ISSN (online): 2349-784X

Object Removal and Region Filling Based on Exemplar-based Method

Pritam N. Shet
UG Student
Department of Electronics and Telecommunication Engineering
Don Bosco College of Engineering, Fatorda-Margao, Goa

Ashpak Shaikh
UG Student
Department of Electronics and Telecommunication Engineering
Don Bosco College of Engineering, Fatorda-Margao, Goa

Vivek Tendulkar
UG Student
Department of Electronics and Telecommunication Engineering
Don Bosco College of Engineering, Fatorda-Margao, Goa

Vivek Singh
UG Student
Department of Electronics and Telecommunication Engineering
Don Bosco College of Engineering, Fatorda-Margao, Goa

Vyshakh. V
UG Student
Department of Electronics and Telecommunication Engineering
Don Bosco College of Engineering, Fatorda-Margao, Goa

Abstract

Image Inpainting is a technique used to restore the lost parts of an image and rectify any alterations in the original image in visually appealing manner. Inpainting technique focusses upon getting any modified or damaged image back to its original form. This technique has got a list of applications including removal of occlusions, such as stamps, subtitles, text on photographs, rebuilding of damaged photographs and films, removal of unwanted objects or blur and red eye correction. In this paper we provide a brief review of Exemplar based image Inpainting technique.
Keywords: Inpainting, Exemplar based method, Texture synthesis, Distance matrix, patch priorities

I. INTRODUCTION

Restoration of missing or damaged portions of an image is very important practice used extensively in artwork restoration. This very technique is known as inpainting or retouching. This practice consists of filling in the missing areas of any old picture degraded over ages or modifying the damaged pictures in non-detectable manner [1]. In addition to common inpainting applications like restoration of photographs films and paintings, removal of superimposed objects or characters such as text, subtitles, stamps and publicity from images, inpainting can also be used to produce special effects. Traditionally, skilled artists and studio professionals have to perform image inpainting manually using popular licensed tools such as Photoshop. Firstly, Bertalmio et al [1,2] had introduced algorithm for digital inpainting of still images that produced very impressive results exactly fitting into the definition of inpainting. This algorithm, however, required several minutes on windows personal computers for the inpainting of relatively small areas. From this point, extensive work on inpainting has been carried out across the globe and still counting on developing algorithms capable of producing similar results in just a few seconds. A very effective algorithm that gets rid of this unacceptable time consumption for inpainting of small areas is Exemplar based inpainting method. This method relies upon a simpler and faster algorithm and the results produced by this algorithm are comparable to those found in the literature, but two to three times faster in speed. We illustrate the effectiveness of our approach with examples of reconstructed photographs and vandalized images.

II. EXEMPLAR-BASED METHOD

Based on seminal work on texture synthesis [2,3], a family of inpainting methods has appeared in the last decade with the aim of better recovering the texture of the missing area rather than simply replacing a patch with any matching exemplar. The goal of texture synthesis also referred to as sample-based texture synthesis is to create a texture from a specified background texture sample in such a way that, the produced texture is with a similar visual appearance as that of the background. Exemplar-based inpainting employed for retouching a large part, is inspired by local region-growing methods that grow the texture pixel-by-pixel or patch-by-patch at a rate of one pixel or one patch at a time, while maintaining coherence with nearby pixels or patches. Local pixel-based texture synthesis techniques widely rely on Markov random fields (MRFs) modelling of textures [3,4]. But in our algorithm we propose to tackle this modelling technique which runs very complex graphical approaches to find information of missing pixels, by running algorithms that exploit both locality (the colour and data of a pixel being assumed to depend on its neighbouring pixels) and stationarity (the best replaceable local pixel is independent of the pixel location), the missing pixels are learned by sampling and copying the central pixel of a patch from the sample texture that best matches the known input pixel to be synthesized, according to the probabilistic confidence in using that pixel. Similarly, the output image is generated pixel-per-pixel in a raster scan order choosing a pixel from the sample image at each step whose neighbourhood is most similar to the currently available neighbourhood in the texture being synthesized. Texture synthesis methods have evolved from pixel-based to patch-based techniques, with recent enhancements relying on elaborated blending and quilting strategies.

A. Pixel Based Texture Synthesis Technique

The simple pixel-based texture synthesis technique proceeds as follows:

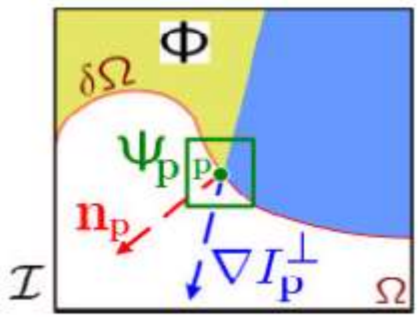

Fig. 1: Notation diagram. Given the patch Ψ_p, n_p is the normal to the contour δΩ of the target region Ω and $\nabla I_p^{\perp}$ is the isophote (direction and intensity) at point P. The entire image is denoted with I.

Let Px be a pixel located at position x in the image I, and Px W be the patch centred on the pixel Px. This patch has a known part () p S xW and an unknown part () p U xW. \\The idea is to search for the patch pj W (centred on) pj the most similar to the input patch Px W. The central pixel pj having a neighborhood most similar to the known neighborhood of Px is then copied to recover Px. Image information is therefore pixel-per-pixel propagated from the known part to the unknown part of the image. This pixel-per-pixel recovery algorithm suffers from a high computational cost, even if its complexity can be reduced by constraining the search for best matching patches among the candidates of the neighboring pixels that have been already inpainted. Another limitation is the principle of exemplar-based methods: search for the patch the most similar to the known part of the input patch to be completed and copy the central pixel for (a) pixel-based approaches or (b) a set of pixels for patch-based approaches.

The texture synthesis problems: difficult for this type of approach to synthesize textures that are not frontal (with some perspective transformations) and to fill in large and dispersed holes. Moreover, although performing better than diffusion methods on textured areas, the pixel-based synthesis techniques often suffer from synthesis errors propagation and from repetitive patterns, which look unnatural especially in the case of stochastic textures. They also run into difficulties when synthesizing texture formed by an arrangement of small objects. Approaches synthesizing entire patches rather than only one pixel at a time have then emerged to cope with the drawbacks just mentioned. Instead of synthesizing the missing region pixel per pixel, the idea of patch-based solutions is to recover entire patches in one step by sampling and copying texture patterns (entire patches) from the source.

The first step for estimating the pixels in the unknown part p U xW of the patch again consists in searching for the patch pj W (centred on), pj which is the most similar to the patch. p S xW but this time, all the pixels from pj W which are located at the same position as p U xW are copied to estimate the unknown pixels of the input patch. The terminology exemplar-based inpainting now mostly refers to the patch based filling methods that synthesize entire patches by learning from patches in the known part of the image. Since they synthesize entire patches at once, these methods are faster than pixel based approaches.

B. Variants of Exemplar-based Inpainting

1) Distance Matrix

Several matrices exist for measuring similarity between image pixels or image patches. One of the most widely used matrices and the one used in our algorithm is: cross-correlation matrix that plots pixel colour values as matrix elements. The sum of squared differences (SSD) favours the copy of pixels from uniform regions. Where the SSD is multiplied by-(1+ dBC). When two patches have the same distribution, the distance value is equal to the SSD between the two patches, this is the major limitation found in SSD technique [5].

For efficient search of best matching patches in Exemplar-based inpainting method, following stepwise approach is used. First, search for K-nearest neighbours (K-N×Ns) within the known part of the image. Best solution to the N×N search is to compute the distance from the query patch to all possible ‘candidate patches’, repeating it for each replaceable patch in whole image. A comparatively faster and approximate N×N search method [6] could be developed which organize the candidates in specific space-partitioning data structures, such as a tree structure. A tree structure helps to quicken the search by eliminating large portion of search space and checking only a small portion of candidates. The kd-tree-based matching is one of the most widely used algorithms [7] for finding the nearest patch. However, its number of searched nodes increases exponentially with the space dimension. When the dimension is large (e.g., higher than 15), its search speed becomes very slow. The second step is to decide patch processing order. The missing regions in an image are generally composed of textures and structures. It is very important to separate these two components and start by first recovering the structures. This led to proposing patch processing orders that are defined to work on the patches on structures first.

In general, the processing order is given by a patch priority (() Pp x) measure defined as the product of two terms: the confidence term (() Cp x) and the data term (() Dp x). i.e. (() Pp x = () Cp x *() Dp x). The first term accounts for the amount of known pixels versus unknowns in the input patch called as confidence term, whereas the second term () Dp x, called data term, specifies presence of some structure in the patch. A gradient-based data term [8] favours patches in which the isophote is perpendicular to the front line of pixel Px. The data term is defined as the absolute value of the inner product between the isophote direction (perpendicular to the gradient) and the normal to the front line. While reconstructing any missing patch, unknown pixels at the edge of an object are given higher priority than pixels located on flat image areas due to above mentioned confidence measurements.

C. *The Experiment Results and Effect Analysis*

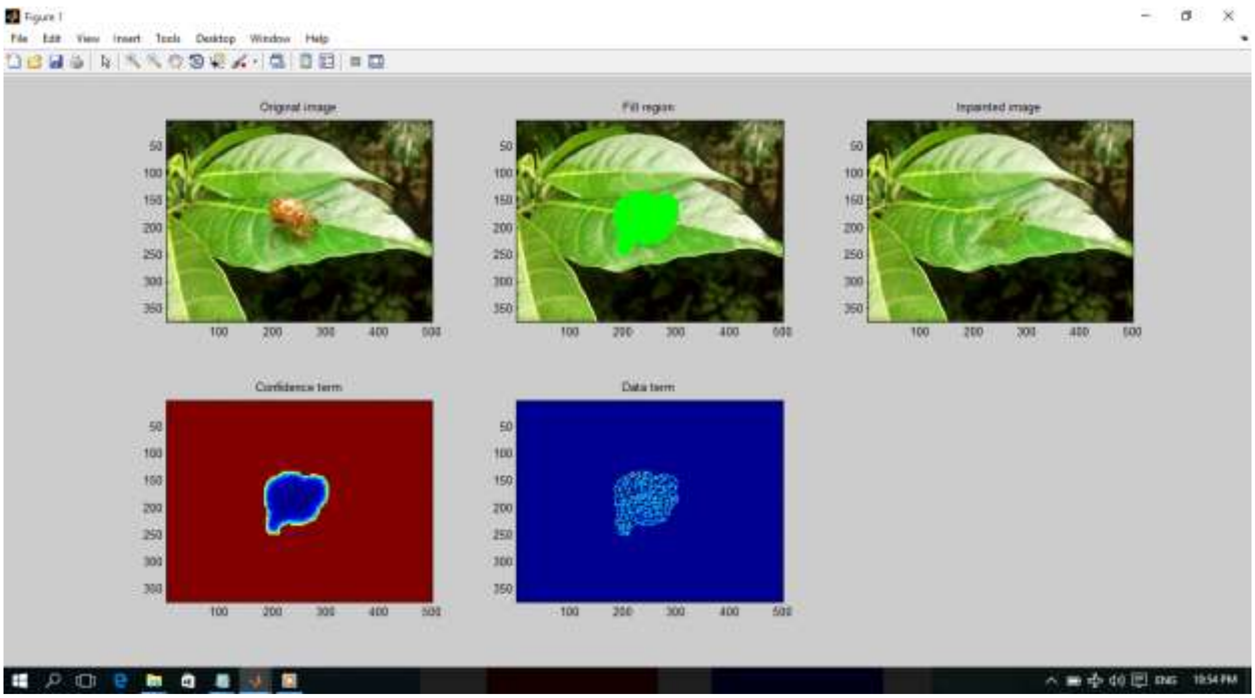

Fig. 1: A) Original Image B) Patched Image C) In-painted Image D) Confidence Term E) Data Term

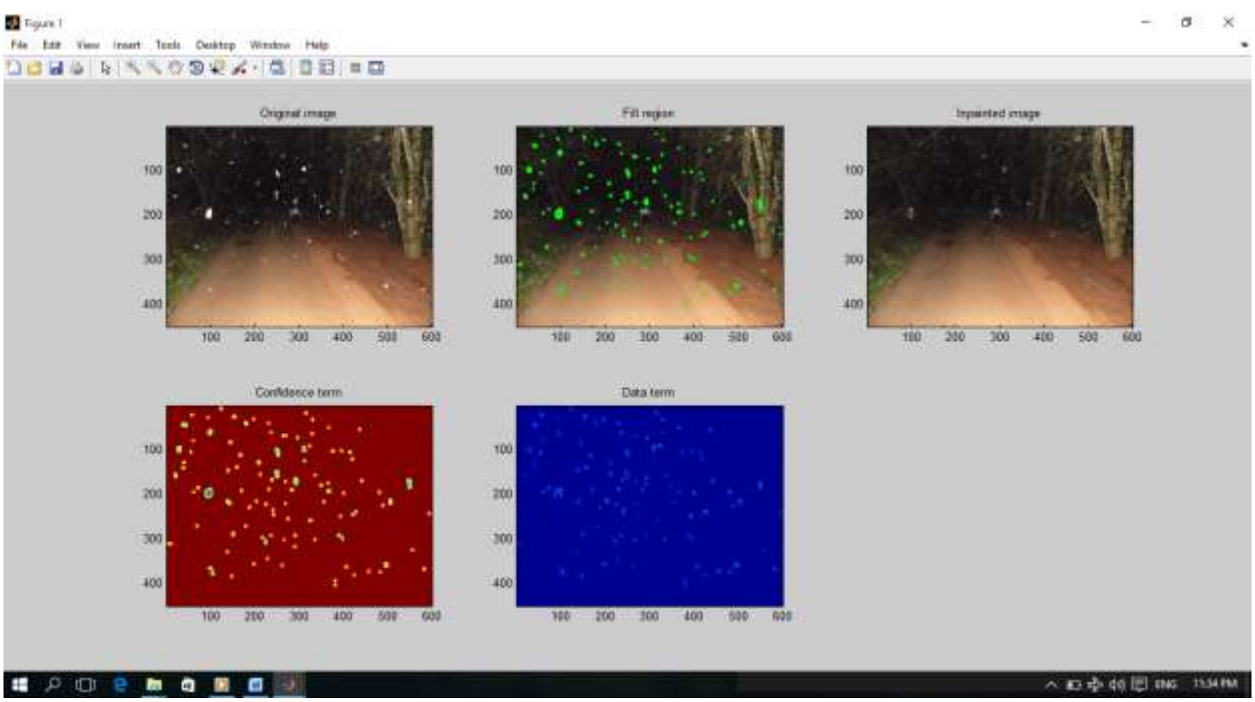

Fig. 2: A) Original Image B) Patched Image C) In-painted Image D) Confidence Term E) Data Term

D. To Find Boundary of Target Region, We Can Convolve The Fillregion Matrix With Laplacian Filter

Table- 1 Laplacian Filter

1	1	1
1	-8	1
1	1	1

E. Working with Patches

1) As with all other exemplar based algorithms, this algorithm replaces the target region patch by patch.
2) This patch is generally called the template window, Ψ.
3) The size of Ψ must be defined for the algorithm.

F. Finding Patch Priorities

1) Thus, for a patch centered at a point p for some p£δΩ , its priority can be defined as the product of two terms

$$\rho(\rho) = c(\rho) * d(\rho) \quad (1)$$

2) Where c (p) represents the confidence term for the patch and D (p) the data term for the patch.

$$C\ (p) = \frac{\sum_{q \in \varphi\rho\emptyset} c(q)}{|\varphi_p|} \quad (2)$$

$$D\ (p) = \frac{|\nabla I_p^{\perp}.n_p|}{\alpha} \quad (3)$$

III. Applications

The need of inpainting is of utmost importance in various image processing applications requiring image restoration, editing (e.g., object removal), and image-based rendering processes like texture synthesis, image resizing (e.g., enlargement), interpolation, loss concealment, etc.

A. Other Applications Include

1) Repairing Photographs: with age, photographs often get damaged or scratched. These deteriorations could be reverted using inpainting.
2) Remove unwanted objects: using inpainting, we can remove unwanted objects, text, etc. from the image.
3) Special Effects: Inpainting may be used in producing special effect or in production of animation based films (Video Effects).

IV. Conclusion and Future Work

With regards to fast growing field of software technology, many image editing tools may be developed in near time. Henceforth new Inpainting techniques would also evolve. Numerous and different types of inpainting approaches have been proposed with varying applicability in past few years. Here we have presented an image inpainting algorithm based on isotropic diffusion model with explicit function of Background texture synthesis. The results produced by this simple algorithm are, in many cases, comparable to previously known non-linear inpainting algorithms, but two to three orders of magnitude faster, thus making inpainting practical for interactive applications. Also the faster inpainting techniques motivate us to think upon methods that can stabilise a video (by parsing it in to number of picture frames) and then in paint them as a whole.

Also in this algorithm, the sensitivity of Image restored could be measured as a function of finest details (∂Ω) that are restored. Ideally, the mask Ω should include exactly the region to be retouched. If any smaller area inside this mask, ∂Ω, contains spurious information, like text etc., this information has to be reconstructed properly. If some, possibly important information is discarded then the sensitivity of the restored image is decreased. Being able to create and refine **Ω** interactively can greatly improve the quality of the reconstruction. The presented algorithm is intended for filling in locally small areas. For larger inpainting domains, a scale-space approach can be used to preserve the algorithm's speed at the expense of reconstruction quality.

References

[1] Bertalmio, M, Sapiro, G., Caselles, V., Ballester, C. Image Inpainting. SIGGRAPH 2000, pages 417-424.
[2] M. Bertalmio, L. Vese, G. Sapiro and S. Osher, "Simultaneous structure and Texture image inpainting," IEEE Transaction Image Processing, vol. 12, no. 8, pp. 882–889, Aug. 2003
[3] A. Efrosand T. Leung, "Texture synthesis by non-parametric sampling," in Proc. Int. Conf. Computer Vision (ICCV), Sept. 1999, pp. 1033–1038.
[4] L. Weiand M. Levoy, "Fast texture synthesis using tree-structured vector quantization," in Proc. ACM SIGGRAPH, July 2000, pp. 479–488.
[5] A. Bugeau, M. Bertalmio, V. Caselles, and G. Sapiro, "A comprehensive framework for image inpainting," IEEE Trans. Image Processing, vol. 19, no. 10, pp. 2634–2645, Oct. 2010.

[6] N. Kumar, L. Zhang, and S. K. Nayar, “What is a good nearest neighbour’s algorithm for finding similar patches in images” in Proc. European Conf. Computer Vision (ECCV), Oct. 2008, pp. 364–378.
[7] J. Bentley, “Multidimensional binary search trees used for associative searching,” Commun. ACM, vol. 18, no. 9, pp. 509–517, Sept. 1975.
[8] A. Criminisi, P. Pérez, and K. Toyama, “Region filling and object removal by examplar-based inpainting,” IEEE Trans. Image Processing, vol. 13, no. 9, pp. 1200–1212, Sept. 2004.