ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Video Inpainting for Moving Object Removal Using Patch Sparsity

Meenal R. Dusane[*1], Dr. G. S. Sable[2]
[*1] SPWEC (PG Student), Aurangabad, India
[2] Professor and Head(Electronics & Telecomm.Dept) SPWEC , Aurangabad, India
meer_dusane@yahoo.com

Abstract

The process of removing and reconstructing the specific area in video is known as video Inpainting. Most of the automatic techniques of video Inpainting are computationally intensive and unable to repair large holes. To overcome this problem, exemplar based video Inpainting method is extended by incorporating the Sparsity of natural image patches using background registration technique is proposed in this paper.

Keywords: Background registration, moving object removal, patch Sparsity, video Inpainting

Introduction

Video Inpainting refers to a field of Computer Vision that aims to remove objects or restore missing or tainted regions present in a video sequence by utilizing a technique of patch Sparsity to fill-in the missing parts of video sequence taken from a static camera using background registration method. The overriding objective is to generate an inpainted area that is merged seamlessly into the video so that visual coherence is maintained throughout and no distortion in the affected area is observable to the human eye when the video is played as a sequence. Inpainting technique is the modification of the images in an undetectable form.

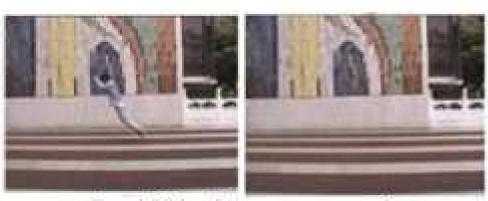


Fig. 1 Video Inpainting example

Video is considered to be the display of sequence of framed images. Normally twenty five frames per second are considered as a video. Less than twenty five frames per second will not be considered as a video since the display of those will appear as a flash of still image for the human eye. The main difference between the video and image inpainting methods using texture synthesis is in the size and characteristics of the region to be inpainted. For texture synthesis the region can be much larger with the main focus being the filling in of two-dimensional repeating patterns that have some associated stochasticity i.e. textures. In contrast, inpainting algorithms concentrate on the filling in of much smaller regions that are characterized by linear structures such as lines and object contours. Criminisi et al. [4] presented a single algorithm to work with both of these cases provided both textures and structures in images are present. Fig 1 shows the example that moving object is removed from images and inpainted.

Removing unwanted objects or artefacts from videos is a common task in professional video and movie productions. For instance, when filming in public locations, it is often necessary to remove walking people and other objects that accidentally occlude the scene. Objects may also have to be erased from a video sequence due to copyright issues. In other cases, the film crew needs to be in a scene for technical reasons, and needs to be removed in post-processing

Various researches are performed in video inpainting due to varied and important application of video inpainting. The main applications include undesired object removal such as removing unavoidable objects like birds or aeroplane that appear during filming, to censor an obscene gesture or action that is not deemed appropriate for the target audience, but for which it would be infeasible or expensive to re-shoot the scene. Another main application is the restoration of the videos that are damaged by scratches or dust spots or frames that may be corrupted or missing. When the video is transmitted over unreliable networks, there is a possibility of losing significant portions of video frames. To view the video again in its original form, it is necessary to repair these damaged scenes in a manner that is visually coherent to the viewer.

Bertalmio [1] is the first pioneer to video inpainting. The author repaired a video using image inpainting frame by frame. A user-provided mask specifies the portions of the input image to be retouched and the algorithm treats the input image as three separate channels (R, G and B). For each channel, it fills in the areas to be inpainted by propagating information from the outside of the masked region along level lines (isophotes). Isophote directions are obtained by computing at each pixel along the inpainting contour a discredited gradient vector (it gives the direction of largest spatial change) and by rotating the resulting vector by 90 degrees. This intends to propagate information while preserving edges. Wexler [6] defined an optimal function to search the best matching patch in a foreground and use the found patch to repair the moving foreground. This algorithm filled the video frame patch by patch. Space-time completion of video (Wexler, Shechtman & Irani,2007),Wexler et al. consider video inpainting as a global optimization problem, which inherently leads to slow running time due to its complexity. Finally, there is a very interesting paper by Patwardhan[8] et al. (Patwardhan, Sapiro, & Bertalmio, 2007), which proposes a pipeline for video inpainting. However, the pipeline is so simple that it failed to perform well in many cases. [9] zhang uses a motion layer segmentation algorithm to separate a video sequences to several layers according to the amount of motion. Each separate layer is completed by applying motion compensation and image completion algorithms. Except for the layer with objects to be removed, all of the remaining layers are combined in order to restore the final video. However, temporal consistency among inpainted areas between adjacent frames was not taken care of in [9]. Criminisi et al. [11] designed an examplar-based inpainting algorithm by prop- agating the known patches (i.e., examplars) into the missing patches gradually. To handle the missing region with composite textures and structures, patch priority is defined to encourage the filling-in of patches on the structure. Wu [12] proposed a cross-isophotes examplar-based inpainting algorithm, in which a cross-isophotes patch priority term was designed based on the analysis of anisotropic diffusion.

In this paper, video inpainting for static camera with a stationary background and moving foreground is considered in the spatial-temporal domain. First, the video is converted into image frames. Second, the edges are found by using SOBEL edge detection method. Next, the object to be removed is inpainted using novel examplar based image inpainting using patch sparsity. The known patch values are propagated into the missing region for every time frame of image to reproduce the original image. Last, the inpainted image frames are displayed to form the inpainted video. Here a video of short duration is considered for inpainting and the temporal domain information for each image frame is utilized to display the inpainted image frames as a video. In this paper, an efficient video inpainting algorithm that can remove moving object and reconstructing that part with visually plausible data is proposed. The rest of this paper is organized as follows. The section II discusses background Registration technique. Section III gives overview of examplar based image Inpainting. Section IV describes the proposed video Inpainting algorithm Finally, Section V concludes this paper

Background Registration

The background registration is divided into five major steps as shown in *Fig. 2*. The first step is to calculate the frame difference of mask by thresholding the difference between two consecutive input frames. Then, according to the frame difference mask of past several frames, pixels which are not moving for a long time are considered as reliable background in the background registration step. This step maintains an up-to-date background buffer as well as a background registration mask indicating whether the background information of a pixel is available or not. By the third step, the background difference mask is generated by comparing the current input image and the background image stored in the background buffer. This background difference mask is our primary information for object shape generation. In the fourth step, an initial object mask is constructed from the background difference mask and the frame difference mask. If the background registration mask indicates that the background information of an pixel is available, the background difference mask is used as the initial object mask. Otherwise, the value in the frame difference mask is copied to the object mask. The initial object mask generated in the fourth step has some noise regions because of irregular object motion and camera noise. Also, the boundary region may not be very smooth. In the last step, these noise regions are removed and the initial object mask is filtered to obtain the final object mask.

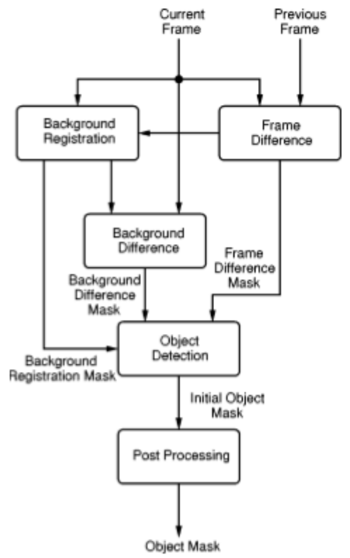


Fig. 2. Block diagram of video segmentation.

Examplar Based Image Inpainting

The Examplar based image inpainting is based on patch propagation. It is done automatically by the algorithm by inwardly propagating the image patches from the source region into the interior of the target region patch by patch. Patch selection and patch inpainting are the two basic steps of patch propagation which is to be iterated continuously till the inpainting process is complete

Let us first describe the terms used in inpainting literature.

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 **δΩ**

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.

It has been demonstrated for textures, repeating 2 dimensional patterns with some randomness. For Structure recreation algorithm, 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. The examplar based inpainting method combines the advantages of these two algorithms. Patch propagation is important in this type of inpainting, patch selection and patch inpainting are two basic step of patch propagation. Patch selection means which patch is to given highest priority to fill-in first. This best first filling strategy which is employed to fill the desired region is based on calculated patch priorities.

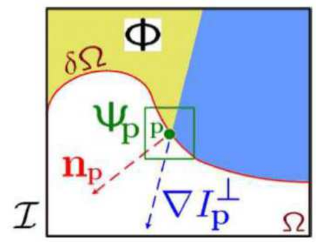


Fig 3. Ψ_p denotes the patch to be filled. n_p is the normal to the region contour δΩ of the target region Ω and $\Delta I_p^{\perp}$ is the isphote obtain at point p. I denote the entire image.

As the Fig.3 illustrates, the target region to be inpainted is denoted by Ω with its contour given by δΩ. This contour moves in an inward direction during the execution of the texture synthesis algorithm. A best first filling strategy is employed to fill the desired region based on calculated patch priorities. Two key ideas guide the calculation of these priorities:

1. Patches on the continuation of strong edges are preferred.

2. As well as those patches that are surrounded by high-confidence (in terms of their value) pixels.

Referring to Figure 3, given the patch Ψp, the priority P(p) of this patch is defined as:

$$P(p) = C(p)D(p)\,. \qquad (1)$$

Where in this equation, C(p) is the so-called confidence term and D(p) is the data term. Intuitively, the confidence term attempts to represent the “reliability” of the information surrounding the pixel p. Patches are deemed reliable if one or more of the following criterion are met:

1. More pixels are already filled in within the relevant patch.

2. Pixels in the patch were filled in early on in the execution of the algorithm.

3. The patch contains pixels that were not initially part of the target region.

In contrast, the data term attempts to give priority to the filling in of linear structures (e.g. people as opposed to textures) first, and this is quantified by measuring the strength of the isophotes (contours of equal luminance in an image) arriving at the contour $\delta\Omega$. Mathematically, these terms are defined as follows:

$$c(p) = \frac{\Sigma q \in \Psi p \cap (I - \Omega) c}{|\Psi p|} \quad (2)$$

$$D(p) = \frac{|\nabla I\, p \perp \cdot|}{\alpha} \quad (3)$$

Here $|\Psi_p|$ is the area of Ψ_p, α is a normalization constant, n_p is a unit vector orthogonal to the boundary $\delta\Omega$ at point p. A patch is associated with each point on the boundary. Given these equations, the algorithm proceeds to propagate the data from a source patch to a target patch, where the target patch is that patch having the highest cal- culated priority p using Equation 1. The source patch is found by applying a distance measure (specifically the sum of squared distances measure) to ascertain the similarity between patches from the source region and the tar- get patch. The patch in the source region having the most "similarity" to a patch in the target region is the patch selected for copying to the target patch. Once the copying has been performed, the confidence values of all those pixels in the intersection of the target region and the region enclosed by the patch are then set to the confidence value of the highest priority pixel. The algorithm then proceeds in this manner until the desired region is inpainted.

Proposed Algorithm

In this proposed method, below four major steps have been performed,

Step 1:- First video is converted into individual image frames, as Video is the display of the image frames in sequence

Step 2:- From each image frame the moving object is detected by background registration *method* and after detection that object is removed.

Step 3:- Next, the inpainting procedure using patch Sparsity is performed separately for each time frame of the image on removed object

Step 4:- Finally the inpainted image frames are displayed in a sequence, so that it appears as a video

Video is the display of the image frames in sequence. Image inpainting is done in spatial domain, whereas the video inpainting is performed in both spatial and temporal domain. It is used to remove objects or restore missing regions in the video sequence. Video inpainting may also be considered as the combination of frame by frame image inpainting. Generally all the natural images are composed of structures and textures. The primal sketches of an image like the edges, corners, etc are referred to as the structure of an image and the image regions with feature statistics or homogenous patterns including the flat patterns are referred to as the texture of an image. .

A. Patch Propagation

In our proposed algorithm, the examplar-based inpainting algorithm has been implemented through patch propagation. The two basic procedures of patch propagation are:

- Patch selection
- Patch inpainting.

In the patch selection, a patch on the missing region boundary with the highest priority is selected for further inpainting. .. Traditionally, the patch priority is defined based on the inner product between isophote direction and the normal direction of the missing region boundary. In the patch inpainting, the selected patch is inpainted by the candidate patches (i.e., exemplars) in the known region. The approach in [3] Criminisi's examplar-based algorithm, P. Perez, and K. Toyama utilizes the best match candidate patch to inpaint the selected patch. The approach Wong's[10] examplar-based algorithm uses a nonlocal means of the candidate patches for robust patch inpainting.

B. Patch Sparsity

To better address the problems of patch selection and patch inpainting, two novel concepts of patch sparsity of natural image, are proposed and applied to the examplar-based inpainting algorithm.

- Patch Structure Sparsity
- Patch Sparse Representation

C. Patch Structure Sparsity

We define a novel patch priority based on the sparseness of the patch's nonzero similarities to its neighboring patches. This sparseness is called structure sparsity. It is based on the observation that a patch on the structure has sparser nonzero similarities with its neighboring patches compared with the patch within a textured region. Compared with the priority defined on isophote, this definition can better distinguish the texture and structure, and be more robust to the orientation of the boundary of missing region.

D. Patch Sparse Representation

To inpaint a selected patch on the boundary of missing region, we use a sparse linear combination of exemplars to infer the patch in a framework of sparse representation. This linear combination of patches are regularized by the sparseness prior (regularization) on the combination coefficients. It means that only very few exemplars contribute to the linear combination of patches with nonzero coefficients. This representation is called patch sparse representation. The patch sparse representation is also constrained by the local patch consistency constraint. This model extends the patch diversity by linear combination and preserves texture without introducing smooth effect by sparseness assumption. In summary, the structure sparsity and patch sparse representation at the patch level constitute the patch sparsity. The patch structure sparsity is inspired by the recent progress

Fig .4 (a) & (b) shows the patch selection process which is used to select the patch with highest priority for further inpainting. The sparseness of the nonzero similarities of a patch to its neighboring patches is called as the structure Sparsity which is used for assigning patch priority. As shown in the Fig. 4(b), ψp and ψp' are the patches which are centered at pixel p and p' which lie in the edge structure and the flat texture region of the image respectively. The patch ψp has sparser nonzero similarities than patch ψp', so it is given larger patch priority. The patch on the boundary with the highest priority is first selected and inpainted.

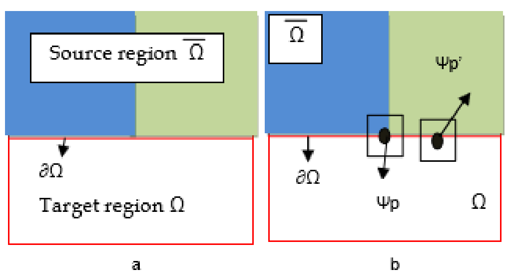


Fig.4 (a) shows that is the missing region is the known region, and is the fill-front of patch propagation Fig.4 (b) shows the two examples of the surrounding patch ψp and ψp' which are located at edge and flat texture region respectively

Fig. 5(a) & (b) shows the procedure of patch inpainting in which the selected patch on the boundary should be inpainted. The selected patch on the fill-front is the sparse linear combination of the patches in the source region regularized by sparseness prior. In this paper, a single best match examplar or a certain number of exemplars in the known region to infer the missing patch is not used.

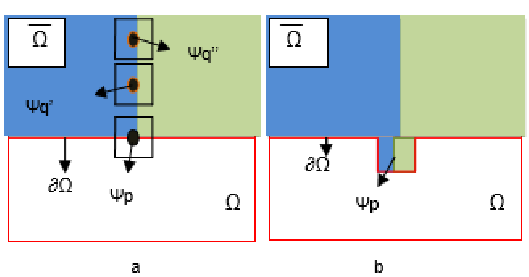


Fig.5 (a) shows that for the selected patch ψp , sparse linear combination of candidate patches { ψq', ψq''... ψq N} is used to infer the missing pixels in patch. Fig.5(b) Shows the best matching patch in the candidates set has been copied into the position occupied by ψp, thus achieving partial filling of Ω.

The most likely candidate matches for ψp lie along the boundary between the two textures in the source region, e.g., ψq' and ψq''. The best matching patch in the candidates set has been copied into the position occupied by ψp thus achieving partial filling of **δΩ.** The target region **Ω** has, now, shrunk and its front has assumed a different shape. Thus image inpainting is performed by the sparse linear combination of candidate patches weighted by coefficients, in which only very sparse nonzero elements exist. The above process is repeated until the missing region is completely filled by the known values of the neighboring patches

The sequence of display of all the individual inpainted image frames is referred as the video inpainting. Image inpainting process is iterated for all the image frames of the video.

In summary, our proposed video inpainting algorithm is given as follows.

1. Given input video is converted into number of image frames.
2. The moving object is tracked using background registration on all images, without intervention of user (see Section II).
3. Masking
4. Apply Inpaint to complete holes (see image inpainting in Section IV (A)-(D)).

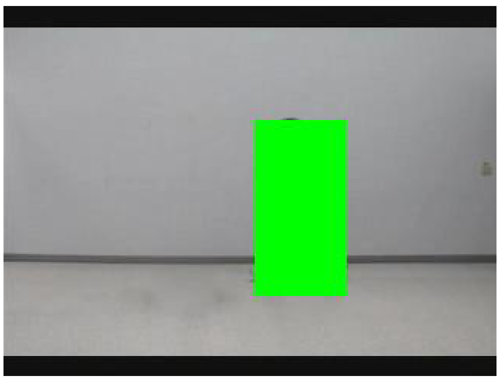
Experimental Results

A. Performance Analysis

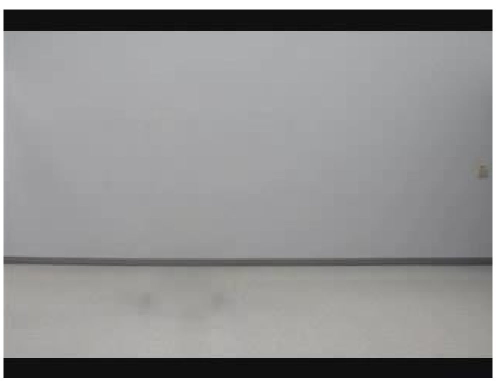
In this section, we do some experiments to evaluate the performance of our proposed method. We used several types of video examples for experiments. We apply our algorithm to the applications of moving object removal. In these examples, we compare our algorithm with two representative existing exemplar-based algorithms.

The proposed system is demonstrated on five videos including two real time videos. A first video sequence of 6.6 seconds run time is considered for inpainting and is converted into 93 image frames. Figure 6. (a) – (c) shows the procedure of inpainting on randomly selected image frame.

(a)



(b)



(c)

Figure 6.video inpainting (a)-source video frame (b)-Masking (c)-Inpainted video frame

The moving objects region from each image frame is detected by the Background Registration method. Once we obtained the detected objects we applied that frames to masking algorithm so that detected moving object region will be in green colour. It is shown in Figure 6.(b)

After we got masking on all 93 frames of first video, the inpainting algorithm performed for each image frames separately. A good definition of patch priority should be able to better distinguish the structures and textures, and also be robust to the orientation of the fill-front. In this paper, a novel definition of patch priority is proposed to meet these requirements. The result for inpainting algorithm is shown in figure 6.(c). The time required for completing masking and inpainting algorithm on all 93 frames is 45 min17 millisecond. Other examples of video inpainting are shown from Figures (7) – (10). Each video sequence example includes a source video masking and an inpainted video, illustrates the result on remaining four videos.

This system was implemented on an Intel Core Duo 4.0 GHz PC. We have tested the system on video sequences on different scenarios. Real time video sequences are used to demonstrate the knowledge discovery process i.e. object tracking and its removal using the proposed framework. All the videos chosen for moving object removal have same light intensity and have been taken during day time.

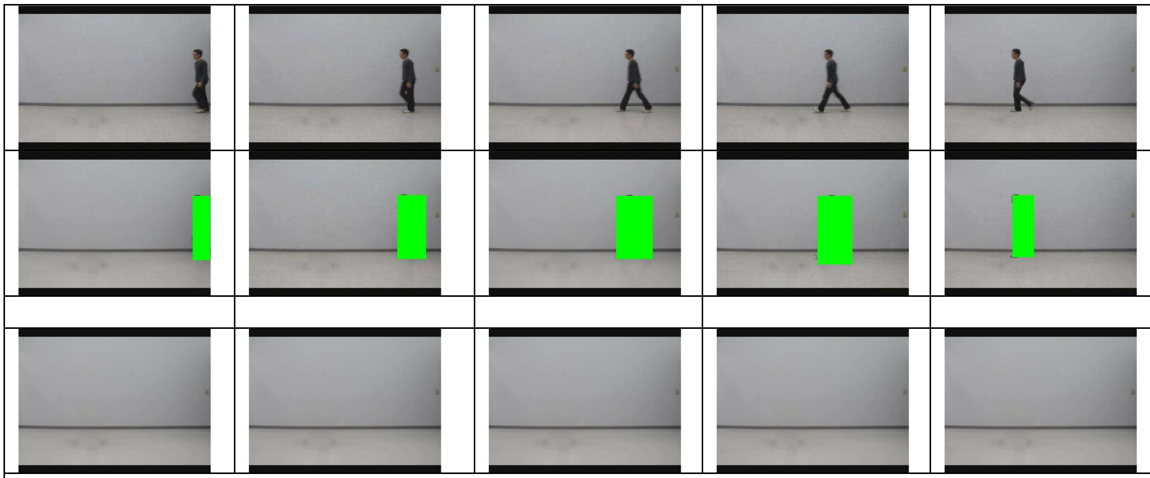
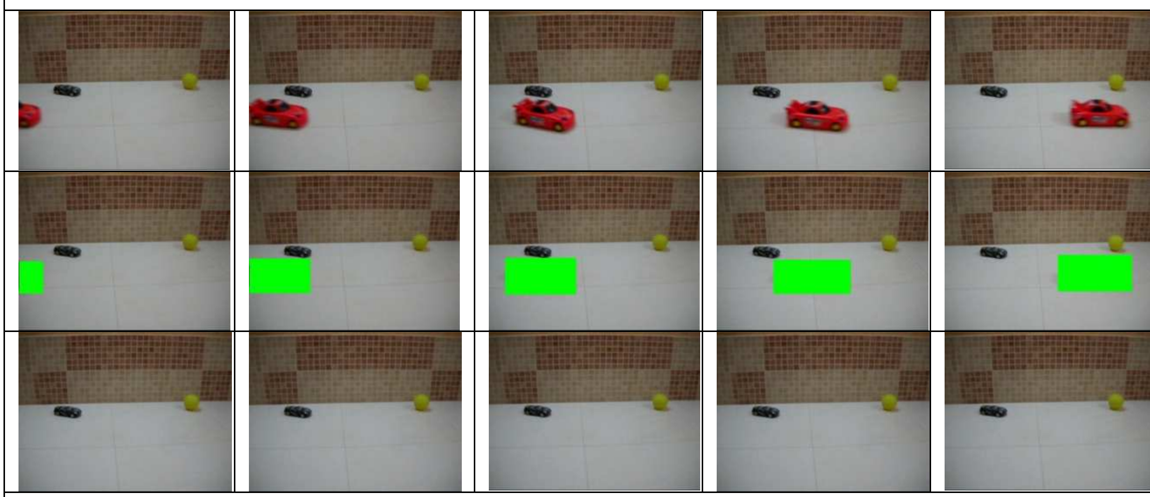
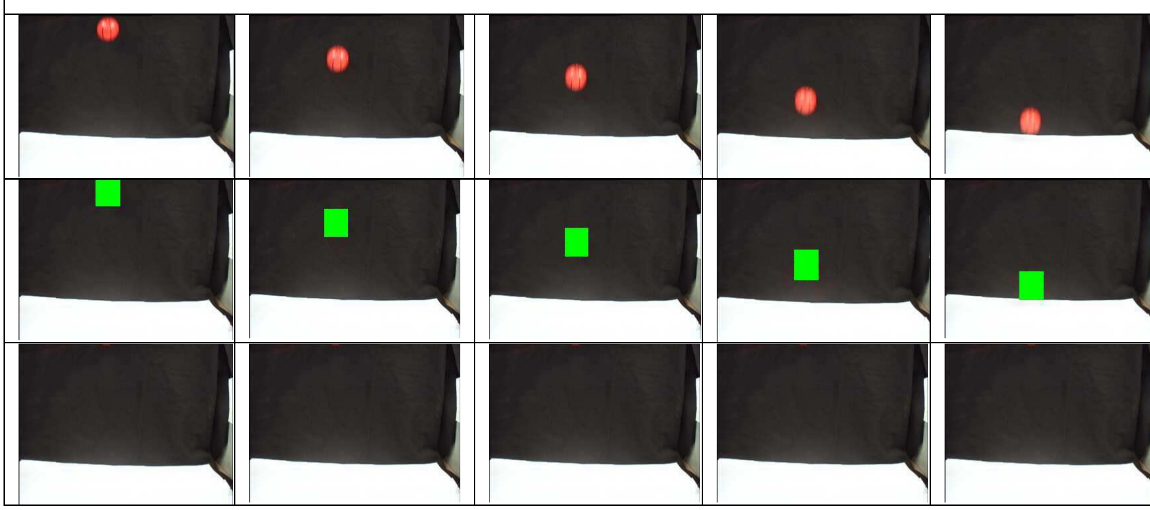
B. Comparison with existing methods

To validate the proposed video inpainting algorithm, the results are compared with some of the existing proposed frameworks. We qualitatively compare our approach with three related works [4], [13],[14] in Table I.

In general, all inpainting algorithms listed in Table I require the user to manually select the object to be removed. However, depending on the tracking or layer separation algorithm, the amount of human interaction varies. In [4] the technique presented was about region filling and object removal. The results using Criminisi's approach were not that promising whereas our algorithm achieved better results. The difference in the results occurred while searching for the best exemplar patch and user interaction in selecting the target region. In Criminisi's approach, nothing is described about which patch to select if we get two patches with same minimum error. During our implementation of Criminisi's algorithm, to overcome such cases, the confidence of the structure can be modelled for a patch by measuring the sparseness of its nonzero similarities to the neighboring patches and also to select the target region we used background registration method which avoids the user intervention.

In [13] it presented a technique of moving object removal from stationary and non stationary background under the constraint of camera motion. It used extended examplar method and the technique proposed in [13] efficiently ghost shadows, but the problem occurred are used tracking procedure does not take advantage of shadows due to real world light sources, Patch matching technology should need to improved as it does not maintains spatial –temporal continuity, the user interaction

ISSN: 2277-9655
Impact Factor: 1.852

	Video 1: Walking Man video Size: 352 X 240 Avg time: 0.44sec/frame
	Video 2: Running Car Size: 640 X 480 Avg time: 2.829 sec/frame
	Video 3: Bouncing ball Size: 320 X 240 Avg time: 0.334sec/frame

is required for deciding the region to be removed, and if multiple objects are to be removed then for each object user have to select region to be removed first, then applying inpainting procedure .therefore user has to run the program several times. As compared to this in our approach we used patch matching technique is patch

ISSN: 2277-9655
Impact Factor: 1.852

sparsity which is produces excellent results and also we required less time for detection ,masking and inpainting. As our algorithm automatically tracks he moving object so user interaction is totally have been removed.

In [14],the system is developed for fast and enhanced technique for image inpainting, which mainly give impact on time required for inpainting. This [14] paper had reduces time required for image inpainting as compared to its previous methods, but when we compared this system with our approach it is observed that fast algorithm first selects the target region and then it requires 4 sec to complete the inpainting for the 1693 damaged area(in pixels) from 60492 image size (in pixels),whereas in our approach the algorithm requires only 0.334 sec per fram

TABLE I

Algorithm	Image size (Px)	Area to be removed (Px)	Image inpainting time (Actual) (Sec.)	Calculated area (Px)	Calculated Image inpainting time	Human interaction for mask image	Auto Masking image accuracy
[2]	60492	1693	11.5	5905	40.11	Required	-
[14]	76800 (320 x 240)	1876	0.334	5905	1.05	Required	-
[31]	60492	1693	4	5905	13.95	Required	-
Our algorithm	76800 (320 x 240)	5905	0.334	5905	0.334	Not Required	100%

for the 5905 damaged pixels from 76800 (320X 240) image size ,and this time includes detection , masking and inpainting, It is observed that our approach requires very much less time without interaction of human, when compared with [4],[13],[14] approaches.

The TABLE I shows the comparison of our method With three previous approaches, in this table in fifth column we assumed the damaged area taken same for all previous algorithms and in next column we calculated time required for that same area ,then it is observed that our proposed approach require very lesser time as compared with other ones.

Conclusion

In this paper we proposed an efficient video inpainting algorithm by background registration using patch Sparsity. The presented system here describes a new examplar-based video inpainting method using patch Sparsity, which is implemented in MATLAB .In this paper, static camera with constant background is considered. It is mainly focused on the object removal in the image frames. A background registration technique is used to construct reliable background information from video, Then each incoming frame is compared with the background image If the luminance value of a pixel differ significantly from the background image, the pixel is marked as moving object; otherwise, the pixel is regarded as background. Finally, a post-processing step is used to remove noise regions and produce a more smooth shape boundary of desired or moving object. Once object is detected, Patch priority and patch representation which are the two major steps involved in the proposed examplar-based inpainting algorithm for an image is applied. Structure Sparsity was designed by measuring the sparseness of the patch similarities in the local neighborhood.

From the experiments, it is observed that the proposed extended examplar based image inpainting is effective compared to the basic examplar based image inpainting. We applied our algorithm to the applications for object removal and its completion. Also applied to real time videos which give sharp inpainted region. We

ISSN: 2277-9655
Impact Factor: 1.852

compare our algorithm with the previous region filling and object Removal by Exemplar-Based Image Inpainting, fast and enhance enhance algorithm for examplar-based, and Exemplar-Based Video Inpainting without Ghost Shadow Artefacts by Maintaining Temporal Continuity. From comparison we observed that we required very much less time for inpainting approximately three times lesser time than previous methods, also we removed the user intervention making it completely automatic.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in Proc. SIGGRAPH, 2000, pp. 417–424.

[2] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier–Strokes, fluid dynamics, and image and video inpainting," in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, pp. 417–424, 2001.

[3] B.Vidhya.,S.Valarmath.,"Novel Video Inpainting Using Patch Sparsity" IEEE International Conference on Recent Trends in Information Technology, pp. 810- 814, 2011

[4] Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by examplar-based image inpainting," IEEE Trans. Image Process., vol. 13, pp. 1200–1212, 2004.

[5] Z. Xu and J.Sun "Image Inpainting by Patch Propagation Using Patch Sparsity"in proc. IEEE transactions on image processing, vol. 19, no. 5, may 2010,pp.1153-1165.

[6] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," Proceedings. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, 2004

[7] J. Jia and C. K. Tang, "Image repairing: Robust image synthesis by adaptive and tensor voting," in Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recogition, pp. 643–650, 2003.

[8] K.A. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting of occluding and occluded objects", in Proc. ICIP 2005.Vol. II, pp. 69-72.

[9] Y. Zhang, J. Xiao, and M. Shah, "Motion layer based object removal in videos," 2005 Workshop on Applications of Computer Vision, 2005.

[10] S.-Y. Chien,S.-Y. Ma, and L.-G. Chen",Efficient Moving Object Segmentation Algorithm Using Background Registration Technique" Proceedings 2002 IEEE transactions on circuits and systems for video technology, vol. 12, no.7, pp. 577-586, JULY 2002

[11] Wong and J. Orchard, "A nonlocal-means approach to examplar based inpainting," presented at the IEEE Int. Conf. Image Processing, 2008.

[12] P.Dhabekar, G.Salunke "The Examplar-based Image Inpainting algorithm through Patch Propagation" procc in International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-1, Issue-4, October 2012

[13] T. K. Shih, N. C. Tang, and J.-N. Hwang, "Exemplar- based video inpainting without ghost shadow artifacts by maintaining temporal continuity,"IEEE Transactions on Circuits and Systems for Video Technology, vol. 19, no. 3, pp.347–360, Mar. 2009.

[14] Anupam, Goyal, P., & Diwakar, S. (2010)" Fast and Enhanced Algorithm for Exemplar Based Image Inpainting" Fourth Pacific-Rim Symposium on Image and Video Technology, (pp. 325-330).