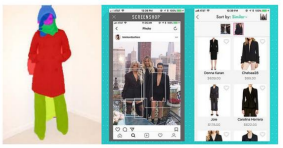
CS230: Content-Based Image Retrieval System (CBIR) for eCommerce Using Deep Neural Networks

Authors:
Nicholas Sinthunont,
Lee Reed



The Problem

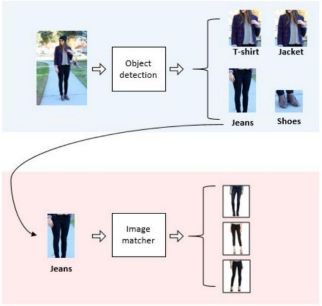
Online retailers are *failing* to design products which cater to the behaviour of online shoppers in two ways:

- **Search Limitations -** eCommerce platforms typically allow for text based inputs even though consumers typically rely on images for fashion inspiration
- **"Not what I am looking for" syndrome -** existing retail platforms require users to scroll through dozens of pages of products which may or may not be something they are interested in

Our project explores the following to solve the above problems:

- Is it possible to detect different fashion objects in a given image?
- Is it possible to classify these objects into different classes?
- Is it possible to find similar items to those objects within their respective classes?

Our Approach / Model



Object Detection

- Utilized existing Masked R-CNN implementation utilized for detection of "objects in the wild" of multiple classes
- Avoids spurious edges of FCNs & retains full image
- Object mask provided as an output to reduce noise & feed image matcher exact image

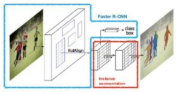
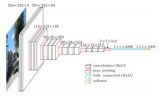


Image Matcher

- Input image fed into VGG16 model, with weights pre-trained on ImageNet, to extract image features
- Feature vectors fed into a ANN (Approximate Nearest Neighbors) implementation to identify top 5 images



Data Inputs

Data Sources:

1) **eCommerce Sites -** Scraping of retail websites such as Nordstroms across various classes including jeans, t-shirts, Shorts, dresses,etc.
2) **Google Open Images v4** - Dataset with over 15 million bounding boxes across 600 classes (clothing bounding boxes in the order of 10^6):

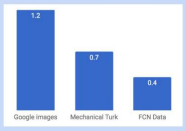

Results - Object Detection

Google Open Image

The result was that the Mask R-CNN was able to detect bounding boxes to a sufficient degree of accuracy when related to the "footwear" class but had high errors for others

This was assumed to be the case because "footwear" typically appears in rectangle boxes but something like a t-shirt has a more abstract shape thus requires more data to process

Bottom-out error



Mechanical Turk

Single vs multi class

FCN Data Generations

Example of generating data

Potential issues with FCNs

Results - Image Matcher

Object Similarity Search - Dresses

Sample Similarity Matching

Looking Forward

As next step we are considering the following:

- Rigorous testing to validate and tune model against different fashion classes e.g. men vs. women jeans
- Further experimentation with similarity search algorithm to compare Annoy performance with triplet loss approach
- Incorporating unsupervised techniques to automatically determine classes for boundary boxes
- Removal of background noise from detection output

Key References:

1. K. He, G. Gkioxari, P., Dollar and R. Girshick. Mask R-CNN. arXiv:1703.06870v3 [cs.CV] 24 Jan 2018
2. D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. International Journa of Computer Vision, 2004
3. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014.
4. M. Mirza, S. Osindero. Conditional Generative Adversarial Nets. arXiv:1411.1784v1 [cs.LG] 6 Nov 2014.
5. Xiaodan Liang, Ke Gong, Xiaohui Shen, and Liang Lin, "Look into Person: Joint Body Parsing & Pose Estimation Network and A New Benchmark", T-PAMI 2018.
6. https://github.com/spotify/annoy

Note: See Project Report for full list of citations