AMD **Accelerated** Parallel Processing TECHNOLOGY

SAMPLE

Convolution_Aparapi

1 Overview

1.1 Location

`$<APPSDKSamplesInstallPath>\samples\aparapi\examples`

1.2 How to Run

See the *Getting Started* guide for how to build samples. You first must compile the sample.

Use the command line to change to the directory where the executable is located. The pre-compiled sample jar is at
`$<APPSDKSamplesInstallPath>\samples\aparapi\examples\Convolution`

Under Windows, type the following command(s).

1. `conv.bat`
   Performs convolution on the default image `testcard.jpg`.
2. `conv.bat -h`
   This prints the help file.

Under Linux, type the following command(s).

1. `./conv.sh`
   Performs convolution on the default image `testcard.jpg`.
2. `./conv.sh -h`
   This prints the help file.

1.3 Command Line Options

Table 1 lists, and briefly describes, the command line options.

Table 1 Command Line Options

Short Form	Long Form	Description
`-h`	`--help`	Shows all command options and their respective meaning.
	`--device`	Devices on which the program is to be run. Acceptable values are `cpu` or `gpu`.
`-q`	`--quiet`	Quiet mode. Suppresses all text output.
`-e`	`--verify`	Verify results against reference implementation.
`-t`	`--timing`	Print timing.
`-v`	`--version`	AMD APP SDK version string.
`-i`	`--iterations`	Number of iterations for kernel execution.

2 Introduction

Using Image filters on photos, desired effects can be sought by applying these filters either on the entire image or just a portion of it. In image filtering, a 2D filter matrix is applied on the 2D image. Then, for every pixel of the image, take the sum of products, where each product is the value of the current pixel or a neighbor of it, with the corresponding value of the filter matrix. The center of the filter matrix must be multiplied with the current pixel, the other elements of the filter matrix with corresponding neighbor pixels. This is called Convolution or Correlation. The difference between Convolution and Correlation is that for Convolution a mirror of the filter matrix is used. The filters with convolution are relatively simple, and its size is uneven, so that it has a center, for example 3x3, 5x5, and 7x7. Depending on the weights chosen, a filter can be made to function as a blur, emboss, pass-through, darken filter, brighten filter, etc. Applying these filters per pixel is still a time-consuming operation Parallel processing using Aparapi can be used to reduce the execution time.

3 Implementation Details

In this example, the `com.amd.aparapi.Kernel` is extended in `ConvolutionKernel.java`. The `public void Kernel.run()` method is overridden in this source file, and convolution using 3x3 image filter per pixel is implemented in this method. A given pixel is filtered only if it is not a boundary pixel; that is, it has valid pixels surrounding it. A call to the `Kernel.execute(range)` method is made in this class, which is used to initiate the execution of `Kernel.run()` over the range 0...n. This class also implements the reference implementation, which is used for verifying that the Aparapi-calculated filtered images match that of the reference. `ConvolutionFilter.java` holds filter weights for a 3x3 filter to be applied to each pixel in the image. This example applies BLUR, EMBOSS, and NONE (pass through) filters on an input image. It repeats the application of these filters on the image over the given number of iterations. `Convolution.java` implements the control code and displays the original image along with NONE, BLUR, and EMBOSS filtered images in a 2x2 grid.

In case the `--quiet` or `--verify` option is used, the display window is closed automatically; otherwise, it waits for the user to close the display window.

4 Recommended Input Option Settings

For best performance, enter the following on the command line: `-i 10 -q -t`

5 References

1. http://code.google.com/p/aparapi/
2. http://lodev.org/cgtutor/filtering.html

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: developer.amd.com/appsdk
Developing: developer.amd.com/
Forum: developer.amd.com/openclforum

The contents of this document are provided in connection with Advanced Micro Devices, Inc. (“AMD”) products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD’s Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD’s products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD’s product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks

© 2014 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, ATI, the ATI logo, Radeon, FireStream, and combinations thereof are trademarks of Advanced Micro Devices, Inc. OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos. Other names are for informational purposes only and may be trademarks of their respective owners.