Baboons on the Move

Refresher

1) Overall Project Goal

2) Where we are currently at

Performance Optimization Team

– Binghong(Leo) Li, Debaditya Basu, Sananya Majumder, Baiqiang Zhao, Zhi Wang

Motivation

- The current implementation of the algorithms is not feasible for relatively-long video processing
- Current implementation with Python spends around 1.5 hours to process around one minute of the video

Python → C++

CUDA

Goals

1. Implement the remaining Baboon Tracking algorithms in C++ and verify that the functionality matches the Python code.

2. Re-implement the Kalman Filter/Particle Filter code base.

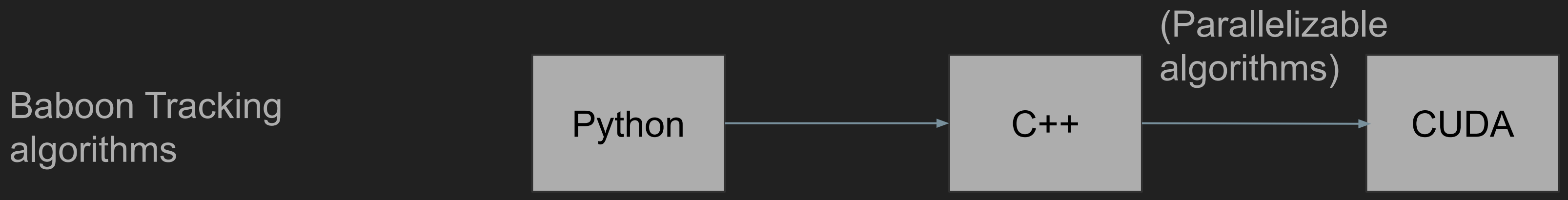
3. Improve the performance of the algorithms in C++ over Python code.

4. Do a feasibility study and implement parallelizable algorithms on CUDA and compare the performance with C++ code.

Workflow

Some of the algorithms currently used for Baboon Tracking in Python-

- Motion Tracking - CUDA?
- DBScan - C++/CUDA?
- Hysteresis - C++/CUDA?
- Particle Filter/Smoother* - C++/CUDA?



Baboon Tracking algorithms

Python

C++

(Parallelizable algorithms)

CUDA

Timeline (optimistic)

3rd Week	Understanding existing Python based implementation
4th Week	Literature review on (multi-target) motion tracking/prediction
5th Week	Ensuring functionality correctness of C++ based code
6th Week	Optimising code to achieve better performance
7th Week	Feasibility study of parallelizable algorithms
8th Week	Ensuring functionality correctness of CUDA based code
9th Week	Comparing Performance
10th Week	Final Wrap Up and Presentation Preparation

Thank you!