Image Rotation Rectification

in steroscopic 3D on multi-core architectures

Ivan Velciov

"Politehnica" University
Timisoara, Romania
velciov.ivan@gmail.com

Cormac Brick, Marius Predut, Valentin Muresan

Movidius Ltd.
Dublin, Ireland
cormac.brick@movidius.com,
predutionut@yahoo.co.uk,
valentin.muresan@movidius.com

***Abstract*---In this paper, a rectification procedure necessary for obtaining a quality stereoscopic 3D effect, is presented. Rotation rectification is necessary due to misalignment of the pair of image sensors. In particular when the misalignment angle is, greater than 0.7 degrees, the perceived quality of the final 3D stereoscopic image is degraded. This paper offers a low-power, mobile, multi-core solution.**

Keywords - stereoscopic 3D; rectification; VLIW; multi-core

I. INTRODUCTION

In this paper, a rectification procedure necessary for obtaining a high quality stereoscopic 3D effect, will be presented. The 3D effect is obtained by taking two identical pictures, of the same scene, with two identical image sensors, which should be coplanar and with parallel axis. The distance between the two sensors should match the average interoccular distance (63 - 65 mm) [1]. Although the problem of rectification and/or calibration of cameras has been well covered in Computer Vision literature [2] [3] [4], this paper offers a low-power, mobile, multi-core implementation. Even though, in theory, the camera sensors are considered to be coplanar, parallel and having the same rotation angle, but due to the manufacturing process certain errors can be introduced.

The rectification of these errors is what this paper addresess. The first set of rectifications needed, have to do with component placement. Even a component placement tolerance of +/- 0.1 mm (or even 0.025 mm) would introduce noticeable vertical offsets, which need to be corrected. The second problem refers to the fact that a certain physical rotation of the sensors is possible. Starting at an angle of 0.7 degrees, the rotation becomes apparent to the viewer, introducing discomfort. The second set of rectifications, are sensor dependent and involve colour gain, white balance, focal range, lens distorsions, to name just a few.

The next section will go into more detail about stereoscopic 3D, continued by a short presentation of Movidius' platform, on which the rotation rectification was implemented.

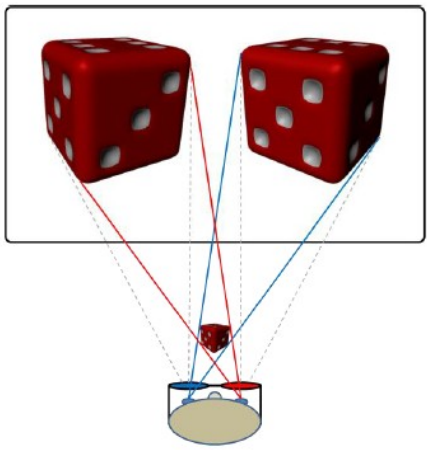

Figure 1. Stereoscopic 3D © 2010 CyberLink Corp.

The results section will describe the achieved performance. The entire test process will be presented afterwards. Finally, conclusions will be drawn.

II. STEREOSCOPIC 3D

Each eye offers a different perspective of the same visualized scene. That is because the eyes are situated at a certain distance, known as interocular distance. This fact will become important later on [1]. Now, an interesting fact is that only one image is perceived, not two. This is due to a process called *stereopsis*, which takes place in the mind. The word *stereopsis* comes from the Greek words *stereo*, meaning *solid*, and *opsis*, meaning *sight*. Besides the two dimensional image depth, distance can also be perceived [6]. How can depth and distance be inferred from two slightly different 2D images of the same scene, captured by two sensors? This is done through depth cues. Depth cues can be monocular or binocular. Monocular cues are perspective, relative size, occlusion, lighting and shadows, relative motion. Perspective refers to the fact that as distance grows objects get smaller. Relative size has to do with the proportions of known objects, for example a mouse

 ISBN: 978-1-61208-195-3

is smaller than a cat. Occlusion is the blocking of view of one object by a second, which is considered to be in the foreground. Lighting and shadows can indicate if an object is sitting on a surface. Objects further away seem to move more slowly than objects in the foreground [7]. This would represent relative motion. When it comes to binocular cues there are three relevant factors: parallax, accommodation and convergence. Parallax refers to the fact that each eye sees a different image. A more detailed explanation will be provided in the next subsection. Accommodation is the muscle tension needed to change the focal length of the eye lens in order to focus at a particular depth. Convergence is the muscle tension required to rotate each eye so that it is facing the focal point [8].

A. *Parallax*

Interocular distance is sometimes referred to as retinal disparity. Parallax and disparity are similar notions, disparity is measured at the eye level while parallax is measured on the display screen, as the distance between two corresponding points in the left and the right view. Parallax can be classified in 3 categories as seen in Fig. 2. Zero parallax, when the eyes converge on the plane of the screen. In other words, the optical axes intersect in a point on the screen. Next category would be positive parallax, when the parallax value is close to the value of the retinal disparity and the optical axes are parallel. In this case, an object would appear to be "inside" the screen. Negative parallax refers to the situation when the optical axes intersect in a point in front of the screen. In this case the objects would appear to be "in front" of the screen [9].

B. *Accomodation / Convergence*

This can be more easily explained through a short example. When you focus on an object 1 meter away from you, two things happen. Your eye changes shape or *accommodates* so that the focal length becomes 1 m and your eyes move so that their axis *converge* on the object. For a graphical representation see Fig. 3 [10].

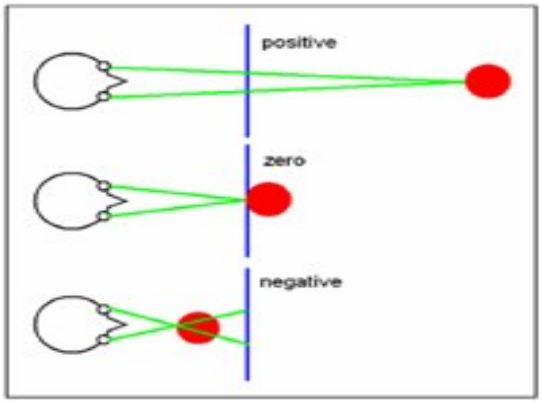

Figure 2. Parallax Classification.

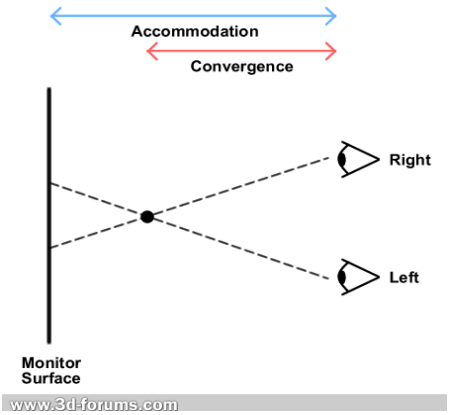

Figure 3. Accomodation / Convergence.

C. *Viewing stereoscopic 3D*

How to display and view stereoscopic images? The simplest method would be to obtain an anaglyph image, which with classic red/cyan glasses, can be viewed on any type of display. Putting it in simpler terms, all that is needed is to overlap the two images, taking into consideration the parallax value. The red/cyan glasses will filter part of the colour spectrum, so that each eye sees the corresponding image, left or right. Because the glasses filter a large portion of the colour spectrum, the viewer experiences a limited colour palette. This problem has been solved with polarized glasses, such as those currently employed in Movie Theatres. The latest commercial technology involves Frame Sequential Displays, paired with active shutter glasses. These displays show in an alternate fashion one frame for one eye and the next frame for the other eye. They need to have a 120Hz refresh rate to avoid flicker. The shutter glasses are synchronized with the display so that the correct frame is displayed at the right time [7].

III. MOVIDIUS PLATFORM

The algorithms for the rectification of the rotated sensor, have been implemented on a Movidius 8 core SoC (System-on-Chip) with a VLIW (Very Long Instruction Word) architecture. The Movidius SoC has 8 DSP (Digital Signal Processor) cores with a VLIW architecture and one RISC (Reduced Instruction Set Computer) core used for control of the DSPs and peripheral system. The SoC is intended for use as a coprocessor to main host processor on a mobile platform. Most Image Processing algorithms display a lot of SIMD (Single Instruction Multiple Data) operations, processing more than one pixel at a time. The Movidius platform really shines when it comes to SIMD operations. Another plus of Movidius' platform, when it comes to Image Processing, is the presence of multiple cores. One could split an image into a batch of lines and apply the same algorithm on them, significantly reducing processing time.

 ISBN: 978-1-61208-195-3

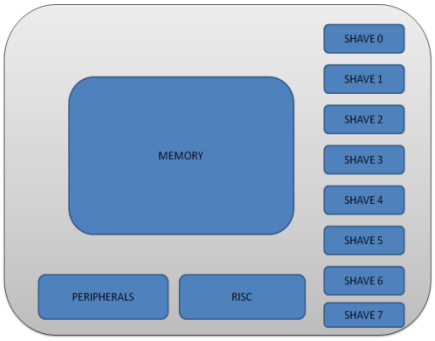

Figure 4. Movidius SOC.

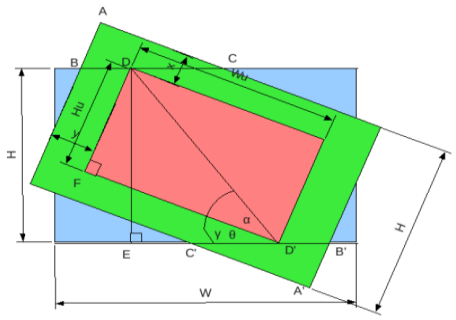

Figure 5. Rotation Compensation.

IV. ROTATION RECTIFICATION

As previously stated, the rectification process requires two main stages. The image - based computation of the rotation angle of the sensor and the actual rectification process. In the next subsections these two stages will be presented in a more detailed fashion.

A. Angle Computation

There are two methods to determine the rotation angle. The two algorithms make use of interest points, the points used to compute the parallax. In other words, using a feature extraction algorithm, such as the Harris corner detector, on one of the images and then use SAD (sum of absolute differences) to find matching points in the other image. Only after these points have been found can the angle computation algorithm begin. The rotation angle is considered to be relative to one of the images. One method requires the selection of 8 interest points: four, with the highest horizontal distance between them, and the other four with the highest vertical distance. Once these eight points have been selected, for each group of four points all the possible lines they can form, are determined. Having the line equations, the angle between this line and the corresponding line in the other image, can be obtained. The obtained angle values are then averaged, with weights, obtaining a single value that represents the angle between the two images. The second approach is based on finding the minimum area convex polygon with the interest points, either being on the polygon edges or inside its area. This can be done with a complexity of O(N * H), where H is the number of points of the polygon. Considering the polygon edges as vectors, they are summed up. The same is done on the second image's polygon. Thus, the relative rotation of the image will be the angle between the two resulting sum vectors.

B. Rotation Rectification

The most basic way to do this would be to rotate the image back. The main problem with this solution is that area loss occurs, in the four corners. The loss of information is directly proportional with the rotation angle. For a sensor rotation not larger than 0.7 degrees this loss is negligible. A useful area is defined as the largest area that can be rotated without area loss. The actual rotation is done only on this area and then the area is resized back to its original resolution. The actual useful area can be determined by finding the width and height of the area and the coordinates of the top left corner, using the following equations.

$$FD' = W_u = \cos\alpha * DD' = \cos\left(arctg\frac{H}{W}\right) * \frac{H}{\sin\left(\theta + arctg\frac{H}{W}\right)} \quad (1)$$

$$FD = H_u = \sin\alpha * DD' = \sin\left(arctg\frac{H}{W}\right) * \frac{H}{\sin\left(\theta + arctg\frac{H}{W}\right)} \quad (2)$$

$$x = \frac{H - H_u}{2}$$
$$y = \frac{W - W_u}{2} \quad (3)$$

, where

H-Height W-Width θ - rotation angle
H_u-useful height W_u - useful width

The equations can be easily obtained by looking at Fig. 5. To facilitate efficient implementation, an algorithm was sought that could be easily parallelized and would work in-place when accessing memory. Alan Paeth's rotation by shear algorithm was chosen [11]. According to this algorithm a rotation can be obtained by three shear operations. With reference to Fig. 6 and equation (7) the first shear is done on one of the axis (shearX(α)), the next one on the other axis

 ISBN: 978-1-61208-195-3

(shearY(β)), and the third one on the first axis, (shearX(γ)). Alternatively one may also construct this operation with two Y-axis operations, and one X-axis operation if more convenient. A two-dimensional shear operation has the following matrix representation, one for each axis [12].

$$ShearX(\alpha)=[1\alpha;01] \quad (4)$$

$$ShearY(\beta)=[10;\beta\ 1] \quad (5)$$

Starting with the familiar rotation matrix,

$$Rot(\Theta)=[\cos(\Theta),-\sin(\Theta);\ \sin(\Theta),\ \cos(\Theta)] \quad (6)$$

, expressing this matrix to the product of the three shear operation matrices equation (8) is obtained.

$$shearX(\alpha)shearY(\beta)shearX(\gamma)=Rot(\Theta) \quad (7)$$

$$[1\alpha;01][10;\beta1][1\gamma;01] = [1+\alpha\beta,\alpha+\gamma+\alpha\beta\gamma;\beta,1+\beta\gamma] \quad (8)$$

Solve for α, β, γ in terms of Θ and obtain:

$$\alpha=\gamma=-\tan(\Theta/2) \quad \beta = \sin(\Theta) \quad (9)$$

Taking into consideration that the maximum rotation angle is 0.7 degrees, for an image with height H, width W and rotation angle α, the total area loss can be determined with the following formula:

$$2*H^2*\tan(\alpha/2) + W^2 * \sin(\alpha) \quad (10)$$

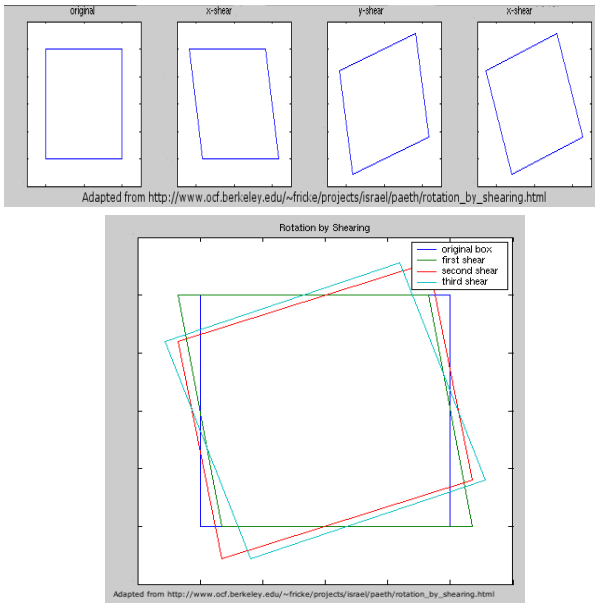

Figure 6. Rotation by Shear.

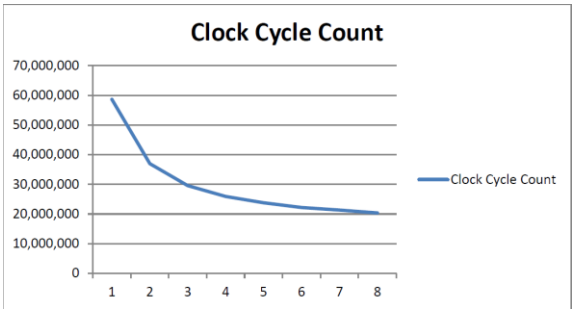

Figure 7. Perfomance Test (clock cycles / no of cores).

V. RESULT

The goal of this solution, was to enable the streaming of HD (High Definition) 3D stereo content at 30 fps (frames per second). The HD resolution achieved was HD 720p (1920x720). The rotation rectification was necessary due to the fact that the 3D stereo algorithm employed, considered, both cameras, to have the same parameters. Due to manufacturing errors this was not possible so this solution became imperative. The implementation and testing was done on Movidius' MV117 development board, equipped with a Myriad SoC, clocked at 180 MHz. The Myriad SoC is a low-power, multi-core, mobile solution, enabling mobile phone manufactures to bring 3D stereo to the mobile world. An in place 3-stage rotation implementation was chosen, for both memory and computational efficiency. This also facilitates data parallel processing so the algorithm may be easily split across a number of processing cores. The input image may be segmented into batches of lines, which can then be processed in parallel. In this case, the number of lines used was 9. A series of tests were performed in order to obtain the optimum number of cores, to use, in a real-time scenario. For this, the clock cycle count was measured as the number of cores was increased. The obtained measurements can be found in Fig. 7.

VI. CONCLUSION

In this paper, it has been shown that many problems must be addressed for a quality stereoscopic 3D image. These problems arise, due to differences in camera sensors and manufacturing placement errors. Although two sensors are identical, because of the manufacturing process they differ in small but essential points for stereoscopic 3D. One such difference was the incorrect rotated PCB (Printed Circuit Board) placement of the sensor. By using Movidius' platform, it becomes clear that a software implementation, on a powerful multi-core, mobile, low-power architecture can handle well stereoscopic 3D video content, at HD 720p resolutions.

 ISBN: 978-1-61208-195-3

ACKNOWLEDGMENT

This research has been supported from the EU Structural Funds Research Project POS-CCE 499-11844 "Falx Daciae – Software Tools and Development Processes for Advanced Multimedia Applications on Mobile Phones".

REFERENCES

[1] L. Kaufman, "Sight and mind: An introduction to visual perception," New York, Oxford University Press, 1974, ISBN-10: 0195017633

[2] J. Zhou, "New image rectification schemes for 3d vision based on sequential virtual rotation," PhD Thesis, June 2009 www.public.asu.edu/~jzhou19/phd_thesis.pdf, last accessed 10/11/2011

[3] R. Hartley, "Estimation of relative camera positions for uncalibrated cameras," Proc. of ECCV-92, G. Sandini Ed., LNCS-Series, vol. 588, pp. 579–587, Springer-Verlag, 1992

[4] C. C. Slama, "Manual of Photogrammetry, Fourth Edition," American Society of Photogrammetry, Falls Church, Va, 1980, ISBN-10: 1570830711

[5] C. Harris and M.J. Stephens, "A combined corner and edge detector," 4th Alvey Vision Conference, pp 147–152, 1988.

[6] D. L. MacAdam, "Stereoscopic perceptions of size, shape, distance and direction," SMPTE Journal, vol. 62, pp. 271-293, 1954

[7] "3D WP Principles of 3D Video and Blu-ray 3D", copyright © 2010 CyberLink Corp. All rights reserved

[8] N. A. Valyus, "Stereoscopy," London, New York: Focal Press, 1962, ISBN-10: 0240387953

[9] L. Lipton, "Binocular symmetries as criteria for the successful transmission of images," Processing and Display of Three-Dimensional Data II, SPIE vol. 507 , 1984

[10] C. Wheatstone, "On some remarkable, and hitherto unobserved, phenomena of binocular vision (Part the first)," Philosophical Transactions of the Royal Society of London, pp. 371-394, 1838

[11] A.W. Paeth, "A fast algorithm for general raster rotation," Proceedings of Graphics Interface, pp. 77-81, 1986

[12] T. Fricke, http://www.ocf.berkeley.edu/~fricke/projects/israel/paeth/rotation_by_shearing.html, last accessed 10/11/2011

 ISBN: 978-1-61208-195-3