Title	Evaluating Performance and Productivity of OpenCL on HiFP2.0 Algorithm
Author(s)	Nguyen, Minh Tien
Citation	
Issue Date	2021-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/17150
Rights	
Description	Supervisor:井口　寧，先端科学技術研究科，修士（情報科学）

Evaluating Performance and Productivity of OpenCL on HiFP2.0 Algorithm

1810445 Minh Tien Nguyen

OpenCL (Open Computing Language) is a high-level synthesis & cross-platform standard for building high-performance computing programs and be able to distribute to various processors and hardware accelerators, which contains central processing units (CPUs), graphics processing units (GPUs), digital signal processors (DSPs), field-programmable gate arrays (FPGAs). OpenCL was born to help software developers accelerating their algorithms on hardware with eases. Since the first release in 2008, many large firms have supported and developed OpenCL (such as Intel, AMD, Nvidia, Xilinx). Now, it is being maintained by Khronos Group.

Audio fingerprinting is an efficient technique to represent the characteristics of an audio file. It is lightweight in representation, then requires a small memory capacity and computational cost for management.

HiFP2.0 is an audio fingerprinting algorithm. By just requiring a small computational cost, it can acquire high throughput and then take advantage of a high-speed network, which is 40Gbps Ethernet or 100Gbps Ethernet.

In the previous works, Araki succeeded in developing HiFP2.0 executed on CPU and had good results in throughput and accuracy. However, it is not sure about the ability when applying the algorithm on other platforms.

In this study, we implement HiFP2.0 in OpenCL executing on Intel FPGA Arria 10, which contains two types of programming models: task-based and data-based parallelism. We tend to provide an insight into how well the OpenCL works on HiFP2.0 algorithm. We investigate the performance and productivity aspects.

To summarize our results, some main points can be listed by the following:

1. The number of code lines can be reduced from 5 - 5.54 times when using OpenCL (a high-level synthesis language) instead of VHDL (a register-transfer level programming language) on an FPGA.

2. The optimized algorithm of OpenCL can achieve more 17.4% performance than the origin, which is intensionally designed for CPUs.

3. The data-based method has the best throughput when applying 512 work-items per work-group and 50 work-groups per execution.

Keywords: OpenCL, HiFP2.0, High-level synthesis, FPGA, Arria 10.