Journal of Engineering and Applied Sciences 12 (16): 4113-4117, 2017
ISSN: 1816-949X
© Medwell Journals, 2017

Inserting Virtual Static Object into Appropriate Region of Video

Israa Hadi Ali and Roa'a M. Al-Airaji
Information Technology College, University of Babylon, Babylon, Iraq

Abstract: This study presents a methodology to insert virtual static object into video to enrich a video environment with virtual information and make virtual video objects appear as it are coexist in the same environment. Several problems need to be overcome are: occlusions of video objects by virtual object and vice versa, the illumination distribution of the video environment is extraction in order to render the virtual object with this illumination model and the rigid registrations of virtual object into the video environment. The proposed method applies the following steps: rebuild background model to isolate background from the foreground then subtract original video frames from the background to get foreground objects. After that, apply background segmentation to segment background regions. Then extract features for each segment, virtual object and select appropriate segment (region) for virtual object. The final step is to insert virtual static object into video in selected region and create new video.

Key words: Background model, virtual object, video editing, chain code, video tracking, Iraq

INTRODUCTION

The seamless integration of virtual objects into dynamic objects captured by video is a challenging problemin computer graphics (zhang *et al*., 2011). The main subtasks are estimate positions of camera andseamless blending of video. Estimate positions of camera is to find positions and orientations of the target object, there are two type of methods to estimate positions of camera are: marker-based methods that used white and black marker to detect and track of the marker region and motion-based methods that track many feature pointsand recover camera motion parameters from the point correspondences. Video blending tasks arerelating to the overlay of virtual objects, replacing dynamic objects with virtual objects or removing dynamic objects (Park *et al*., 2005).

In recent years, mixed objects have been widely studied and attract a great deal of attention in media industries such as film making, image/video editing, television broadcasting, advertising, virtual reality, augmented reality, production of images and animated sequences for cinema (Sato, 2005; Da Silva *et al*., 2013). However, there is problem in inserting virtual objects into video comparing to embedding objects into images, the main challenges exist in inserting virtual objects into video are acquiring the motion of videoobjects, ensuring the behavior consistency between the virtual and video objects and correctly presenting the occlusion between them without the 3D information of the video objects (Idris *et al*., 2010; Bui *et al*., 2016). To enhance the quality of new video that produced from the mixed of virtual and video objects, there are three aspects taken into account (Abad *et al*., 2003):

- Consistency of geometry, virtual static objects must be located at a desired and correct location in the video scene
- Consistency of illumination, the virtual object must cast a correct shadow and match to that of other objects in the scene
- Consistency of time, moving of virtual and video objects has to be correctly coordinated

Literature review: James M. Rehg and Sing Bing Kang in 2000 presented method to video editing, they segment video into multiple layers are foreground and background then segment and tracking foreground video. Then composition of the layers based on image-based rendering (Rehg *et al*., 2000).

Zhiguo Ren and Wenjing Gai insert virtual pedestrians into pedestrian real video. The researcher's studies technique to mix virtual pedestrians and video of real pedestrian groups. They proposed an approach to efficient dynamic path planning consistent to the motion path of pedestrians in real video then the estimate distribution of pedestrians, lastly insert virtual characters into real video after processing every frame in video (Ren *et al*., 2013).

Israa Hadi and Adil Abbas develop idea of accumulation histogram methodto background estimationand detect moving objects trajectories in video.

Corresponding Author: Israa Hadi Ali, Information Technology College, University of Babylon, Babylon, Iraq

The researchers development idea of accumulation histogram by divide each frame in video to blocks with the same size and find mean value for each block then using mean value instead of image pixels to find background, they found this method efficiency in terms of time and gives accurate results. After that, they apply segmentation and tracking to find trajectory of foreground objects (Israa and Adil, 2016).

Mohamed Sakkari used hand as support to insert virtual object inaugmented reality. The researcher based on the real component of the scene without introducing a marker, he used hand using hand as support to insert virtual object. The researcher first extracts skin regions then identify and detect the hand by neuron network.

MATERIALS AND METHODS

Proposed method: Figure 1 explains work steps of the proposed method. The proposed method consists of many steps as follows:

Video file: In this step, we open AVI structure for the video file and get list movie (still images/sequence frames) then get information such as total of frame, width of the frame and height of the frame from AVI header then split the video into frames (still image).

Re-build background model: Background model is an efficient way to get background of the video and obtain foreground objects. There are several methods to get background of the video this study improved method of accumulation histogram. The general color histogram shows representation of colors distribution in image, it is feature to find occurring probability of different colors in the image. In general color histogram the x-axis represent color value and y-axis represent color occurring frequency in the image while in accumulation histogram the x-axis represent color value and y-axis represent color accumulation occurring frequency in the image.

This study improved accumulation histogram by dividing image into blocks and calculate median of colors for each block. The improved accumulation histogram work as follow; take certain number of frames (specific period of the video) and divide each frame into K blocks, each with U.V size. For each frame, calculate median value for the first block and save these values in a temporally matrix.

Calculate frequent of median values that saved in a temporally matrix and find the most frequent of median value, the most frequent of median value represent number of frames that contain almost same color values in the first block. After that, the first block in the background model is reconstruct in which values of blocks with the most frequent median value represent values of the first block in the background model. Advance to the next block and repeat Steps 2-4. These steps continue until reach to the last block.

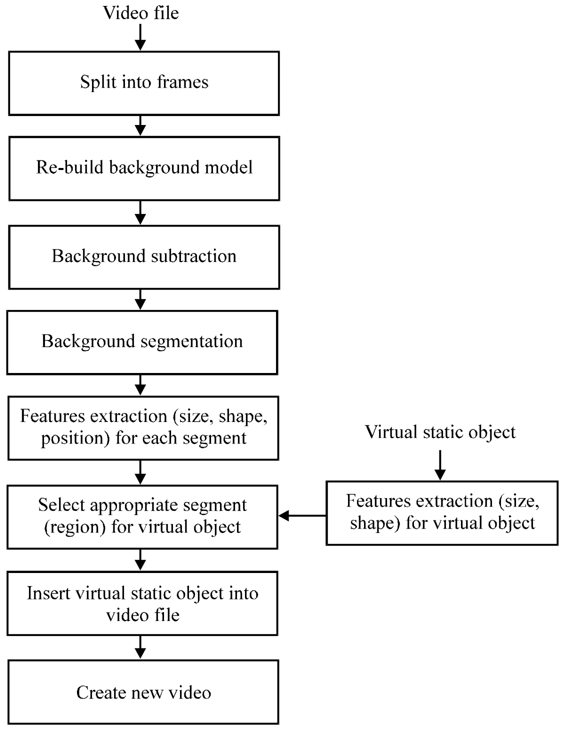
Video file
Split into frames
Re-build background model
Background subtraction
Background segmentation
Features extraction (size, shape, position) for each segment
Virtual static object
Features extraction (size, shape) for virtual object
Select appropriate segment (region) for virtual object
Insert virtual static object into video file
Create new video

Fig. 1: Block diagram of the proposed method

Background subtraction: The main step in the process of inserting virtual object into video objects is to detect and track moving video objects, this study using background subtraction to detect and track moving objects instead of traditionally tracking methods. Background subtractionis an approach for identifying the moving objects in a video sequence from background. The simplest method to perform background subtraction is to compare frames obtained at time t with the background obtained from the previous step:

$$\left|F_t(x) - B(x)\right| > T$$

Where:

x = The pixel position intensity
F_t = Represent current frame at time t
B = Background model
T = The threshold value

Pixel value is background pixelif the difference between pixel in current frame at time t and corresponding background pixel equal zero otherwise it is foreground pixel. However, pixel value may be background pixel if the difference slightly greater than zero because lighting change so threshold value is used and it is small value between 5 and 15. After applying background subtraction method on all frames in video the output is new video with 0's value for background and 1's value for the foreground.

Background segmentation: This step divides to two stages.

First stage: The first stage is creating an accumulation image from the binary video that we obtained from the previous step. The accumulation image contains trajectories of moving objects and regions that the moving objects not move on it. Accumulation image saved in matrix of 0's value for the regions that the moving objects not move on it and 1's value for the trajectories of moving objects. Procedure bellow explains how we obtain on the accumulation image, the accumulation image started by zero values and update iteratively from first frame until reach to the last frame in the binary video. If $F_t(x) = 0$ and accumulation image (x) = 0 then accumulation image (x) = 0, else accumulation image (x) = 1

Second stage: Applying segmentation on the accumulation image that we obtained from the first stage, in this study region-growing segmentation is used to segment regions that the objects not move on it. The region grown method start with number of seed points then the regions from these seed points are growing by allocate those neighboring pixels that have similar features with the seed point to the respective region, the features that used to measure the similarity are texture, color, intensity and gray level.

Features extraction for each segment: Extract features for each segment that we obtained from the previous step, this step necessary to select appropriate segment to locate virtual object in it, the features need to be calculate are: size, shape, position. Size of segment meant total number of pixels for that segment.The size of segment can be compute using equation bellow:

$$\text{Size} = \sum_{x}\sum_{y} S\,(x, y)$$

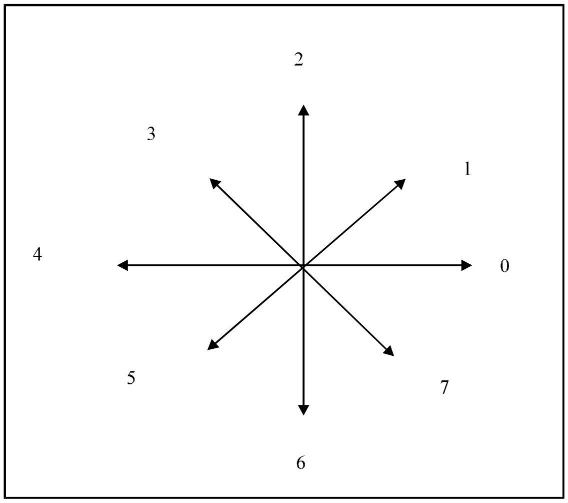

Fig. 2: 8-connected numbering scheme

where, S(x, y) refers to the pixel segment. Shape of segment meant represent the region by describing its ocean. This can be doing using chain code approach; chain code is approach torepresent the boundary of an object or region by linked series of straight-line segments of specified length and direction. Chain code work as follow:

For each such region, point anywhere on the region boundary is selecting.then from this selected point traverse the region in clockwise manner. As the boundary is traverse, numbering scheme used to specific the direction of each chain segments. Points of boundary that we traversed saved in an array or list. This step is continuing until reach the end pixel. Figure 2 explains 8-connected neighborhoodnumbering schemeto specific directions of each chain segment.

Features extraction for virtual object: Extract features such as size and shape for virtual object.

Select appropriate segment (region) for virtual object: Based on features that we extracted for virtual object and the features of each region, we select region with size and shape appropriate to size and shape of virtual object and in appropriate position.

Insert virtual static object into video file: After selected appropriate region in video, the last step is embedding (insert) virtual object into video, the method of the insertingis semi-automated, center of the virtual static objectis calculated and position in the appropriate region is selecting. Then pixel in the center of the virtual object is located in the selected position of the appropriate region. After that pixels around center of the virtual object are located in the appropriate positions and this positions are determine based on the center of the virtual object as follow.

Virtual static object:

```
If (V_Row< Xc)
   R_Row = (X-(Xc–V_Row))
  Else
If (V_Row >Xc)
R_Row = (X+ (Xc –V_Row))

If (V_Col<Yc)
     R_Col = (Y-(Yc –V_ Col))
Else
If (V_ Col >Yc)
   R_ Col = (Y+ (Yc –V_ Col))
```

Where Xc, Yc is center position of the virtual object. V_Row, V_Col is position of the current pixel of the virtual object. X, Y is the selected position of the appropriate region. R_Row, R_Col is the appropriate position in the appropriate region to insert current pixel of the virtual object in it.

After that, the user decides if the object inserted in the desire place or not if not selects other positionin appropriate region and so on until virtual object inserting in the desire place. After that create new video contain virtual and original video objects.

RESULTS AND DISCUSSION

Example Fig. 3-10 explains results of the proposed method. After features extraction for each region and for the virtual object, the region with size and shape appropriate to the virtual object is the region with the red boundary that explains in Fig. 9.

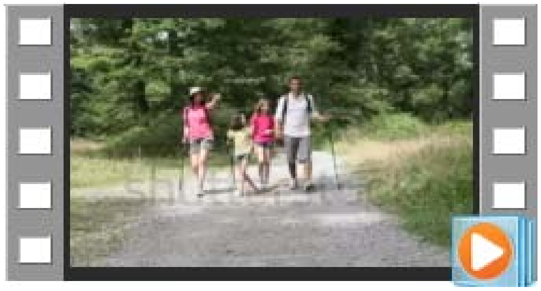

Fig. 3: Original video (pedestrians in road)

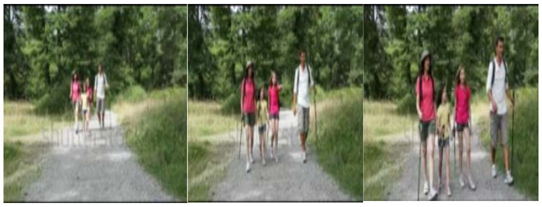

Fig. 4: Samples of video frames

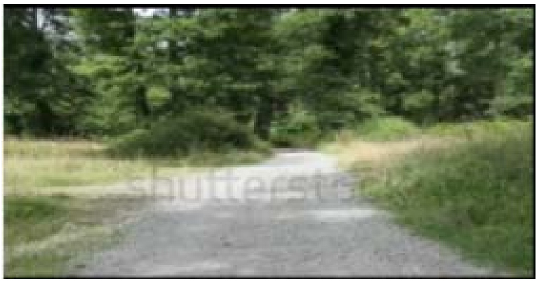

Fig. 5: Video background

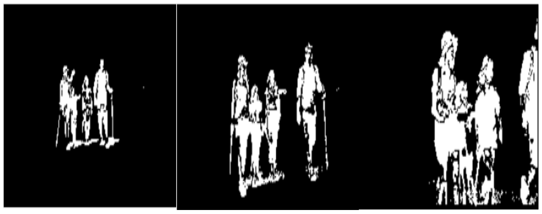

Fig. 6: Samples of binary video frames after background subtraction

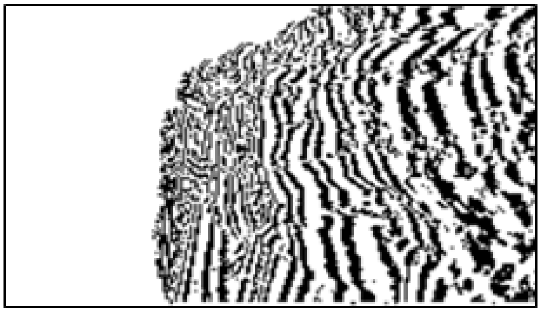

Fig. 7: Accumulation image (white regions are regions in video where the moving objects not move on it and black regions are trajectories of the moving objects)

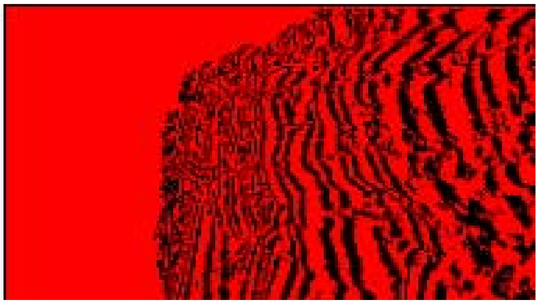

Fig. 8: Accumulation image segmentation (red regions are the segment regions)

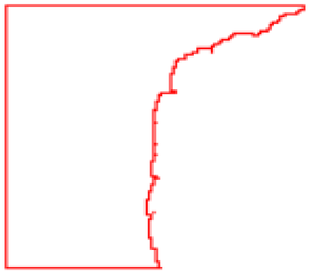

Fig. 9: Appropriate region

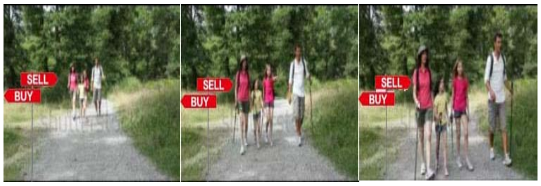

Fig. 10: Samples of new video frames after inserting virtual object (guidance signal) into video

CONCLUSION

This study present method for integrating virtual objects into video, the proposed method can achieve a high level of realism and can used inany application that requires a realistic virtual object occlusion simulation such as design analyses, games and cinema. The cinema has been the driving force of this kind of application. The results demonstrate that the virtual object in the produced video appear as it is coexisting and interact normally with the original video objects in the same space and time.

REFERENCES

Abad, F., E. Camahort and R. Vivo, 2003. Integrating synthetic objects into real scenes. Comput. Graphics, 27: 5-17.

Bui, G., B. Morago, T. Le, K. Karsch and Z. Lu et al., 2016. Integrating videos with LIDAR scans for virtual reality. Proceedings of the 2016 IEEE Conference on Virtual Reality (VR), March 19-23, 2016, IEEE, Greenville, South Carolina, ISBN:978-1-5090-0836-0, pp: 161-162.

Da Silva, M.L.M., F.A. Caetano and R.L.D.S. Da Silva, 2013. A Semi-automatic approach to mix real and virtual objects for an augmented reality game. Proceedings of the SB Games 2013 Workshop on Virtual Augmented Reality and Games, October 16-18, 2013, Southern Baptist Convention, Sao Paulo, Brazil, pp: 1-4.

Idris, W.M.R.W., Y.M. Saman, A. Aziz and A.S.M. Noor, 2010. Managing virtual environments with tree-structured, Hierarchy-embedded virtual objects. Int. J. Comput. Sci. Network Secur., 10: 115-125.

Israa, H. and A.M. Adil, 2016. A method for background establishment using accumulate histogram for detection of object trajectory for video tracking applications. Q. Adjudicated J. Nat. Eng. Res. Stud., 3: 55-65.

Park, J.S., M. Sung and S.R. Noh, 2005. Virtual object placement in video for augmented reality. Proceeding of the 6th Pacific Rim Conference on Advances in Multimedia Information Processing 2005, November 13-16, 2005, Springer, Jeju Island, Korea, pp: 13-24.

Rehg, J.M., S.B. Kang and T.J. Cham, 2000. Video editing using figure tracking and image-based rendering. Proceedings of the 2000 International Conference on Image Processing Vol. 2, September 10-13, 2000, IEEE, Vancouver, British Columbia, Canada, ISBN: 0-7803-6297-7, pp: 17-20.

Ren, Z., W. Gai, F. Zhong, J. Pettre and Q. Peng, 2013. Inserting virtual pedestrians into pedestrian groups video with behavior consistency. Visual Comput., 29: 927-936.

Sato, I., M. Hayashida, F. Kai, Y. Sato and K. Ikeuchi, 2005. Fast image synthesis of virtual objects in a real scene with natural shadings. Syst. Comput. Jpn., 36: 102-111.

Zhang, Y., J. Pettre, J. Ondrej, X. Qin and Q. Peng *et al*., 2011. Online inserting virtual characters into dynamic video scenes. Comput. Anim. Virtual Worlds, 22: 499-510.