International Journal of Computational Intelligence and Informatics, Vol. 5: No. 3,December 2015

Implementation of a High - Quality Image Scaling Processor

Jennifer Eunice.R
Department of Electronics and communication
Dr.SivanthiAditanar College of Engineering
Tiruchendur, India
jenni_259@yahoo.in

***Abstract*-A high quality algorithm is proposed for VLSI Implementation of an image scaling processor. Scaling is a process, which is non-trivial that involves a trade-off between efficiency, smoothness and sharpness. Enlarging an image (up sampling or interpolating) is generally common for making smaller images to fit in full screen mode. The proposed algorithm comprises a spatial sharpening filter, a clamp filter and a bilinear interpolator. The Spatial sharpening filter and a clamp filter are used together as a pre-filter to remove blurring and aliasing effect from the bilinear interpolator. To minimize the Memory buffers and computational complexity for the proposed design, a T-model and inverse T-model filter realization is done by both sharpening and clamp filters. By using the T – model and inverse T – model filter the memory buffer requirement is reduced to one-line-buffer.**

Keywords-Bilinear interpolator, Spatial Sharpening filter, Clamp filter, Combined filter.

I. INTRODUCTION

In recent days, different sizes of images are delivered to users from various multimedia sources such as Mobile phones, Digital camera, Internet. When the resolution of the received image is low then the user desires to magnify the image using a high resolution (HR) display devices. For obtaining a High resolution of an image we go for image scaling.

Image scaling is the process of resizing a digital image which is used in many fields ranging from consumer electronics to medical imaging. To enhance the quality of an image, various algorithms are used. They include Winscale algorithm, Curvature interpolation algorithm, Bi-cubic algorithm, Bilinear interpolation algorithm. According to the required computational resources and memory space the existing scaling methods are divided into two categories, which include low complexity scaling and high complexity scaling. The basic concept of image scaling is to resample a two-dimensional function on a new sampling grid (W. K. Pratt,1991). Already, many algorithms for image scaling are proposed. The simplest method is the **nearest neighbor**(S. Fifman, 1973), which samples the nearest pixel from original image. It is regarded as the zero-order sample-and-hold and has good high frequency response, but degrades image quality due to aliasing. The most widely used method is the **bilinear**(R.C.Gonzalez and R.E. Woods) which is regarded as the first-order sample-and-hold. In **bilinear**, the output pixel value changes linearly according to sampling position. There is more complex method called**bicubic**(H. S. Hou and H. C. Andrews). The weakness of **bilinear**and **bicubic**is blur effect causing bad high frequency response. Recently, many other methods using polynomial (S. Andrews and F. Harris), adaptive (N. Shezaf, H. Abramov-Segal, I. Sutskover, and R. Bar-Sella*et al*), or correlative property (H. C. Kim, B. H. Kwon, and M. R. Choi *et al*) have been proposed. However, these methods have complex computation comparatively. In another method (S. Carrato and L.Tenze) the scale ratio is fixed at powers of two. It prohibits the method from being used in screen resolution change requiring a fractional scaling ratio. The bi-cubic algorithm is not very easy to implement since it has high computational complexity, large memory requirements and high hardware implementation cost. Non – polynomial – based interpolation algorithms are Adaptive 2-D autoregressive modelling [1], Curvature interpolation [4], bilateral filter [5] and autoregressive model [2]. Though these methods enhance the quality of an image by reducing the blocking and aliasing artifacts, these methods have the characteristics of high memory requirement.

To achieve the demand of real – time image scaling applications some previous techniques have proposed low-complexity methods for VLSI implementation. Those previous techniques include Winscale algorithm [3] by *kim et al* and Edge oriented technique [2] by *Lin et al.*

ISSN: 2349 - 6363

In this proposed paper we use bilinear interpolation algorithm because it has low computational complexity and requires minimum memory space. The bilinear algorithm makes use of linear interpolation models to calculate unknown pixels that are widely used because of its computational efficiency and image quality. Though the result of this algorithm exhibits blurring edges and aliasing effect after scaling, these effects can be reduced by using combined filters.

II. LITERATURE SURVEY

A. Winscale : An Image scaling algorithm using Area pixel model

In this paper, we introduce Winscalealgorithm, a new resampling method, which uses domain filtering utilizing area coverage of original pixels for calculating new pixels of a scaled image. This method has good high frequency characteristics and better image quality than bilinearmethod. This method is suitable for digital display devices in many aspects. For example, it preserves the edge characteristics of an image well, can handle streaming data directly, and requires only small amount of memory: four line buffers rather than a full frame buffer.

B. VLSI implementation of an edge oriented image scaling processor

In this paper, we present an edge-oriented area-pixel scaling processor. To achieve the goal of low cost, the area-pixel scaling technique is implemented with a low - complexity VLSI architecture in our design. A simple edge catching technique is adopted to preserve the image edge features effectively so as to achieve better image quality. The proposed scaling processor can support floating-point magnification factor and preserve the edge features efficiently by taking into account the local characteristic existed in those available source pixels around the target pixel. Furthermore, it handles streaming data directly and requires only small amount of memory: one line buffer rather than a full frame buffer.

C. A novel image interpolation method using the bilateral filter

In this paper, we propose a novel interpolation framework in which denoising and image sharpening methods are embedded. In the proposed framework, the image is first decomposed using the bilateral filter into the detail and base layers which represent the small and large scale features, respectively. The detail layer is adaptively smoothed to suppress the noise before interpolation and an edge- preserving interpolation method is applied to both layers. Finally, the high resolution image is obtained by combining the base and detail layers. Experimental results show that the proposed algorithm outperforms the conventional methods while suppressing blurring and jagging.

D. A low-cost high-quality adaptive scalar for real-time multimedia applications

A novel scaling algorithm is proposed for the implementation of 2-D image scalar. The algorithm consists of a bilinear interpolation, a clamp filter, and a sharpening spatial filter. The bilinear interpolation algorithm is selected due to its having low complexity and high quality. The clamp and sharpening spatial filters are added as pre-filters to solve the blurring and aliasing effects produced by bilinear interpolation. Furthermore, an adaptive technology is used to enhance the effects of clamp and sharpening spatial filters.

TABLE I

COMPARISON TABLE FOR DIFFERENT SCALING ALGORITHMS IMPLEMENTED USING VLSI ARCHITECTURE

	Winscale Algorithm	***Edge oriented Method***	***Adaptive Algorithm***	***Bilinear Interpolation***
Frequency	65 MHz	200 MHz	280 MHz	280 MHz
Gate counts	29 K	10.4K	9.28K	6.08K
Simulation power	N/A	16.47Mw	9.6mW	6.45mW
Line Buffer	4 Lines	1 Lines	4Lines	1Lines

Table 1.Inference from literature survey

III. PROPOSED SCALING ALGORITHM

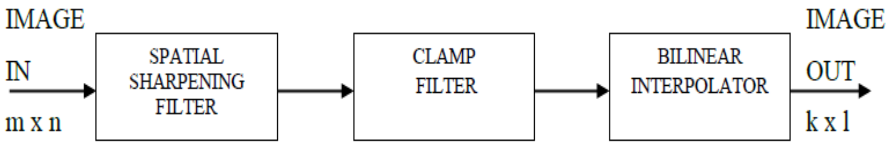


Figure3.1. Block diagram for proposed scaling algorithm

The proposed scaling algorithm consists of a spatial sharpening filter, a clamp filter and a bilinear interpolation. A spatial sharpening filter which acts as a High pass filter is used to remove the blurring effect from an image and Clamp filters which acts as a Low pass filter is used to reduce the aliasing artifacts produced in an image due to bilinear interpolation. Both the spatial sharpening filters [6] and clamp filters are used as pre-filters to minimize the blurring and aliasing artifacts produced by the bilinear interpolation. First, the input pixels of the original images are filtered by the sharpening spatial filter to enhance the edges and remove associated noise. Next, the filtered pixels are again filtered using clamp filters to smooth unwanted discontinuous edges of the boundary regions. Finally, the filtered pixels are passed to bilinear interpolating for up/down scaling. In order to reduce the computational resources and memory buffer, these two filters are combined together as a combined filter.

A. Spatial Sharpening Filter

The proposed scaling algorithm consists of a spatial sharpening filter, a clamp filter and a bilinear interpolation. A spatial sharpening filter which acts as a High pass filter is used to remove the blurring effect from an image and Clamp filters which acts as a Low pass filter is used to reduce the aliasing artifacts produced in an image due to bilinear interpolation. Both the spatial sharpening filters [6] and clamp filters are used as pre-filters to minimize the blurring and aliasing artifacts produced by the bilinear interpolation. First, the input pixels of the original images are filtered by the sharpening spatial filter to enhance the edges and remove associated noise. Next, the filtered pixels are again filtered using clamp filters to smooth unwanted discontinuous edges of the boundary regions. Finally, the filtered pixels are passed to bilinear interpolating for up/down scaling. In order to reduce the computational resources and memory buffer, these two filters are combined together as a combined filter.

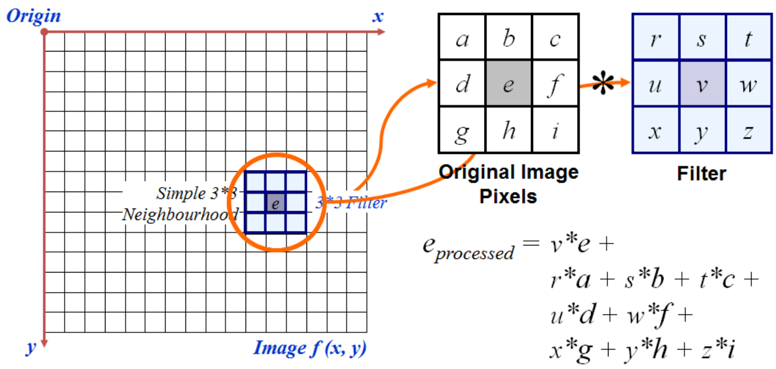


Figure3.1 Spatial sharpening filter method

The first order derivatives includes; i)Zero in areas of constant intensity, ii) Non-zero at the onset and end of an intensity step or ramp and iii) Non-zero along ramps of constant slope.
The first order derivative includes,

$$\frac{\partial y}{\partial x} = f(x+1) - f(x) \tag{1}$$

The seconddderivative must be, i) Zero in areas of constant intensity; ii) Non-zero at the onset and end of an intensity step or ramp; iii) Zero along ramps of constant slope.

$$\frac{\partial^2 y}{\partial x^2} = f(x+1) + f(x-1) - 2f(x) \tag{2}$$

B. Clamp Filter

The clamp filter is a kind of low pass filter, is a 2-D Gaussian spatial domain filter and composed of a convolution kernel array. The clamp filter is used to reduce aliasing artifacts and smooth the unwanted discontinuous edges of the boundary regions. Smoothing filters are used for blurring and noise reduction. Blurring may be implemented in pre-processing tasks to remove small details from an image prior to large object extraction.

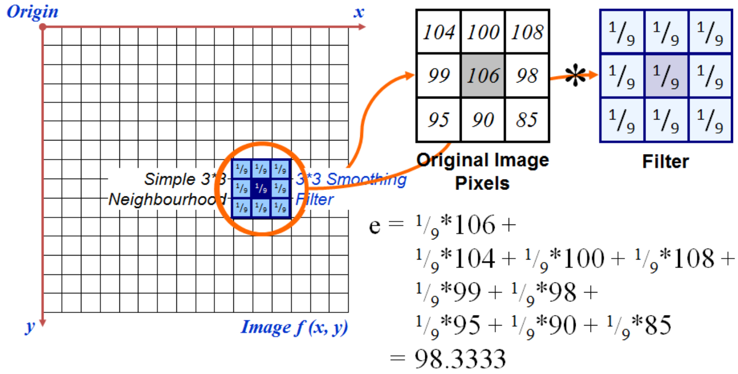


Figure 3.2 Smoothening filter technique

In order to reduce the complexity of a 3 x 3 convolution kernel T – Model and inverse T- Model convolution kernels are proposed for realizing both filters.

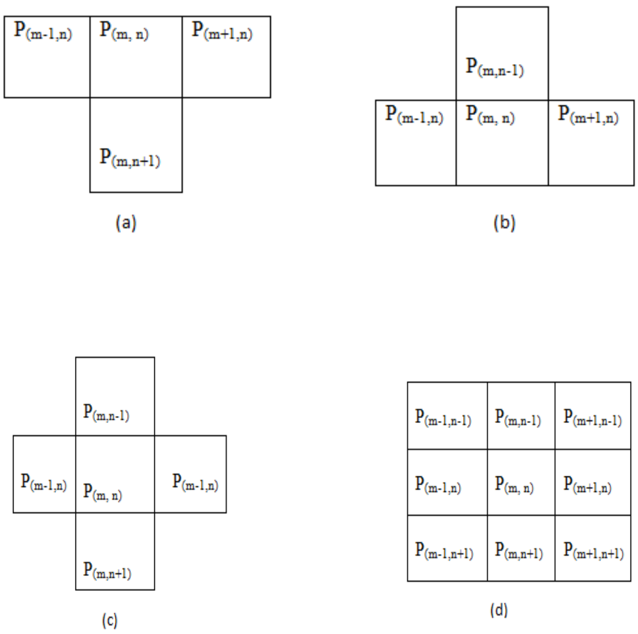


Figure3.3. Weights of the convolution kernels (a) & (b) T-model and inverse T-model convolution kernels (c) cross-model convolution kernels (d) 3 x 3 convolution kernel

The output of a smoothing (averaging or lowpass) linear spatial filter is the average of the pixels contained in the neighbourhood of the filter mask. Both these clamp and spatial sharpening filters can be represented by convolution kernels. High quality of image images can be obtained by using large kernels. But larger the size of convolution kernels will require large memory and hardware cost. For example a 6 x 6 convolution filter requires at-least a five-line-buffer memory and 36 arithmetic units whereas a 3 x 3 convolution filter requires only two-line-buffer memory and nine arithmetic units.

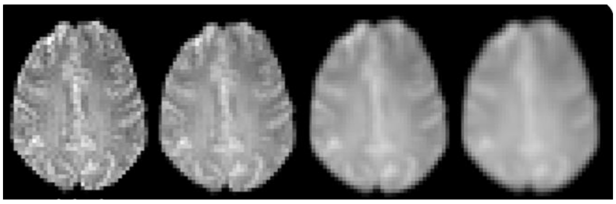


Figure3.4 Example of smoothened image

The T-model or inversed T-model filter is simplified from the 3 x3 convolution filter of the previous work [15], which not only efficiently reduces the complexity of the convolution filter but also greatly decreases the memory requirement from two to one line buffer for each convolution filter. The T-model and the inversed T-model provide the low-complexity and low memory- requirement convolution kernels for the sharpening spatial and clamp filters to integrate the VLSI chip of the proposed low-cost image scaling processor.

C. Bilinear interpolator

In proposed scaling algorithm, the input image is filtered by a sharpening spatial filter and then filtered by a clamp spatial filter again. Although the sharpening spatial and clamp filters are simplified by T- Models and inversed T-models, it still needs two line buffers to store input data or intermediate values for each T-model or inversed T-model filter. Thus, to be able to reduce more computing resource and memory requirement, sharpening spatial and clamp filters, which are formed by the T-model or inversed T-model, should be combined together into a combined filter as,

$$P'_{(m,n)} = [\, P^*_{(m,n)} \begin{bmatrix} -1 & S & -1 \\ 0 & -1 & 0 \end{bmatrix} / (S-3)\,] * \begin{bmatrix} 1 & C & 1 \\ 0 & 1 & 0 \end{bmatrix} / (C+3) \qquad (3)$$

Where S and C are the sharp and clamp parameters and $P(m,n)$ is the filtered result of the target pixel $P(m,n)$ by the combined filter. In the proposed scaling algorithm, the bilinear interpolation method is selected because of its characteristics with low complexity and high quality.

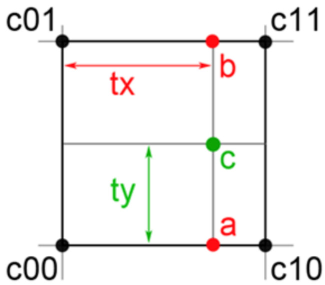


Figure 1.5 Bilinear Interpolation

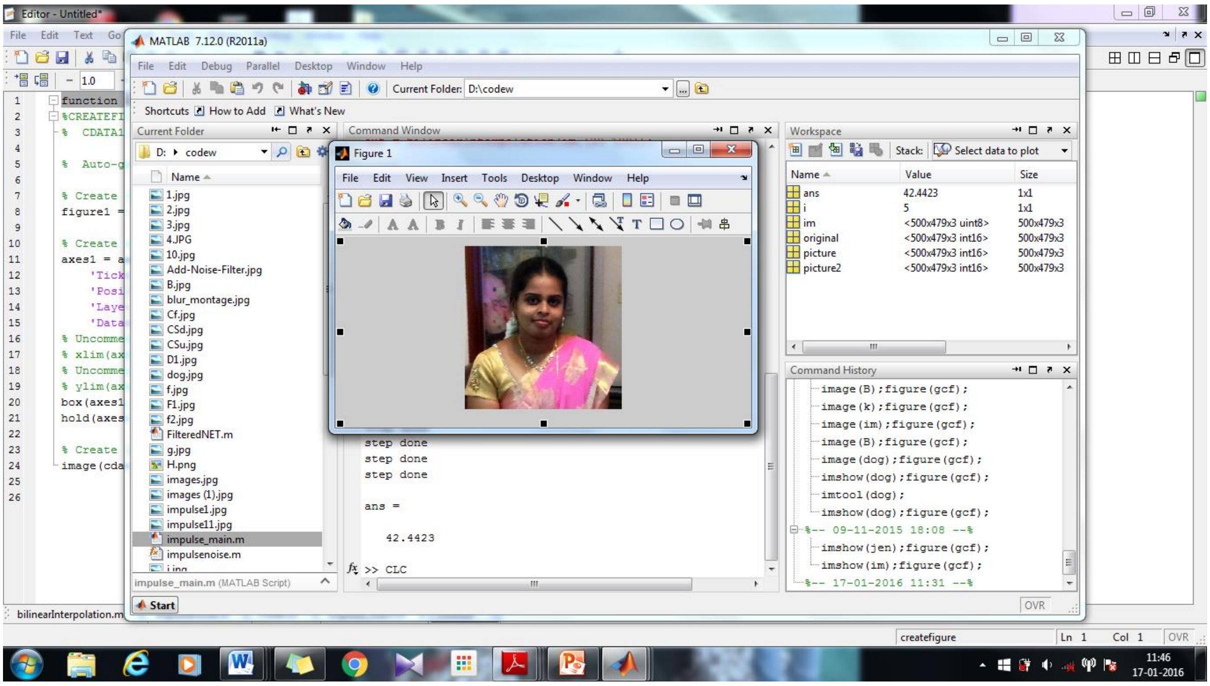
The bilinear interpolation is an operation that performs a linear interpolation first in one direction and, then again, in the other direction. The output pixel $P(k,l)$ can be calculated by the operations of the linear interpolation in both x- and y-directions with the four nearest neighbor pixels. Where $P(m,n)$, $P(m+1,n)$, $P(m,n+1)$, and $P(m+1,n+1)$ are the four nearest neighbor pixels of the original image and the dx and dy are scale parameters in the horizontal and vertical directions we can easily find that the computing resources of the bilinear interpolation cost eight multiply, four subtract, and three addition operations.

$$P_{(k,l)} = (1-dx) \times (1-dy) \times P_{(m,n)} + dx \times (1-dy) \times P_{(m+1,n)} + (1-dx) \times dy \times P_{(m,n+1)} + dx \times dy \times P_{(m+1,n+1)} \qquad (4)$$

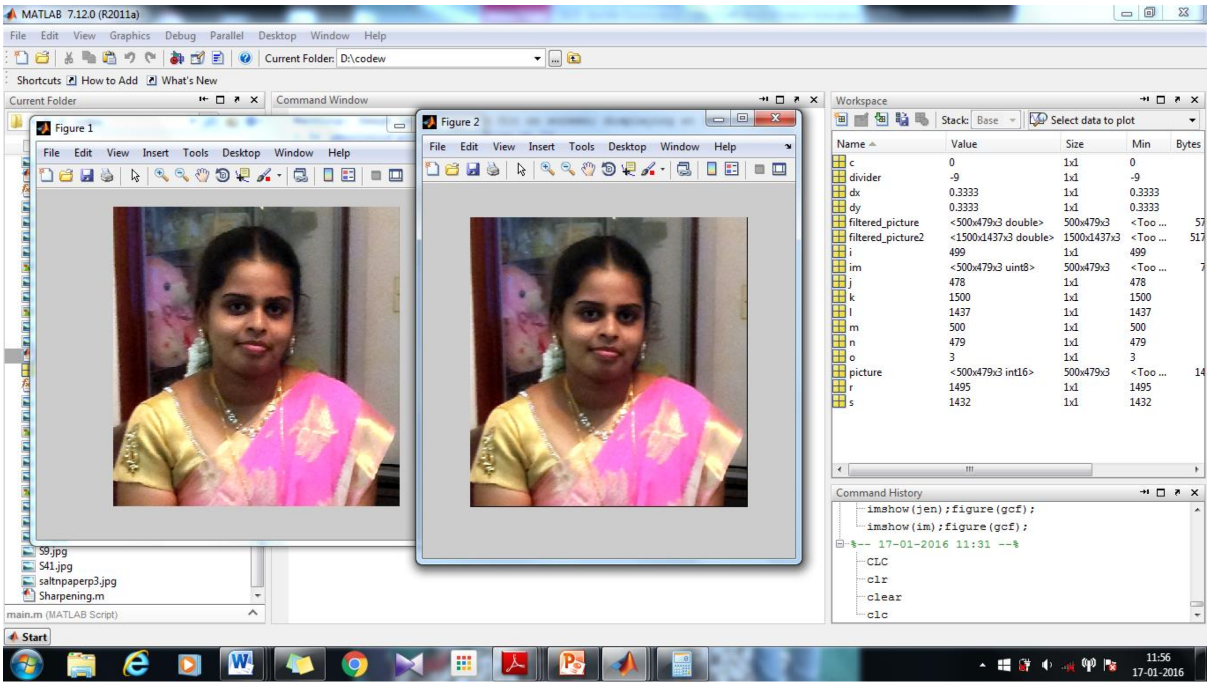
Thus, an algebraic manipulation skill has been used to reduce the computing resources of the bilinear interpolation.

IV. RESULTS AND DISCUSIONS

The blurring and aliasing artifacts produced by the bilinear interpolation are removed by adding the sharpening spatial filters and clamp filters as pre - filters**.** The Sharpening spatial filters and Clamp filters are realized by using a T- model and inverse T-model Filters. The proposed scaling algorithm is verified usingMATLAB R 2011(a) software .Figure.5.(a), the filtered image is scaled up to 1500 x1437 and scaled down to 499 x 478.In Figure.5.(b), the input image having a pixel resolution of 500x479 resolution with some noise, It is filtered out using combined filters. The output result obtained is free from noise when compared to the original image. The PSNR value of the given noise is about 42.4423. The output of k x l matrix is simulated using Modelsim.



(a)



(b)

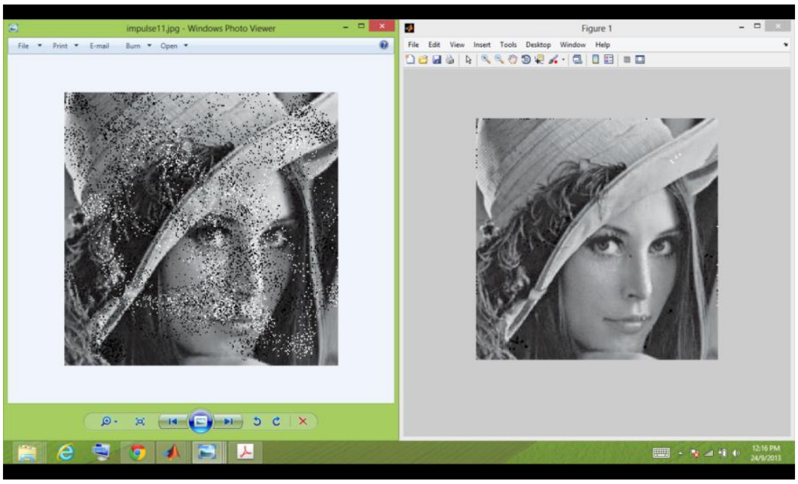


Figure 5(a). Output from combined filter,5(b). Output of scaled up/down image,5(c). Example of noise removal using combined filter

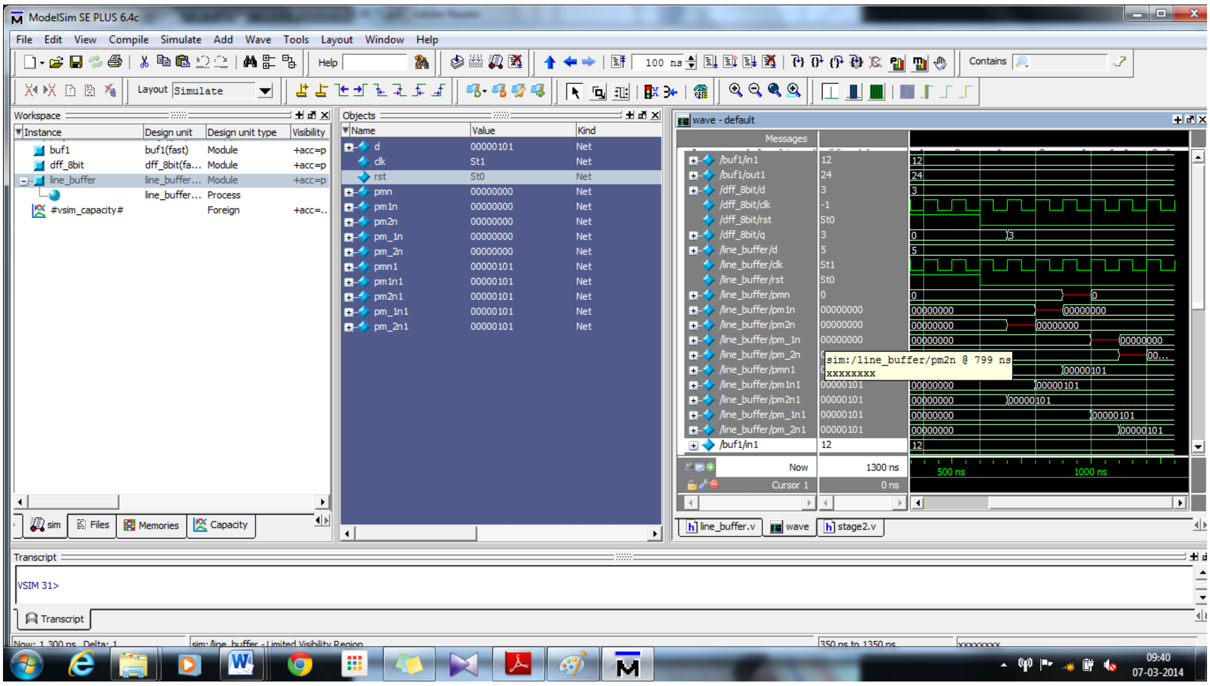


Figure5.1 Simulated result of the generated matrix using Modelsim

References

[1] S. L. Chen, H. Y. Huang, and C. H. Luo, "A low-cost high-quality adaptive scalar for real-time multimedia applications," *IEEE Trans. Circuits Syst.Video Technol.*, vol. 21, no. 11, pp. 1600–1611, Nov. 2011.

[2] P. Y. Chen, C. Y. Lien, and C. P. Lu, "VLSI implementation of an edge oriented image scaling processor," *IEEE Trans. Very Large Scale Integr.(VLSI) Syst.*, vol. 17, no. 9, pp. 1275–1284, Sep. 2009.

[3] C. H. Kim, S. M. Seong, J. A. Lee, and L. S. Kim, "Winscale : An image scaling algorithm using an area pixel model," *IEEE Trans. Circuits Syst.Video Technol.*, vol. 13, no. 6, pp. 549–553, Jun. 2003.

[4] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int'l Conf. Computer Vision (ICCV'98)*, 1998.

[5] C. C. Lin, M. H. Sheu, H. K. Chiang, W. K. Tsai, and Z. C. Wu, "Real-time FPGA architecture of extended linear convolution for digitalmage scaling," in *Proc. IEEE Int. Conf. Field-Program. Technol.*, 2008, pp. 381–384.12] P. Y. Chen, C. Y. Lien, and C. P. Lu, "VLSI implementation of an edge oriented image scaling processor," *IEEE Trans. Very Large Scale Integr.(VLSI) Syst.*, vol. 17, no. 9, pp. 1275–1284, Sep. 2009.

[6] S. Ridella, S. Rovetta, and R. Zunino, "IAVQ-interval-arithmetic vector quantization for image compression," *IEEE Trans. Circuits Syst. II, AnalogDigit. Signal Process.*, vol. 47, no. 12, pp. 1378–1390, Dec. 2000.

[7] J. W. Han, J. H. Kim, S. H. Cheon, J.O.Kim, and S. J. Ko, "Anovel image interpolation method using the bilateral filter," *IEEE Trans. Consum.Electron.*, vol. 56, no. 1, pp. 175–181, Feb. 2010.

[8] H. Kim, Y. Cha, and S. Kim, "Curvature interpolation method for image zooming," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 1895–1903,Jul. 2011.

[9] C. C. Lin, M. H. Sheu, H. K. Chiang, C. Liaw, and Z. C. Wu, "The efficient VLSI design of BI-CUBIC convolution interpolation for digital image processing," in *Proc. IEEE Int Conf. Circuits Syst.*, May 2008,pp. 480–483.

[10] P. Y. Chen, C. C. Huang, Y. H. Shiau, and Y. T. Chen, "A VLSI implementation of barrel distortion correction for wide-angle camera images,"*IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 1, pp. 51–55, Jan. 2009.

[11] [9] E. Meijering, K. J. Zuiderveld, and M. A. Viergever, "Image reconstruction by convolution with symmetrical piecewise nth-order polynomial kernels," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 192–201, Feb.1999.

[12] Electronic Publication: Digital Object Identifiers (DOIs):

Book Chapters:

[13] *Rafael* C. *Gonzalez, "Intensity Transformation and Spatial Filtering," in Digital Image Processing,IIIrd ed. Upper saddle river,New Jersey, Pearson education,Inc.,2008,ch.3,sec. 3.4, pp.145-189.*

[14] *Rafael* C. *Gonzalez, "Color Image Processing," in Digital Image Processing,IIIrd ed. Upper saddle river,New Jersey, Pearson education,Inc.,2008,ch.6,sec. 6.6, pp.439-442.*

.

Jennifer Eunice.Rwas born in Tuticorin, Tamil Nadu (TN), India, in 1991. She received Bachelor of Engineering degree in Electronics and communication (B.E ECE) from Arulmigu Meenakshi Amman College of Engineering, Vadamavandhal, Kancheepuram district, TN, India, in 2012 and the Master of Engineering degree in VLSI Design from Anna University (AU), Chennai, TN, India.She is currently working in the field of teaching as Assistant professor, Dr. Sivanthi Aditanar Colege of EngineeringTiruchendur, TN, India.Her research interests include Image processing,Cryptography and Network security, low power VLSI and Fuzzy logic.