

Search Window Based Exemplar Approach for Image Inpainting

Mayur Rank, N.M. Patel

Abstract: Image inpainting is a technique to fill missing region or reconstruct damage area from an image. It removes an undesirable object from an image in visually plausible way. For filling the part of image, it uses information from the neighboring area. In this dissertation work, we present an Exemplar based method for filling in the missing information in an image, which takes structure synthesis and texture synthesis together. Image inpainting is done in such a manner, that it fills the damaged region or holes in an image, with surrounding color and texture. The algorithm is based on patch based filling procedure. First find target region using mask image and then find boundary of target region. For all the boundary points it defined patch and find the priority of these patches. It starts filling the target region from the highest priority patch by finding the best match patch from the search window. This procedure is repeated until entire target region is inpainted. We have perform experiment on many images and found that our algorithm successfully inpaint the target region.

Index Terms: Texture Synthesis, Structure Synthesis, Patch Propagation

I. INTRODUCTION

In real world, many people need a system to recover the damaged photographs, artwork, designs, drawings etc. Damage may be due to various reasons like scratches, overlaid text or graphics, scaled image etc. Nowadays, powerful photo-editing tools are available for retouching, drawing, and removing object by scissors from images. But, to fill the missing information or reconstruct damage area in an image is still difficult task.

This system could enhance and return a good looking photograph using a technique called image inpainting. Image inpainting modify and fill the missing area in an image in an undetectable way, by an observer not familiar with the original image [1]. The technique can be used to reconstruct image damage due to scratches, to remove dates and titles etc. from image.

This method starts with original image and mask image as input. Here mask image specify the object to be removed from the original image. The object to be removed has to be marked by user because it depends on the subjective choice of user. And give the output as reconstructed image. Image inpainting is different from other general image enhancement algorithms in the sense that image enhancement assumes that pixel in the damaged portion of image, contain both the information about real data and the noise, while in image

inpainting, the pixel values are all assumed to be missing in the filling domain.

The concept of image inpainting was first introduced by Bertamio et al. [1][2]. The method was inspired by the real inpainting process of artists. The image smoothness information interpolated by the image Laplacian is propagated along the isophotes directions, which are estimated by the gradient of image rotated by 90 degrees. The work in [3] introduced another method using the framework of the Navier-Stokes equation it give good result. The approach uses ideas from fluid dynamics to propagate isophote lines from the exterior into the region to be inpainted. Chan and Shen [4] introduced the space of Bounded Variation images to an inpainting system, which recovers missing information with total variation.

The major problem with above methods is that they are unable to reconstruct texture regions. These methods have several disadvantages. The main disadvantage is that these approaches may lead to blurred filling in texture missing regions and large missing regions. Consequently, inpainting approaches based on texture synthesis are proposed to avoid this problem.

The work in [5] decomposes the original image into two components, one of which is processed by inpainting and other by texture synthesis. The output image is sum of two processed components. This approach still remains limited to the removal of small gaps; however, as the diffusion continues to blur the field region

Exemplar Based method proposed by Criminisi et al. [6] used a best exemplar patch to propagate target patch including missing pixels. This technique uses an approach which combine structure propagation with texture synthesis and hence produced very good results. Wong and Orchard [7] combined plural non-local exemplar patches to propagate.

In this paper we present exemplar based approach using search window. In exemplar based approach best match patch is finding from full image, but through experiment we implement approach in which best match patch is find only in search space window which reduced time complexity. Paper is organized as follows. Section1 is introduction part Section2 discusses the exemplar based approach using search window. Some result getting using this approach shown in Section3. And finally Section4 present conclusion and indicate future work.

II. EXEMPLAR BASED APPROACH USING SEARCH WINDOW

Exemplar based approaches perform well for two dimensional texture as well as with liner image structure.

Manuscript received on July, 2012

Mayur Rank, ¹Department of Computer Engineering, Birla Vishvakarma Mahavidhyalaya, Gujarat, INDIA.

Prof. N.M. Patel, Associate Professor of Computer Engineering Department, BVM Engineering College, Vallabh Vidyanagar, Gujarat, INDIA.

First, given an input image, the user selects the object to be removed. This step requires user interaction because object to be removed depends on the subjective choice of the user. The part of the image from where the object is to be removed is known as target region or inpainting domain Ω . The source region is entire image minus the target region. The size of the template window must be specified. This can be 9 x 9 pixels, but in practices required the user to set it to be slightly larger than the largest distinguishable texture element, or “texel”, in the source region [6]. In addition to this, user also needs to specify the size of the search window. Use of search window improves execution time because it reduces the searching time for finding a best match patch later in the algorithm. The size of search window depends on the region to be filled and the kind of structure and texture in image.

Once these parameters are specifying, the region- filling proceeds automatically. During the algorithm, patches along the fill-front are assigned a temporary priority value, which determines order in which they are filled. Then the algorithm iterates following three steps until all pixels have been filled.

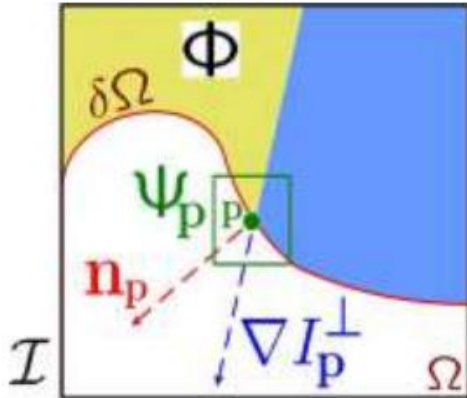


Figure 1 Notation diagram

Above figure shown the missing regions i.e. target region or inpainting region is denoted by Ω and its boundary $\delta\Omega$ also specify and the source region is denoted by Φ . Ψ_p is the best edge patch find from boundary of target region. Approach is divided in three parts.

A. Find Best Edge Patch

In the first step, a best edge patch Ψ_p is picked out using priority [6]. This algorithm uses best- first filling strategy that entirely depends on the priority values which are assigned to each patch on the fill-front. The priority computation is biased toward those patches which (1) are on the continuation of strong edges and (2) are surrounded by high-confidence pixels.

Given a patch Ψ_p centred at the point P for some $P \in \delta\Omega$ is shown in figure 1 Priority $P(p)$ is defined as the product of two terms

$$P(p) = C(p) D(p)$$

Here, $C(p)$ is the confidence term and $D(p)$ is the data term. They are defined as follows

$$C(p) = \frac{\sum_{q \in \Psi_p \cap (\mathcal{I} - \Omega)} C(q)}{|\Psi_p|}$$

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

Where $|\Psi_p|$ is the area of Ψ_p , α is a normalization factor (e.g., $\alpha=255$ for a typical grey-level image), n_p is a unit vector

orthogonal to the fill-front $\delta\Omega$ in the point P , and \perp denotes the orthogonal operator. The priority is computed for every border patch, with distinct patches for each pixel on the boundary of the target region. During initialization, the function $C(p)$ is set to $C(p) = 0, \forall p \in \Omega$ and $C(p) = 1, \forall p \in \mathcal{I} - \Omega$.

The confidence term $C(p)$ may be thought of as a measure of the amount of reliable information surrounding the pixel P . The idea is to fill first those patches which have more of their pixels already filled. This automatically incorporates preference toward certain shapes of the fill-front. For example, patches that include corners and thin tendrils of the target region will tend to be filled first, as they are surrounded by more pixels from the original image.

The data term boost the priority of the patch in which a liner structure flows into. This term is very important because it allows broken lines to correct.

B. Find Best Match Patch

Once priority is finding for all patches on boundary then take patch Ψ_p which has highest priority. Then find the patch Ψ_q in search space window which is most similar to patch Ψ_p shown in figure 2. The most similar patch Ψ_q is the one which has the minimum difference in the pixel value with patch Ψ_p .

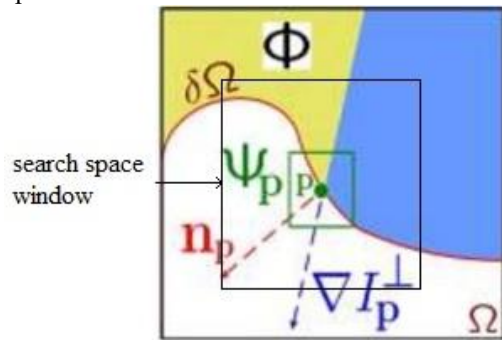


Figure 2 Search space window

Difference between any two pixels p and q given by using sum of squared difference (SSD) method. It define as

$$SSD(\Psi_p, \Psi_q) = \sum_{i=1}^M \mu_i (\Psi_p(i) - \Psi_q(i))^2$$

Where, $\Psi_p(i)$ and $\Psi_q(i)$ are the i -th pixel value in respective patches. M is the size of the patch. μ_i is pixel mask function. An exemplar patch Ψ_q is a patch with the lowest SSD value. Which is define as

$$\Psi_q = \min_{\Psi_q \in \Phi} SSD(\Psi_p, \Psi_q)$$

Above equation give the patch which is most similar to the patch Ψ_p in the image which has minimum SSD value. Here SSD method takes color value of two pixels for difference.

C. Copying Best Match Patch and Updating Confidence Values

Once the patch Ψ_p is filled with new pixel value Ψ_q , confidence value in the area is updated as follows.

$$C(q) = C(p) \text{ for all } q \text{ belonging to } \Psi_p \cap \Omega.$$

This simple update rule allows us to measure the relative



confidence of patch on the fill front. After completion of these three steps then update boundary with updated target region and repeat these three steps until all the pixels in the target region not fill.

Algorithm:

Steps of algorithm are given below:

- Step1.** Initialize mark variable for all pixels. If pixel belongs to inpainting region set mark variable with 0 else set 1.
- Step2.** Find boundary of region to be inpaint, if boundary is “empty set” than exit.
- Step3.** Find priority for all patches on the boundary.
- Step4.** Select the patch which has highest priority, call that patch, P.
- Step5.** Find the patch from search window which is best match to patch P, call that patch, Q.
- Step6.** Copy pixels of patch Q to the patch P, update only those pixels of patch P which has mark value 0 and set mark variable to 1, go to step2.

III. RESULT AND ANALYSIS

A. Time comparison of results with and without Using search window

Some resulted images of both approach with or without used of search window is given below and also give the execution time of both approach.

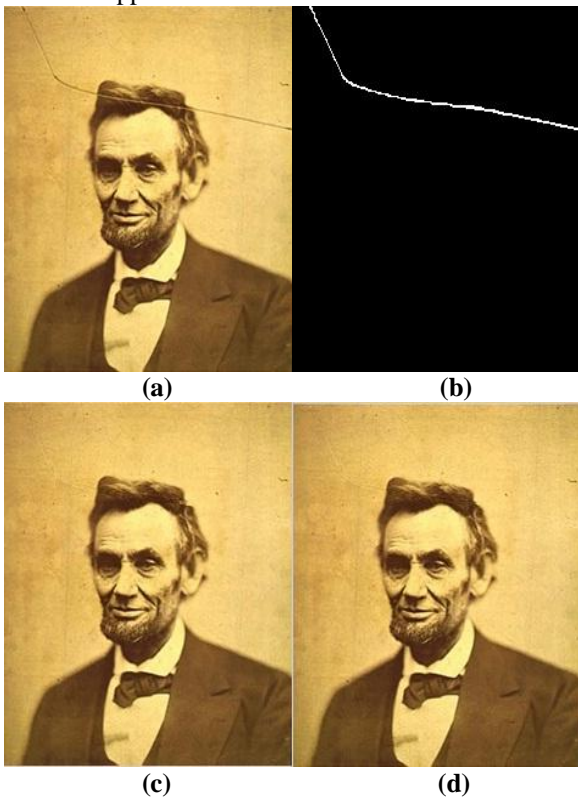


Figure 3 example of scratch removal from an image

Figure 3 (a) Show the image damage due to the scratch. (b) Show the mask specifying scratch from original image to be removed. (c) Show the inpainted result using search window with 81 x 81 and patch size 4x4. (d) Show the inpainted result without using search window and patch size 4x4.

Table 1: Show time required to execution.

	Using search window	Without using search window
Time(sec)	9.050320 s	51.294651 s

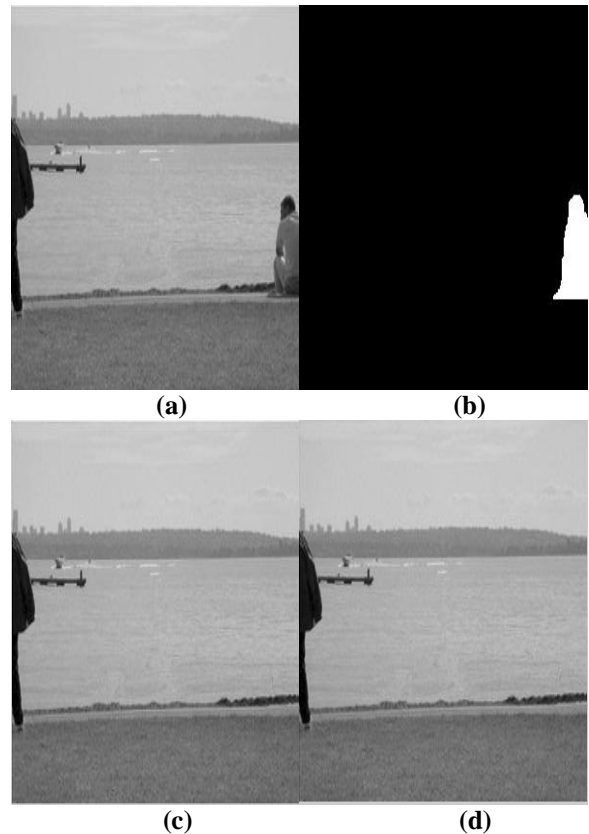


Figure 4 example of object removal from an image.

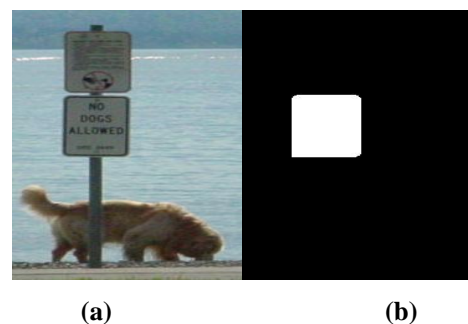
Figure 4(a) show the original image with object to be removed. Figure 4(b) is the mask specifying object from original image is to be removed. Figure 4(c) is show the inpainted result using search window with 81 x 81 and patch size 21x21. Figure 4(d) is show the inpainted result without using search window here patch size 21x21.

Table 2: Show time required to execution.

	Using search window	Without using search window
Time(sec)	10.552858 s	51.947192 s

Figure 3 and 4 show the inpainted result with both the approaches and table 1 and 2 show time comparisons of that result, table show that result with exemplar based approach using search window required less time then result with exemplar based approach without using search space window.

B. Results with varying patch size





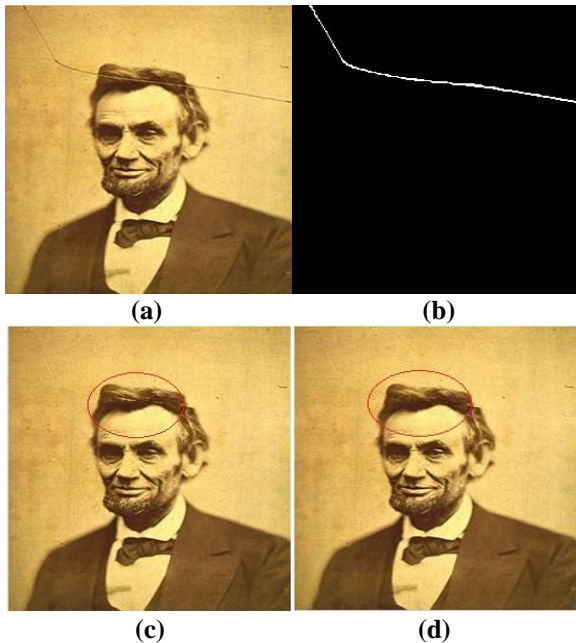
(c) (d)

Figure 5 is example of object removal from an image.

Figure 5(a) show the original image with object to be removed. Figure 5(b) is the mask specifying object from original image is to be removed. 5(c) is show the inpainted result with search window size 101 x 101 and patch size 9x9. Figure 5(d) is show the inpainted result without using search window here patch size 15x15.

In figure 5(c) and figure 5(d) red mark show result differences with different patch size. Here we conclude that result is also depend on the patch size with varying patch size result is also vary.

C. Results with varying search window size



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

Figure 6 show the image damage due to the scratch.

Figure 6(a) Show the image damage due to the scratch. Figure 6(b) is the mask specifying scratch from original image is to be removed. Figure 6(c) is show the inpainted result with search window size 31 x 31 and patch size 9x9. Figure 6(d) is show the inpainted result with search window size 81 x 81, patch size 9x9. In figure 6(c) and figure 6(d) red mark show result differences with different search window size.

IV. CONCLUSION AND FUTURE WORK

Image inpainting is a technique to fill missing region or reconstruct damage area from an image. In this dissertation, we have implemented exemplar based approach for image inpainting. This technique considers structure propagation and texture synthesise together which reduced blur in inpainted image. It takes patches window from damage

region for inpainting. In exemplar based approach to find best patch it search entire image. In this approach we have searched only in the predefined search window which reduced time complexity without effecting quality of the restored image. Using above experiment we have concluded that output image quality is depend on patch size as well as search window size.

Digital inpainting algorithm aims to automate the process of inpainting, and therefore also need to minimizing the user interaction. We will try to make algorithm which detect inpainted region without user interaction.

ACKNOWLEDGMENT

It is honour and pleasure to express my heartfelt gratitude to those who helped me and also contributed towards the preparation on this research. I am indebted to my guide Prof.N.M.Patel whose invaluable guidance and timely suggestion and constructive encouragement inspired me to complete the dissertation work in the present form. I would like to thank to the entire team of M.E. staff whose direct and indirect suggestion helped me creating this work. I would like to pay a special thanks to my parents for the sparing their invaluable time and inspiring me. Although there remain some names but none are remain un-thanked.

REFERENCES

1. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. (2000) "Image inpainting." in Proc. SIGGRAPH, pp. 417-424.
2. C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, and G. Sapiro, (June 2001)"A variational model for filling-in gray level and color images," in Proc. Int. Conf. Computer Vision, Vancouver, BC, Canada, pp. 10-16.
3. M. Bertalmio, A. L. Bertozzi, and G. Sapiro. (2001) "Navier-Stokes, fluid dynamics, and image and video inpainting," in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, pp. 417-424
4. T. Chan and J. Shen. (2002.) "Mathematical models for local non-texture inpaintings." SIAM J. on Appl. Math, vol. 62, pp. 1019-1043.
5. M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. (2003) "Simultaneous structure and texture image inpainting," in Proc. Conf. Comp. Vision Pat-tern Rec., Madison, WI.
6. A. Criminisi, P. Perez, and K. Toyama. (2004) "Region filling and object removal by exemplar-based image inpainting." IEEE Trans. Image Process, vol. 13, pp. 1200-1212.
7. A. Wong and J. Orchard, (2008) "A nonlocal-means approach to exemplar-based inpainting." in Proc. the IEEE Int. Conf. Image Processing.