Deep Learning Pipelines for Big Images: How to Revolutionize an Industry

SFL Scientific

Michael Segala, Co-Founder & CEO

info@sflscientific.com | www.sflscientific.com

Background & Overview

- **SFL Scientific:** Data science consulting firm & professional services, bespoke solutions in data engineering, big data, and AI.

- **Team:** US-based in NYC and Boston, PhDs in CS, Engineering, Physics.

- **Talk:** 3 different disruptive technology use-cases: insurance, medical, and fashion/retail.

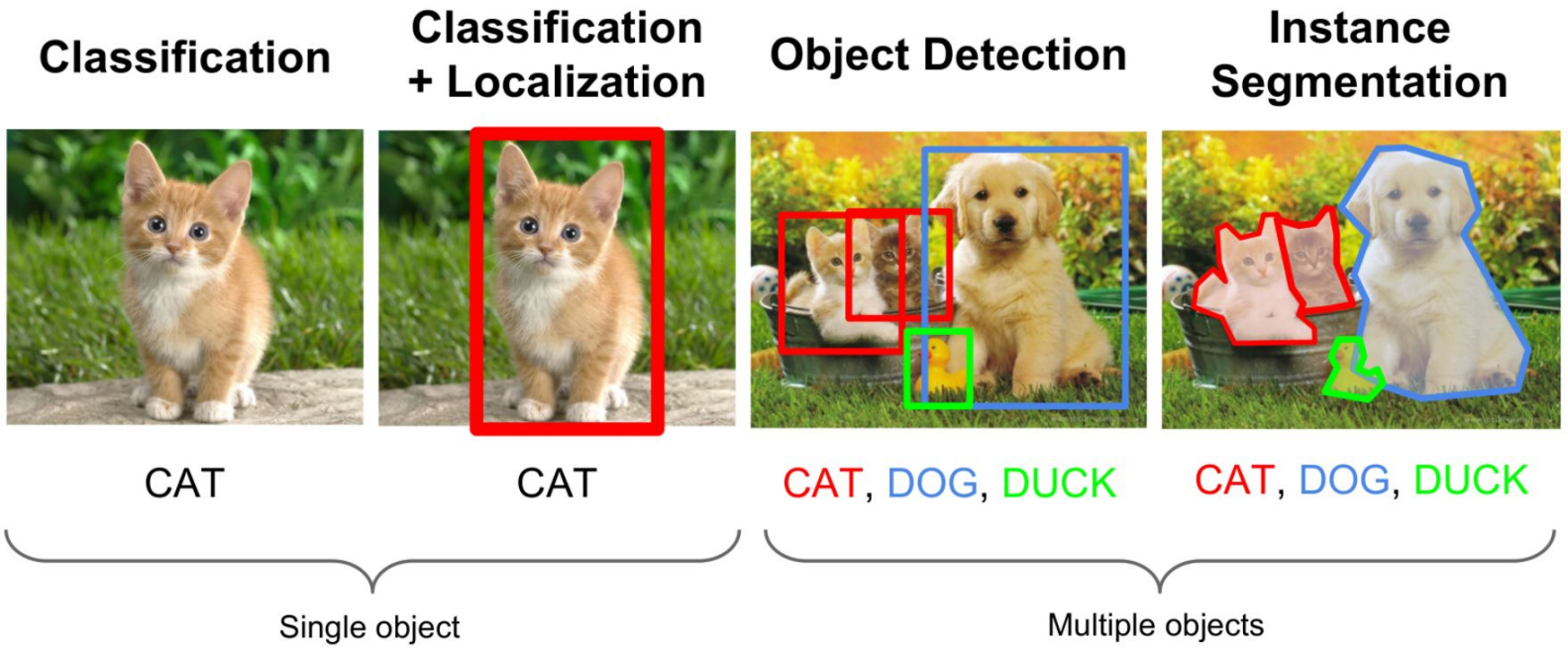
Understanding the problem: *quantitative big imaging*

- **Images:** Facebook, Instagram, etc...
  - **~1.46 MegaPixels** (roughly less than HD resolution).
  - 4K Ultra HD: 8.3 MP; 8K Ultra HD: 33.2 MP.
  - IMAX: ~100 MP (highest-resolution single-sensor digital cinema).

- **Big Images:** Digital Histopathology, Satellite, etc...
  - **~2.4 GigaPixels = 1646.1×** larger than Instagram!
  - **~289.2×** larger than 4K Ultra HD; **~72.3×** larger than 8K Ultra HD.
  - **~24.1×** larger than IMAX.
  - All of Imagenet dataset can fit into one image

- **Take Home Message:** *Big Images* are an **entirely different** data object for CV4DL:
  - **Hardware Resources:** file sizes/storage needs, RAM, CPU, GPU/VRAM, etc...
  - **Computer Vision:** image annotation (bounding boxes, polygons), denoising, superpixels, etc...
  - **Deep Learning:** data ingestion/batch, normalization, augmentation, etc...

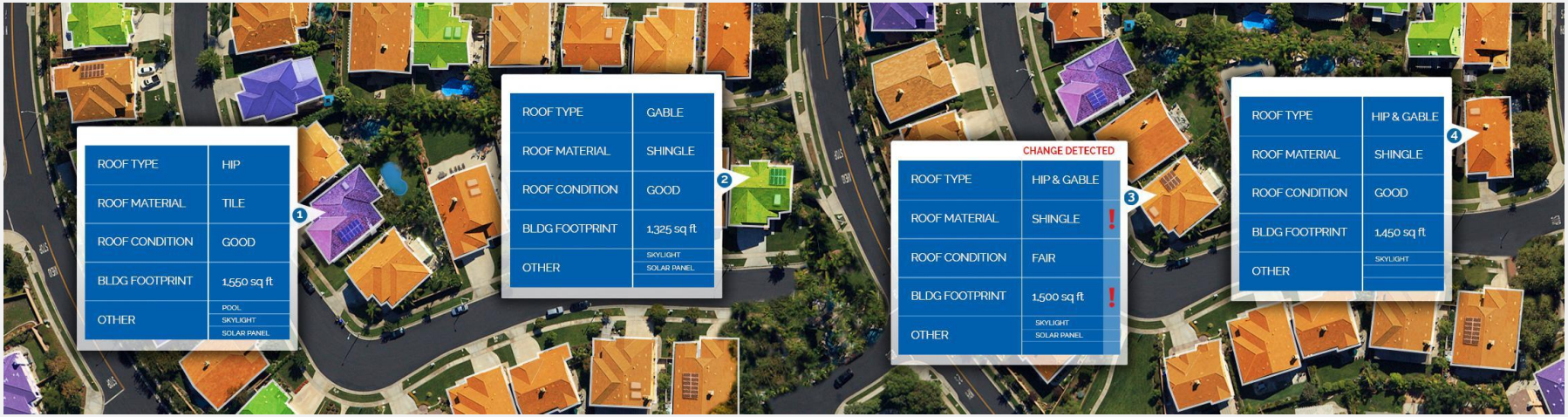
Understanding the problem: *deep learning*



Understanding the problem: *big satellite/aerial imaging*

Insurance Project Overview:

Full automation of the detection and classification of desirable feature in big images, e.g. roofs, vegetative overhang (none/low/high), solar panels (yes/no), etc. for >80% of US homes




DigitalGlobe
© Satellite Image ©2017 DigitalGlobe

Client Context

- **Core Business:**
  - Insurance company with access to big satellite imagery covering over ~80% of the habitable US.
  - Underwriting services: evaluate property risk.
- **Technical Challenges:**
  - Volume and variability of necessary data for classification and segmentation;
  - Combination of open and private datasets;
  - Process automation in their insurance pipeline.

Client Past History

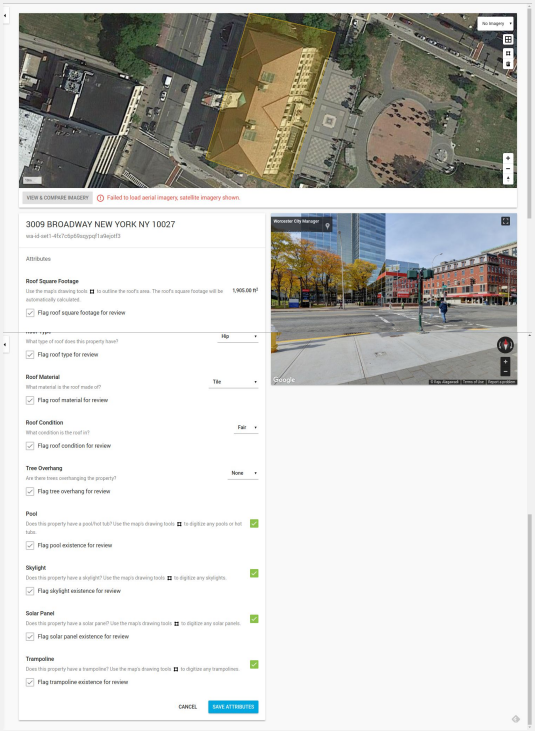
- **Manual Labeling:** Annotation and quality control interface & pipeline
  - 30+ SMEs working on randomized image batches:
  - Each image evaluated at least by 2 SMEs;
  - Disambiguation performed by a third independent SME.
- **Classic Computer Vision Techniques (openCV):**
  - Contour Detection;
  - Texture Analysis;
  - Segmentation, etc.
- High false positives rate for classical methods (lighting, seasons, etc). Needed to move to GPU accelerated Deep Learning.

Client: Quantitative Big Image Solution

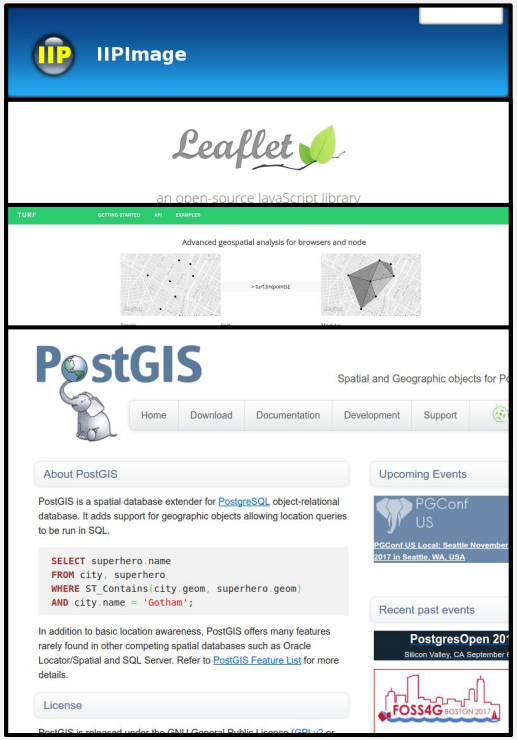
- **Hardware Provision & Deployment:** Storage & computational servers (>1 PB of data)
  - Training done on 4 GPUs
- **Open & Private Datasets:**
  - Public: DPEAG 2016, Vegetative Overhang, Solar Panels;
  - Private: Satellite & Drone
- **Annotation & Ingestion Pipeline:** Nginx, IIPImage server, Leaflet, Turf, PostGIS
- **YOLO9000:**
  - *de novo* optimized implementation: Tensorflow + NVIDIA GPU, data augmentation, hyperparm sweep;
  - Performance: Jaccard Index (Intersection-over-Union) $\gtrsim 0.6$.
- **Timeline:** 6 months (first phase)

Quantitative Big Image Solution

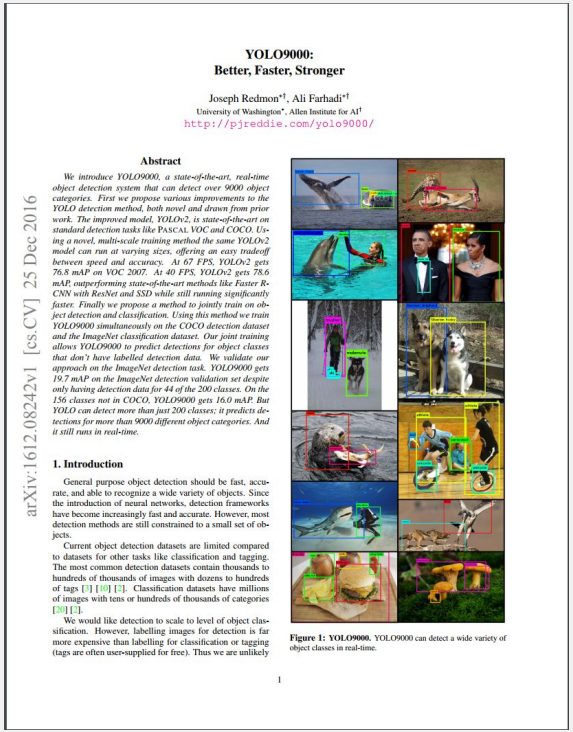
Annotation UI/UX



Ingestion Backend



YOLO9000



YOLO9000:
Better, Faster, Stronger

Joseph Redmon*†, Ali Farhadi*†
University of Washington*, Allen Institute for AI†
http://pjreddie.com/yolo9000/

Abstract

We introduce YOLO9000, a state-of-the-art, real-time object detection system that can detect over 9000 object categories. First we propose various improvements to the YOLO detection method, both novel and drawn from prior work. The improved model, YOLOv2, is state-of-the-art on standard detection tasks like PASCAL VOC and COCO. Using a novel, multi-scale training method the same YOLOv2 model can run at varying sizes, offering an easy tradeoff between speed and accuracy. At 67 FPS, YOLOv2 gets 76.8 mAP on VOC 2007. At 40 FPS, YOLOv2 gets 78.6 mAP, outperforming state-of-the-art methods like Faster R-CNN with ResNet and SSD while still running significantly faster. Finally we propose a method to jointly train on object detection and classification. Using this method we train YOLO9000 simultaneously on the COCO detection dataset and the ImageNet classification dataset. Our joint training allows YOLO9000 to predict detections for object classes that don't have labelled detection data. We validate our approach on the ImageNet detection task. YOLO9000 gets 19.7 mAP on the ImageNet detection validation set despite only having detection data for 44 of the 200 classes. On the 156 classes not in COCO, YOLO9000 gets 16.0 mAP. But YOLO can detect more than just 200 classes; it predicts detections for more than 9000 different object categories. And it still runs in real-time.

1. Introduction

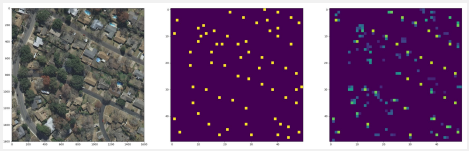
General purpose object detection should be fast, accurate, and able to recognize a wide variety of objects. Since the introduction of neural networks, detection frameworks have become increasingly fast and accurate. However, most detection methods are still constrained to a small set of objects.

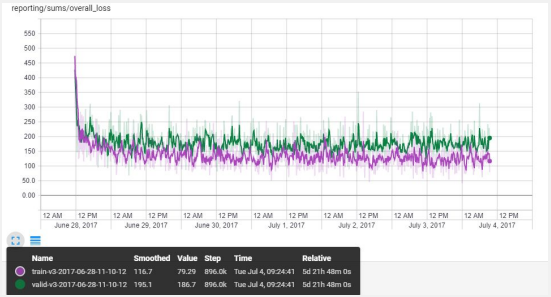
Current object detection datasets are limited compared to datasets for other tasks like classification and tagging. The most common detection datasets contain thousands to hundreds of thousands of images with dozens to hundreds of tags [3] [10] [2]. Classification datasets have millions of images with tens or hundreds of thousands of categories [20] [3].

We would like detection to scale to level of object classification. However, labelling images for detection is far more expensive than labelling for classification or tagging (tags are often user-supplied for free). Thus we are unlikely

Figure 1: YOLO9000. YOLO9000 can detect a wide variety of object classes in real-time.

Client: Results





Business Returns

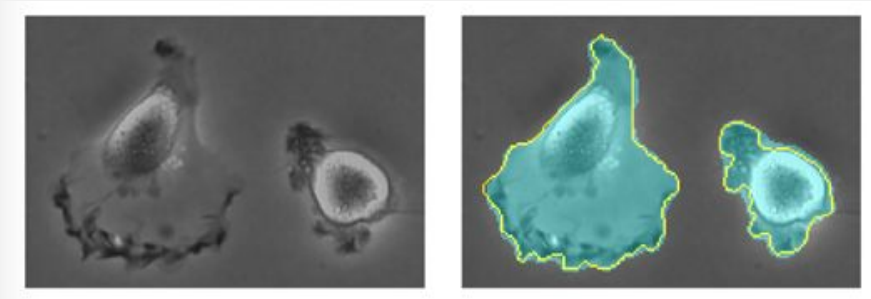
- Reduce the number of adjusters in the field, expand coverage & speed. Infinitely scalable process.
- Eliminate Manual Labeling: Annotation and quality control interface & pipeline.
- Create automated accurate historical record of the customer.
- Establish themselves as market leader with real competitive advantage.
- Paid for itself in less than 6 months

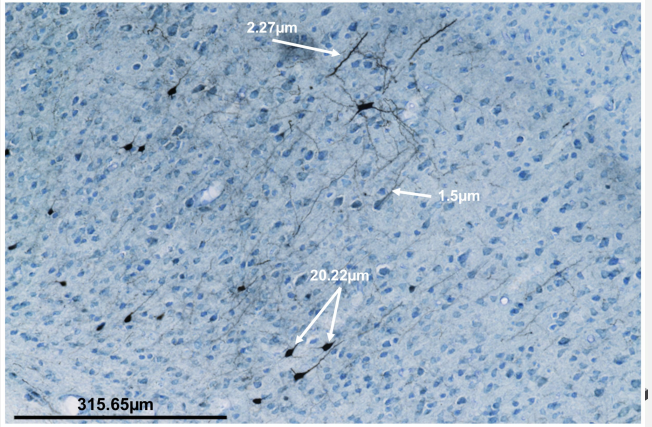
Medical Imaging Project Objective:

Full automation of detection and classification of desirable features in big images, *e.g.*, **cancer** (patch color, moles, mass), **tissue structure** (cell density/organization), **robustness** (minimize prediction errors), **etc...**

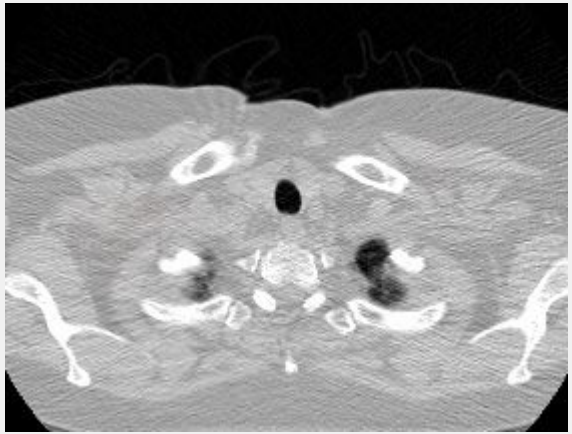
Use Case: Pixel Level Patterns & Shapes

- **U-Net & SegNet base:** Segmentation & masks for desired objects ~98%+ common
- **Dataset Challenges:**
  - **Multiple Markers:** Heterogeneous & complex cell shapes, types, boundaries, tubules, structures, overlap, artifacts
  - **Data Size:** Imaging systems produce 1GB/minute raw data, common to have 250MB-4GB compressed files
  - **Structure:** Highly unstructured and variable input [preparation, staining protocol, focus, debris, illumination/light source, tissue type, species, etc.]





Use Case: Disease Detection & Diagnostics

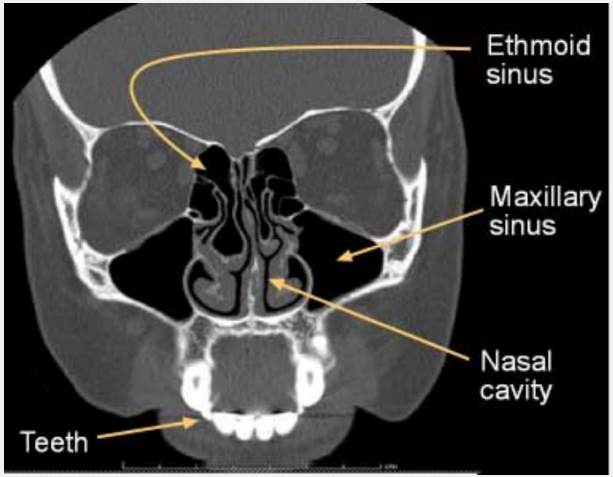


Tech Stack:

Languages: Python 3
Database: Redis
Server Side: EC2/NVIDIA GPUs

Frameworks: keras, tensorflow, numpy, scipy, pandas, scikit-learn, ITK
Input: DICOM, 20k+ scans

Use Case: Disease Detection in 3D CTs & MRIs



Goal: Develop system to classify and predict 40+ separate disease states from input scan, head CT. Assess patient's disease and severity, risk profile, for surgery and other interventions.

Data: Low volume, raw CT data. Annotate via radiologists for disease states.

Challenges: Annotation cost, binary classification, low-volume of bounded scans (time, cost), multiple features per zone, class imbalances.

Augmentation: Lossy and lossless sets, transformations, contrast, SNR, up to 200x of ~10k+ DICOM

- Initiative will require between 50-75k annotated patient cases for viability.

Tech Stack:

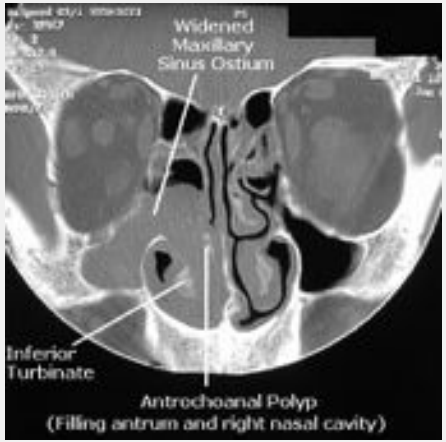
Languages: Python 3
Server Side: EC2/3/NVIDIA GPUs

Frameworks: keras, tensorflow, numpy, scipy, pandas, scikit-learn, ITK

Development & Workflow

Annotation: Complex pipeline to identify dozens of feature - large and small

- Manual review of scans, 0.4-1 mm resolution (CT), ~400x400x300 pixel
- Progression: Binary classification, zone segmentation, boxed/labels
- Rich labeling requires increased time and cost



Development: Complete data science lifecycle for heterogeneous
Low volume features and unique cases (>500)

- Annotation errors & independent review/QA - 3-5% uncertainty
- Create single model per feature (3D U-net/FCN)
- High-capacity network required, training time 1-4 weeks even for simple tuning
- Timeline: 4-6 month pilot, 9-12 months production.

Use Case: Disease Detection in 3D CTs & MRIs

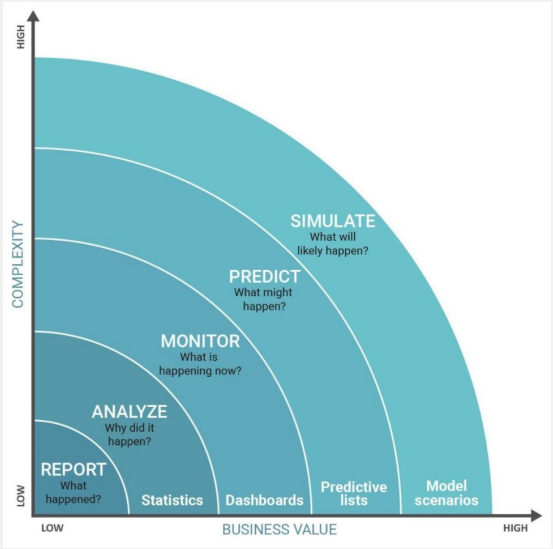
Progress: Over 20 3D models built for our solutions that will be ensembled.

- Pipeline: Data denoising, data augmentation, data segmentation “ROI/VOI”
- Learning: Curriculum learning, VGG16 3D models as baseline.
- Future: Deeper annotations; annotation cost or development cost?

Short Term Impact: Cognitive workflow solutions, diagnostic support, that enhance productivity.

Future: Incorporate clinical knowledge into models.

- Monitor -> Predict -> Simulate.
- Recommend follow-up CT periods.
- Diagnose if the patient will develop/worsen condition.
- Simulate and infer risk via longitudinal tracking.



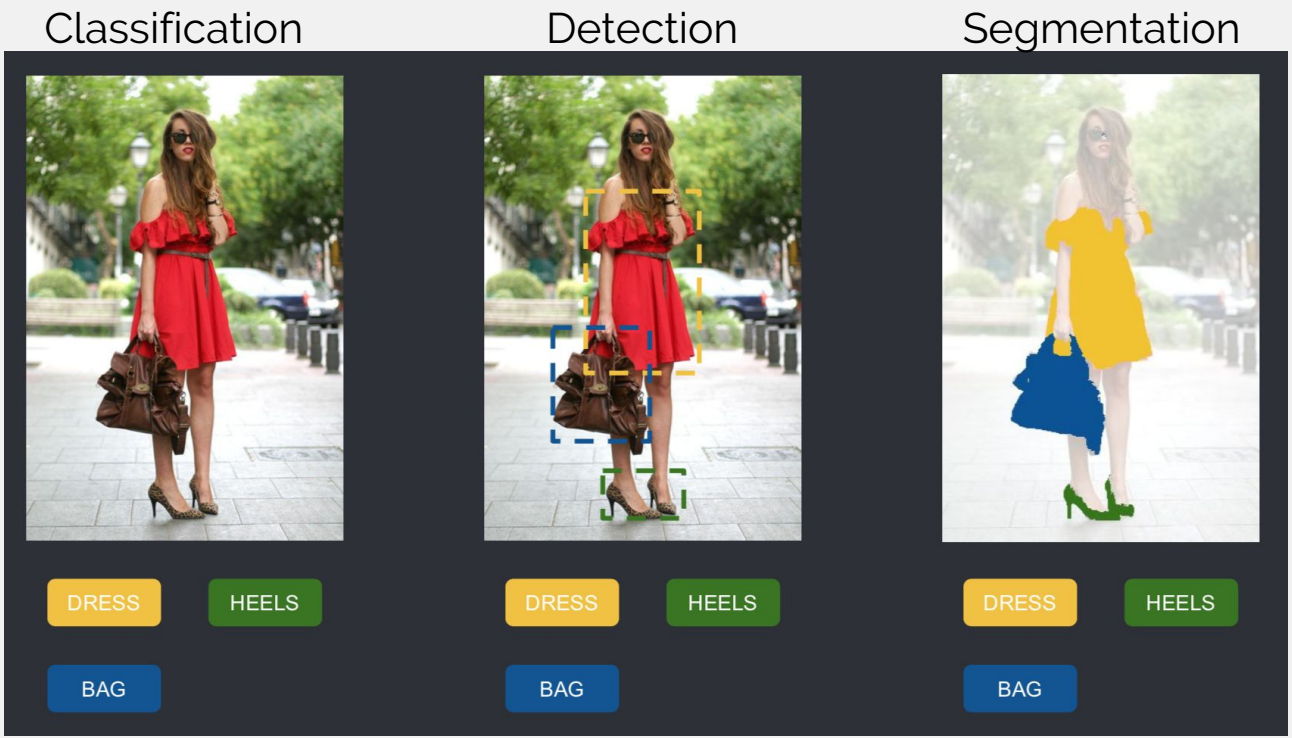
Business Returns

- **Long-Term:** Generate end-to-end processing pipelines coupled with EMR
  - Combine EMR and patient history / outcomes with automated diagnosis from medical images - validate historical diagnosis and treatment outcomes
  - Reduce variability across provider and clinical systems in diagnosis/outcomes
- **Short-Term:** Reduced bias & manual process in R&D, provide decision tools
  - Diagnostic time reduced from hours per sample to a semi-automated < 3 min. 100× gain! Compute time is ~400x cheaper than human.
  - 1,000+ images generated daily - enable trials, testing, and accurate decisions

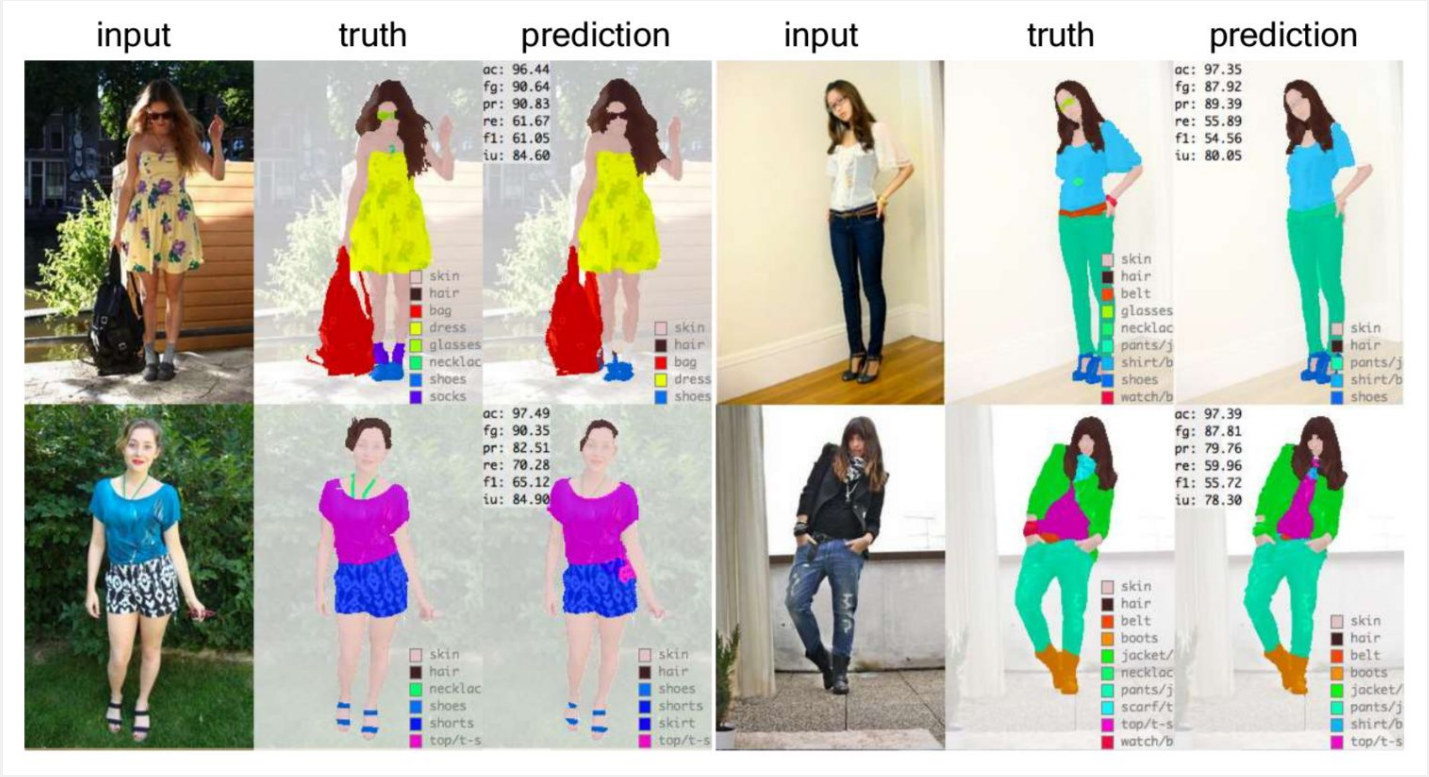
Retail Project Objective:

Full automation of Retail and Fashion images. From product labels to chairs to handbags....

Fashion Object Detection



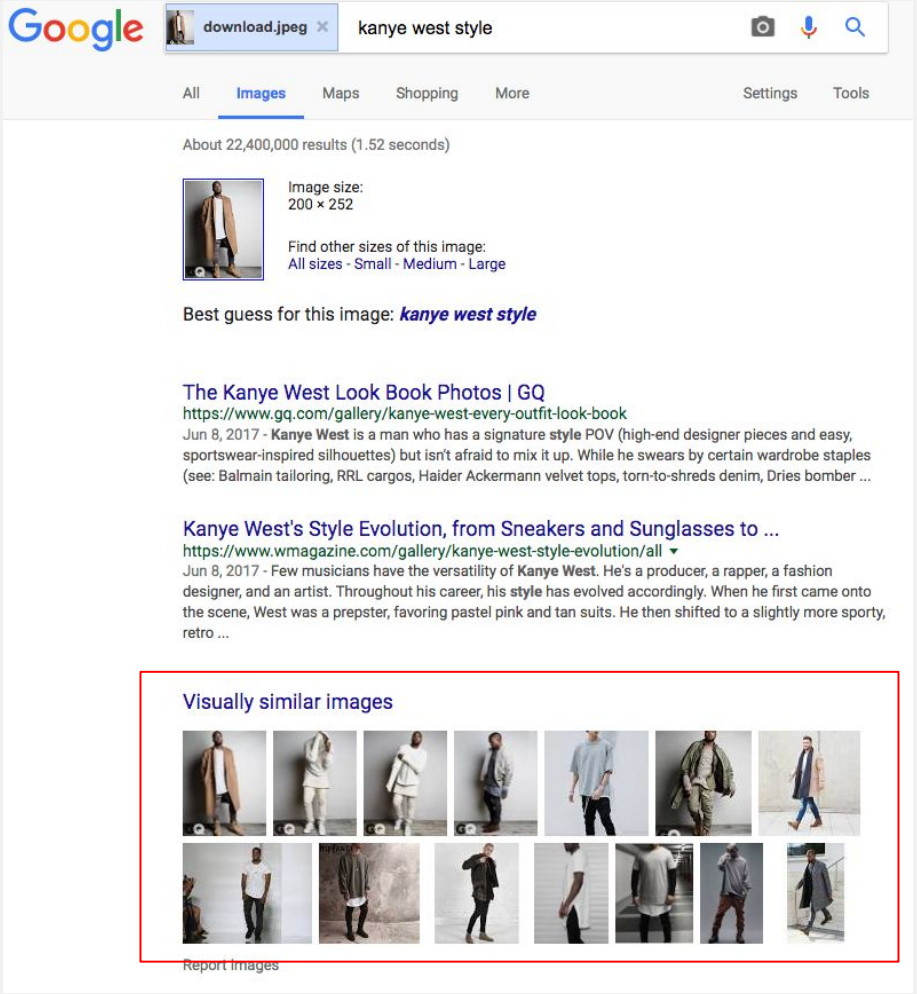
Sample Algorithmic Output



Use Cases & Goals

- **Goal:** Detect and segment clothing & accessories - dresses, bags, shoes, pants, jeans, jackets, etc.
- **Create:** End to end pipeline to catalog and serve information (street to shop, visual marketing, mine best selling clothing features, etc.)
    - **Object Detection Engine:** Deep learning, YOLO, etc.
    - **Visual Search Engine:** Word attributes do not fully capture images, poor results
    - **Semantic-Linked Search Engine:** Focus on specific domains, predict tags, NLP engine to extract information and expand taxonomy and attributes

Current Image Search Performance:



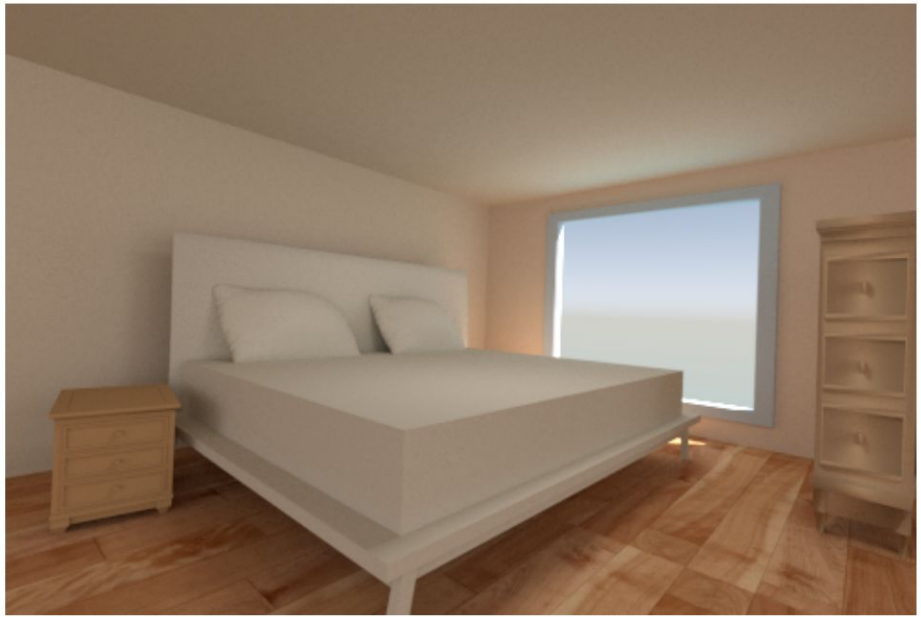
Progressive Image Search Performance

Natural Language Processing

Learning semantic relationships between clothing and context attributes and concepts

3D modeling of 2D photos

Goal: Take a single photo and turn it into a 3D CAD file



3D modeling of 2D photos