YOLOnet

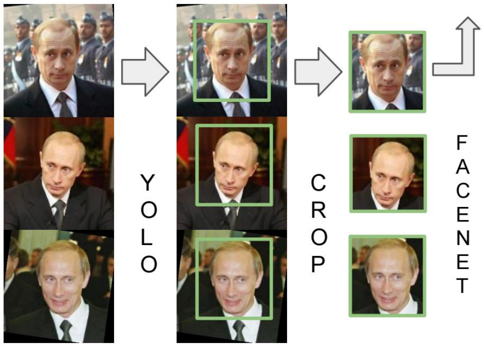
Will Lauer and Hannah DeBalsi
(wlauer, hdebalsi)@stanford.edu

Vladimir Putin

Idea

We applied two well-known and successful algorithms, YOLO and FaceNet, in an attempt to combine the speed of YOLO with the accuracy of FaceNet on facial recognition.

In our project, we relied on Allan Zelener's Yad2k and David Sandberg's FaceNet.



Data

We utilized WIDER Face[3] images and Labeled Faces in the Wild.[4]

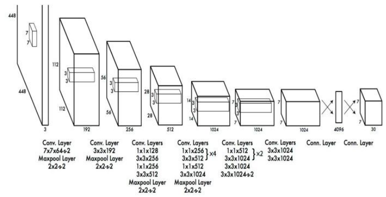
For each LFW image, we created 5 randomized croppings, with each dimension ⅔ that of the original. These were split into 50%/50% train/dev sets, since we were using a pretrained FaceNet model[2] and thus needed less new data.

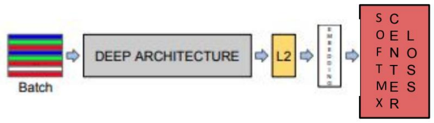
WIDER Face images are fed into a variant of YOLO with a single output class. These cropped images are sent to a FaceNet model pre-trained on celebrity image data. FaceNet then identifies the person selected by YOLO.

Data: WIDER Face

Data: Labeled Faces in the Wild

Architectures





Loss Functions

$$
\begin{aligned}
&\lambda_{\text{coord}} \sum_{i=0}^{S^2}\sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i-\hat{x}_i)^2+(y_i-\hat{y}_i)^2\right] \\
&+ \lambda_{\text{coord}} \sum_{i=0}^{S^2}\sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i}-\sqrt{\hat{w}_i}\right)^2+\left(\sqrt{h_i}-\sqrt{\hat{h}_i}\right)^2\right] \\
&+ \sum_{i=0}^{S^2}\sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i-\hat{C}_i\right)^2 \\
&+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2}\sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i-\hat{C}_i\right)^2 \\
&+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c)-\hat{p}_i(c))^2
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{L} &= \mathcal{L}_S + \lambda \mathcal{L}_C \\
&= -\sum_{i=1}^{m} \log \frac{e^{W_{y_i}^T \boldsymbol{x}_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T \boldsymbol{x}_i + b_j}} + \frac{\lambda}{2}\sum_{i=1}^{m} \|\boldsymbol{x}_i - \boldsymbol{c}_{y_i}\|_2^2
\end{aligned}
$$

Analysis

	Accuracy	Relative Error	Image Size
Baseline	99.2	-	250x250 pix
Crop pre-training	98.2	+1.0	166x166 pix
Crop post-training	98.7	+0.5 (50% reduction in error disparity)	166x166 pix

Conclusion and Future Work

By simulating YOLO output by using random croppings on LFW images, we were able to reduce base image size by 54%. Further training on FaceNet allowed for comparable error rates.

In its original state, FaceNet relies on MTCNN, an image-pyramid based bounding box locator; far slower than YOLO. Cutting out the alignment step in favor of YOLO should allow faster pre-processing, without loss of accuracy in classification.

Before the final write-up, we will run two more additional tests with a modified FaceNet, to see how speed improves with full-sized inputs of size 250 and 166, respectively. Eliminating image size normalization will give us a definitive idea of how much YOLO croppings can speed up FaceNet.

Given an additional six months, we would optimize the entire pipeline by merging the two models into a single framework, speeding it up by eliminating excess file reading.

References: [1] YOLO https://github.com/allanzelener/YAD2K, [2] FaceNet https://github.com/davidsandberg/facenet, [3] WIDER Face http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/, [4] LFW http://vis-www.cs.umass.edu/lfw/