ORIGINAL RESEARCH PAPER

Volume-7 | Issue-12 | December-2018 | PRINT ISSN No 2277 - 8179

INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH

VIDEO INPAINTING WITH SHOT-TERM WINDOWS: APPLICATION TO OBJECT REMOVAL AND ERROR CONCEALMENT

Engineering

Vivek H Raval PG Student, EC Department, Aadishwar College of Technology - Venus, Gandhinagar

Shailee Patel* Assistant Professor, EC Department, Aadishwar College of Technology-Venus, Gandhinagar *Corresponding Author

ABSTRACT

In this paper, we propose a new video inpainting method which applies to both static and free-moving camera videos. The method can be used for object removal, error concealment and background reconstruction applications. To limit the computational time, a frame is inpainted by considering small number of neighboring pictures which are grouped into a group of pictures (GoP). More specifically, to inpaint a frame, the method starts by aligning all the frames of the GoP. This is achieved by a region-based homograph computation method which allows us to strengthen the spatial consistency of aligned frames. Then, from the stack of aligned frames, an energy function based on both spatial and temporal coherency terms is globally minimized. This energy function is efficient enough to provide high quality results even when the number of pictures in the GoP is rather small, e.g. 20 neighboring frames. This drastically reduces the algorithm complexity and makes the approach well suited for near real-time video editing applications as well as for loss concealment applications. Experiments with several challenging video sequences show that the proposed method provides visually pleasing results for object removal, error concealment and background reconstruction context.

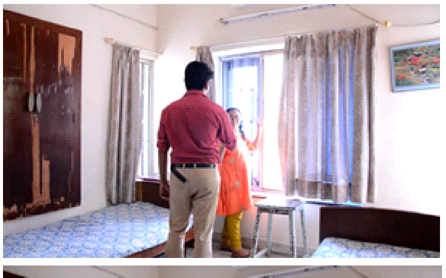
KEYWORDS

Video Inpainting, Object Removal, Error Concealment,

I. INTRODUCTION

Inpainting is the process of reconstructing lost or missing parts of images and videos. For instance, in the case of a valuable painting, this task would be carried out by a skilled image restoration artist. In the digital world, inpainting (also known as image interpolation or video interpolation) refers to the application of sophisticated algorithms to replace lost or corrupted parts of the image data (mainly small regions or to remove little defects. Image inpainting is the collective name under which all related methods reconstruct missing parts of an image by using information in pixels surrounding the holes or inpainting domains. The objective for image inpainting is that not to recover the original image, but to create some image that has a close resemblance with the original image.

There are many applications of image inpainting. It can be used in cinema and photography for "restoration", for removing effects like scratches, dust spot from images (called deterioration). It can also be used for removing some object from image or removing red eye removal. Image denoising is famous problem in image processing field. Image inpainting and image denoising are not same. Below figure shows the difference between both.



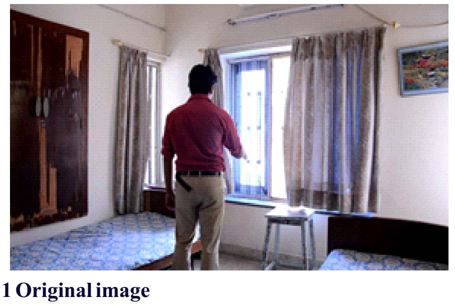


Figure 1 Original image
Figure 2 Inpainted image

Approaches of Image Inpainting

Image Inpainting methods can be classified broadly into:-

1) Texture synthesis algorithms**:** These algorithms sample the texture form the region outside the region to be inpainted. It has been demonstrated for textures, repeating 2dimensional patterns with some randomness.
2) 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.

II. PROPOSED ALGORITHM

This method is concerned with the inpainting of the image. It receives the image from the user where the region to be inpainted is marked in green (R = 0, G = 255, B = 0). As mentioned earlier, we have chosen green color because of its use in the creation of special effects in movies etc.

Let us first describe the terms used in inpainting method:

a) The image to be inpainted is represented as **I**.
b) The target region (i.e., the region to be inpainted) is represented as Ω.
c) The source region (i.e., the region from the image which is not to be inpainted and from where the information can be extracted to reconstruct the target region) is represented as **Φ**.

$\Phi = I - \Omega$

d) The boundary of the target region (i.e., the pixels that separate the target region from the source region) is represented as **δ Ω**.

We have extended the algorithm developed by Criminisi et. al. with a few modifications. As with all other exemplar based algorithms, this algorithm replaces the target region patch by patch. This patch is generally called the template window, ψ. The size of ψ must be defined for the algorithm. This size is generally kept to be larger than the largest texture element in the source region. We have kept the default patch size of 9 x 9 but we may have to vary it for some images. Once these parameters are assigned the remaining process is completely automatic. The algorithm now proceeds as follows:

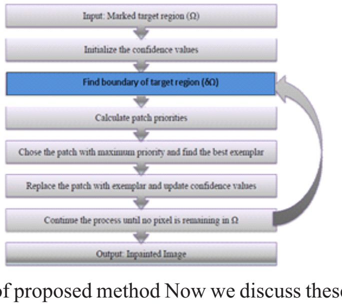


Figure 3 Flow of proposed method Now we discuss these steps one by one.

The input to the inpainting module is the image with target region marked in green color.

The first step is to initialize confidence values. First let us understand what these values represent. 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. Once we have the target region, we find the boundary of the target region. For this, we first construct a Boolean matrix where we put 1's corresponding to pixels which are in the target region (Ω) and zero at other places. Let us call this matrix as fill Region as it denotes the region that is to be filled. We can then find the boundary of the target region by convolving the fill Region matrix with a Laplacian filter. We use the following Laplacian filter for this task.

Laplacian Filter

1	1	1
1	-8	1
1	1	1

The next step is to compute the priorities for the patches centered on the pixels in $\delta\Omega$. The result of the inpainting algorithm depends on the order in which the target region is filled. In criminisi method a different method for estimating the filling order is defined which takes into account the structural features of the image. They fill the target region in a best-first-filling order that depends entirely on the priority values assigned to the patches on the fill front ($\delta\Omega$). They have developed the priority term such that it is biased towards the patches that contain the continuation of edges and are surrounded by high confidence pixels.

III. IMPLEMENTATION OF PROPOSED ALGORITHM

Implementation of Proposed algorithm, we take the original image which we have to inpaint. Initialize the target region. This is generally performed separately from the inpainting process and requires the use of an additional image processing tool. This is performed by marking the target region in some special colour. Without any loss of generality, let us consider that the colour that the target region will be marked in is green (i.e. R = 0, G = 255, B = 0). In proposed algorithm the addition of weights to different components in the definition of priority term so that a balance between confidence and data term could be maintained. where regularizing factor for controlling the curve smoothness. Results of our proposed approach is shown below.

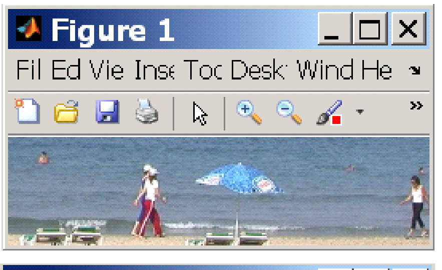
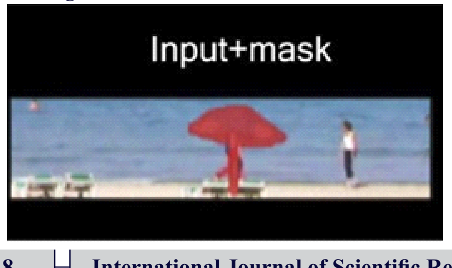


Figure. 4 Original Video
Figures. 5 Region Fill video



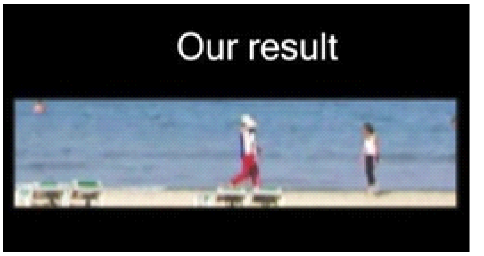


Figure. 6 Final Output

IV. APPLICATIONS

There are many objectives and applications of this technique. In photography and cinema, is used for film restoration; to reverse the deterioration (e.g., cracks in photographs or scratches and dust spots in film; see infrared cleaning). It is also used for removing red-eye, the stamped date from photographs and removing objects to creative effect. This technique can be used to replace the lost blocks in the coding and transmission of images, for example, in a streaming video. It can also be used to remove logos in video.

V. CONCLUSION

This survey shows the various patch based and object based video inpainting algorithms. Patch based video inpainting algorithms are usually extended from the existing video inpainting algorithms while object based video inpainting algorithms was introduced since most of the patch based algorithms were unable to perform both spatial and temporal aspects simultaneously. Each algorithm has its own merits and demerits. We can choose our algorithm depending on our own requirements.

VI. REFERENCES

[1] Anupam, PulkitGoyal and SapanDiwakar, "Fast Enhanced Algorithm for Exemplar Based Image Inpainting in Information Technology IIIT.
[2] Ms.M.D.Kawade and Mr. A.S.Ufade, "Exemplar Based Video Inpainting" in International Journal of Modern Trends in Engineering and Research Date: 2-4 July, 2015.
[3] Abbas KOOCHARI and Mohsen SORYANI, "Exemplar-based Video Inpainting for Occluded Objects" in International Journal of Computer Applications (0975 – 8887) Volume 81 – No.13, November 2013.
[4] Anu Rachel Abraham, A. KethsyPrabhavathy and J. Devi Shree, "A Survey on Video Inpainting" in International Journal of Computer Applications (0975 – 8887) Volume 55– No.9, October 2012.
[5] Kedar A. Patwardhan, Guillermo Sapiro and Marcelo Bertalmío, "Video Inpainting Under Constrained Camera Motion" in IEEE TRANSACTIONS ON IMAGE PROCESSING.
[6] Ms. Manisha P Dhande and Prof.AmritPriyadarshi, "Video Inpainting with Video inpainting is using Super Resolution" in International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169Volume: 3 Issue: 6.
[7] Vaishali U. Gaikwad and P. V. Kulkarni, "Exemplar-based video inpainting with large patches" in Koochari et al. / J Zhejiang Univ-Sci C (Comput& Electron).