Project Name: Real-Time Analysis of Microscopic Video Data

Project Objectives

Our goal is to stream in E.Coli and other bacterial microscopy and tag those cells as well as determine important biological features in real time to help biologists more effectively collect data and experiment. The main object would be to develop a lineage model of bacteria between frames as well as determine and tag significant events, such as cells splitting or interacting or dying. This could also provide real time feedback about the window quality of the microscope on the sample, allowing the biologist to move the window to get a better view(such as making sure a minimum of multiple phenotypes are within the region depending on the experiment.)

Significance

Cell microscopy and observation is largely done by specialized human workers as humans are the gold standard of image detection and recognition as well as knowing what important biological features to assess. This is costly as the person's time could be used on other facets of research and human labor itself is costly.

Cell experiments are also performed in stages. That is, one single gene or biological factor is looked for and analyzed through microscopy (generally analyzing the images after the experiment and after applying different filters to the data to make things stand out), after which another experiment is performed. If biological features that are sought after could be analyzed and detected in real time, the experiments could become more modular, allowing a researcher to apply a chain of experiments together, saving time and money.

This is very complex feature analysis that would have extremely accurate though. So the bare bones fundamental structure, such as performing lineage tracking and extracting features, would have to be able to function well.

Features: Use Case/Scenario

This is similar to what is discussed previously in the *Significance* portion, but the exact use cases will be developed based on the research environment from which we obtain our microscopic video data. This could be helping to identify bacteria cells or analyzing phases of yeast cell growth. We can also try tagging the biological mechanisms that the research talks about.

Approach

- Data Sources
    - Published cellular research with microscopy movies (frames taken every 2-5 minutes). Youtube also offers video data of microscopy as well, generally with frames taken within a shorter window, but for lesser duration. The research data also tries to analyze more complicated mechanisms while the youtube data displays features such as budding and reproduction.
    - Research papers that we have looked at may be found in our literature review, and an example of youtube data would be the following:

https://www.youtube.com/watch?v=iOvrq6ssy2Y
https://www.youtube.com/watch?v=BTHYaf-EuYs

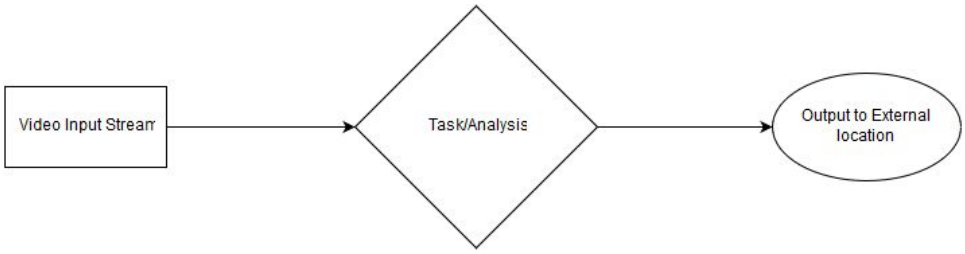
- Analytic Tools
    - Spark
    - OpenImageJ
    - Spark MLLib
- Analytical Tasks
    - Track and detect cells from one frame to another
        - Requires ability to detect individual cells
        - Requires ability to differentiate or map cells from one frame to another
    - Extract many features from cells.
        - Note: This could be physiological such as size and shape or behavior such as splitting and interactions among cells, or triggered, such as turning on a fluorescent light to see a gene expression tag glow.
    - Determining if slide needs to be adjusted relative to microscope
    - Counting number of cells in view
- Expected Inputs/Outputs
    - The expected input would be a video or a streamed video of cell microscopy where we output a lineage tree or video as well as biological tagging (such as knowing when it's splitting) or what gene or protein it contains via fluorescent tagging.
    - Other outputs: Metadata on video such as attribute measurements at a specific time or average values over length of video
- Algorithms
    - Random Forest/Decision Tree algorithms
    - Object detection, aggregation, and segregation using image analysis.
        - Note: The current key point extraction technique using SIFT as shown in Lab2,3 does not accurately find single E.Coli in our videos. E.Coli are all fairly regular and contain a lot of the same curvature (that SIFT tries to find and map as a key point). E.Coli also grow and wiggle changing their shape, making curvature based methods difficult for detecting it. We also have yeast cell data tend to have more distinctive features within them and around them, which could be easier to detect and track.

Related Work

- Open Source Projects
    - MicrobeJ for ImageJ (I don't think MicrobeJ is opens source though, but ImageJ is)
- Literature Reviews (some are from proposal)
    - http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4820279/ (ecoli videos)
    - http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0032621 (yeast)

    - http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003396#s5 (yeast)
    - http://www.ncbi.nlm.nih.gov/pubmed/27572972
    - Chen, Zhou & Wong. “Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy.” IEEE Transactions on Biomedical Engineering, vol. 53, no. 4, April 2006. http://ieeexplore.ieee.org/document/1608529/
    - Dewan, Ahmad & Swamy. “A Method for Automatic Segmentation of Nuclei in Phase-Contrast Images Based on Intensity, Convexity and Texture.” IEEE Transactions on Biomedical Circuits and Systems, vol. 8, no. 5, October 2014. http://ieeexplore.ieee.org/document/6762958/
    - Goutam & Sailaja. “Classification of acute myelogenous leukemia in blood microscopic images using supervised classifier.” 2015 IEEE International Conference on Engineering and Technology (ICETECH), March 2015. http://ieeexplore.ieee.org/document/7275021/
    - Tran, Pham & Zhou. “Cell phase identification using fuzzy Gaussian mixture models.” Proceedings of 2005 International Symposium on Intelligent Signal Processing and Communication Systems, 2005. http://ieeexplore.ieee.org/document/1595447/

- Application Specification
    - OpenImaj/SIFT
    - More to come for Machine Learning (ML) aspect of project
- System Specification/Software Architecture
    - Spark - ML to assign probabilities of specific features in a frame
    - Storm - efficiently handle feature extraction from cell video data
    - Kafka - transferring video data between systems
    - More to come at a later date as we develop ML portion of project
- Design of Big Data Analytics Server
    - Parallelism/Distribution: Task/Data
        - Data parallelism of splitting image into frame then applying ML algorithms to train for object classification.
        - Task parallelism within a frame to determine edges/sift points/biological tagging through color analysis or shape analysis.
    - Features, workflow, technologies:



        - See the following descriptions to better understand the workflow diagram:
            - Video Input Stream: Microscopic Video Data

- Task/Analysis: Task being performed on data
    - Not defined exactly, but see Use Case section to see possible ideas
- Output to External Location: This could be to a user via email or an external monitor on which the video and the analysis is displayed.

- Design of Mobile Client (smartphone/web): N/A
- Existing Applications/Services Used: Time-lapse microscopy videos (http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4820279/)

Implementation

o Implementation of server using AWS to hold data and process via Spark and Storm
o Implementation of mobile client: N/A

Documentation

- Proposal: https://github.com/DRinKC/RT-BigData-Project_Team1/tree/master/Proposal
- Lab3 (Clarifai for acquiring and applying image tags for frames; mainframe detection): https://github.com/DRinKC/RT-BigData-Project_Team1/wiki/Lab-3
- Lab4 (object detection/tracking): https://github.com/DRinKC/RT-BigData-Project_Team1/wiki/Lab-4

Project Management: Implementation status report

- Work completed:
    - Description: Proposal, Lab 3, Lab 4
        - Responsibility:
            - Bill, Justin and David for Proposal: 5 hrs (Brendan joined late)
            - Bill & Brendan for Lab 3 part 1: Clarifai for acquiring and applying image tags for frame (4 hrs)
            - Justin Lab 3 part 2: mainframe detection (3-4 hrs)
            - David Lab 4: object detection/tracking (3-4 hrs)
        - Contributions (members/percentage): roughly 25% each
    - Work to be completed
        - Description:
            - Implementing ML algorithms (and others) to more accurately detect cells and/or cell behavior.
            - Modifying our SIFT detection to see if it's possible to detect E.Coli using the lab tutorial methods. Also test on yeast cells.
            - Knowing location of each cell during a frame of video (this would be classifying that something is a cell). This would be the first step in building a lineage tree.
            - Adding more annotations (i.e., not just labels, but also functions and/or cell lineage, etc.) --- to be fleshed out at a later date

                - Implement a moderately simple way for non-technical scientists to submit data and receive results
                - Figure out how to pass data between processing applications in real time
                - Determining precision of results to increase validity for scientific research
            - Responsibility (Task, Person):
                - Specific tasks not yet specified, but it will involved everyone.
                - Current/Immediate tasks involve getting video data to use.
            - Time to be taken (estimated #hours):
                - Immediate take of getting video data: 5- 10 hrs.
                - Everything else, at least 5-10 hrs per person (rough estimate)
- Issues/Concerns:
    - Gathering enough videos
    - Implementing our project on real-time microscopic data
        - Ability to adjust slide in microscope real time to collect valid data streams
    - Accuracy/Precision of feature recognition on small objects in limited resolution videos
    - Extracting information beneficial to microbiological studies from video data