A COMPARETIVE STUDY OF EXEMPLAR BASED IMAGE INPAINTING FOR OBJECT REMOVAL AND REGION FILLING

S. J. Kale[1] P. D. Gawande[2]
[1]P. G. Scholar [2]Associate Professor
[1]Department of Digital Electronics Engineering
[2]Department of Electronics & Telecommunication Engineering
[1,2]Sipna's College of Engineering & Technology, Amravati, Maharashtra, India.

Abstract— *Inpainting, a set of techniques for making undetectable modifications to images, is as ancient as art itself. Various applications of image in painting ranges from the removal of an object from a scene to the retouching of a damaged painting or photograph. For photography and film, in painting can be used to reverse deterioration (e.g., cracks in photographs, scratches and dust spots), or to add or remove elements from photographs. In each case, the goal is to produce a modified image in which the inpainted region is merged into the image so that a typical viewer is not aware that any modification has occurred. In the past, this problem has been addressed by two classes of algorithms: (i) "texture synthesis" algorithms for generating large image regions from sample textures, and (ii) "inpainting" techniques for filling in small image gaps. The former has been demonstrated for "textures" – repeating two-dimensional patterns with some stochasticity; the latter focus on linear "structures" which can be thought of as one-dimensional patterns, such as lines and object contours. In this paper the simultaneous propagation of texture and structure in- formation is achieved by a single, efficient algorithm. Computational efficiency is achieved by a block-based sampling process.*

Keywords— In painting, Texture Synthesis, Structures, Contours, Efficiency.

I. INTRODUCTION

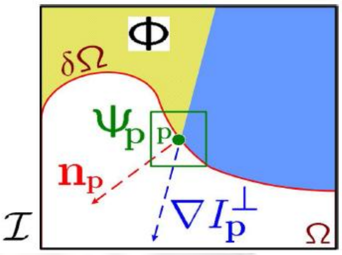
The in painting problem has received recently a growing interest by researchers. This led to various algorithms being developed. Image inpainting technology could be divided into two classes: i) image inpainting based on the geometric image models; ii) image completion based on texture synthesis. Artists generally filled in scratches by trying to follow the original lines and texture from adjacent areas in the painting. Put in mathematical terms, the gradient (or slope) of each brush stroke was estimated. To emulate this process, an algorithm was developed that formulated higher order partial differential equations (PDEs) whose numerical solution propagated information in the best possible direction [1][2]. Although the PDE methods are promising, it might not restore missing images faithfully for the case of simultaneous filling-in of textured and structured backgrounds. Hence, an approach considering the given image as a structural part and a texture part separately was introduced in [3]. The second main approach tackling the inpainting problem uses the existing pixels to map the information of the given image into a state-space model [4], a statistical model [5][6], or a sparse model e.g. [7][8]. The common thing in these approaches is that they extract information from the non-damaged original portions of the image and use it to fill the missing area.

This Algorithm is proposed for removing large objects from digital images. The challenge is to fill in the hole that is left behind in a visually plausible way.[6] presents algorithm that combines the advantages of these two approaches. Exemplar-based texture synthesis contains the essential process required to replicate both texture and structure; the success of structure propagation, however, is highly dependent on the order in which the filling proceeds. This algorithm propose a best-first algorithm in which the confidence in the synthesized pixel values is propagated in a manner similar to the propagation of information in inpainting. The actual color values are computed using exemplar-based synthesis. Computational efficiency is achieved by a block based sampling process. (palette i.e. w)

II. INOTATIONS USED IN IMAGE INPAINTING:

Fig.1 shows a notation diagram in which the meaning of used notation is given.

– Total Image= I
– Target region= Ω
– Its contour= dΩ
– Source region= I- Ω =Φ
– Template size= Ψ
– (palette size)
– Given patch= Ψp
– Exemplar patch=Ψq
– Normal to contour= np
– Abs Ψp= area of Ψp



.

III. KEY OBSERVATIONS

A. *Exemplar-based synthesis suffices*

The core part of algorithm is an isophote-driven image-sampling process. It is well-understood that exemplar-based approaches perform well for two-dimensional textures [9], [10], [11]. But, we note in addition that exemplar-based texture synthesis is sufficient for propagating extended linear image structures, as well; i.e., a separate synthesis mechanism is not required for handling isophotes. Figure 2 illustrates this point. We adopt notation similar to that used in the inpainting literature. The region to be filled, i.e., the target region is indicated by Ω, and its contour is denoted δΩ. The contour evolves inward as the algorithm progresses, and so we also refer to it as the “fill front”. The source region, Φ, which remains fixed throughout the algorithm, provides samples used in the filling process.

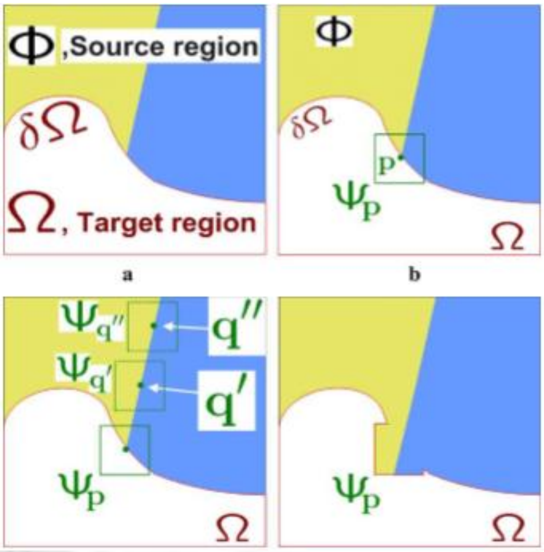


Fig. 2: Structure propagation by exemplar-based texture synthesis (a) Original image, with the target region Ω(b) We want to synthesize the area delimited by the patch Ψp centered on the point p ∈ δΩ.(c) The most likely candidate matches for Ψp lie along the boundary between the two textures in the source region(d) The best matching patch in the candidates set has been copied into the position occupied by Ψp, thus achieving partial filling of Ω.

IV. METHODOLOGY

- In this algorithm each pixel maintains color value (or empty if it is unfilled) and confidence value.
- As the algorithm proceeds the contour is given temporary priority which determines the order in which it is filled.
- The algorithm iterates the following three steps until all pixels are filled:

(1) Computing patch priorities
(2) Propagating texture and structure information
(3) Updating confidence terms

- Given a patch Ψp centered at point p for p є dΩ, the priority P(p)= C(p)*D(p)

Where C(p) is confidence term and given by

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

And D(p) is data term and given by

$$D(\mathbf{p}) = \frac{|\nabla I_{\mathbf{p}}^{\perp} \cdot \mathbf{n}_{\mathbf{p}}|}{\alpha}$$

V. COMPUTING PATCH PRIORITY

Fig. 3 is for deciding patch priority. Higher priority is given to,

(1) Which are on continuation of strong edge.
(2) Which are surrounded by higher confidence term.

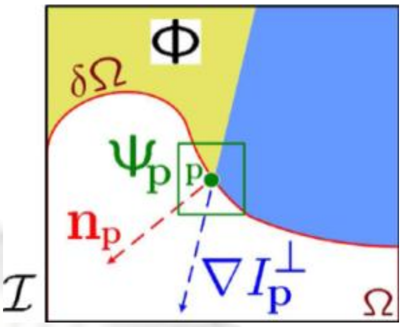


Fig. 3: Patch Priority

P(p)=C(p)D(p),

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

$$D(\mathbf{p}) = \frac{|\nabla I_{\mathbf{p}}^{\perp} \cdot \mathbf{n}_{\mathbf{p}}|}{\alpha}$$

VI. VI. REGION FILLING ALGORITHM

- Extract the manually selected initial front *dΩ.*
- Repeat until done:
- Identify the fill front *dΩ*. If *dΩ* = Φ, exit.
- Compute priorities *P(p)* = *C(p)**D(p)*, \for all *p* \in *dΩ*.
- Find the patch *Ψp* with maximum priority [*P(p)*].
- Find the exemplar *Ψq* \which best matches *Ψp.*
- Copy image data from Ψ*q* to *Ψp.*
- Update *C(p)* \for all *p* | *p* \in (Ψ*p* \intersect *Φ*).[12]

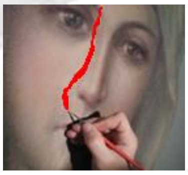
VII. RESULTS AND COMPARISION

Here we applied algorithm to image, and we make side-by-side comparisons with chengs Image Inpainting algorithm and Wavelet Inpainting Algorithm. Following are the results for Exemplar (Criminisi) Algorithm (fig.4), Modified chengs Algorithm (fig.5) and Wavelet Algorithm (Fig.6) with the various palette size (W):

A. Input Images to the Algorithm (Artist Image)

(1)Original image

(2) Mask image



B. Output of Exemplar (Criminisi) Algorithm

W=1 W=3 W=5

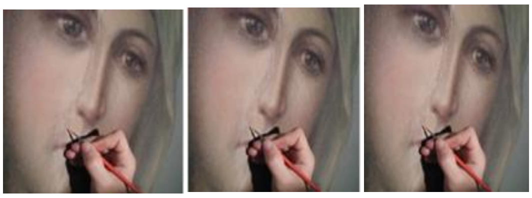


Fig. 4

C. *Out puts of modified chengs algorithm*

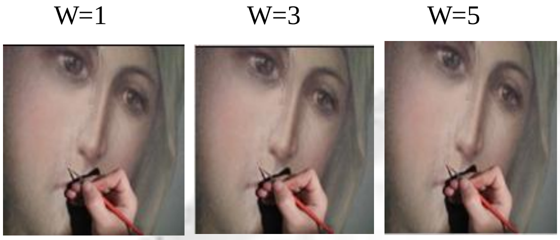


Fig. 5:

D. *Outputs of Wavelet Algorithm*

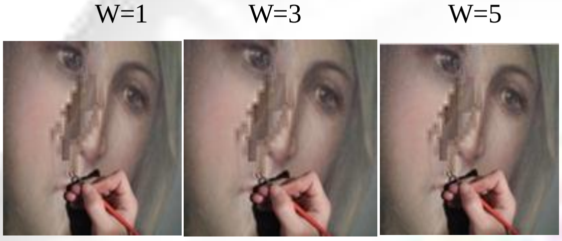


Fig. 6:

Table 1 shows the time required for inpainting and PSNR of different Algorithm.

Algorithm	Image	Palett-e Size (w)	Time taken	PSNR
Exemplar (Criminisi)	Artist	1	100.5690	3.769820e+01
		3	90.9890	3.761603e+01
		5	89.1450	3.769820e+01
Modified chengs	Artist	1	128.6220	3.849888e+01
		3	137.5070	3.849888e+01
		5	138.4440	3.849888e+01
wavelet	Artist	1	19.8900	1.795490e+001
		3	16.9880	2.051519e+001
		5	14.6330	1.729132e+001

Table 1

VIII. CONCLUSION

The inpainted output of Exemplar Algorithm is shown above. The time required and PSNR is also given in the table. As compare to Chengs and Wavelet algorithm the results of Exemplar algorithm are very good and it fills the space with the best matched from the source region.

ACKNOWLEDGMENT

Authors are thankful to Dr. S. A. Ladke, Prof. A. P. Thakare, and Dr. A. A. Gurjar for their encouragement and valuable guidance all the time.

REFERENCES

[1] C. Ballester, M. Bertalmio, V. Caelles, G. Sapiro, and J. Verdera, “Filling-in by joint interpolation of vector fields and gray levels”, IEEE Trans. on Image Processing, vol. 10, no.8, pp. 1200 ~ 1211, Aug. 2001

[2] M. Bertalmio, “Strong-continuation, contrast-invariant inpaiting with a third-order optimal PDE”, IEEE Trans. On Image Processing, vol. 15, no. 7, pp. 1934 ~ 1938, July 2006

[3] M. Bertalmio, L. Vesa, G. Sapiro, and S. Osher, “Simultaneous structure and texture image inpainting”, IEEE Trans. On Image Proc., vol. 12, no. 8, pp. 882-889, Aug. 2003

[4] M. Sznaier, O. Camps, “A Hankel operator approach to texture modeling and inpainting”, Proc. 4th International Workshop on Texture Analysis and Synthesis, pp. 125-130, Oct. 2005.

[5] A. Levin, A. Zomet, and Y. Weiss, “Learning how to inpaint from global image statistics”, Proc. IEEE International conf. Comp. Vision (ICCV 2003), pp. 305-313, Oct. 2003

[6] J. Grim, P. Somol, P. Pudil, I. Mikova, and M. Malec, “Texture oriented image inpainting based on local statistical model”, in the Proc. of the 10th IASTED International conference Signal and Image Processing (SIP 2008), Kailua-Kona, HI, Aug. 2008.

[7] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation”, IEEE Trans. on Signal Processing, vol. 54, no. 11, pp. 4311 ~ 4322, Nov. 2006.

[8] J. Mairal, M. Elad, G. Sapiro, “Sparse representation for color image restoration”, IEEE Trans. on Image Processing, vol. 17, no. 1, pp. 53 ~ 69, Jan. 2008

[9] M. Ashikhmin. Synthesizing natural textures. In Proc. ACM Symposium on Interactive 3D Graphics, pages 217–226, Research Triangle Park, NC, March 2001.

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

[11] A. Hertzmann, C. Jacobs, N. Oliver, B. Curless, and D. Salesin. Image analogies. In Proc. ACM Conf. Comp. Graphics (SIGGRAPH), Eugene Fiume, August 2001.

[12] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, and G. Sapiro. A vari- ational model for filling-in gray level and color images. In Proc. Int. Conf. Computer Vision, pages I: 10–16, Vancouver, Canada, June 2001