



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 Issue: VII Month of publication: July 2014

DOI:

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Analysis and Comparison of Spatial Domain Digital Image Inpainting Techniques

Prof. Mrs Anupama Sanjay Awati¹, Prof. Dr. Mrs. Meenakshi R. Patil²

¹Asst. Prof. Dept of Electronics and communication

KLS Gogte Institute of Technology, Belgaum.

²Prof. Dept of Electronics and communication

A G. M. R. College of Engg & Tech, Hubli.

Abstract— Digital Image Inpainting is an interesting research topic in image processing. Digital Image Inpainting uses spatial or frequency information to restore partially damaged images and paintings. Digital Image Inpainting Algorithms aim at filling the missing or broken regions of an image so that it appears continuous and natural for not so familiar user. This paper compares the inpainted images obtained by Olivera's, modified Olivera's, median filtering, and directional median filtering, convolution using Noori et al's method and bilateral inpainting for an image with 10 pixel broad damage. A large number of experiments show that the two algorithms can efficiently restore the structure and texture information. Results are compared based on the Peak Signal to Noise Ratio (PSNR) and are better than the other algorithms.

Index Terms Median filtering, Directional median filtering, Bilateral filtering, Noori et al's Algorithm, Olivera's Algorithm and Hadhoud Algorithm

I INTRODUCTION

Image inpainting is the art and science behind reconstructing parts of an image in a visually undetectable way. Inpainting has been an art for many centuries, being carried out by a skilled image restoration artist. Research in partnership with these artists and the general increase in computing power gave rise to sophisticated algorithms for recovering the lost or corrupt portions of an image.

While image inpainting is related to noise removal, and sometimes algorithms share a majority of their ideas, image inpainting is fundamentally a different problem to noise removal. Noise often carry some sort of information about their underlying data. For example, additive noise contains the original image data, but with some amount of noise "added" to the image. Image inpainting, on the other hand, the "lost" regions contain absolutely no information relating to the original data. Despite the significant differences between the existent techniques, in the authors manage to draw the main directions that each method must follow. Accordingly, the four

steps that make up the underlying methodology of any inpainting algorithm are given as:

1. "The global picture determines how to fill in the gap, the purpose of inpainting being to restore the unity of the work.
2. The structure of the area surrounding the hole is continued into the gap, contour lines are drawn via the prolongation of those arriving at the boundary of the inpainting domain.
3. The different regions inside the inpainting domain, as defined by the contour lines, are filled with color, matching those of the boundary of the region to be inpainted;
4. The small details are painted (e.g. little white spots on an otherwise uniformly blue sky): in other words, *texture* is added."

Inpainting is used in a primitive form in certain image editing software. They expect the user to specify the area to be inpainted and also specify the sample that has to be put in its place. Digital image inpainting requires the user to specify the area to be inpainted, but fills it automatically using the

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

information available in the surrounding area of the same image. The inpainting problem is also addressed under disocclusion, object removal, image editing etc. The propagation of the information into the inpainted area determines the success of the algorithm. The geometric and the photometric propagation which is generally called as structure and texture propagation poses a major challenge to inpainting algorithms. The algorithms found in literature can be classified as PDE based, convolution based, texture based exemplar based super resolution based DWT based pyramid based and so on, But these algorithms can be broadly classified as diffusion based and exemplar (non diffusion based). The diffusion based algorithms are isotropic (PDE) and anisotropic like convolution.

The first category of methods, known as diffusion-based inpainting, introduces smoothness priors via parametric models or partial differential equations (PDEs) to propagate (or diffuse) local structures from the exterior to the interior of the hole. Many variants of PDE use different models (linear, nonlinear, isotropic, or anisotropic) to favor the propagation in particular directions or to take into account the curvature of the structure present in a local neighborhood. These methods are naturally well suited for completing straight lines, curves, and for inpainting small regions. They, in general, avoid having unconnected edges that are perceptually annoying. However, they are not well suited for recovering the texture of large areas, which they tend to blur.

The exemplar(non diffusion)based category of method does very well to preserve, propagate and extend linear image structure. Exemplar-based texture synthesis is based around finding patches in the rest of the image which are similar to the known parts of the image patch to be filled, and then copying the missing information over. This mechanism preserves image isophotes of the known parts of a target patch. Filling order is critical, and has a large impact on the resulting quality. The algorithm first estimates the importance, or priority of our current boundary pixels of the region to be inpainted, with respect to image isophotes. It then finds the border pixel with the largest priority and copies the missing information by finding a similar patch from the source regions.

In order to understand the algorithms we need to understand the terms used in image inpainting literature. The original image is described as I , the area that has to be inpainted is denoted as Ω , the border of this region and the known region also known as source region is indicated by ϕ . This is shown in Figure



Fig1. Notation Diagram

II SPATIAL DOMAIN IMAGE INPAINTING.

Repairing of the damaged regions of an image can be carried out using convolution in spatial domain. The regions to be inpainted are convolved with a predefined diffusion mask repeatedly.

A convolution-filter based algorithm repairs an image by convolving the neighborhood of the damaged pixels with a diffusion mask. Oliveira et al. [1] presents a fast image inpainting algorithm which uses convolution operation. In this algorithm, the regions to be inpainted are convolved with a predefined diffusion mask repeatedly. The central weight of the diffusion mask is considered zero, because its related pixel in the original image is unknown.

$$\begin{matrix} a & b & a \\ b & 0 & b \\ a & b & a \end{matrix}$$

Fig 2 Diffusion mask for Oliveira's Algorithm where $a=0.073235$, $b=0.176765$, $c=0.125$

Hadhoud et al [2] presents a modification of Oliveira algorithm permitting an implementation time reduction. Both above convolution based algorithms are very quick but they don't have very good results in high contrast damaged edges. This method introduces a modification to Oliveira algorithm to reduce the time of inpainting and increase the quality of the results based on the modification of convolution stage.

$$\begin{matrix} a & b & a \\ b & a & b \\ a & b & 0 \end{matrix}$$

Fig 3 Diffusion mask for Hadhoud's Algorithm. where $a=0.073235$, $b=0.176765$, $c=0.125$

Another convolution based image inpainting is proposed in Noori et al. [5]. The main idea of the proposed convolution based inpainting algorithm, is to use an adaptive kernel permitting a better processing edge regions. The gradient of known pixels in the neighborhood of a missed pixel is used to compute weights in convolving mask. Since gradient values in edge regions are large, and contribution of pixels in adjacent of

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

edges should be less than contribution of pixels in smooth regions, the weights are computed by a predefined function of the image gradient. Small weights are assigned to missed pixels' in the neighborhood of pixels with large local gradients, edges are preserved better. The weights of convolution mask changes adaptively with gradient of pixels in a neighborhood. Thus, the algorithm can estimate missed pixels while preserve sharp edges in image.

The gradient of the pixels in a small neighborhood is obtained and a function is defined by the equation

$$F(x_k) = 1 - \left(\frac{x}{\alpha}\right)^2 \quad \text{if } x \leq \alpha/2 \quad (1)$$

$$F(x_k) = \left(\left(\frac{x}{\alpha}\right) - 1\right)^2 \quad \text{if } \frac{\alpha}{2} \leq x \leq \alpha$$

$$F(x_k) = 0 \quad \text{if } x \geq \alpha$$

The weights of convolution mask are defined as

$$w(k) = \frac{1}{n} F(x_k) \quad (2)$$

The damaged pixel is estimated by using the following equation

$$f'(p) = (1 - \sum_{k=1}^n w(k))f(p) + \sum_{k=1}^n w(k)f(k) \quad (3)$$

where $f'(p)$ is estimated value, $f(k)$ is value of a known pixel in the current neighborhood, n is the number of known pixel in the current neighborhood. Where x is gradient value of the current pixel in the image, α is a parameter giving an estimation of the missed pixel gradient and it control the softness of propagation.

A convolution based technique using bilateral filtering is proposed by [4]. A bilateral filter is an edge-preserving smoothing filter. The bilateral filter replaces a pixel's value by a weighted average of its neighbors in both space and range. This is applied as two Gaussian filters at a localized pixel neighborhood, one in the spatial domain, and the other in the intensity domain. This approach permits preserves sharp edges. Every sample pixel is replaced by a weighted average of its neighbors.

The bilateral filters are efficient in denoising, and hence can be used to inpaint by estimating the lost (damaged) pixel value by its neighbors. Inpainting is proposed using a convolution with a bilateral averaging kernel. Image Inpainting is carried out by using a bilateral filter which preserves the edges and also provides smoothing. The bilateral filter takes a

weighted sum of the pixels in a local neighborhood; the weights depend on both the spatial distance and the intensity distance. Thus edges are preserved well while noise is averaged out. For smooth regions, pixel values in a small neighborhood are similar to each other, and the bilateral filter acts as a domain filter. When the bilateral filter is centered on a pixel on the bright side of the boundary, the kernel function assumes values close to one for pixels on the same side, and values close to zero for pixels on the dark side. For every pixel, the kernels are calculated using its neighbors in both space and range domains to implement bilateral filtering. The range coefficients are modified by using the gradient of the image to be inpainted rather than difference in pixel value.

Let Ω be the damaged area to be inpainted and $\partial\Omega$ be its one-pixel thick boundary. The algorithm consists of convolving the region to be inpainted with a bilateral kernel obtained by multiplying range and space kernels. Every sample is replaced by a weighted average of its neighbors. Let I and I_f represent the input and filtered signal respectively, p is current pixel (position), q is neighbor pixel, $w(p, q)$ is a weight devoted to pixel q , d_s is the Euclidian distance, σ_d and σ_r are two user defined numbers that denote geometric and photometric spread and (x_p, y_p) and (x_q, y_q) are coordinates of pixels p and q , respectively. Let p be the damaged pixel to be inpainted and q presents its undamaged neighbor pixel and P_{dir} is the spatial vector from q to p . The damaged pixel can be approximated by equation 3

$$I_f = \frac{\sum w(p,q)I(q)}{\sum w(p,q)} \quad (4)$$

$$w(p, q) = w_1(p, q)w_2(p, q) \quad (5)$$

$$w_1(p, q) = e^{-1/2\left(\frac{d_s}{\sigma_d}\right)^2} \quad (6)$$

$$w_2(p, q) = e^{-1/2\left(\frac{P_{dir} \cdot \nabla I}{\sigma_r}\right)^2} \quad (7)$$

$$\nabla I = [dI/dx \quad dI/dy]^T \quad (8)$$

$$P_{dir} = [x_p - x_q \quad y_p - y_q]^T \quad (9)$$

There are two parameters that control the behavior of the bilateral filter are σ_d (geometric spread) and σ_r (photometric spread) which characterize the spatial and intensity domain behaviors, respectively. Both σ_d and σ_r determine the level of smoothness. Assigning σ_r to zero reduces the bilateral filter to a simple Gaussian smoothing filter with a cut-off frequency proportional to $1/\sigma_d$. Hence smaller values of σ_d better preserve fine details and edges. The size of the window must be determined in such a manner that at least one known pixel exists in the window region at the first iteration. A small value of σ_d causes a small considered neighborhood and therefore

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

the inpainting process needs more iterations and only a small part of the damaged area will be reconstructed. A large value of σd will provide a blurring effect in both texture and structure regions. A midrate value of σd can inpaint the missed region adequately.

In [3] an iterative method of image inpainting based on median filters is proposed. Firstly the damaged region is identified. Then a median of known pixels in different directions is used to inpaint a damaged region. In [3] the inpainting algorithm is implemented by using median filters which reconstructs the image by filling in holes. The damaged region in an image is selected manually by the user. The proposed algorithm in [3] is as follows after determining damaged regions, the algorithm considers one pixel thick boundary of the missed region. For each missed pixel on the boundary, known pixels in different directions are selected. The median value in each direction is determined, and finally, the damaged pixel value is considered to be the median of these medians. Once, all of damaged boundary pixels are reconstructed, the algorithm has finished its first iteration. In next iteration, the new boundary is first calculated. Then in a similar manner, the boundary pixels are reconstructed

III IMPLEMENTATION

In this paper we have implemented a spatial domain inpainting algorithms. This paper compares the inpainted images obtained by Olivera's, Hadhoud algorithm, median filtering, and directional median filtering, convolution using Noori et al's algorithm and bilateral inpainting algorithm for various damaged input images. The algorithm for spatial domain inpainting technique is as stated below.

A. Olivera's and Hadhoud algorithm

Step1. The region of the colored image to be inpainted is identified.

Step2. The colored image is decomposed into the corresponding RGB components.

Step3. The region to be inpainted is extracted from each plane.

Step4. The identified region is convolved by using the mask function.

$$\begin{matrix} a & b & a \\ b & 0 & b \\ a & b & a \end{matrix}$$

Fig 4. The masks used with the Oliveira algorithm.

$$a = 0.073235, b = 0.176765, c = 0.125$$

$$\begin{matrix} a & b & a \\ b & a & b \\ a & b & 0 \end{matrix}$$

Fig 5. The masks used with the Hadhoud algorithm.

$$a = 0.073235, b = 0.176765, c = 0.125$$

Step5. The colored image is reconstructed by combining the pixels of the corresponding RGB components.

B. Bilateral Filtering Algorithm

Bilateral Filter can be used to estimate the lost (damaged) pixel value from its neighbors. This can be done by obtaining a bilateral averaging kernel for a neighborhood of size w . The space domain kernel is found by equation 3 and range domain kernel is found by equation 4. The algorithm is implemented by convolving the region to be inpainted with a bilateral kernel obtained by multiplying range and space kernels. A damaged pixel is replaced by a weighted average of its neighbors

Step1: Identify the regions to be inpainted.

Step2: Estimate w_1 by equation 6.

Step3: Estimate w_2 by equation 7.

Step4: Calculate w as product of w_1 and w_2 .

Step6: Estimate the value of damaged pixel using equation 1.

Step7: Repeat the above steps for fixed number of iterations.

C Directional median filtering and median filtering algorithm

The algorithm for directional median filtering and median filtering is as follows.

Step1. The region of the colored image to be inpainted is identified..

Step2. The colored image is decomposed into the corresponding RGB components

Step3. The region to be inpainted is extracted from each plane.

Step4 A $[m \times m]$ window is constructed across the missing pixel.

Step6. The centre value of this window is calculated to be median of the pixels in the window for median filtering. The centre value is calculated as the medians of the median in vertical horizontal and two diagonal directions for directional

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

median filtering algorithm. The window is slid across the entire damaged region. This process is repeated for all planes.

Step7. The colored image is reconstructed by combining the pixels of the corresponding RGB components

D. Noori et al's algorithm

Step1. The region of the colored image to be inpainted is identified.

Step2. The colored image is decomposed into the corresponding RGB components.

Step3. The gradient of the pixels in a small neighborhood is obtained

Step4. A function is defined by the equation 1.

Step5. The weights of convolution mask are calculated by using the equation 2.

Step6. The damaged pixel is estimated by using equation 3 in each plane.

Step7. The colored image is reconstructed by combining the pixels of the corresponding RGB components

IV RESULTS

The discussed algorithms are tested on a variety of images to investigate the performance of inpainting. The test set constitutes of Standard images used by image processing community. The original images are clean images, and these images are artificially degraded by drawing random lines on them. The random lines are treated as scratches or areas of inpainting. These degraded images are then inpainted. The comparison of algorithms is carried out by implementing in MATLAB 7.7.0.471 R2008b.. In our study, the quantitative valuation is performed by calculating peak signal to noise ratio (PSNR) and is defined as

$$PSNR = 20 \log \log_{10} 255 / RMSE(u, v)$$

$$RMSE = \sqrt{(u_{i,j} - v_{i,j})^2 / MN} \quad (10)$$

where u is original image, v is inpainted image and $M \times N$ is size of u and v . Output Images for spatial domain inpainting algorithms are as tabulated below. Table 1 and Table 2 show Time elapsed and PSNR for all Algorithms for an image with the same amount of damage

Table 1 Results for all algorithm's for image1.

Algorithm	Elapsed time	PSNR
Median Filtering iterations =5	2.17 sec	55.86
Bilateral Filtering iteration 10	44.14 sec	49.44
Noori's Algorithm iteration 20	18.14 sec	49.13
Olivera's Algorithm iteration 20	0.62 sec	49.65
Directional median filtering iteration 5	3.94 sec	56.76
Hadhoud's Algorithm.iteration=1	0.47 sec.	44.86

Table 2 Results for all algorithm's for image2.

Algorithm	Elapsed time	PSNR
Olivera's iterations=50	0.50 sec	55.82
Bilateral filtering iterations=50	151.12 sec	55.15
Hadhoud's Algorithm iteration=1	0.41 sec	54.07
Noori's Algorithm iterations=20	12.28sec	56.97
Directional median filtering iteration=5	2.20 sec.	58.29
Median filtering iteration 20	2.10 sec.	58.29

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)



Fig 6. Elapsed Time 62.36 seconds. PSNR 51.14 iterations=70;

Median filtering



Fig 7. Elapsed Time 85.45 seconds. PSNR 51.14 iterations=100

Directional median filtering Algorithm



Fig 8 Elapsed time is 629.12 seconds. PSNR 79.84 iterations=50 Bilateral filtering Algorithm



Fig 9 Elapsed time is 67.97 seconds. 46.36 iterations=20; Noori et al's Algorithm



Fig 10 Elapsed time is 1.27seconds. PSNR 44.41 Olivera Algorithm



INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Fig 11 Elapsed time is 0.73 seconds. 43.84 Hadhoud Algorithm

Figures 6 to figure 11 demonstrate the results of spatial domain inpainting Algorithms with the left side showing the damaged image and the right side showing the reconstructed image. From the reconstructed images it can be seen that the visual quality of the images is better.

V CONCLUSION

The results of Spatial Domain Inpainting Algorithms for Median filtering, Directional median filtering, Bilateral filtering, Noori et al's Algorithm, Olivera's Algorithm and Hadhoud Algorithm are presented in this paper. The performance can be compared on the basis of PSNR and time required. Hadhoud's Algorithm performs the best in a single iteration. Bilateral filtering takes more time for reconstruction and generates a moderate PSNR. In future the algorithms can be implemented in Frequency domain and the performances can be compared.

REFERENCES

- [1] Manuel M. Oliveira, Brian Bowen, Richard McKenna and Yu-Sung Chang, "Fast Digital Image Inpainting" Imaging and Image Processing (VIIP 2001), Marbella, Spain, September 3-5, 2001.
- [2] Mohiy M. Hadhoud, Kamel. A. Moustafa and Sameh. Z. Shenoda, "Digital Images Inpainting using Modified Convolution Based Method", International journal of signal processing, image processing and pattern recognition.

[3] H. Noori, Saeid Saryazdi, "Image Inpainting using Directional Median Filters", Shahid Bahonar University of Kerman. 2010 International conference on computational Intelligence and communication network.

[4] H. Noori, S. Saryazdi and H. Nezamabadi-Pour, "A Bilateral Image Inpainting", IJST, Transactions of Electrical Engineering, Vol. 35, No. E2, pp 95-108, 2011, printed in the Islamic Republic of Iran, Shiraz University.

[5] Noori H, Saryazdi S and Nezamabadi-pour H 2010 "A convolution based image inpainting", In the 1st International Conference on Communication and Engineering, University of Sistan & Baluchestan, December.

[6] RAJKUMARL BIRADAR, and VINAYADATTV KOHIR "A novel image inpainting technique based on median Diffusion".

[7] Marcelo Bertalmio, Guillermo Sapiro, Vicent Caselles and Coloma Ballester, "Image Inpainting", SIG- GRAPH 2000.

[8] A Bilateral Image Inpainting, H. Noori, S. Saryazdi and H. Nezamabadi-Pour, IJST, Transactions of Electrical Engineering, Vol. 35, No. E2, pp 95-108, 2011.

[8] Christine Guillemot and Olivier Olivier Le Meu Image "Inpainting Overview and recent advances", 1053-5888/14/\$31.00©2014IEEE IEEE SIGNAL PROCESSING MAGAZINE [127] JANUARY 2014.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)