

A NEW PROCEDURE FOR 2D TO 3D CONVERSION USING DEPTH ARRAY CONVERSION

M. Kavitha and E. Kannan

Department of CSE, Vel Tech University, Chennai, India

Received 2013-12-13; Revised 2014-01-07; Accepted 2014-02-10

ABSTRACT

Entertainment in the real world becomes a popular era and it is long lasting for several decades. The major advancement in television as well as in projection leads to highest quality in terms of realistic pictures, images and movies. Various researchers and research forums are under the development of stereoscopic conversion of 2D program material into 3D realistic. Due to less availability of 3D videoscopic displays, we proposed a new methodology to convert 2D array of video sequence into stereoscopic 3D. The proposed methodology focuses on depth array conversion of back grounds from the 2D video and subtracting the active objects in each scenes at every frames, this yields to obtain the depth information from the video which requires heavy computation in real time, which was achieved by Cryengine with AMD QUADCORE A6 CPU, as well with Radeon graphics. Experimental results shows that 3D recognition rate of a single image in the 2D video sequence is up to 78% and display rate of the particular sequence are at 29 frames per second. Experimental results has been included and 3D display rate for the converted sequence is up to 92% accuracy and frame rates be 32 frames per second for FULL HD (1080 p).

Keywords: 2D to 3D Conversion, Depth Array Conversion, FULL HD, 1080p, Cryengine, Quad Core, AMD, A6 CPU, Stereoscopic, 2D Array, 3D Array

1. INTRODUCTION

In real world, 3D images are emerging in many applications such as Multimedia, Augmented reality, virtual reality, 3D-TV, 3D movies. Now a day peoples are paying more attention towards 3D because of its realistic results (Tam and Zhang, 2006). Hence it is difficult to capture the 3D videos around, since 3D video capturing device is very complex in carrying and installing it. Recent 3D videos are evolving based on conversion technologies. Preparing an instance 3D evolving in the conversion of 2D to 3D, where the 2D videos are converted into 3D sequence (Fergus *et al.*, 2003; Fritz *et al.*, 2004; Park *et al.*, 2007; Veltkamp and Tanase, 2001; Abdul-Nafiu, 2013; Zhang *et al.*, 2011; Al-Marshadi, 2011); end user final output will be the 3D sequence.

In this study we are going to evolve a new technique for converting 2D array into 3D array using Depth array conversion. **Table 1** represents the different supportive 2D colour models available with supportive formats.

Table 1 represents the different supportive 2D colour models available with supportive formats.

1.1. DIBR Method

In this method, initially depth information of the pixel is coded along with the pixel intensity, which is fully achieved using Luminance Intensity. Luminance Intensity is used to code the depth information of each pixels in the auxiliary images, where closer distance holds the lighter values and farther distance holds the darken values:

$$x_{\text{left}} = x_c + \frac{t}{2} \frac{f}{Z}$$
$$x_{\text{right}} = x_c - \frac{t}{2} \frac{f}{Z}$$

Since the equation denotes the left eye and right eye images perspective on virtual camera positioned at C_l and C_r which was clearly denoted in the **Fig. 1**.

Corresponding Author: M.Kavitha, Department of CSE, Vel Tech University, Chennai, India

Table 1. Supportive 2D Colour model along with their supportive formats

2D			
Colour model	Accuracy	Jitter	Format
RGB	High	Yes	.avi,.vob,.avi
CMYK	High	Yes	.avi
HSV	High	Yes	.MPEG

Table 2. Supportive 3D Colour model along with their supportive formats

3D			
Colour model	Accuracy	Jitter	Format
RGB	High	Yes	.mkv,.avi
CMYK	High	No	.mkv
HSV	Low	Yes	.MPEG

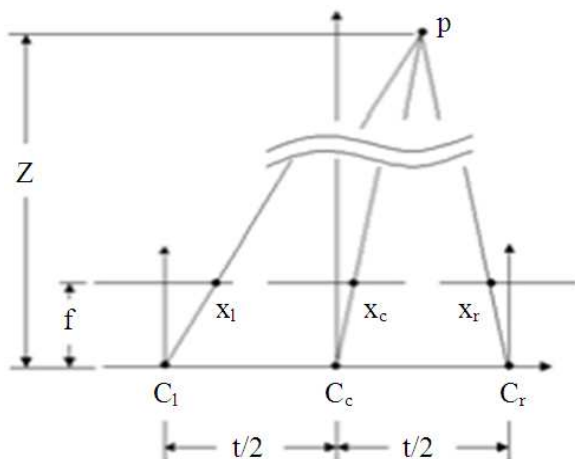


Fig. 1. Denotes the DIBR-virtual stereoscopic images (Zhang and Tam, 2005; Tam and Zhang, 2006)

1.2. 2D to 3D Rendering

Basically, 2D images are 2D arrays which are all transformed in the form of rows and columns according to the pixel grid (size of the image). The various format of images are with the extension .JPEG, .JPG, .PNG, .GIF, .BMP, .TIFF where JPEG images hold a vital role in lossless compression when applying any transforms. Since these kinds of 2D arrays do not produce any realistic outperformance, hence 3D rendering is achieved in order to obtain realistic accuracy in stereoscopic images (Table 2).

The basic principle about 2D to 3D conversion lies on the proof of stereoscopic viewing with human visual-binocular processing of two variant dissimilar images. Actual perception deals with the minor subtraction of disparities between left eye view and right eye view, later it was transformed to achieve different depths of objects outside 2D plane. Hence the fundamental 2D to 3D conversion methodology involves horizontal shifting of pixels in order to create new images (Tam and Zhang, 2006). Horizontal shift is achieved on the distance of feature object at the stereoscopic view at that corresponding pixel (Tam and Zhang, 2006). Table 3 represents the supportive colour model for converted 3D sequence with their supportive formats.

Table 3. Supportive Colour model for converted 3D sequence with their supportive formats

3D Converted sequence			
Colour model	Accuracy	Jitter	Format
RGB	High	Yes	.avi,.MPEG,.MPEGII,.vob,
CMYK	Low	Yes	.mkv,.FLV,.mpg,.mp4,.wmv
HSV	Low	Yes	

1.3. Video Classification

Video classification is done before any conversion techniques, it is necessary to categorize and classify a video sequence in order to improve the accuracy rate and computational speed. Initially the motion activities of all similar frames depth cues are estimated. After these all the frames are splitted, content complexity is analysed and classified as follows:

- Large frame difference
- Large frame difference with high jitter
- Large frame difference with low jitter
- Large frame difference with high background complexity
- Large frame difference with low background complexity
- Large frame difference with high background complexity along with high jitter
- Large frame difference with high background complexity along with Low jitter
- Low frame difference
- Low frame difference with high jitter
- Low frame difference with low jitter
- Low frame difference with high background complexity
- Low frame difference with low background complexity
- Low frame difference with high background complexity along with high jitter
- Low frame difference with high background complexity along with Low jitter

At each level texture pitch is calculated in order to solve the content complexity. Each frame difference is captured at a periodic interval and defined as:

$$X^d = \frac{1}{M_H M_W M_S} \sum_x \sum_y \sum_t V(I(x, y, t) - I(x, y, t-1) - T_{d_i})$$

Where:

$V(*.*)$ = Denotes frame function

M_H and M_W = Denotes the height and width of the frames

T_{d_i} = Denotes the threshold rate of the frames

1.4. Parameter Evaluation

Parameter evaluation is major term in conversion techniques. It is one of the challenges which are occurring with high faulty conditions with high rate of jitters. Basic parameters are cue, saturation, depth perspersion, active object, Texture pitch (gradient), motion evaluation (parallax).

1.5. Working Methodology

1.5.1. Depth Array Conversion

Finding the prominent area of finding depth of a 2D image is having a vast advantage of research. The depth of an image is obtained from the focal length of the camera and perspersion based on the view towards the focus, which states that the actual depth obtained from the focal length of the camera is based on the modelling perspersion towards the active object. Here we use inverse filtering to obtain the depth of focus of an object; again we are applying the defocus operator in order to achieve the depth information and focal length/distance of the depth. Local region parameters are estimated using local cue and depth perspersion parameter with multi resolution wavelet analysis where the high frequency region will be subjected to non-zero wavelet transformation and analysis and low frequency region will be of large number.

Local region containing large textures in back ground surface and high cut off edges can be predicted using luminance intensity values, however estimation through this methodology is only supportive for RGB colour model, since HSV and CMYK will not support this type of sequences. Texture pitch parameters are used to analyse the texture gradient and its intensity, since this could leads to stereoscopic view of a particular active objects. We can analyse the depth information based on geometric perspersion, here single input image is

classified, based on the classification the depth information is supposed to manage:

$$d^{\text{background}}(x, y, t) = 255 * (y - y_{vp}) / (N_H - y_{vp})$$

With all of these parameters we can locate the Depth array of a 2D array and applying Depth information and DIBR we can generate 3D array. Object segmentation is used to extract foreground active objects from the 2D images by applying morphological operation.

1.6. Procedure for Applying Depth Array Conversion

Pseudo 2d to 3d 1080p (unsigned char* depth, 2D, filter, DIBR, transform, frgd, bkgd, view)

Input: 2D, filter, DIBR, Depth, transform

Output: 3D array, Frames, stereoscopic view

Class 3D

3D 3dobj = new 3D (inputreader);

Dilate (3d);

Erode (3d);

Img.growth (3d);

Img.Boundary (local, global, perception);

Imgout = Img.Transform (3d, DWT);

Img.Thres (Imgout);

Frgd = thres (Imgout, bkgd);

Grad = Gradient (frgd);

Img.write (grad);

End

Class Morphology

Object1 = Classify (grad, frgd);

ApplyLI(object1);

// li--- luminance intensity

Lere = Convert(grad→view(lefteye, righteye))

DepthfocusImg = DIBR(lere);

Depthinfo = Depthfocus (DepthfocusImg);

Out = convert (Depthinfo→stereoscopic);

3D = Frame (out);

Display (3D);

End

In order to obtain depth information process the input images of 2D sequence. Fix the depth towards the centre of focus and eliminate vanishing focal points. Apply modelling perspersion towards the object and inverse filtering to obtain the depth information. Then find local region and depth perspersion. Finally apply multi resolution wavelet transform and get the high frequency region and low frequency region.

In order to obtain texture for foreground and background get the local region and back ground surface region, apply cut off threshold for getting edges and subtract the foreground from the back ground. Apply texture pitch to obtain gradient and generate stereoscopic view to predict active objects. Then apply morphological operation, analyse depth information from the texture and classify the input.

Finally to obtain 3D stereoscopic Video classify the input, apply the threshold based on depth information. Get the luminance intensity of both images; convert the input images into left eye view image and right eye view images. Get the depth information of the array for newly converted images (left eye and right eye). Then apply DIBR (Input left eye view and right eye view) and apply Depth of focus to generate stereoscopic view of 3D array, finally convert the 3D array into frames.

1.7. 3D Stereoscopic Video Generation

After defining left images and right images, Foreground and back ground are subjected using depth array conversion methodology, final output sequence is obtained by means of following function:

$$s(x, y, t) = \max(d^{bg}(x, y, t), d^{fg}(x, y, t))$$

where baseline parameter of s sequence is used to convert the realistic display on human perspection based on converted disparityback ground (d^{bg}). The full video sequence is generated based on DIBR (Yeh, 2008) technique using our proposed methodology.

1.8. Experimental Results and Performance Analysis

In order to evidence the proposed methodology, we select the converted images from the video sequence, the frame size is of tile 1360×768 (from a 1080 p video).

Experimental results match the perception visually with 3D reality after conversion of video sequence. **Figure 2 and 3** denotes the 3D array and followed by CMYK colour model representation of the converted sequence. CMYK colour model is used in order to outperform higher accuracy rate. **Figure 4** shows the synthesized images of row interlaced 3D converted images which evidence to be 92% accuracy in order to render the 3D video sequence with same quality (1080p) at minimal loss rate. **Figure 2-6** defines clearly about the working methodology.



Fig 2. 2D array of an input image



Fig 3. Converted 3D stereoscopic image of an 2D image

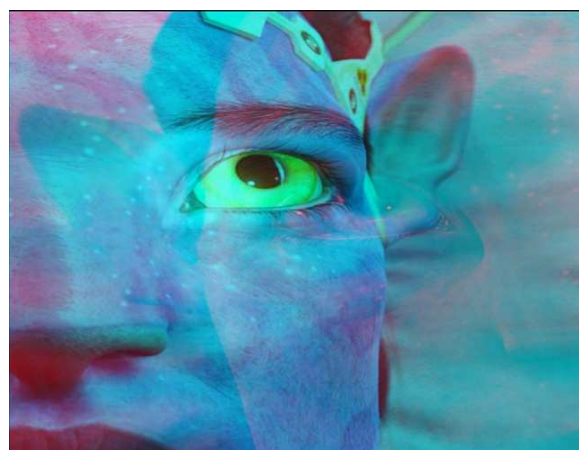


Fig 4. Right eye view image



Fig. 5. Left eye view image



Fig. 6. Converted 3D stereoscopic images

2. CONCLUSION

Here, we proposed a new methodology to convert 2D video sequence into 3D video sequence. Our proposed methodology Depth array conversion deals with better accuracy in conversion of 2D array into 3D array, each parameters are consider during evaluation (cue, saturation, depth perspection, active object, Texture pitch (gradient), motion evaluation (parallax). Hence in ordering the right (3D image) and left (3D image) images of converted sequence produces the depth array of the 3D video sequence. The left image and right image of the converted sequence be denoted clearly in the **Fig. 2 and 3**. By using these sequences of images (left image and right image) stereoscopic videos are generated. Hence we conclude that the Depth array conversion methodology outperforms with 92% accuracy

(achieved with CRYENGINE) in 1080p 3D video, frame rates at 32 frames/second.

2.1. Future Enhancement

Since depth array conversion methodology uses left eye view and right eye view, these criteria leads to increased rate of buffer. To overcome this issue we deployed our methodology in CRYENGINE, since deploying large number of app leads the buffer to overflow. Enhancement of our work in future is towards managing buffer overflow without any additional resource engine like "CRYENGINE". We would like to enhance our work with HSU, monochromatic colour model with varying intensity in red and blue without deploying our model in additional buffer

3. REFERENCES

- Al-Marshadi, A.H., 2011. New procedure to improve the order selection of autoregressive time series model. *J. Math. Stat.*, 7: 270-274. DOI: 10.3844/jmssp.2011.270.274
- Abdul-Nafiu, A.K., 2013. Effects of electrode of 2D resistivity imaging to delineate subsurface features. *Am. J. Applied Sci.*, 10: 64-72. DOI: 10.3844/ajassp.2013.64.72
- Fritz, G., L. Paletta and H. Bischof, 2004. Object recognition using local information content. *Proceedings of the 17th International Conference on Pattern Recognition*, Aug. 23-26, IEEE Xplore Press, pp: 15-18. DOI: 10.1109/ICPR.2004.1333968
- Fergus, R., P. Perona and A. Zisserman, 2003. Object class recognition by unsupervised scale-invariant learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June. 18-20, IEEE Xplore Press, pp: 264-271. DOI: 10.1109/CVPR.2003.1211479
- Park, J.H., S.H. Baeg, J. Koh, K.W. Park and M.H. Baeg, 2007. A new object recognition system for service robots in the smart environment. *Proceedings of the International Conference on Control, Automation and Systems*, Oct. 17-20, IEEE Xplore Press, Seoul, pp: 1083: 1087. DOI: 10.1109/ICCAS.2007.4407060
- Tam, W.J. and L. Zhang, 2006. 3d-Tv Content Generation: 2d-To-3d Conversion. *Proceedings of the IEEE International Conference on Multimedia and Expo*, Jul. 9-12, IEEE Xplore Press, Toronto, pp: 1869-1872. DOI: 10.1109/ICME.2006.262919

- Veltkamp, R.C. and M. Tanase, 2001. Content-based image retrieval systems: A survey. *Wiskunde en Informatica*.
- Yeh, C.Y., 2008. Stereoscopic conversion technique based on monocular view source. National Chung Cheng University.
- Zhang, L. and W.J. Tam, 2005. Stereoscopic image generation based on depth images for 3D TV. *IEEE Trans. Broadcast.*, 51: 191-199. DOI: 10.1109/TBC.2005.846190
- Zhang, L., C. Vazquez and S. Knorr, 2011. 3D-TV content creation: Automatic 2D-to-3D video conversion. *IEEE Trans. Broadcast.*, 57: 272-283. DOI: 10.1109/TBC.2011.2122930