# CS776A: Deep Learning for Computer Vision

- **End Term Project Presentation: Active Learning for Deep Object Detection**

Team: Brute Force

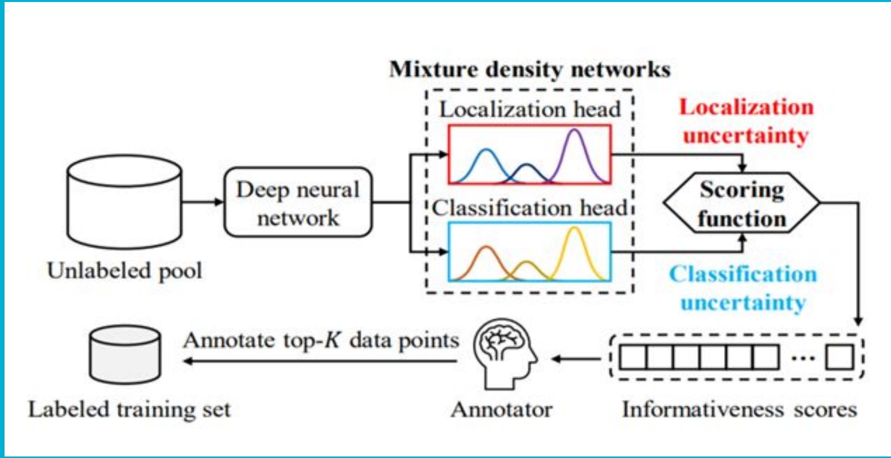
Team Members:

Abhinav Kumar (16907018)

Nishant Kiran Valvi (21111407)

# Problem Statement

- The performance of deep neural networks for object detection depends on the size of the labelled data. Creating large amount of labelled data is a very costly task.
- The goal of active learning is to reduce labelling costs by selecting only the most informative samples on a dataset [1].



# Motivation

- The performance of a deep detection network is highly dependent on the size of labelled data.
- Getting labelled data is a costly and time consuming task as each image has to be annotated by an oracle.
- Moreover, it is not necessary that all data points have enough value which is to say even if we discard these low value images the model would still learn fine.
- Active learning solves all these issues by selecting data points using a scoring function and training in an online manner.

# State of the Art

- We narrowed down to the following papers:
  - Active Learning for Deep Object Detection via Probabilistic Modeling (2021) [1] (Base Paper)
  - BayesOD: A Bayesian Approach for Uncertainty Estimation in Deep Object Detector (2019) [2]
  - Deep Bayesian Active Learning with Image Data (2017) [3]
  - Localization-Aware Active Learning for Object Detection (2018) [4]

# Proposed Solution

Following the approach mentioned in [1] we apply active learning on the YOLOv3 algorithm to solve the problem statement.

1. Firstly, we take the COCO dataset [8] . We divide it into two parts, one for training and one as an unlabelled pool of data.
2. We then train the YOLOv3 [9] algorithm on M data points on Kaggle using transfer learning.
3. We now use the trained model in an online fashion, i.e., it sees the unlabeled data in an incremental way and it selects the most informative K data points from it.
4. We use scoring functions to calculate the informativeness of an image by utilizing bounding box uncertainty and class probability to select these K data points.
5. Selected images are added to the training set and the model gets re-trained on the new updated training dataset.
6. We compare the performance of our model with that of YOLOv3 trained without active learning.

# Basic Scoring Function: Score-B

- Score of an image is calculated by
    - Avg( box_confidence * box_class_probs )

- We rank images based on this score and selected K images with minimum score for next iteration.

# Aleatoric and Epistemic Uncertainties

The predictive uncertainty is decomposed into aleatoric and epistemic uncertainty.

1. Aleatoric Uncertainty
   - It refers to the inherent noise in the data such as sensor noise, and can be attributed to occlusions or lack of visual features.
2. Epistemic Uncertainty
   - It refers to the lack of knowledge of the model and is inversely proportional to the density of training data.

# Proposed Uncertainty Based Scoring Function: Score-U

- From YOLO we get the proposed bounding boxes for all the detected classes. These bounding boxes are like clusters around a detected class.
- Assuming K bounding boxes in a cluster we utilize the following scoring function

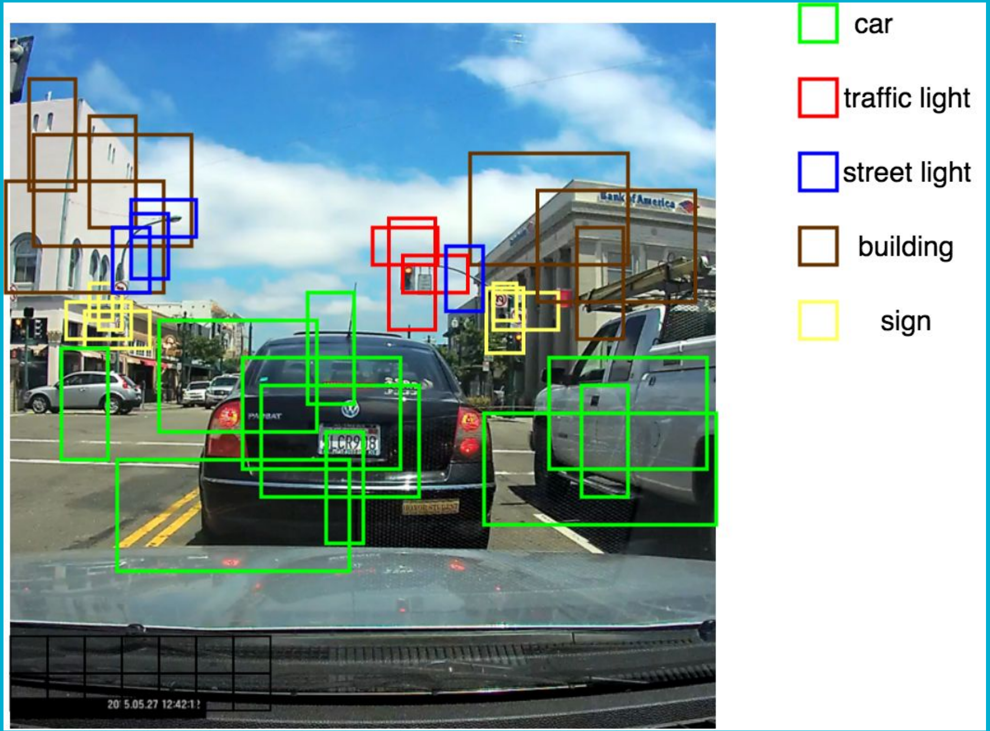
$$u_{al} = \sum_{k=1}^{K} \pi^k \Sigma^k, \quad u_{ep} = \sum_{k=1}^{K} \pi^k \|\mu^k - \sum_{i=1}^{K} \pi^i \mu^i\|^2,$$

- Here $u\_al$ i stemic uncertainty for an image.
- We also receive scores associated with a bounding box, we normalize them so that they add up to 1 and use them as the weights $\pi$ in the above scoring function.

# Uncertainty Based Scoring Function: Score-U

- We use the coordinates $x1, y1, x2, y2$ of the $K$ bounding boxes to calculate mean $\mu$ and variance $\Sigma$ for each of the coordinate to get means: $\mu\_(x1)$, $\mu\_(y1)$, $\mu\_(x2)$, $\mu\_(y2)$ and variances: $\Sigma\_(x1)$, $\Sigma\_(y1)$, $\Sigma\_(x2)$, $\Sigma\_(y2)$.
- We add up all the uncertainties for each of the clusters for all classes to get the final uncertainty value $u\_final = u\_al + u\_ep$.
- We use this final uncertainty value to rank data points to be labelled from the unlabelled pool of data.

# Working of Score-U




- Takes a cluster of bounding boxes
- Removes bounding boxes with IoU>0.5 overlap with highest score box
- Uses the rest of the bounding boxes to calculate mean and variance of the bounding box coordinates
- Utilizes these values to calculate epistemic and aleatoric uncertainty.

Src: https://medium.com/@sarangz/object-detection-using-non-max-supression-over-yolov2-382a90212b51

# Pseudo Code

- Divide dataset into training set and unlabelled pool of data points
- Train Yolo model with some M data points
- for i in range(iterations):
  - Calculate scores of images using Score-B/Score-U
  - Select images with lowest K scores and remove them from unlabelled pool
  - Add these images to already labelled data points to get M = M + K data points
  - Train Yolo model with this new updated dataset M
- End for loop

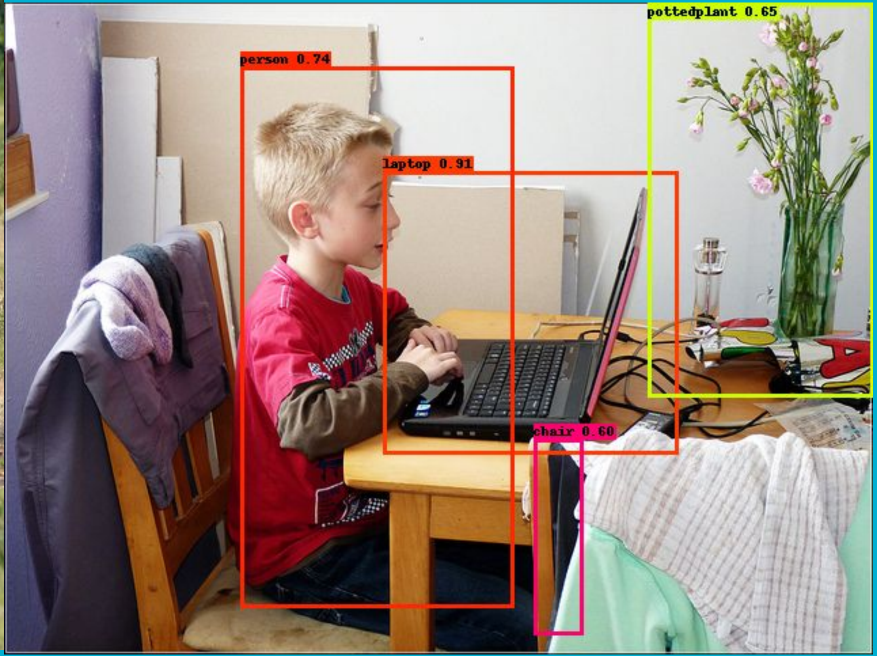
Total unlabelled data was 10,000. Initially we trained on M = 500 data points.

In the online active learning loop we kept selecting K = 500 data points for iterations = 5 iterations, resulting in total of 3,000 data points.

# Results

| Algorithms/ Scores | Pre-trained Yolov3 | Baseline Yolov3 (10k data-points) | Score-B (3k data-points) | Score-U (3k data-points) |
|---|---|---|---|---|
| MAP | 30.35 % | 23.62 % | 18.98 % | 16.62 % |
| MAP 50 | 48.61 % | 38.73 % | 32.73 % | 29.98 % |
| MAP 75 | 33.54 % | 25.73 % | 20.22 % | 16.73 % |
| MAP Large | 47.34 % | 36.53 % | 30.08 % | 25.94 % |
| MAP Medium | 27.67 % | 20.36 % | 15.12 % | 14.69 % |
| MAP Small | 6.71 % | 4.9 % | 3.10 % | 3.7 % |

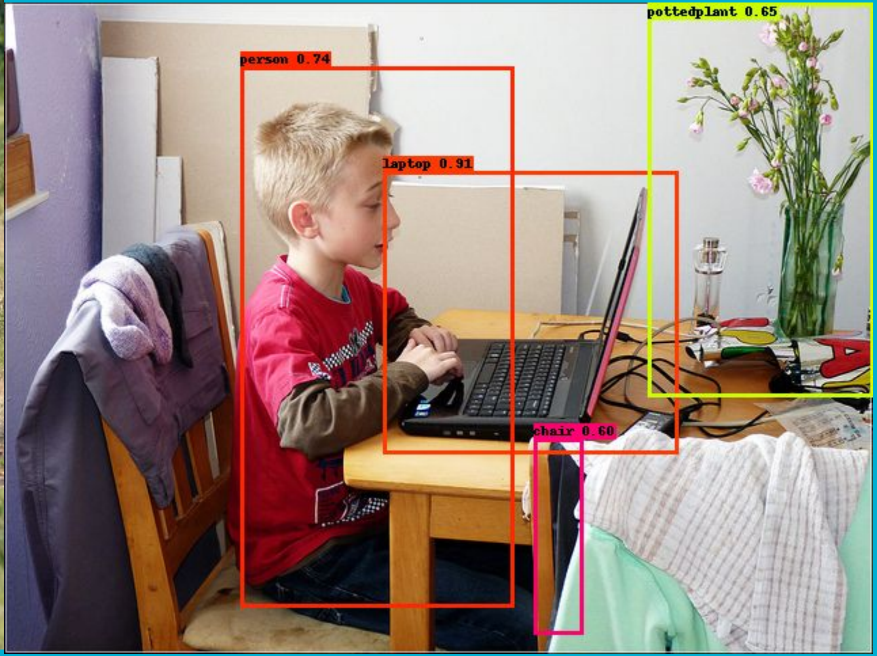
# Results: Score-B


Score: 0.6 - more informative

Score: 0.99 - less informative

# Results: Score-B


Score: 0.6 - more informative

Score: 0.99 - less informative

# Results: Score-U


Low score - more informative


High Score - less informative

# Results: Score-U



Predicted result

True label

# Contributions

| Member Name | Literature Survey | Code | Presentation |
|---|---|---|---|
| Abhinav Kumar | 50% | 50% | 50% |
| Nishant Kiran Valvi | 50% | 50% | 50% |

# References


1. Active Learning for Deep Object Detection via Probabilistic Modeling
2. BayesOD: A Bayesian Approach for Uncertainty Estimation in Deep Object Detectors
3. Deep Bayesian Active Learning with Image Data
4. Localization-Aware Active Learning for Object Detection
5. Variational Adversarial Active Learning
6. A Survey of Deep Active Learning
7. A survey on active learning and human-in-the-loop deep learning for medical image analysis
8. Microsoft COCO: Common Objects in Context
9. YOLOv3: An Incremental Improvement

# Thank You