InstaFashion: Clothing Detection and Classification with YOLO

Kevin Fry, Xianming Li, Vivian Yang
{kfry, xmli, vivianca} @ stanford.edu | CS 230 Deep Learning

Stanford | ENGINEERING
Computer Science

Abstract

Online shopping is an exponentially growing market, but tracking down that perfect shirt or pants you see someone wearing on the street is still really difficult. Our work, InstaFashion, allows users to identify clothes from just an image. We compared 2 architectures, YOLO v1 and VGG-16 as a baseline.

Dataset

Dataset: Images of people photographed in everyday settings with bounding boxes (t,l,w,h) around items of clothing
Sample data point: {"photo": 2281, "product": 7871, "bbox": {"width": 112, "top": 335, "height": 204, "left": 59}}
Specifications:

- 18k images
- 31k bounding boxes
- 11 classes: bags, belts, dresses, eyewear, footwear, hats, leggings, outerwear, pants, skirts, tops

Preprocessing

(1) Convert training dataset into 3D tensor of size 18397 x 15 x 6 [number of images x max number of bounding boxes per image x (id, class, x_c, y_c, w, h)]

(2) Resize photos into standardized size 448 x 448, filling in black margins vertically and horizontally, as needed

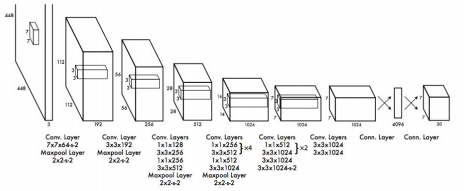
(3) Split dataset into approx 90-5-5: training (16439), validation (979), and test sets (979)

(4) For training the baseline we cropped images to bounding boxes and then resized to 224 x 224 before feeding them into the network, which was trained to minimize categorical cross-entropy loss.

Model

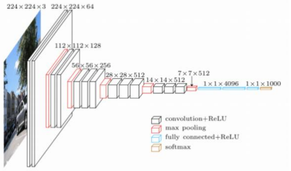
The YOLO Model

We implement YOLO v1 with a 7 x 7 grid and 5 bounding boxes per grid. We use pretrained weights trained on ImageNet for the first 20 layers of the model, and train the final four convolutional and two fully connected layers on our dataset to fine-tune our model. We trained two models following this architecture: one with a static learning rate (1e-5), and one with a dynamic learning rate that changed over the course of training.



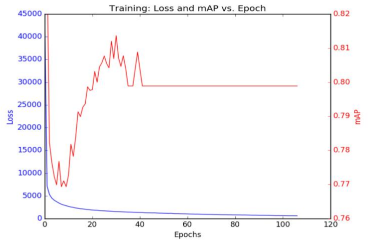
The Baseline Model

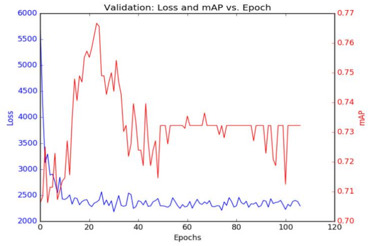
For the baseline we implemented a simple sliding window model that performs image classification on each window using VGG16 model. We take the feature extractions from the KERAS pretrained VGG16 model and then train two fully-connected layers to learn class probabilities on our dataset. This very crude model is not very good, but is decent baseline to compare our YOLO models against.



Results

mAP Results	Loss Results
Static LR: 0.719	Static LR: 5018.02041015625
Dynamic LR: 0.689	Dynamic LR: 7850.24677734375
Baseline: 0.559	





Discussion

Our best YOLO model achieves a test mAP of 0.720,outperforming the baseline sliding window model, which only achieves a mAP of 0.56. Further validating the correctness of our YOLO implementation, our test mAP is similar to what the original YOLO paper achieved on the PASCAL VOC dataset.

References

1. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. arXiv preprint arXiv:1506.02640, 2015.
2. M. Hadi Kiapour, Xufeng Han, Svetlana Lazebnik, Alexander C. Berg, and Tamara L. Berg. 2015. Where to Buy It: Matching Street Clothing Photos in Online Shops. In Proc. ICCV
3. Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In CVPR, 2016.
4. K. Simonyan, A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition