Scientific Journal of Impact Factor (SJIF): 5.71
e-ISSN : 2348-4470
p-ISSN : 2348-6406

International Journal of Advance Engineering and Research Development

Volume 5, Issue 06, June -2018

PATCH BASED IMAGE INPAINTING

[1]Amol Laxman Raskar, [2]Prof. S.R. Gulhane

[1]*Department of electronics & Telecommunication, D. Y. Patil College of Engineering, Talegaon-Ambi*
[2]*Assistant Professor, Department of electronics & Telecommunication, D. Y. Patil College of Engineering, Talegaon-Ambi.*

Abstract —*In-painting of image photograph is the art of refilling in-missing photo picture data in an image. The main objective is to restructure missing regions so that, it is apparent reasonable to the human eye. There have been several approaches proposed for the same. This paper presents an algorithm that improves earlier proposed algorithms. Using proposed algorithm, one can not only in-paint large regions but also recover small portions (e.g. repair a photograph by removing scratch). The basic idea behind exemplar based in-painting approach is to find examples from the photo and replace it with the lost image data. Technique can be used in restoring old photographs or damaged film. It can also used for removing superimposed text like dates, titles or even entire objects from the image.*

Keywords- *Inpainting, exemplar, object removal*

I. INTRODUCTION

"Inpainting" is an art world's tern borrowed from restoration artists. This activity consists of filling in the missing areas or modifying the damaged ones in a non-detectable way by human eyes. Image inpainting is originally an artistic procedure to recover a damaged painting or picture. It has been introduced in and received attention from many researchers in computer vision and photo-image processing. Image-based algorithms have a variety of applications in computer vision, graphics and image recovery. The filling-in of missing or damaged area information is a very important topic in image processing. The object of inpainting is to reconstitute the damaged or missing portions of the work, to make it more legible. The goal and application of inpainting is to restore the damaged portion of paintings and photographs. This practice is called retouching or inpainting. An algorithm for the simultaneous filling-in of texture and structure in regions of missing information is presented in this paper. This paper also presents an extremely simple algorithm to address the texture synthesis problem. The main idea is to synthesize new textures by sampling patches of existing texture and pasting them together with the minimum boundary matching error. The purposes remain the same: to revert deterioration (e.g., cracks in photographs or scratches and dust spots in film), or to add or remove elements (e.g., removal of stamped date and red-eye from photographs, the infamous "airbrushing" of political enemies.

II. OBSERVATIONS MADE BY CRIMINISI

The two most important observations made by Criminisi are:

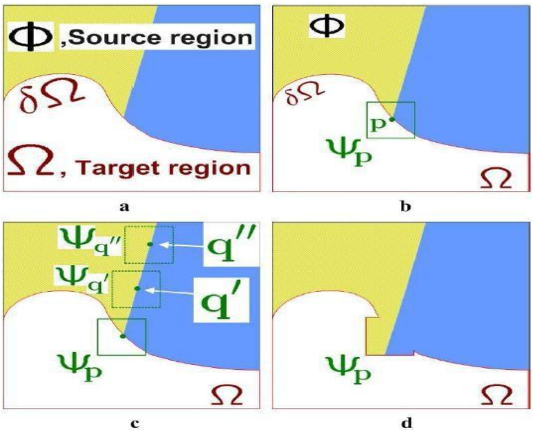
- Exemplar based Synthesis suffices
- Filling order is critical

A. Exemplar Based Synthesis Suffices

The main idea of this algorithm is an isophote-driven image sampling process. Exemplar based approaches perform well for two dimensional textures. But, in addition to that, exemplar-based texture synthesis is sufficient for propagating extended linear image structures, called as isophotes. Criminisi had important point to that, a separate synthesis mechanism is not required for handling isophotes

Fig1 illustrates the point. The region to be filled, target region, is indicated by Ω and its contour is indicated by δΩ the contour evolves inwards as the algorithm progresses. Hence it is referred as the "fill-front". The source region Φ. which remains fixed throughout the complete algorithm provides samples used in the filling process. One iteration of the algorithm to show how structure and texture are adequately handled by exemplar based synthesis is stated here. Suppose that the square template $\Psi p \in \Phi$ centred at point P is to be filled (fig.1b). The best-match sample from source region comes from the patch $\Psi q \in \Phi$, which is most similar to those parts that are already filled in Ψp lies on the continuation of an image edge; the most likely best matches will lie along the same edge.

It is required to propagate the isophote inwards is a simple transfer of the pattern from the best-match source path. Isophote orientation is automatically preserved. In the figure, despite the fact that the original edge is not orthogonal to the target contour δΩ the propagated structure has maintained the same orientation as the source region. So we focus on patch based work as opposed to pixel-based filling



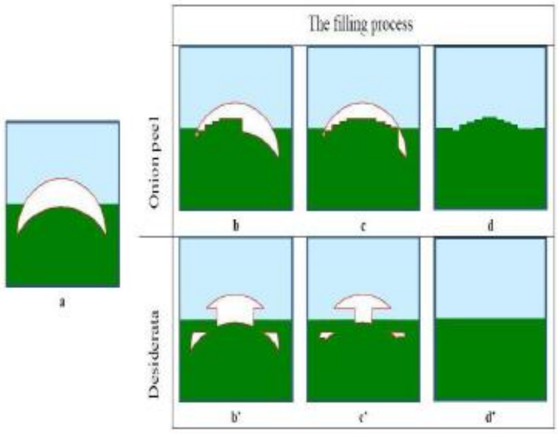


Fig1.Structure propagation by Exemplar based Synthesis ***Fig2.Filling order importance with concave target regions***

B. Filling Order is Critical

This section shows quality of the output image synthesis is highly influenced by the order in which the filling process proceeds. A comparison between the standard concentric layer filling (onion-peel) and the desired filling behavior is illustrated in fig2. Fig2 b, c, d show progressive filling of a concave target region via an anti-clockwise onion peel strategy. This ordering of the filled patches produces the horizontal boundary between the background image regions to be unexpectedly reconstructed as a curve. A better filling algorithm would be one that gives higher priority of synthesis to those regions of the target area which lie on the continuation of image structures, as shown in the figures 2 b',c',d'. Another important property of a good filling algorithm is that of avoiding "over-shooting" artifacts that occur when image edges are allowed to grow indefinitely.

III. PROPOSED ALGORITHM

In given input image the user selects a target region Ω manually to be removed and filled. The source region, Φ may be defined as the total image minus the target region Φ= I- Ω. Next the size of window Ψ must be specified. Criminisi has stated it to be 9X9. After these parameters are defined, the region filling algorithm proceeds automatically. In this algorithm, each pixel maintains a colour value and a confidence value, which reflects confidence in the pixel value, and which is frozen once a pixel has been filled. During the course of algorithm, patches along the fill front are also given a temporary priority value, which determines the order in which they are filled. Then this algorithm iterates the following three steps until all pixels have been filled

- Computing patch priorities.
- Propagating structure and texture information.
- Updating confidence values

A. Computing Patch Priorities

The algorithm performs the synthesis task through a best-first filling strategy that depends entirely on priority values that are assigned to each patch on the fill front. The priority computation is biased toward those patches which:

- Are on the computation of strong edges
- Are surrounded by high-confidence pixels

Given a patch Ψp centered at the point p for some p ϵ δΩ its priority is defined. Priority P(p) is product of two terms:

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

C(p)is the confidence term and D(p) is the data term.
These are defined as follows:

$$C(P)=\left(\sum_{q\in\Psi_p\cap(I-\Omega)} C(q)\right) \div |\Psi p|$$

$$D(p)=|\Delta Ip^{\perp}.\ np|$$

| Ψp | is area of Ψp, α is normalization factor (e.g. α =255 for typical grey level image), np is unit vector orthogonal to front δΩ in point p. The priority P(p) is computed for each border patch with distinct paths for each pixel on boundary of target region.During initialization C(p) is set to C(p)=0 for all p ϵ Ω and C(p)=1 for all p ϵ I - Ω. The confidence term C(p) may be thought of as a measure of the amount of reliable information surrounding the pixel p. the intention is to fill first those patches with more of their pixels already filled, with additional preference given to pixels that were filled early

on.As it is illustrated in fig 3a, this automatically incorporates preference towards 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 original image. These patches provide more reliable information against which to match. Conversely, patches at the tip of “peninsulas” of filled pixels jutting into the target region will tend to be set aside until more of the surrounding pixels are filled in.

At a coarse level, the term C(p) approximately enforces the desirable concentric fill order. As filling proceeds, pixels in the outer layers of the target region will tend to be characterized by greater confidence values, and therefore be filled earlier; pixels in the centre of the target region will have lesser confidence values.

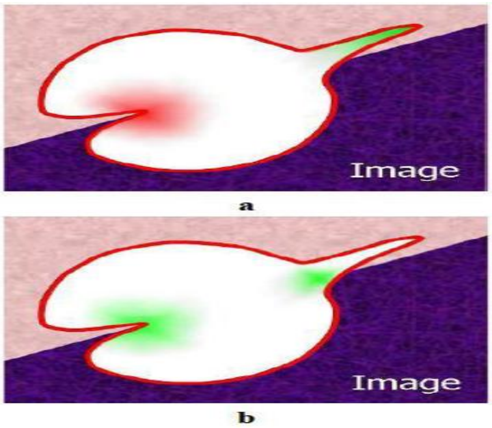


Fig3. Effects of C(p) and D(p)

The data term D(p) is a function of the strength of isophotes hitting the front $\delta\Omega$ at each iteration. This term boosts the priority of the patch that an isophote flows into. This factor is of fundamental importance because it encourages linear structures to be synthesized first, and, then propagated securely into the target region.

B. Propagating Structure and Structure Information

After all priorities in the fill front have been computed , the patch $\Psi\hat{p}$ with highest priority is found . We then fill it with data extracted from source region Φ. Image texture is propagated by direct sampling of source region . Patch which is very similar to $\Psi\hat{p}$ is searched in source region.

$$\text{Formally } \Psi\hat{q} = \arg\max \Psi q \in \Phi \; d(\Psi\hat{p} , \Psi\hat{q})$$

Where distance $d(\Psi a, \Psi b)$ between two generic patches Ψa and Ψb is defined as sum of squared differences of already filled pixels in the two patches . Having found the source exemplar $\Psi\hat{q}$ the value of each pixel to be fill p' | $p' \in \Psi\hat{p} \cap \Omega$ is copied from its corresponding position inside $\Psi\hat{q}$ .This suffices to achieve the propagation of both structure and information from Source Φ to target Ω one patch at a time.

C. Updating Confidence Values

After the patch $\Psi\hat{p}$ has been filled with new pixels values , the confidence C(p) is updated in the area delimited by $\Psi\hat{p}$ as follows:

$$C(p) = C(\hat{p}) \text{ for all } p \in \hat{p} \cap \Omega$$

This simple update rule allows measuring the relative confidence of patches on the fill front, without image-specific parameters. As filling proceeds, confidence values decay, indicating less conformity of colour values of pixels near the centre.

IV. RESULTS

The algorithm is applied to both images as shown in figures and its PSNR (db), Total time required (sec) are given as follows. This paper has presented a algorithm for removing large objects from digital photographs. The result of object removal is an image in which the selected object has been replaced by a visually plausible background that mimics the appearance of the source region. This is robust algorithm for exemplar based image inpainting, which can be adapted to any image contents of different characteristics.

A. Region Filling



Fig4.1. Original Image ***Fig4.2. Masked Image*** ***Fig4.3.Inpainted Image***

Results:
PSNR (dB):51.46
Total time taken to inpainting the image (sec):64.41

B. Object Removal



Fig.5.1.Original Image ***Fig5.2. Masked Image*** ***Fig5.3.Inpanted Image***

Results:
PSNR (dB):44.40
Total time taken to inpainting the image (sec):863.06

V. CONCLUSION

This paper involves Criminis's algorithm for removing large objects from digital photographs. The result is an image in which the selected object has been placed by a visually plausible background that mimics the appearance of source region. Criminis's algorithm uses an exemplar-based synthesis technique modulated by a unified scheme for determining the fill order of the target region. Pixels maintain a confidence value, which together with image isophotes, influence their filling priority. The technique is capable of propagating both linear structure and 2D textures into target region with single, simple algorithm.

The advantages of Criminisi's algorithm are:

- Preservation of edge sharpness
- No dependency on image segmentation
- Balanced region filling to avoid over-shooting effects.
- Patch-based filling helps achieve speed efficiency, accuracy in synthesis of texture and accurate propagation of linear structures.

The limitations of Criminisi's algorithm are:

- The synthesis of regions for which similar patches do not exist does not produce reasonable results
- Algorithm does not handle depth ambiguities

Future works will certainly involve extensions to current algorithm to handle accurate propagation of curved structures in images. Also investigation of efficient searching scheme and on the automatic discovery of component weights for different types of images as well as removing objects from video, which promise to impose totally new set of challenges.

REFERENCES

[1] A. Criminisi, P. Pérez, and K. Toyama. "Region filling and object removal by exemplar-based image inpainting". IEEE Trans. Image Process., vol. 13, pp. 1200-1212, 2004G.

[2] P. Harrison. "A non-hierarchical procedure for re-synthesis of complex texture." In Proc.Int. Conf. Central Europe Comp. Graphics, Visua. And Comp. Vision, Plzen, CzechRepublic, February 2001J.

[3] Nikhil Sharma, Niharika Mehta. "Region Filling and Object Removal by Exempeler Based Image Inpainting." International Journal of Inventive Engineering and Sciences (IJIES) ISSN: 2319–9598, Volume-1, Issue-3, February 2013.

[4] M.Bertalmio, A.L. Bertozzi, and G. Sapiro. Navier-stokes, "Fluid dynamics, and imageand video inpainting". In Proc. Conf. Comp. Vision Pattern Rec., pages I:355–362, Hawai, December 2001.

[5] A. Efros and W.T. Freeman. "Image quilting for texture synthesis and transfer". In Proc.ACM Conf. Comp. Graphics (SIGGRAPH), pages 341–346, Eugene Fiume, August 2001.

[6] A. Zalesny, V. Ferrari, G. Caenen, and L. van Gool. "Parallel composite texture synthesis". In Texture 2002 workshop - ECCV, Copenhagen, Denmark, June 2002.

[7] A. Criminisi, P. Perez, and K. Toyama. "Object removal by exemplar-based inpainting." In Proc. Conf. Comp. Vision Pattern Rec., Madison, WI, Jun 2003.

[8] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. "Image inpainting." In Proc. ACM Conf. Comp. Graphics (SIGGRAPH), pages 417–424, New Orleans, LU, July 2000. http://mountains.ece.umn.edu/ ~guille/inpainting.htm.

[9] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. "Simultaneous structure and texture image inpainting." In Proc. Conf. Comp. Vision Pattern Rec., Madison, WI, 2003. http://mountains.ece.umn.edu/ ~guille/inpainting.html

