White Paper

Visual Technologies
Multi-device Enabling

Unlocking the Power of Intel® Deep Link

Part Three: Accelerating the Inferencing Process Using Intel® GPUs

This paper is part three in a series of white papers designed to provide details regarding openly available development tools that can be used to take full advantage of Intel® Deep Link Technology.

Together, these papers will introduce and demonstrate some of the tools and processes that can be used to leverage Deep Link and allow developers to build better performing and more efficient applications. Included will be use cases that showcase Deep Link's current and future potential.

This third paper focuses on applying Deep Link concepts for improving throughput using various configuration options, presents PowerDirector™ by CyberLink® as a use case, and features an interview with the Intel® application engineer who designed and implemented the process improvements.

Authors

Roman Borisov
Senior Software Application Engineer

Brittney Clark
Software Application Engineer

YW Lei
CyberLink® VP, Product Management

Table of Contents

Introduction 1
Deep Link Technology 2
Neural Style Transfer 2
CyberLink® PowerDirector™ 3
Acceleration Suggestions 4
Developer's Journey 7
Test Configuration 8
Conclusion 8

Introduction

With the release of the Intel® 11th Generation mobile processor and the Intel® Iris® Xᵉ and Intel® Iris® Xᵉ MAX graphics architecture, Deep Link was introduced to the world and a new era of innovation was born.

Developers now have the ability to strategically apply computing power that was previously unavailable, and to assign tasks to parts of the machine which would otherwise just lie dormant. Imagine having the ability to significantly boost the performance of your application using not much more than a strategic approach and some lines of code.

That is the power of Deep Link.

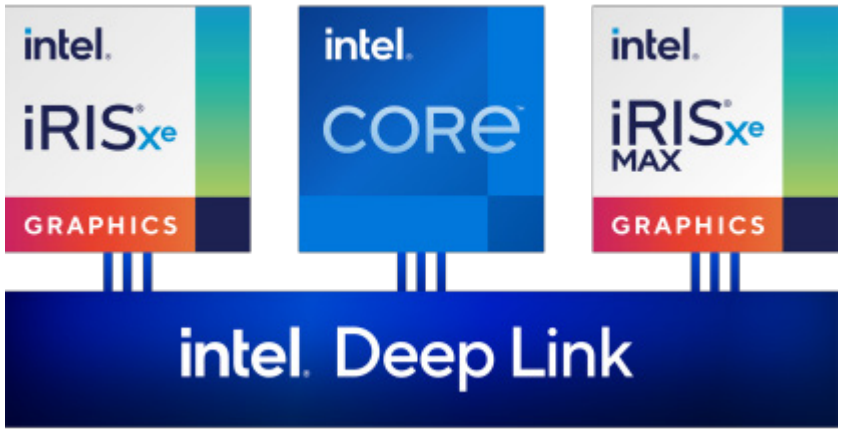

Figure 1. Intel® Deep Link combines an 11th Generation processor with an Iris® Xᵉ integrated GPU and an Iris® Xᵉ MAX discrete GPU, and manages the use of multiple GPUs simultaneously.

Deep Link Technology

At its core, Intel® Deep Link Technology is a reimagining and reshaping of the way that a machine's Central Processing Unit (CPU) and Graphics Processing Units (GPUs) interact. Already available on a number of devices, Deep Link offers the ability to combine the computing power of a discrete GPU with that of a powerful integrated GPU.

Using Deep Link, complex workloads and pipelines can be constructed which use multiple computing elements with a high degree of code re-use, thereby simplifying code development and reducing overall effort. This approach offers significant gains in both performance and efficiency, boosting the functional capabilities of a given application by offering expanded computing capabilities and processing options.

Deep Link Puts Multiple GPUs to Work

By partitioning computational elements into logical segments, Deep Link GPUs can equitably share the workload - each one working independently from and concurrently with the other. The Intel® Iris® Xᵉ (also referred to as the integrated, or iGPU) and Intel® Iris® Xᵉ MAX (also referred to as the discrete, or dGPU) can be used together to equitably split tasks, often allowing a workload to be completed in roughly half the time.

Deep Link Gets the Most out of Intel® Hardware

Deep Link enables Intel® computing components to work together at a level of speed and efficiency that was not available before by combining the computing power of multiple GPUs with similar characteristics. Because the two graphics processors use the same kernel code and have similar computing power and performance characteristics, there is little additional overhead introduced when partitioning tasks between the two.

Neural Style Transfer

As Artificial Intelligence (AI) tools continue to develop and become bigger parts of our daily lives, the creative space is becoming an increasingly popular destination for its applications. Even though in some ways they are just beginning to scratch the surface of the depth and breadth of what AI tools are capable of, artists, photographers and filmmakers are finding that AI tools are helpful in bringing their creative aspirations to life.

One of the creative outlets that has seen widespread use over the last several years is style transfer. There are several websites which offer image styling, and these readily available tools are being used by thousands of people every day. Neural Style Transfer is the AI-assisted process of combining an image (or a frame of video) with a particular style reference image in such a way that it retains the core elements of the original image, but complements, resembles or even mimics the desired style reference. Shown below is an image that was processed using one of the more common style references, Vincent van Gogh's painting *Starry Night*.

Figure 2. Graphic showing the style reference image and the original image being blended together to form a fully stylized image.

Neural networks can be designed and trained to statistically analyze the original image (for elements such as color, density, distribution, contrast) and determine the proper positioning, proportion and scale of the style reference to be applied. As shown in the images above, the shapes and tones of the original image still exist in the stylized image, but they have now been blended with elements from the reference image.

Using this method, transferring style to a single image is a process that takes very little time at all. In just a few moments, you can transform your image into one that resembles a favorite painting, uses a particular color palette, or looks like it was created using watercolor paints, colored pencils or stained glass. In some applications you can even use your own design as a style reference. The possibilities are nearly endless, and the results are nearly immediate.

Transferring style to a single image, though, doesn't satisfy the requirements for many of today's content publishers. According to statistics posted by biteable.com in 2021[1], over one billion hours of video content is viewed every day on YouTube, and by 2022 82% of all consumer internet traffic is expected to be used to stream video content. The amount of video content on the web is exploding, and many of those content producers rely on style transfer to get their videos just right.

Transferring style to video images, though, is a time-consuming process. This is due, in large part, to the number of images (frames) that are required to be transformed. For most modern displays, each second of video typically requires between 25-60 frames. This translates to 1500-3600 frames that need to be processed for each minute of video. For certain applications, such as gaming, the frame rate requirements can be much higher. The sheer volume of images required to process a lengthy video means that the speed at which you can transform video frames can have significant impact on the project and its bottom line - and there is always room for improvement.

In the following section we will explore multiple ways that we used Intel Deep Link technology to add speed and efficiency to one of the most-used titles in the video processing industry: CyberLink® PowerDirector™.

CyberLink® PowerDirector™

The video editing tools that are made available using the CyberLink® PowerDirector™ application allow users to add styles and effects to their video content, and it supports all of the latest video formats. Intel® and CyberLink® have been working together to cut processing times and increase frame throughput using the additional computational power that is made available using our Deep Link technology.

Figure 3. Image of user operating CyberLink® PowerDirector™ using two monitors.

As shown in Figure 4 below, the CyberLink® style transfer process consists of two major Deep Neural Network (DNN) Stages: Style Transfer and Refine, along with a number of pre- and post-processing steps.

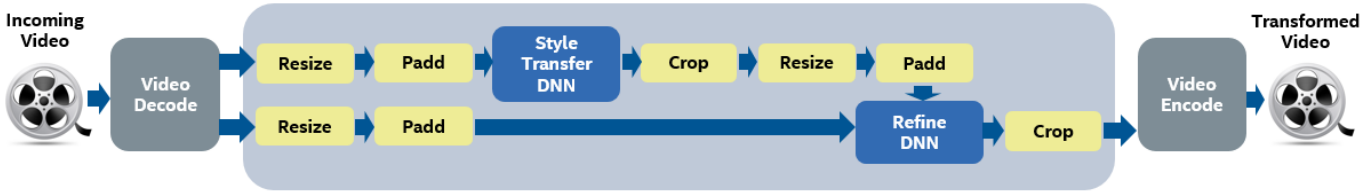

Figure 4. CyberLink® style transfer process showing pre/post processing steps and two DNN stages.

In the pre-processing stages of the incoming video, each frame is run through a video decoder and the resulting image is downscaled as needed to boost processing speed and efficiency. The Style Transfer DNN then applies the style update to each image according to the user settings. Following some additional processing steps, the Refine DNN increases the size of each frame to match the desired output.

1 - https://biteable.com/blog/video-marketing-statistics/

In the initial implementation currently used by CyberLink® in the PowerDirector™ web app, all pre/post process functions are handled by the CPU, and OpenVINO™ is used to route the DNN stages (Style Transfer and Refine) through the iGPU.

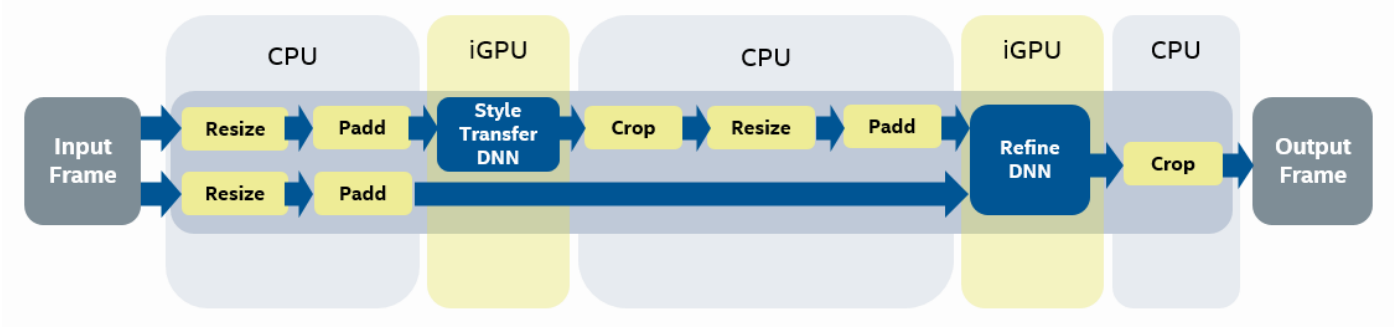

Figure 5. Current PowerDirector™ web app process which uses the iGPU for inferencing and the CPU for pre/post-processing.

This approach is similar to the one demonstrated in the OpenVINO™ samples, with OpenCV being used for pre/post-processing, and the OpenVINO™ Inference Engine being used for DNN (for more information on the OpenVINO™ Inference Engine you can download the first whitepaper in this series).

As shown in Table 1 below, using the iGPU for inferencing instead of the CPU provides significant throughput improvement over using the CPU alone for all processes.

	CPU Only	INIT
Device for Style Transfer DNN inference	CPU	iGPU
Device for Refine DNN inference	CPU	iGPU
Device for pre-post-processing	CPU	CPU
Throughput (FPS)	**3.4**	**8.7**
CPU Utilization	95%	62%
iGPU Utilization	n/a	60%

Table 1. FPS Comparison: CPU Only to Initial Implementation.

A 150% increase right off the bat is a great start, but as you can see, splitting the processes in this way leaves both the CPU and the iGPU significantly underutilized - a clear indication that there is room for improvement.

NOTE: Performance is measured and provided in Frames per Second (FPS), which is a typical benchmarking metric.

Acceleration Suggestions

Because the style transfer process utilizes multiple stages, there are a number of options to consider when trying to decide how best to accelerate and optimize the existing process, and - just like the decision to use the iGPU for running inference - each option has its own pros and cons. The three options which showed the best results are presented below.

Optimization Suggestion 1: Use iGPU for Pre/Post-processing

In the initial implementation, although the iGPU was tasked with handling all of the inferencing, the CPU was used to handle all of the pre/post-processing tasks. We can try to improve the throughput even more by running all of the pre- and post-processing operations through the iGPU, leaving the CPU to handle the pipeline initialization and interoperability operations between OpenVINO™ and OpenCV.

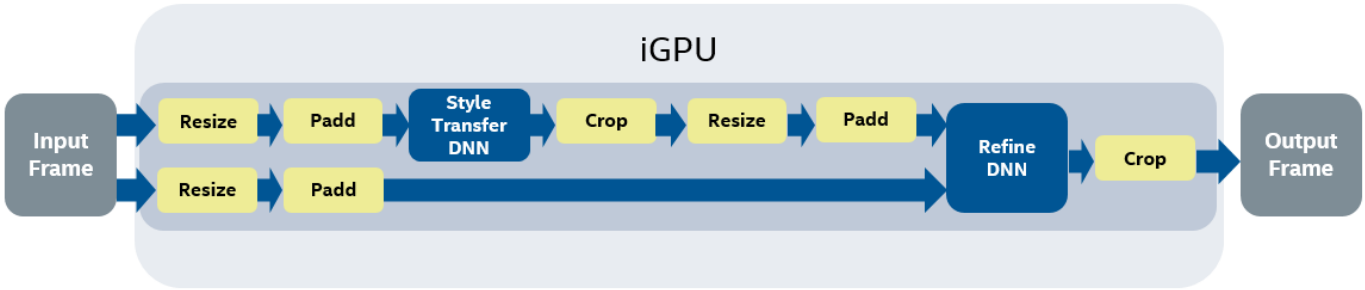

Figure 6. Optimization Suggestion 1 process which runs all functions through the iGPU.

This can be done by using the GPU Remote Blob interface of the OpenVINO™ Inference Engine and OpenCV::UMat interface instead of Mat. The result is a nearly 45% improvement in FPS over the initial implementation, as shown in Table 2 below.

	INIT	OS 1
Device for Style Transfer DNN inference	iGPU	iGPU
Device for Refine DNN inference	iGPU	iGPU
Device for pre-post-processing	CPU	iGPU
Throughput (FPS)	**8.7**	**12.6**
CPU Utilization	62%	64%
iGPU Utilization	60%	96%

Table 2. FPS Comparison: Initial Implementation to Optimization Suggestion 1.

Using this method also shows that the iGPU utilization has increased to 96%, an indication that it is being well utilized. With the use of the iGPU maximized in this instance, it is clear that in order to further increase performance it is going to be necessary to engage the second GPU.

Optimization Suggestion 2: Engage Both the iGPU and the dGPU

One of the great things about Intel's Deep Link architecture is the presence of both the iGPU and the dGPU, and the two together make a powerful combination.

In this implementation we split the pipeline apart to run the Style Transfer portion on the dGPU and the Refine portion on the iGPU. The appropriate portions of the pre/post-processing were split between the dGPU and the iGPU as well.

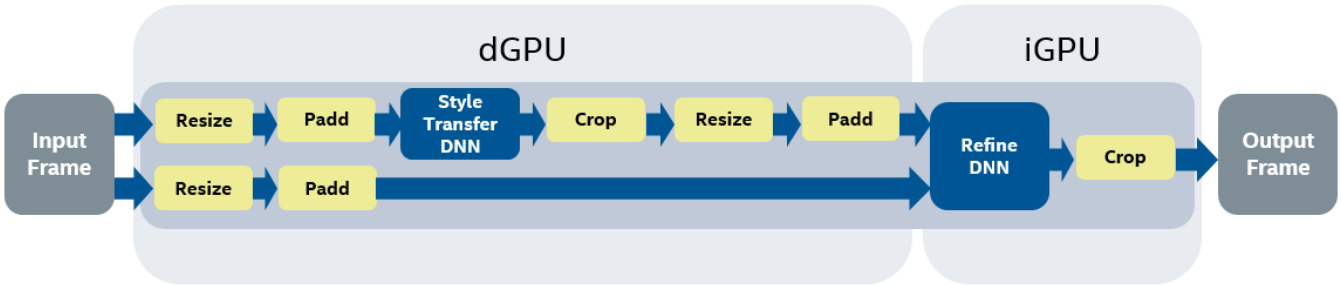

Figure 7. Optimization Suggestion 2 process which splits the inferencing modes between the dGPU and the iGPU.

NOTE: In order to utilize both GPUs at the same time and get performance gains, we made sure that the two inferences were run asynchronously.

This implementation showed an additional 10% improvement over Optimization Suggestion 1, as shown in Table 3 below.

	OS 1	OS 2
Device for Style Transfer DNN inference	iGPU	dGPU
Device for Refine DNN inference	iGPU	iGPU
Device for pre-post-processing	iGPU	iGPU/dGPU
Throughput (FPS)	**12.6**	**14.1**
CPU Utilization	64%	85%
iGPU Utilization	96%	61%
dGPU Utilization	n/a	61%

Table 3. FPS Comparison: Optimization Suggestion 1 to Optimization Suggestion 2.

Utilization totals for the iGPU and dGPU remain low, though, because there are extra processing steps required. When the process is transferred from the dGPU to the iGPU following completion of the Style Transfer stage, data from the dGPU video memory has to be transferred to system memory, and then transferred again from system memory to iGPU video memory.

Optimization Suggestion 3: Full Pipeline on Each GPU

The transfer of data between system memory and the dGPU/iGPU can be avoided by running the full image processing pipeline in parallel, one on each GPU. This increases the load time (since the models for both DNNs will have to be loaded onto each of the two GPUs), but should decrease the overall processing time.

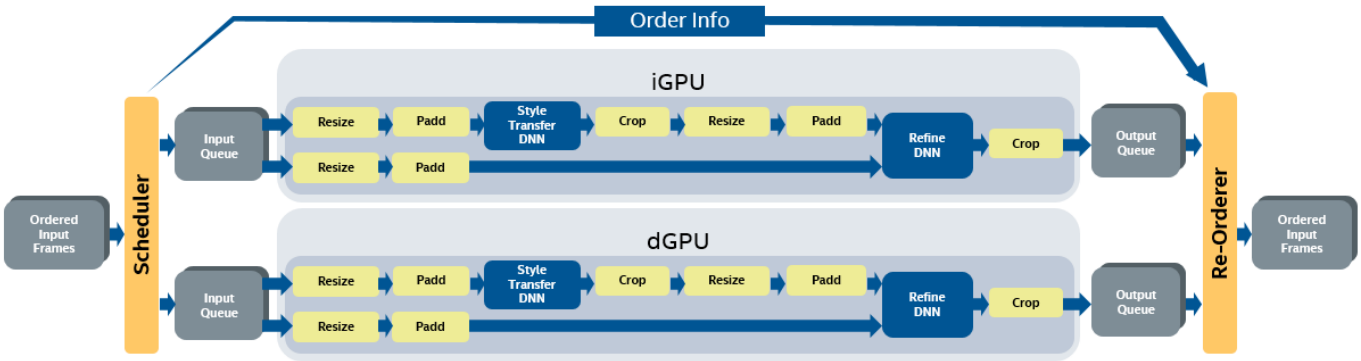

Figure 8. Optimization Suggestion 3 process which runs inferencing modes in parallel on both the iGPU and the dGPU.

As each frame is prepared for processing, the application will select the appropriate GPU based on its current input queue level and send it to/through the GPU with the shorter input queue. This allows for very efficient workload balancing, but it is essential that the original order of the frames be preserved. The Scheduler and Re-Orderer blocks are part of the application that must be implemented by the developer. As shown in Table 4 below, this approach provided an 80% performance gain over Optimization Suggestion 1.

	OS 1	OS 3
Device for Style Transfer DNN inference	iGPU	iGPU/dGPU
Device for Refine DNN inference	iGPU	iGPU/dGPU
Device for pre-post-processing	iGPU	iGPU/dGPU
Throughput (FPS)	**12.6**	**22.8**
CPU Utilization	64%	100%
iGPU Utilization	96%	97%
dGPU Utilization	n/a	77%

Table 4. FPS Comparison: Optimization Suggestion 1 to Optimization Suggestion 3.

As shown in the following figure, when comparing the initial implementation to the result of Optimization Suggestion 3 an overall FPS increase of more than 160% was achieved.

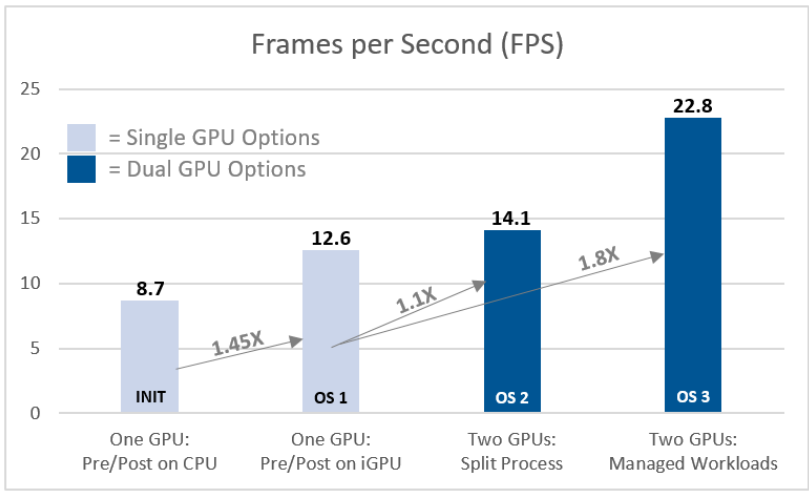

Figure 9. FPS Comparison: Initial Implementation to all three Optimization Suggestions.

Of course, the various process methods devised and used in this experiment are just some of the many options available to developers thanks to Intel's Deep Link architecture and the associated tools. There really is no limit to the number of process and application innovations that are possible with the computing power made available by Intel® Deep Link Technology.

Developer's Journey: An Interview with Roman Borisov

As Intel's collaboration with CyberLink® really began to show significant results, I decided that I wanted to tell part of the story from a developer's perspective, in order to provide a window into the thought processes that we engaged in as our acceleration efforts were taking off. I sat down with an interviewer, and we discussed some of the triumphs and missteps that were part of the process. Below are the questions and answers that we exchanged. My hope is that these insights will be helpful to other developers who might be interested in beginning or continuing their own journey into using Intel tools and hardware to build efficiency in their own applications.

Interviewer: How did CyberLink® engage and begin collaborating with Intel to accelerate PowerDirector™ processing?

RB: Intel has had an ongoing collaboration with CyberLink®, working with them on optimizing PowerDirector™ in particular. The work that we are doing is based around an Intel initiative to determine whether our Deep Link technology could be used to speed up the AI-based style transfer offered by PowerDirector™.

Interviewer: What about the PowerDirector™ style transfer process makes it a good fit for acceleration using Deep Link?

RB: PowerDirector™ Style Transfer is a very compute intensive workload, so it was a good candidate to achieve significant gains using the additional compute power made available on Intel Deep Link platforms. At its core, since style transfer is really just repeated image processing (including DNN processing), it is GPU "friendly" and will benefit from the additional processing power afforded by the use of one or more GPUs.

Interviewer: Were there obvious avenues of improvement that were explored right away?

RB: As happens a lot in development, the first implementation we attempted made sense to us intuitively but ended up being rather naïve. We initially broke the processing pipeline into two parts - running one part of the process on the integrated GPU (iGPU) and the other part on the discrete GPU (dGPU). While there certainly was some acceleration using this approach, I was not happy with the overall GPU utilization. It was obvious that both GPUs were being significantly underutilized, which meant there is some room for improvement.

Interviewer: Is there an obvious starting point for developers who want to apply these techniques to their applications?

RB: Initial analysis of the process and workload is required to see if using Deep Link to speed up an application is justified. Obviously, the first step is to make sure the workload is compute bound, because if it is not then adding more compute power will not be very helpful.

It is also important to use the correct function libraries. Utilizing the proper software libraries will help any developer achieve the best performance on an Intel hardware platform. For computer vision and AI applications, the combination of OpenVINO™ and OpenCV™ can be recommended.

Interviewer: For a developer, where should the acceleration process start?

RB: A good starting point to accelerate your application using Deep Link is to determine whether or not your app can run efficiently using a single GPU. If the app runs efficiently using a single GPU there is no need to add additional hardware. If the GPU is nearing its utilization limits or is overstressed, however, adding another processor path is likely to provide additional useful compute power and increase efficiency.

Intel® VTune™ Profiler is a good tool to help understand the bottlenecks (in both the CPU and the GPU) within an application and help a developer to maximize the utilization. VTune™ measures and reports the time it takes to complete computing tasks, as shown in the following example:

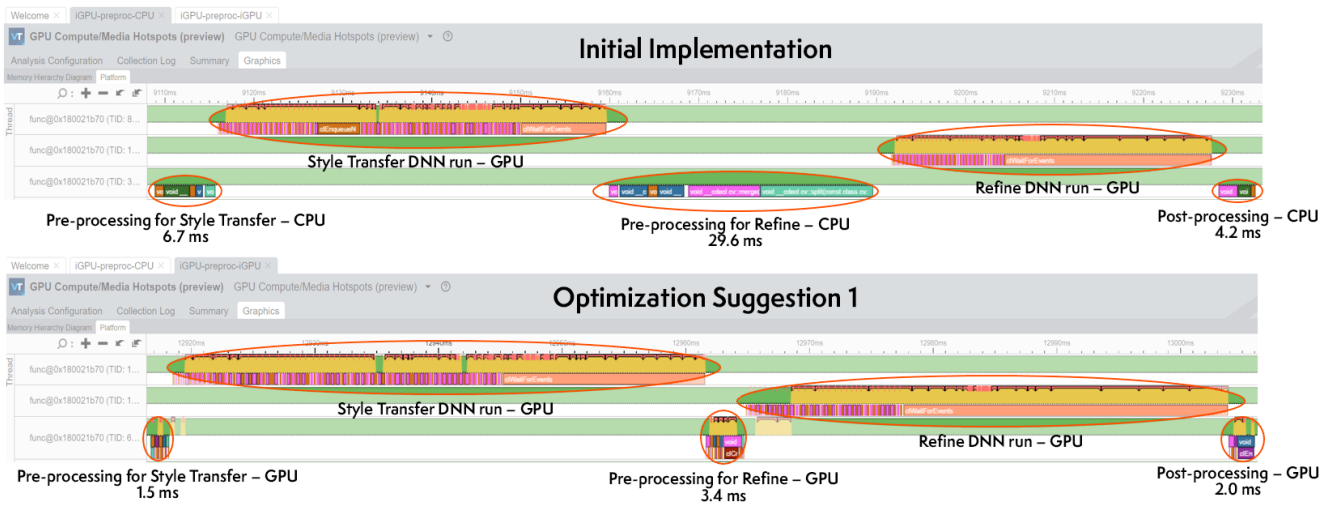

Figure 10. VTune™ comparison of stage completion times of Initial Implementation vs. Optimization Suggestion 1.

Interviewer: How will a developer know that the process is finished?

RB: Often we can find possible improvements even with already performant code. Typically, the closer we get to the theoretical limit (the sum of the available CPU computing power and the GPU computing power) the harder it becomes to find avenues to further improvement. So, in practice we can stop when our workload reaches the target performance. Another practical criteria which is specific to Deep Link could be to reach high (more than 90%) utilization for both GPUs.

Test Configuration

All throughput totals presented in this paper were collected using a 720p workload on a Deep Link-enabled ASUS® VivoBook™ with an Intel® Core i7-1165G7 processor. The optimization options were implemented using sample code that was shared with CyberLink. All tests were performed in August, 2021.

Additional system information:

OS: Windows® Pro 10.0.19043
BIOS Version and Date: American Megatrends International, LLC. TP470EZ.300, 11/4/2020
Memory: 16GB, LPDDR4
Graphics: Intel® Iris® Xᵉ (integrated); Intel® Iris® Xᵉ MAX (discrete)
Disk: NVMe, Intel® SSDPEKNW512G8 [512 GB]

Conclusion

Intel® Deep Link Technology offers new avenues and paths for data processing that are just begging to be explored, and tools like OpenVINO™ and VTune™ make implementation and evaluation much easier. Whether you are building an application from scratch or attempting to build additional functionality or efficiency into an existing application, the development process will be smoother and more effective when these tools are utilized together.

The optimizations that are presented in this paper are by no means specific to CyberLink®, Power Director™ or the Style Transfer process. These solutions (and others presented in this white paper series) may directly apply to your AI application, allowing you to get the most out of your development efforts.

It is our sincere hope that you have found the information in this white paper helpful and informative. Other papers in this series promise to bring to light other tools that can be paired with Deep Link to provide exceptional processing speed and performance for your new and existing applications. From dynamic power sharing to accelerated video rendering and much more, the opportunities for performance improvement are nearly endless.