Exploring the features of OpenCL 2.0

Saoni Mukherjee, Xiang Gong, Leiming Yu, **Carter McCardwell**, Yash Ukidave, Tuan Dao, Fanny Paravecino, David Kaeli

Northeastern University

IWOCL '15, May 12-13, 2015, Palo Alto, CA, USA
ACM 978-1-4503-3484-6/15/05.
http://dx.doi.org/10.1145/2791321.2791326

Outline

- Introduction and evolution of OpenCL
- OpenCL 2.0- new features
- Applications used to explore these features
- Result and analysis

OpenCL

- Programming and runtime framework
- Executes applications across heterogeneous platforms
- First version, OpenCL 1.0 was released in 2009

OpenCL 1.0: Basic programming model

OpenCL 1.1/1.2: Memory management & fine grain control

OpenCL 2.0: Support for emerging hardware capabilities & improved programmability

OpenCL 2.0 Features

- Shared Virtual Memory
- Dynamic Parallelism
- Generic Address Space
- Image Support
- Android Installable Client Driver Extension

Features

- **Shared Virtual Memory**
- **Dynamic Parallelism**
- **Generic Address Space**
- **Image Support**
- **Android** Installable Client Driver Extension

Bigger picture

- **Goal**: A benchmark and micro benchmark suite with OpenCL 2.0 applications
- Features that are interesting in HSA and OpenCL 2.0

HPC	Mobile/Embedded	Big Data
1. Spectral Clustering 2. Connected Component Labeling 3. Graph-based Segmentation 4. Periodic Greens Function 5. Feature Selection and Outlier Detection 6. 2-D Finite Difference Time-Domain	1. N-channel IIR Filtering 2. Multi-channel Noise Filter using FIR Filtering 3. Speech Recognition using Hidden Markov Models 4. AES encryption/ decryption 5. Convolution Neural Network 6. Shallow Water Simulation 7. Color Histogramming	1. Rating System using MapReduce 2. K-means Clustering 3. Page Rank 4. Bayesian Estimation for Adaptive Spam Filtering/Learning 5. Gene Sequencing

Bigger picture

- **Goal**: A benchmark and micro benchmark suite with OpenCL 2.0 applications
- Features that are interesting in HSA and OpenCL 2.0

HPC

1. Spectral Clustering
2. Connected Component Labeling
3. Graph-based Segmentation
4. Periodic Greens Function
5. Feature Selection and Outlier Detection
6. 2-D Finite Difference Time-Domain

Mobile/Embedded

- N-channel IIR Filtering
- Multi-channel Noise Filter using FIR Filtering
- Speech Recognition using Hidden Markov Models
- AES encryption/ decryption

5. Convolution Neural Network

- Shallow Water Simulation

7. Color Histogramming

Big Data

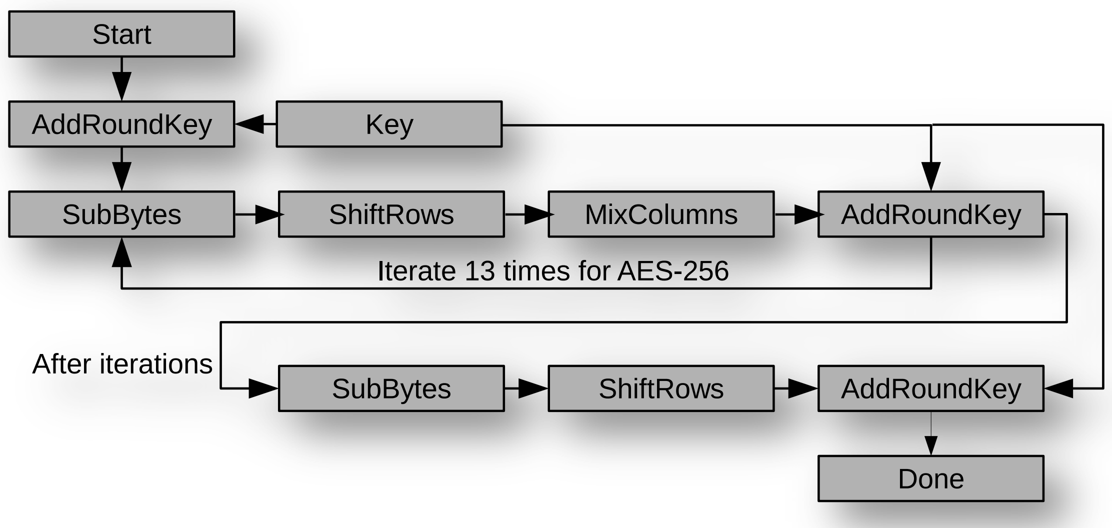
1. Rating System using MapReduce

- K-means Clustering

3. Page Rank
4. Bayesian Estimation for Adaptive Spam Filtering/Learning
5. Gene Sequencing

Exploring the benefits of OpenCL 2.0

CyberSecurity: The Advanced Encryption Standard (AES)

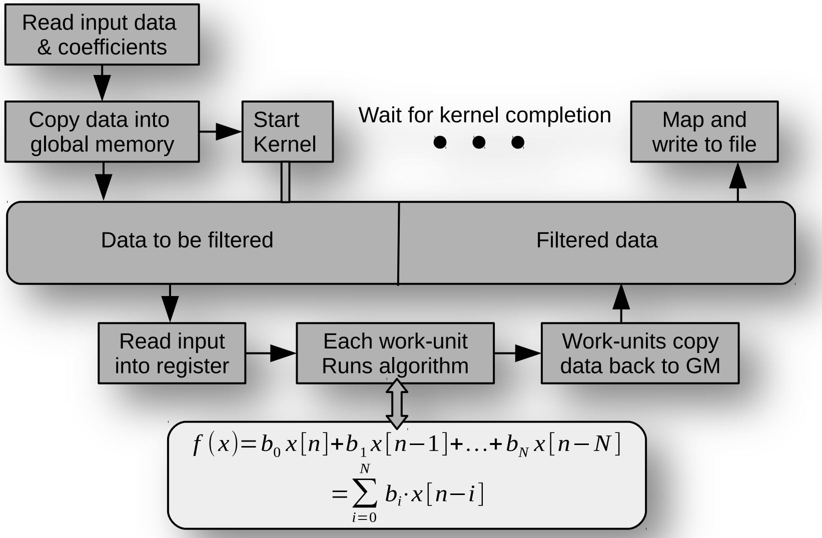


- Adopted by US government for encryption
- Input as plain text with 256 bit key produces cipher text
- Blocks running concurrently
- Our results show that key expansion is faster on CPU than GPU
- 14 rounds of AES-256 are performed on GPU

Exploring the benefits of OpenCL 2.0

Signal Processing: Finite Impulse Response Filtering

- **Impulse Response** of finite duration
- Input: x[1…n] and b[1…N] ⟶ output: f[x]
- Number of taps: N = 1024
- Synthesized audio stream input
- Uses weighted reduction - very common parallel operation



Exploring the benefits of OpenCL 2.0

Signal Processing: Infinite Impulse Response Filter

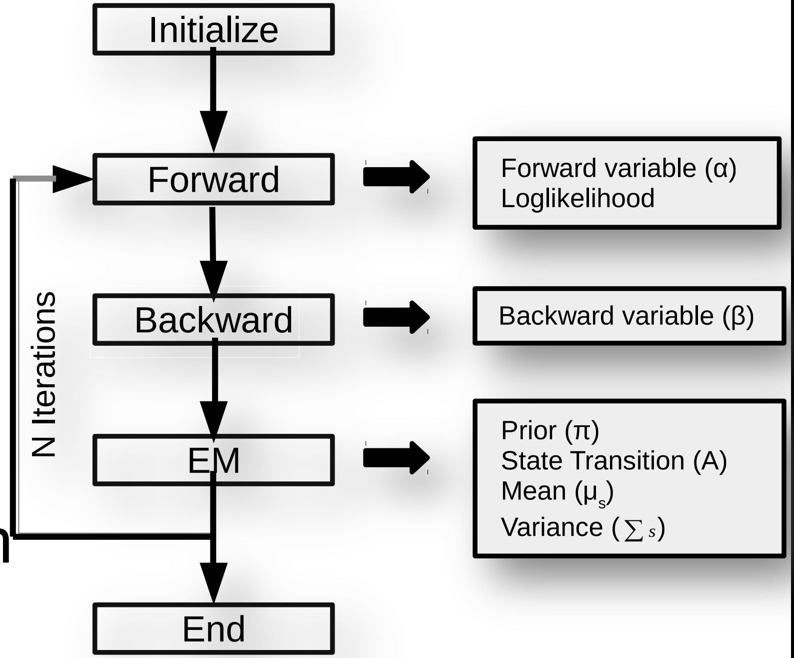
- Less processing power than FIR for same design
- Decomposed into multiple parallel 2nd-order (real and complex) IIR for performance
- N_1 – number of real poles
- N_2 – number of complex poles
- Number of channels = 64
- FIR coefficient: $c_0 = 3.0$
- Synthesized audio stream input

$$H^z(z) = c_0 + \sum_{i=1}^{N_1} \frac{f_i}{1 + e_i z^{-1}} + \sum_{i=1}^{N_2} \frac{f_{N_1+2i-1} + f_{N_1+2i} z^{-1}}{1 + e_{N_1+2i-1} z^{-1} + e_{N_1+2i} z^{-2}}$$

Exploring the benefits of OpenCL 2.0

Statistical Modeling: **Hidden Markov Models**

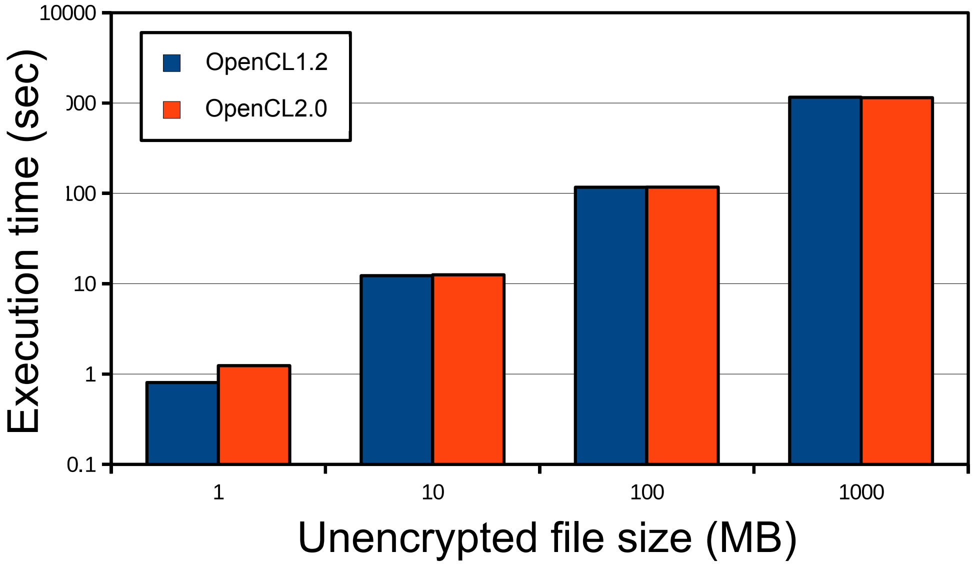
- Probabilistic meaning of hidden states without prior knowledge
- Targeting isolated word recognition
- Matrix form used for coalescing and computational efficiency
- Uses operations including
  - Matrix multiplication
  - Matrix vector
  - Parallel reduction
- Uses data & thread level parallelism



Ongoing OpenCL 2.0 Evaluation

- Baseline: OpenCL 1.2
- GPU model: AMD Radeon R9 290x (reported in paper)
  - Current use: AMD A10-7850K Radeon R7, Kaveri APU
- GPU Architecture:
  - Compute Cores: 12 (4 CPU & 8 GPU)
  - Global Memory: 512 MB
  - Max Clock frequency: 720 MHz
- GPU Driver: 1642.5 (VM)

AES Results



Optimizations explored:

✓ SVM

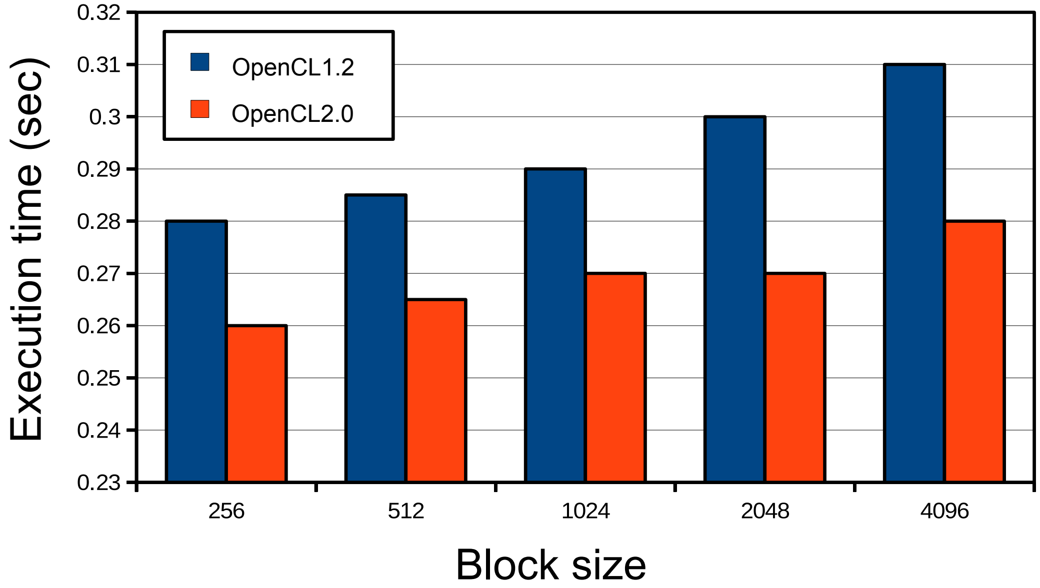
X Dynamic Parallelism

- Input files contain excerpts of a book
- Input sizes are varied from 1MB to 1,000MB with constant 256 bit key
- Small benefits from SVM, which grow with input file size
- Child kernel is memory intensive, inhibiting dynamic parallelism

FIR Results

Optimizations explored:

✓ SVM

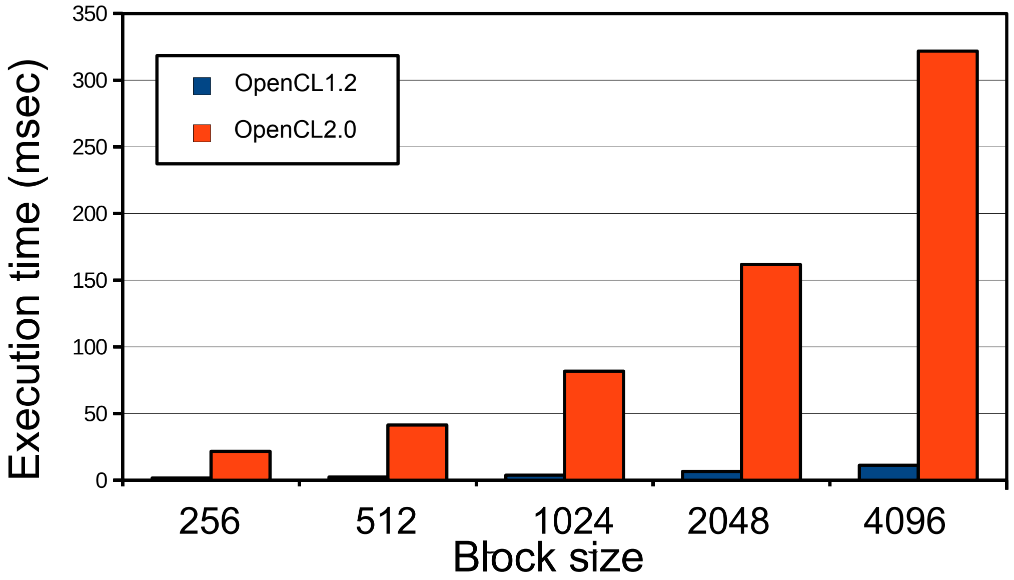


- FIR is a streaming application with different block sizes
- Results show that same kernel runs faster in OpenCL 2.0
- Consistent benefits from SVM, which grow with input block size

IIR Results

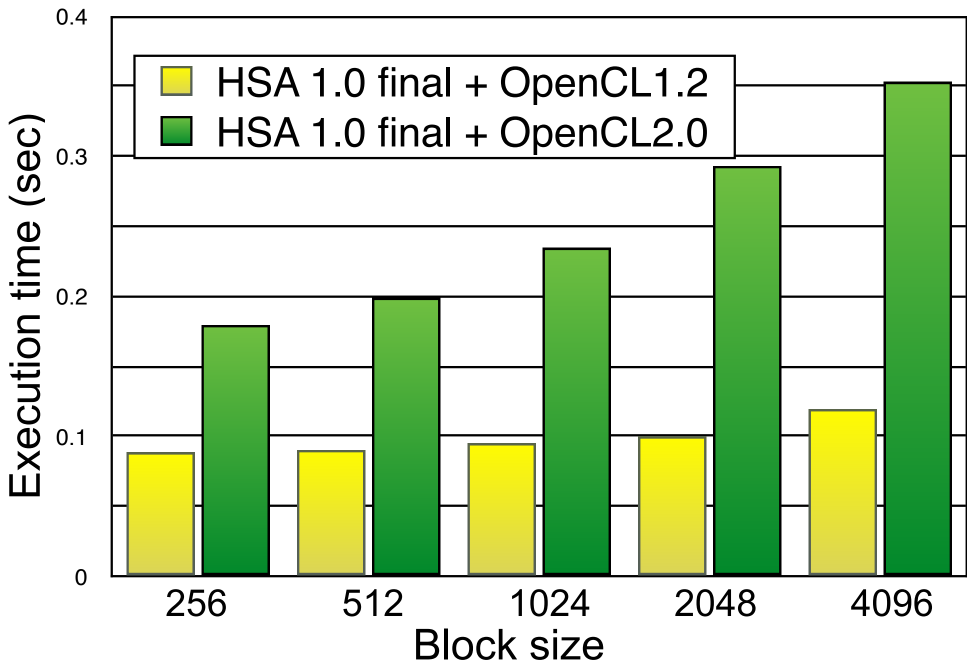
Optimizations explored:

✓ SVM
X Workgroup function



- Interesting feature - parallel reduction
- Workgroup function is useful for reduction, but did not work well

Exploring Workgroup Function further in IIR

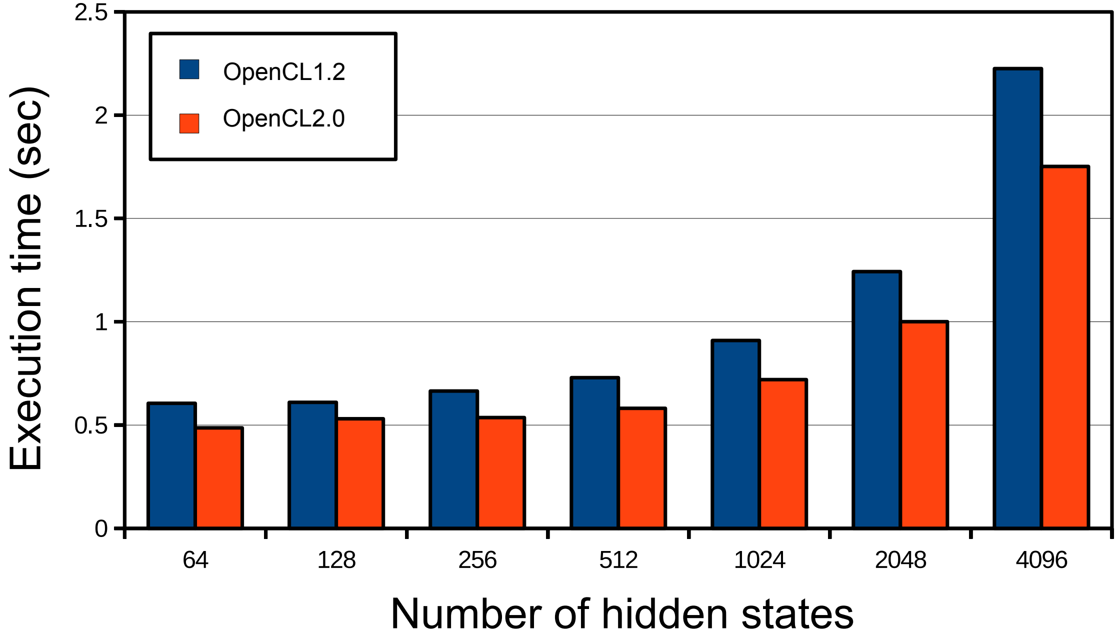


- Workgroup function is useful for reduction, but did not work well in OpenCL 2.0
- It works better in HSAIL on HSA, but not as good as reduction

Hidden Markov Model Results

Optimizations explored:

✓SVM
✓Dynamic Parallelism



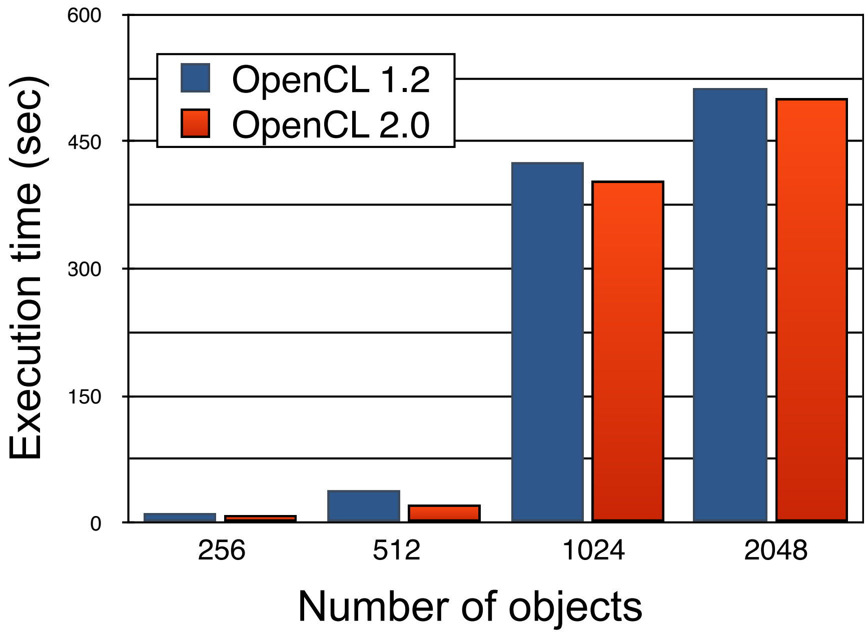
- Updating expected values for each hidden state is an independent operation - perfect for Dynamic Parallelism!

K-means Results

Data Mining: K-means algorithm

Optimizations explored:

✓ SVM



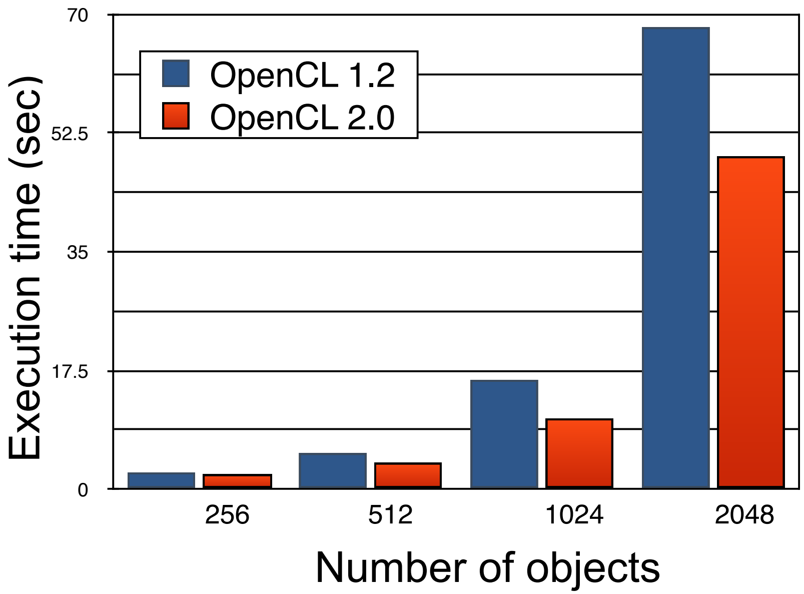
- Well known clustering algorithm.
- K-means with different number of objects, 34 features, 5 clusters
- Input file contains features and attributes
- Consistent benefits from SVM

Shallow Water Simulation Results

Physics simulation: Shallow Water Engine

Optimizations explored:

✓ SVM



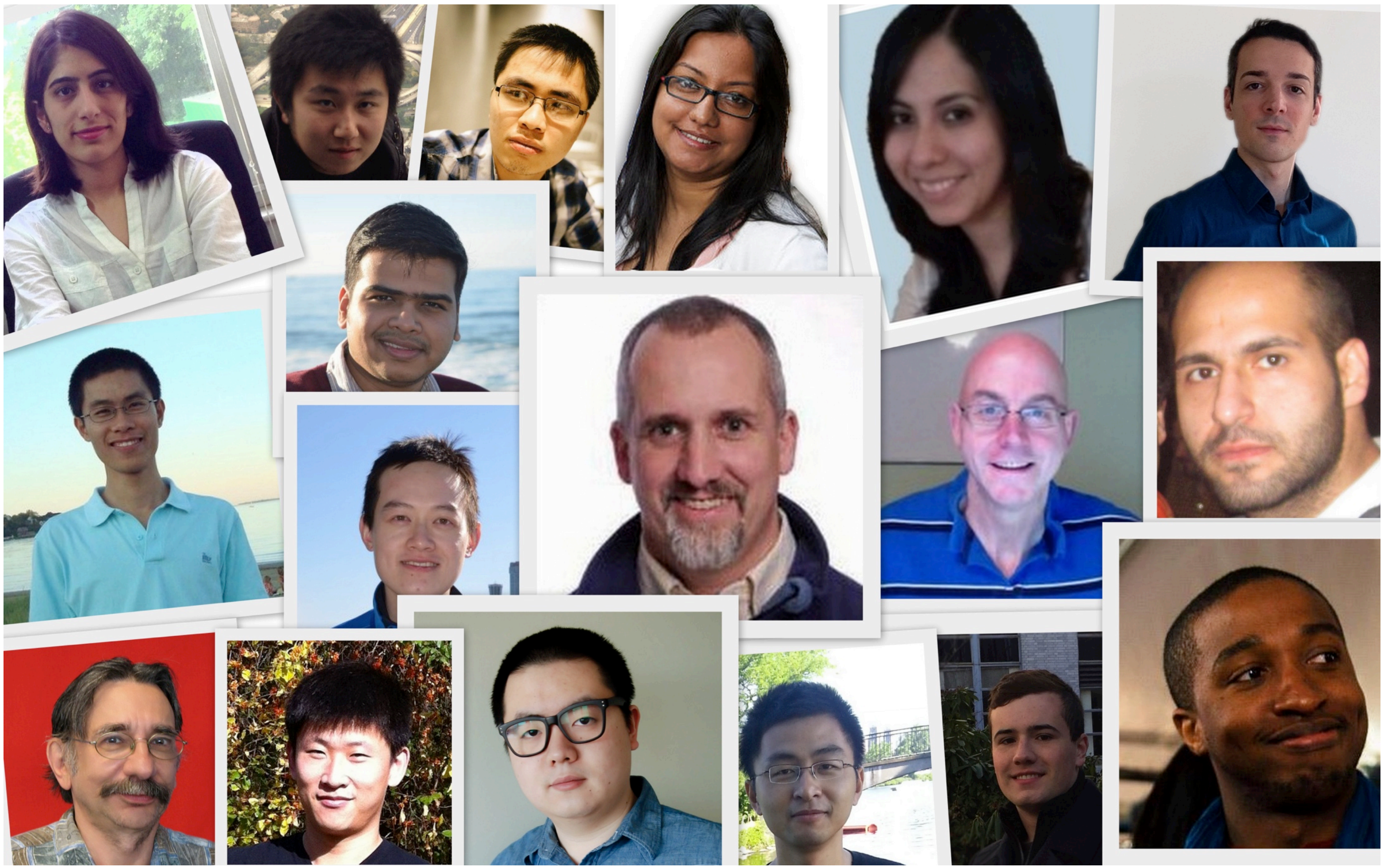
- Depicts complex behavior of fluids, wave modeling for interactive systems
- Predicts matters of practical interest, e.g. internal tides in strait of Gibraltar
- Mathematically and computationally intense, so expensive to do real-time

Summary

- OpenCL 2.0 introduced new features
- We have explored the benefits of using them with some benchmarks from a variety of domains
- SVM provides consistent benefits
- Exploring issues with utilizing the work-group function
- The benchmark suite will be released Summer 2015

Northeastern University Architecture (NUCAR) Group

http://www.ece.neu.edu/groups/nucar



THANK YOU TO OUR SPONSORS

NSF

AMD

QUALCOMM

SAMSUNG

NVIDIA
CUDA RESEARCH CENTER

MIT Lincoln Laboratory

ANALOG DEVICES

ANALOGIC

GORDON CenSSIS

HSA FOUNDATION

BK Medical