© June 2022 | IJIRT | Volume 9 Issue 1 | ISSN: 2349-6002

Fast and efficient Anti-Aliasing Algorithm using DLSS

Narasimha Murthy H C[1], Anand B[2], Akshay Anandan[3], Hemanth H S[4], Dr. Manjula K[5]
[6]*Associate Professor, Global Academy of Technology, Bangalore*
[1,2,3,4] *Undergraduate Student, Global Academy of Technology, Bangalore*

Abstract— **"NVIDIA Deep Learning Super Sampling (DLSS)" is a pioneering AI rendering solution that uses dedicated Tensor Core AI processors on GeForce RTXTM GPUs to boost graphics performance. To enhance frame rates and provide beautiful, clear visuals for your games, DLSS uses the power of a deep learning neural network. Video game images are becoming more computationally expensive, and graphics processing gear is failing to keep up. Computer scientists must devise novel approaches to improving the performance of graphical processing hardware. Our study employs DLSS (deep learning super sampling technique) to improve image quality and up-scale a low-resolution image to boost its resolution. When enabled, DLSS has been found to up sample internally produced 1080p or 1440p images to a higher resolution.4K or 8K photographs with no discernible loss in visual quality and, in some situations, better image sharpness.**

I.INTRODUCTION

Image super-resolution (SR) is a term used to describe the process of recovering high-resolution (HR) pictures from low-resolution (LR) images in computer vision and image processing. It has a wide range of applications in the real world, including medical imaging surveillance and security. In addition to improving picture perception quality, it assists in the improvement of other computer vision tasks.

The software method of making the edges of visual objects or typefaces smoother is known as anti-aliasing. This is done by inserting extra pixels between the boundaries of an object and its background. Anti-aliasing enhances the appearance of polygon edges on the screen by smoothing them down and making them less "jagged."

Averaging down to the true pixel level after sampling an image at a greater resolution than the display. If the sample had been done at display resolution, the averaged results would contain more information about the genuine image. This method is used to lessen the effects of aliasing in photographs.

NVIDIA DLSS (Deep Learning Super Sampling) is a ground-breaking AI rendering method that uses dedicated Tensor Core AI processors on GeForce RTXTM GPUs to boost graphics performance. Nvidia trains the DLSS neural network on various supercomputers in data centres using their in-house neural graphics framework, termed NGX. The input layer of a proprietary convolutional auto encoder network is referenced with ultra-high quality 15360 X 8640 pixel images.

We will explore numerous approaches and methodologies for fast and efficient anti-aliasing Algoritm using DLSS in this survey.

The strategies used by Xiaolin Wu [1] Anti aliased line and circle generators are based on the intuitive notion of anti aliasing, and they need even less integer arithmetic than Bresenham's approaches. Unlike its predecessors, the new anti aliasing method is developed in the spatial domain (raster plane) under a subjectively meaningful error measure to preserve the dynamics of curve and object borders. The novel anti-aliasing technique is also subjected to a frequency domain formal examination. Our anti-aliasing approach is shown to create the same anti-aliased photos as Fujimoto's programme Iwata at a quarter of the cost. The simplicity of the new anti-aliased line and circle generators corresponds to their ease of hardware implementation.

[2] LeiXiao The paper "Neural Super sampling for Real-time Rendering" describes a machine learning method for converting low-resolution input pictures to high-resolution outputs for real-time rendering. This upsampling approach employs neural networks that are trained on scene characteristics to recover crisp features while avoiding the computational burden of displaying these details in real-time applications.

[4] Yuxiang WangAuto-stereoscopic displays use an additional optical layer to throw various pictures in different directions to generate a 3D visual impression without the use of special glasses. Such a layer's topology strikes a balance between the number of various views created and the spatial resolution per view, which is a fraction of the complete 2D spatial resolution. Slanted and non-rectangular sub-sampling grids are commonly used to create a compromise, although this results in aliasing effects. When 2D material, such as graphics and text, is projected on auto-stereoscopic screens, these artefacts are highly noticeable and unpleasant. To counteract this impact, we create efficient anti-aliasing filters in our work. For a 3D display situation, two types of filters are investigated.

Thomas Auzinger[5]suggested a parallel approach for high-quality edge anti-aliasing We compare a closed-form solution to the associated prefilter convolution to typical graphics hardware approaches, which rely on large oversampling to combat aliasing difficulties. This opens the door to a large number of filter functions with adjustable kernel sizes, as well as generic shading techniques like texture mapping and complicated illumination models. Our results are correct in the mathematical sense because we used analytic solutions. They give objective ground-truth for alternative anti-aliasing approaches and allow for a thorough comparison of different models and filters. There are multiple comparisons to existing techniques and of various filter functions, as well as an efficient implementation on general purpose graphics hardware.

Jaedon Leeand colleagues [3] Vector graphics is an important technology for creating 2D graphics on mobile devices. The effective solution for vector graphics becomes increasingly significant as screen resolution grows and multi-touch interfaces become more common in mobile devices. Vector graphics should be processed with great performance and low power in future mobile contexts. We present an effective anti-aliasing technique for mobile vector graphics in this paper. Selective multi-sampling, which uses multi-sampling just for the pixels that meet the contour of the image, can considerably reduce sample counts. A basic pixelprimitive intersection test can be used to detect the multi-sampling pixel.

AUTHORS	METHODOLODY	CONTRIBUTION
Xiaolin Wu	"An Efficient Antialiasing Technique"	Antialiased line and circle generators are built from an intuitive idea of antialiasing.
Lei Xiao	"Neural Super sampling For Real-time Rendering"	The image pixels are point-sampled, yet the temporal dynamics are exact.
Jaedon Lee	"Selective multi-sample anti-aliasing for mobile vector graphics"	Vector graphics a key technology for drawing 2D graphics images on the mobile devices
Atanas Boev	"Anti-aliasing filtering of 2D images for multi-view auto-stereoscopic displays"	Filtering of images to be displayed on auto-stereoscopic displays.
Thomas Auzinger	"Non-Sampled Anti-Aliasing"	In rasterization, a parallel approach for high-quality edge anti-aliasing is used.

III.APPLICATIONS OF DLSS

Deep learning algorithms to be used on super sample photos Used to improve the visual quality of photos that are used in high-resolution gaming. To increase frame rates, produce frames at a lower resolution than displayed and then upscale the frames using deep learning such that they look as sharp as expected at the native resolution.

IV. ALGORITHM AND METHODOLOGY

For cross-platform interoperability, the NVIDIA Image Scaling SDK offers a single spatial scaling and sharpening algorithm. The scaling technique employs a 6-tap scaling filter, as well as 4 directional scaling and adaptive sharpening filters, to produce smooth pictures with crisp edges. Additionally, the SDK includes a cutting-edge adaptive directional sharpening technique for use in applications that do not need scaling.

NVScaler is the name of the directional scaling and sharpening method, whereas NVSharpen is the name of the adaptive-directional-sharpening-only algorithm. Both techniques are available as compute shaders, allowing developers to use them in their own applications. It's worth noting that if you use NVScaler, you shouldn't use NVSharpen because NVScaler already has a sharpening pass.

V. CONCLUSION

Real-time rendering of pictures has gotten more difficult as resolutions have increased and scene complexity has increased. While deep learning techniques are not perfect, theyare the best alternatives to RTR at high resolutions. Despite the excellent outcomes, there are still challenges to be faced. More research is needed to improve the efficiency and simplicity of the specified algorithms.

VI. RESULTS AND DISCUSSION

A convolutional autoencoder is a type of AI network that identifies what should be done pixel-by-pixel using the low resolution current frame and the high resolution preceding frame.

REFERENCES

[1] Xiaolin ,University of Western Ontario London, Ontario, Canada N6A 5B7“An Efficient Antialiasing Technique”

[2] Lei Xiao, Salah Nouri, Matt Chapman “Neural Supersampling For Real-Time Rendering”

[3] Jaedon Lee; Jeong-Joon Yoo; Las Vegas, NV, USA. “Selective Multi-Sample Anti-Aliasing For Mobile Vector Graphics”

[4] Yuxiang Wang UC Santa Barbara Chris Wyman. “Decoupled Coverage Anti-Aliasing”

[5] Thomas Auzinger, Przemyslaw Musialski “Non-Sampled Anti-Aliasing” Vision, Modeling and Visualization (VMV 2013)At: Lugano, Switzerland

[6] AKELEY, K. 1993. Reality engine graphics. In Proceedings of SIGGRAPH ’93, ACM, New York, NY, USA, 109–116.

[7] AMD. 2012. EQAA modes for AMD 6900 series graphics cards. Tech. rep., AMD.

[8] CARPENTER, L. 1984. The A-buffer, an antialiased hidden surface method. ACM Siggraph Computer Graphics 18, 3, 103–108.

[9] CIGOLLE, Z., DONOW, S., EVANGELAKOS, D., MARA, M., MCGUIRE, M., AND MEYER, Q. 2014. A survey of efficient representations for independent unit vectors. Journal of Computer Graphics Techniques 3, 2, 1–30.

[10] CLARBERG, P., TOTH, R., AND MUNKBERG, J. 2013. A sort-based deferred shading architecture for decoupled sampling. ACM Trans. Graph. 32, 4 (July), 141:1–141:10