IJCSC International Journal of Computer Science & Communication Vol. 1, No. 1, January-June 2010, pp. 193-197

Digital Inpainting Based Image Restoration

S. R. Goyal

Computer Engineering Department, S.V.National Institute of Technology, Surat, Gujrat, INDIA
srg@coed.svnit.ac.in.

ABSTRACT

Inpainting is an image interpolation problem with broad applications in image and vision analysis. Digital inpainting is related to automatic filling-in of user-selected regions in digital images. It refers to removal of image defects such as scratches and blotches as well as removal of disturbing objects as, for instance, subtitles, logos, dates etc., and restoring the same with information surrounding them, so that it blend seamlessly with rest of the image.

An extensive survey of previous work related to the digital inpainting is presented to fill in the gap and a new algorithm is proposed, which is motivated by the quality of results, easier implementation, efficiency, and effectiveness. The restored images obtained with this algorithm look plausible in general and surprisingly good in some cases. In addition, it is also having edge over other algorithms in terms of computation time and image quality while filling-in large region.

Keywords: Image Inpainting, Variation Approach, PDE.

1. INTRODUCTION

Bertalmio et al. [1] have introduced a technique for the problems mentioned above named as digital inpainting. "Inpainting" is to restaurate missing or damaged parts of an image. Application of digital inpainting ranges from the restoration of paintings to scratched photographs or films to the removal or replacement of arbitrary objects in images such as the aforementioned. Inpainting can furthermore be used to create special effects (e.g., in movies). Ultimately, inpainting should be carried out in such a way that when viewing the end-result, it should be almost impossible for an arbitrary observer to determine that the image has been manipulated or altered. Digital inpainting process image automatically without any interaction required by the user after user selects the region to be inpainted. Next, information is propagated inward from the region boundaries, i.e., the known image information are used to fill in the missing areas.

2. RELATED WORK

There has been considerable work in the field of digital inpainting, approaching the problem from the directions of noise removal (due to compression and transmission), film and video artifact removal, and texture synthesis. Bertalmio et al. [1] pioneered a digital image inpainting algorithm based on partial differential equations (PDEs). A number of algorithms specifically address the image filling issue for the task of image restoration, where speckles, scratches, and overlaid text are removed [1, 6, 9, 10]. These image inpainting techniques fill holes in images by propagating linear structures into the target region via diffusion. They are inspired by the partial differential equations of physical heat flow and work convincingly as restoration algorithms. Their drawback is that the diffusion process introduces some blur, which becomes noticeable when filling larger regions. Inspired by the work of Bertalmio et al., Chan and Shen proposed two imageinpainting algorithms [8, 9]. The Total Variational (TV) inpainting model [8] uses an Euler-Lagrange equation and inside the inpainting domain the model simply employs anisotropic diffusion [7] based on the contrast of the isophotes. This model was designed for inpainting small regions and while it does a good job in removing noise, it does not connect broken edges (single lines embedded in a uniform background) [8]. The Curvature-Driven Diffusion (CDD) model [9] extended the TV algorithm to also take into account geometric information of isophotes when defining the "strength" of the diffusion process, thus allowing the inpainting to proceed over larger areas. CDD can connect some broken edges, but the resulting interpolated segments usually look blurry. Masnou and Morel [11, 12] proposed an inpainting algorithm based on level lines. This variational approach was based on the calculation of generalized elastica energy, and joining the level lines having the lowest energy of all compatible T-junctions. The angle with which the level lines arrive at the boundary of the holes is not preserved, and the algorithm uses straight lines to join equal gray value pixels.

Digital inpainting has found its wide application in image processing, vision analysis, and the movie industry. Recent examples include: automatic scratch removal in digital photos and old films [1, 8], text erasing [1, 3, 8, 9, 10], special effects such as object disappearance [1, 2, 4, 5, 9], disocclusion [11, 12], spatial/temporal zooming and super-resolution [8, 10, 13], lossy perceptual image coding [8], and removal of the laser dazzling effect [9], and so on. On the other hand, in the engineering literature, there also have been many earlier works closely related to inpainting, which include image interpolation [13, 14, 15], image replacement [16] and error concealment [17] in communication technology. As scattered as the applications are, the methods for the inpainting related problems have also been very diversified, including the nonlinear filtering method, the Bayesian method, wavelets and spectral method, and the learning-and-growing method especially for textures (see, for example, recent work [16, 18]).

3. THE METHOD

The implemented algorithms are the Digital Image Inpainting Algorithm [1], Fast Digital Image Inpainting Algorithm [3], Multiscale Method for Automated Inpainting [4], and Exemplar Based Inpainting Algorithm [2].

The choice of implemented algorithms is motivated by the quality of results, easier implementation, efficiency, and effectiveness. In addition, the first three algorithms focus on small regions while the fourth algorithm focuses mainly on large regions. While the first method uses a PDE based approach, the second method shows that virtually same result may be accomplished by using a simple convolution operation commonly used in image processing.

3.1. Digital Image Inpainting

A pioneering inpainting algorithm regarding the use of partial differential equations (PDE's) is presented in [1]. The basic idea is to iteratively fill Ω by prolonging the isophote lines arriving at $\partial\Omega$. The direction may be obtained by for each pixel along $\partial\Omega$ computing a discretized gradient vector and rotating this vector by $\pi/2$ radians. Each color channel (red, green, and blue) of the image is treated as separate gray-level images. To estimate the variation of the intensity, a discrete two-dimensional laplacian is used. This estimation is spread in the isophote direction in order to maintain smooth intensity changes. To conclude the method: In order to get a visually pleasing result it is important to propagate both the geometry (the gradient direction) and the photometry (the intensity).

3.2. Fast Digital Image Inpainting

The method presented in [3] is specifically designed for inpainting relatively small regions. When focusing on small regions simpler models can be used. Instead of relying on any specific mathematical geometrical theories, this method takes into account a constraint of the sampling theorem.

The inpainting is performed by repeatedly convolving Ω using isotropic diffusion (i.e., using the linear heat equation). The diffusion process propagates information from $\partial\Omega$ into Ω. The number of iterations may be predefined or determined by looking at the value of each pixel belonging to Ω. The process is stopped if the changes of the value lie within a certain threshold.

When using the weighted sum of a pixel's neighborhood (e.g., as is the case when convolving an image with a kernel), noticeable blurring may be introduced. It is easy to realize that this implies that edges may be broken. The blurring is especially noticeable when the neighborhood or parts of the neighborhood of a pixel are made up of significant contrast changes. To solve this problem, a diffusion barrier may be introduced to stop the algorithm from performing inpainting where the contrast changes are significant.

3.3. Multiscale Method of Automated Inpainting

The technique presented in [4] is based on resolution of the images. If one reduces the resolution of an image by a factor of 2, 4 etc., one can still see the grossest features. Therefore, the regions to be inpainted are matched with rest of the image to find best match at lower resolution. The set of best matches is then used as the starting set for a similar matching process at a higher resolution. Regions around each of the best matches are then examined at the higher resolution to again find the best match with the original. This limits the search space, yet allows for some variation in best match from one scale to another. The process is repeated until the final target image resolution, whereupon the best match is used to generate the reconstructed pixel.

3.4. Object removal by exemplar based inpainting

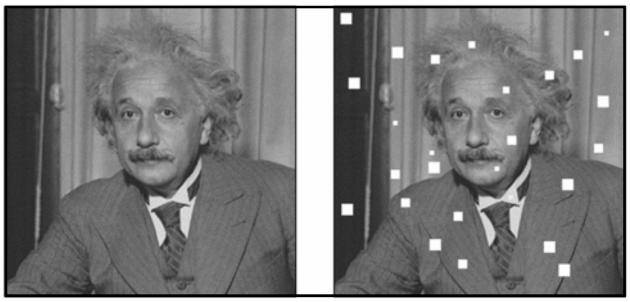
Another combined texture synthesis and inpainting algorithm is presented in [2]. An exemplar-based technique is used for replacing the region Ω. This kind of technique implies a computationally cheap and effective way of generating new texture. The texture generating is done by sampling and by copying color values from the remaining parts of the image. Exemplar based texture synthesis is well suited for preserving both texture and structure; thereby a separate mechanism for handling the isophotes is not needed. The algorithm is iterative and fills Ω by defining $\partial\Omega$ as the fill front.

The filling order is important. A best-first filling order based on every pixels priority value is used. The pixels are then filled in an order according to their priority values. The priority value is in turn based on a confidence term and a data term. The confidence term is updated at the end of each iteration. The data term takes into account the isophote direction for the current pixel. The current $\partial\Omega$ is filled with the best matching samples. The best match is based on a sum of squared differences (SSD).

4. RESULTS

Now, we will discuss the results of the implemented algorithms. The algorithms have been tested for different types of cases with different type of complexity. The original image, the used mask, and the corresponding results are shown for each case. In addition, data such as the computation time, the mask size, i.e., the total number of pixels in image, the total number of pixels filled, and image size of the image are presented.

The first thing to test is how the algorithms perform for simple cases with small areas to be inpainted. The tested regions are interesting since they are made of structure (straight and curved lines), discontinuous intensity levels, and texture. Figure 1 (a) and Figure 1 (b) show the original image and the test mask respectively.



(a) **(b)**

Fig 1: Einstein - before Inpainting. (a) The Original Image. (b) The Mask Image.

Figure 2 shows the results of various inpainting algorithms. We can clearly observe from the results that all inpainting algorithms are performing well except some areas where blurring is noticeable. The areas where blurring is noticeable are marked with boxes as shown in figure 2. Diffusion based algorithms have resulted in blurring in those areas. Notice the collar where noticeable blurring is introduced. The algorithms introduces a slight blurring on some portions concludes that these are not suitable for regions that grows beyond a certain size. When the size of the region increases the blurring gets more visible.

For this case computation time for various algorithms is presented in table1. From the table we can notice that computation required for fast digital image inpainting algorithm is minimum while for that of PDE based digital image inpainting algorithm [1] is maximum. In addition, the new proposed algorithm is performing better in case of image quality and computation time. Since areas to be filled are not so big the advantages of fix space exemplar search inpainting are not visible.

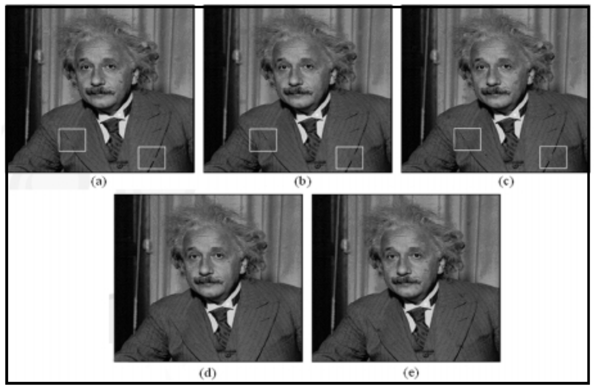


Fig 2: Einstein-after inpainting using various algorithms. (a) Multiscale algorithm, (b) Fast digital inpainting, (c) Digital image inpainting, (d) Exemplar based inpainting, (e) Fix space exemplar search inpainting.

Table 1
Computation Times for Filling in Small Missing Patches.
(Pixel Filled =1722, Total Number = 65536,
Image Size = 256 × 256 × 1).

Method	*Time taken on P IV, 2.4 GHz (sec)*
Multiscale inpainting	8.3900
Fast digital image inpainting	1.6870
Digital image inpainting	159.4840
Exemplar based inpainting	34.2810
Fix space exemplar search algorithm	5.1870

In this section all the algorithms are not considered since we are demonstrating the use of inpainting in special effects. Moreover, most of algorithms are diffusion based so they cause blur when area of larger size is restored. Diffusion based algorithms are digital image inpainting [1] and fast digital algorithm [3]. So these are not used. It has been noticed for the diffusion-based algorithms, that if region to be restored is of considerable size (more than 9x9), the blur introduced is noticeable. In addition, we are showing that our proposed algorithm is taking very less time in restoration and is very effective also. Figure 6.6 demonstrate the use of inpainting in special effects.

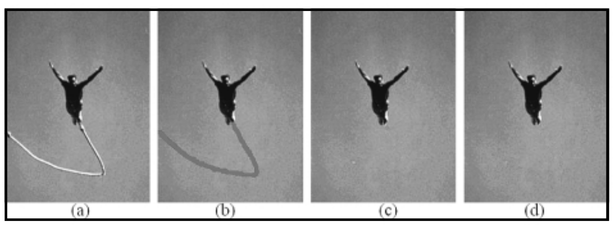


Fig 3: Bungee Jump Man- before and after Inpainting.

(a) Original image. (b) Mask image. (c) Result using exemplar inpainting. (d) Result using fix space exemplar search inpainting.

Table 2
Computation Times for Two Algorithms used to Create Special Effects.

Image	*Number of pixels filled*	*Total number of pixels in the image*	*Time taken by exemplar inpainting (sec)*	*Time taken by fix space exemplar search algorithm (sec)*
Bungee man	1202	47376	18	3.1570

From the figure 3, we can observe that stunning special effects can be produced in images and motion pictures. In addition, we can notice the time taken by our algorithm is considerably less than the original algorithm as shown in the table 2. We observe that the region that has been reconstructed is undetectable by an observer who is not familiar with original image. Since, diffusion based algorithms introduce blur in the region to be inpainted when it contains texture or edges or other complex structure, those are not considered in this section as discussed earlier. In future, these algorithms can be used to produce stunning visual effects in movies.

5. CONCLUSION

Digital image inpainting proposed by Bertalmio et al. [1] has introduced a technique for digital inpainting of still images that produces very impressive results. Their algorithm, however, usually requires several minutes on recent personal computers for the inpainting of relatively small areas. Such times are unacceptable for interactive sessions and motivate researchers to design a simpler and faster algorithm capable of producing similar results in just a few seconds. Digital image inpainting algorithm performs significantly better than the fast digital image inpainting algorithm regarding regions with discontinuous intensity levels. Again, this clearly depends on the size of the inpainting region. It is only recommended to use this algorithm for small regions. One advantage of the algorithm is that it performs well in re-creating structure. However, the significant amount of required computation time compared to the fast digital image inpainting algorithm is a clear disadvantage.

Fast digital image inpainting algorithm presented in [3] is a good solution to the problem of computation time for inpainting of small regions. This algorithm uses a simple convolution based approach to synthesize small missing portions of the images. The foremost advantage of the fast digital inpainting algorithm is the simplicity of the algorithm. The algorithm is very easy and straightforward to implement. Furthermore, the algorithm is efficient, especially regarding the computation time. However, the algorithm only produces satisfactory results for small regions, making it useful only for tasks such as removing super-imposed text and scratches. The algorithm is more or less unusable for large regions. Another significant disadvantage of the algorithm is that it cannot recreate structure. This is easy to realize as the isophotes are not taken into account at all. Furthermore, the algorithm introduces visible artifacts in the form of blurring for regions with significant intensity changes, i.e., where the intensity levels are discontinuous (e.g., white and black colors next to each other). This is due to the properties of convolution. In order to cope with this problem [3] suggests the use of a diffusion barrier to stop the diffusion process where the intensity is discontinuous. This has not been tested because it is easy to realize rather then its implementation. Similar is the case with multiscale method automated inpainting, but it produces results better than fast digital image inpainting algorithm in terms of image quality.

There are many advantages of the exemplar-based algorithm proposed in [2]. It is able to re-create texture as well as structure. The fact that it works for large arbitrary regions makes it the number one choice, because in most cases one wishes to inpaint large regions. The exemplar-based inpainting algorithm performs exhaustive searches in the image in order to find best matches according to certain conditions to fill the region. This enables the filling-in of large arbitrary regions.

However, exhaustive search to find best exemplar, increases computation time drastically for images with larger size. Therefore, a constrained space exemplar search algorithm has been proposed which uses fix space to search for best match. For this algorithm, it has been assumed that the best match can be found in a fix region around the pixel to be filled. Clearly, fix space exemplar search is having all the advantages of original exemplar based inpainting. In addition to these, the proposed approach requires very less time to inpaint large regions. Moreover, this algorithm uses a simple difference method to find the best match instead of using sum of squared differences like conventional exemplar algorithm.

REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image Inpainting" *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, SIGGRAPH2000, 2000, Page(s): 417-424.

[2] A. Criminisi, P. Perez, and K. Toyama, "Region Filling and Object Removal By Exemplarbased Image

Inpainting", *IEEE Transactions on Image Processing*, **13**, Issue 9, Sept 2004, Page(s): 1200 - 1212.

[3] M. M. Olivieira, B. Bowen, R. Mckenna, and Y. S. Chung, "Fast Digital Image Inpainting", *Proceedings of the International Conference on Visualization, Imaging and Image Processing VIIP 2001*, Marbella, Spain 2001, 3-5 Sep 2001, Page(s): 261 - 266.

[4] R. J. Cant and C. S. Langensiepen, "A Multiscale Method for Automated Inpainting", ESM2003, 2003.

[5] A. Criminisi, P. Perez, and K. Toyama, "Object Removal by Exemplar based Inpainting", *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **2**, 18-20 Jun 2003, Page(s): II 721 - II-728.

[6] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting", *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, 8-14 Dec2001, Page(s): I-355-362.

[7] P. Perona and J. Malik, "Scale-Space Edge Detection using Anisotropic Diffusion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, Issue 7, Jul 1990, Page(s): 629 - 639.

[8] T. F. Chan and J. Shen, "Mathematical Models for Local Non-Texture Inpainting", *SIAM Journal of Applied Mathematics*, **62**, Number 3, 2001, Page(s): 1019 - 1043.

[9] T. F. Chan and J. Shen, "Non-Texture Inpainting by Curvature-Driven Diffusions (CDD)", *Journal Visual Communication and Image Representation*, **12**, Number 4, 2001, Page(s): 436 - 449.

[10] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by Joint Interpolation of Vector Fields and Gray Levels", *IEEE Transaction on Image Processing*, **10**, Issue 8, Aug 2001, Page(s): 1200 - 1211.

[11] S. Masnou, "Disocclusion: A Variational Approach using Level Lines", *IEEE Transactions on Signal Processing*, **11**, Issue 2, Feb 2002, Page(s): 68- 76.

[12] S. Masnou and J.M. Morel, "Level Lines based Disocclusion", *Proceedings of the IEEE International Conference on Image Processing*, 1998, Page(s): 259 - 263.

[13] V. Caselles, J. M. Morel, and C. Sbert, "An Axiomatic Approach to Image Interpolation", *IEEE Transactions on Image Processing*, **7**, Issue 3, Mar 1998, Page(s): 376 - 386.

[14] A. C. Kokaram, R. D. Morris, W. J. Fitzgerald, and P. J. W. Rayner,"Detection of Missing Data in Image Sequences", *IEEE Transactions on Image Processing*, **4**, Issue 11, Nov 1995, Page(s): 1496 - 1508.

[15] A. C. Kokaram, R. D. Morris, W. J. Fitzgerald, and P. J. W. Rayner, "Interpolation of Missing Data in Image Sequences", *IEEE Transactions on Image Processing*, **4**, Issue 11, Nov 1995, Page(s): 1509 - 1519.

[16] H. Igehy and L. Pereira, "Image Replacement through Texture Synthesis", *Proceedings of the IEEE International Conference on Image Processing*, **3**, 26-29 Oct 1997, Page(s): 186 - 189.

[17] W. Kwok and H. Sun "Multidirectional Interpolation for Spatial Error Concealment", *IEEE Transactions on Consumer Electronics*, **39**, Issue 3, Aug 1993, Page(s): 455 - 460.

[18] A. Aldroubi and K. Gr¨ochenig, "Nonuniform Sampling and Reconstruction in Shiftinvariant Spaces", *SIAM Review*, **43**, Number 4, 2001, Page(s): 585 - 620.