Design Implementation of Generic Architecture for Image Processing Applications and its Verification with UVM Framework

Sougata Bhattacharjee, Samsung Semiconductor India Research (SSIR), Senior Staff Engineer, Bangalore, India,

sougata.b@samsung.com

***Abstract*— The idea of the paper is to design a generic architecture that acts as a hardware accelerator and comprises the capabilities of image scaling algorithms like Bilinear, Nearest neighbor, and Box Filter. The generic architecture has the capabilities of a DMA controller and contains a scaler unit that helps in resizing images, bilinear scaling, noise & interference reduction, and also provides the best-downscaled images. The scaler unit acts as a hardware accelerator which helps in memory-to-memory transfer. The paper aims towards building the generic architecture by including the Box Filter along with Bilinear and Nearest neighbor so that it supports more data formats and features keeping in mind the area and speed requirement which are the main aspects of any SOC-based design. The architecture supports NPU (Neural Processing Unit) data format and Image format along with aligning corners true and False. Finally, the design is verified with the help of the UVM infrastructure testbench and creating proper stimulus to check whether the design is verified properly and working as per the specification.**

Keywords—Bilinear, Nearest Neighbour(NN), Box Filter, Hardware accelerator, NPU

I. Introduction

Scaling algorithms are promising and emerging research areas in the current era of image processing, neural networks, and machine learning. Image scaling is a technique of resizing the digital image and while doing so it ensures that the picture quality is not compromised. The factors considered for image scaling are resizing, resolution, and picture quality. There are two forms of Image scaling which are upscaling and downscaling.

A generic architecture for the scaling algorithm has been designed which acts like a hardware accelerator and has the capabilities to support features of multiple Scaling algorithms like Bilinear, Nearest Neighbour, and Box Filter. The idea behind taking the generic architecture is to accommodate the features of all the scaling algorithms to build a common hardware. While considering various scaling algorithms for building the generic architecture, features like ease of implementation, area requirement, level of complexity, speed, upscaling/downscaling, and noise reduction are given high importance and due to the reason mentioned above a generic architecture has been designed which uses the features of box filter along with the added benefits of nearest neighbor and bilinear.

The design is verified with the UVM (Universal Verification Methodology) Testbench infrastructure by implementing all the UVCs. One of the complex parts of this architecture is the implementation of the Scoreboard which takes care of multiple streams of data arising out of separate algorithms and synchronizes them with the help of TLM (Transaction Level Modelling) Analysis FIFO and `uvm_analysis_imp_decl macro for the corresponding write function. Moreover, the comparison is performed between the Golden reference model and the actual stream of data with the help of the Out of Order Comparator (OOO).

II. DIFFERENT SCALING ALGORITHMS

This section illustrates all the scaling algorithms considered for building the Generic Architecture and their related usefulness for having the scaler unit act as a hardware accelerator.

A. Bilinear Algorithm:

The bilinear algorithm [5][3] is an extension of the linear interpolation technique which is done w.r.t both x and y coordinates twice. It takes 4 pixels (2 x 2) does the weighted average of the four closest pixels and maps it to the specified output coordinates. In general, it iterates over all the points in the image and performs the following operation for each pixel:

- Compute the weighted average w.r.t the distance of the top right and top left pixel
- Compute the weighted average w.r.t the distance of the bottom left and bottom right pixel
- Compute the weighted average w.r.t the y distance of two generated colors in pixels.

It is a resampling method that performs linear interpolation in one direction and then repeats it in another direction.

Half Pixel Center (HPC) is a variant of the bilinear algorithm that can support both NPU data and Image format. It is useful when handling textures of different resolutions to make sure the grids are aligned and the upscale or downscale operation properly works.

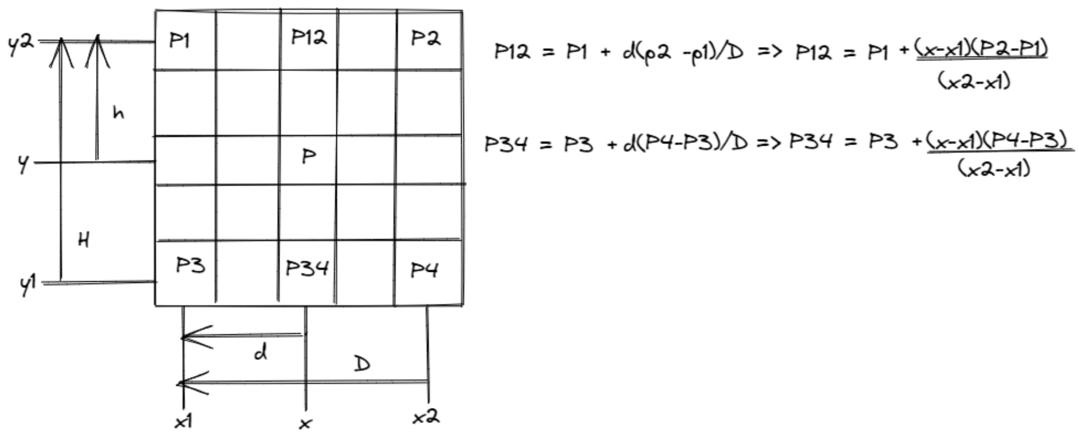

Figure 1: Bilinear Interpolation Technique

Advantages:

- Bilinear Interpolation uses only 4 nearest pixels which are located in the diagonal direction to find the appropriate color values of the pixel.
- Less computation and hence produces a smoother surface on the image resolution.
- Bilinear is best used to calculate continuous data like elevation and raw slope values.

Disadvantages:

- Lower quality of downscaled Images.
- Bilinear uses a weighted average for the four nearest cell centers but the output value could be different than the nearest input and since the values can change this algorithm is not recommended to calculate categorical data.
- Slower in comparison to the nearest neighbour resampling because the calculation took place for every output pixel.
- Lower quality of blur correction.

B. Nearest Neighbour (NN) Algorithm:

The nearest neighbor known as proximal Implementation is the simplest algorithm to implement. Like bilinear, it also iterates over all the points in the images but here, the exception is that for each pixel in the image, it assigns the color to be the nearest equivalent pixel in the image.

In the nearest neighbour the output is calculated using the first value of the vertical pixel and the first value of the horizontal pixel and the algorithm supports both NPU data format & image format. If the coordinates don't exist we will round up the coordinate to the nearest integer thus forming the nearest neighbour. We then use Euclidian distance to compute the closest nearest point.

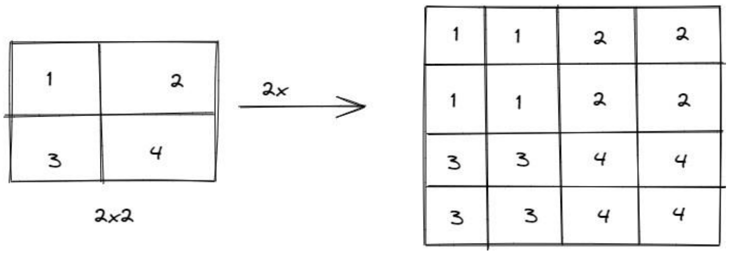

Figure 2: Nearest Neighbour (NN) Algorithm

Advantages:

- Simple to implement
- Fast as compared to Bilinear Interpolation
- Produces better-downscaled images than bilinear for the edges that are not antialiased and able to preserve hard edges.

Disadvantages:

- NN does not produce new pixel vectors into the image statistical distribution whereas bilinear is used to produce new vectors for computation.
- Prone to the problem of Aliasing (Aliasing is a problem where smooth curves are rasterized using pixels)

C. Box Filter Algorithm:

Box Filter or Box Blur algorithm [2] is the most widely used technique to produce better-downscaled Images as compared to the other algorithm mentioned above. As the name suggests it considers every pixel in the downscaled image as a rectangular box and due to that it measures the average of the color inside the box.

Additionally, it also addresses the problem of Aliasing and resolves the issue with the help of its inherent feature of acting as a Low Pass Filter. It is a 4x2 technique; hence, the area requirement is more than Bilinear and NN but at the cost of this, we are getting better-downscaled images and reduced aliasing. One more advantage of using this technique is that the number of logic gates is less as there is no requirement to store the partial sum.

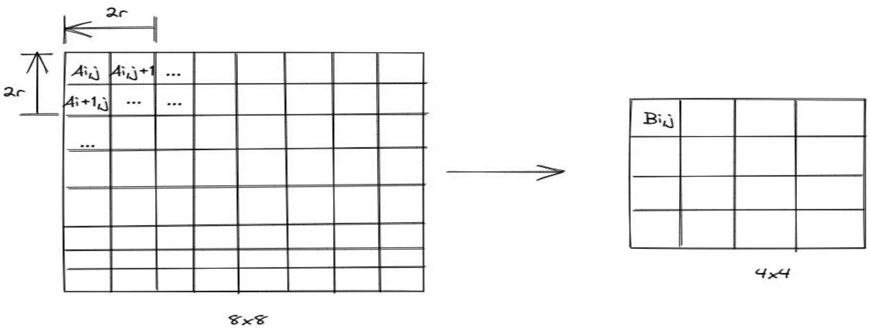

Figure 3: Calculation in Box Filter Algorithm

Box Filter acts as a linear filter where each pixel calculated is the average of the pixel values in a square centered at the pixel. But the main advantage here is that the box filter does not use any weighted average like other algorithms. Box filters are typical low-pass filters that block the high-frequency component of the signal, pass the low-frequency component, and also help in noise reduction. But one disadvantage is that the edges also might get blurred. The maximum use of the Box filter can be achieved with the use of a filter kernel which helps in resizing, sharpening, edge detection, and motion blur of the image.

There are two blurring techniques available for Box Filtering which are mentioned below:

[1] Average Blurring:

In this kind of technique, each pixel value in an image is replaced by the weighted average of the neighborhood intensity pixel values. The most commonly used filter is Box blur which has equal weights and the minimum value of the filter kernel is 3x3.

[2] Median Blurring:

This is a non-linear filtering technique and this takes the median of all the pixels under kernel area and replaces the central element with the median value. This blurring technique is very efficient in noise reduction and the amount of blurring is also less as compared to the other linear filters. Since this is a median the output image does not consist of any new pixel values other than the input image. In the case of even-order median filters, the output is the average of the central two samples after sorting.

Advantages:

- Best downscaled image without compromising resolution.
- Reduced aliasing effect and noise cancellation feature.
- Better Gamma Correction, blur correction, and better resolution of greyscale images.

Disadvantages:

- Area requirement is more

III. DESIGN OF GENERIC ARCHITECTURE

A. Design consideration:

- Buffer
- Area requirement
- Ease of implementation / Complexity
- Upscaling and Downscaling
- Speed
- Noise and Interference Reduction

B. Design of Generic Architecture:

The generic architecture is a kind of Hardware accelerator that operates the transfer of data between I/O devices and memory like DMA (Direct Memory Access) but along with that performs the resizing operation of the data using the image scaling algorithms and performs the computation on NPU (Neural Network Processing Unit).

The scaler module acts as a Hardware accelerator that performs the resizing operation. It supports NPU Data & image format and can be programmed through the generic architecture’s Register setting. The image format can be classified into Luma and chroma and supports both upscaling and downscaling. Apart from that, it supports integer data format.

The other features that the generic architecture supports are gamma correction, blur correction, noise cancellation, and interference reduction and it does this with the help of the feature Box Filter Algorithm. One of the most important features this architecture supports with the help of Box Filter in addition to Bilinear and Nearest Neighbour is to produce the best-downscaled Image. Box Filter has an inbuilt filtering mechanism and works as the low pass filter where it filters out the unwanted noise and it can be possible by plotting in a Nyquist plot and performing the frequency convolution.

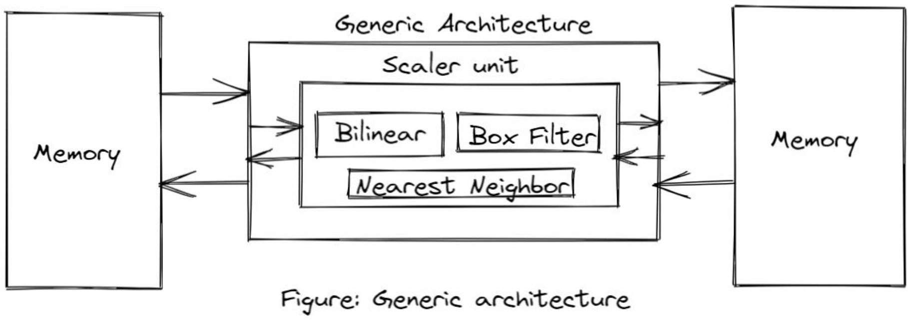

Figure 4: Design of Generic Architecture

Precisely, the generic architecture is a common hardware that can be configured as per the need to support bilinear, nearest neighbor, or box filter for image formats, upscaling or downscaling, gamma correction, noise cancellation, and interference reduction and speed.

It also supports the align corner option with both true and false as the output. When scaled up with aligning corner true, corner data of input and output have the same value irrespective of whether the data has been scaled up or down. But when scaled up with aligning corner false, corner data of input and output have different outputs irrespective of whether it has been scaled up or down resulting in a different output as compared to the previous case. This is the feature of supporting aligning corners true/false.

NPU is a specialized processor of network applications using data-driven parallel computing architecture especially good at processing massive multimedia data including images. Generally, GPU/CPUs can also perform similar functions but the usage of NPU is due to the massive rise of machine learning applications and its high computing power. In general, NPU data is represented in NCHW format, and considering the size of the block of data, it has been subdivided into smaller subgroups to fit into the processing unit. In NCHW format, N represents the batch, C represents the number of channels, H represents the height and W represents the width.

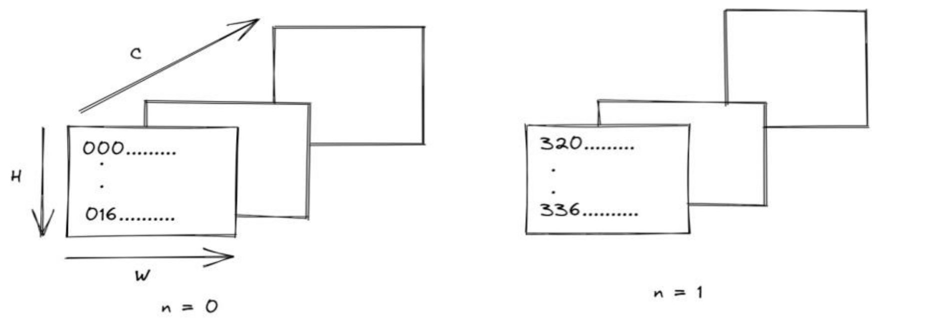

Figure 5: NHCW format for NPU

The value at the position (n, c, h, w) is generated using the following formula value (n, c, h, w) = n * CHW + c * HW + h * W + W

The use of nchw is there to represent that w is the innermost dimension or width means that two elements adjacent to the memory would share the same indices of n,c, and h and their index of w would differ by 1. On the other hand, n represents here the outermost dimension and it means that if we need to take the same pixel (c, h, w) but on the next image we need to jump over the whole image of C*H*W. In the case of generic architecture, the receiver will interleave the data for merge and the scaler unit will process the data for the different data formats.

C. Design requirement:

- Buffer for storing the input
- Reusability
- Memory Control Unit to access the buffer
- Scaler Control Unit
- MAC core for doing the computation
- Register Setting to control the flow of data

D. Design Implementation:

Box Filter also known as moving average uses a rectangular kernel coefficient and 4x2 to perform the weighted average. As the buffer requirement of the box filter is two, the buffer can be reused for the bilinear interpolation (2x2), and nearest neighbor, hence thereby increasing the reusability and optimizing the area requirement. A memory control unit is used to access the memory used in the buffer. The generic architecture uses the scaling unit to scale the source data and store it in the memory.

The DMA transfers the data from memory to memory for scaling operations. The data is read from the memory and then processed into the scaler unit where the scaling operation is performed by the different image scaling algorithms and then the output data is written into another memory. The scaler unit using the generic architecture can act as a hardware accelerator for improving the performance of various processing scenarios like image processing, neural processing, etc. The Scaling unit is capable of handling both NPU data and image formats.

E. Reusability:

Since the line buffer requirement is reused for Bilinear and NN from Box Filter, there is a significant area requirement in the design (approx. ~ 24%). Moreover, in addition to the reduction of area requirements, there are other advantages like downscaled images, and noise & interference reduction which can be accomplished by adopting Box filter along with the other algorithms. Box filters also can be reused both horizontally and vertically by optimizing the design i.e. it can be decomposed into a sequence of simpler operations across each dimension by applying the data horizontally across each row and vertically down in each column.

IV. VERIFICATION OF GENERIC ARCHITECTURE

The generic architecture has been verified through a UVM-based Testbench by modeling all the UVCs (Driver, Monitor, Scoreboard, Agent, Sequencer, etc) but the most challenging part is to model the scoreboard since it has to deal with the multiple streams of data arriving in out of order fashion from different scaling algorithms and due to that the data needs to be synchronized. Eventually, the comparison logic needs to be coded in the evaluator module of the scoreboard.

A. *Types of Scoreboard:*

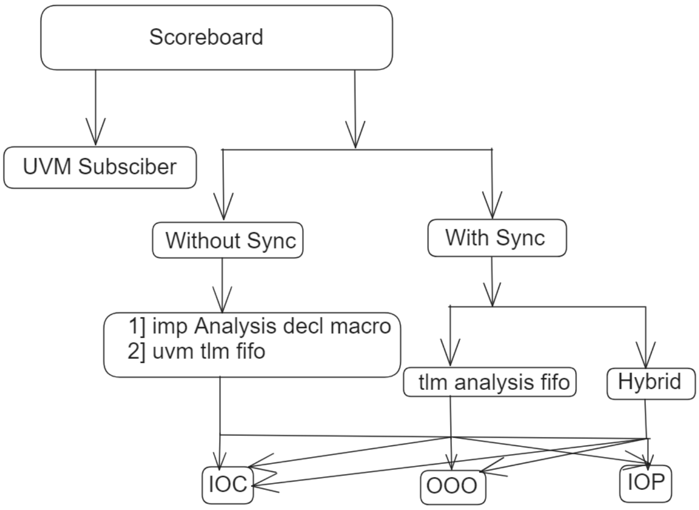

Figure 6: Classification of Scoreboard

As can be seen from the above diagram, a single stream of data can be implemented with the help of the uvm subscriber class but here in this case we need to deal with multiple streams of data from different interpolation and image processing algorithms. The data needs to be synchronized and also considering the out-of-order nature, the logic for the evaluator part is implemented with the hybrid nature of the scoreboard i.e. using `uvm_analysis_imp_decl for each corresponding write function of the streams and then using tlm analysis to synchronize them. Finally, an associative array is implemented along with the index ID [7] to handle the out-of-order comparison which has been shown in the below diagram.

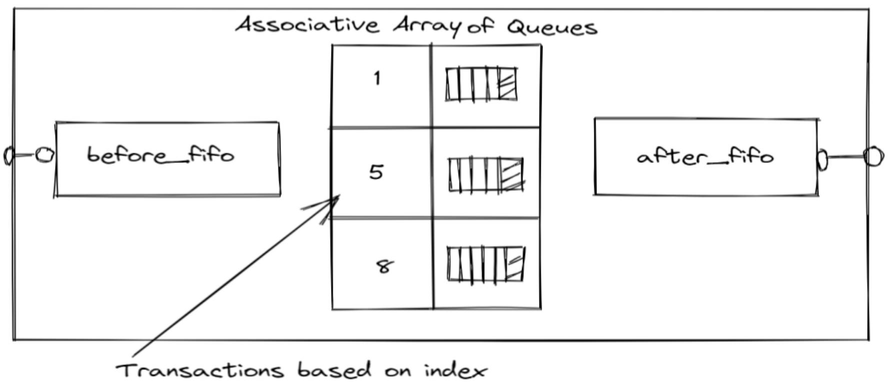

Figure 7: Out of Order comparison with associative array

The predictor or the golden reference model has been kept as a separate entity so that the scoreboard can be reused for each different function without modifying the functionality. The streams of data are taken into the scoreboard through analysis export which comes from the reference model and it is compared with the data of the input monitor to check for data mismatches.

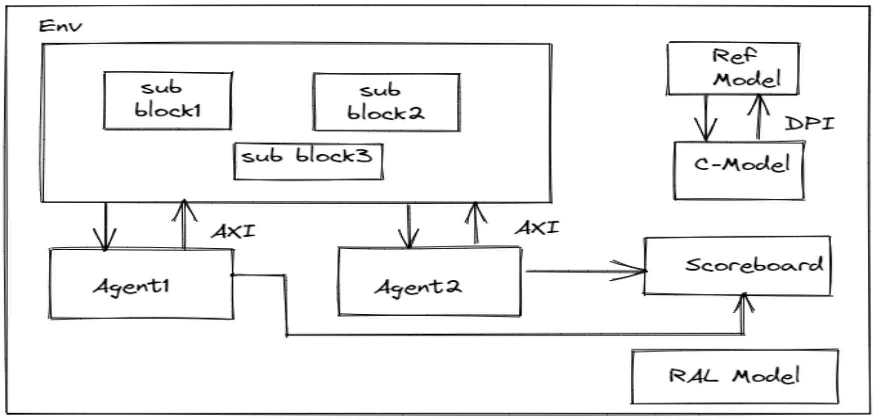

Figure 8: Testbench infrastructure

V. RESULT & CONCLUSION

	Bilinear	NN	Box Filter	Bicubic	Lanczos	Sinc Filter	CNN	Fourier Transform	Gaussian
Level of complexity	Simple	Simplest	Simple	Complex	Complex	Medium	Complex	Complex	Complex
Aliasing issue	Found	Found	Reduced	Found	Nil	Reduced	Reduced	Reduced	Best
Area requirement	Less	Less	Medium	More	More	More	More	More	More
Noise cancellation	Absent	Absent	Reduced	Better	Best	Better	Better	Better	Best
Downscaled image	Average	Average	Best	Medium	Better	Better	Better	Best	Better
Multiple computation	Average	Average	Average	Average	Better	Better	Best	Better	Best
Image quality	Medium	Medium	Better	Better	Best	Medium	Best	Better	Best

Multiple Image Scaling algorithms have been explored but finally, the Box Filter algorithm has been considered along with Bilinear and NN for Image Processing applications due to the factors mentioned above. With the help of Box Filter, more features can be supported along with best-downscaled images and reduced Aliasing effect. Finally, the Generic Architecture is verified using UVM where the majority portion of complexity is involved with Scoreboard and its out-of-order comparison.

REFERENCES

[1] Approximating Image Filters with Box Filters, 2011 18th IEEE International Conference on Image Processing

[2] Area Efficient Box Filter Acceleration by Parallelizing with Optimized Adder Tree, 2019 IEEE Computer Society Annual Symposium on VLSI

[3] Advanced Bilinear Image Interpolation based on Edge Features, 2009 First International Conference on Advances in Multimedia

[4] Generalized Parallel Bilinear Interpolation Architecture for Vision Systems, International Conference on Reconfigurable Computing and FPGA 2008

[5] A novel approach to real-time Bilinear Interpolation, Second IEEE International Workshop on Electronic Design, Test and Applications

[6] Sub Window Box Filter, 2008 IEEE Visual Communication and Image Processing

[7] UVM cookbook from Verification Academy