CS5543 Real-Time Big Data Analytics Project Increment I
Pooja Shekhar(23)

- Project Title & Team Member

  Real-time Object Detection & Tracking and Scene Categorization on Traffic Data in Real-time video.

  – Pooja Shekhar

- Project Goal and Objectives

  Object detection and tracking are important and challenging tasks in many computer vision applications such as surveillance, vehicle navigation, and autonomous robot navigation. Video surveillance in a dynamic environment, especially for humans and vehicles, is one of the current challenging research topics in computer vision. It is a key technology to fight against terrorism, crime, public safety and for efficient management of traffic. The work involves designing of the efficient video surveillance system in complex environments. In video surveillance, detection of moving objects from a video is important for object detection, target tracking, and behavior understanding. Detection of moving objects in video streams is the first relevant step of information and background subtraction is a very popular approach for foreground segmentation. In this thesis, we have simulated different background subtraction methods to overcome the problem of illumination variation, background clutter and shadows. Detecting and tracking of human body parts is important in understanding human activities. Intelligent and automated security surveillance systems have become an active research area in recent time due to an increasing demand for such systems in public areas such as airports, underground stations and mass events. In this context, tracking of stationary foreground regions is one of the most critical requirements for surveillance systems based on the tracking of abandoned or stolen objects or parked vehicles. Object tracking based techniques is the most popular choice to detect stationary foreground objects because they work reasonably well when the camera is stationary and the change in ambient lighting is gradual, and they also represent the most popular choice to separate foreground objects from the current frame. Surveillance networks are typically monitored by a few people, viewing several monitors displaying the camera feeds. It is very difficult for a human operator to effectively detect events as they happen. Recently computer vision research has to address ways to automatically some of this data, to assist human operators. Detecting and recognize objects inside video stream (object, face, ...) and auto-tag them using OpenCV library and classify the topic video relates to.

- System Features & Objectives

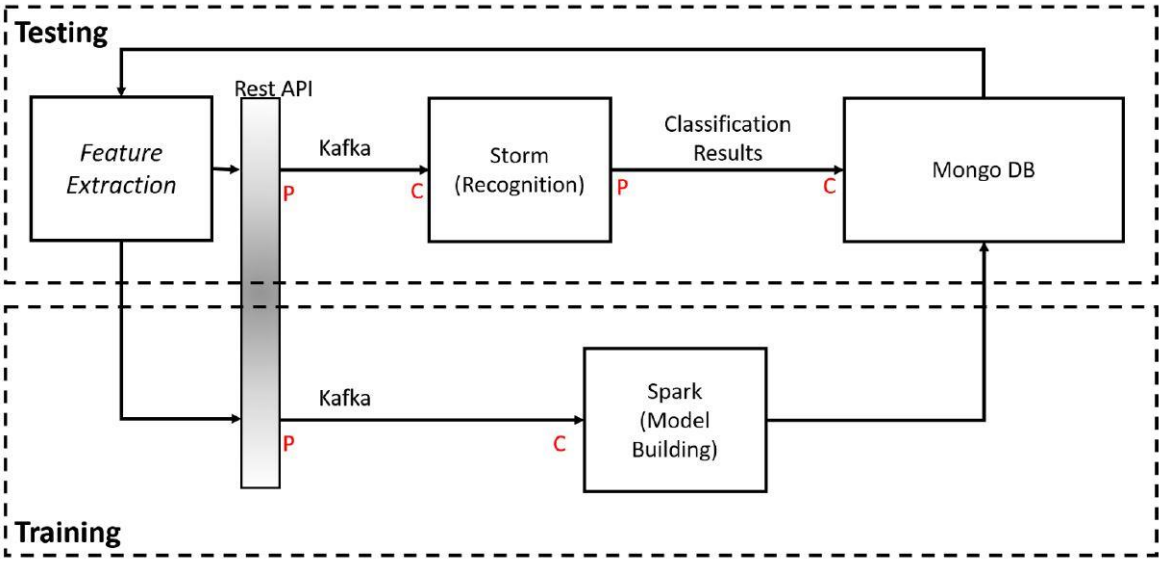
  Details of it includes: -
  - extract feature corresponding to a given metadata
  - tracking these features across video frames
  - learning from these features (clustering, classification, ...)
  -tagging the learnt faces.

- 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
- Current Implementation

Our current implementation is to use SIFT features to determine important key points For an object and identify those points within the video for each frame. Training on more images and aggregating those features through image analysis could improve our results.

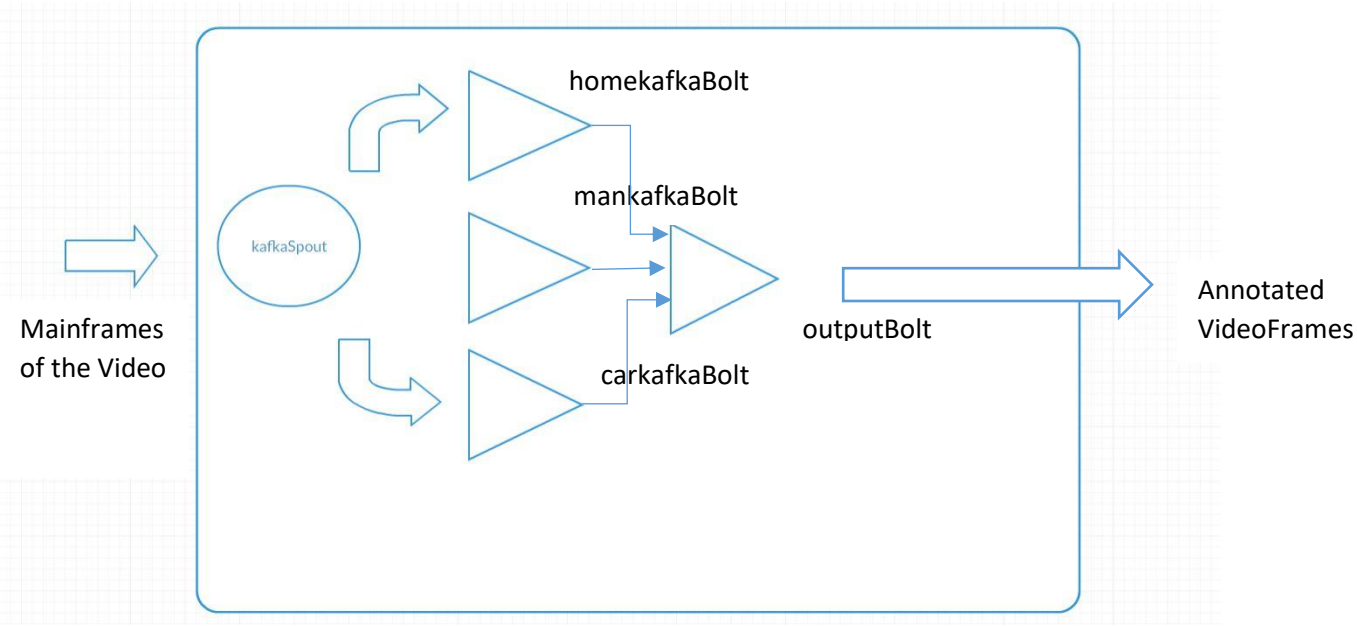
✓ better results than just training and trying to detect one image. *Scene Categorization* is another important part of my use case which is a term that is usually used to refer to the problem of recognizing the semantic label (e.g. travel, home, transportation etc.) of a single image. This project designs a technique to identify the categories of a scene.

- Workflow III : Full Workflow (as taken from lecture):

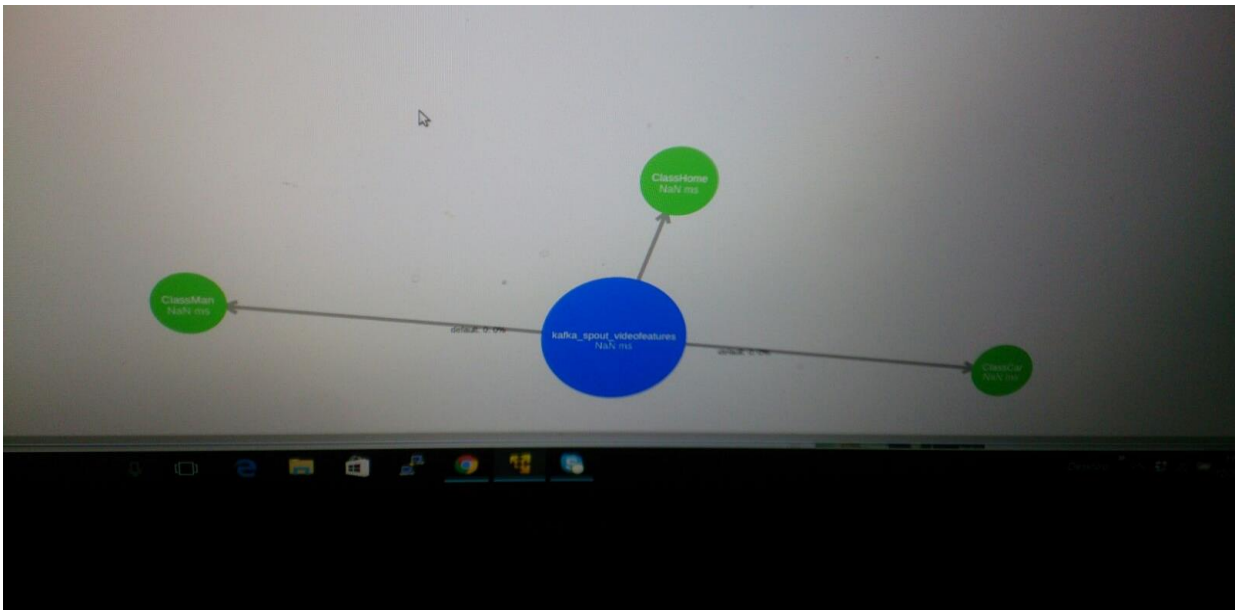


- Training: Data is extracted and sent to Spark (via Rest API). Spark builds a model and sends the corresponding If-Else decision tree to MongoDB
- Testing: Data is extracted and sent to Storm (via Rest API). Storm classifies cells within video, draws a box around classified cells and labels based on the chosen classification. The annotated video is stored in MongoDB, and can be retrieved by the client.

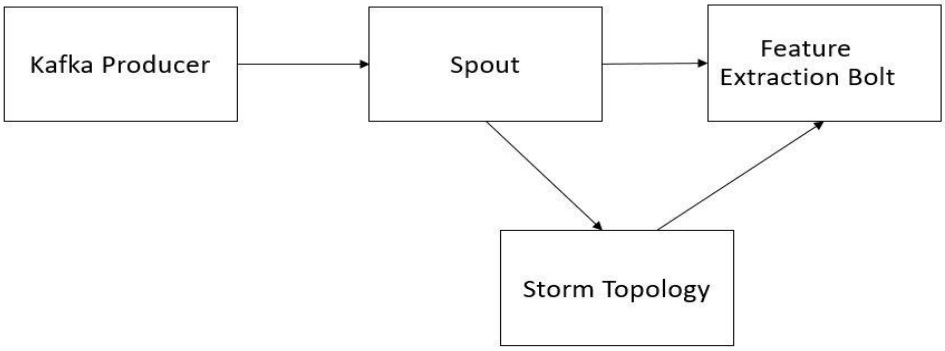
Storm Topology :



- KafkaSpout: Consumes streaming data from client
- homekafkaBolt: Bolt that determines (based on generated If-Else decision tree) if incoming feature data is a HOUSE. If so, it identifies the house and labels it.
- mankafkaBolt: Bolt that determines (based on generated If-Else decision tree) if incoming feature data is a MAN. If so, it identifies the man and labels it.
- carkafkaBolt: Bolt that determines (based on generated If-Else decision tree) if incoming feature data is a CAR. If so, it identifies the car and labels it.
- OutputBolt: Combines data in the form of an annotated video and outputs video to MongoDB

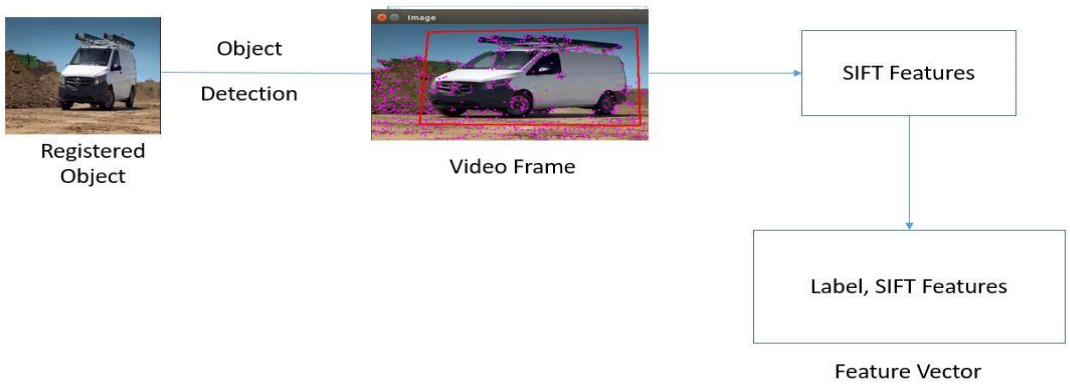


Workflow II : Kafka Storm Workflow



- ***KafkaSpout*:** takes in encoded video data from client and decodes it
- ***PreProcessing Bolt:*** Prepares video to be analyzed to key frame extraction
- bolt by dividing video into separate frames.
- ***KeyFrameExtraction Bolt***: Runs Main Frame Detection program and extracts the key frames to analyze that will be processed for feature detection.
- ***VideoProcessing Bolt:*** Processes video to make detection more accurate
- (e.g., changes contrast of given video).
- ***FeatureExtraction Bolt:*** Detects and extracts SIFT feature vectors that are sent to Spark. It also receives input from Spark to improve detection. Vectors are also sent to Annotation bolt so the frame can be annotated properly.
- ***Annotation Bolt:*** Annotates frame by labelling specific objects based on given SIFT feature vectors
- ***RebuildVideo Bolt:*** Rebuilds new video with annotated frames

- ## Workflow I : Training Data Feature Extraction



- Analytic Tools
  - **OpenIMAJ** is a library that allows the user to extract keypoints and frames through SIFT point detection and tagging.
  - **Spark MLLib**- the Spark MLLib is a rich resource of machine learning, allowing us easy access to decision trees, random forests, feature analysis, and neural network schemes.
  - **Storm**- Storm allows us to perform task parallelism so that when we stream in videos as frames, we are able to perform many tasks on those frames in order to determine where cells are and what state they are in.
  - Kafka - It's a streaming platform that lets you publish and subscribe to streams of records. In this respect it is similar to a message queue or enterprise messaging system. It lets us store streams of records in a fault-tolerant way.

- Algorithms
  - SIFT aggregation schemes: Training on one image and using it as representative of an entire data.
  - Random Forest/Decision Tree algorithms
  - Object detection, aggregation, and segregation using image analysis.

- Implementation
  - Local client application to extract key feature data
  - Send training data to SparkModelBuilder, which sends output of Classification Model in the form of an If-Else decision tree to MongoDB
  - Storm topology is generated based on model in MongoDB
  - Created topology is uploaded and run on storm server
  - Test data from client is sent to storm and is classified based on object type (i.e.car,home,man).
  - Output of classification is sent to MongoDB, which includes video that boxes cell type with label.
  - Video can be retrieved for the client from MongoDB via REST call.

- Documentation
  - Proposal: https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/blob/master/Project/Pre-Proposal.pdf

  - Lab3/4 (Clarifai for acquiring and applying image tags for frames; mainframe detection/ object detection/tracking): https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/wiki/Lab-3-4

  - Lab 5 (Image Classification) : https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/wiki/Lab-5

  - Lab 6 : (Development of Simple **Kafka - Storm** Application: https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/wiki/Lab-6

- Lab 7/8 : https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/wiki/Lab-7-8
- Lab 9/10 : https://github.com/PoojaShekhar/CS5543-Real-Time-Big-Data-Analytics--Lab-assignments/wiki/Lab-9&10

- Output
  - **Object Detection & Tracking with annotation:**

Video Annotation

Image Classification – Decision Tree Results

- CONFUSION MATRIX:

Predicted / Actual	home	man	car
Home	166.0	63.0	47.0
Man	281.0	1427.0	405.0
Car	84.0	140.0	364.0

- F-MEASURE: 0.6817523304581611
- RECALL: 0.6573731944910984
- PRECISION: 0.7384695474593782

Features posted to MongoDb

- **Scene Categorization**

- Related Work
  http://autoscout.adsc.illinois.edu/applications/autoscout/

- Projects done by others (include the URLs in Bibliography)
  - http://autoscout.adsc.illinois.edu/applications/autoscout/
  - http://sparkbigdata.com/tutorials/104-case-studies
  - https://books.google.com/books?id=TI3T9Yo7xkEC&pg=PA309&lpg=PA309&dq=video+analytics+motivation&source=bl&ots=aJMX5sqrQr&sig=iEty0gVZe4JRtM3oL00dWXQyyFc&hl=en&sa=X&ved=0ahUKEwiSi-S7qPnOAhVGbiYKHebwAMoQ6AEIQTAJ#v=onepage&q=video%20analytics%20motivation&f=false