

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

Region Filling and Object Removal By Exemplar Based Image Inpainting

Satish D. Thakar¹, Dr.R.P.Borole²

Student, Dept. of E&TC, DYPCOE AMBI Pune, Savitribai Phule Pune University, Pune, India

Professor, Dept. of E&TC, DYPCOE AMBI Pune, Savitribai Phule Pune University, Pune, India

ABSTRACT: Image Inpainting is the art of filling in missing data in an image. The purpose of inpainting is to reconstruct 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 inpaint large regions but also recover small portions (e.g. repair a photograph by removing scratch). The basic idea behind exemplar based inpainting this approach is to find examples from the image and replace the lost data with it. This 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,priority, objectremoval

I. INTRODUCTION

The changes of images in a way that is not detectable for an observer who doesn't know about the original image. The aim of Inpainting is to recovering the missing or distorted portions of images photograph, in order to restore its original identity. The need to touch up image in an undesirably noticeable way extended naturally from paintings to digital films. The main aim remains same, to add or remove elements.

Image inpainting and texture synthesis these two are most commonly used techniques to fill the gap after object removals. Digital image inpainting refers to the inpainting process performed on digitized images. It is desired to have an algorithm which can fill in a region marked by the user. The user provides the image with the marked region to be inpainted, which shall be referred to as the mask. The fill in of these marked regions by the algorithm is in such a way that resulting image looks natural and without alteration. Object removal from images is an image handling technique.

Texture reproduction and extension is used to fill in the gaps in images. Many approaches were developed for synthesizing textures including statistical and image based methods. Application of these methods for filling the gaps in image is called constrained texture synthesis. It is used for filling images, with more textured areas. It is observed that structural inpainting, texture synthesis both cannot provide the quality solution. The combination of these two produces better results. Image inpainting considers the image as the collection of structures, shapes, objects that are separated from one another by sharp edges but each one being smooth itself and texture considers the low stochasticity features of the image. Number of approaches are introduced to reconstruct images using inpainting techniques. These are listed as level lines, PDE based inpainting, fluid interpolation, Eluer's elastic, bounded variation and heat transfer [1,2,3,4,5,6]. All such methods have their own advantages and limitations. The proposed Exemplar based inpainting approach overcomes many of the limitations of earlier approaches.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

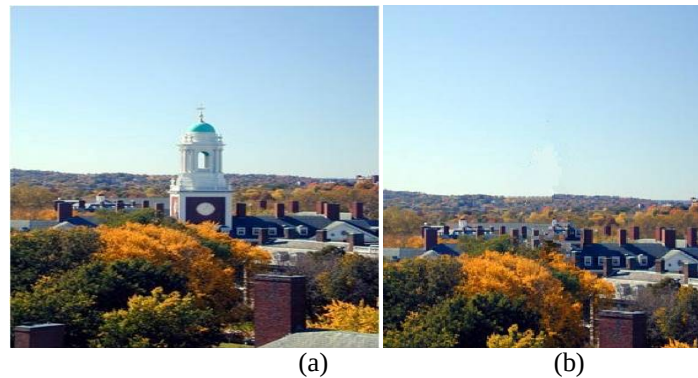


Fig.1: Removing large objects from an image. (a) Original image. (b) The building that had been selected manually automatically removed from the image and the information from background is merged into the missing region.

Digital image inpainting first brought into the field of image processing[1]. This algorithm simulates the traditional inpainting processes, such as identification of the area to be corrected, the marking of boundary of the region to be filled, and continuing the lines of similar colour. These techniques are effective only for small scratches. The algorithms work well in images that are relatively smooth and low noise or texture. The process is unable to fill in regions that are highly textured or contain large noise that leads to non-continuation and blurring of the edges. Since the time, there have been a number of algorithms proposed to solve the inpainting problem [4,5,6]. These are used to fill only narrow gaps in images. They fail to reconstruct large damaged regions in the images. Motion estimation and autoregressive models to interpolate losses in films from adjacent frames are used [2]. The basic idea is to copy the right pixels from neighbouring frames into the gap. The technique cannot be applied to still images or films, where the regions to be inpainted span over many frames. The work [9] decomposes the original image into two components; which processed separately by inpainting and texture synthesis. The outputs of the two are summed up as resultant image. This approach still remains limited to the removal of small image cracks; and diffusion process continues to blur the filled region. Drori et al [14] describes an algorithm that interleaves a smooth approximation with example-based detail synthesis for image completion. Both of these algorithms proposed are found to be extremely slow. First attempt to use exemplar-based synthesis specifically for object removal was proposed [15]. He proposed the order in which a pixel in the target region is filled was dictated by the level of “texturedness” of the pixel’s neighbourhood. Although the intuition is sound, strong linear structures were often over ruled by nearby noise, reducing the value of the extra computation. A related technique drove the fill order by the local shape of the target region, but fails to explicitly propagate linear structures [13].

Several researchers have considered texture synthesis as a way to fill large image regions with “pure” textures. Pure textures are defined as repetitive two-dimensional textural patterns with moderate stochasticity. This is based on a large research carried out on texture synthesis, which seeks to replicate texture from the given small source sample of pure texture. These exemplar-based techniques cheaply and effectively generate new texture by sampling and copying colour values from the source [10-12].

In contrast with previous approaches, it was needed to introduce the technique that does not require the user to specify where the information to fill comes from. And, this should be automatically accomplished by the algorithm. The technique should allow to simultaneously fill-in numerous regions containing completely different structures and surrounding backgrounds. In addition, no limitations should be imposed on the topology of the region to be inpainted.

Therefore, exemplar-based approach of Criminisi et al. seems to be more appropriate [8] and a still further modification is being proposed in the present paper. This results in an algorithm that has the efficiency and qualitative performance of exemplar-based texture synthesis, as well as linear structure inpainting. Software currently available for this task is

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

named 'Photo-Wipe' [18] by Hanov Solutions. It provides tools for selecting the region to be inpainted and then provides several options to carry out the inpainting process with varying time and quality.

Paper outline-The details of the proposed algorithm is described in section II. Section III contains implementation details of the algorithm. Finally, a large gallery of results on both synthetic images and real-scene photographs is presented in section IV. Section V concludes the paper with comments on results.

II. KEY OBSERVATIONS

Both, image inpainting and texture synthesis have their strengths and weaknesses. Image inpainting extends linear structure into the gap by utilizing the isophote information of the boundary pixels leads to natural propagation of linear structures into the gap. However, since the extension actually using the diffusion techniques, artefacts e.g. blur could be introduced. On the other hand, texture synthesis copies the pixels from existing parts of the image, thus avoids the blur. The shortcoming of texture synthesis is that it focuses on the whole image space, and not resolving priorities of pixels on texture and linear structures around the boundary of the gap. As a result, the texture propagation dominates and linear structures do not be naturally extended into the gap. The result would likely to have distorted lines, and noticeable differences between the gap and its surrounding area.

One interesting observation is that even though image inpainting and texture synthesis seem to differ, yet they might actually complement each other. If we could combine the advantages of both approaches, we would get a clear gap filling that is the natural extension from the surrounding area. Criminis, et.al. proposed algorithm that exactly does this [8]. They use the sampling concept from Efros' et.al.[7] approach. The improvement over Efros'[7] is that the new approach takes isophote into consideration and give higher priority to those points which are on boundary of gap as well as on structures, isophote, corners leads to natural extension of those into the gap. To identify those points, Criminis [8] gives a priority value to all the pixels on the boundary of the gap. These important points on structure and corners will get higher priorities according to the algorithm, and thus the linear structures would be extended first. For each pixel on the boundary, a patch is constructed with that pixel at the centre (fig2). The patch's priority is the product of two elements: a confidence term $C(p)$ and a data term $D(p)$.

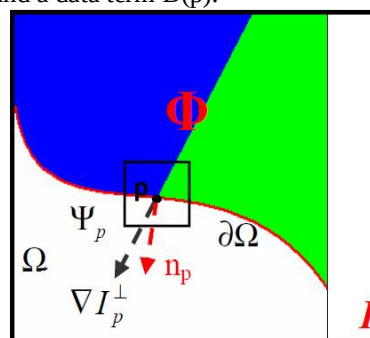


Fig.2: Notation diagram. Given the patch ψ_p , n_p is the normal to the contour $\partial\Omega$ of the target region Ω and $\nabla I_p^\perp$ is the isophote (direction and intensity) at point p . Source region is denoted by ϕ . The entire image is denoted with I

III. IMPLEMENTATION DETAILS

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 $\Phi = I - \Omega$. Next the size of window Ψ must be specified. Criminis 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 executes the following three steps until all pixels have been filled

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

- 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 totally 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 and which are surrounded by high-confidence pixels. Given a patch Ψ_p centered at the point p for some $p \in \partial\Omega$ 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:

$$C(p) = \frac{\sum_{q \in \Psi_p \cap (I - \Omega)} C_q}{|\Psi_p|} \quad \text{and} \quad D(p) = \frac{\nabla I_p \cdot n_p}{\alpha}$$

$|\Psi_p|$ is area of Ψ_p , α is normalization factor (e.g. $\alpha = 255$ for typical grey level image), n_p is unit vector orthogonal to front $\partial\Omega$ 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 zero for all $p \in \Omega$ and $C(p) = 1$ for all $p \in I - \Omega$. The confidence term $C(p)$ may be thought of as a measure of the amount of reliable information surrounding the pixel p . the 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.

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.

$$\Psi_{\hat{q}} = \arg \min_{\Psi_q \in \Phi} d(\Psi_{\hat{p}}, \Psi_q)$$

Where distance $d(\Psi_a, \Psi_b)$ between two generic patches Ψ_a and Ψ_b 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 $p' \mid p' \in \Psi_{\hat{p}} \cap \Omega$ 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:

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

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.

Pseudo Code

- Start
- Extract the manually selected initial front $\partial\Omega$
- Repeat until done:
 - 1a. Identify the fill front $\partial\Omega^t$. If $\Omega^t = \varnothing$, exit.
 - 1b. Compute priorities.
 - 2a. Find the patch with the maximum priority.
 - 2b. Find the exemplar that minimizes the error.
 - 2c. Copy image data from source patch to target patch.
 3. Update Confidence term.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

- End.

't' indicates the iteration number.

IV. RESULTS AND COMPARISONS

This section presents comparison of results of the algorithm in [1],[18] and our exemplar based inpainting algorithm. Here, algorithm is applied to a color images, as shown in fig.3 that include complex structure and textures. The patch size is set to be greater than the largest texture element or the thickest structure (e.g., edges) in the source region. The additional experimental results with different photographs are shown in fig 4 to fig.6. In all results (a) is degraded image, (b) is the mask image and (c) is the result of our inpainting algorithm. Result of Fig.7,8 shows large object removal and reconstruction of relative smooth background. The statistical comparisons of a few representative results (Time and PSNR) are tabulated in table 1.

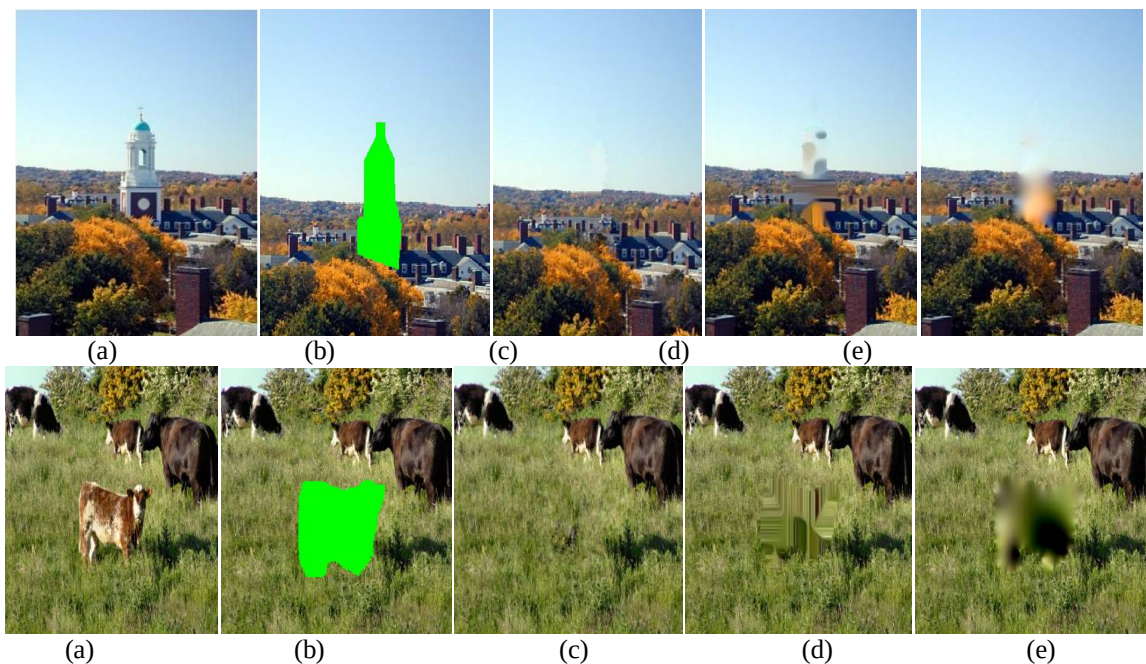


Fig.3: (a) The original image (b) Masked image (c) Output of proposed algorithm (d) Output using Photo Wipe's Full Quality Inpainting option (e) Output of PDE based inpainting

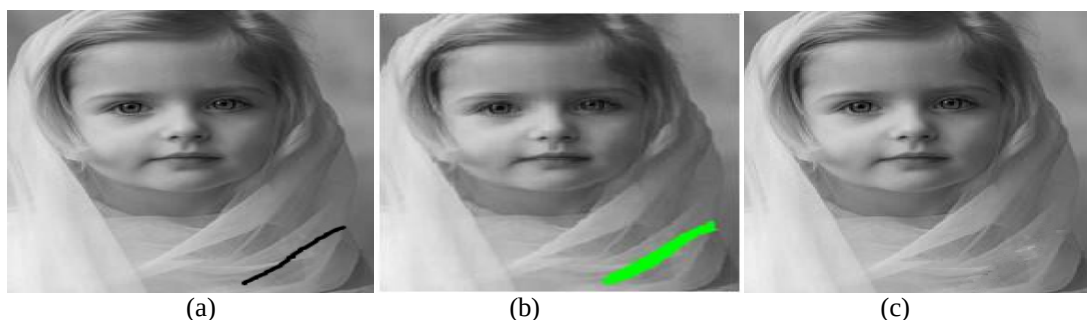


Fig.4: Results with smaller scratch size(nice1)

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

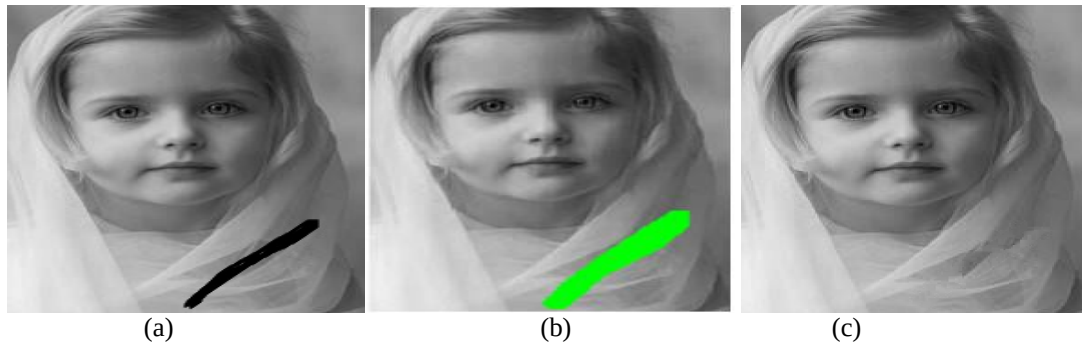


Fig.5:Results with intermediate degradation (nice2).

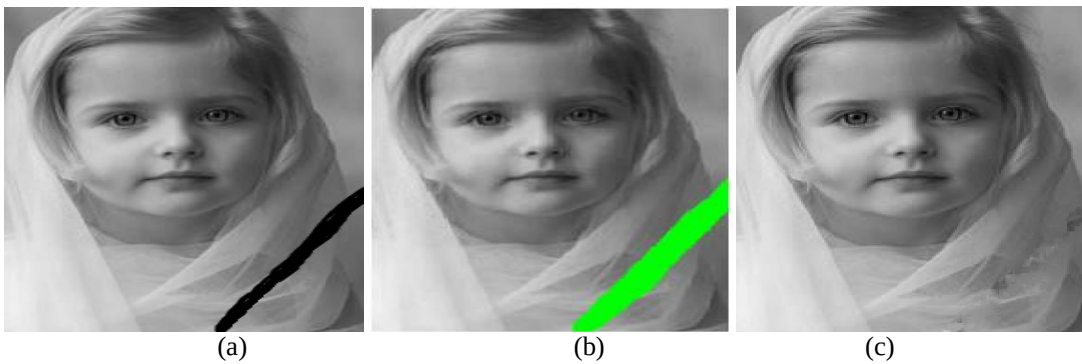


Fig.6: Results with larger degradation (nice3).



Fig.7: Object removal - textured background.



Fig.8: Removing an object on a highly textured background.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

Image	Nice1	Nice2	Nice3
Size of Image	256x157	256x157	256x157
Pixels to be filled	691	1548	2238
PSNR1(Original & Degraded)	24.06	20.45	19.27
PSNR2(Original & Inpainted)	45.33	37.88	37.68
Time(s)	7.69	17.46	25.18

Table1: Comparison of PSNR for different Images

V. CONCLUSIONS

We present an algorithm that can remove objects from the image in a way that it seems reasonable to the human eye. It can also restore old photographs (e.g. removal of scratches). Our approach explains an exemplar based inpainting method along with a priority term that defines the filling order in the image. In this algorithm, pixels maintain a confidence value and are chosen based on their priority that is calculated using confidence and data term. The approach defines a way of differentiating between patches that have the same minimum mean squared error with the selected patch. This approach is capable of propagating both linear structures and two dimensional textures into the target region. This technique can be used to fill small scratches in the image as well as to remove larger objects from them. It is also computationally efficient and works well with larger images.

REFERENCES

- [1] Bertalmio M., Sapiro G., Caselles V., Ballester C., Image Inpainting, *SIGGRAPH*, pp. 417-424, 2000.
- [2] Kokaram A., Morris R., Fitzgerald W., Rayner, P., Interpolation of Missing Data in Image Sequences, *IEEE Transactions on Image Processing*, pp. 1509-1519, 1995.
- [3] Hirani, A., Totsuka, T., Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal, *SIGGRAPH*, pp. 269-276, 1996.
- [4] S.Masnou and J.-M.Morel., Level lines based disocclusion, *Int. Conf. Image Processing*, Chicago, 1998.
- [5] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, and G. Sapiro., A variational model for filling-in gray level and color images. *Proc. Int. Conf. Computer Vision*, pp. 10– 16, Vancouver, Canada, 2001.
- [6] T. F. Chan and J. Shen., Non-texture inpainting by curvature-driven diffusions (CDD), *J. Visual Comm. Image Rep.*, pp. 436–449, 2001.
- [7] A.Efros and T.Leung, "Texture synthesis by non-parametric sampling" in 17th IEEE International conference on computer vision, 1999 pp 1033-1038.
- [8] A. Criminisi, P. Perez, K. Toyama., Region filling and object removal by exemplar-based inpainting, *IEEE Transactions on Image Processing* 9, pp. 1200-1212, 2004.
- [9] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher., Simultaneous structure and texture image inpainting, *Proc. Conf. Comp. Vision Pattern Rec.*, Madison., 2003.
- [10] A.Efros and W.T. Freeman., Image quilting for texture synthesis and transfer, *Proc. ACM Conf. Comp. Graphics(SIGGRAPH)*, pp. 341–346, Eugene Fiume, 2001.
- [11] L. Liang, C. Liu, Y.-Q.Xu, B. Guo, and H.-Y.Shum., Real-time texture synthesis by patch-based sampling, *ACM Transactions on Graphics*, 2001.
- [12] M. Ashikhmin., Synthesizing natural textures, *Proc. ACM Symposium on Interactive 3D Graphics*, pp. 217–226, Research Triangle Park, NC, 2001.
- [13] R. Bornard, E. Lecan, L. Laborelli, and J-H.Chenot., Missing data correction in still images, *Proc. Conf. Comp. Vision Pattern Rec.*, Madison., 2003.
- [14] I. Drori, D. Cohen-Or, and H. Yeshurun. Fragment-based image completion., *ACM Trans. on Graphics (SIGGRAPH2003 issue)*, pp 303–312, San Diego, US, 2003.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly Peer Reviewed Journal)

Vol. 5, Issue 2, February 2016

- [15] P. Harrison., A non-hierarchical procedure for re-synthesis of complex texture., In *Proc. Int. Conf. CentralEurope Comp. Graphics, Visua. And Comp. Vision*, Czech Republic, 2001.
- [16] R. P. Borole and S. V. Bonde, "Image analysis & restoration by exemplar inpainting," in *Proceedings of National Conference on Communication, Computing and Networking (NCCCNT'13)*, Nanded, India, 2013.
- [17] R. P. Borole and S. V. Bonde, "Patch-based inpainting for object removal and region filling in images," *Journal of Intelligent Systems*, vol. 22, no. 3, pp. 335-350, 2013.
- [18] Photo Wipe, <http://www.hanovsolutions.com/?prod=PhotoWipe>.