representation for image restoration. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1259–1266, 2011.

[3] L. Fang and A. O. C. Novel 2-D MMSE subpixel-based image down-sampling for matrix displays. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pages 986–989, 2010.

[4] L. Fang, K. Tang, O. Au, and A. Katsaggelos. Anti-aliasing filter design for subpixel down-sampling via frequency domain analysis. *Image Processing, IEEE Transactions on*, (99), 2011.

[5] S. Jian, X. Zongben, and S. Heung-Yeung. Image super-resolution using gradient profile prior. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[6] K. I. Kim and Y. Kwon. Single-image super-resolution using sparse regression and natural image prior. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PP(99):1–1, 2010.

[7] D. S. Messing and S. Daly. Improved display resolution of subsampled colour images using subpixel addressing. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–625–I–628 vol.1, 2002.

[8] K. Tang, O. C. Au, L. Fang, Z. Yu, and Y. Guo. Image interpolation using autoregressive model and gauss-seidel optimization. In *Image and Graphics (ICIG), 2011 Sixth International Conference on*, pages 66–69, 2011.

[9] J. Yang, J. Wright, T. Huang, and Y. Ma. Image super-resolution as sparse representation of raw image patches. *CVPR*, 2008.

[10] X. Zhang and X. Wu. Image interpolation by adaptive 2-D autoregressive modeling and soft decision estimation. *Image Processing, IEEE Transactions on*, 17(6):887–896, 2008.