

Implementing Cepstral Filtering Technique using Gabor Filters

Sheena Sharma^{*}, Harshit Agarwal[†] and C.M. Markan^{*}

^{*} Dept. of Physics & Comp. Sc., Dayalbagh Educational Institute, Dayalbagh, Agra, India

[†] Faculty of Engineering, Dayalbagh Educational Institute, Dayalbagh, Agra, India

Received 3rd May 2011; accepted 3rd Feb 2012

Abstract

Cepstral filtering technique is applied on an interlaced image, the pattern similar to that which is found in layer IV of Primate Visual Cortex. Unless the signals from left and right eyes are placed simultaneously, the disparity cannot be detected. Therefore, it has a great significance in the sphere of stereo vision. It involves Power spectrum in computation, which is square of absolute of Fast Fourier Transform (FFT), is a complicated and hardware unfriendly. This paper shows the estimation of the Cepstral technique using a set of Gabor filters. The Ocular Dominance Column pattern analysis by the Gabor function is comparable to the perception in the human visual and makes the algorithm closer to biology. We propose an algorithm in which Gabor filters, instead of Power Spectrum, are applied to an interlaced image in the Cepstral algorithm. This scheme makes it hardware friendly as it gives the flexibility of working with modules which can be imitated in hardware. Building a FFT module is a tough task in analog circuit but determining Gabor Energy, an alternative to it, can be achieved by elementary circuits. The Phase, Energy Models and other methods use multi-lambda Gabor filters to compute disparity. The proposed method uses sum of absolute difference to choose a single Gabor filter of appropriate lambda that fits to find the disparity. The algorithm inherits the quality of both Gabor filter and Ocular Dominance Pattern and hence a biologically inspired and suitable for hardware realization. The proposed algorithm has been implemented on the test data image. A hardware scheme has also been proposed that can be used to estimate disparity and the idea can be extended in building complex modules that can perform real time - real image operations with a handful of resources as compared to employing complex digital FPGAs and CPLDs.

Key Words: Computer Vision, Image Analysis, 3D Reconstruction.

1 Introduction

BIOLOGICAL stereo vision computes the depth of an object from the disparity in positions of points matched between the left and right eye images of the object. The visual signal is captured by the photoreceptors of the retina and the signal is sent through the Lateral Geniculate Nucleus to the Primary Visual Cortex (PVC). Before PVC, the signals are strictly monocular and only in PVC, the left and right eye views of the scene interact, and this interaction begins with the formatting of binocular data as thin strips of cortex, which receive terminals from left and right eye, forming the pattern known as Ocular Dominance Pattern. Stereo vision or stereoscopic vision probably evolved as means of survival. With this, we can see where objects are in relation to our own bodies with much greater precision especially when those objects are moving toward or away from us in the depth dimension.

Bio-inspired systems aim to extract some interesting features from living beings, such as adaptability and fault tolerance, for including them in human-designed devices. In engineering and science, one can find several examples of bio-inspiration: airplane's wing has been inspired from birds, sonar and radar takes direct inspiration from bats' navigation system. The biological vision systems routinely accomplish complex visual tasks such as object recognition, obstacle avoidance, target tracking, which continue to

Correspondence to: <sheenasharma11@gmail.com>

Recommended for acceptance by <João Manuel Tavares>

ELCVIA ISSN: 1577-5097

Published by Computer Vision Center / Universitat Autònoma de Barcelona, Barcelona, Spain

challenge artificial systems. If any cell in the brain is dead, other cell takes over the dead cell and brain works in the normal way. Any bio-inspired system must be any day superior to any artificial method.

In computer vision, binocular disparity refers to the same difference seen by two different cameras instead of eyes. There are many computational models for disparity estimation, which have been proposed, with different approaches such as-area-based approach [1], [2] and [3], feature-based [4], [5]. There are different classes of algorithms and out of which, some are motivated from biological functioning of stereoscopy, such as windowed Cepstral filter, which is operated on an interlaced image, inspired by structure of Ocular Dominance Column in PVC (ODP) [6] and [7]. Other methods are Phase [8] and Energy method [9], [10] and [11], involve Gabor filters to decode disparity. The Gabor filters are the band pass filters [12] and [13] which has both limited spatial width and finite bandwidth and whose space is similar to the receptive field profile of simple cells in primate visual cortex [13]. It is important to realize that such an algorithm does not require that the phase be made explicit anywhere in the visual system [8]. Cepstral filtering method is strongly motivated by architectural features of visual cortex. It provides a computational justification for the existence of the columnar interlacing. Cepstral technique involves Power spectrum in computation, which is square of absolute of Fast Fourier Transform (FFT), is a complicated and hardware unfriendly.

This paper gives a new outlook to the existing algorithm and presents the algorithm which is close to the biological functioning to find stereoscopic disparity. Here, the power spectrum is estimated with the Gabor filters, which depicts the receptive fields, present in the Cortex. Cepstral estimation with Gabor filters on the ODC pattern makes the system closer to biology and an interesting algorithm. Further, the paper also provides the methodology to replace FFT with the simple and hardware friendly filter (Gabor filter) in the Cepstral technique to make a possible hardware implementation of the algorithm.

2 Theoretical Background

2.1 Cepstral Filtering techniques

Cepstral filtering is a technique, which was developed to measure echo, which is a shifted version of signal. Now the Cepstral filtering technique is widely used in signal and speech processing and extended to image processing. Cepstrum of a signal is Power Spectrum of Log of its Power Spectrum and is operated on an interlaced image. Yeshurun and Schwartz [6] claim that Cepstral filter can be implemented using a set of band pass filters similar to those found in the visual cortex. The authors append the left image with the right image. Let an interlaced image $f(x, y)$ be composed of a single columnar pair consists of an image patch $s(x, y)$ (the left image- $s(x, y) * \delta(x, y)$) and a shifted patch butted against it (the right image - $s(x, y) * \delta(x-D, y)$). The interlaced image can be represented as follows.

$$f(x, y) = s(x, y) * \{\delta(x, y) + \delta(x - D, y)\} \quad (1)$$

Cepstrum is defined as:

$$\text{Cepstrum} = |F(\ln\{|F(f(x, y))|^2\})|^2 \quad (2)$$

Thus, we find the disparity of the patch by locating the largest delta function. The logarithm part separates the disparity signal from the image signal in the final output and makes the Cepstral filter to be non-linear [14]. The technique is unique, as it is the method, which operates on a pattern similar to Ocular Dominance Pattern showing the similarity with biological working. But the method is difficult to realize in hardware.

Cepstral algorithm consists of functions, such as power spectrum (square of absolute of FFT), which is a complicated function to implement in hardware. The solution to this is to find an easy way to estimate power spectrum, which is a simple and easy to realize in hardware.

2.2 Sum of Absolute Differences

Sum of absolute difference is widely used, an extremely simple algorithm for finding the correlation between image blocks. It works by taking absolute difference between each pixel in the original block and the corresponding pixels in the corresponding block used for computation. The equation is as follows:

$$SAD(x, y, d) = \sum_{x,W} |I_{left}(x, y) - I_{right}(x - d, y)| \quad (3)$$

2.3 Gabor Filters

Gabor filters have been used in many applications, such as texture segmentation, target detection, fingerprint matching, edge detection, image coding and image reconstruction. It is a filter that is created by modulating sinusoidal with a Gaussian. Gaussian function acts as a window and decides the extension of the Gabor function. This filter is given by equation (4), [15]:

$$g(x, y; \lambda, \theta, \varphi, \sigma, \gamma) = \exp\{-(x'^2 + y'^2 \gamma'^2)/\sigma^2\} \cdot \cos(2\pi(x'/\lambda) + \varphi) \quad (4)$$

where

$$x' = x\cos\theta + y\sin\theta \text{ and } y' = -x\sin\theta + y\cos\theta \quad (5)$$

The parameters used in the above equation for g are explained below:

1. λ (lambda) specifies the wavelength of the cosine factor of the Gabor filter.
2. θ (theta) specifies the orientation of the normal parallel stripes of the Gabor filter. The valid values are between 0 and 360.
3. φ (phi) is the offset of the cosine factor and determines the symmetry of the Gabor filter. Valid values are between -180 and 180. The values 0 and 180 correspond to centre-symmetric ‘centre-on’ and ‘centre-off’ function, respectively, while -90 and 90 correspond to anti-symmetry functions.
4. σ (sigma) is the standard deviation of the Gaussian factor and determines the size of its receptive field.
5. γ (gamma) is called the spatial aspect ratio and specifies the ellipticity of the Gaussian factor. Default value is 0.5.

The even Gabor filter is obtained by putting $\varphi = 0^\circ$ and odd Gabor filter by putting: $\varphi = 90^\circ$. It has been seen that most of the cells can be combined in pairs, one cell of each pair has even symmetry and the other is odd symmetry. This observation can be modelled by a cosine function and sine function as wave be the chosen function [16]. Gabor filter can be combined as follows:

$$Gabor = gabor_{even}(Gc) + gabor_{odd}(Gs) \quad (6)$$

where Gc and Gs are symmetric and asymmetric Gabor filters. The Gabor Energy is a combination of symmetric and asymmetric Gabor filter.

$$Gabor\ Energy = \sqrt{Gc^2 + Gs^2} \quad (7)$$

2.4 Relation between Gabor Filters and FFT

Gabor filters are closely related to the Fourier transform. In fact, the complex exponential component of the filter is actually identical to the kernel of the Fourier transform [17]. Further, the Gaussian window improves the Cepstral output [1]. Hence, the Fourier Transform of the Gaussian window would be equivalent to the Gabor filtering. There is another relation between Gabor filter and FFT. Power Spectrum, which is square of absolute of FFT is closely related to Gabor Energy [18].

$$Power\ spectrum = (Gabor\ Energy)^2 \quad (8)$$

Now there are two ways in which power spectrum can be estimated using Gabor filters. In our paper, we use the relation between Power Spectrum and Gabor Energy, because the computational steps are less as compared to the other relation. The absolute is part of Power Spectrum, which needs not to calculate separately if calculating the Gabor Energy. But if we use the kernel of Fourier transform, then the step would increase to calculate the absolute and squaring to estimate power spectrum.

3 Proposed algorithm: Cepstral filtering using Gabor filters

Cepstral filtering is a technique, which is applied within windows which spans an ocular dominance column pair. But then FFT, which is part of Power Spectrum, is a complicated function to realize in hardware. The power spectrum is calculated using Gabor filters instead of FFT by using the relation between Gabor Energy and Power Spectrum. In this paper, we propose an algorithm which is more close to biological functioning, in which Cepstral filtering technique is implemented using Gabor filters, which makes it hardware friendly. In the algorithm, there are two images, left and right images. Set the left image as the original image and the right image as the matching image. Compute the sum of absolute difference on each pixel of left and right image to determine the matching range. Find out the best matching range based on sum of absolute difference. Further, there is not a single lambda or central frequency that will fit all possible images for a given stereoscopic system. Firstly because the modulating frequency is related to the maximum disparity and it will be dependent on the contents of the scene. Secondly the sigma is related to the amount of smoothing. Therefore, it can be expected that different regions in the scene will be better processed with different combination of modulating frequency and sigma. Phase and Energy models use multiple Gabor filters of different values of modulating frequency are used to find the disparity. In this proposed algorithm, appropriate wavelength of Gabor filter is applied. This reduces the use of multiple Gabor filters on a single pixel, unlike other methods which use multiple Gabor filters. Then apply combination of selected even and odd Gabor filters to a window, comprises of left and right patch placed next to each other to calculate the Gabor energy. Replace Power Spectrum with the calculated Gabor Energy in Cepstral algorithm and generate disparity map. The contribution of this paper is to use Gabor filtering instead of Power Spectrum in order to implement in hardware. And also, the use of single selected Gabor filter with contrary to Phase and Energy method. The proposed algorithm is biological plausible as it involves gabor filters (receptive fields) applied to windowed interlaced image (Ocular Dominance Column) with usage of selected a single Gabor

filter which makes the algorithms simple and suitable for hardware realization. Further, we have proposed an equivalent analog VLSI implementation of the algorithm which can be used to estimate disparity and the idea can be extended to real world images. Two stereo images, left and right image are taken and patched together and their electrically equivalent voltages are fed to the circuit. This could be done with help of CCDs or other equivalent transducers. The electrically equivalent image is convolved with a Gabor filter of appropriate wavelength (lambda). We have convolved the incoming image with 7 points even and odd Gabor filter. After the convolution, power spectrum of image is calculated using Gabor filters. Following this is the logarithmic equivalent of the voltages. The log equivalents of voltages are again convolved with the Gabor and again the power spectrum is determined. Observing this power spectrum we can estimate disparity. The distance between the voltage peaks in the left image and right image is the disparity.

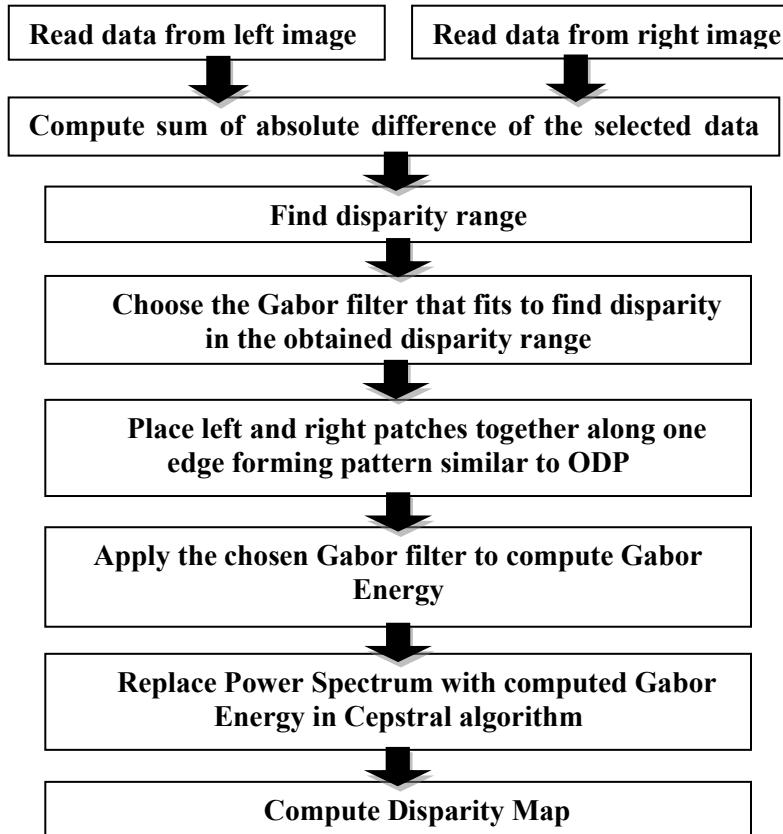


Figure 1: Flow chart to reconstruct disparity.

The algorithm of computing Cepstral filtering technique using Gabor filters:

1. Convolve even and odd Gabor filters with the spliced image, i.e., a window comprises of left and right image spliced along one edge. Compute Gabor Energy from the real and imaginary values obtained from the convolution. Square it to compute Power Spectrum.
2. Logarithmic Transformation of vector from step 1.
3. Now repeat step 1 with the spliced image vector being replaced by vector computed from log transformation from step 2. Doing so we have computed Cepstrum from which disparity could be inferred.

4 Experiment

We implement the proposed algorithm to stereo images, which has been widely utilized in other stereo algorithms. The website [19] provides the used stereo image and ground disparity map. Further, the obtained disparity map is refined by using diffusion technique. The bad-match percentage error measure quantitatively evaluates the performance of a stereo algorithm. The Gabor method does seem to produce disparity maps with high level of noise, but the use of bio-inspired functions come at the cost of accuracy. This paper focuses only on bio-inspired and hardware friendly functions. In the proposed work, the Gabor filters replace the power spectrum to make the algorithm closer to the biological function and to make a model suitable for hardware implementation. The algorithm does not compete on either speed or accuracy, just focus on the bio-function such as Gabor filter and interlaced pattern. The highly efficient algorithms make use of sophisticated method makes the algorithm to produce accurate results, and can-not be compared with the bio-inspired algorithms. For example, if we compare brain with the calculator. The calculator gives precise result of any complicated calculation in no time, but is not better than the functioning of the brain.

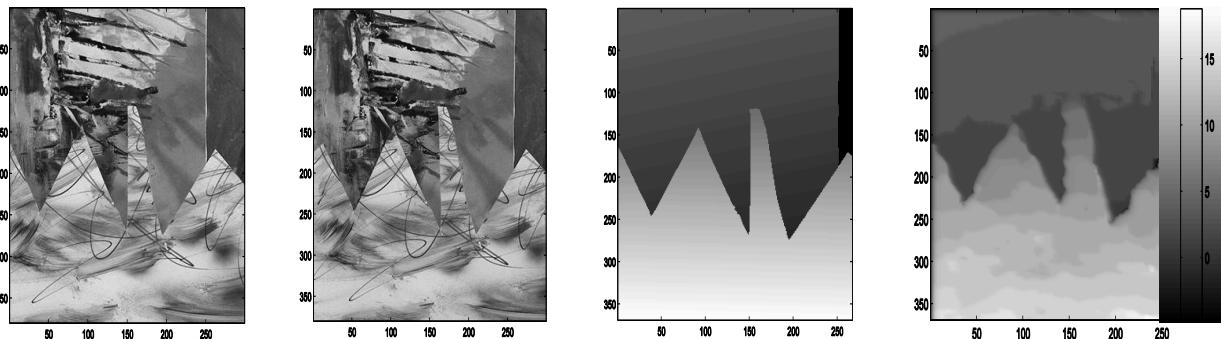


Figure 2: (a) Stereo image (sawtooth)

(b) True Disparity and Obtained disparity map

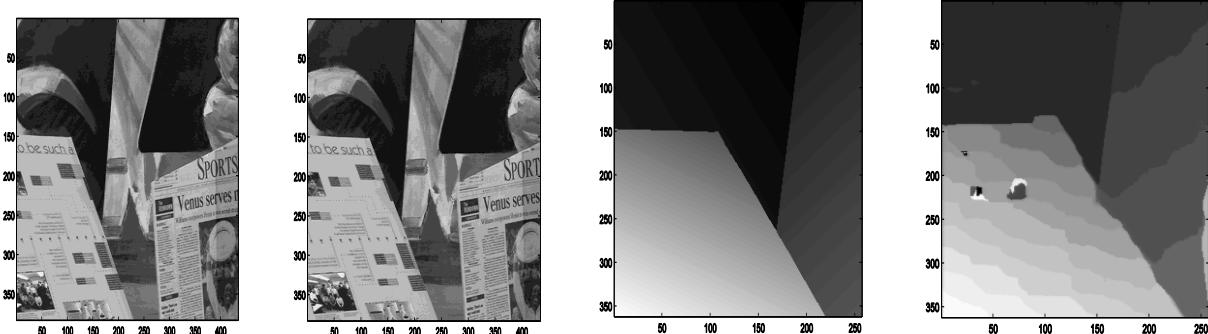


Figure 3: (a) Stereo image (venus)

(b) True Disparity and Obtained disparity map

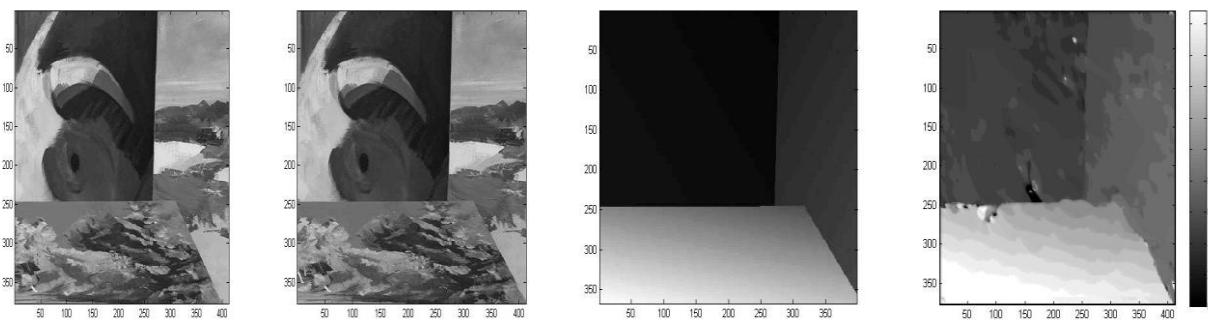


Figure 4: (a) Stereo image (Bull)

(b) True Disparity and Obtained disparity map

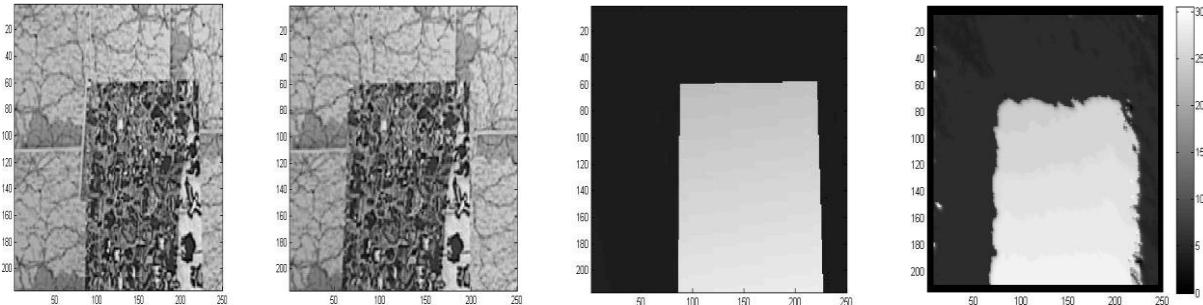


Figure 5: (a) Stereo image (Map) (b) True Disparity and Obtained disparity map

Test Stereo Images	Error (%)
Sawtooth	2.23%
Venus	2.91%
Bull	3.45%
Map	2.04%

The Cepstral technique is the only algorithm, which has a likeness with the biological phenomenon, as it is applied on a pattern similar to Ocular Dominance Column (ODC) in Primate Visual Cortex (PVC). It is already proved that unless the information from left eye and right eye are placed together, the disparity or depth cannot be estimated. The Fourier transformed of the windowed signal can also be thought as local power spectrum. The local power spectrum is defined as a square modulus of the Fourier transform of the product between the signal and a window function, which is closely related to the Gabor energy. The filter results of a symmetric (even part) and anti symmetric filters (odd part) can be combined in a single quantity, called as Gabor energy. This feature is related to a model of complex cells in Primary Visual Cortex.

Therefore, it won't be wrong in concluding that the Cepstral technique using band pass filters (Gabor filters) applied on the Ocular Dominance pattern makes the algorithm close to a - 'a bio-inspired algorithm'. It gives an essence of the Gabor filters, which depict the receptive fields and Ocular Dominance Pattern.

The proposed algorithm shows the superiority behavior as this algorithm is a combination of the Ocular Dominance Column and Band pass filters. On comparing the proposed algorithms with the algorithms dealing with the Gabor filters, the proposed algorithm reduces the use of multiple Gabor filters on a single pixel, hence reducing the computational time. The algorithm has proposed a methodology to replace the power spectrum or FFT with the band pass filters to make it a simple algorithm to realize in an analog hardware.

5 Hardware Impelmentation

Analog VLSI can implement algorithms by decomposing them into numerous, simple, parallel computational units. Although the computational units are simple, highly complex and non-linear behaviors can still be obtained from them, These circuits have several advantages over implementing logic on digital as the former are cheaper, faster resource efficient.

5.1 Hardware for Gabor and image convolution

For the imaginary and real Gabor filters we have considered 7- discrete points which give equivalent 1-d even Gabor profile. We chose a disparity range -2 to 2. The values of Gabor filter used are:

$$\text{Gabor filter (imaginary)} = [-0.0498 \quad 0.4103 \quad -0.6952 \quad 0 \quad 0.6952 \quad -0.4103 \quad 0.0498]$$

$$\text{Gabor filter (real)} = [-0.0258 \quad 0.1862 \quad -0.6099 \quad 0.9059 \quad -0.6099 \quad 0.1862 \quad -0.0258]$$

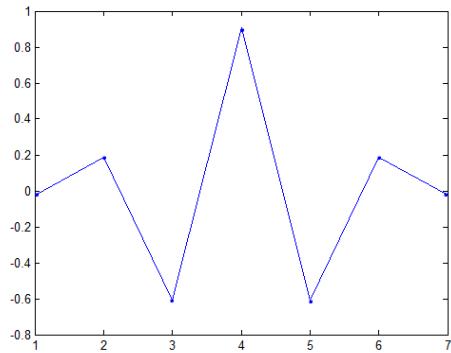
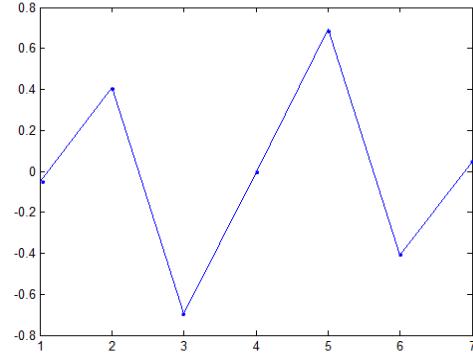


Figure 6: (a) Even Gabor filter



(b) Odd Gabor filter

Basically convolving the image with these filters is equivalent to weighted sum of voltages and shifting the inputs and again convolving with the new weights. The weights are decided by the corresponding Gabor filters. The input conductance of the Gabor cells is made equal to the even and odd Gabor's values. The even/odd Gabor cells are realized using opamp as summing amplifier.

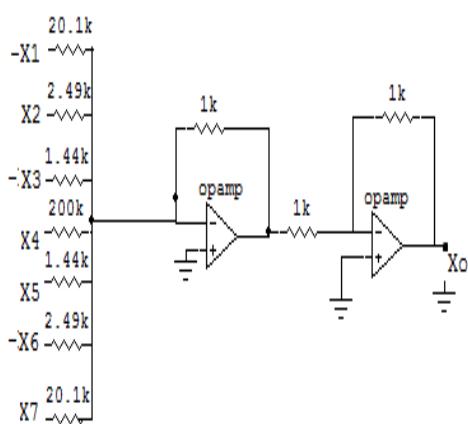
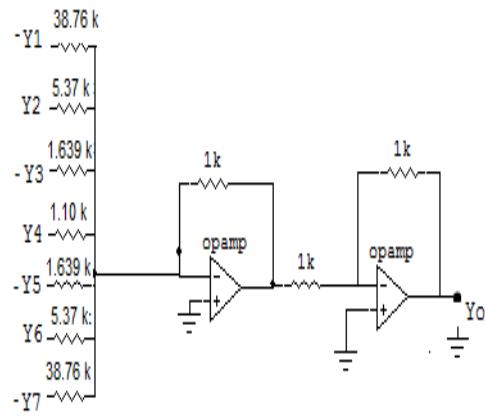


Figure 7: (a) Odd Gabor Cell



(b) Even Gabor Cell

The output voltage of odd Gabor cell, X_o is

$$X_o = (-0.049X_1 + 0.410X_2 - 0.695X_3 + 0.0005X_4 + 0.695X_5 - 0.410X_6 + 0.049X_7)$$

$X_1, X_2, X_3, X_4, X_5, X_6, X_7$ are the presented inputs to the gabor cell. Negative voltages in the circuit basically refer to presenting inverted voltage.

And similarly the output voltage of even Gabor cell, Y_o is

$$Y_o = (-0.0258Y_1 + 0.1862Y_2 - 0.6099Y_3 + 0.9059Y_4 - 0.6099Y_5 + 0.1862Y_6 - 0.0258Y_7)$$

This will perform just one convolution operation so we need to shift the Gabor cell over various inputs. Instead of physically shifting the Gabor, we have replicated these Gabor cells and they are fed by various sets of input data. And this is how we have realized a convolution unit. So these gabor cells are replicated since convolution needs shifting of inputs as well, so 10 such cells were replicated for convolution with odd gabor and 10 for even gabor filters respectively. The convolution unit is fed with inputs which are I1, I2, I3, I4, I6, I7, I8, I9, I10, equivalent voltages of the test image. '0' refers to the input pin connected to ground. The scheme is depicted below.

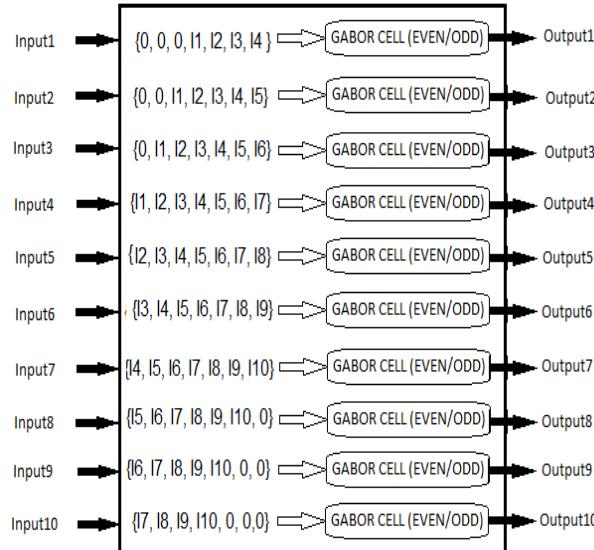


Figure 8: Convolution Scheme

5.2 Input voltage squaring circuit

The squaring circuit consists of a differential amplifier and the flipped voltage follower. The differential amplifier, consisting of the transistors M3 and M4, creates the squaring signal while the transistor M1. And M2 act as flipped voltage follower [20].

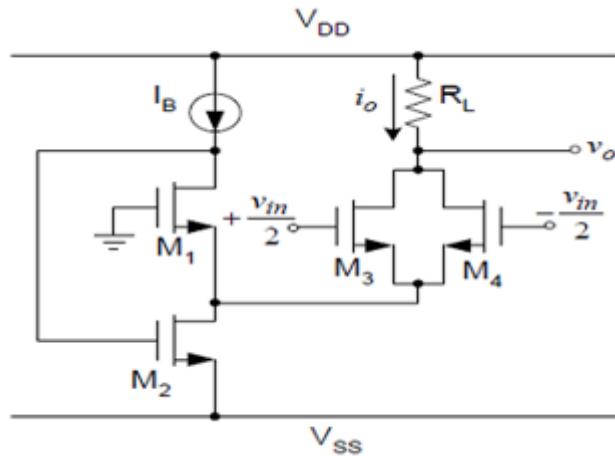


Figure 9: FVF Squaring circuit

For the purpose of squaring, the squaring module was modified a bit to suit our requirement. The input voltage V_{in} is split into $+V_{in}/2$ and $-V_{in}/2$ and then fed to FVF squaring circuit. The corresponding output voltage can be measured at the output node.

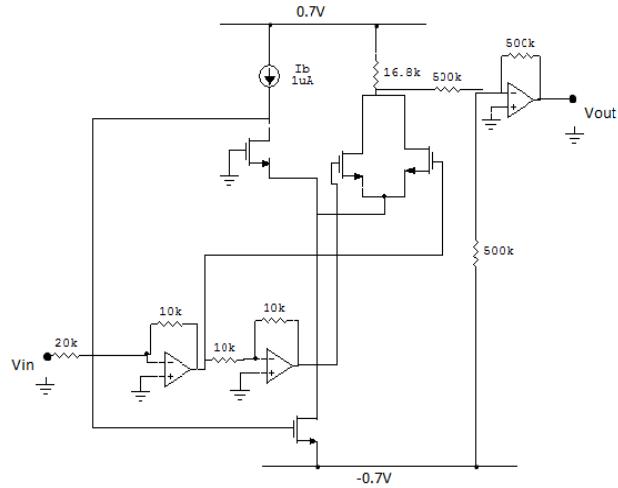


Figure 10: Modified Squaring Circuit

5.3 Logarithmic Transformation

Once the power spectrum has been determined there is a need to compute its natural log. Though there are several circuits available for determination of log of voltages, but in our test cases the voltages were falling in the range 50mV to 1V, so most of the circuits failed to deliver the correct result. For a wider range it can be closely approximated with a higher order polynomial. Since we wanted a simple circuit with only handful of resources we estimated it with a quadratic equation.

The equivalent function, $f(x) = -3.1*(x^2) + 5.7*x - 2.8$

Both the profile have been plotted and showed in figure 11 (blue-log profile, cyan-estimated polynomial) and figure 12 presents a scheme to implement equivalent log circuit.

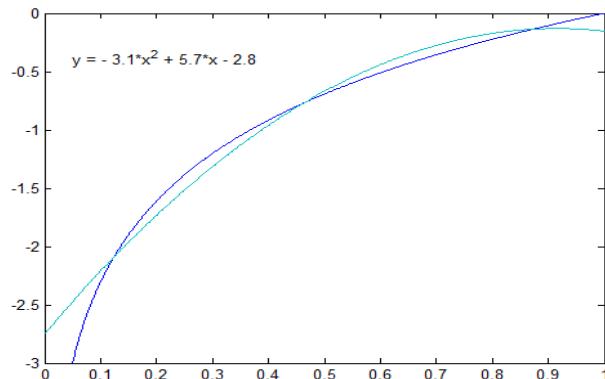


Figure 11: Comparison of log function and estimated polynomial

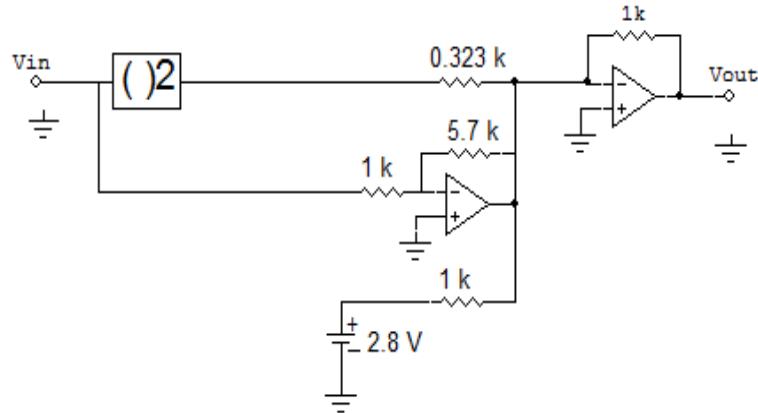


Figure 12: Equivalent logarithmic Converter

5.4 Test Case

We performed our operations on a test image, shown below:

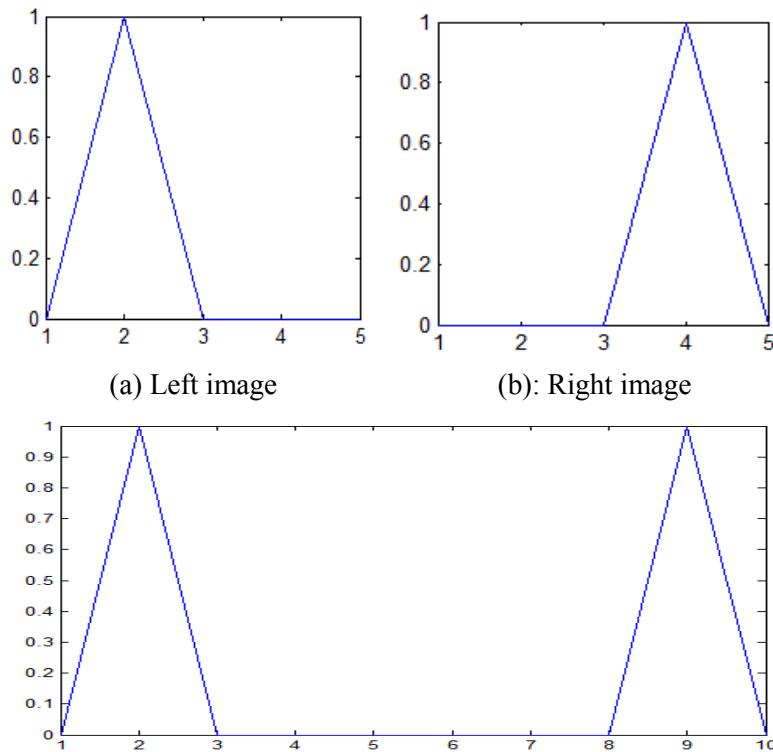
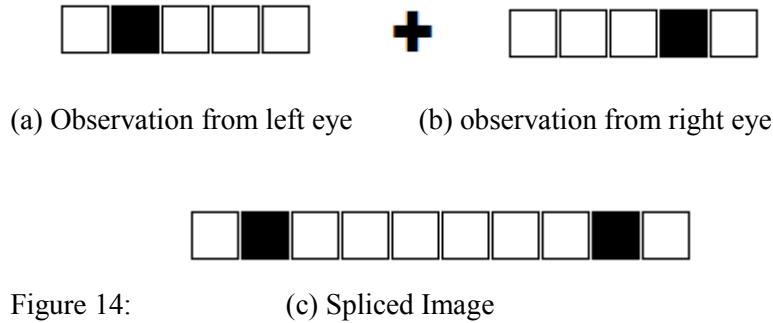


Figure 13: (c) Spliced Image

The Voltage equivalent of this image are $0,1,0,0,0$ V for the left image, shown in Fig 13.a and $0,0,0,1,0$ V for the right image, shown in Fig 13.b. Therefore the equivalent spliced image is $(0,1,0,0,0,0,0,0,1,0)$ V shown in fig 13c. Thus disparity is 2 in the given case, as the image is shifted by distance '2' in horizontal direction.

It can be also understood as 5 pixel image with black pixel's electrical value be 1V and white pixel's electrical voltage value be 0V,



5.5 Experiment

To test the algorithm we simulated the algorithm on Matlab as well as the equivalent analog VLSI implementation was simulated on T-Spice and from both the simulations the final vectors were observed. The observations are plotted with continuous curve representing result from Matlab simulation and broken curve plotted according to spice simulation. Both Matlab and T-Spice produced logically same results. Disparity is half the distance between the two peaks and is 2.5 from the graph. Since the disparity is always an integer value it was observed that the disparity was ‘floor’ value of the result, which was the case for other test cases as well. We tried our circuit for another test cases having disparity within the range -2 to 2 and correct result was observed.

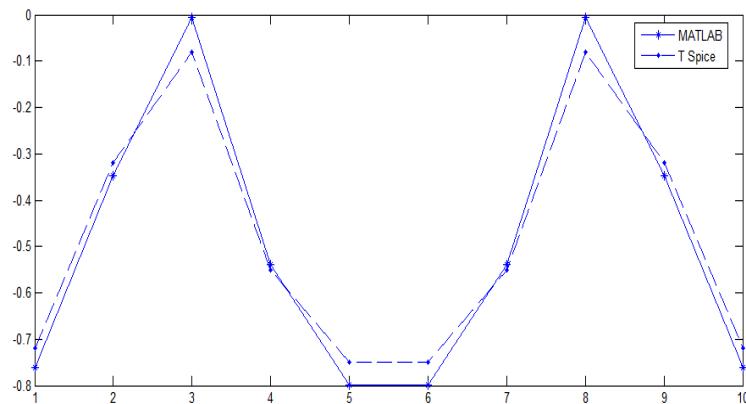


Figure 15: Final resultant vectors in Matlab and TSpice.

6 Conclusion

The paper presents a technique for matching stereo images in the framework of Cepstral algorithm. The contribution of the paper is related to the use of Gabor filtering instead of Power Spectrum in order to make possible hardware implementation. Disparity is computed by using paired even and odd Gabor filters. The used scheme makes the algorithm close to biological functioning. The presented algorithm gives the essence of Gabor filters, which depicts the receptive fields and Ocular Dominance Pattern. Its usage of simple functions makes the algorithm suitable for hardware realization. Sum of absolute difference is used to select a single Gabor filter. It reduces the use of multi Gabor filters on a single pixel to estimate the disparity. Our work is to find out the model which depicts the brain to reconstruct the stereoscopic disparity and should be

hardware-friendly. The hardware scheme has also been presented for a simple case. Further the improvement of the new algorithm is our current work to achieve more satisfactory results and to implement hardware for real images.

References

- [1] L. D. Stefano, M. Marchionni, S. Mattoccia, G. Neri, "A fast area-based stereo matching algorithm", 15th International Conference on Vision Interface, vol 22, pp 983-1005, 2002.
- [2] J. Sun, H. Y. Shum, N. N. Zheng, "Stereo matching using belief propagation", Pattern Analysis And Machine Intelligence, vol 25, pp 787-800, 2003.
- [3] Y. Boykov, O. Veksler, , R. Zabih, "Fast approximate energy minimization via graph cuts", Pattern Analysis and Machine Intelligence, vol 23, pp 1222-1239, 2001.
- [4] X. Qin, I. Li, S. Tian, "A new image matching algorithm with modified greedy for remote sensing imagery", ISPRS Workshop on Service and Application of Spatial Data Infrastructure, 2006.
- [5] M. Gong, Yang, Yee-Hong, "Near real-time reliable stereo matching using programmable graphics hardware", IEEE Computer Vision and Pattern Recognition Conference, vol 1, pp 924-931, 2005.
- [6] Yeshurun, E. Schwartz, "Cepstral Filtering on a columnar Image Architecture: A fast Algorithm for binocular Stereo Segmentation", IEEE Transaction on Pattern And Machine Intelligence, vol 11, pp 759-767, 1989.
- [7] K.O. Ludwig, H. Neumann, B. Neumann, "Local Stereoscopic Depth Estimation Using Ocular Stripes", Lecture Notes in Computer Science, Proceeding of the Second European conference on computer Vision, vol 588, pp 373-377, 1992.
- [8] T.D. Sanger, "Stereo disparity computation using Gabor filters", Biol. Cybern, vol 59, pp 405-418, 1988.
- [9] I. Ohzawa, G.C. DeAngelis, R.D. Freeman, "Stereoscopic depth discrimination in the visual cortex: Neurons ideally suited as disparity detectors", Science, vol 249, pp 1037-1041, 1990.
- [10] I. Ohzawa, G.C. DeAngelis, R.D. Freeman, "Encoding of binocular disparity by simple cells in the cat's visual cortex", J. Neurophysiol, vol 75, pp 1779-1805, 1996.
- [11] I. Ohzawa, G.C. DeAngelis, R.D. Freeman, "Encoding of binocular disparity by complex cells in the cat's visual cortex", J. Neurophysiol, vol 77, pp 2879-2909, 1997.
- [12] D. Gabor, "Theory of Communication", J. Inst. Electr. Eng vol 93, 1946.
- [13] S. Marcelja, "Mathematical Description of the Responses of Simple Cortical Cells", J. Opt. Soc. Am, vol 70, 1980.
- [14] U. M. Leloglu, "Artificial Versus natural stereo depth perception", Hints from Life to AI. Edited by Ugur Halici, Metu, 1994.
- [15] N. Petkov, M.B. Wieling, "Gabor filter for image processing and computer vision", (On line), <http://matlabserver.cs.rug.nl/edgedetectionweb/index.html>.
- [16] V. S. Vyas, P. Rege, "Automated Texture Analysis with Gabor filter", GVIP Journal, vol 6, 2006.
- [17] M. W. Maimone, "Characterizing Stereo Matching Problems using stereo Spatial Frequency", PhD thesis, 1996.
- [18] T. Andrysiak, M. Choras's, "Image Retrieval Based On Hierarchical Gabor Filters", Int. J. Appl. Math. Comput. Sci., vol 15, pp 471-480, 2005.
- [19] Website: <http://vision.middlebury.edu/stereo/>
- [20] C. Sakuk, K. Dejhan: "Squaring And Square-Root Circuits Based On Flipped Voltage Follower And Applications", International Journal of Information Systems and Telecommunication Engineering, vol.1 (2010)

Appendices

a. TSPICE Code for Even gabor subcircuit

```
* subcircuit for gabor_cos
.subckt gabor_cos in1 in2 in3 in4 in5 in6 in7 in8 in9 gab_cos_out
R1 in1 nod1 38.76k
R2 in2 nod1 -5.37k
R3 in3 nod1 1.639k
R4 in4 nod1 -1.1k
R5 in5 nod1 1.639k
R6 in6 nod1 -5.37k
R7 in7 nod1 38.76k
Xinv 0 nod1 gab_cos_out opamp741
Rinv nod1 gab_cos_out 1k
Rout gab_cos_out 0 1
.ends gabor_cos
```

b. TSPICE Code for Odd gabor subcircuit

```
* subcircuit for gabor_sin
.subckt gabor_sin inp1 inp2 inp3 inp4 inp5 inp6 inp7 inp8 inp9 gab_sin_out
Rs1 inp1 node2 20.1k
Rs2 inp2 node2 -2.49k
Rs3 inp3 node2 1.44k
Rs4 inp4 node2 -200k
Rs5 inp5 node2 -1.44k
Rs6 inp6 node2 2.49k
Rs7 inp7 node2 -20.1k
Xinv2 0 node2 gab_sin_out opamp741
Rinv2 node2 gab_sin_out 1k
Rout2 gab_sin_out 0 1
.ends gabor_sin
```

c. TSPICE Code for Squaring Circuit

```
* subcircuit for squaring
.subckt square in_sc1 output_sq
r11_sc in_sc1 in_sc2 20k
r12_sc in_sc2 in_sc3 10k
r13_sc in_sc3 in_sc4 10k
r14_sc in_sc4 in_sc5 10k
X1_sc 0 in_sc2 in_sc3 opamp741
X2_sc 0 in_sc4 in_sc5 opamp741
Vdd_sc 1_sc 0 dc 0.7V
R1_sc 1_sc out_sc 16k
M3_sc out_sc in_sc5 4_sc source_node cmosn W=10um L=4um
M4_sc out_sc in_sc3 4_sc source_node cmosn W=10um L=4um
Ib_sc 1_sc 2_sc 1uA
M1_sc 2_sc 0_4_sc source_node cmosn W=10um L=1um
M2_sc 4_sc 2_sc source_node source_node cmosn W=15um L=2um
Vss source_node 0 dc -0.7V
X5_sc 0 n_sc output_sq opamp741
R5_sc source_node n_sc 500k
R6_sc out_sc n_sc 500k
R7_sc n_sc output_sq 500k
.model cmosn NMOS level = 3 tox = 5.7e-9 nsb = 1e17 gamma = 0.4317311 phi = 0.7 vto = 0.67 delta = 0 uo =
425.6466519 eta = 0 theta = 0.1754054 kp = 2.501048e-4 vmax = 8.287851e4 kappa = 0.1686779 rsh = 4.062439e-3
nfs = 1e12 tpg = 1 xj = 3e-7 ld = 3.162278e-11 wd = 1.232881e-8 cgdo = 6.2e-10 cgso = 6.2e-10 cgbo = 1e-10 cj = 1.81211e-3 pb = 0.5 mj =
0.3282553 ejsw = 5.341337e-10 mjsw = 0.5
.ends
```

d. TSPICE Code for Equivalent Logarithmic converter

```
*subcircuit for log
.subckt logf in_log out_log
X1_log in_log in2_log square
R1_log in2_log com_log 0.323k
R2_log in_log co_log 1k
R3_log co_log com_log 5.7k
V_log in_con 0 dc 2.8V
R4_log in_con com_log 1k
X2_log 0 co_log com_log opamp741
Rf_log com_log out_log 1k
X3_log 0 com_log out_log opamp741
.ends
```