Multidisciplinary Journal for Applied Research in Engineering and Technology (MJARET)

Vol. 1 Issue 5, December – 2021, Pages: 27-30

An extremely efficient picture scaling processor is implemented in FPGA.

Mr. Sakthivel
Assistant Professor, Department of IT
Anand Institute of Higher Technology

Abstract: Many industries, including consumer electronics and medical imaging, use image scaling to enhance the quality of images. Many real-time applications rely on scalability, which necessitates a low-cost VLSI solution. This study offers a low-complexity, low-memory need, and high-quality VLSI implementation of an image scaling processor. Using a clamp filter and bilinear interpolation, the proposed picture scaling method uses a sharpening spatial filter. Prior to bilinear interpolation, spatial and clamp prefilters are utilised to reduce the blurriness and aliasing effects. To preserve memory buffers and computing resources, we use a hybrid approach that combines a spatial and a clamp filter. In order to save money on hardware, the reconfigurable unit utilises several adders. Filter latency will be increased as a consequence. To decrease latency, we describe a high-speed parallel prefix-based link adder. The results of the experiments will show that the proposed architecture is better in terms of space and power overhead than earlier techniques to scaling pictures of lesser complexity.

Keywords— IQ (image quality), RCUs (reconfigurable calculation units), multiplier–adder (MA).

1. INTRODUCTION

Digital image processing relies heavily on picture interpolation. An innovative interpolation method based on adaptive quadratic signal class detection from local image behaviour is presented in this study. Interpolating directly by any factor and including data collecting system characteristics into the algorithm are two advantages of the new interpolation technique.

There are no jagged edges in the new interpolation because it is superior to prior interpolation methods in terms of the underlying image model, but it is also superior in terms of aesthetics because of its success in eradicating jagged edges.

Polynomial-based and non-polynomial-based picture scaling algorithms are the two main approaches. The nearest neighbour algorithm is the simplest polynomial-based method. If you're looking for a simple solution, scaling up a photo isn't the best option. The most often used scaling method is bilinear interpolation [1], which use the linear interpolation model in both horizontal and vertical directions to obtain the target pixel. The bicubic interpolation technique [15], which uses an extended cubic model to get at the target pixel over a 2-D regular grid, is another popular polynomial-based approach. There have been a number of high-quality non-polynomial-based algorithms in recent years [2–4]. Interpolation based on curvature [2, 3], bilateral filter [3, and autoregressive model [4] are some of the successful tactics that significantly improve the quality of the images in these unique ways.

2. ALGORITHMS FOR INTERPOLATION

There are two main types of interpolation algorithms: adaptive and non-adaptive. Non-adaptive approaches treat all pixels the same, while adaptive methods modify the interpolation (sharp edges vs. smooth texture). Non-adaptive algorithms include the closest neighbour, bilinear, bicubic, spline, sinc, Lanczos, and other types of algorithms. When interpolating, they use nearby pixels ranging from 0 to 256 (or more) depending on how complicated the image is. They can improve precision by incorporating additional pixels that are close by, but doing so comes at a significant performance penalty. These techniques allow you to distort and resize images.

A. bilinear Interpolation

Two of the most used interpolation techniques are adaptive and non-adaptive. There is a distinction between adaptive and non-adaptive interpolation (sharp edges vs. smooth texture). Other non-adaptive algorithms include the closest neighbour and the bilinear and bicubic algorithms. In order to interpolate between adjacent pixels, they need a range of nearby pixels ranging from 0 to 256 (or more). However, the processing time required to achieve more precision comes at the expense of adding more neighbouring pixels. These techniques allow for the distorting and resizing of images.

3. SCALING BASED ON CONVOLUTION

For all kinds of images, there isn't a universal scaling method. As a result, the user's desired print or display

resolution and scale factor determine the final image quality (IQ). Any convolution-based scaling approach has three steps: an anti-aliasing filter, an image reconstruction using continuous convolution, and a resampling to the final grid. It's possible to do two-dimensional scaling and downscaling at a low hardware cost using this formal approach, which we show here. On the basis of a 2-D-separable interpolation kernel, we describe a discrete convolution engine. We also demonstrate how to leverage the kernel and scale factor to reduce memory footprint. Last but not least, we've developed a flexible filter that may be used in any image, regardless of the frequency content. In order to improve the intelligence of an image, the interpolation kernel and parameters may be customised by a user according to the kind of image being processed. For image interpolation methods that reconstruct a high-resolution image from a low-resolution counterpart, edge structures must be preserved. By combining directional filtering with data fusion, we've developed a new technique to non-linear edge-guided interpolation. A pixel's value may be estimated using two observation sets, one in each of two orthogonal orientations. By increasing the intensity of the centre pixel in comparison to its surrounding pixels, sharpening spatial filters are high-pass filters that remove blurring artefacts. The clamp filter [6, a low-pass filter based on a convolution kernel array, is a 2-D Gaussian spatial domain filter. A single positive number is typically in the middle, surrounded by ones [15]. aliasing and discontinuous boundary edges may be smoothed down by using the clamp filter.

A. Memory and resource requirements:

Before applying a clamp spatial filter to the input image, a sharpening spatial filter is used to filter it. Because of the ease of sharpening spatial and clamp filters with T- and inversed-T-models, there are still two line buffers required for each T- and inversed-T-model filter. T-model or inversed T-model spatial and clamp filters should be merged into a single filter to conserve computing resources and memory:

$$p`(m,n) = \left[p`(m,n)\begin{bmatrix} -1 & S-1 \\ -1 & \end{bmatrix}/(S-3)\right] * \begin{bmatrix} 1 & C \\ 1 & \end{bmatrix}/(C+3)$$

$$= p`(m,n)\begin{bmatrix} -1 & S-C & SC-2 & S-C & -1 \\ -2 & & S-C & -2 & \\ & & -1 & & \end{bmatrix}/[(S-3)*(C+3)]$$

$$\approx P`(m,n)\begin{bmatrix} -1 & S-C & SC-2 & S-C & -1 \\ -2 & & S-C & -2 & \end{bmatrix}/[(S-3)*(C+3)]$$

Fig. 1 depicts the scaling method presented in this paper. Clamp filter and bilinear interpolation are used in conjunction with a spatial sharpening filter to get the final result. When using bilinear interpolation, the blurring and aliasing effects are exacerbated because of the spatial and clamp filters [6].

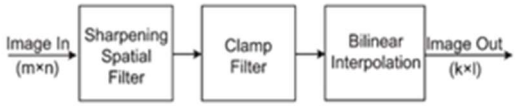

Fig 1 Block diagram of the proposed scaling algorithm

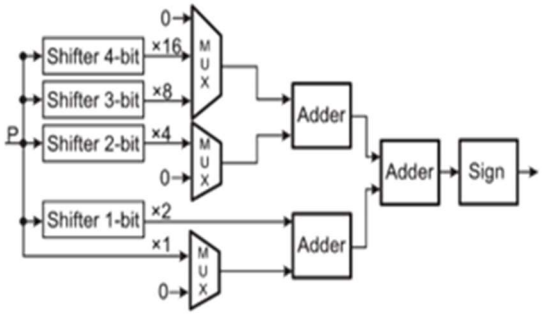

Fig 2 Receiver architecture

B. Combination filter

Convolution function of sharpening spatial and clamp filters is discussed in Section II and the equation is shown in Section III (1). Figure 5 illustrates the six-stage pipelined design of the combined filter and bilinear interpolator, which leverages pipeline technology to decrease the delay and boost performance. A T-model combination and an inverse T-model filter are shown in phases 1 and 2 of Fig. 3. The T-model filter, also known as the inversed T-model filter, has three reconfigurable computation units (RCUs), one multiplier–adder (MA), three adders (+), three subtracters (), and three shifters (S). Using the convolution equation in place of the T-model combined filter's hardware architecture (1). Using the register bank described above, the values of the 10 source pixels may be retrieved.

Multidisciplinary Journal for Applied Research in Engineering and Technology (MJARET)

Vol. 1 Issue 5, December – 2021, Pages: 27-30

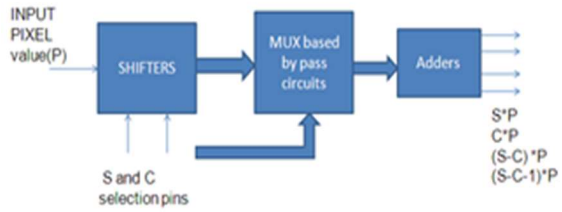

Fig 3 Reconfigurable MUX-based bypasses

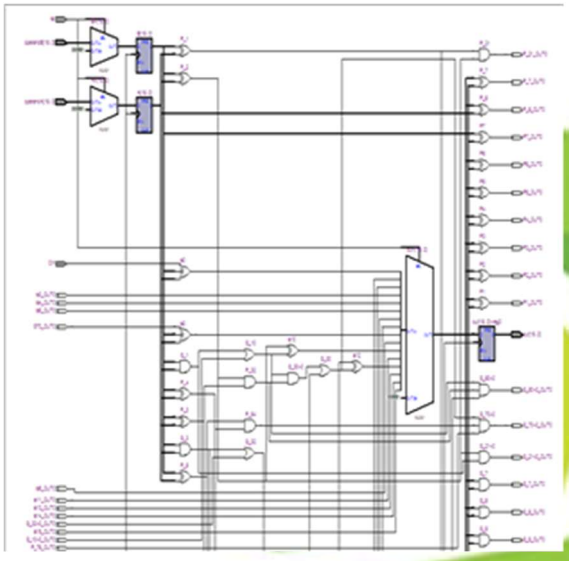

Fig 5 RTL view

4. PERFORMANCE OUTCOMES

Due to the use of shifters and adders, the parallel prefix-based technique is suitable for VLSI implementation. For a quick multiplier-less approximation of the RCU, Liang et al. [7] offer a lifting technique based on binary shift and addition operations. As a result, the hardware complexity of the VLSI implementation is relatively low. The modified unfolded RCU is used to produce a VLSI implementation that needs just binary shift and addition operations. A trade-off between the complexity of the hardware and the approximation error may also be employed to determine the computing accuracy. In addition, our technique provides a stable post-scaling factor, making it ideal for scaled RCU implementation.

TABLE I

The effectiveness of a parallel prefix adder over a CLA

Adder type	AREA(LE's)	F max(MHz)
CLA	64	330.58
PPA	59	360.62

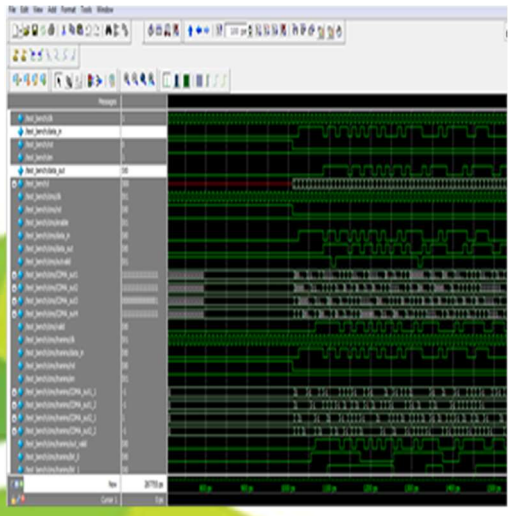

Fig 4. Output simulation

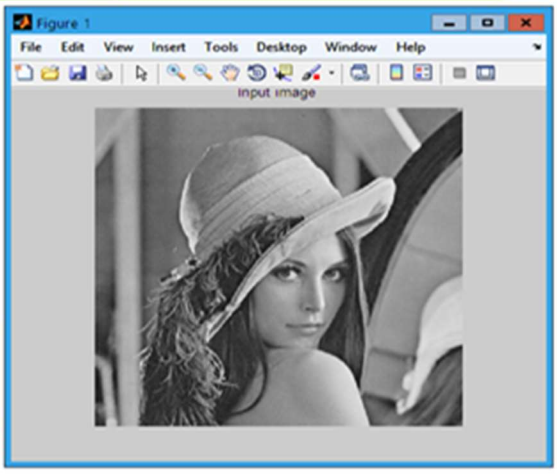

Fig 5. image input

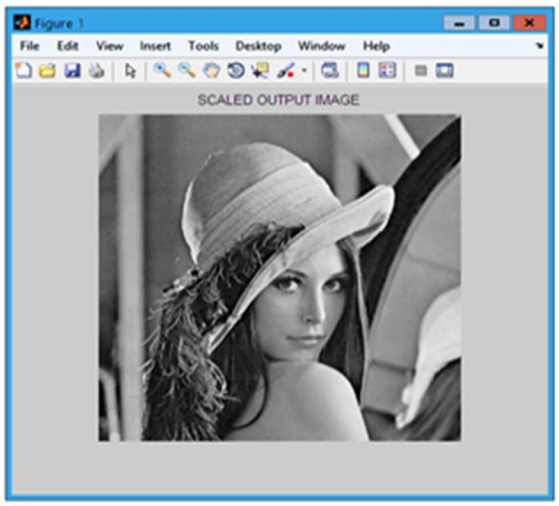

Fig.6 image that has been scaled

5. CONCLUSION

Because it relies on shift registers and adders, the RCU processor has low hardware needs and costs. The number of gates required in hardware implementation, such as on an FPGA, is low because of the reduced complexity of the circuitry. The QUARTUS II EDA tool together with MATLAB were used to create the final verified scheme.

REFERENCES

[1] "Subpixel edge localization and the interpolation of still images," K. Jensen and D. Anastassiou, IEEE Trans. Image Process., vol. 4, no. 3, pp. 285–295, Mar. 1995.

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

[3] F. Cardell's-Tormo and J. Arnabat-Benedicto, "Flexible hardware-friendly digital architecture for 2-D separable convolution-based scaling," IEEE Transactions on Computers. IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 7, July 2006, pp. 522–526,

[4] Saponara S., Fanucci L., Marsi S., Ramponi G., Kammler D., and Witte E. M., "Application-specific instruction-set processor for Retinex link image and video processing," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 54, no. 7, pp. 596–600, July 2007.

[5] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," IEEE Trans. Image Process., vol. 17, no. 6, June 2008, pp. 887–896.

[6] 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 Transactions on Image Processing, vol. Exp. Briefs, IEEE Trans. Circuits Syst. II, vol. 56, no. 1, pp. 51–55, Jan. 2009.

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

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