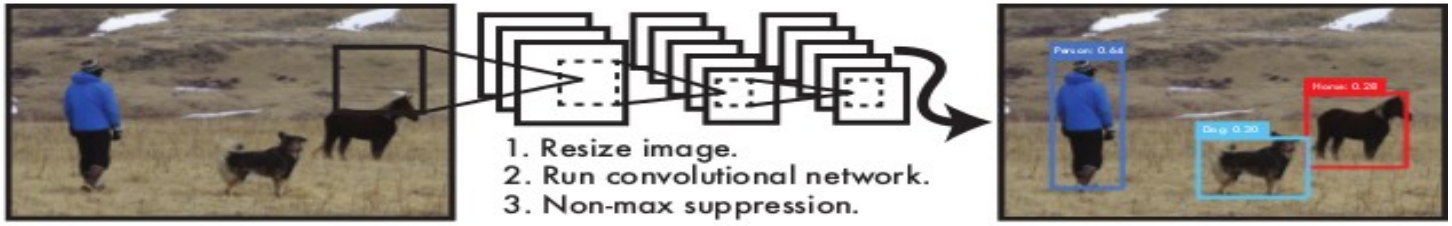
You Only Look Once: Unified, Real-Time Object Detection

https://arxiv.org/pdf/1506.02640v5.pdf

Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi
University of Washington, Allen Institute for AI,Facebook AI Research
http://pjreddie.com/yolo/



1. A single neural network predicts bounding boxes and class probabilities. 2. Base YOLO model runs at 45 FPS. A smaller version of the network, Fast YOLO, runs astounding 155 FPS second, outperforms DPM (deformable parts models) and R-CNN.

Figure 1: The YOLO Detection System. (1) resizes image to 448 × 448, (2) runs a single convolutional network on the image, and (3) thresholds the result by the model's confidence.

Tutorial: https://pjreddie.com/darknet/yolo/

GitHub, Inc. (US)

```
git clone https://github.com/pjreddie/darknet
```

From Handwritten Digits Detection to Yolo Object Detection

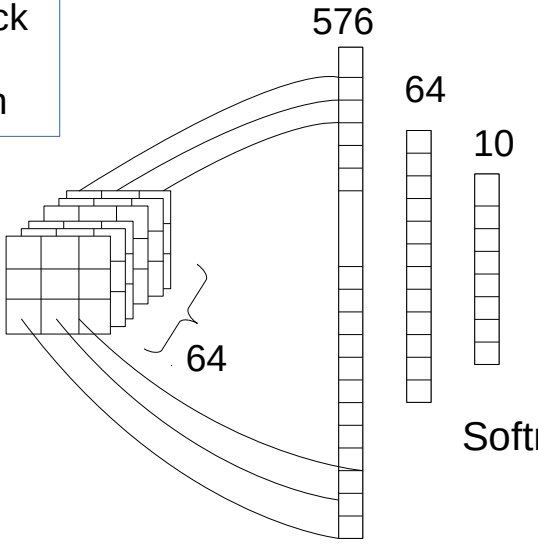
28x28

Image of digits 28x28

BackboneM NIST CNN

Output Block Softmax Activation

Backbone: CNN for feature extraction



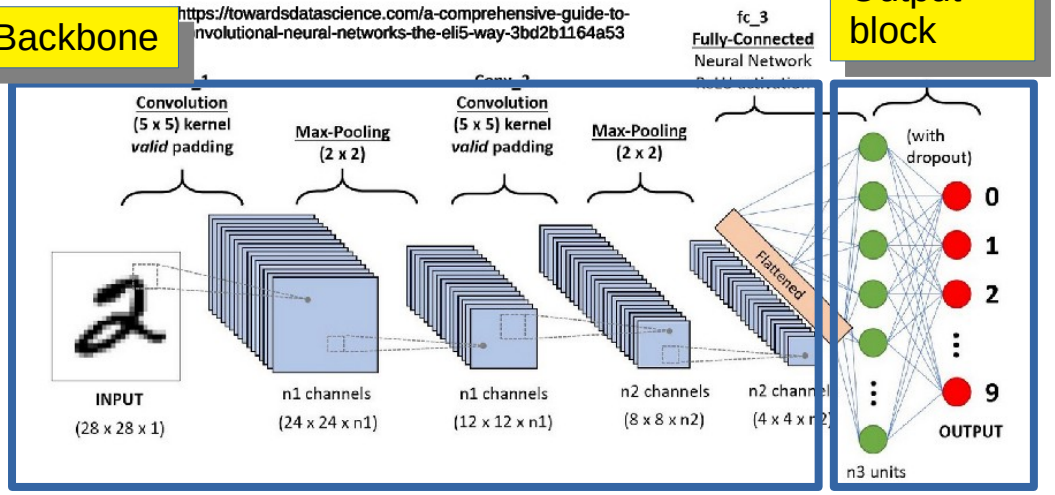
The last backbone layer CNN: 64 layers, each layer is 3x3

Illustration of A CNN for Digits Recognition

https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

Backbone

Output block

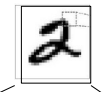


Backbone

Output block

$$C_1 M_1 C_2 M_2 C_3 \quad F_{576} D_{64} D_{10}$$

From 10 Patterns (Objects) to More Objects (80 and beyond)



28x28

Image of digits 28x28

BackboneM NIST CNN

Output Block Softmax Activation

576

64

10

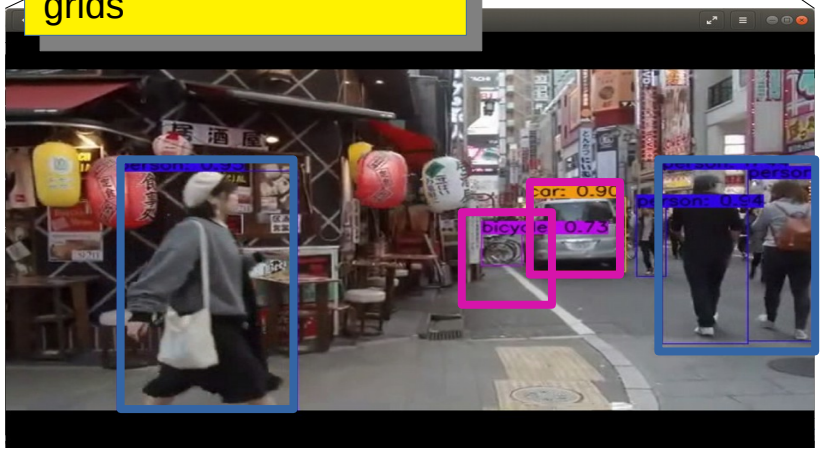
Softmax

64

Backbone: CNN for feature extraction

The last backbone layer CNN: 64 layers, each layer is 3x3

1. Image Size Increases, typical 2000 grids

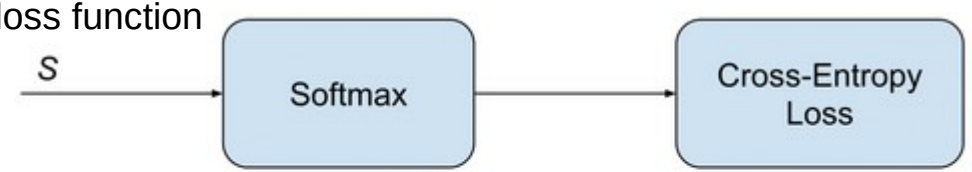


2. CNN feature extraction has to be more powerful: lead to 4096 feature vectors

3. Output block used SVM (support vector machine) to classify object classes up to 80, 120 and beyond etc.

Modification of Loss Functions

MNST Loss Function:
Categorical cross entropy loss function



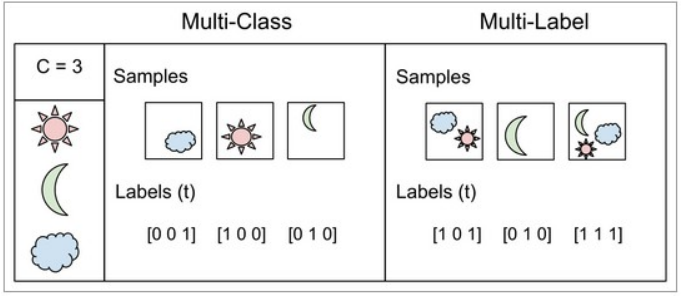
$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \qquad CE = -\sum_i^C t_i log(f(s)_i$$

2. The loss function or the objective function is ”categorical_crossentropy”, which minimzes the loss function based on the probability of the likelihood of the predicted output class.

s_i: score, output from the NN
t_i: ground truth

Weighted ground truth t_i with log(1/prob(s_i))

https://gombru.github.io/2018/05/23/cross_entropy_loss/



Yolo

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

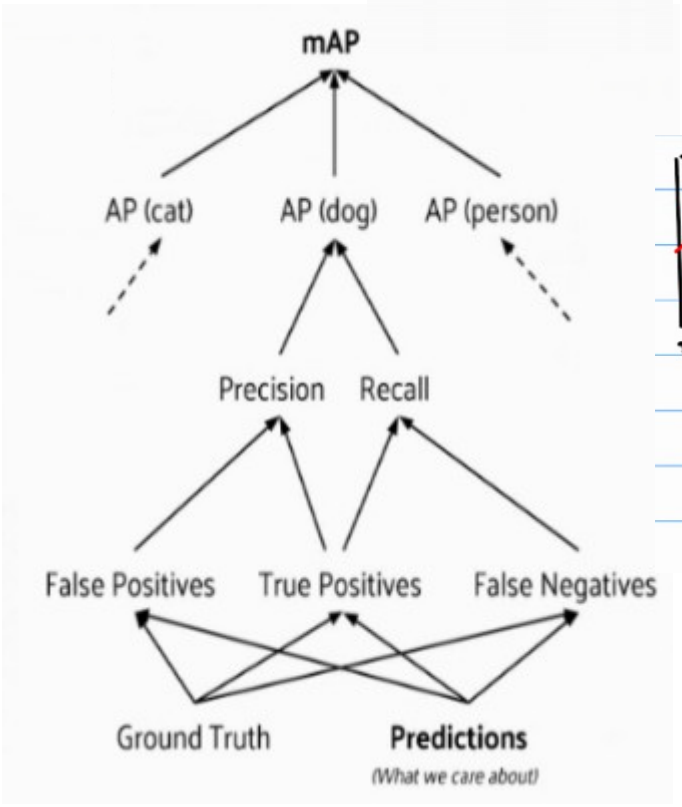
$$+ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2$$

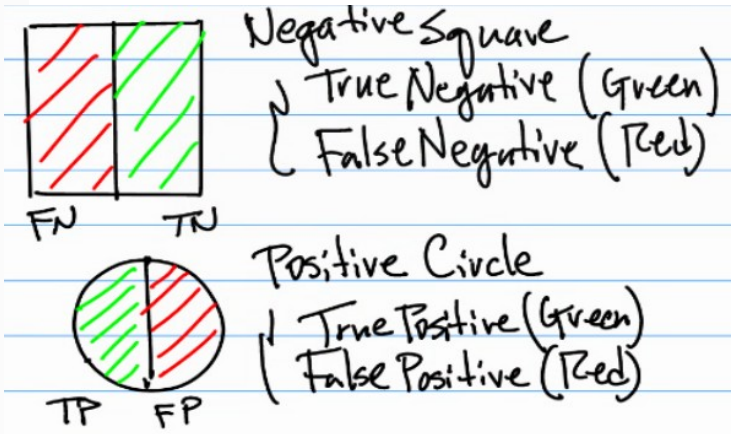
$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2$$

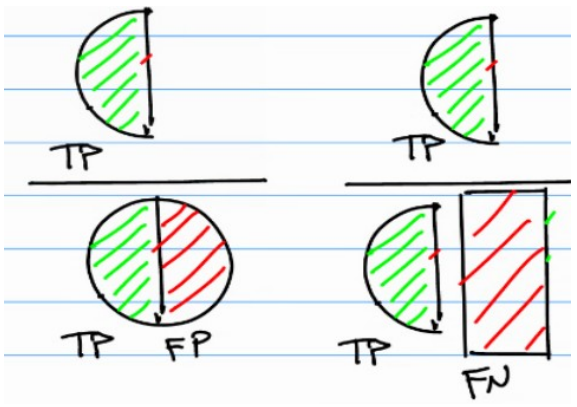
$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

MAP (Mean Average Precision) Calculation



$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$





3 Steps MAP Calculation

Step 1. Computer P (precision) and R (recall) and create a ranking table based on descending order of r value;

Step 2. Interpolate r for the missing r points; and estimate interpolated P (precision) for each point of r, e.g., 0, 0.1, 0.2, …, 0.9, 1.0;

Step 3. Compute AP based on the interpolated precision;

Step 4. Repeat Step 1 – 3 for each class, find AP for each class, then take average.

$$AP = \int_0^1 P(r)\,dr \quad \cdots (1)$$

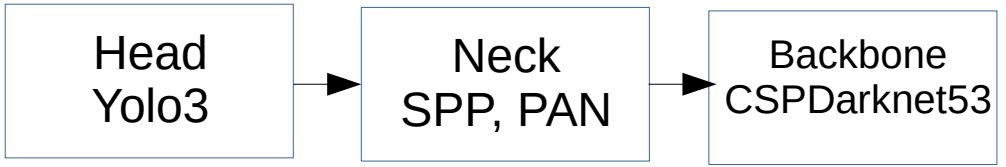
$$\approx \sum_{i=0}^{N-1} P(r_i) \quad \cdots (2)$$

$$\tilde{P}(r) = \max_{r \geq r_k} P(r_i) \qquad k \leq i \leq N-1 \quad \cdots (3)$$

$$AP = \frac{1}{N} \sum_{i=0}^{N-1} \tilde{P}(r_i)$$

Yolo4 Architecture

https://medium.com/voxel51/fifteen-minutes-with-fiftyone-yolov4-180cf66923a9



Backbone: CNN for feature extraction purpose

Note: The region proposal network (RPN) to find the Regions of Interests (ROI), xbar, ybar, width, height, and confidence