Asian Journal of Applied Science and Technology (AJAST)
(Peer Reviewed Quarterly International Journal) Volume 3, Issue 1, Pages 215-221, Jan-March 2019

High Resolution Image Scaling Using Fuzzy Based FPGA Implementation

R.Karthick, A.Manoj Prabaharan, P.Selvaprasanth, N.Sathiyanathan & A.Nagaraj

Assistant Professor, Sethu Institute of Technology, Kariapatti, Virudhunagar, Tamilnadu, India.

Article Received: 30 August 2018 Article Accepted: 29 December 2018 Article Published: 15 March 2019

ABSTRACT

The Intrepolation which estimates unknown target pixel value from surrounding neighborhood known pixels, the quality of Intrepolation algorithm depends on number of pixels used in estimation. Intrepolation is one of the widely used in many applications Image processing such as scaling, resampling and retargeting. An image consists of edge and smooth pixels. The hybrid model estimation pixel value from two kind of algorithm, they fuzzy gradient Intrepolation for edge pixels and area Intrepolation for smooth pixels. Hence To exploiting content of the image, we propose model hybrid model for Intrepolation techniques to reduce cost and increase speed and as well as reduce noise appearance.

Index Terms: FPGA, Intrepolation, Low power, Image scaling.

I. INTRODUCTION

In [1] digital image scaling, image Intrepolation algorithms are used to convert an image from one resolution to another resolution without losing the visual content in the image. In the colour, image Intrepolation is the process of estimating the missing colour samples to reconstruct a full colour image. Image scaling is widely used in many fields, ranging from consumer electronics, such as digital camera, mobile phone, tablet, display devices and medical imaging like computer assisted surgery (CAS) and digital radiographs.

In many applications, from consumer electronics to medical imaging, it is desirable to improve the restructured image quality and processing performance of hardware implementation. In this technological endeavor several Intrepolation techniques have been developed ranging from very simple to highly complex techniques. Image Intrepolation becomes the preprocessing step for other image processing tasks like image registrations, image rotation. Image registration needs Intrepolation to accurately register the image at subpixel level.

Image up scaling methods are implemented for a variety of computer equipment's like printers, digital television, media players, image processing systems, graphics renderers, and so on. On the other hand, high resolution image may need to be scaled down to a small size in order to fit the lower resolution of small liquid crystal display panels. That is, the image scaling is a challenging and very significant issue in digital image processing.

II. RELATED WORK

Image Intrepolation can be defined as fitting of a continuous function through discrete points in digital image. Since the Intrepolation function is continuous we can find the new pixel value at our desired location. In other words Intrepolation reconstructs the pixel values that are lost in sampling by smoothing with an Intrepolation function.

CITE THIS ARTICLE: R.Karthick, A.Manoj Prabaharan, P.Selvaprasanth, N.Sathiyanathan & A.Nagaraj, "High Resolution Image Scaling Using Fuzzy Based FPGA Implementation", *Asian Journal of Applied science and Technology*, Volume 3, Issue 1, Pages 215-221.

A. Win scale Image Intrepolation

Image scaling using winscale [2] is proposed by Kim et al. The scaling algorithm is to improve the quality and to reduce the computational complexity by reducing number of operations per pixel. This algorithm can perform both scale up and scale down transform using an area pixel model rather than a point pixel model. To provide low complexity, it uses a maximum of four pixels of an original image to calculate one pixel of a scaled image. Also, it can provide better quality by the characteristics such as fine-edge and changeable smoothness.

The hardware of winscale is implemented using an FPGA for displaying scenes in a liquid crystal display panel. Winscale has good scale property with low complexity. Further, this work tells that the winscale algorithm has low RMSE and it has better image quality than the bilinear algorithm. This implementation utilizes 29000 gates at operating frequency of 65MHz.

B. VLSI Design of Winscale for Digital Image Scaling

Lin et al. proposed a VLSI design of winscale algorithm [3] for digital image scaling. This winscale technique uses an area pixels model to evaluate a scaled pixel. The original image and the scaled image are treated as rectangular and the intensity pixel is evenly scattered in rectangular area. The scaled image could be obtained by source image with scaling up/down in various ratios. The scale ratio that comprises horizontal scale ratio (HSR) and vertical scale ratio (VSR) is greater than 1.0 for scale up and less than 1.0 for scale down.

The VLSI architecture of this scaling algorithm is implemented by UMC 0.18 μm CMOS standard cell library. The total gate count is 17414 with the operating frequency of 130.24MHz. This architecture occupies 450 × 450 μm2 core area of chip and total power consumption is 19.41mW. Further, this can be used for processing image Intrepolation for high definition television (HDTV) in real time.

C. Real-Time FPGA Linear Convolution Intrepolation

A linear Intrepolation is [4] presented by Lin et al., which is low-cost hardware architecture with digital image scaling for real-time requirements. The scheme has the advantage of low operation complexity which reduces the coefficient generating effort and hardware cost with the Intrepolation quality, compatible to that of bi-cubic convolution Intrepolation.

The operation of linear convolution Intrepolation requires 16 weighting coefficients, generated from 16 neighboring pixels of source image. Therefore, the number of adders and subtractions used to generate weighting coefficients in this method are much less than the bi-cubic algorithm. The hardware architecture of this algorithm is implemented on Virtex-II FPGA. But the winscale and bi-cubic achieve lower PSNR as 28.1 and 33.05, respectively for the 3/2 upscaling image after 2/3 downscaling.

III. PROPOSED SCALING

A digital image pixel's [5] intensities composed of smooth and object's edge pixels. The background pixels have more similarity with neighboring pixels or almost equal intensities. And also foreground object inner pixels have

similar properties. But boundary separating objects from background has more dissimilarity pixels. Our proposed system use two different of Intrepolation techniques to carried out Intrepolation on image based on pixel similarity of neighboring pixels in the image.

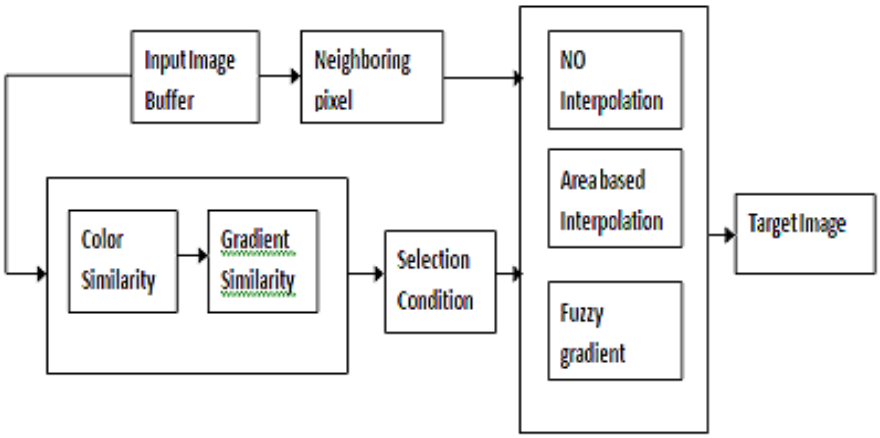


Fig 3.1 Block diagram of Proposed for Image Scaling

In state-of-the-art low-cost VLSI implementations for 2-D signal Intrepolation, performances such as circuit area in terms of number of configurable logic blocks (CLBs in field-programmable gate array (FPGA) implementation) or gate count and processing frequency in terms of pixel rate are of primary concern.

A. Neighboring Pixel Similarity and Dissimilarity

An [6] unknown pixel is estimated from known 16 neighboring pixels. The neighboring read from buffer and the similarity and dissimilarity measurement is carried out among the 16 neighboring pixels. If the pixel similarity is greater than a threshold value, the estimating known pixel is in the coarse region otherwise the pixel in fine region or neighboring pixel's average value is equal to unknown pixel value.

Two type of Intrepolation techniques are carried out Intrepolation based on pixel intensity similarities. If estimating an unknown pixel of smooth or fine is estimated by nearest neighboring Intrepolation and estimation unknown pixel value of coarse region or boundary of object is carried out by fuzzy gradient Intrepolation.

B. Iterative I-D Linear Intrepolation

In [6] the 1-D case, we use fuzzy logic concept to estimate linearly the impact of neighbor pixels on the gradient of the target pixel. Then, we apply linear approximation again to estimate the target pixel and derived the Intrepolation polynomial.

The 1-D ILI estimates target point Px according to four sample points P1, P2, P3, and P4, as shown in Fig. 2. The distances among sample points P1 to P4are fuzzified with fuzzy sets characterized by membership functions as shown in Fig. 3, where x_1 to x_4 are the x-axis coordinates of P_1 to P_4, respectively. In Figs. 2 and 3, x is the coordinate of the target point Px and $x \in [x_2, x_3]$. We traverse the sample points P1 to P3 from left to right to observe how the gradients of vectors P_1,P_2 and P_2,P_3 affect the gradient of P_x.

 Online ISSN: 2456-883X

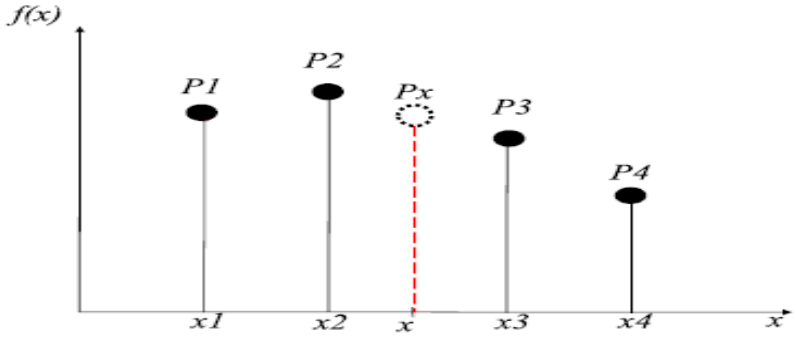


Fig 3.2 Fuzzy Gradient Neighbor Pixel Selection

$$q_{12} = \frac{f(x_2) - f(x_1)}{x_2 - x_1} \quad (3.1)$$

$$q_{12} = \frac{f(x_3) - f(x_2)}{x_3 - x_2} \quad (3.2)$$

$$q_{12} = \frac{f(x_4) - f(x_3)}{x_4 - x_3} \quad (3.3)$$

The gradient impacts qL and qR estimated on the left and right sides of P_x are as. The impacts of sample points on the target point are related here using TS inference. For physical meaning in, the known gradients q12 and q23 have impacts on gradient of P_x from the left side while q23 and q34 have impacts on gradient of P_x from the right side. Both denominators in equal 1 if the fuzzy sets adopt triangular membership functions. Therefore, computation of requires two adders and four multipliers both in uniform and nonuniform sampling cases. Moreover, we see that gradient impacts qL and qR are linear combinations of q12, q23, and q34 if the fuzzy sets L1, R1, L2, and R2 adopt triangular membership functions

$$q_L = \frac{q_{12} \times L_1 + q_{23} \times R_1}{L_1 + R_1} \quad (3.4)$$

$$q_R = \frac{q_{23} \times L_2 + q_{34} \times R_2}{L_2 + R_2}. \quad (3.5)$$

We regard qL as the slope between P_x and P2 and qR as the slope between Px and P3. According to the distance between x and x_2, we estimate the difference between P2 and Px by $(x-x_2)\times qL$ and similarly we estimate the difference between P_3 and P_x by $(x-x_3)\times qR$. Thus, fromthe left side we estimate the value f (x) of P_x by $\hat{} f_L(x)$, and from the right side we estimate the value of Px by $f_R(x)$.

We have quadratic polynomials. Finally, we estimate Px by linear approximation again as (5) where we need only one additional adder to obtain the final estimation

 Online ISSN: 2456-883X

$$\hat{f}(x) = \frac{\hat{f}_L(x) + \hat{f}_R(x)}{2}. \quad (3.6)$$

C. 2-D ILI BASED ON MULTIPLE 1-D ILI

The [6] 2-D ILI comprises multiple 1-D ILI Intrepolations. Suppose we are estimating the target at (x, y) as shown in Fig. 5 where 16 sample points (x_{i+k} , y_{i+k}), k = 1, 2, 3, 4 are given and $x_{i+2} < x < x_{i+3}$, $y_{j+2} < y < y_{j+3}$. First, four virtual points V_{ks} are estimated using 1-D ILI Intrepolations in the respective vertical directions. The dashed boxes indicate that the virtual points are obtained by performing vertical 1-D ILI at verticals x_{i+1} to x_{i+4}, respectively.

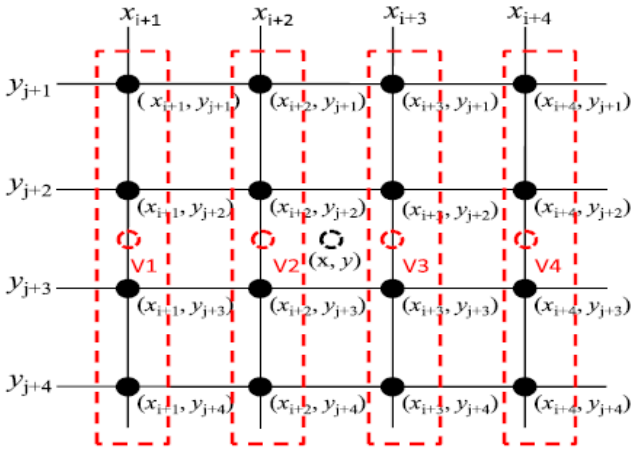


Fig 3.3 Proposed 2D Image Scaling

At each vertical x_{i+k} , the sample points (x_{i+k} , y_{j+1}), (x_{i+k} , y_{j+2}), (x_{i+k} , y_{j+3}), and (x_{i+k} , y_{j+4}) serve as inputs of 1-D ILI to produce V_k . Then, using these four virtual points V_{ks} on the same horizon, additional 1-D ILI Intrepolation give the estimation of the target point at (x, y).

D. Color pixel Similarity

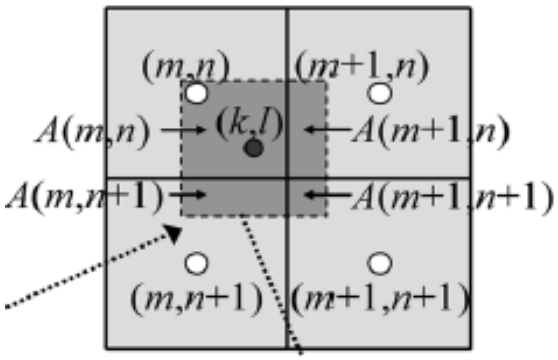
The similarity [6] measurement is carried among neighboring pixels, in order to select appropriate Intrepolation techniques to reduce computation cost. The Manhattan distance metric used here for intensity similarity measurements. Each pixel has vector of color component (red, green and blue) intensity value. The Manhattan distance function computes the distance that would be traveled to get from one data point to the other if a grid-like path is followed. The Manhattan distance between two items is the sum of the differences of their corresponding components. The formula for this distance between a pixel X and Y, X=(X1, X2, X3) and a point Y= (Y1, Y2, Y3) is:

$$d = \sum_{i=1}^{n} |x_i - y_i| \quad (3.7)$$

 Online ISSN: 2456-883X

E. Area based Intrepolation

A window is put on the current target pixel to calculate its estimated luminance value. As shown in Fig 3.4 the number of source pixels overlapped by the current target pixel window are one, two, or a maximum of four.



Let the luminance values of four source pixels overlapped by the window of current target pixel at coordinate be denoted as $F_S(m,n)$, $F_S(m+1,n)$, $F_S(m,n+1)$, and $F_S(m+1,n+1)$ respectively.

The estimated value of current target pixel, denoted as can be calculated by weighted averaging the luminance values of four source pixels with area coverage ratio as

$$F(K,L)=f(m,n)\times w(m,n)+f(m+1,n)\times w(m+1,n)+f(m,n+1)\times w(m,n+1)+f(m+1,n+1)\times w(m+1,n+1)$$

where W(m,n), W(m+1,n), W(m,n+1), and W(m+1,n+1), represent the weight factors of neighboring source pixels for the current target pixel at (k,l). Assume that the regions of four source pixels overlapped by current target pixel window are denoted as A(m,n), A(m+1,n), A(m,n+1), and A(m+1,n+1) respectively, and the area of the target pixel window is denoted as A_{sum}. The weight factors of four source pixels can be given as

$$W(m,n)=\frac{A(m,n)}{A_{sum}} \qquad (3.8)$$

IV. EXPERIMENTAL & RESULTS

Table 4.1: Performance of proposed method for image scaling

Image	**Scaling**	**PSNR(db)**	**MSE**
Camerman (128x128)	2	26.32	151.68
Lena (128x128)	2	26.90	132.73
Mandril (128x128)	2	24.87	211.98

 Online ISSN: 2456-883X

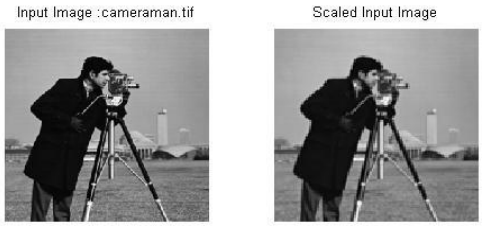


Fig 4.1 Proposed Method Image Scaling Output

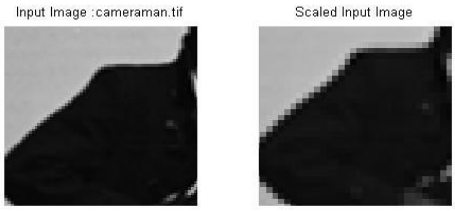


Fig 4.2 Proposed Method Image Scaling Output and Noise on Edges

V. CONCLUSION

The both Intrepolation techniques processed based on similarity measurements of neighboring pixels. The FPGA implementation hardware cost almost as same as fuzzy gradient technique. However speed and power consumption execution improved. We have proposed a hybrid Intrepolation, namely fuzzy gradient Intrepolation for coarse pixels and area based Intrepolation for smooth pixels. The power and speed improvising based on number of number smooth pixels contained in the interpolating image.

REFERENCE

[1] Karthick, R and Sundararajan, M: "A Reconfigurable Method for TimeCorrelatedMimo Channels with a Decision Feedback Receiver," International Journal of Applied Engineering Research 12 (2017) 5234.

[2] Karthick, R and Sundararajan, M: "PSO based out-of-order (ooo) execution scheme for HT-MPSOC"Journal of Advanced Research in Dynamical and Control Systems 9 (2017) 1969.

[3]Karthick, R and Sundararajan, M: "Design and Implementation of Low Power Testing Using Advanced Razor Based Processor," International Journal of Applied Engineering Research 12 (2017) 6384.

[4]Karthick, R and Sundararajan, M: "A novel 3-D-IC test architecture-a review," International Journal of Engineering and Technology (UAE)7 (2018) 582.

[5] R.Karthick, P Selvaprasanth, A Manoj Prabaharan, "Integrated System For Regional Navigator And Seasons Management," Journal of Global Research in Computer Science 9(4),2018(11-15).

[6] R. Karthick,N Sathiyanathan, and M. Eden, "Medical Image Compression Using View Compensated Wavelet Transform" Journal of Global Research in Computer Science 9(9),2018(1-4).