Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

Science Letters:

Super-resolution inpainting

SHIH Timothy K., CHANG Rong-chi

(*Department of Computer Science and Information Engineering, Tamkang University, Taipei, Taiwan 25137, China*)

E-mail: tshih@cs.tku.edu.tw; roger@mail.mine.tku.edu.tw

Received Apr. 20, 2005; revision accepted Apr. 30, 2005

Abstract: Image or video resources are often received in poor condition, mostly with noise or defects making the resources hard to read. We propose an effective algorithm based on digital image inpainting. The mechanism can be used in restoring images or video frames with very high noise or defect ratio (e.g., 90%). The algorithm is based on the concept of image subdivision and estimation of color variations. Noises inside blocks of different sizes are inpainted with different levels of surrounding information. The results showed that an almost unrecognizable image can be recovered with visually good result. The algorithm can be further extended for processing motion picture with high percentage of noise.

Keywords: Image inpainting, Super resolution, Image processing
doi:10.1631/jzus.2005.A0487 **Document code:** A **CLC number:** TP391

INTRODUCTION

Video sequences can be interfered with by noise. When the noise ratio of a video is high, even though it is not possible to restore the original, to repair video frames or damaged artifacts with visually good result is a hopefully achievable goal (see example in Fig.3), especially for archival resources in digital museum projects. One of the simplest but efficient mechanisms is presented in (Oliveira *et al.*, 2001). The diffusion kernel can repair damages of small area with a barrier to deal with high contrast edges. The computation presented in (Oliveira *et al.*, 2001) had been proved to be very fast. By using a similar interpolation concept, with extension to allow different thresholds on different interpolated pixels, the mechanism discussed in (Bornard *et al.*, 2002) allows an adaptive computation scheme. Thus, inpainted pixels far away from inpainting boundary can be calculated to achieve better visual result. Considering inpainting from another perspective, the concept of isophote lines reaching the boundary of an inpainted area is introduced in (Bertalmio *et al.*, 2000). By preventing crossed isophote lines, the mechanism can partially preserve structural information in a damaged image. The above mechanisms are computations in the spatial domain of an image via pixel interpolation or extrapolation. With proper heuristics, structural information can be reconstructed to achieve a reasonable effect. However, sometimes the resulting image is blurred. One drawback of the spatial domain approach is that textural information is hard to restore. The work discussed in (Yamauchi *et al.*, 2003) uses a multi-resolution texture synthesis mechanism to combine both spatial and frequency information, which can achieve a better result in restoring textures. Another example combining spatial and frequency information for restoration is also found in (Bertalmio *et al.*, 2003).

Algorithms discussed in the previous paragraph are effective and efficient in restoring damages of small and discontinuous areas. To restore damages of larger area, for instance, to remove a person from a picture, it is necessary to take another approach. Simple pixel interpolation or texture synthesis cannot produce visually good result. One useful approach is to replace the selected area (e.g., a person) with background information copied from the same picture. The exemplar-based synthesis approach (Criminisi *et al.*, 2004) computes a priority for each patch to be selected to inpaint the removed area. After a target area is inpainted, a confidence value is assigned or re-computed. The confidence values are then used for computing the next inpainting area. The work discussed in (Drori *et al*., 2003) also uses a confidence map (values) and adaptive neighborhood search (finding useful patch of the highest priority). In addition, the picture is refined from coarse level to fine level, which will take several iterations on the same inpainted area. The initial inpainted information relies on a fast estimation scheme known as the push-pull filtering method, to down-sample and up-sample the image in a multi-resolution hierarchically.

The algorithm proposed in this article looks at a damaged picture or video frame at different levels of resolutions. Color distribution variance is used in determining inpainting strategy. We present the algorithm in the next section with results shown in the appendix.

THE SUPER-RESOLUTION INPAINTING ALGORITHM

Considering an image or a video frame as a large Damaged Image Block (DIB), as shown in Fig.1, a DIB can be subdivided into several Image Blocks (IBs), each of which may or may not contain damaged pixels. Furthermore, each Image Block is subdivided into several Pixel Blocks (PBs) which are elementary objects to be inpainted. The recursive algorithm SR_Inpaint(DIB) (Fig.2) takes a Damaged Image Block as input, and produces a visually good result.

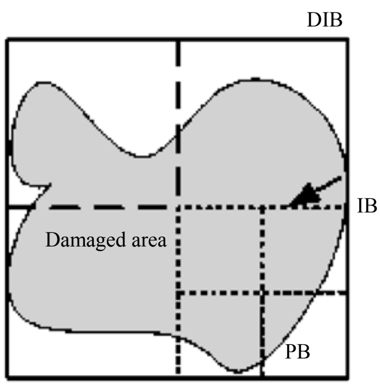

Fig.1 Damaged Image Block (DIB), Image Block (IB) and Pixel Block (PB)

$SR\_Inpaint(DIB)\leftrightarrow$
$DIB=\in$ or
$\forall IB\in DIB\cdot(var(IB)>\alpha\rightarrow SR\_Inpaint(IB)$ or
$\forall PB\in IB\cdot(DamagePercent(PB)>\beta_2$
$\rightarrow Fill(PB, MeanColor(IB))$
$\beta_2\geq DamagePercent(PB)>\beta_1$
$\rightarrow Fill(PB, MeanColor(PB))$
$\beta_1\geq DamagePercent(PB)\rightarrow Interpolate(PB))$

Fig.2 Functional representation of the super-resolution inpainting algorithm

The basic assumption is that color variance is a strong indication of the degree of details in an Image Block. The threshold α sets the criterion on whether a recursive call to the super-resolution inpainting algorithm is required (the algorithm terminates when the *DIB* is small). In our implementation, the value of α is a percentage of color variance of an IB (i.e., the maximum $var(IB)$ is 100). If the color variance of IB is greater than the threshold α, the algorithm is called recursively to handle the next level of details. Otherwise, the algorithm further divides an IB into several pixel blocks (i.e., PBs). Another criterion is the percentage of damaged pixels. We argue that if the percentage is too high, using surrounding color information to fix a pixel is less realistic as compared to using a global average color. In some severe cases, it is impossible to use neighborhood colors. Note that, both thresholds are adjustable for the sake of analysis. The algorithm iterates through each of the PBs in an IB. If the percentage of damaged pixels in a PB is too high (i.e., greater than β_2), the mean IB color is used [i.e., $MeanColor(IB)$]. One example is that the entire PB is damaged, so we must use the mean IB color. The function $DamagePercent(PB)$ simply counts the number of damaged pixels in a PB. And, the function $Fill(PB, C)$ takes a Pixel Block and a color C, and fills the Pixel Block with the color. Alternatively, if the percentage is still high (i.e., greater than β_1), the mean PB color is used. Note that, the computation of mean colors does not take damaged pixels into account. If the percentage is low enough (i.e., less than β_1), neighbor pixels are used for interpolation. The function $Interpolate(PB)$ implemented in our algorithm uses a bi-linear interpolation technique.

We further take an optimization step to improve the algorithm. When the filling function is called, we add noise on the boundaries of Pixel Blocks. The super resolution inpainting algorithm is called again to remove these bounding boxes. Thus, block effect is reduced.

There are three thresholds in the above algorithm, α, β_1 and β_2. We use all combinations of the following values:

$$\alpha = 50, 60, 70, 80$$
$$\beta_1 = 65, 70, 75, 80, 85, 90$$
$$\beta_2 = 95$$

The selection of β_2 is aimed at testing the usage of mean color. Unless a pixel block is completely damaged, the mean color should be used. Thus, the selection of β_2 should be high. Since $\beta_1<\beta_2$, we select the values of β_1 accordingly. The threshold α is used to check the color variance. We try to cover a wide spectrum. The combinations of the above thresholds are all tested using more than 1000 pictures. The values of α, β_1 and β_2 have great impact on the outcome. In general, if α is less than 70, the average *PSNR* values of repaired pictures with respect to other parameters are stable. So we use α=80 in our implementation of an automatic inpainting tool. We chose β_2=95 through our analysis. This means that unless the percentage of damaged pixels in a pixel block is higher than 95, the mean color of an outside big block should not be used. The value of β_1 is critical. If β_1 is less than 60, the result is not as good as expected. However, the *PSNR* values of the fixed pictures become stable when the value of β_1 becomes 80, 85, or 90. Conclusively, we choose α=80, β_1=85, and β_2=95.

The analysis takes another perspective on the noise percentage of damaged pixels. Fig.4 presents our experimental results. The first row shows the damaged images and the percentages of noise pixels. The second row presents the inpainted images and PSNR values. The results showed that even with very high percentage of noise, the picture is still visible.

CONCLUSION

The article presents an effective algorithm for image inpainting, which achieves visually good result, even with very high percentage of noises in the pictures. We have tested more than 1000 pictures of different categories, including Chinese and western paintings, photos, and cartoon drawings. The experimental results proved that super resolution inpainting is especially powerful for photos. This result encourages us to take a further step to extend the proposed mechanism for video inpainting. Our objective is not to detect spikes or long vertical lines commonly reported in the literature. We aim to develop a technique to recover video signal with very high percentage of noise and achieve visually good output.

References

Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C., 2000. Image Inpainting. ACM SIGGRAPH Conference on Computer Graphics, SIGGRAPH 2000, p.417-424.

Bertalmio, M., Vese, L., Sapiro, G., Osher, S., 2003. Simultaneous structure and texture image inpainting. *IEEE Transactions on Image Processing*, **12**(8):882-889.

Bornard, R., Lecan, E., Laborelli, L., Chenot, J.H., 2002. Missing Data Correction in Still Images and Image Sequences. Proc. ACM International Conference on Multimedia, p.355-361.

Criminisi, A., Perez, P., Toyama, K., 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions Image Processing,* **13**:1200-1212.

Drori, I., Cohen-Or, D., Yeshurun, H., 2003. Fragment-based image completion. *ACM Transactions on Graphics*, **22**(3):303-312.

Oliveira, M.M., Bowen, B., McKenna, R., Chang, Y.S., 2001. Fast Digital Image Inpainting. International Conference on Visualization, Imaging and Image Processing, p.261-266.

Yamauchi, H., Haber, J., Seidel, H.P., 2003. Image Restoration Using Multiresolution Texture Synthesis and Image Inpainting. Computer Graphics International, CGI'03, p.108-113.

APPENDIX A

We add random noises to the original pictures illustrated in Fig.A1a. The damaged pictures are shown in Fig.A1b, recovered images are shown in Fig.A1c. The noise rations with respect to the areas of original pictures and the *PSNR* values of both damaged and recovered pictures are given in Table A1.

(a) (b)

Fig.3 Super-resolution inpainting. (a) Image with 90% Defects; (b) The inpainted result

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

Fig.4 Experimental results in different noise conditions

(a) Noise ratio=61.2%, *PSNR*=30.1 dB; (b) Noise ratio=73.1%, *PSNR*=28.16 dB; (c) Noise Ratio=80.6%, *PSNR*=30.9 dB; (d) Noise Ratio=90.3%, *PSNR*=22.1 dB

Table A1 PSNR values of image with different ratios of noise

Figures	Noise ratios	*PSNR* values (dB)	
		Damaged	Recovered
#1	53.6%	6.88	29.86
#2	60.9%	8.40	26.89
#3	70.1%	6.31	24.31
#4	81.2%	5.68	25.56
#5	90.3%	5.29	28.83

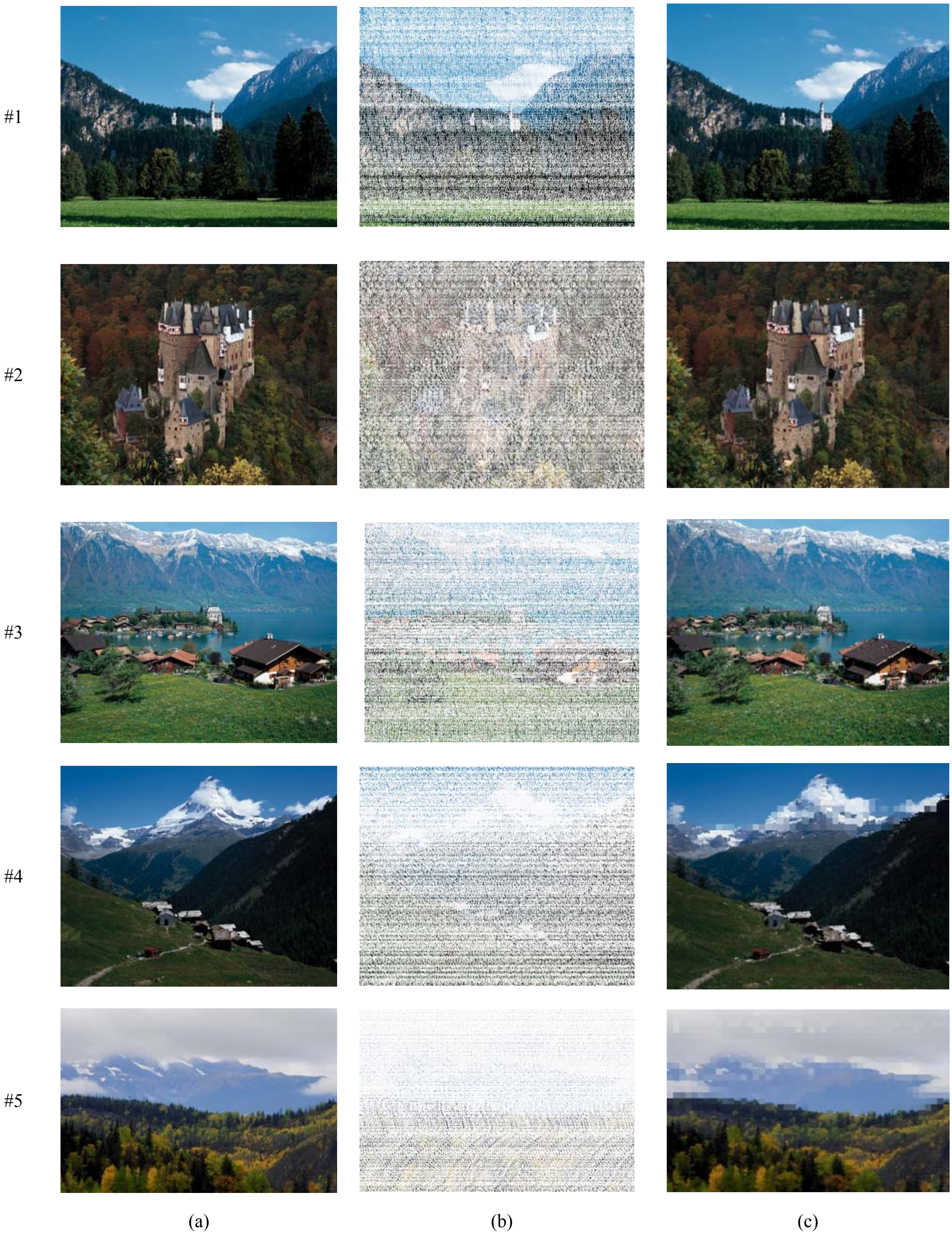

Fig.A1 Experiment results by the super-resolution inpainting algorithm
(a) Original pictures; (b) Pictures with noise; (c) Recovered pictures