

# An Efficient Video Interpolation Framework for Unwanted Object Removal

Dnyanesh S. Gaikwad<sup>1</sup>, Prof. D. B. Kshirsagar<sup>2</sup>

PG Student, Computer Engineering, SRES's College of Engineering, Kopergaon, India<sup>1</sup>

HOD, Computer Engineering, SRES's College of Engineering, Kopergaon, India<sup>2</sup>

**Abstract:** When videos and photographs are captured some unwanted objects come in. On removal of such unwanted objects through image editing tools result in blurred image. It may not be removed properly. There may be case that one wants to replace that undesired object with some new object. In such case both texture and structure of the surrounding region should be preserved. Due to motion of camera some portion contains blur pixels. Some images might appear with missing portions and having cracks. There might be titles, headings written on photographs can be removed using image inpainting. In this paper, a framework is implemented for video inpainting using exemplar based image inpainting. Initially, a video containing unwanted objects is processed and frames are extracted containing unwanted objects and inpainted using exemplar based image inpainting. After inpainting, inpainted frames are saved and all frames are merged to a video. Quality of inpainted video is measured by PSNR.

**Keywords:** Unwanted object, Image Inpainting, Video Inpainting, Texture, Blurred Image.

## I. INTRODUCTION

In old days images are only used for capturing memories. But now images have become more important. Images can be two-dimensional, or three-dimensional. They may be captured by optical devices such as cameras, mirrors, and lenses. Today, for encryption, processing, authentication, sharing etc. purpose images can be used. But the main aim of image is still being preserve i.e. to store the memories. We know that video is a sequential recording of collection of images. While capturing video some unwanted objects like headings, shadows, titles, captions come in video sequence. There may be blur due to motion of camera. Some text in a video sequence may contain valuable information. Though they provide additional information, not all of them are important. They should be removed with preserving texture and structure of image. This arises the need of an automatic approach to remove unwanted objects from video sequences by using surrounding information of image.[1]

In this system, user loads the video for inpainting purpose. Then play the video and extract the frames from video containing undesired objects. After frame extraction, user can select the frames for inpainting from where frames are saved. User can select one image at a time. Here user needs to be select the frame of the video that contains the object which needs to be removed from the video. All frames needs to be inpainted those contains targeted object one by one. After selecting image, user can select the area of an image where object is placed. Target area is selected by plotting points along the area of unwanted object. User can select the area simply selecting the pixels which are surrounding to the targeted object. If first pixel is the last pixel then it can be considered as selection process is completed and it can filled with green color. It can be

considered as selected area of the targeted object. Exemplar based image inpainting process is applied to remove the object. User has two options for inpainting an image, first one is run means it can considered whole image as library from where it can search for best patch and second is the fast run where part of the image can considered for the best patch. Finally, if selected area of the object completed then message will be displayed to the user for intimation purpose only.



Figure 1. Before and after inpainting

After completion of inpainting, user can save an image from where it was selected. User can change the path for save the in painted image by clicking save as menu item. Finally, user can create the video by using inpainted frames. All frames are inpainted those were containing object. After completion of inpainting, these frames are combined and video is generated. This video is the inpainted video.

II. LITERATURE SURVEY

Olivier Le Meur, Mounira Ebdelli, and Christine Guillemot et al.[1] introduced a framework for image inpainting. They have used super-resolution algorithm to recover details on the missing areas of image. They have combined the results by loopy belief propagation and details are recovered by a singleimage super-resolution algorithm. M. Bertalmio, A. L. Bertozzi, G. Sapiro et al.[3] uses ideas from classical fluid dynamics to propagate isophote lines continuously from the exterior into the region to be inpainted. This method applies ideas of using PDEs in classical fluid dynamics to continuously propagate isophote lines of the image from the exterior into the inpainting zone. The method is directly based on the Navier-Stokes equations of fluid dynamics. Mahmoud Ghoniem, Youssef Chahir et al.[4] introduced a discrete p-Laplacian regularization on a weighted graph is utilized in the video inpainting scheme. They exploit temporal and spatial redundancies in order to obtain high quality results by considering a video sequence as a volume. Kedar A. Patwardhan et al.[5] introduced moving object segmentation is performed to separate foreground video from the background in order to reduce the search space for completion of partially occluded moving objects. Guillermo Sapiro et al.[6] introduced a video inpainting technique to handle various camera motions. In this method a motion confidence value is employed to find the priority of the area to be filled in the foreground completion task. Timothy K. Shih, Nick C. Tang et al.[7] presented a technique to replace the removed objects in a video sequence by implanting objects from another video. In this method, modifications are carried out based on analysis of continuities on stationary and nonstationary videos to find the best priority.

A. Conclusions from Literature Survey

Existing system does not take into effect the spatial and temporal information available in the video and cannot handle it. They treats the video as individual images. Due to extensive smoothing, it does not reproduce the texture information and suffers from severe blurring artifacts. They are effective only in restoring small scratches or spots occurring in archival footage. They have high over-smoothing of the reconstructed pixels. Some systems are sensitive to noise as they involve local motion estimates by a derivative-based process.

III.SYSTEM DESIGN

Missing parts in the video sequence are caused by undesired object removal or damages. Restoration problem of such missing regions is modeled by video inpainting. The remaining holes after text removal in our framework are fixed by a video inpainting algorithm. It is worth noting that preserving the spatio temporal consistency is a necessity which makes video inpainting schemes rather different from image inpainting ones. System architecture of implemented system is shown in fig 2

A. Frame Extraction module:

Initially, A video containing unwanted objects is input to frame extraction process. Frames are extracted to a specific location by using following steps. MPEG Frame Grabber Class is initialized by loading the video from which the frame is to be extracted. Then Frame grabber class is started by calling start method from frame grabber object. For each image I from framegrabber grab() method, store the grabbed image into a BufferedImage class and write the BufferedImage into physical image. For all the grabbed frames from framegrabber store the images into a folder. The frame into which the unwanted object is seen is taken for inpainting and then select that frame to inpaint.

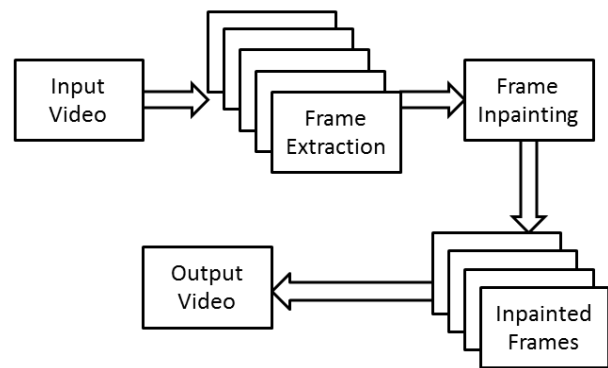


Fig. 2. System Architecture

B. Exemplar-based Image Inpainting module:

In this video inpainting system, exemplar based image inpainting algorithm is used for inpainting [8]. Because this algorithm is capable of propagating both linear structure and two-dimensional texture into the target region with a single algorithm. It has the ability to preserve edge sharpness and balanced region filling to avoid over shooting artefacts.[8]

After frame extraction user can select the images from where frames saved. User can select one image at a time. Here user needs to be select the image frame of the video that contain the object which needs to be removed from the video. All frames needs to be in painted those are contains targeted object one by one.

After selecting image user can select the area of an image where object is placed. User can select the area simply selecting the pixels which are surrounding to the targeted object. If first pixel is the last pixel then it can be considered as selection process is completed and it can filled with green color. It can be considered as selected area of the targeted object. Now user can start his inpainting process for remove the object. User having two options for inpaint an image, first one is run means it can considered whole image as library from where it can search for best patch and second is the fast run where part of the image can considered for the best patch. After selecting one of them inpainting process will start and according to the selection of type of inpaint it takes time.

Finally, if selected area of the object completed then this message will displayed to the user for intimation purpose only.

C. Frame Merging module:

Initially, all inpainted and non-inpainted frames are collected. The JMF library .getProcessor class is used for defining the Frame rate of output video. The images are set into ImageBuffer. Single image is read one by one and store into empty buffervideo. After all frames are read into buffer video, set the frame rate defined earlier. Then output video file is created.

IV. EXPERIMENTAL RESULTS

We have used the ICDAR dataset for performance evaluation. The performance of video inpainting scheme is evaluated by calculating peak signal to noise ratio. If PSNR of inpainted video sequence is high, then it can be said that a high performance is achieved with this approach. PSNR is calculated as:

$$PSNR = 20 \cdot \log_{10} \left( \frac{(255)^2}{\sqrt{MSE}} \right)$$

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

Where,

MSE = mean squared error between the original pixels and the inpainted image pixels.

m = width of image.

n = height of image.

I(i,j) = ith , jth pixel of Input image 'I'.

K(i,j) = ith , jth pixel of Inpainted image 'K'.

The output of the implemented modules of the system is as follows:

A. Input video selection:

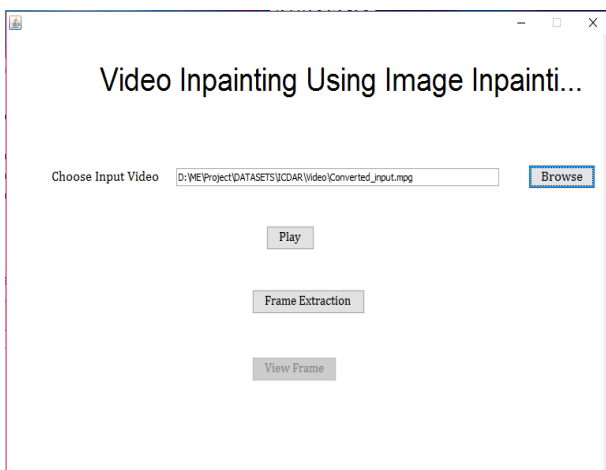


Fig. 3. Selected object

B. Frame extraction:

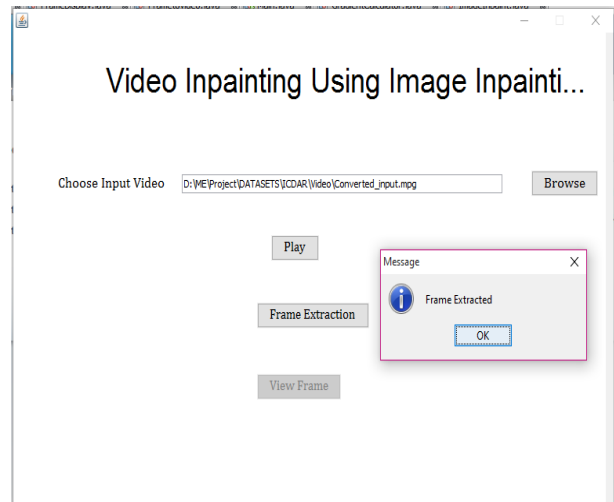


Fig. 4. Frame extraction

C. Input frame selection:

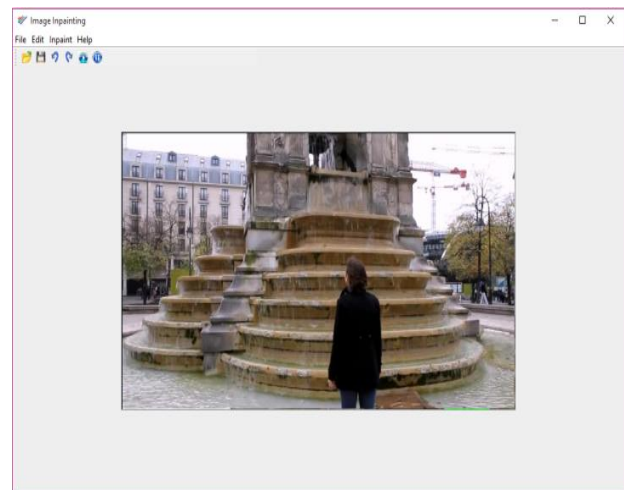


Fig. 5. After frame extraction a frame is selected

D. Object selection for Inpainting:

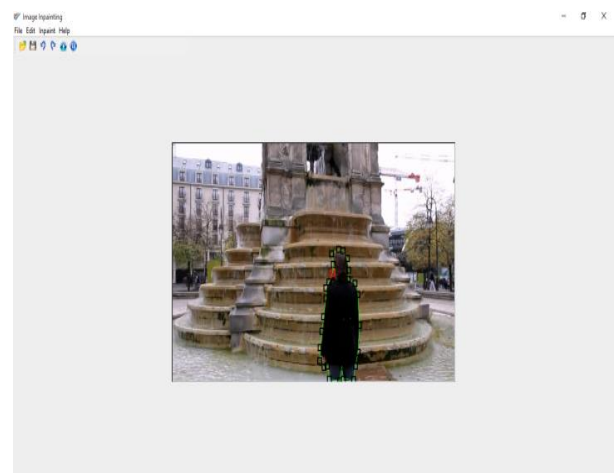


Fig. 6. Target object selection

E. Object Selection:



Fig. 7. Selected object

F. Output Inpainted frame:

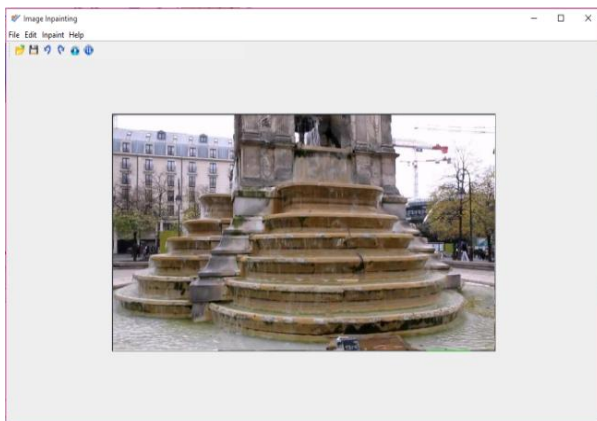


Fig. 8. Inpainted Frame

G. Creation of Inpainted video:

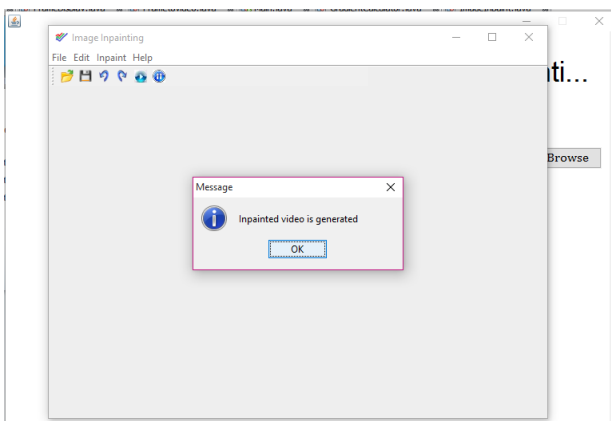


Fig. 9. Creation of Inpainted video

## V. CONCLUSION

The video inpainting system we have implemented attributed to high performance by resulting high peak signal to noise ration. As per experimental results obtained, we can conclude that this strategy can make

more proficient output than other existing procedures. It fills the missing region accurately to avoid over-shooting artefacts. It is able to synthesis the texture accurately. It uses the exemplar-based image inpainting algorithm to fills the gaps in the image. The major improvement over existing systems is that the given system does inpainting of the videos with underlying image inpainting technique and it can result in better and efficient output because we are combining multiple inpainted images to get the totally inpainted video. In future work, we have planned to test other methods for inpainting to bring more robustness to the method. We have planned to apply automatic object detection in frame which is to be removed. Object will be detected and removed automatically from all the frames

## ACKNOWLEDGMENT

I would like to take this opportunity to express our profound gratitude and deep regard to my Project Guide **Prof. D. B. Kshirsagar** sir, for his exemplary guidance, valuable feedback and constant encouragement throughout the duration of the work. His valuable suggestions were of immense help throughout my project work. His perceptive criticism kept me working to make this project in a much better way. Working under him was an extremely knowledgeable experience for me.

## REFERENCES

- [1] Olivier Le Meur, Mounira Ebdelli, and Christine Guillemot "Hierarchical Super-Resolution-Based Inpainting", IEEE, vol. 22, no. 10, Feb 2014
- [2] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting", 27th Annu. Conf. Comput. Graph. Interact. Tech. SIGGRAPH, pp. 417424, 2000.
- [3] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting", in Proc. IEEE Comput. Soc. Conf. CVPR, pp. 355362, Jan. 2001.
- [4] M. Ghoniem, Y. Chahir, and A. Elmoataz, "Geometric and texture inpainting based on discrete regularization on graphs", in Proc. IEEE ICIP, . pp. 13491352, Nov 2009.
- [5] K. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting of occluding and occluded objects", Proc. IEEE ICIP, pp. 6972, Sep. 2005.
- [6] K. A. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting under constrained camera motion", IEEE Trans. Image Process., vol. 16, no. 2, pp. 4545553, Feb. 2007.
- [7] T. K. Shih, N. C. Tang, W.-S. Yeh, T.-J. Chen, and W. Lee, "Video inpainting and implant via diversified temporal continuations", Proc. ACM Int. Conf. Multimedia, pp. 133136, 2006.
- [8] Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting", IEEE Trans. Image Process., vol. 13, no. 9, pp. 12001212, Sep. 2004.
- [9] T. K. Shih, N. C. Tang, W.-S. Yeh, T.-J. Chen, and W. Lee, "Video inpainting and implant via diversified temporal continuations", Proc. ACM Int. Conf. Multimedia, pp. 133136, 2006.
- [10] T. Shiratori, Y. Matsushita, X. Tang, and S. B. Kang, "Video completion by motion field transfer", Proc. IEEE Comput. Soc. Conf. CVPR, pp. 411418, Jun. 2006.
- [11] N. Tang, C.-T. Hsu, C.-W. Su, T. Shih, and H.-Y. Liao, "Video inpainting on digitized vintage films via maintaining spatiotemporal continuity", IEEE Trans. Multimedia, vol. 13, no. 4, pp. 602614, Aug. 2011.
- [12] Y. Zhang, J. Xiao, and M. Shah, "Motion layer based object removal in videos", Proc. 7th IEEE WACV/MOTIONS, pp. 516521, Jan. 2005.



- [13] X. Li and Y. Zheng, "Patch-based video processing: A variational Bayesian approach", IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 1, pp. 2740, Jan. 2009.

### **BIOGRAPHY**



**Dnyanesh S. Gaikwad** received the B.E. degree in Computer Engineering from University of Pune, Pune, Maharashtra, India in 2014. He is currently pursuing the M.E. degree in Computer Engineering with SRES Sanjivani College of Engineering, Kopergaon, Savitribai Phule Pune University, Pune, Maharashtra, India. His current research interests include Image Processing, Networking and cloud computing, Mobile and wireless communication/ Networking.