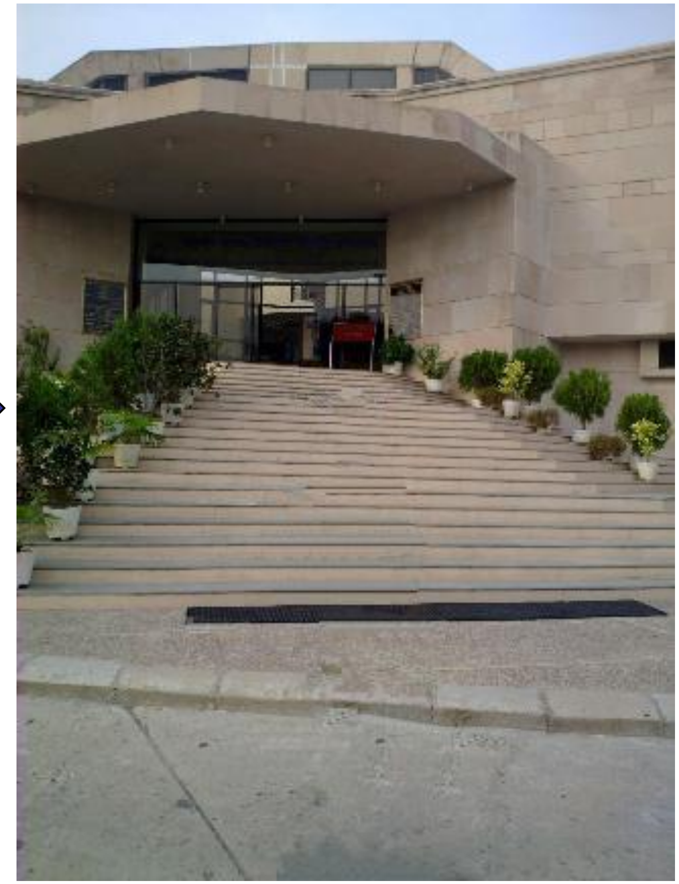
Fast and Enhanced Algorithm for Exemplar Based Image Inpainting (Paper# 132)

Anupam Agrawal Pulkit Goyal Sapan Diwakar
Indian Institute Of Information Technology, Allahabad

Image Inpainting

- Inpainting is the art of restoring lost/selected parts of an image based on the background information in a visually plausible way.
- Large areas with lots of information lost are harder to reconstruct, because information in other parts of the image is not enough to get an impression of what is missing.
- Details that are completely hidden/occluded by the object to be removed cannot be recovered by any mathematical method.
- Therefore the objective for image inpainting is not to recover the original image, but to create some image that has a close resemblance with the original image.

Example



Approaches

Image Inpainting methods can be classified broadly into :-

- Texture synthesis algorithm: These algorithms sample the texture form the region outside the region to be inpainted. It has been demonstrated for textures, repeating two dimensional patterns with some randomness.

- Structure recreation: These algorithms try to recreate the structures like lines and object contours. These are generally used when the region to be inpainted is small. This focuses on linear structures which can be thought as one dimensional pattern such as lines and object contours.

Terms used

Terms used in the literature:

Image: **I**

Region to be inpainted: **Ω**

Source Region (I- Ω): **φ**

Boundary of the target region: **δΩ**

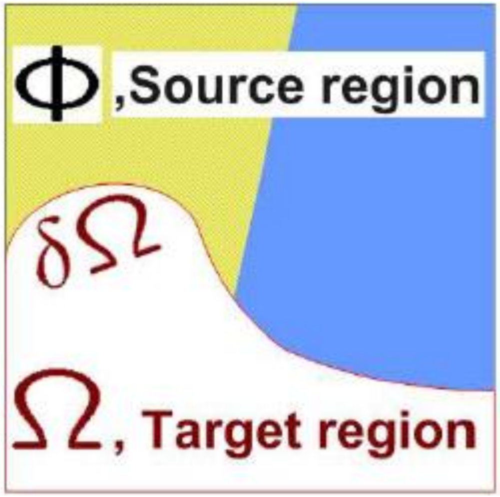


Figure 3: Terminologies used in inpainting [2]

Exemplar Based Approach

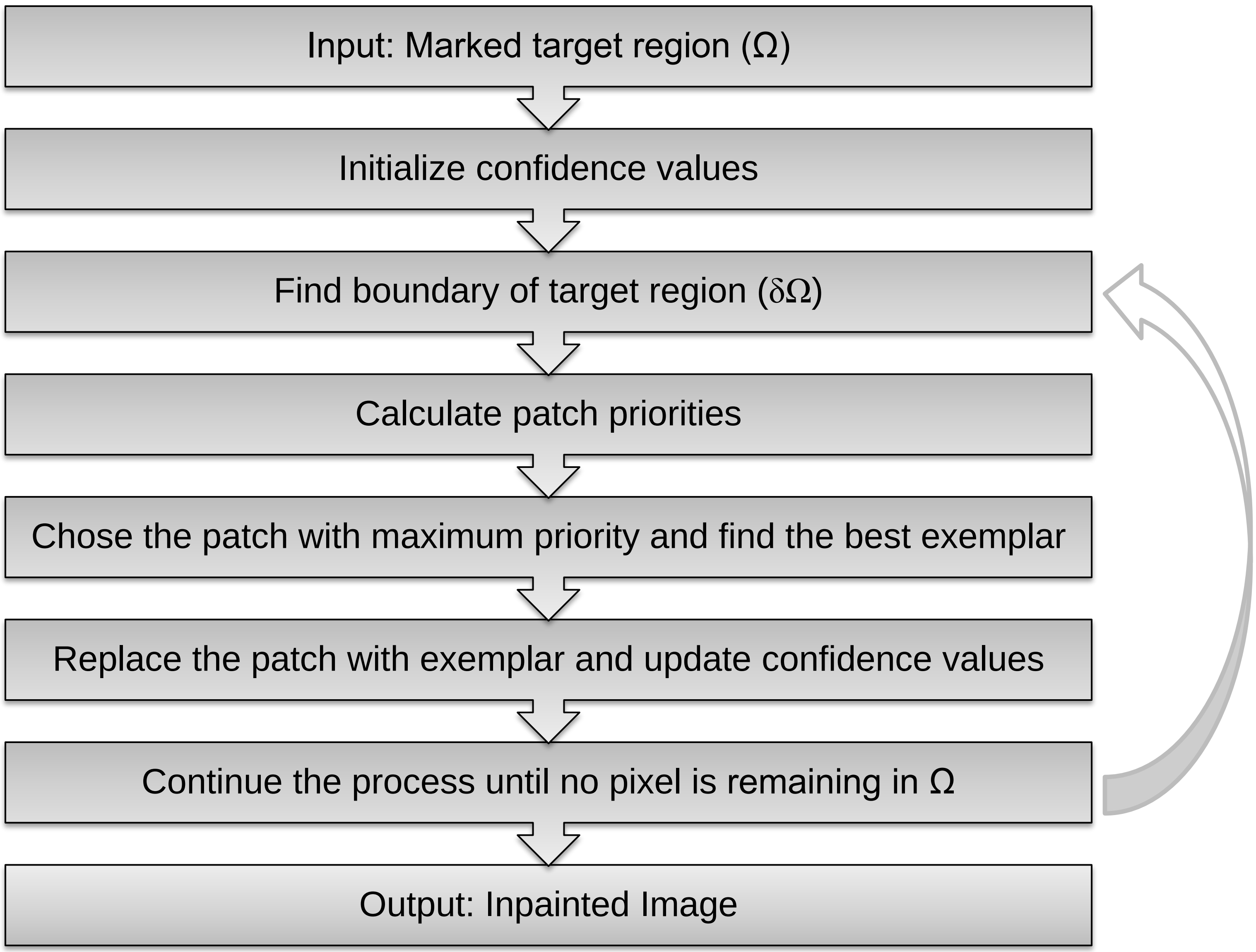
Exemplar based approaches work as follows:

- ***Computing Filling Priorities****, in which a* predefined priority function is used to compute the filling order for all unfilled pixels $p \in \delta\Omega$ *in* the beginning of each filling iteration.

- ***Searching Example and Compositing****, in which* the most similar example is searched from the source region Φ to compose the given patch *Ψ (of size N × N pixels) that centred on the* given pixel p.

- ***Updating Image Information****, in which the* boundary δΩ of the target region Ω and the required information for computing filling priorities are updated.

Methodology



Confidence Value

In this algorithm, each pixel maintains a confidence value that represents our confidence in selecting that pixel.

- This confidence value does not change once the pixel has been filled.

- We initialize the confidence value for all the pixels in the source region (Φ) to be 1 and the confidence values for the pixels in target region (Ω) to be 0.

Priority

To calculate the filling order, we assign priorities to all the patches on the fill front and then take the patch with maximum priority.

$$\mathbf{P}(\mathbf{p}) = \boldsymbol{\alpha} \times \mathbf{R_c}(\mathbf{p}) + \boldsymbol{\beta} \times \mathbf{D}(\mathbf{p})$$

where,

$$C(p) = \frac{\sum_{q \in \psi_p \cap \phi} C(q)}{|\psi_p|},$$

$$D(p) = \frac{|\nabla I_p^{\perp} . n_p|}{\alpha'} \text{ and}$$

$$R_c(p) = (1 - \omega) \times C(p) + \omega.$$

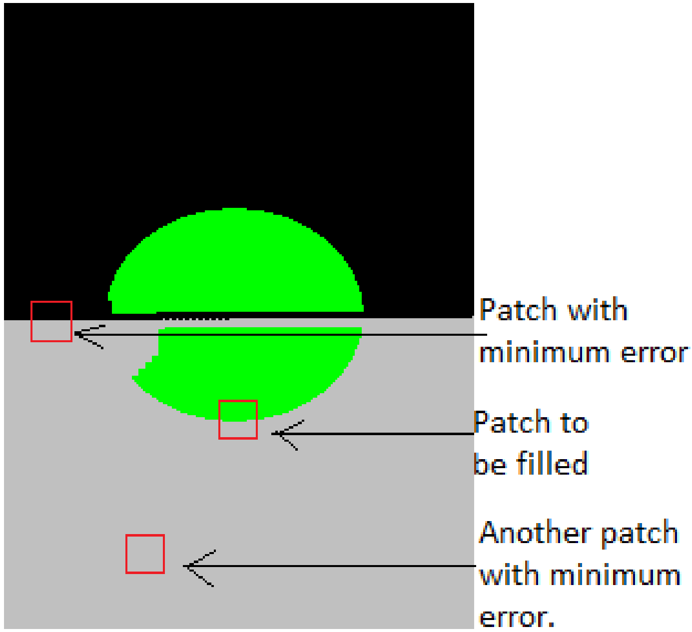
where $|\psi_p|$ is the area of the patch, ψ_p and α' is the normalization factor, n_p is a unit vector orthogonal to the front (δΩ) at the point p and $\boldsymbol{\nabla I_p^{\perp}}$ represents the perpendicular isophote at point p. α, β and ω are constants.

Finding the best Exemplar

- The next step is to find the patch that best matches the selected patch. Mean Square Error can be used to do the same.
- Mean Square error between two patches P and Q is defined as

$$MSE = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} \left(P(i,j) - Q(i,j)\right)^2}{m \times n}$$

- It may happen that two or more patches have the same MSE.



Dealing with patches with same MSE

- Variance can help in differentiating among such patches.
- Find variance of the pixel values of the patch with respect to the mean of the pixels from the same patch that correspond to the pixels belonging to source region from the patch to be inpainted (i.e. pixels that correspond to $\forall p \in \phi \cap \Psi$).

$$\text{i.e., Mean, } \mathbf{M} = \frac{\sum \mathbf{f}_{p\in \phi\cap \Psi}}{\#\{\mathrm{p} \mid p\in \phi\cap \Psi\}} \ \textbf{and}$$

$$\text{Variance, } \mathbf{V} = \frac{\sum(\mathbf{f}_{p\in \phi- \Psi}-\mathrm{M})^2}{\#\{\mathrm{p} \mid p\in \phi- \Psi\}}.$$

where 'f' denotes the pixel value of the element, #{..} represents the cardinality of the set.

Improving Efficiency

- Earlier approaches searched the complete image to find best exemplar.
- We search only the surrounding portions from the image to find the best exemplar.
- The diameter of the surrounding region to search is calculated at run time by taking into account the region to be inpainted.
- We search for the best exemplar from a rectangle defined by (StartX, startY) and (endX, endY), where,

$$startX = \max\left(0, p - \frac{n}{2} - c_r - \frac{D_x}{2}\right) \qquad startY = \max\left(0, p - \frac{m}{2} - c_c - \frac{D_y}{2}\right)$$

$$endX = \min\left(w, p + \frac{n}{2} + c_r + \frac{D_x}{2}\right) \qquad endY = \max\left(h, p + \frac{m}{2} + c_c + \frac{D_y}{2}\right)$$

where,

m = number of rows in the patch.
n = number of columns in the patch.
c_r = maximum number of continuous green pixels in one row
c_c = maximum number of continuous green pixels in one column
D_x and D_y are constants.

Applications

- **Repairing Photographs:** With age, photographs often get damaged or scratched. We can revert deterioration using inpainting.

- **Remove unwanted objects:** Using inpainting, we can remove unwanted objects, text, etc. from the image.

- **Special Effects:** This may be used in producing special effect.

- **Video inpainting:** If extended to video inpainting, it would be able to provide a great tool to create special effects etc.

Results

References

[1] M. Bertalmio, G. Saprio, V. Caselles, and C. Ballester, "Image Inpainting," Proceedings of the 27th annual conference on Computer graphics and interactive technique, 417-424, 2000.

[2] M. Burger, H. Lin, and C.B. Schonlieb, "Cahn-Hilliard Inpainting and a Generalization for Grayvalue Images," UCLA CAM report, 08-41, 2008.

[3] A. Criminisi, P. Perez, and K. Toyama, "Region Filling and Object Removal by Exemplar- Based Image Inpainting," IEEE Transactions on Image Processing, 13(9), 1200-1212, 2004.

[4] M. Elad, J.L. Starck, P. Querre, and D.L. Donoho, "Simultaneous Cartoon and texture image inpainting using morphological component analysis (MCA) ," Journal on Applied and Computational Harmonic Analysis, 340-358, 2005.

[5] P. Elango, and K. Murugesan, K, "Digital Image Inpainting Using Cellular Neural Network," Int. J. Open Problems Compt. Math., 2(3), 439-450, 2009.

[6] M.J. Fadili, J.–L Starck, and F. Murtagh, "Inpainting and zooming using Sparse Representations," The Computer Journal, 64-79, 2009.

References

[7] G. Forbin, B. Besserer, J. Boldys, and D. Tschumperle, "Temporal Extension to Exemplar- Based Inpainting applied to scratch correction in damaged image sequences," Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2005), Benidorm, Espange, , 1-5, 2005.

[8] R.C. Gonzalez, and R.E. Woods, Digital Image Processing, 2nd ed. Pearson Education, 2002.

[9] A.C. Kokaram, R.D. Morris, W.J. Fitzgerald, and P.J.W. Rayner, "Interpolation of missing data in image sequences," IEEE Transactions on Image Processing 11(4), 1509-1519, 1995.

[10] M.M. Oliveira, B. Bowen, R. McKenna, and Y.S. Chang, "Fast Digital Image Inpainting," Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain, 261-266, 2001.

[11] Photo Wipe, http://www.hanovsolutions.com/?prod=PhotoWipe

[12] Restore Inpaint, http://restoreinpaint.sourceforge.net/

[13] C.B. Schonlieb, A. Bertozzi, M. Burger, and H. Lin, "Image Inpainting Using a Fourth-Order Total Variation Flow," SAMPTA'09, Marseille, France, 2009.

[14] T. Shih, et al., "Video inpainting and implant via diversified temporal continuations," Proceedings of the 14th annual ACM international conference on Multimedia, 133-136, 2006.

[15] Studio Lighting, http://www.studiolighting.net/

References

[16] Wen-Huang Cheng, Chun-Wei Hsieh, Sheng-Kai Lin, Chia-Wei Wang, and Ja-Ling Wu, “Robust Algorithm for Exemplar-Based Image Inpainting,” The International Conference on Computer Graphics, Imaging and Vision (CGIV 2005), Beijing, China, 64-69, 2005.

[17] A. A. Efros and T. K. Leung, “Texture synthesis by nonparametric sampling,” Proceedings of IEEE International Conference on Computer Vision, Greece, 1033-1038, 1999.

[18] Q. Chen, Y. Zhang and Y. Liu, “Image Inpainting With Improved Exemplar-Based Approach,” Multimedia Content Analysis and Mining, LNCS, vol. 4577/2007, pp.242-251, 2007

Thank You.

Authors: Anupam Agrawal, Pulkit Goyal, Sapan Diwakar
anupam@iiita.ac.in {pulkit, sapan}@daad-alumni.de