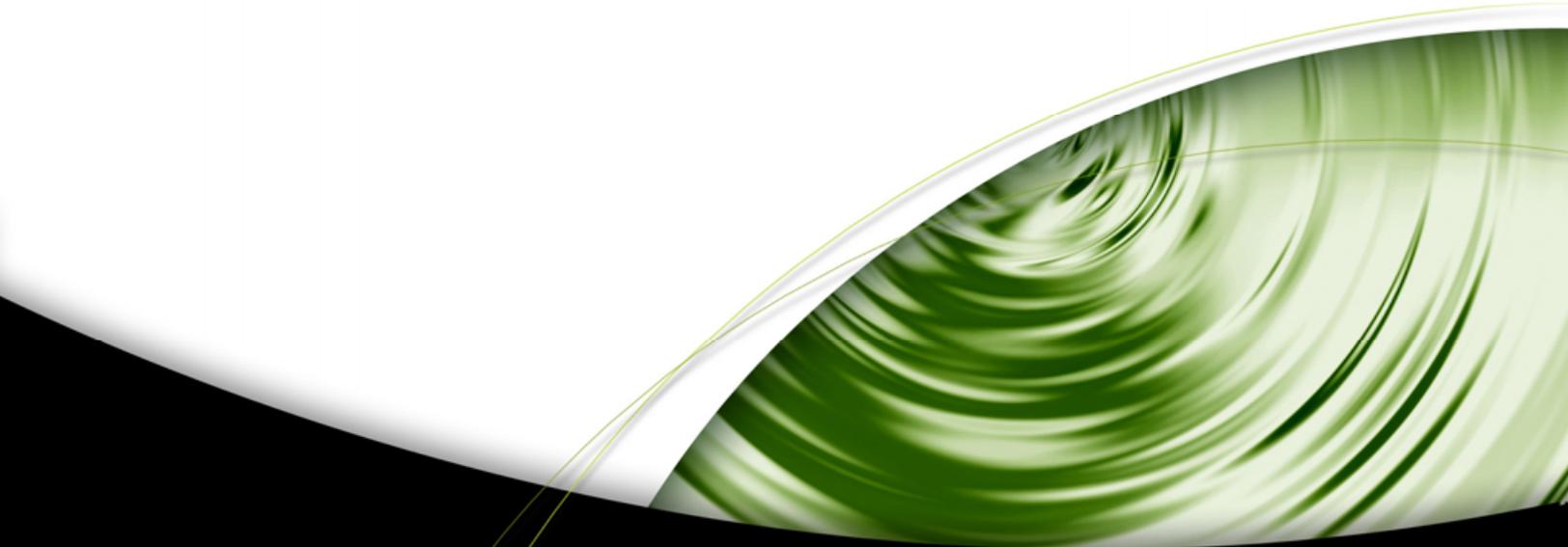




Technical Brief

Introduction to Image
Processing on the GPU



DEVELOPMENT

A decorative graphic at the bottom of the page consists of a white, curved shape that appears to be peeling away from a black background, revealing a vibrant green, swirling, liquid-like pattern underneath. The word "DEVELOPMENT" is printed in a white, bold, sans-serif font across the bottom of the page.

Table of Contents

Image Processing on the GPU	1
Overview	1
Key Resources	2
Other Web Links of Interest.....	3

Introduction to Image Processing on the GPU

Kevin Bjorke

kbjorke@nvidia.com

Image Processing on the GPU

In the most recent generations of Graphic Processing Units (GPUs), the capacities of per-pixel and texturing operations have greatly increased. Millions of these GPUs are already present in the computers of consumers worldwide. Today you can easily apply those texturing and pixel engines, originally designed for 3D modeling and rendering, to many classic image-processing problems to provide tremendous speed increases over CPU-only implementations—and without any compromise in final image quality. This short introduction describes the basic methods of GPU usage for image processing and provides useful pointers to documentation, demo programs, and other developer tools.

Overview

In general, GPUs approach 2D image processing as a restricted form of 3D processing. Quite simply, a quadrilateral polygon is aligned to the desired image-screen rectangle and rendered; either to the screen or to an off-screen buffer in memory. The rendering can be done using either of the common graphics APIs: OpenGL or DirectX.

Each pixel in the rendered image can have image-based texturing applied (up to 16 simultaneous input images per pass can be accessed), and each pixel can run one or more small programs, called *pixel shaders*, to generate the final output color at each individual pixel. The GPU executes these shaders for many pixels at a time in parallel. Multiple passes of rendering may be executed, and the GPU provides additional image-blending hardware to permit images to be built-up in composited layers of arbitrary complexity. The results of each rendering pass, or any disk image, can likewise be passed back into the GPU pixel shader engine as another texture. This means that arbitrarily complex compositing operations can also be expressed as pixel shader operations. Image pixels can even be used as address indices into other images.

Pixel shaders operate in an IEEE floating-point execution environment, though their input and output images can be expressed as 8-bit or 16-bit integers per channel, or as 16-bit or 32-bit IEEE floating-point values per channel. This flexibility and precision permits pixel operations to be executed on the GPU without any loss of image quality when compared to traditional CPU-intensive methods. Since the results can be available at high speed, often the GPU-calculated result in a “preview” window can now also be the final, delivered result.

Key Resources

The developer wanting to learn more about GPU image processing should first look to the recent books *GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics* (2004) and its companion *GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation* (2005). Both books contain detailed bibliographies along with matching CDs containing code samples and demo programs. For details on these publications, go to <http://developer.nvidia.com/GPUGems> and <http://developer.nvidia.com/GPUGems2>.

The *GPU Gems* books contain numerous articles on real-time image processing: real-time glow effects, color controls and profile matching, depth of field techniques, GPU filtering methods and filter-size estimation. The first *GPU Gems* describes the “OpenEXR” high-quality image format (details at <http://www.openexr.com/>); and contains an overall application framework for image processing. The second *GPU Gems* book contains detailed articles on using GPU imaging for such high-performance applications as motion-picture image grading at Sony Pictures and Apple’s GPU-accelerated *Motion* video editing suite.

On the internet, the NVIDIA SDK 9.0, freely available at http://developer.nvidia.com/object/sdk_home.html, provides numerous running samples of 2D and 3D image effects, with source code and documentation for both Open GL and DirectX developers. 2D developers in particular, are guided to the effects “Image Processing” (the latest implementation of the framework described in *GPU Gems*); “Motion Blur as a 2D Post-Processing Effect;” a cellular automata example, “Game of Life;” and several straightforward imaging samples: “Simpler PBuffer,” “Simple RenderTexture,” “Simple Texture Rectangle,” “Filter Blitting,” and “Texture Shader with OffsetTexture2D.”

The SDK also provides a GPU-based C++ framework for use in Photoshop plug-ins is also a part of the NVIDIA SDK, and a complete sample NVIDIA GPU-accelerated HDR paint application that uses the GPU to drive large airbrushes and special effects brushes like “liquefy.” Painting can occur with HDR images in OpenEXR format as well.

NVIDIA’s FX Composer™ shader-development environment, part of the SDK (and also freely available, at http://developer.nvidia.com/object/fx_composer_home.html) is a DirectX-only environment for editing, testing and viewing shaders and their source code. In FX Composer, the developer can work directly with the shader instructions without being distracted by the complexities of managing the DirectX API. FX Composer ships with numerous example shaders, including 2D image processing samples: a number of supplied shaders provide common image-processing functionality (e.g., `scene_edgedetect.fx`, `screenBlend.fx`, `tonemap.fx`). Some even implement simple paint programs, either on a 2D canvas (`scene_paintBrush.fx`) or painting directly onto a 3D model (`scene_paint3d.fx`). Shaders for image-processing include recoloring effects (`scene_colorMatrix.fx`); color-space conversions (`cmyk.fx`); glows (`scene_bloom.fx`); trails (`scene_glowtrail.fx`); flaming halos (`scene_corona.fx`); motion blur (`spinBlur.fx`); and more. Many of these effects are implemented in a context-free “scene command” format that allow these effects to be easily applied to any image or even on top of an existing 3D scene.

Additional GPU tools may be found at <http://developer.nvidia.com/page/tools.html>, including the NVIDIA Texture Tools, a suite of CPU-based texture-creation tools for command-line use or as plugins compatible with Adobe Photoshop™.

Other Web Links of Interest

http://developer.nvidia.com	Product information, samples, tools, and white papers for GPU developers.
http://www.gpgpu.org	An authoritative clearinghouse of general purposes uses for GPUs.
http://www.opengl.org	News and tools for users of the OpenGL API.
http://www.shadertech.com	Shader and GPU news, user and developer forums, sample shader libraries.
http://www.math.sintef.no/gpu/pdf/Dokken_SCCG_2005.pdf	Partial differential equations in image processing using GPUs



Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA, the NVIDIA logo, and FX Composer are trademarks or registered trademarks of NVIDIA Corporation in the United States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2005 NVIDIA Corporation. All rights reserved.



NVIDIA.

NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050
www.nvidia.com