Deep Video Analytics

Akshay Bhat, Cornell Tech, Cornell University.

Gains in accuracy have enabled end user applications

A good overview of computer vision research by Tomasz Malisiewicz

http://www.computervisionblog.com/2015/01/from-feature-descriptors-to-deep.html

Key takeaway

New libraries and programming frameworks are important in driving the research forward.

Libraries & Datasets

- OpenCV
- ROS
- Caffe (model zoo!), Theano
- Torch
- Tensor Flow

- Caltech 101
- Imagenet
- COCO
- Too many to keep track!
  - Youtube 8M, Open Images
  - Soundnet
  - Mapnet
  - CMU Video patch dataset

Question

What comes after well tested libraries, large datasets and competitions?

Answer

We need a platform which seamlessly combines Data + Models + User Interface.

Some attempts at building a platform CloudCV & NVidia DIGITS

- **CloudCV**
  - CloudCV: Large Scale Distributed Computer Vision as a Cloud Service
  - Generic system Intended for researchers, and non-researchers
  - Support for OpenCV, Graphlab, Caffe
  - Image Classification, VQA, Image stitching and several other algorithms
- **NVidia DIGITS**
  - ”DIGITS (the Deep Learning GPU Training System) is a webapp for training deep learning models. ”
  - Load/create datasets, train models, deploy models.
  - Aimed at researchers
  - Written in Python/Flask with Torch & Caffe supported

Relational data : Postgres, MYSQL, SQLite

::

Text, HTML : Lucene/Solr, Elasticsearch

::

Videos & Images : _____

Relational data : Postgres, MYSQL, SQLite

::

Text, HTML : Lucene/Solr, Elasticsearch

::

Videos & Images : ***Deep Video Analytics***

Relational data : SQL

::

Text, HTML : inverted word index, Page Rank

::

Videos & Images : ***Approximate Nearest Neighbor***

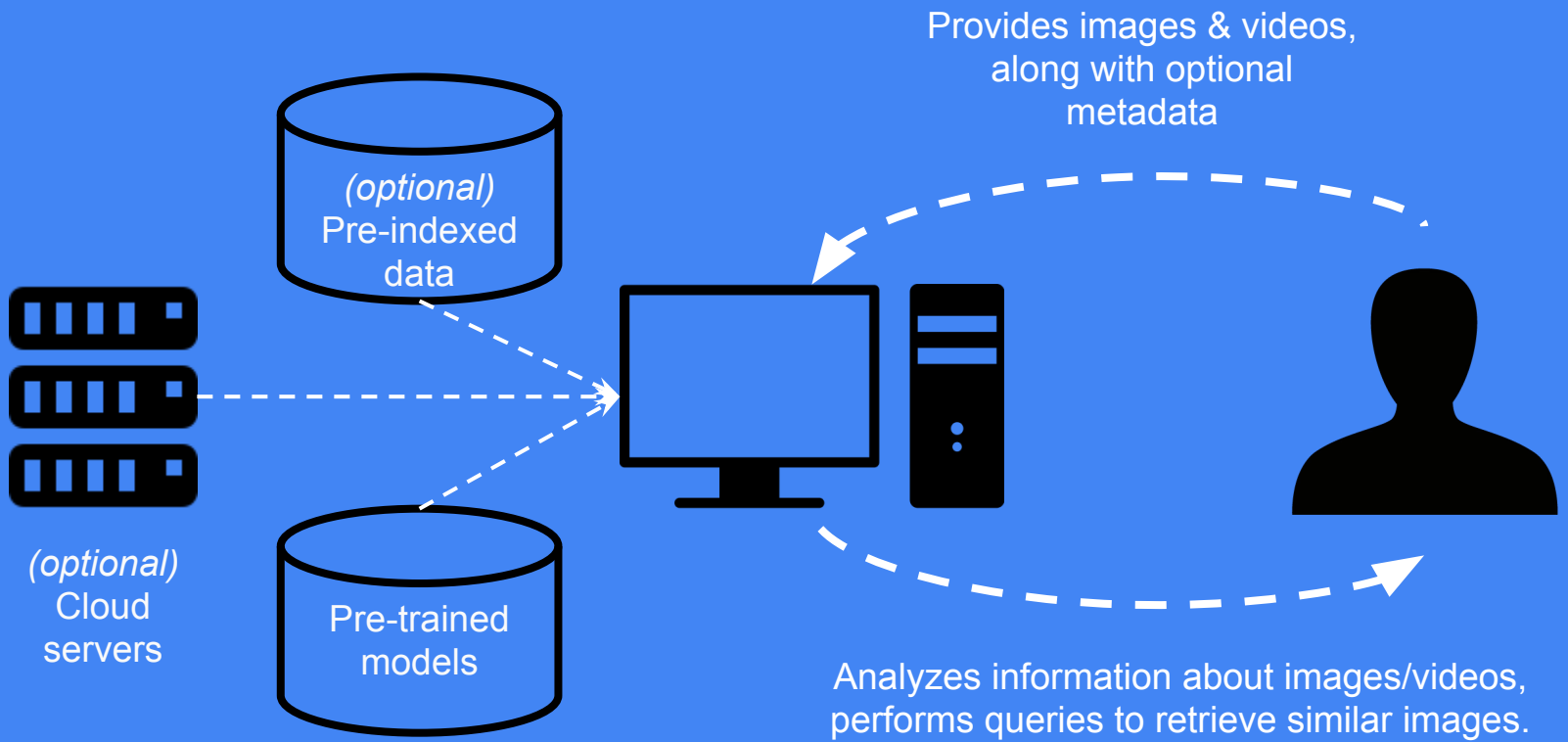
Question

Why not modify just lucene to index images as vectors?

Answer

Visual Search is significantly different compared to full text search. It requires a new user interface and ability to handle detections, segmentations, videos, etc.

High level overview of the system



Deep Video Analytics

Visual Search as a “Primary User Interface”

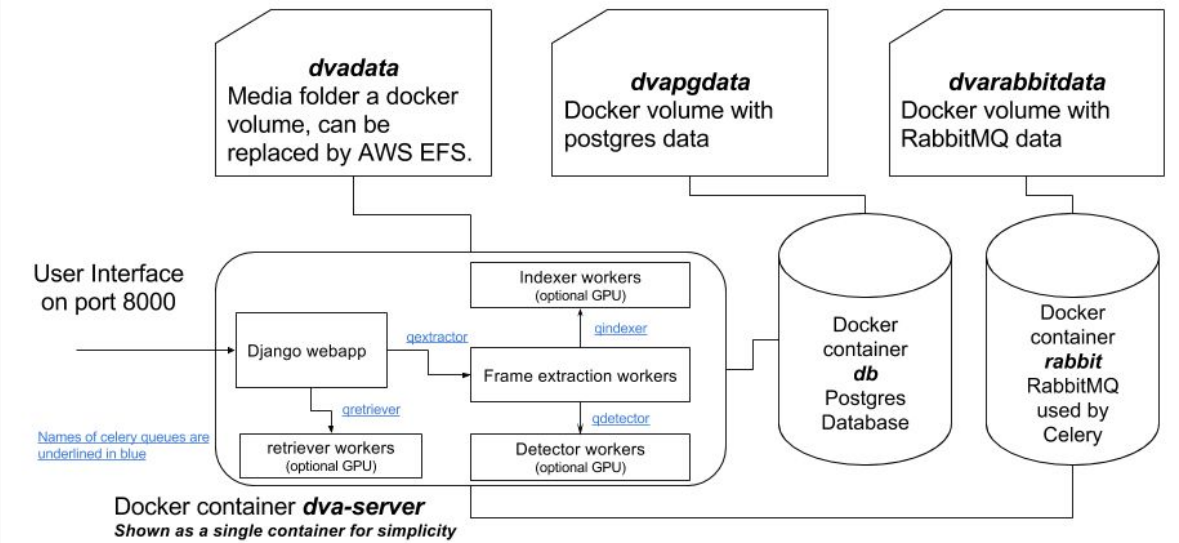
- Intended for **non-researchers**
- Make it easy for users provide data (uploads, youtube-dl, etc.)
- Batteries-included approach with an indexing and detection pipeline
  - Tensor Flow Inception v3
  - Single Shot Detector trained on VOC & YOLO 9000
  - Face detection / alignment / recognition
  - More algorithms such Text detection, Audio features planned.
- Pre-indexed datasets from different domains can be quickly loaded
- Can be easily customized by developers & researchers.

Technical requirements

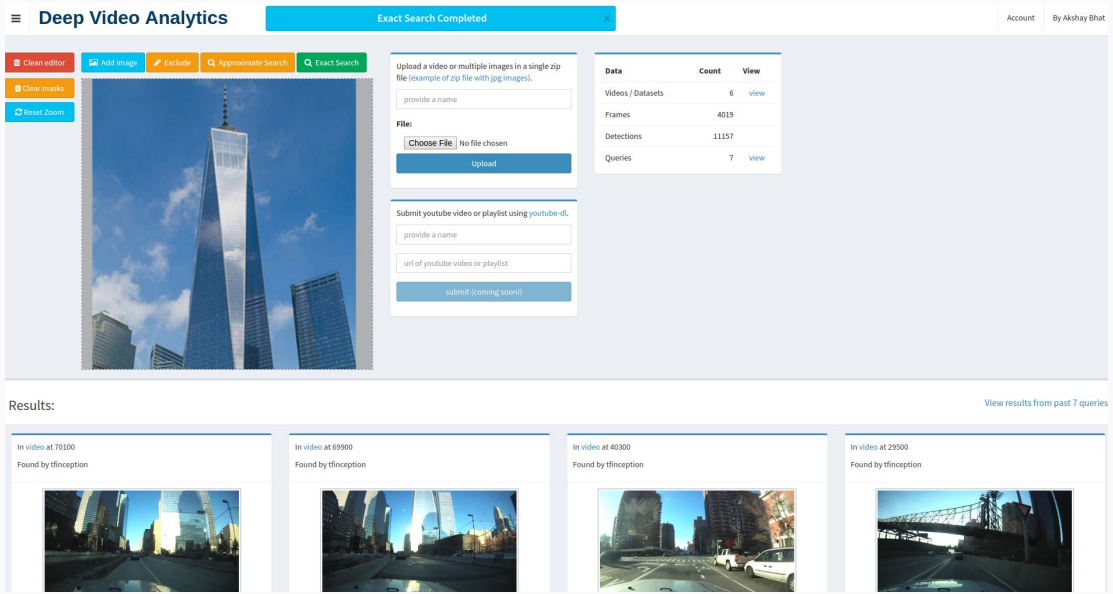
- Should work on machines with and without GPUs
- Should allow uploads and reindexing operations
- Easy to adapt by technical users
- Easy to dynamically scale out using cloud computing

Emulating datacenter on a machine
Docker, Docker-compose, Nvidia-docker

Docker enables same codebase across all configurations {a laptop, multi-GPU machine, datacenter} .

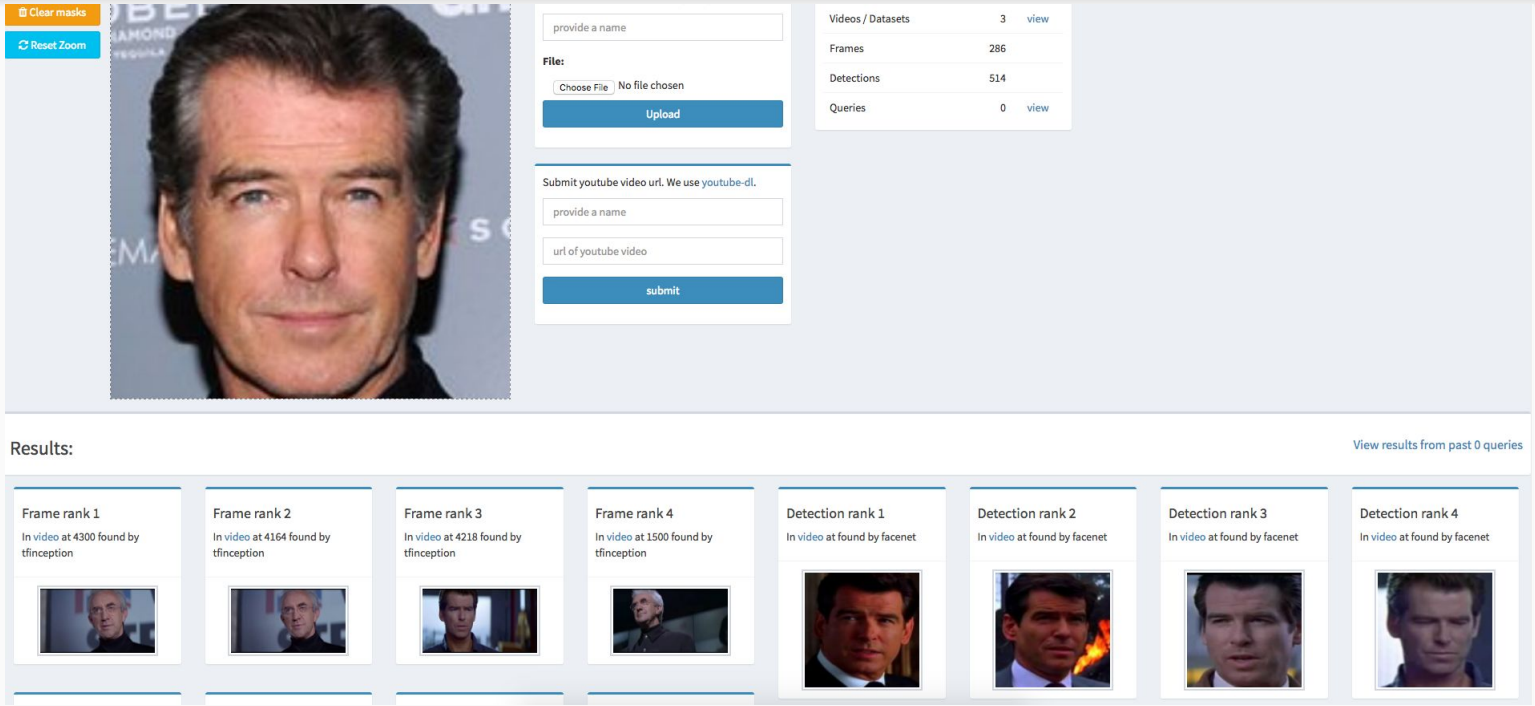


User Interface: Visual Search

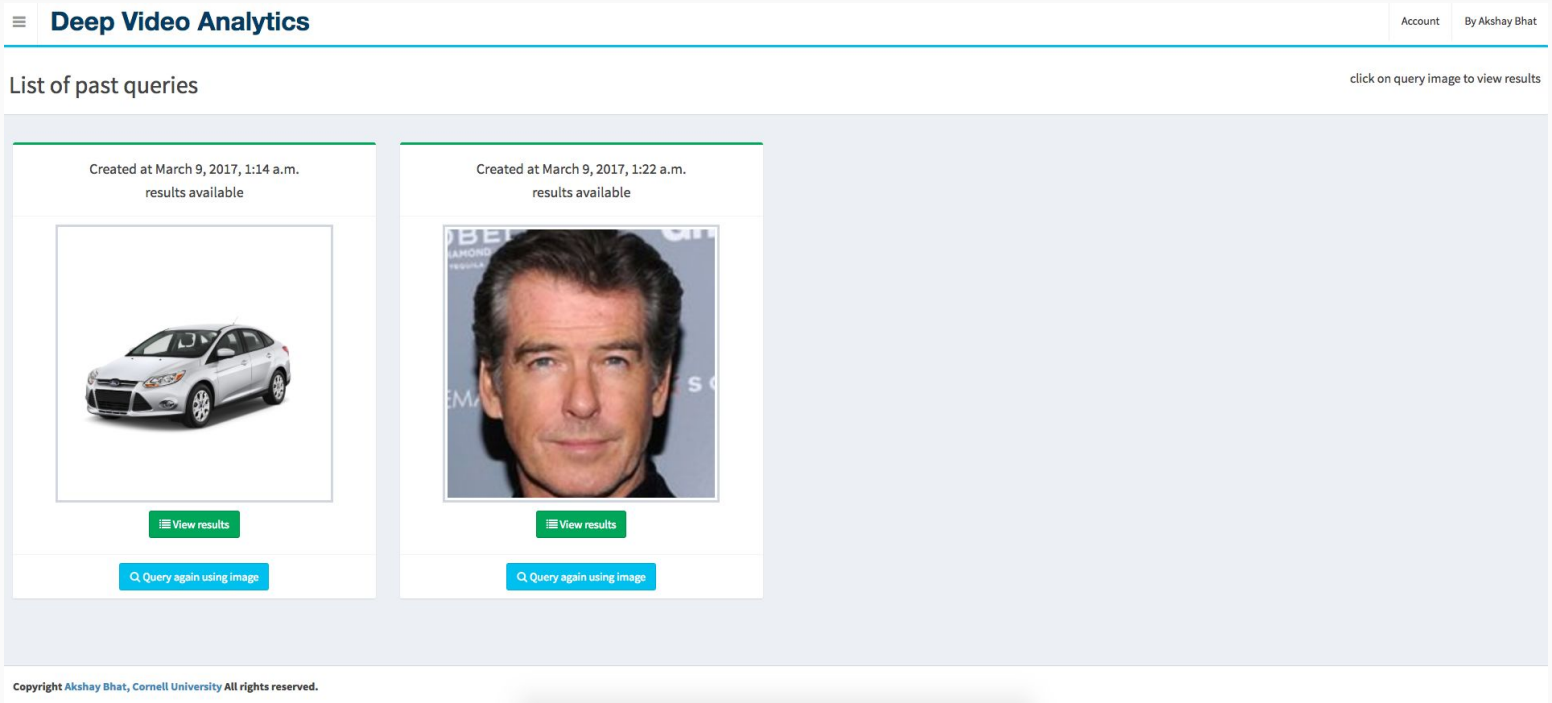


User Interface:

Search across frames + detections (faces, etc.)

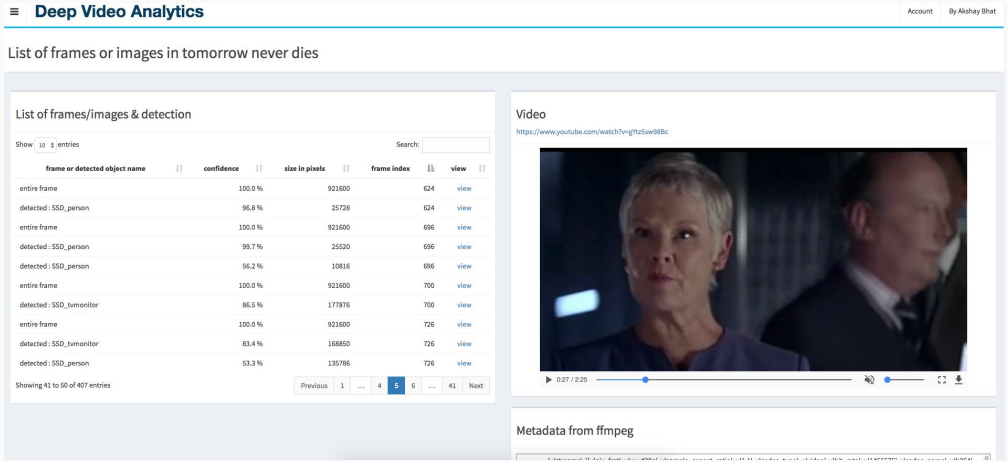


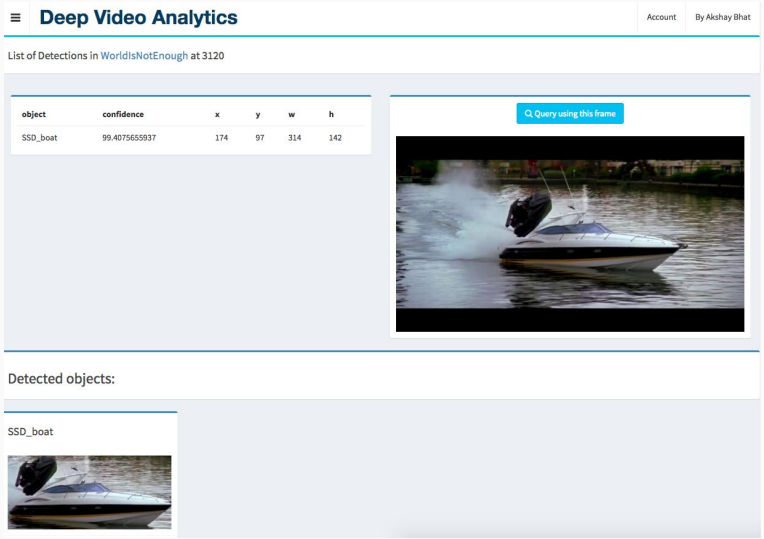
User Interface:
Browse previous queries



User Interface:

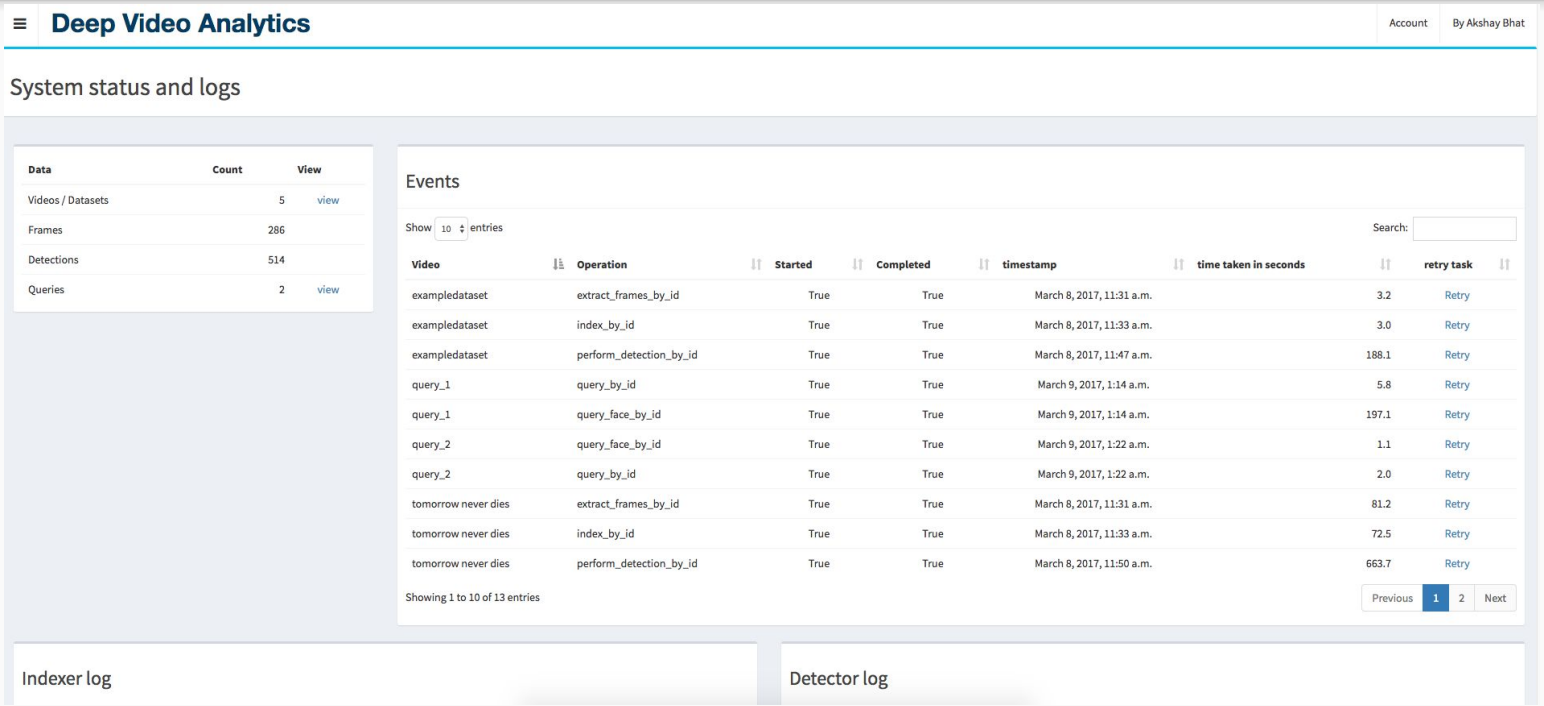
Browse videos, frames/images and detected objects





User Interface:

Get status of running and finished tasks & resubmit tasks.



Several open questions:
A work in progress

- How to balance fast/static vs slow/dynamic indexes?
- How to rank results using auxiliary information?
- How to incorporate text data extracted from images?
- Can we create a real time plug-in?
- Can the system continuously learn new categories?
- How do we incorporate external (pre & un) indexed data?

Thanks!

Contact me:

akshayubhat@gmail.com
www.akshaybhat.com