Deep Learning Realtime Upsampling Techniques in Video Games

Biruk Mengistu
mengi024@morris.umn.edu
University of Minnesota, Morris

Outline

- **Introduction**
- **Background**
- **Deep Learning Super Sampling Techniques**
  - DLSS 1.0
    - Convolutional Auto-Encoder Neural Networks
  - DLSS 2.0
    - Multi-Frame Super Resolution
  - DLSS 3.0
    - Optical Flow Frame Generation
- **Results and Performance**
- **Conclusion**

Introduction

The Trade-Off between Video Resolution and Frame Rate in Video Games

- Video games are getting more demanding
- Advancements in hardware are unable to keep up
- Need a different approach

Super-Resolution Technology in Video Games

- **Super-resolution (SR):** a game is rendered at a lower resolution, then upscaled using a neural network
- Improve performance at a given resolution
- Example: Deep Learning Super Sampling (DLSS)

Background

The Graphics Rendering Pipeline

- Introduction ✓
- **Background**
- Deep Learning Super Sampling Techniques
  - DLSS 1.0
    - Convolutional Auto-Encoder Neural Networks
  - DLSS 2.0
    - Multi-Frame Super Resolution
  - DLSS 3.0
    - Optical Flow Frame Generation
- Results and Performance
- Conclusion

The Graphics Rendering Pipeline

- Objects in video games are a series of vertices
- Have X,Y,Z coordinates
- Vertices make polygons
- Many polygons combine together to make objects

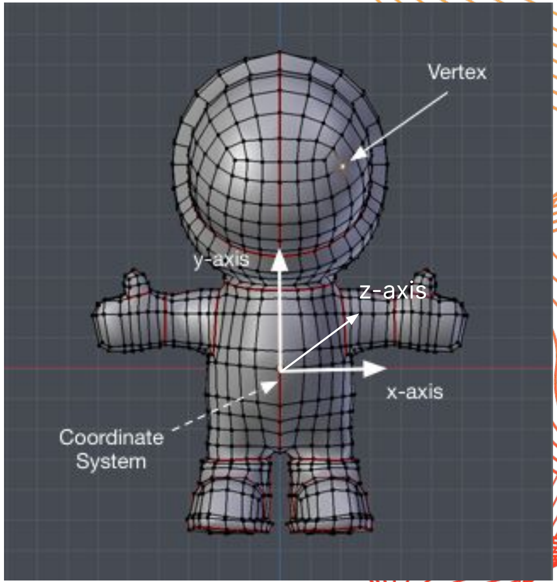
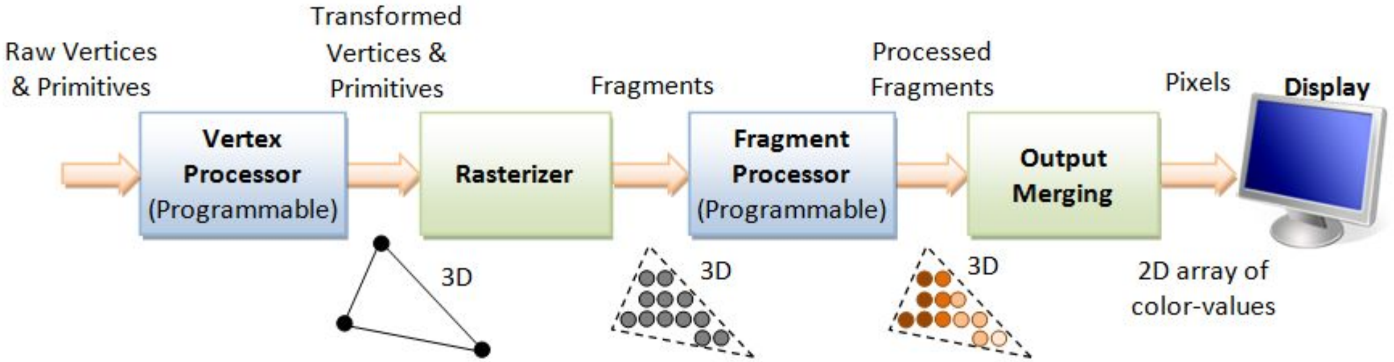


Image Credits: Serrano, Harold. “How Do Game Controllers Rotate Game Characters?” Untold Engine, 19 Jan. 2019, https://www.haroldserrano.com/blog/how-do-game-controllers-rotate-game-characters

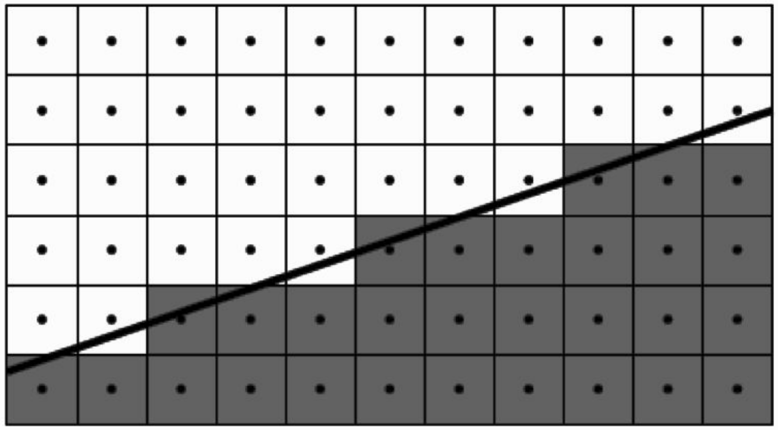
The Graphics Rendering Pipeline



Aliasing

- ➔ Results in distortion and/or pixelation.
- ➔ **Two ways to counter aliasing:**
  1. Increase Render Resolution
  2. Anti-Aliasing

Both reduce performance.



Introducing DLSS

Deep Learning Super Sampling (DLSS):a set of new techniques that uses deep learning algorithms to upscale lower resolution images in real-time.

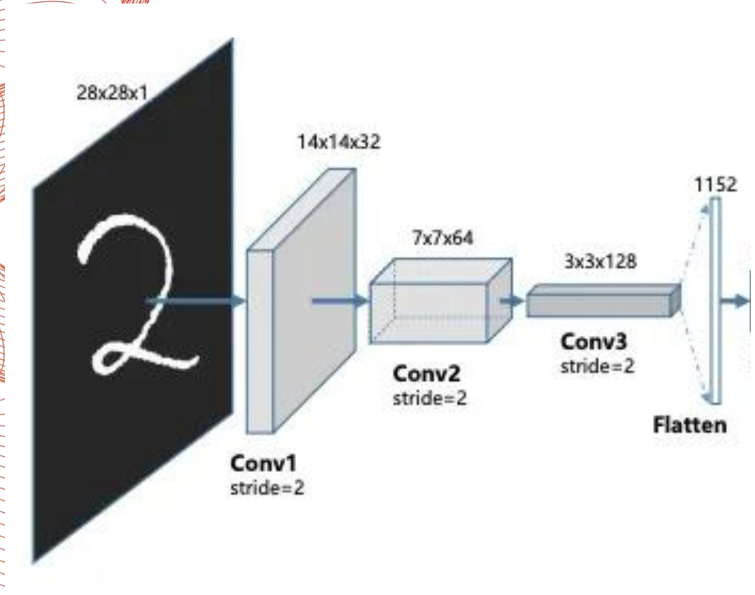
- Reduces aliasing
- Increases performance
- Higher quality graphics at lower resolutions
    - Improved frame rates

Deep Learning Super Sampling: Techniques

- Introduction ✓
- Background ✓
- **Deep Learning Super Sampling Techniques**
  - DLSS 1.0
    - Convolutional Auto-Encoder Neural Networks
  - DLSS 2.0
    - Multi-Frame Super Resolution
  - DLSS 3.0
    - Optical Flow Frame Generation
- Results and Performance
- Conclusion

DLSS 1.0

- The simplest and earliest form of DLSS (Deep Learning Super Sampling).
- Primarily an image upscaler that uses **Convolutional Auto-encoder Neural Networks (CNNs)**.

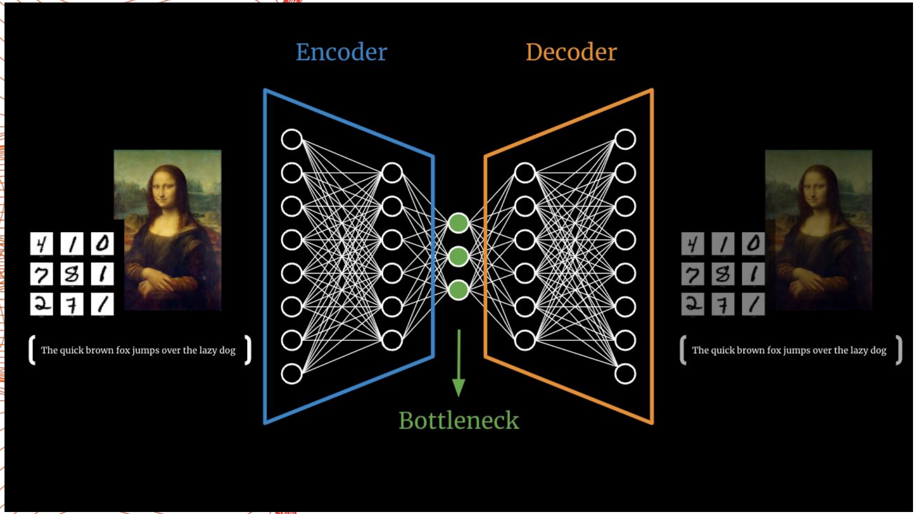


Convolutional Auto-encoder Neural Networks

First part:

- **Convolutional Neural Network**
  - Uses a mathematical operation called *convolution*
  - Especially good at processing image/video data
  - Three layers: input layer, hidden convolutional layers, output layer
  - Nodes and parameters (ex. filters)
  - Layers are two dimensional to preserve image spatial information

Image Credits:Serokell. "Introduction to Convolutional Neural Networks." Serokell, 2022, https://serokell.io/blog/introduction-to-convolutional-neural-networks.



Convolutional Auto-encoder Neural Networks

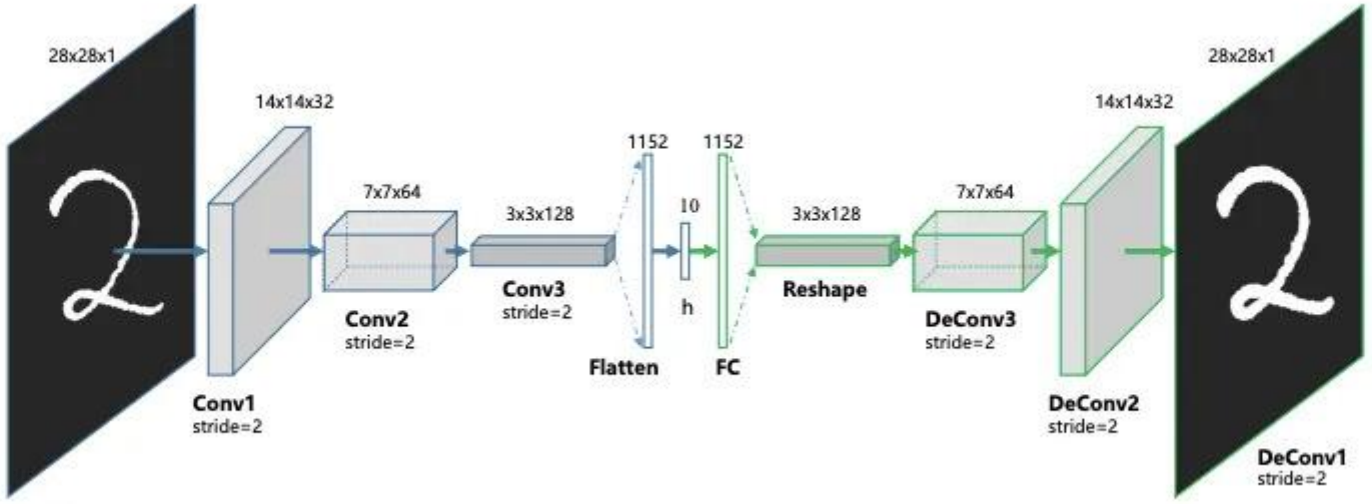
Second part:

- **Auto-Encoder**
  - Uses two processes:
    - Encoding
    - Decoding
  - Image is compressed further and further in encoder
  - The decoder takes the compressed image and recreates the original image as close as possible

Image Credits: Saber HQ. "Why are Autoencoders so Effective?" Saber HQ, 8 Sep. 2022, https://www.saberhq.com/blog/autoencoders.

How does it work?

Putting everything together:

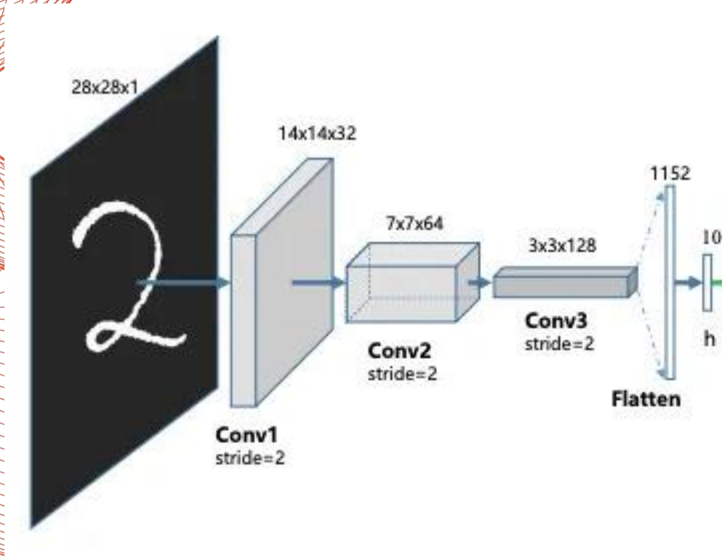


How does it work?

- **A convolution is a dot product of:**
    - Portion of an image
    - Filter: a small vector

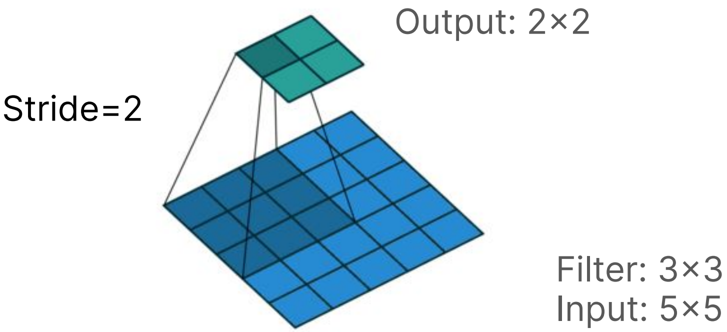
$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} e & f \\ g & h \end{bmatrix} = ae + bf + cg + dh$$

Dot product



How does it work?

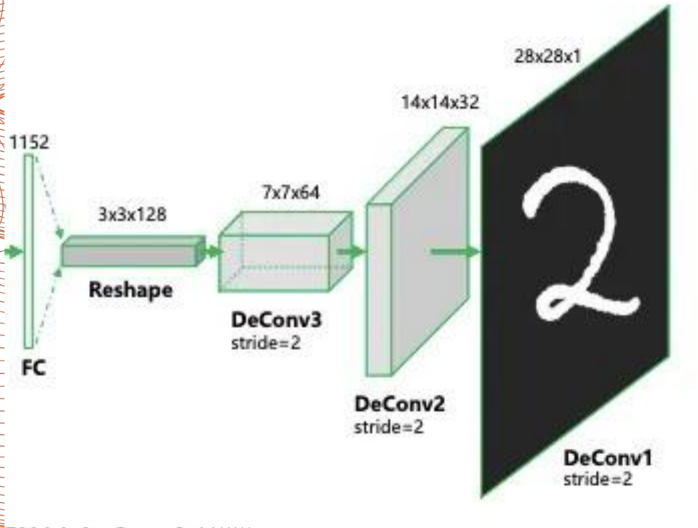
Encoding:



Convolution during encoding

Dot product with filters → Passed on to next layer

Image Credits: "Convolutional Neural Networks Demos.", 2022, hannibunny.github.io/mlbook/neuralnetworks/convolutionDemos.html.

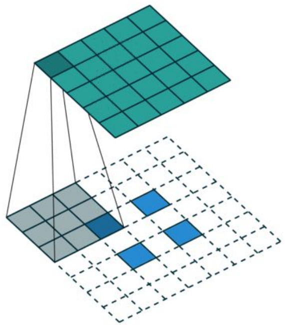


How does it work?

Decoding:

Output: 5×5 (*feature map*)

Stride=2



Filter: 3×3
Input: 2×2

Convolution during decoding

Dot product with filters → Passed on to next layer

Image Credits: "Convolutional Neural Networks Demos.", 2022, hannibunny.github.io/mlbook/neuralnetworks/convolutionDemos.html.

Two Phases of Using a CNN Auto-Encoders

Training

- ➔ Data is fed into the network
- ➔ Parameters (ex. filters) are modified
- ➔ Process is done repeatedly on thousands of data sets
- ➔ Needs large computing power

Inference

- ➔ An image is taken by the neural network
- ➔ Image is passed through the pre-trained network
- ➔ Outputs a prediction (denoised image, upscaled image etc.)

In the context of DLSS:

Training

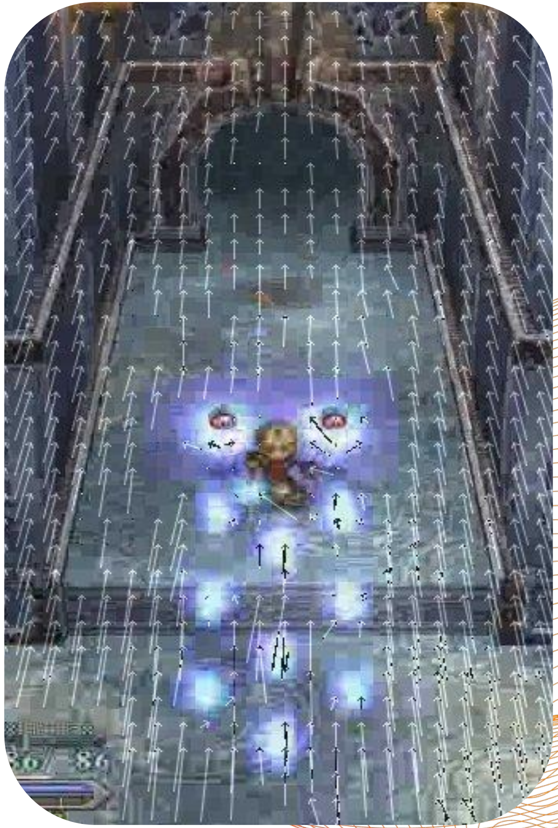
- ➔ Millions of frames are fed into the network
- ➔ The network gets better and better at recreating the original image
- ➔ **Done on a per-game basis**
- ➔ Done on an NVIDIA supercomputer

Inference

- ➔ Pre-Trained Models are shipped to consumers in the form of graphics driver updates
- ➔ Game frames are upscaled
- ➔ Needs much less computing power
- ➔ Accelerated through **Tensor Cores** (specialized processors that are especially good at vector multiplication)

Additional Data Used in Training

- **Motion Vectors**
  - Used in video games to represent the movement of in game options
  - Can help better predict appearance of objects in motion
- **Altered Frames**
  - Rotated
  - Added noise
  - Zoomed into



Motion vectors overlaid on top of a video game scene

Pros and Cons of Version 1.0:

Pros

- ➔ Can improve performance
- ➔ More demanding games can run on lower end machines

Cons

- ➔ Image is often blurry
- ➔ Frequent Artifacts (ghosting, smearing, noise)
- ➔ Needs to be trained on a per-game basis
- ➔ More work for developers as well as NVIDIA

Why the limitations?

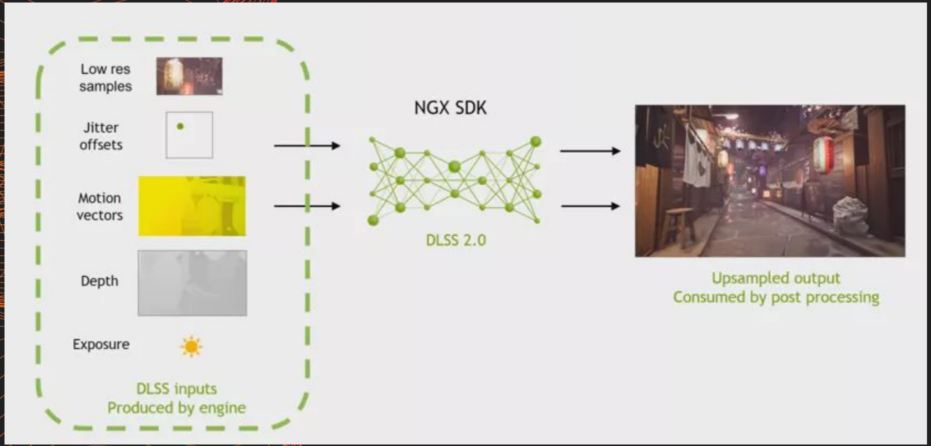
- ➔ It uses **Single-Image Super Resolution**
- ➔ Only has information from the current frame
- ➔ Has to create data that wasn’t there
- ➔ Noise and other visual artifacts

Deep Learning Super Sampling: Techniques

- Introduction ✓
- Background ✓
- Deep Learning Super Sampling Techniques
  - DLSS 1.0 ✓
    - Convolutional Auto-Encoder Neural Networks ✓
  - **DLSS 2.0**
    - Multi-Frame Super Resolution
  - DLSS 3.0
    - Optical Flow Frame Generation
- Results and Performance
- Conclusion

Introducing DLSS 2.0

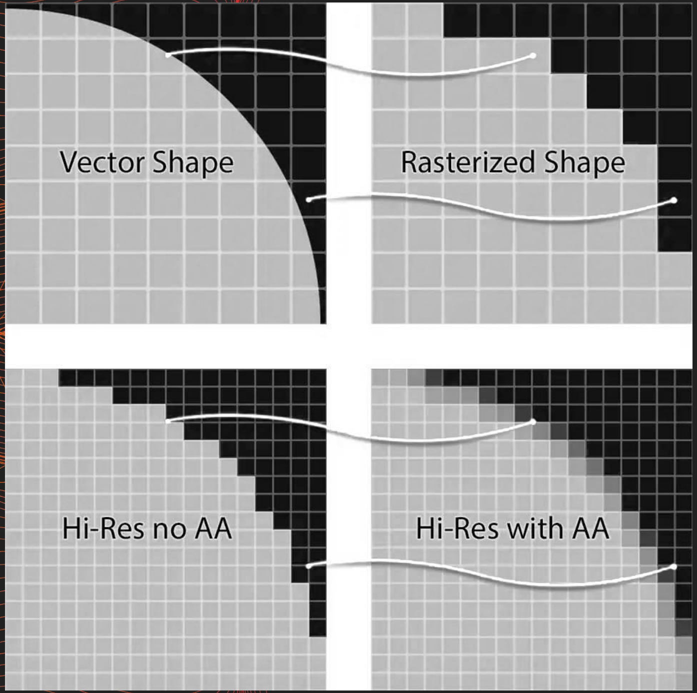
- Mitigates the shortcomings of DLSS 1.0
- Is a general model
- Utilizes **Multi-Frame Super Resolution**, which utilizes data from previous frames to 'fill in the gaps'



Additional Data

Besides incorporating current frame data and motion vectors, it utilizes:

- Temporal Data (previous frames)
- Depth Information
- Exposure
- Brightness



Deep Learning Anti-Aliasing

In addition to upscaling, DLSS 2.0 also incorporates Deep Learning Anti-Aliasing (DLAA)

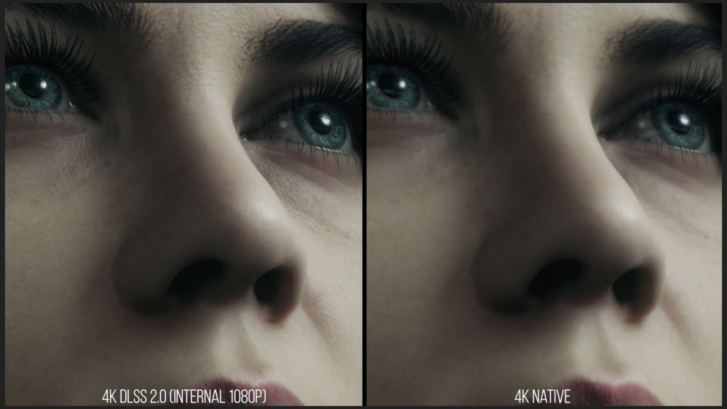
- Smoother appearance of edges in the game
- Reduces the stair-like pattern commonly seen
- Done at the same time as upscaling

Pros of DLSS 2.0

Much better appearance than DLSS 1.0 - images are very close to native rendering

Offers additional performance by offering a 4x upscaling option

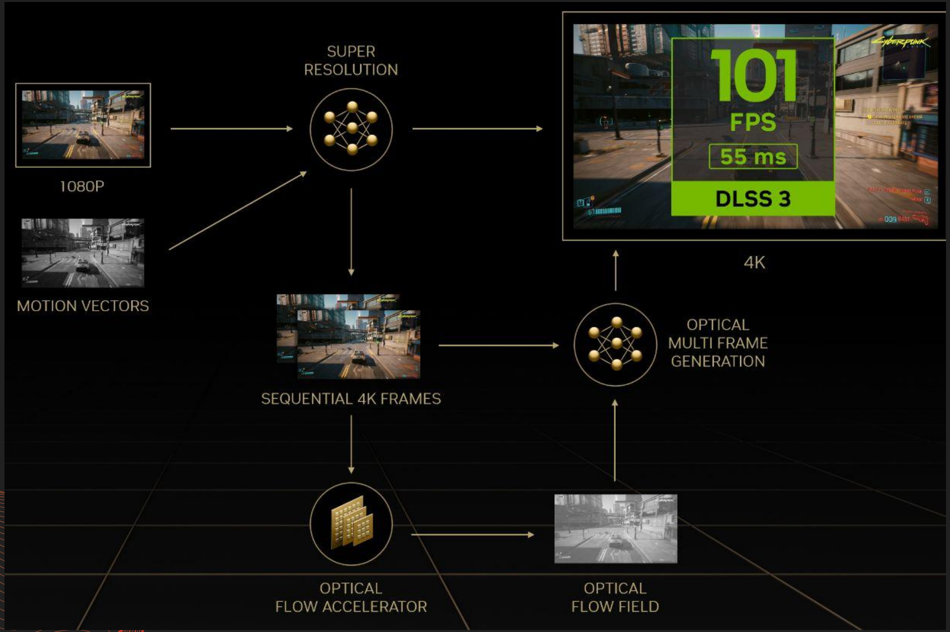
Does not need to be trained on a per-game basis, reduces development time



Deep Learning Super Sampling: Techniques

1. Introduction ✓
2. Background ✓
3. **Deep Learning Super Sampling Techniques**
    a. DLSS 1.0 ✓
        i. Convolutional Auto-Encoder Neural Networks ✓
    b. DLSS 2.0 ✓
        i. Multi-Frame Super Resolution ✓
    c. **DLSS 3.0**
        i. Optical Flow Frame Generation
4. Results and Performance
5. Conclusion

Introducing DLSS 3.0

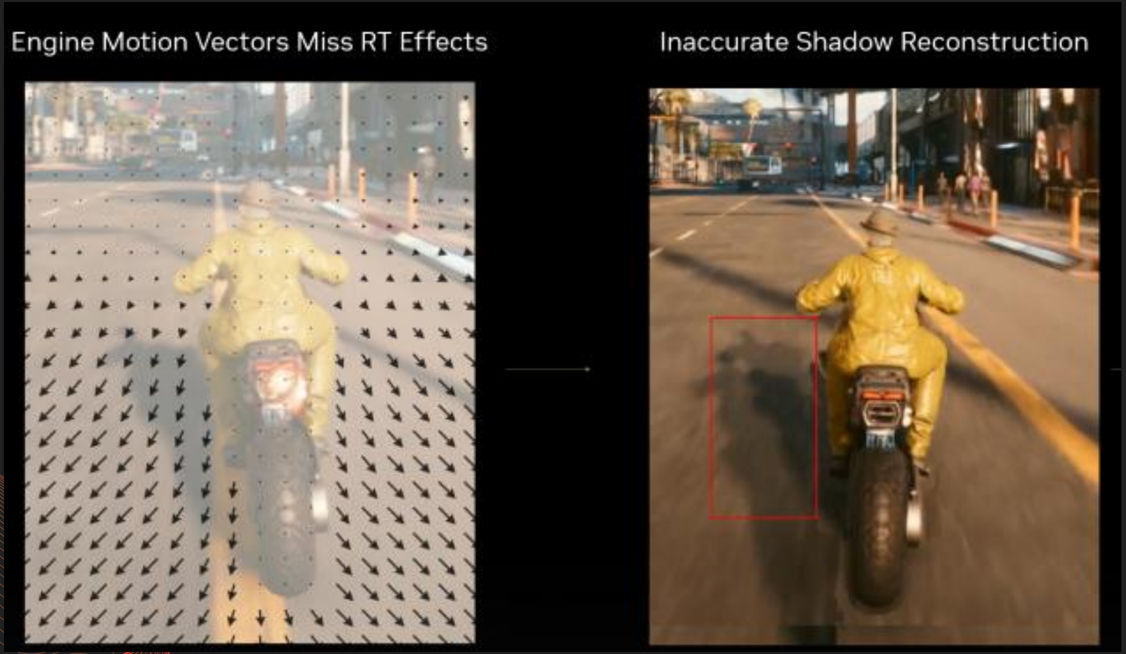


Improves performance significantly by introducing **Optical Flow Frame Generation**

Frame generation: inserts a completely generated, artificial frame in between every two frames

Optical flow: a technique that looks at two consecutive frames and determines the motion of in game objects

How is it different from Motion Vectors?



Motion vectors: from video games represent the motion of **objects** in a scene

- From game engine
- Motion is relative
- Leads to false representations of motion in objects that are not actually moving on screen

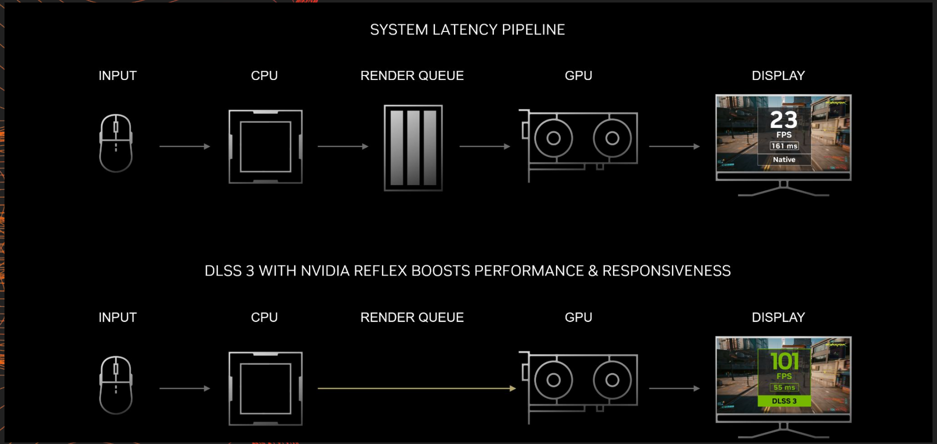
Optical flow: predicts the movement of on screen pixels by comparing consecutive frames

How and why is it used?

- ➔ DLSS 3.0 utilizes both motion vectors **and** optical flow
- ➔ This allows it to better generate intermediary frames to put between every two ‘actual’ frames
- ➔ Objects in motion and still objects are represented accurately

Latency Concerns

- Adds latency because the game has to wait until the intermediary frame is generated before proceeding
- Can be mitigated
- **Reflex:** a set of techniques that reduce system latency to offset the added latency from upscaling and frame generation



Reflex

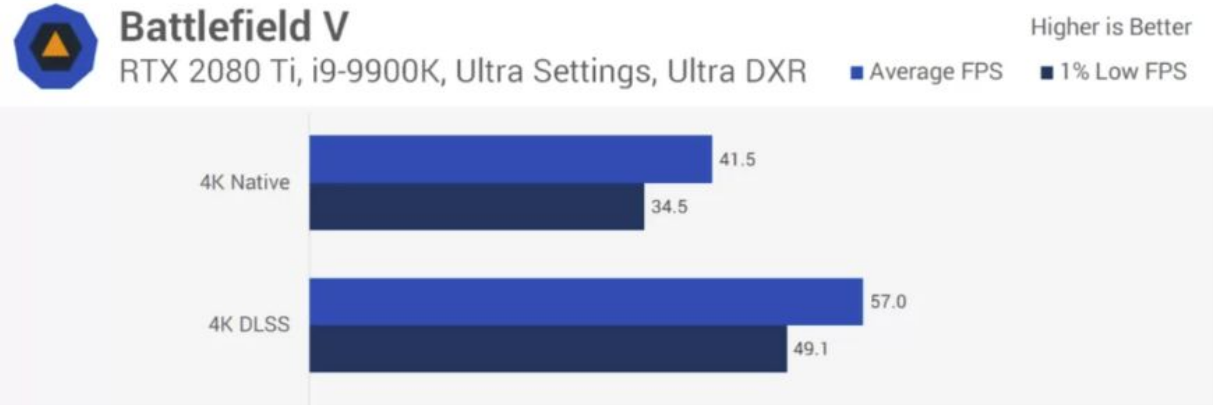
- GPU bottlenecked cases create a render queue, which adds **latency**.
- Reflex zeroes the queue
- Offsets the added latency
  - More playable
  - More responsive

Deep Learning Super Sampling: Techniques

- Introduction ✓
- Background ✓
- **Deep Learning Super Sampling Techniques**
  - DLSS 1.0 ✓
    - Convolutional Auto-Encoder Neural Networks ✓
  - DLSS 2.0 ✓
    - Multi-Frame Super Resolution ✓
  - DLSS 3.0 ✓
    - Optical Flow Frame Generation ✓
- **Results and Performance**
- Conclusion

Results

- **DLSS 1.0:**
  - Can improve frames per second by 30-40%
  - FPS: frames per second
  - Suffers from frequent visual artifacts



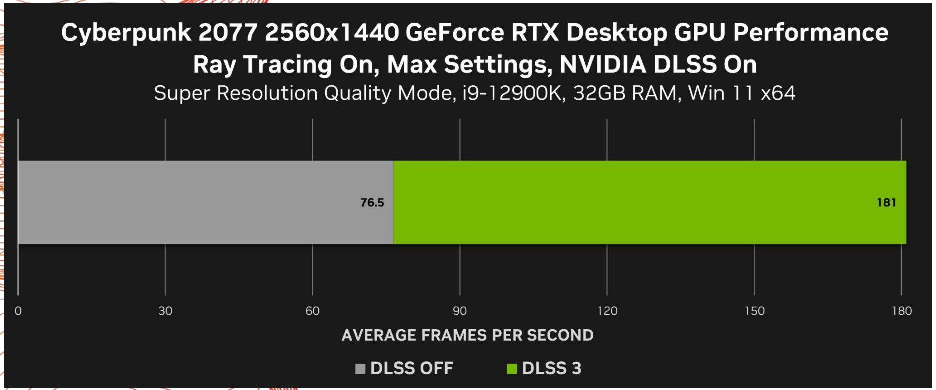
Example

- ➔ Reduced texture quality
- ➔ Blurrier appearance
- ➔ Ghosting and smearing
- ➔ Visual Artifacts

Video Credit: Hardware Unboxed, “Battlefield V DLSS Tested, The Biggest RTX Fail Of Them All”, Feb 18, 2019, https://www.youtube.com/watch?v=3DOGA2_GETQ

Results

- **DLSS 2.0:**
  - Similar FPS gains as 1.0
  - But with significantly better image quality
  - Less visual artifacts
  - Better representation of objects in motion



Results

- **DLSS 3.0:**
  - Similar visual fidelity as 2.0
  - Significant improvements in FPS with optical flow frame generation
  - 2x-3x FPS in some scenarios

Deep Learning Super Sampling: Techniques

- Introduction ✓
- Background ✓
- **Deep Learning Super Sampling Techniques**
    - DLSS 1.0 ✓
        - Convolutional Auto-Encoder Neural Networks ✓
    - DLSS 2.0 ✓
        - Multi-Frame Super Resolution ✓
    - DLSS 3.0 ✓
        - Optical Flow Frame Generation ✓
- Results and Performance ✓
- **Conclusion**

Conclusion

- Games will demand more and more processing power
- DLSS presents a new approach that uses neural networks to mitigate this
- Previous versions have had varying levels of success
  - Current version is very viable

Questions?

K. Kapse. 2021. An Overview of Current Deep Learned Rendering Technologies. In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS). IEEE. https://ieeexplore.ieee.org/document/944182

Mozilla. 2023. Explaining basic 3D theory - Game development. https://developer.mozilla.org/en-US/docs/Games/Techniques/3D_on_the_web/Basic_theory. Accessed: 2023-03-16

Wenming Yang, Xuechen Zhang, Yapeng Tian, Wei Wang, Jing-Hao Xue, and Qingmin Liao. 2019. Deep Learning for Single Image Super-Resolution: A Brief Review. IEEE Transactions on Multimedia 21, 12 (2019), 3106–3121. https://doi.org/10.1109/TMM.2019.291943

NVIDIA. 2020. High-Performance Computing and AI Innovations at GTC 2020. https://developer.nvidia.com/gtc/2020/video/s22698. Accessed on 23 March 2020

Ankit Sharma. 2018. Convolutional Autoencoders for Image Noise Reduction. Towards Data Science (2018). https://towardsdatascience.com/convolutional-autoencoders-for-image-noise-reduction-32fce9fc1763