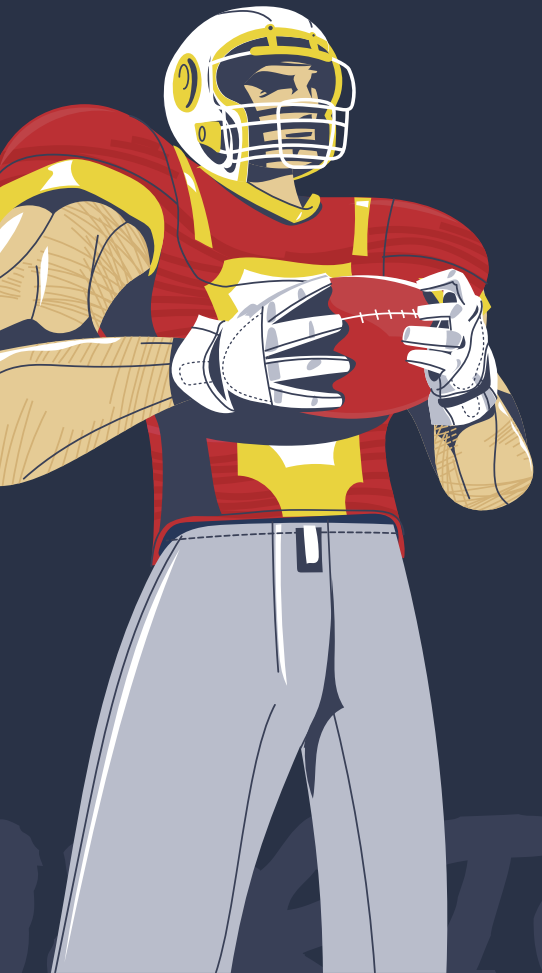


Helmet Tracking With YOLOv5 and Deepsort

NURFARIESYA FATIN BINTI AHMAD SAFUAN | A175116
AMAL MAJIDA BINTI MUNIR | A174807

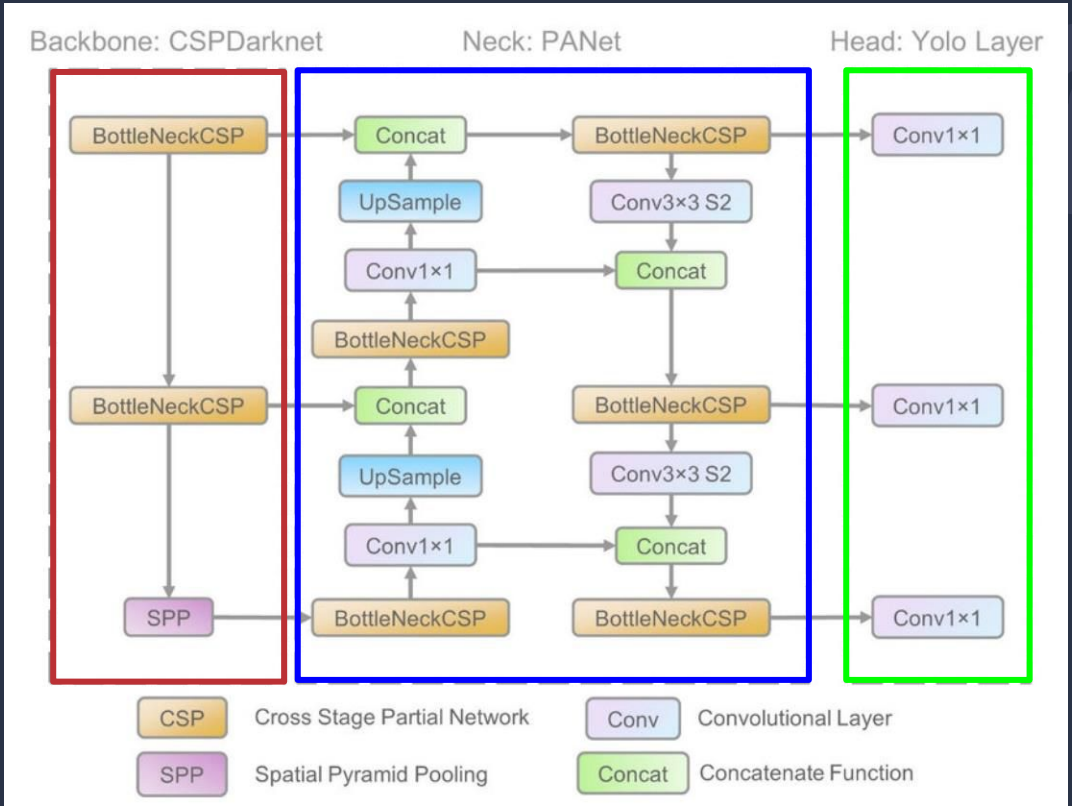
INTRODUCTION

- The National Football League (NFL) and Amazon Web Services (AWS) want to develop a sports injury surveillance by detecting helmet on each player in the field.
- Helmet in rugby play an important role to each player as wearing it can prevent soft tissue injuries to the head and ears.
- Wearing helmet can effectively reduce the risk of lacerations and abrasions to a player's head, as well as the risk of sustaining cauliflower ear.

YOLOv5 Architecture

- There are 3 important parts in YOLOv5 which are
  - Model Backbone
  - Model Neck
  - Model Head
- Model Backbone is used to extract important features in the image.
- Model Neck will generate feature pyramid by using PANet to perform aggregation on the features and pass to Model Head.
- Model Head will perform a final detection (Result).

YOLOv5 Architecture



STEP USE IN THIS PROJECT

01 Collecting Data

- Delete some data
- Modify certain data and label

02 Setup the Environment

- Installing environment and requirements

03 Training Weight

04 Result / Output

DATASET

- The dataset used in this project is NFL Health & Safety - Helmet Assignment dataset obtained from kaggle.
- There are a total of 9947 images of rugby gameplay in this dataset.
- Due to the large number of images, we decided to only used 1200 images for this project.
- We split the dataset with ratio of 80% for training and 20% for validation as shown in the figure below.

```
[ ]  list_images = helmet_pd['image'].unique()
     len(list_images)

     1200
```

Total images

```
[ ]  from sklearn.model_selection import train_test_split

     lst_train, lst_valid = train_test_split(list_images, test_size=0.2, random_state=42)
     print(len(lst_train))
     print(len(lst_valid))

     1080
     120
```

Train

Validation

The Codes!

TRAINING WEIGHT

- We mount the google drive in the codes to access the dataset and ensure that all files are save in the same directory.
- The labels for our dataset are done in .csv format (imageLabels.csv)

```
[ ]  import pandas as pd
     import os
```

```
[ ]  from google.colab import drive
     drive.mount('/content/gdrive')
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force_remount=True).
```

```
helmet_pd = pd.read_csv("/content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/imageLabels.csv")
helmet_pd.head()
```

	image	label	left	width	top	height
0	57502_000480_Endzone_frame0495.jpg	Helmet	403	30	296	33
1	57502_000480_Endzone_frame0495.jpg	Helmet	421	27	335	41
2	57502_000480_Endzone_frame0495.jpg	Helmet	439	24	374	49
3	57502_000480_Endzone_frame0495.jpg	Helmet	457	21	413	57
4	57502_000480_Endzone_frame0495.jpg	Helmet	475	18	452	65

Installing YOLOv5 environment

```
[ ]  %cd /content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet
     !git clone https://github.com/ultralytics/yolov5  # clone repo
     %cd yolov5
     !git checkout v6.0
     !pip install -r requirements.txt
```

Configures file and directory structure for train and valid

```
%%writefile NFL.yaml

train: /content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/images/train
val: /content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/images/valid

# number of classes
nc: 1

# class names
names: ['Helmet']
```

```
Writing NFL.yaml
```

```python
import matplotlib.pyplot as plt
import matplotlib.patches as patches
from PIL import Image
import random

file_name = random.choice(lst_train).split(".")[0]

label_path = "/content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/labels/train/"+file_name+".txt"
fl = open(label_path, 'r')
data = fl.read().split("\n")
fl.close()

im = Image.open('/content/gdrive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/images/train/'+file_name+'.jpg')
# Create figure and axes
fig, ax = plt.subplots()
fig.set_figheight(15)
fig.set_figwidth(15)
# Display the image
ax.imshow(im)
```

This code is used to ensure that each images has the correct labels. The image are selected randomly.

Start Training the weight

```
!WANDB_MODE="disabled" python train.py --img 1280 --batch 8 --epochs 15 --data NFL.yaml --weights yolov5m6.pt
```

```
Using 2 dataloader workers
Logging results to runs/train/exp
Starting training for 15 epochs...

      Epoch   gpu_mem       box       obj       cls    labels  img_size
       0/14       11G    0.1407   0.08964         0       146      1280: 100% 135/135 [13:29<00:00,  5.99s/it]
                Class     Images     Labels          P          R     mAP@.5 mAP@.5:.95: 100% 8/8 [00:18<00:00,  2.36s/it]
                  all        120       2506      0.357      0.377      0.294      0.069
```

Result - Accuracy: 0.885

```
      Epoch   gpu_mem       box       obj       cls    labels  img_size
      14/14      11.3G   0.06058   0.08012         0       226      1280: 100% 135/135 [13:27<00:00,  5.98s/it]
                Class     Images     Labels          P          R     mAP@.5 mAP@.5:.95: 100% 8/8 [00:16<00:00,  2.08s/it]
                  all        120       2506      0.954        0.8      0.886      0.514

15 epochs completed in 3.432 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 71.4MB
Optimizer stripped from runs/train/exp/weights/best.pt, 71.4MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model Summary: 378 layers, 35248920 parameters, 0 gradients, 49.0 GFLOPs
                Class     Images     Labels          P          R     mAP@.5 mAP@.5:.95: 100% 8/8 [00:23<00:00,  2.95s/it]
                  all        120       2506      0.954        0.8      0.885      0.514
Results saved to runs/train/exp
```

YOLOv5 AND DEEPSORT

Mount google drive

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
[ ] %cd /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet
    !git clone --recurse-submodules https://github.com/mikel-brostrom/Yolov5_DeepSort_Pytorch.git
```

```
/content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet
```

Installing requirements and dependencies

```
[ ] %cd /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/Yolov5_DeepSort_Pytorch
    !pip install -r requirements.txt
```

```
/content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/Yolov5_DeepSort_Pytorch
Requirement already satisfied: matplotlib>=3.2.2 in /usr/local/lib/python3.7/dist-packages (from -r requir
Requirement already satisfied: numpy>=1.18.5 in /usr/local/lib/python3.7/dist-packages (from -r requiremer
Requirement already satisfied: opencv-python>=4.1.2 in /usr/local/lib/python3.7/dist-packages (from -r req
```

```
import torch
##torch.cuda.is_available()
from IPython.display import Image, clear_output

clear_output()
print(f"Setup complete. Using torch {torch.__version__} ({torch.cuda.get_device_properties(0).name if torch.cuda.is_available() else 'CPU'})")
```

```
Setup complete. Using torch 1.10.0+cu111 (Tesla K80)
```

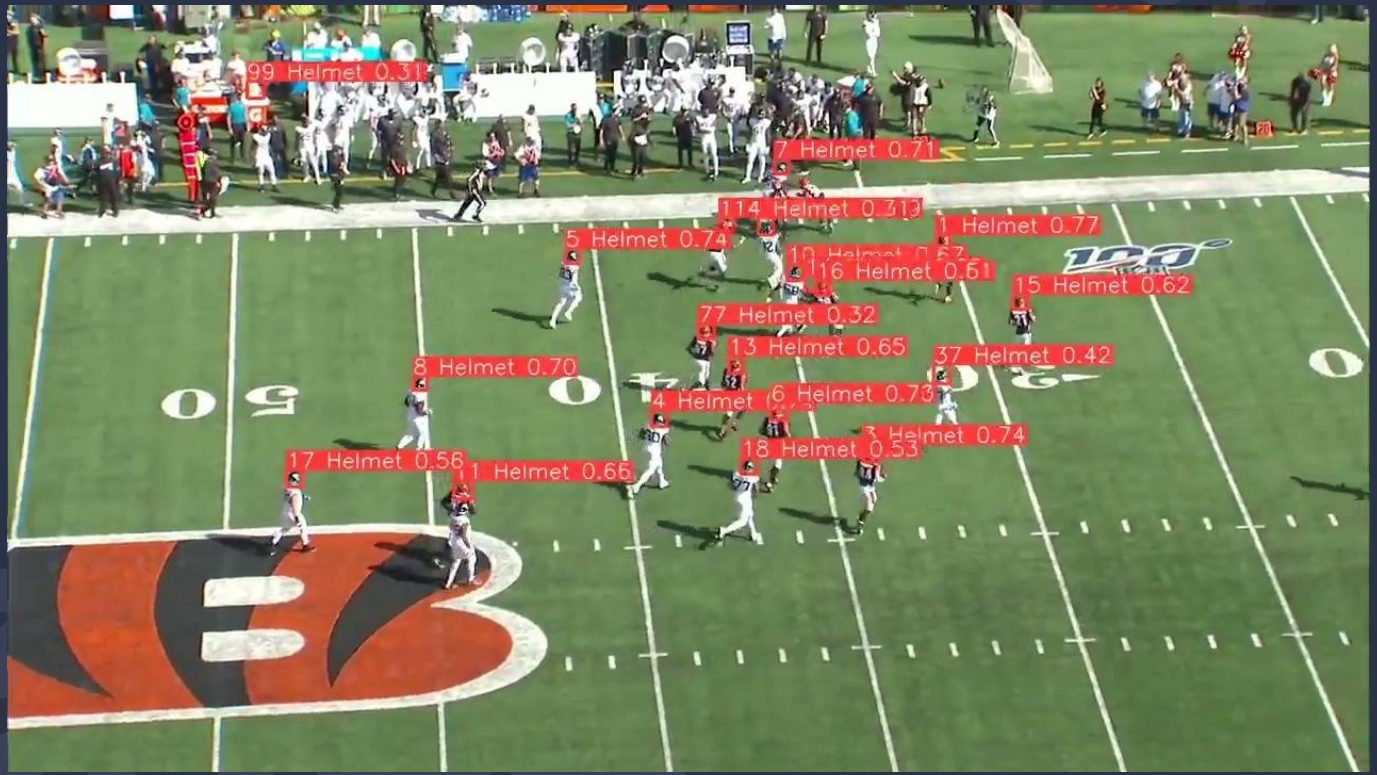
```
%cd /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/Yolov5_DeepSort_Pytorch
```

```
/content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/Yolov5_DeepSort_Pytorch
```

```
!python3 track.py --yolo_model "/content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Helmet/Yolov5_DeepSort_Pytorch/best.pt" --source "/content/drive/Shareddrives
```

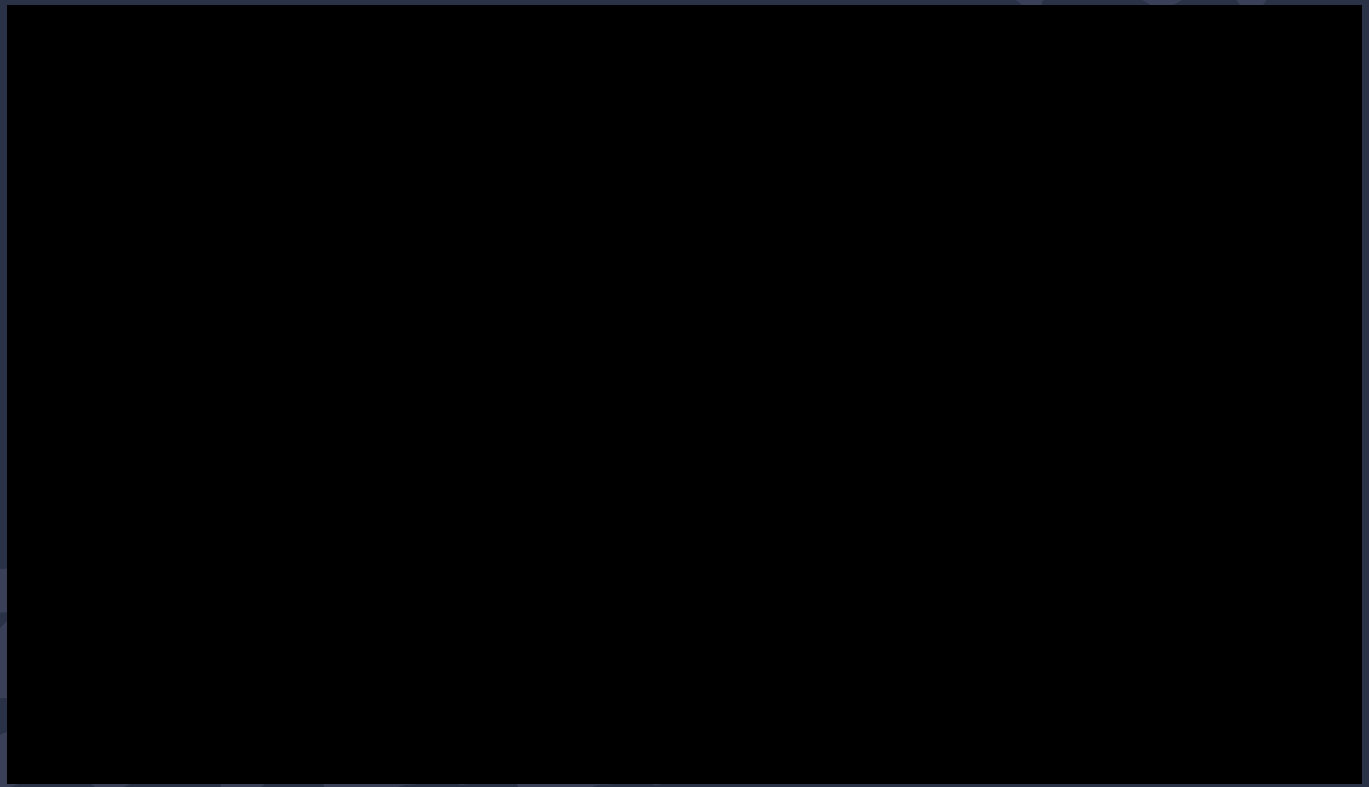
```
video 1/1 (655/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.068s), DeepSort:(0.061s)
video 1/1 (656/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 18 Helmets, Done. YOLO:(0.068s), DeepSort:(0.062s)
video 1/1 (657/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.068s), DeepSort:(0.055s)
video 1/1 (658/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.068s), DeepSort:(0.056s)
video 1/1 (659/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.068s), DeepSort:(0.053s)
video 1/1 (660/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.068s), DeepSort:(0.055s)
video 1/1 (661/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.066s), DeepSort:(0.054s)
video 1/1 (662/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.066s), DeepSort:(0.058s)
video 1/1 (663/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 17 Helmets, Done. YOLO:(0.066s), DeepSort:(0.056s)
video 1/1 (664/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.067s), DeepSort:(0.058s)
video 1/1 (665/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.067s), DeepSort:(0.059s)
video 1/1 (666/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 23 Helmets, Done. YOLO:(0.066s), DeepSort:(0.068s)
video 1/1 (667/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.067s), DeepSort:(0.060s)
video 1/1 (668/710) /content/drive/Shareddrives/Balance Unlimited/Robot Projek 2/Video/Video 2 (Slowmo).mp4: 384x640 19 Helmets, Done. YOLO:(0.067s), DeepSort:(0.059s)
```

OUTPUT



RUGBY TOURNAMENT

OUTPUT (CLOSE UP)



THANK YOU