**CS 219 Projects, Winter 2021**

**Topic 1: Cloud Anchor Extraction with Spark (Mentor: Jinghao Zhao** [**jzhao@cs.ucla.edu**](mailto:jzhao@cs.ucla.edu) **)**

To enable the collaborations between multiple users’ views for AR applications, feature points need to be extracted from objects in each camera stream. If users share some common objects in the views, these objects can be the anchor to establish the connection between real-world positions. However, extracting the feature points with current algorithms like SIFT is computation-intensive. This project explores the server cluster to speed up the processing with Spark, including:

* Decoupling the procedures in the feature extraction to build the distributive processing with Spark
* Combining the feature extraction algorithm like SIFT with ML models like Yolo to improve the accuracy
* Comparing the performance between the single-server solution and the distributed solution

**Topic 2: Collaborative Video Stitching with Spark (Mentor: Boyan Ding** [**dboyan@cs.ucla.edu**](mailto:dboyan@cs.ucla.edu) **)**

With the collaborations between multiple users, the single view from one user can be extended by combining several real-time streams together. 360-degree video can be stitched at the cloud from multiple webcams. When the resolution and number of views increases, the processing overhead will make real-time stitching challenging. This project explores the server cluster to speed up the processing with Spark, including:

* Decoupling the tasks in the video stitching to build the distributive processing with Spark
* Comparing the performance between the single-server solution and the distributed solution

**Topic 3: Distributed OpenPose with Spark (Mentor: Jinghao Zhao** [**jzhao@cs.ucla.edu**](mailto:jzhao@cs.ucla.edu) **)**

OpenPose provides video analytics for body, foot, face, and hands estimations. Due to the processing overhead, performing the human skeleton analytics with OpenPose on large-scale videos is still challenging. The server cluster in the cloud provides new opportunities and source videos can be split into frames, which can be processed distributively. This project explores to leverage the server cluster to speed up the OpenPose processing with Spark, including:

* Develop the distributed processing system with Spark for video analytics so that the video frames can be split into several server nodes for further processing
* Comparing the OpenPose performance between the single-server solution and the distributed solution

reference: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>

**Topic 4: Distributed Large-Scale Mobile Data Analytics (Mentor: Zhaowei Tan** [**tan@cs.ucla.edu**](mailto:tan@cs.ucla.edu) **)**

For the past 5 years, we have collected a total of 6.4TB of logs for LTE networks on the mobile devices. The overhead of processing all of them for analytical tasks, even on a powerful cloud server, is computation-intensive and time-consuming. This project aims at setting up a platform for scalable mobile data analytics. The programs for single-server processing are readily available and will be provided. Your tasks in this project mainly include:

* Decoupling the current mobile data analytics pipeline into modules;
* Porting the analytical codes for each module to a parallel Spark version;
* Comparing the performance between the single-server solution and the distributed solution.

**Topic 5: Scalable Base Station Analytics at the Edge (Mentor: Zhaowei Tan** [**tan@cs.ucla.edu**](mailto:tan@cs.ucla.edu) **)**

We have deployed a virtualized 4G base station in a Linux server. It is valuable yet challenging to analyze the log data generated by the base stations; especially with the future dense base station deployment for 5G scenarios. We thus target at setting up an edge server cluster which collects and analyzes the data from multiple base stations. You will be granted access to the virtualized base station testbed. Your tasks in this project mainly include:

* Streaming the detailed base station log data into the edge Spark platform;
* Realizing a few example analytical programs that process the data distributedly;

**Topic 6: Parallel Misconfiguration Detection (Mentor: Zhehui Zhang** [**zhehui@cs.ucla.edu**](mailto:zhehui@cs.ucla.edu) **)**

Network operators deploy millions of base stations to provide network service and deploy complex configurations to manage these base stations. To guarantee configurations works as expected, operators run misconfiguration detection to verify deployed configurations are errorless. However, it takes hours to process collected configurations from mobile devices. This project aims to paralyze existing detection scheme and speed up the processing with cluster. Your tasks in this project mainly include:

* Decoupling misconfiguration detection for different base stations to enable parallelization.
* Converting the legacy detection implementation to a parallel Spark version;
* Comparing the performance between the legacy solution and the parallel solution.

**Topic 7: Performance Debugging and Improvement of Multimedia Streaming Applications in Spark (Mentor: Boyan Ding [dboyan@cs.ucla.edu](mailto:dboyan@cs.ucla.edu) )**

While Spark is mainly designed to process large-scale table-based data. The introduction of streaming functionality opens up the possibility for multimedia processing. However, the performance can often be suboptimal for certain reasons, especially on smaller clusters or mobile devices. This project explores the performance aspect of this kind of spark application, including:

* Use profilers to debug performance bottlenecks of existing or self-developed spark streaming applications that processes multimedia content (e.g. object recognition). ***Ideally, you need to have some low-level knowledge of JVM and linux operating system, and know how profilers work***.
* According to the bottlenecks found, make improvements to various components, including application code, Spark parameters, or even internal Spark logic to improve application performance.