International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Approximate Processing of DIBR Process for 2D to 3D Conversion of Images

Sanjeev Jaiswal[1], Jigyasha Soni[2]

[1]CSVTU University, Rungta College of Engineering & Technology, Kohka Road Bhilai, India

[2]Assistant Professor of Rungta College of Engineering & Technology, CSVTU University Sector 9 Bhilai, India

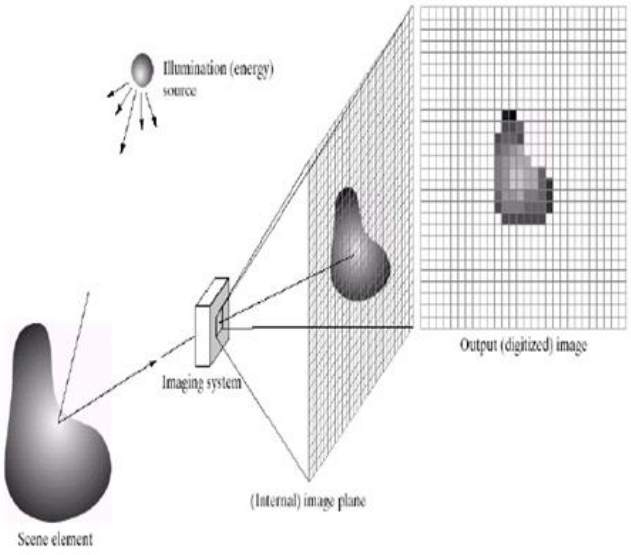
Abstract: ***In the present era all portable devices like mobile, camera etc are needed multimedia applications. In current stage those multimedia applications are require more power which is not suitable for these portable devices. Depth image based rendering (DIBR) process is most useful for 2D to 3D conversion application. In present stage 3D technology is highly in demand. Existing DIBR approaches are facing the problem of timing, complicity at algorithm level, power and area at hardware unit. So in this work I will propose a design which will reduce that complexity. Proposed design is based on approximation approach which is tolerable by human eye. Implementation of algorithm will do on Matlab and hardware implementation will be done by using of Verilog at Xilinx 14.1 platform.***

Keywords: Depth Image Base Rendering (DIBR), SSIM, PSNR, GMSD, FSIM, FPGA, HDL, Image quality parameters.

1. Introduction

A digital image is a representation of a two dimensional image as a finite set of digital values called picture elements or pixels. Pixel values generally represent gray levels, colors, heights and opacities etc.

Remember digitization implies that a digital image is an approximation of a real scene.



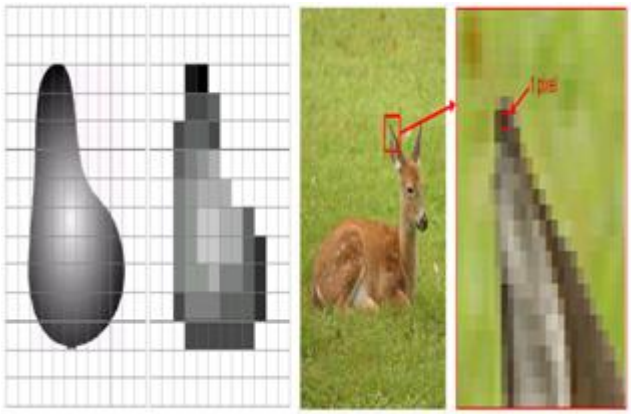


Figure 1: Digitization of image, pixel view of an image

Common image formats include:

1. Sample per point (B&W or Grayscale)
2. Samples per point (Red, Green, and Blue)
3. Samples per point (Red, Green, Blue, and "Alpha")

Figure 2: Grayscale, RGB and RGB alpha view of an image

Digital image processing focuses on two major tasks:

Change of pictorial data for human elucidation and Processing of picture for capacity, transmission and representation for the independent machine observation.

The utilization of advanced picture preparing procedures has blasted and they are presently utilized for a wide range of undertakings in a wide range of zones like:

Image enhancement/restoration,
Artistic effects,
Medical visualization,
Industrial inspection,
Law enforcement,
Human computer interfaces and
3D images for 3D videos etc.

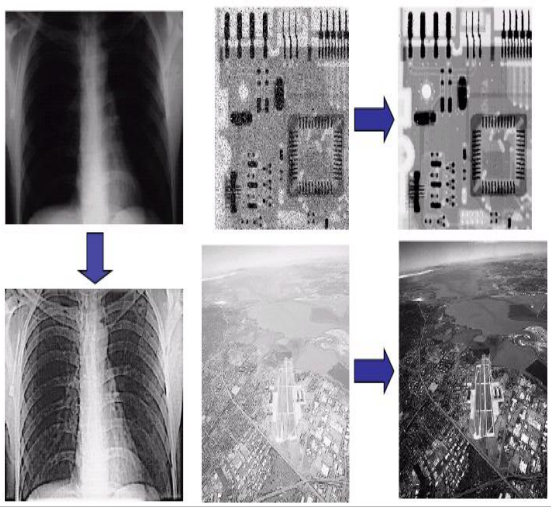


Figure 3: Use of digital image processing technique.

2. Literature Review

2.1. RGB to YC_BC_R (Color Space Transformation):

First, the image should be converted from RGB into a different color space called YC_BC_R. It has three components Y, C_B and C_R: Y component represents the brightness of a pixel, C_B and C_R components represent the chrominance (split into blue and red components). This is the same shading space as utilized by advanced shading TV and additionally computerized feature including feature DVDs and is like the way shading is spoken to in simple PAL feature. The YC_BC_R color space conversion allows greater compression without a significant effect on perceptual image quality. The compression is more efficient as the brightness information which is more important to the eventual perceptual quality of the image is confined to a single channel more closely representing the human visual system. This conversion to YC_BC_R is specified in the JFIF standard and should be performed for the resulting JPEG file to have maximum compatibility.

Notwithstanding, some JPEG usage in "most noteworthy quality" mode don't make a difference this step and rather keep the shading data in the RGB shading model where the picture is put away in partitioned channels for red, green and blue luminance. This outcomes in less productive pressure and would not likely utilized if record size was an issue.

2.2. ITU-R BT.601 Conversion [1]:

The form of $Y'C_BC_R$ that was defined for standard definition television use in the ITU-R BT.601 (formerly CCIR 601) standard for use with digital component video is derived from the corresponding RGB space as follows:

$\mathbf{K_B = 0.114}$
$\mathbf{K_R = 0.299}$

From the above constants the following can be derived for ITU-R BT.601. Analog YC_BC_R from analog R'G'B' is derived as follows:

$$\mathbf{Y' = 0.299 \cdot R' + 0.587 \cdot G' + 0.114 \cdot B'}$$
$$\mathbf{P_B = -0.168736 \cdot R' - 0.331264 \cdot G' + 0.5 \cdot B'}$$
$$\mathbf{P_R = 0.5 \cdot R' - 0.418668 \cdot G' - 0.081312 \cdot B'}$$

Digital $Y'C_BC_R$ (8 bits per sample) is derived from analog R'G'B' as follows:

$$\mathbf{Y' = 16 + (65.481 \cdot R' + 128.553 \cdot G' + 24.966 \cdot B')}$$
$$\mathbf{C_B = 128 + (-37.797 \cdot R' - 74.203 \cdot G' + 112.0 \cdot B')}$$
$$\mathbf{C_R = 128 + (112.0 \cdot R' - 93.786 \cdot G' - 18.214 \cdot B')}$$

Or simply by component wise as

$$\mathbf{(Y', C_B, C_R) = (16,128,128) + (219 \cdot Y, 224 \cdot P_B, 224 \cdot P_R).}$$

The resultant signals range from 16 to 235 for Y' (C_B and C_R range from 16 to 240).

The values from 0 to 15 are called footroom while the values from 236 to 255 are called headroom.

Alternatively, digital $Y'C_BC_R$ can be derived from digital $R'_DG'_DB'_D$ according to the following equations:

$$\mathbf{Y' = 16 + \frac{65.738 \cdot R'_D}{256} + \frac{129.057 \cdot G'_D}{256} + \frac{25.064 \cdot B'_D}{256}}$$
$$\mathbf{C_B = 128 - \frac{37.945 \cdot R'_D}{256} + \frac{74.494 \cdot G'_D}{256} + \frac{112.439 \cdot B'_D}{256}}$$
$$\mathbf{C_R = 128 + \frac{112.439 \cdot R'_D}{256} - \frac{94.154 \cdot G'_D}{256} - \frac{18.285 \cdot B'_D}{256}}$$

In the above formula the scaling factors are multiplied by 256/255. This allows for the value 256 in the denominator which can be calculated by a single bit shift.

If the $R'_DG'_DB'_D$ digital source includes footroom and headroom. The footroom offset 16 needs to be subtracted first from each signal, and a scale factor of 255/219 needs to be included in the equations. This form of $Y'C_BC_R$ is used primarily for older standard definition television systems, as it uses an RGB model that fits the phosphor emission characteristics of older CRTs.

$$\mathbf{Y = \frac{77}{256} E'_{R_D} + \frac{150}{256} E'_{G_D} + \frac{29}{256} E'_{B_D}}$$

$$\mathbf{C_R = \frac{131}{256} E'_{R_D} - \frac{110}{256} E'_{G_D} - \frac{21}{256} E'_{B_D} + 128}$$

$$\mathbf{C_B = -\frac{44}{256} E'_{R_D} - \frac{87}{256} E'_{G_D} + \frac{131}{256} E'_{B_D} + 128}$$

Then take the nearest integer coefficient of base 256. To obtain the 4:2:2 components Y, C_B, C_R, low-pass filtering and sub sampling must be performed on the 4:4:4 C_B, C_R signals described above. Note should be taken that slight differences could exist between C_B, C_R components derived in this way and those derived by analogue filtering prior to sampling.

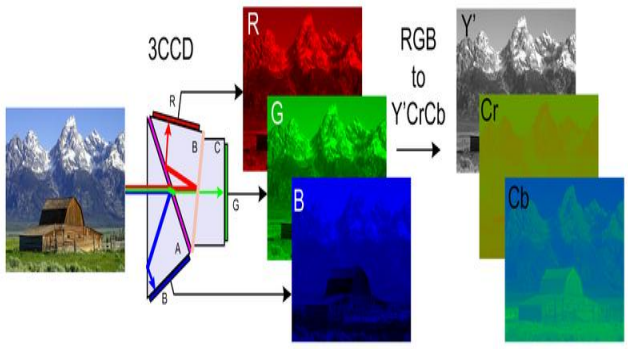


Figure 4: A color image and its Y, C_B, C_R components. The Y image is essentially a grayscale copy of the main image.

2.3. Depth-Image Based Rendering (DIBR):

DIBR is the process of synthesizing 'virtual' views of a scene from still or moving images and associated per-pixel depth information [4][5]. Conceptually this novel view generation can be understood as the following two-step process:
At first, the original image pixel points are re-projected into the 3D world and utilizing the respective depth data. After that, these 3D space points are again projected into the image plane of a "virtual" camera, which is located at the desired viewing position. The concatenation of re-projection (2D-to-3D) and subsequent projection (3D-to-2D) is usually called 3D image warping in the Computer Graphics (CG) literature and will be derived very briefly in the following.

2.4. Stereoscopic Image Creation:

On a stereoscopic or auto-stereoscopic 3D-TV display, two slightly different views of a 3D scene are produced simultaneously on a joint image plane. The horizontal difference between these left and right-eye view so-called screen parallax value are interpreted by the human brain and two pictures are melded into a solitary 3D picture.

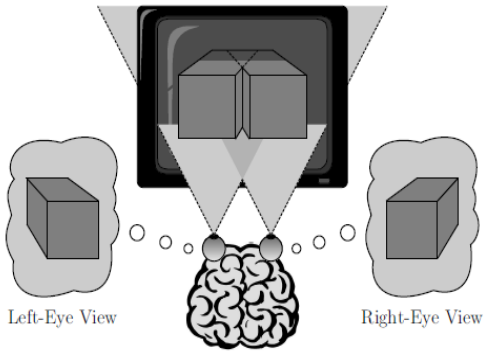


Figure 5: Binocular depth reproduction on a stereoscopic 3D-TV display. Two different perspective views, i.e. one for the left eye and one for the right eye, are reproduced simultaneously on a joint image plane.

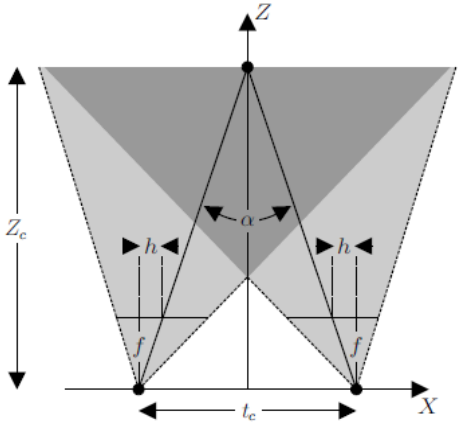


Figure 6: Shift-sensor stereo camera setup. In a shift sensor stereo camera setup, the convergence distance is established by a shift of the camera's CCD sensors.

2.5. DIBR Algorithm:

DIBR is the process of creating virtual views from mono-carpic video and the associated depth information [7]. Consider an arbitrary 3D space point M, the two perspective projection equations in two views result to:

$$\widetilde{m} = AP_n\widetilde{M}, \widetilde{m^*} = A^*P_nD\widetilde{M}.$$

$$\widetilde{m^*} = \widetilde{m} + \frac{A^*t}{depth} + \begin{bmatrix} h \\ 0 \\ 0 \end{bmatrix}, with\ t = \begin{bmatrix} t_x \\ 0 \\ 0 \end{bmatrix}$$

Where $\widetilde{m}$ *and* $\widetilde{m^*}$ represent two 2D image points in left and right view respectively. The matrix D contains the rotation matrix and the translation matrix t that transforms the 3D point from the world coordinate system into the camera coordinate system. The matrices A and A^* specify the intrinsic parameters of the camera. The identity matrix P designates the normalized perspective projection of matrix. Below figure shows a virtual camera setup for rendering of virtual views. The parameters f and t_c represents focal length and baseline distance between two virtual cameras C_l and C_r respectively. Z_c means the depth value of the ZPS (Zero parallax setting).

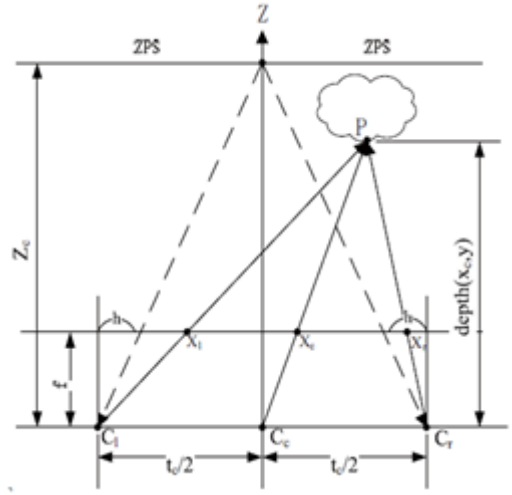


Figure 7: Camera configuration used for generation of virtual stereoscopic

So, the pixel position (x, y) of each warped image point can simply be calculated as:

$$\mathbf{x^* = x + \frac{\alpha_u t_x}{depth} + h, with\ y^* = y}$$

Where α_u is a parameter related to left-right eye distance and eye to screen distance. The pixel position (x_c, y), (x_l, y) and (x_r, y) of the reference view and two virtual views corresponding to the point P with depth has the following relationship:

$$\mathbf{x_l = x_c + \frac{t_c f}{2depth(x_c, y)} + h,}$$
$$\mathbf{x_r = x_c - \frac{t_c f}{2depth(x_c, y)} - h}$$

The offset **h** between reference view and target view can be computed by:

$$\mathbf{h = -\frac{t_c f}{2Z_c}}$$

2.6. Algorithm Optimization:

After hole-filling, the whole left and right view images would be translated horizontally by h pixels to replace the addition and subtraction of h. And as the division operation is time and area consuming by hardware implementation, the DIBR algorithm is optimized to eliminate the division operation. The 1/depth ranges between 0 and 1. Since the value of 1/depth only represents the relative distance of the pixels but not real distance and people are only sensitive about the objects which have smaller depth value, (256−depth)/256 could be used to replace 1/depth, then we get

$$\mathbf{x_l = x_c + k\frac{t_c f}{2}\frac{256 - depth(x_c, y)}{256}},$$

$$\mathbf{x_r = x_c - k\frac{t_c f}{2}\frac{256 - depth(x_c, y)}{256}}$$

In the implementation, for each depth value between 0 and 255, the average deviation between practical value and the theoretical value is about 3.6%. Then, the division operation can be simply implemented by shift operation in hardware. In the implementation, the white color in the depth map (i.e. 255) represents the nearest plane while the black color (i.e. 0) represents the farthest. It is opposite to the real. We use D to represent the value that obtained from the depth map and get:

D = 256 – depth

Now l.offset and r.offset are used to represent the value of offset in the left-and-right view images respectively. So while implementing the DIBR algorithm in hardware there is need to compute left view offset l.offset and right view offset r.offset as:

$$\begin{cases} \mathbf{l.offset = D \cdot pos/256} \\ \mathbf{r.offset = pos - l.offset} \end{cases}$$

where pos = k t_c f is a parameter related to the eye-screen distance and the screen size. pos is set to 1/32 of the width of the screen [6]. This will give a good visual experience and can be simply calculated by shifting operation. After the l offset and r offset are calculated, the left-view image and the right-view image can be obtained as:

$$\begin{cases} \mathbf{lpic(x, y) = pic(x - l.offset, y)} \\ \mathbf{rpic(x, y) = pic(x - r.offset, y)} \end{cases}$$

Where **lpic** and **rpic** represent the left view and right view of image respectively. **pic** is the original 2D image and (x,y) means pixel position in the image.

Pre-Processing of depth image is usually a smoothing filter because depth image with horizontal sharp transition would result in big holes after warping. Smoothing filter is applied to smooth sharp transition so as to reduce the number of big hole. However, if we blur the whole depth image, we will not only reduce big holes but also degrade the warped view. This is because the depth map of non-hole area is smoothed [6].

2.7. 3D Image Warping:

3D image warping maps intermediate view pixel by pixel to left or right view according to pixel depth value. In the other words, 3D image warping transforms pixel location according to depth value. The 3D image warping formula is as following:

$$\mathbf{x_l = x_c + (\frac{t_x}{2}\frac{f}{Z}),}$$

$$\mathbf{x_r = x_c - (\frac{t_x}{2}\frac{f}{Z})}$$

The x_l is the horizontal coordinate of the left view and x_r is the horizontal coordinate of the right view. Besides x_c is the horizontal coordinate of the intermediate view. Z is depth value of current pixel, f is camera focal length and t_x is eye distance. The formula shows that 3D warping maps

pixel of intermediate view to left and right view in horizontal direction.

3. Research Gap

Previous DIBR model [7] [10] having the problem of timing complexity, Proposed the hardware unit of the DIBR but still that hardware unit is having the complexity issues of speed, power and area [10]. All previous approaches are not making justice with SPAA (Speed, power, area and Accuracy).

2.8. Implementation Details

In this section we will discuss about the implementation of previous existing design.

Old DIBR: In this approach author follows the real 3D recording camera setup and according to that setup he will generate some virtual camera configuration formula. According to that formula he calculates left and right image using original 2D image and depth image. This approach is implemented on Matlab.

Modified DIBR: In this approach we will follow the virtual 3D recording camera setup which is generated previously. But in previous approach there is problem of time complexity and when that particular system is implemented on hardware level there is a need of one divider logic so for reduction of these problems we generate one new formula which will calculate left and right image very fast. This approach can also be implemented on Matlab.

2.9. Approximation:

The approximation [14] [15] [16] can increase performance or reduce power consumption with a simplified or inaccurate circuit in application contexts where strict requirements are relaxed. For applications related to human senses, approximate arithmetic can be used to generate sufficient results rather than absolutely accurate results. Approximate design exploits tradeoff accuracy in computation versus performance and power. However, required accuracy varies according to applications. In present era for multimedia application hardware complexity and memory conception are main issue. Human eye sense accuracy of 90-95% so why we need accurate complex hardware, for reduction of complexity issue we can apply approximation unit.

2.10. Image Quality Parameter:

Various parameters are used to evaluate the proposed algorithm at both levels. The various parameters are:

1. PSNR (Peak signal-to-noise ratio)
2. SSIM (Structural-similarity based image quality assessment)[14].
3. FSIM (Feature Similarity Index for Image Quality Assessment) [14].
4. Gradient Magnitude Similarity deviation (GMSD) [11].
5. Riesz-transform based Feature Similarity metric (RFSIM) [12].

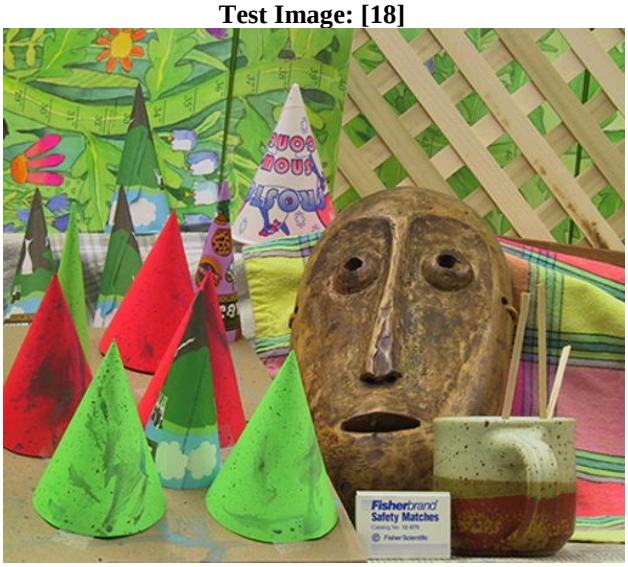
4. Expected Outcome

In our proposed approach it will reduce the time complexity problem also it will make justice with SPAA matrices. We expect in our proposed approach there will approximate two time reduction in time complexity and SPAA matrices.

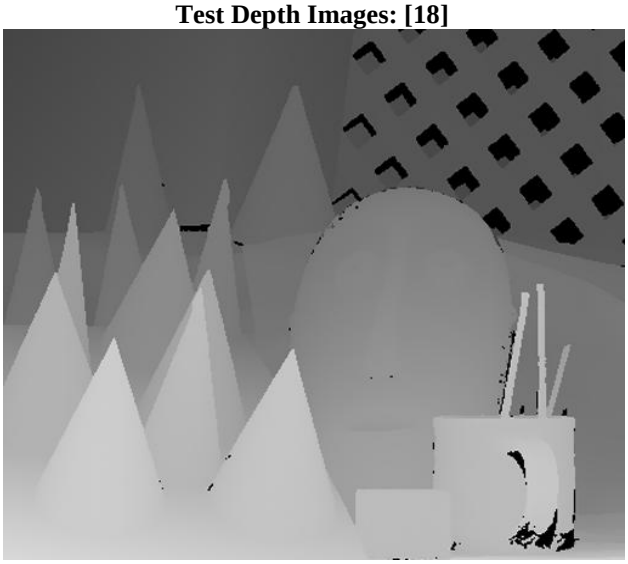
5. Conclusion

In this paper, approximate processing of DIBR for 2D to 3D conversion of Image is proposed to reduce the time complexity issue. We enhance the parameters like speed and power and reduction of hardware area by reducing the accuracy of the image pixels.

Test Image: [18]



Test Depth Images: [18]



References

[1] Rec. ITU-R BT.601-5, Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide-

screen 16:9 Aspect Ratios, (1982-1986-1990-1992-1994-1995), Section 3.5.
[2] Robust Semi-Automatic Depth Map Generation in Unconstrained Images and Video Sequences for 2D to Stereoscopic 3D Conversion Raymond Phan, Student Member, IEEE, and Dimitrios Androutsos, Senior Member, IEEE. Vol. 16 no 1 Jan 2014.
[3] Yu-Cheng Fan, IEEE Senior Member, Yi-Chun Chen, and Shih-Ying Chou. Vivid-DIBR Based 2D to 3D Image Conversion System for 3D Display IEEE 2013.
[4] Kwanghee Jung; Young Kyung Park; Joong Kyu Kim; Hyun Lee; Kugjin Yun; Hur, N.; Jinwoong Kim, "Depth Image Based Rendering for 3D Data Service Over T-DMB" 2008 , vol., no., pp.237,240, 28-30 May 2008.
[5] Hang Shao; Xun Cao; Guihua Er, "Objective quality assessment of depth image based rendering in 3DTV system," vol., no., pp.1,4, 4-6 May 2009.
[6] Nguyen, Q.H.; Do, M.N.; Patel, S.J., "Depth image-based rendering with low resolution depth," Image Processing (ICIP) 2009 vol., no., pp.553, 556, 7-10 Nov. 2009.
[7] Fehn, Christoph. "A 3D-TV approach using depth-image-based rendering (DIBR)." Proc. of VIIP. Vol. 3. 2003.
[8] Wan-Yu Chen; Chang, Yu-Lin; Shyh-Feng Lin; Li-Fu Ding; Liang-Gee Chen, "Efficient Depth Image Based Rendering with Edge Dependent Depth Filter and Interpolation," Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on , vol., no., pp.1314,1317, 6-6 July 2005.
[9] Chao-Chung Cheng; Chung-Te Li; Liang-Gee Chen, "A novel 2Dd-to-3D conversion system using edge information," Consumer Electronics, IEEE Transactions on , vol.56, no.3, pp.1739,1745, Aug. 2010.
[10] Hao Liu; Wei Guo; Chao Lu; Jizeng Wei, "An Efficient Stereoscopic Game Conversion System for Embedded Platform," Trust, Security and Privacy in Computing and Communications (TrustCom), 2011 IEEE 10th International Conference on , vol., no., pp.1235,1240, 16-18 Nov. 2011 (BASE PAPER).
[11] Xue, W.; Zhang, L.; Mou, X.; Bovik, A., "Gradient Magnitude Similarity Deviation: A Highly Efficient Perceptual Image Quality Index," Image Processing, IEEE Transactions vol. PP, no.99, pp.1, 1 Feb 2011.
[12] Lin Zhang; Zhang, D.; Xuanqin Mou, "RFSIM: A feature based image quality assessment metric using Riesz transforms," Image Processing (ICIP), 2010 17th IEEE International Conference on, vol., no., pp.321,324, 26-29 Sept. 2010.
[13] Lin Zhang; Zhang, D.; Xuanqin Mou; Zhang, D., "FSIM: A Feature Similarity Index for Image Quality Assessment," Image Processing, IEEE Transactions on , vol.20, no.8, pp.2378,2386, Aug. 2011.
[14] Zhou Wang; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P., "Image quality assessment: from error visibility to structural similarity," Image Processing, IEEE Transactions on, vol.13, no.4, pp.600, 612, April 2004.
[15] K. Lee, M. Kim, N. Dutt, and N. Venkatasubramanian, "Error-exploiting video encoder to extend energy/QoS tradeoffs for mobile embedded systems," vol. 271, pp. 23–34, 2008.
[16] Zhu, W.-L. Goh, G. Wang, and K.-S. Yeo, "Enhanced low-power high-speed adder for error-tolerant application," pp. 323–327, Nov 2010.
[17] P. Kulkarni, P. Gupta, and M. Ercegovac, "Trading accuracy for power with an under designed multiplier architecture," in VLSI Design, 2011 24th International Conference on, Jan 2011, pp. 346–351.
[18] http://www02.lps.ufrj.br/~eduardo/MMP/3D_results.html.

Author Profile

Sanjeev Kumar Jaiswal received BE degree in Electronics and Telecommunication Engg from Chouksey Engineering College Bilaspur Bilaspur in 2012. Now Pursuing M.Tech degree from Rungta College of Engineering and Technology.