Emerging Issues in Cloud Workshop of CloudCom 2014

An Instructional Cloud-Based Testbed for Image and Video Analytics

Thomas J. Hacker
Computer & Information Technology
Purdue University
West Lafayette, Indiana USA
thhacker@purdue.edu

Yung-Hsiang Lu
Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana USA
yunglu@purdue.edu

Abstract— **This paper describes a cloud-based software infrastructure for teaching big data analytics. Using this infrastructure, students can retrieve and analyze real-time visual data retrieved from globally distributed network cameras.**

Keywords—cloud computing, video analysis, education

I. Introduction and Motivation

Delivering engaging courses in big data analytics depends on using relevant real-world examples and data for class projects and assignments for students. Large static datasets are a potential source of information; however they can become quickly outdated, and updating them between semesters can be a challenge when storage resources are limited. One possible approach is to utilize live streaming sources of data from the Internet from distributed sensors and streaming data sources. Over the past year, a group at Purdue led by co-author Yung-Hsiang Lu has developed a new network camera infrastructure called Continuous Analysis of Many CAMeras (CAM2) that seeks to simplify the process of discovering and using thousands of Internet connected video cameras (network cameras). CAM2 contains many geographically distributed cameras as the data source and uses cloud computing instances to perform analytics on the data from these cameras.

II. The Need for Dynamic Datasets for Courses

Courses in big data analytics have emerged over the past few years. These courses seek to provide education and training in the underlying theoretical concepts of data analytics, such as algorithms, complexity, and graph theory, as well as the practical aspects of designing, creating, and using cyberinfrastructure to support analytics. Providing traditional lectures and assessment is fairly straightforward for these courses. The problem of providing relevant, recent, and vital datasets for student use, however, is problematic. The challenges facing the instructors seeking to provide these data are several fold. First, large datasets from sources such as Twitter can become quickly outdated. Students are often very interested in exploring the most recent and relevant datasets, especially when they arise from current events in the news or society. Second, very large datasets can be expensive to store, preserve, and transfer between repositories and computational facilities. Another aspect of this problem is the need for persistent infrastructure for student use over the course of a semester. Ideally, each student would have access to a small-sized cluster consisting of several computational nodes over the course of the semester. The costs involving in providing an infrastructure at this scale for a large class would be substantial, and ultimately unsustainable over a long period.

Ultimately, the problem is: how can instructors provide recent and relevant sources of large datasets to support teaching for data analytics courses as well as support research activities (for students and faculty) based on these data?

The approach described in this paper is based on exploiting publicly available data from a large collection of global Internet-connected video cameras. These cameras can be accessed to retrieve a stream of MJPEG images, or streaming MPEG or H.264 over a network connection. Using the CAM2 system [3, 6], the group at Purdue discovered over 50,000 cameras by searching the Internet.

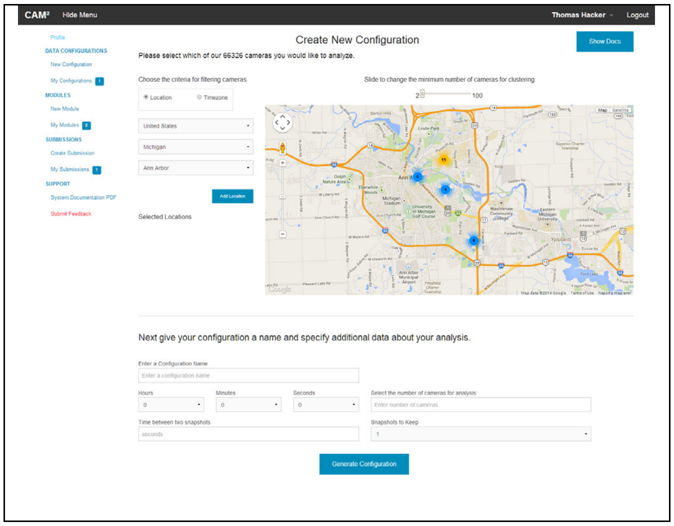
An example of a big data analytics course that is driving this need for a testbed is a joint course in big data analytics that is held simultaneously at Purdue University in West Lafayette, Indiana, USA and the University of Stavanger in Stavanger, Norway co-taught by co-author Thomas Hacker. This course focuses on topics such as data science, MapReduce, Hadoop, functional programming, high performance computing, and reliability. Students are expected to build a small cluster over the semester to be able to run MapReduce jobs as well as use the Hadoop file system. In the past, we set up a small cluster using recycled desktop computers at Purdue University running VMware. Students had difficulties accessing the cluster and downloading large scale datasets for their MapReduce programs and projects. This year, we are using virtual machines on Amazon EC2 to allow students to configure and run their own Hadoop and MapReduce clusters on EC2. Even with this improvement in infrastructure, the problem of identifying and distributing compelling large-scale datasets for student projects remains pressing.

The types of student projects that we envision that could exploit the image data from these cameras include detecting environmental conditions (severe weather, precipitation, smog, etc.), object identification for traffic analysis (i.e. automobiles, trucks, wildlife), and data collection from urban environments.

III. Technology for a Data Intensive Testbed

The CAM2 system developed at Purdue University (cam2.ecn.purdue.edu) [6] is an excellent example of a platform that could be used by students to discover, access, and download streaming image and video sources for analytics. CAM2 currently includes 50,000 cameras available on the Internet that are publically accessible. This system has demonstrated the ability to count people from 2.5 million images over 3 hours using 15 cloud instances. These cameras are a persistent form of a video sensor that collects and transmits image data in various formats (i.e. MJPEG, MPEG), resolutions (frame size in pixels), and frame rates (captured and transmitted frames per second). The development of the CAM2 system was motivated by the need to locate cloud computing instances geographically nearby video sources to ensure a steady, reliable, and high quality source of data to feed computational analytics of the camera image data. Essentially, moving computation to the data. The main objectives of the CAM2 project include:

- provide a computing platform that exploits a database of discovered cameras;
- offer an interface to allow researchers to invoke analytics on collections of camera sources;
- move computing nearby the set of streaming sources based on their geographic location; and
- exploit potentially low prices and quick availability of cloud computing resources near the cameras.



Create New Configuration

Please select which of our 66326 cameras you would like to analyze.

Choose the criteria for filtering cameras

● Location ○ Timezone

United States

Michigan

Ann Arbor

Slide to change the minimum number of cameras for clustering

2 100

Selected Locations

Next give your configuration a name and specify additional data about your analysis.

Enter a Configuration Name: Enter a configuration name

Hours: 0 Minutes: 0 Seconds: 0

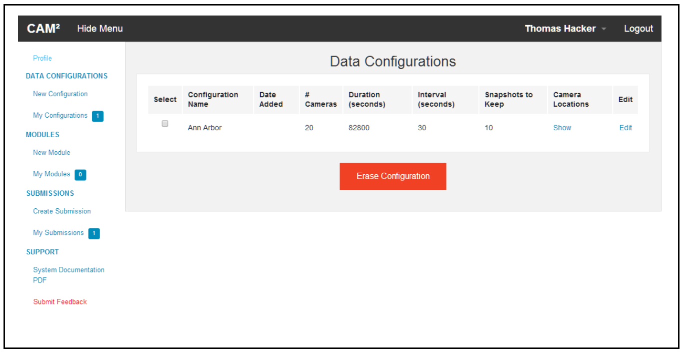
Select the number of cameras for analysis: Enter number of cameras

Time between two snapshots: seconds

Snapshots to Keep: 1

Fig. 1. Selecting cameras based on geographic location in CAM2.

The CAM2 system provides a camera gateway website that allows users to create an account, compose collections of cameras, and to upload or select python analytics modules to process or save image data collected from camera collections.

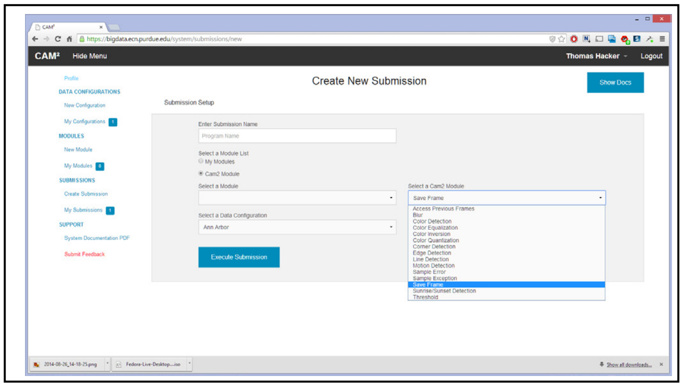


Data Configurations

Select	Configuration Name	Date Added	# Cameras	Duration (seconds)	Interval (seconds)	Snapshots to Keep	Camera Locations	Edit
□	Ann Arbor		20	82800	30	10	Show	Edit

Fig. 2. Configured camera set in the CAM2 system.

An example of a camera collection is shown in Figure 1. In this figure, a map of a geographic area shows a summary of the number of cameras available in a subregion. A user can select cameras in this subregion to create a camera set. Figure 2 shows a configured camera set in which an image is captured every 30 seconds from a collection of 20 cameras located in the Ann Arbor, Michigan area over a period of 23 hours.



Create New Submission

Submission Setup

Enter Submission Name: Program Name

Select a Module List

○ My Modules

● Cam2 Module

Select a Module

Select a Data Configuration: Ann Arbor

Select a Cam2 Module: Save Frame

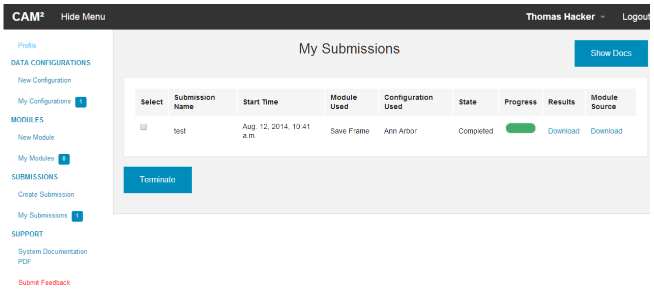
Access Previous Frames
Blur
Color Detection
Color Equalization
Color Inversion
Color Quantization
Corner Detection
Edge Detection
Line Detection
Motion Detection
Sample Error
Sample Exception
Save Frame
Sunrise/Sunset Detection
Threshold

Fig. 3. Selecting an analysis module for the configured camera set

Users can upload their own python analytics modules, or use existing modules provided by the system. One module, for example, simply captures frames for downloading. Figure 3 shows a sample of preexisting python modules available for use within the CAM2 system. Users may also upload their own python analysis modules. Figure 4 shows the resulting analysis queue that allows users to establish an analysis run with a fixed start time and duration to collect and process data using the python modules selected or uploaded by users.

CAM2 currently supports input data of two formats: JPEG images and MJPEG videos. The output can be images as well as text files. The output images may be the raw data or the processed data. The text data may include numeric information, such as the number of detected objects.

The types of data collected by CAM2 include motion JPEG (providing a sequence of JPEG images) and MJPEG. The amount of data collected for analysis and download depends on the size of the camera collection, the capture frequency, and the duration of capture. The CAM2 system can package the resulting images files into a compressed ZIP file for downloading, shown in Figure 5.

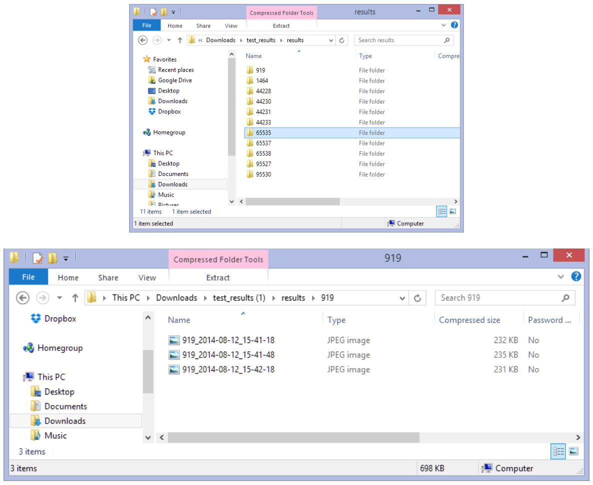


My Submissions

Select	Submission Name	Start Time	Module Used	Configuration Used	State	Progress	Results	Module Source
□	test	Aug. 12, 2014, 10:41 a.m.	Save Frame	Ann Arbor	Completed	🟩	Download	Download

Fig. 4. Analysis for a camera set in CAM2.

In terms of image analysis, experience from the another big-data project, the George E. Brown, Jr. Network for Earthquake Engineering Simulation (NEES) [1] has shown that a clear and time indexed JPEG image is needed for information extraction from an image. For example, image analysis can be used to analyze crack propagation in a concrete structure or measure displacement of a structure under test loads. In NEES, video cameras are used to collect video information from experiments as they are conducted. The video images are stored as a time-series collection of jpeg images that can be individually analyzed and used to understand changes in test specimens over the course of the experiment.



« Downloads › test_results › results

Search results

Name	Type	Compre
919	File folder	
1464	File folder	
44228	File folder	
44230	File folder	
44231	File folder	
44233	File folder	
65535	File folder	
65537	File folder	
65538	File folder	
95527	File folder	
95530	File folder	

11 items 1 item selected

1 item selected

This PC › Downloads › test_results (1) › results › 919

Search 919

Name	Type	Compressed size	Password ...
919_2014-08-12_15-41-18	JPEG image	232 KB	No
919_2014-08-12_15-41-48	JPEG image	235 KB	No
919_2014-08-12_15-42-18	JPEG image	231 KB	No

3 items

3 items 698 KB

Fig. 5. Downloaded image data from the CAM2 system.

Figure 6 shows a sample of a captured and downloaded image from the CAM2 system. Figure 7 shows an example of the result of processing (object detection) an image using the CAM2 system using OpenCV [7] for background subtraction.

Some of the issues identified involved in the collection and processing of large amounts of image data include: wide area network performance, distance from camera to cloud processing instance, and the need for an environment to develop and debug analysis codes. In terms of WAN performance, TCP throughput is directly impacted by the round trip time from data source to data sink.

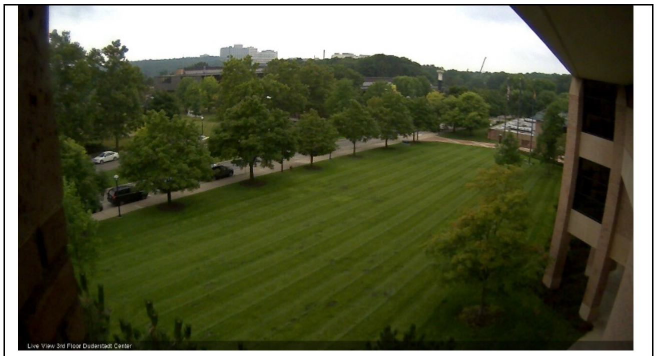


Fig. 6. Example of caputed image from the CAM2 system.

As the distance between the cameras and cloud computing instances grows, the maximum achievable TCP bandwidth is directly affected by the network RTT. Mathis [2] found a direct inverse relationship between TCP bandwidth and network RTT. This issue is especially relevant to streaming video data, which arises from a continuous source that cannot tolerate significant data transmission delays, and for which the analysis may need to be close in time to the captured event in the physical world. If we assume that each streaming video source has a corresponding network diameter surrounding it based on the RTT distance from source to sink, then the achievable maximum TCP bandwidth is a direct characteristic of the network diameter from the camera source. If an analysis task requires a minimum video frame rate or resolution, the analysis for a streaming video source will need to reside within a network diameter frontier around the camera. If data from several cameras are collected for analysis, and if these cameras are near each other geographically (such as within the same city or state), then the aggregate set of frontiers would provide guidance to where the cloud instances would need to be placed.

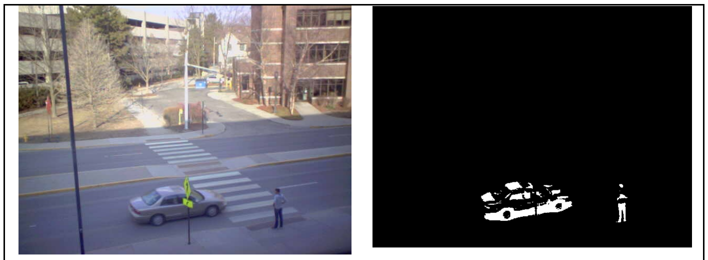


Fig. 7. Example of image analysis from the CAM2 system.

For the MJPEG format, which is not adaptive to changing network conditions, long RTTs can severely limit achievable end-to-end network bandwidth. Other formats, such as MPEG and H.264 are adaptive. However, they sacrifice image quality to maintain frame rate. Hence, if the RTT is long, the usability of adaptive video streams may have limited use for automated analytics.

These types of issues are directly relevant to the building and use of large scale data collection and processing systems, and would provide a useful addition to a course in data analytics. Students using the CAM2 system as a testbed would be able to directly observe the effects of the network on collection rates, and to experiment with camera counts and locations within a camera set within the CAM2 system.

IV. Using the CAM2 testbed for courses

Using the CAM2 system, students and instructors can easily identify relevant sources of streaming image data, capture data from a set of cameras, optionally perform initial filtering or analysis on the images, and then easily download the resulting datasets to their own desktop systems or to their cloud instance for a course.

The CAM2 system could be used by classes in big data analytics with several potential learning objectives. Among these are allowing students to learn to work with diverse, possibly live, and large datasets from a widely distributed sensor infrastructure; to learn to use a cloud computing infrastructure that uses computation at a remote facility that is linked geographically closely with data sources; and to learn to develop data processing scripts that could be used to prefilter and analyze large volumes of data to distill from these data useful events and information to minimize unnecessary data movement.

There are several potential class projects that could use this system. The first is to study the optimal geographic placement of cloud instances given a distribution of camera sets that are disjoint and geographically distant. Another example of a potential class project is to quantitatively measure how the level of occupancy and furniture use in a classroom and laboratory environment. With the advent of new forms of instruction such as problem based learning (PBL) and "maker spaces" that provide flexible design spaces for students. Institutions such as Olin College and the Stanford Institute of Design (*d.school*) are creating new forms of flexible, well equipped workspaces and classroom environments for students. This approach is a radical departure from the traditional lecture hall format used for many generations and universities. These new spaces are often equipped and movable furniture and fixtures that can be connected together, grouped into a work area, or separated as individual units. A potential student project might focus on measuring the frequency and duration of use of fixtures within these spaces to better understand the use of these spaces and which fixtures best fit the needs of the users of the space. The CAM2 system could be used to collect videos of the space over the course of several weeks or a semester. The video frames could then be analyzed to quantitatively measure items such as fixture location, use, and occupancy of different areas of the flexible space over time. These data could then be used to create predictive stochastic models of fixture and space use to aid learning space designers and furniture manufacturers. An integrative example of the use of the system is to use a MapReduce cluster running on a cloud environment to automatically download and analyze a large set of JPEG images collected from a set of cameras. Yet another possible project would be creating a workflow between the CAM2 analysis engine and a separate cloud analytics instance that would integrate streaming image sources with numerical sensor data (such as wind speed, temperature, or sound level) for post processing or near real-time analytics.

V. Experiments

CAM2 had the first alpha release for interested researchers and educators in July 2014. One faculty member outside Purdue has already formed a group using CAM2 as the platform for a course on software engineering. CAM2 has demonstrated the capability of analyzing very large datasets using multiple cloud instances. The largest experiment so far [6] used 15 extra-large cloud instances analyzing images from 1,200 cameras over three hours. More than 2.5 million and 140GB images, at 107Mbps, were analyzed for counting the number of people.

VI. Related Work

In work related to teaching in big data analytics, King [4] describes topics, tools, and techniques in teaching data mining for undergraduates that involves the analysis of large data sets Satyanarayana [5] describes commercial and open source tools that could be used to teach a data mining course. The CAM2 effort described in this paper is a potential source of big data that could be used for the types of big data courses described by King and Satyanarayana.

VII. Expected outcomes and conclusions

It is clear that a new approach to providing relevant and timely datasets for data analytics courses is needed. The approach described in this paper seeks to provide a new model for leveraging publicly available sources of "big data" for use in courses. The benefits of the approach described in this paper is that it is cost effective, scalable, and free from licensing issues or restrictions. This approach has the potentially to greatly enhance the education experience for students in big data analytics courses.

In conclusion, this paper described an approach under development that could be used to utilize existing public sources of image data that can be used for courses and research focused on big data analytics.

[1] Hacker, Thomas J., Rudi Eigenmann, Saurabh Bagchi, Ayhan Irfanoglu, Santiago Pujol, Ann Catlin, and Ellen Rathje. "The neeshub cyberinfrastructure for earthquake engineering." *Computing in Science & Engineering* 13, no. 4 (2011): 67-78

[2] Mathis, Matthew, Jeffrey Semke, Jamshid Mahdavi, and Teunis Ott. "The macroscopic behavior of the TCP congestion avoidance algorithm." *ACM SIGCOMM Computer Communication Review* 27, no. 3 (1997): 67-82

[3] CAM2 System, http://cam2.ecn.purdue.edu.

[4] King, B. and Satyanarayana, Teaching Data Mining in the Era of Big Data, Proceedings of the 2013 ASEE Annual Conference, June 23-26, 2013, Atlanta, GA.

[5] Satyanarayana, A. Software Tools for Teaching Undergraduate Data Mining Course, Proceedings of the ASEE-2013 Mid-Atlantic Fall Conference, University of the District of Columbia, Oct 11-12, 2013.

[6] Kaseb, A., Berry, E., Koh, Y., Mohan, A., Chen, W., Li, H., Lu, Y., and Delp, E., "A System for Large-Scale Analysis of Distributed Cameras", IEEE Global Conference on Signal and Information Processing 2014

[7] Bradski, Gary. "The OpenCV library, " Doctor Dobbs Journal of Software Tools, vol 25., pp 120-126, Nov. 2000.