2016 29th International Conference on VLSI Design and 2016 15th International Conference on Embedded Systems

Design And Implementation of Blind Assistance System Using Real Time Stereo Vision Algorithm

Vaddi Chandra Sekhar, Satyajit Bora, Monalisa Das,Pavan Kumar Manchi,Josephine S and Roy Paily
Department of Electronics and Electrical Engineering
Indian Institute of Technology Guwahati
Email: cschandrasekhar123@gmail.com,satyajit.bora@iitg.ernet.in,monalisa.das@iitg.ernet.in,
m.pavan@iitg.ernet.in,josephine@iitg.ernet.in,roypaily@iitg.ernet.in,

***Abstract*—The main objective of this paper aims at designing an efficient blind assistance system for the visually impaired people using real time disparity estimation algorithm. The local window based matching algorithms known as sum of absolute differences (SAD) and zero-mean SAD (ZSAD) are used for the disparity estimation and efficient hardware architectures for those algorithms are implemented in FPGA. The SAD and ZSAD algorithms are implemented for the image resolution of 640x360 pixels with square window of size 8x8 pixels and with a disparity range of 0 to 99. An efficient line buffering scheme for the left and right window of the two camera images is implemented to support the parallel processing mechanism. To provide high frame rate per second (fps) to the artificial vision system, parallel processing architecture for disparity estimation is required, but it consumes huge amount of hardware resources which is not desirable for this application. So, semi-parallel architectures are implemented for SAD and ZSAD algorithms for a compromise between hardware resource utilization and speed. The FPGA resource utilization for the ZSAD and SAD algorithms for a frame rate of 30 fps are 43529 LUTs, 50144 FFs and 34548 LUTs, 37544 FFs respectively. ZSAD algorithm consumes around 30% more hardware resources compared to SAD, but to counter the randomized distortion caused by the non-ideal stereo cameras ZSAD is more preferred. Finally from the disparity, the distance of the nearest obstacle is estimated and blind person is alerted through audio device instructions.**

***Index Terms*—SAD, ZSAD, FPGA, Blind Assistance System.**

I. Introduction

According to WHO, it is estimated that there are a total of 285 million visually impaired people around the world (out of which 39 million are blind and 246 million have low vision), whose quality of life is limited since they need to depend on others for their navigation. So, much research has been carried out for developing a better navigation system in order to create a better and independent way of living for the blind. Blind Assistance Systems are mainly categorized as ETAs (Electronic Travel Aids) [1], EOAs (Electronic Orientation Aids), PLDs (Position Locator Devices), bionic eye, and artificial vision systems. Some devices based on ETAs are - ultra-sonic spectacles and waist-belt [2]; Sonic Pathfinder [3]; Mowat-Sensor [4]; Guide-Cane [5]; Sonic-Guide [6]; and Nav-Belt [7] which uses ultra-sonic sensor or laser for detecting the object and calculating the depth, but each has its limitation of detecting ground level objects, the obstacle shape, etc. The ultrasonic sensors have the problem of cross talk if two or more such sensors are used side by side; Blind Navigation Systems based on PLDs are RFID [8] and Drishti [9]. They use GPS wireless technology for locating, but it is not always efficient since it based on satellite communication. Navatar [10] uses orientation aid sensors like accelerometer or compass and mobile devices. Recent research is on bionic eye [11], which consists of a small digital camera, external processor, a microchip implant and stimulating electrodes surgically placed in the back of the eye. This electronically powered eye is very costly and at present is applicable only in two medical eye conditions, i.e., that are retinitis pigmentosa and age-related macular degeneration. Also there are artificial vision systems using retinal implant, optic nerve implant, and cortical implant, that require extensive surgery techniques. Although there are many methods developed for blind navigation, the stereo disparity concept is an efficient method for navigation as it only uses stereo cameras, from where depth is estimated. This makes this system feasible, cost effective, and an independent device, for the blind people to carry out their navigation.

The process of extracting depth information from multiple 2D views of a scene is stereo vision or stereoscopic vision. This works in the same principle as human eye. The same idea is utilized in computer vision techniques to estimate depth from two stereo camera images. The positional difference between two images of the same scene is known as disparity. Depth is calculated from disparity by triangulation method and it is found that depth and disparity are inversely proportional. Thus disparity finding is a very important step for stereo vision concept. For pixel based matching techniques for finding disparity maps, there are Local Method or Area based Method algorithms and Global Method or Feature based Method algorithms. Here, local methods are preferred, though global methods give better efficiency, because it exhibits huge computational costs and require huge storage areas which make them unsuitable for real-time applications. Although Local methods give less accuracy than Global methods, they are much faster and require less computational costs and storage areas, which make them suitable for real-time applications. There are various local disparity estimation methods which are implemented in hardware. The hardware implementation is accomplished using FPGA boards for real time processing.

This paper presents designing an efficient blind assistance navigation system for blind persons using real time disparity estimation methods. The local window based disparity match-

978-1-4673-8700-2/16 $31.00 © 2016 IEEE
DOI 10.1109/VLSID.2016.11

ing algorithms known as Sum of Absolute Differences (SAD) and zero-mean SAD (ZSAD) are used for disparity estimation and efficient hardware architectures for those algorithms are implemented in FPGA. An efficient line buffering scheme for the left and right window of the two camera images are implemented to support the parallel processing mechanism.

The rest of the paper is organized as follows. Section II discusses the existing disparity estimation methods and compares their matlab simulation results. Section III explains the proposed hardware architectures giving details of different stages. The results obtained by implementing the architectures in Xilinx ZC702 SoC board are discussed in section IV and section V concludes the paper.

II. Existing Disparity Estimation Methods

Some local methods for Disparity Estimation Methods include the most common algorithms like,

- Sum of Absolute Differences (SAD)
- Zero Mean Sum of Absolute Differences (ZSAD)
- Sum of Squared Differences (SSD)
- Zero Mean Sum of Squared Differences (ZSSD)
- Normalized Cross Correlation (NCC)
- Zero Mean Normalized Cross Correlation (ZNCC)

For better resource utilization, SAD is preferred than SSD, though both give almost the same result. In SAD absolute value needs to be found while SSD requires squaring each subtracted value. ZSAD and NCC give very high performance compared to SAD in case of radiometric distortion in images. But NCC is more hardware consuming than SAD because of the square root, squaring, multiplication and division operations. ZSAD is better in resource utilization than NCC though it has the same RMS error and it takes far less time than NCC. So here, only SAD and ZSAD are discussed.

A. Simulation Scheme

Standard Tsukuba image is taken for comparison and the following parameters are checked.

1) Time: It is the time to find disparity for one Pixel in Seconds.
2) PSNR (Peak Signal to noise ratio): Defined as the ratio of square of maximum intensity possible in the clean image and the mean square error. It is expressed in db.I_1 is the Clean Image.

$$PSNR = 20\log_{10}\frac{Max(I_1)}{RMS} \tag{1}$$

3) RMS (Root mean Square Error): It is the square root of the mean of the squared error of the image;

$$RMS = \sqrt{\frac{\sum(I_1 - I_2)^2}{m \times n}} \tag{2}$$

where, I_1: Ground Truth Disparity image (Clean Image) I_2: Disparity Map found out.

According to simulation results obtained, if there is no radiometric distortion in the images then both the algorithms, SAD and ZSAD give almost same disparity results. For window size 3x3, SAD has improved performance than ZSAD, but for 8x8 and 11x11 window size, the PSNR is almost same for both SAD and ZSAD. But for the images having radiometric distortion i.e. images having different brightness level, ZSAD gives better performance compared to SAD. Also if we increase the window size from 3x3 to 8x8, the PSNR improvement is very high. But if the window size is further increased to 11x11, the PSNR improvement is not much. So, 8x8 window size is used in the proposed algorithms. ZSAD consumes more hardware resources than SAD because of the need of the mean of all pixels. But using efficient techniques to find the mean and the line buffering schemes, reduction of resources can be obtained. So for better hardware resource utilization and better real time performance, ZSAD architecture with 8x8 window size is preferred for the blind navigation system.

B. Simulation Results

TABLE I
SAD (NON- RADIOMETRIC DISTORTION)

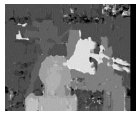
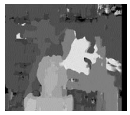
	3x3 Window	8x8 Window	11x11 Window	
				
Time(sec)	44.5695	43.8519	43.6959	Ground Truth
PSNR	15.6839	17.8325	18.171	Disparity Map
RMS	2.4655	1.9252	1.8516	

TABLE II
ZSAD (NON- RADIOMETRIC DISTORTION)

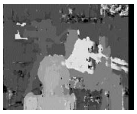
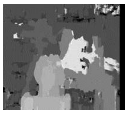
	3x3 Window	8x8 Window	11x11 Window	
				
Time(sec)	97.9218	96.6894	96.1902	Ground Truth
PSNR	13.9552	17.3192	17.6572	Disparity Map
RMS	3.0084	2.0424	1.9644	

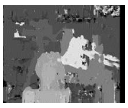
TABLE III
SAD (RADIOMETRIC DISTORTION, LEFT PIXELS ARE SHIFT BY AN OFFSET VALUE OF 10)

	3x3 Window	8x8 Window	11x11 Window	
				
Time(sec)	44.5539	43.8051	43.6491	Ground Truth
PSNR	10.9763	11.7928	12.4371	Disparity Map
RMS	4.2391	3.8588	3.5829	

III. HARDWARE ARCHITECTURE

The block diagram of the proposed architecture is shown in Fig. 1. It has four major blocks namely pre-processing, processing, display and audio. The pre-processing block converts the camera raw data into 8 bit gray format and then compresses the image. The processing block estimates the disparity map and stores it in memory. The display block displays the processed data i.e. disparity image in HDMI. The audio block estimates the obstacles and informs the blind person through audio signals.

TABLE IV
ZSAD (RADIOMETRIC DISTORTION, LEFT PIXELS ARE SHIFT BY AN OFFSET VALUE OF 10)

	3x3 Window	8x8 Window	11x11 Window	
				
Time(sec)	98.3274	96.6894	95.8158	Ground Truth
PSNR	13.9538	17.3192	17.6572	Disparity Map
RMS	3.0089	2.0424	1.9644	

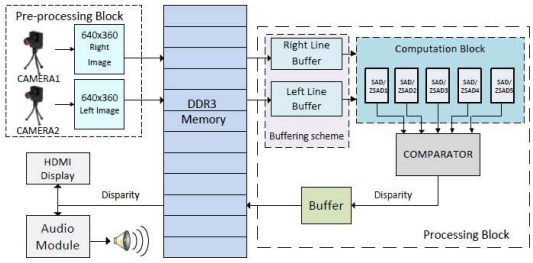


Fig. 1. Hardware architecture

A. Pre-processing

The camera used in the implementation is Avnet on Semiconductor Image Sensor Module with HDMI Input/Output. The camera gives colour image in YCbCr format with a resolution of 1920x1080. But for processing, gray image is required. So the camera data is first converted into 8 bit gray format and then the image is compressed to 640x360. This is done by dividing the original image into small windows of size 3x3 and then replacing each of them by its average value. In this way the image width is reduced to 640 pixels and height is reduced to 360 pixels. The processing can be made for the full resolution image. But this will increase the complexity and in turn will consume a lot of resources. So, in the proposed algorithm 640x360 resolution image is used.

B. Processing

The processing module consists of three units; line buffering scheme unit, computation unit and comparison unit. The purpose of the line buffering scheme unit is to provide the 8x8 windows of interest. Computation unit does the required computation and gives output to the comparison unit. Comparison unit compares the output of the computation unit and gives disparity as output. In the proposed work, hardware architecture is designed for two algorithms, namely ZSAD and SAD. The buffering scheme and comparison unit is same in both the architectures, only difference is the computation part.

1) Line buffering scheme: In the proposed design the right image is taken as reference image and the window size is selected as 8x8. This window is compared with different windows in the left image. This is achieved by designing a line buffering scheme shown in Fig. 2. For 8x8 window size eight line buffers, having a length equal to the length of the row in the image are required. Each of these line buffers has 1 input and 8 outputs. The incoming right image pixels are entered in row buffer 8 and existing pixels in row buffer are shifted to the right. The last 8 pixels of the row buffer are taken as output and last pixel of the row is given as input to row buffer 7. Similarly last output from row buffer 7 is given to row buffer 6 and so on. The primary work of the row buffer is to select a segment of a row of the image for processing. So, 8 line buffers together give an 8x8 window. After initial latency this gives an 8x8 window whenever a new pixel arrives. For the left image, the buffering scheme is shown in Fig. 3. In this case the outputs of the line buffers are increased so that simultaneously we can access more number of 8x8 windows. In the proposed architecture each line buffer has 107 outputs so that we can get 100 numbers of 8x8 windows.

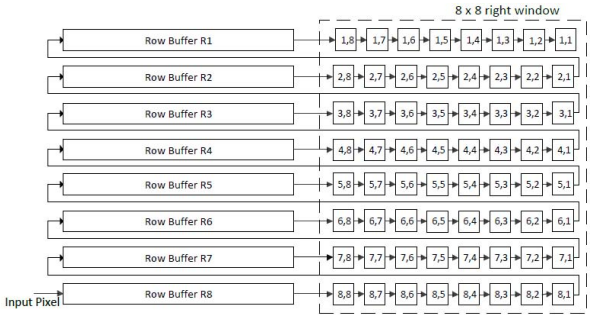


Fig. 2. Right image line buffering scheme

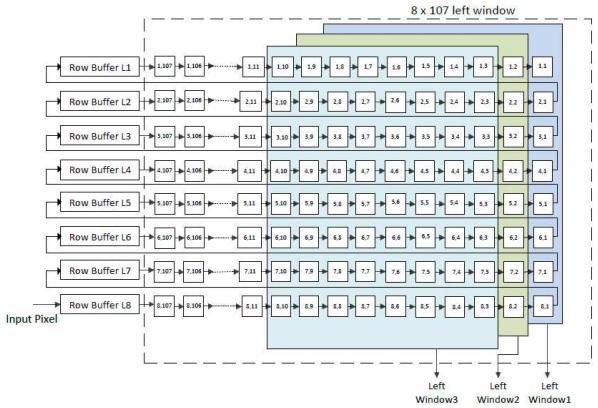


Fig. 3. Left image line buffering scheme

2) Computation ZSAD: The left and right windows are given as input to the ZSAD module. The ZSAD module first computes the average of the pixels of the windows. For average calculation of 8x8 window size, the sum of the pixels should be divided by 64. So in order to divide by 64, the summation is right shifted by 6 bits as 64 is equal to 2^6. Next the right window average (Ravg) is added to the left window pixels and left window average (Lavg) is added to the right window pixels as shown in Eq. 4. Then the absolute difference between the widows is calculated and added to get the required ZSAD

output.

$$ZSAD = \sum_{(i,j)\epsilon w} |L(i+d,j) - L_{avg} - R(i,j) + R_{avg}| \quad (3)$$

$$ZSAD = \sum_{(i,j)\epsilon w} |(L(i+d,j) + R_{avg}) - (R(i,j) + L_{avg})| \quad (4)$$

Where d ϵ [0, 1, 2, ... 99]

In the proposed algorithm, 5 parallel ZSAD modules are used to make the computation faster. More number of parallel ZSAD modules can also be used, but it will consume more area. So, to get a high speed area efficient architecture 5 ZSAD modules are used. The complete architecture for ZSAD is shown in Fig. 4.

3) Computation SAD: In case of SAD architecture also 5 parallel SAD blocks are used. The SAD block first finds the absolute difference between the left and right image pixels and then adds them (as shown in Eq. 5) to get the required SAD output.

$$SAD = \sum_{(i,j)\epsilon w} |L(i+d,j) - R(i,j)| \quad (5)$$

Where d ϵ [0, 1, 2, ... 99]

4) Comparison: The comparison unit takes 5 inputs at a time, compares them and stores the minimum one along with its disparity. In the next clock cycle it again compares the 5 inputs and finds the minimum. If new minimum is less than the stored minimum then it is replaced by the new minimum. In this way it works for 20 clock cycles and at the end of 20th clock cycle, it gives the valid disparity as output.

C. Display

For the blind assistance system, the HDMI display part is not a mandatory step. But this is included to check the disparity output of the architecture. In this step, the disparity output is displayed on a HDMI monitor. In disparity image, black pixels mean the object is far and white pixels mean the object is at a closer distance.

D. Audio

ZSAD and SAD algorithms give a dense disparity output. This is used to calculate the depth information by the following formula [12].

$$D = (b * f)/d \quad (6)$$

where b is the distance between the left and the right cameras, f is the focal length of the cameras and d is the disparity. This gives the depth at each pixel. A particular disparity value corresponds to a particular depth. So, if an object comes at a particular distance, the number of disparity values corresponding to that distance will increase. In the proposed architecture, first the numbers of different disparity values present in a single frame are calculated. If the disparity is more, it can be estimated that there is an object at a distance corresponding to the disparity.

For convenience, in the proposed architecture, the disparity range is divided into 10 equal parts namely 0-9, 10-19, 20-29 and so on. And the numbers of disparity values present in the range are calculated. If the number of disparity values in the range 90-99 is increased above a threshold then the blind person is informed that there is an object within 1 metre. Similarly, if the number of disparity values in the range 80-89 is increased then the blind person is informed that there is an object within 2 metre and so on.

The audio device used is PMODI2S, consisting of three clock signals, Master clock (mclk), Serial or bit clock (sclk/ bclk) and Left-Right clock (lrclk) that is the sampling frequency Fs, (in kHz). The values for these clock frequencies are calculated as

$$\left.\begin{array}{l} mclk = k * Fs \\ sclk = 32 * Fs \end{array}\right\} \quad (7)$$

where k = 64, 128, 256, 512, 1024 for 16 bit data and 96, 192, 384, 768 for 24 bit data. Here, the value of k is taken as 256 for a sampling frequency of 48 kHz. The ratio of the mclk to bclk is 8 and the ratio of bclk to lrclk is 32. As a result clear and audible voice messages are played whenever any obstacle is detected depending on the threshold value set according to the disparity ranges.

IV. RESULTS AND DISCUSSION

The hardware implementation of the proposed designs is done in Zynq ZC702 SoC board. The cameras used are Avnet on Semiconductor Image Sensor Module with HDMI Input/Output. The software platform used is Vivado 2014, SDK 2014. In this work we proposed hardware architecture for two algorithms.

- One of the architecture is for SAD with a window size of 8x8 and disparity range of 99.
- The other architecture is for ZSAD with a window size of 8x8 and disparity range of 99.

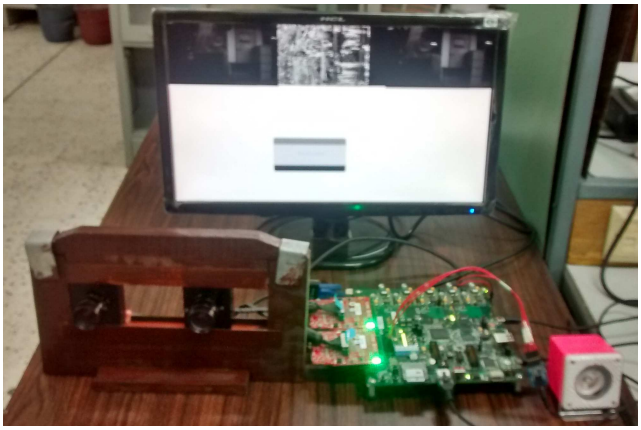


Fig. 5. Experimental Setup

Fig. 6 shows the resource utilization in the two systems. The resource utilization in terms of Memory LUT, I/O, DSP48,

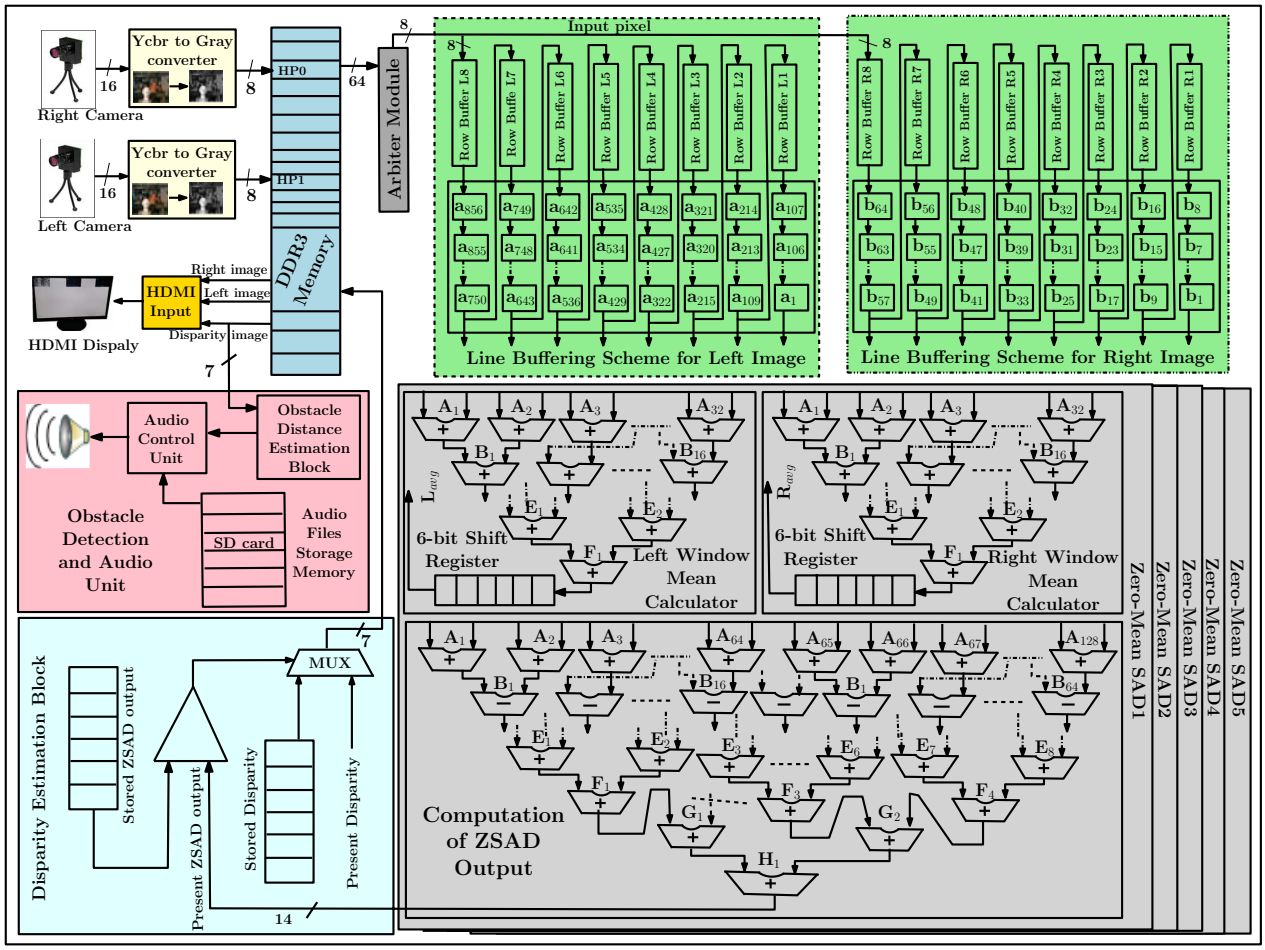


Fig. 4. Architecture of blind assistance system using real time zero-mean sum of absolute differences (ZSAD) stereo vision algorithm. The ZSAD algorithm is implemented with image resolution of 640x360 pixels with square window of size 8x8 pixels and with a disparity range of 100. The left and right image line buffer sizes are Row Buffer Lx = 533 bytes, Row Buffer Rx = 632 bytes respectively.

Utilization - Post-Implementation

Resource	Utilization	Available	Utilization %
FF	37492	106400	35.24
LUT	32062	53200	60.27
Memory LUT	597	17400	3.43
I/O	72	200	36.00
BRAM	43.5	140	31.07
DSP48	8	220	3.64
BUFG	5	32	15.62

(a)

Utilization - Post-Implementation

Resource	Utilization	Available	Utilization %
FF	50144	106400	47.13
LUT	43529	53200	81.82
Memory LUT	596	17400	3.43
I/O	72	200	36.00
BRAM	118.5	140	84.64
DSP48	8	220	3.64
BUFG	5	32	15.62

(b)

Fig. 6. Resource utilization for system with (a) SAD (b) ZSAD

(a) (b)

Fig. 7. (a) Left camera image and (b) SAD disparity

BUFG is almost same in both the systems. But the ZSAD architecture utilizes 12652 FFs, 11467 LUTs and 75 BRAMs more than SAD architecture, which is around 30% of resources used by SAD architecture. Although the ZSAD architecture is consuming more area than SAD architecture, to counter the randomized distortion caused by the non-ideal stereo cameras ZSAD is preferred. Fig. 7 and 8 show the real time disparity output of SAD and ZSAD architectures. The intensity of the pixels indicates the depth. The bright pixel values signify that the object is near to the camera. From the figures it is clear that the output of ZSAD architecture is better than the SAD architecture and hence the accuracy will be more for ZSAD architecture. But if a system needs to be designed with a small amount of resources then SAD will be the better option.

TABLE V
COMPARISON OF THE PROPOSED ARCHITECTURES WITH EXISTING ARCHITECTURE

Resources	Platform used	Resolution	Window Size	Disparity Range	FF	LUT	Frame Rate(fps)
Proposed SAD	Zynq ZC702 Soc	640x360	8x8	99	14981	14727	30
Proposed ZSAD	Zynq ZC702 Soc	640x360	8x8	99	27654	26488	30
[13]	Altera Cyclone II	640x480	5x5	64	275328	54338	35
[14]	Xilinx Virtex XCV-2000E	800x600	8x8	64	NA	NA	20.33
[15]	Xilinx Virtex4 XC4VLX15	512x512	5x5	256	NA	NA	25.6
[16]	FourAlteras Stratix FPGAs	640x480	Multi-scale	128	NA	NA	30

(a) (b)

Fig. 8. (a) Left camera image and (b) ZSAD disparity

In Table V the proposed architectures are compared with some existing hardware architectures for stereo matching techniques. The proposed architectures work at a clock frequency of 150 MHz and give a frame rate of 30 fps which is higher than [14], [15] and equal to [16]. The frame rate of [13] is little higher than the proposed architectures but the resources used is much more than the proposed architectures.

V. CONCLUSION

The navigation system implemented in this paper is quiet efficient for real time applications as it has a good optimization of resources and speed. The semi-parallel architecture presented, for SAD and ZSAD algorithms compromise between hardware resource utilization and speed.The depth of the nearest obstacle is estimated from the disparity and the visually impaired person is alerted through an audio device.

Future work includes more improvement in processed image using different algorithm or combination of algorithm techniques which best suits the system. And object detection can also be implemented, for the blind people to detect the exact obstacle. Initially simple object detection like stair case detection, vehicle detection etc. can be done along with depth information. Also, the complete set-up of the navigation system can be made using pair of glasses with camera fitted on the two sides and an ear-piece connected to the device, to alert the blind person about any obstacle present in the environment.

VI. AKNOWLEDGEMENT

The authors are thankful to DeitY, India for providing software and hardware resources to carry out this work through 'Design and implementation of a Blind Assistance System using FPGAs and Sensors' project.

REFERENCES

[1] D. Dakopoulos and N. Bourbakis, "Wearable obstacle avoidance electronic travel aids for blind: A survey," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 40, no. 1, pp. 25–35, Jan 2010.

[2] S. Bhatlawande, J. Mukhopadhyay, and M. Mahadevappa, "Ultrasonic spectacles and waist-belt for visually impaired and blind person," in *Communications (NCC), 2012 National Conference on*, Feb 2012, pp. 1–4.

[3] D. C.-C. Dodds and C. Howarth, "The sonic pathfinder: an evaluation," vol. 78, no. 5, 1984, pp. 206–207.

[4] Heyes, "A polaroid ultrasonic travel aid for the blind," *Journal of Visual Impairment and Blindness*, vol. 76, pp. 199–201, 1982.

[5] J. Borenstein, "The guidecane-applying mobile robot technologies to assist the visually impaired," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 31, no. 2, pp. 131–136, Mar 2001.

[6] J. Barth and E. Foulhe, "Preview: A neglected variable in orientation and mobility," *Journal of Visual Impairment and Blindness*, vol. 73, no. 2, pp. 41–48, 1979.

[7] S. Shoval, J. Borenstein, and Y. Koren, "The navbelt-a computerized travel aid for the blind based on mobile robotics technology," *Biomedical Engineering, IEEE Transactions on*, vol. 45, no. 11, pp. 1376–1386, Nov 1998.

[8] S. Chumkamon, P. Tuvaphanthaphiphat, and P. Keeratiwintakorn, "A blind navigation system using rfid for indoor environments," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008. ECTI-CON 2008. 5th International Conference on*, vol. 2, May 2008, pp. 765–768.

[9] L. Ran, S. Helal, and S. Moore, "Drishti: An integrated indoor/outdoor blind navigation system and service," in *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04)*, ser. PERCOM '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 23–. [Online]. Available: http://dl.acm.org/citation.cfm?id=977406.978696

[10] "Navatar: Navigating blind users in indoor spaces using tactile landmarks, human computer interaction research university of nevada, reno."

[11] "http://www.the-scientist.com/?articles.view/articleno/41052/title/the-bionic-eye/."

[12] "Luca iocchi, [online]. available: http://www.dis.uniroma1.it/ iocchi/stereo/triang.html."

[13] N. H. Tan, N. Hamid, P. Sebastian, and Y. V. Voon, "Resource minimization in a real-time depth-map processing system on fpga," in *TENCON 2011 - 2011 IEEE Region 10 Conference*, Nov 2011, pp. 706–710.

[14] D. Chaikalis, N. Sgouros, D. Maroulis, and P. Papageorgas, "Hardware implementation of a disparity estimation scheme for real-time compression in 3d imaging applications," *Journal of Visual Communication and Image Representation*, vol. 19, no. 1, pp. 1 – 11, 2008.

[15] S. Perri, D. Colonna, P. Zicari, and P. Corsonello, "Sad-based stereo matching circuit for fpgas," in *Electronics, Circuits and Systems, 2006. ICECS '06. 13th IEEE International Conference on*, Dec 2006, pp. 846–849.

[16] D. Masrani and W. MacLean, "A real-time large disparity range stereo-system using fpgas," in *Computer Vision Systems, 2006 ICVS '06. IEEE International Conference on*, Jan 2006, pp. 13–13.