# Enhancing Digital Privacy:

## Utilizing YOLOv8n for Sensitive Information Detection in WeChat Screenshots

Sara Zhang

# Table of contents

01

# Introduction

# Project Introduction

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

Digital Records Sharing

Digital Privacy

# Project Introduction

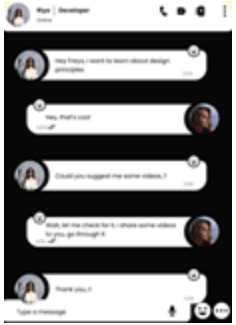
| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Objective**: safeguarding the private information displayed on screenshots before sharing

# Project Introduction

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

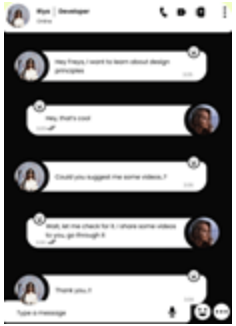
**Objective**: safeguarding the private information displayed on screenshots before sharing



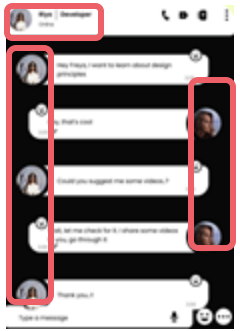
# Project Introduction

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Objective**: safeguarding the private information displayed on screenshots before sharing



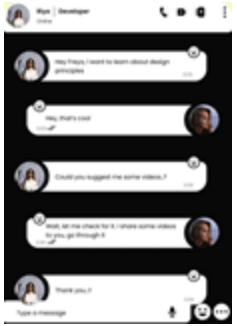
Detect
Private
Information



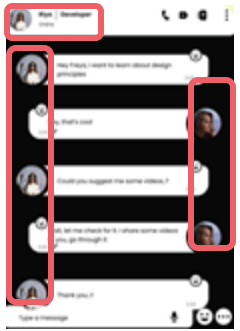
# Project Introduction

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

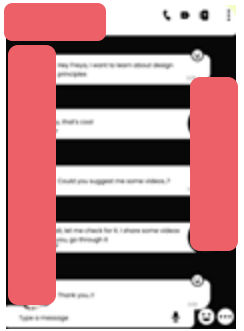
**Objective**: safeguarding the private information displayed on screenshots before sharing



Detect Private Information



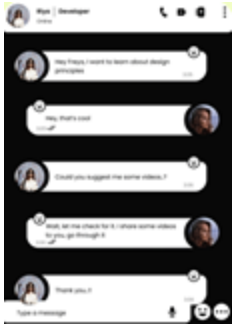
Anonymizing private information



# Project Introduction

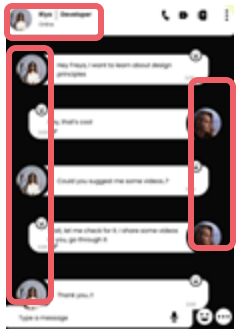
| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Objective**: safeguarding the private information displayed on screenshots before sharing

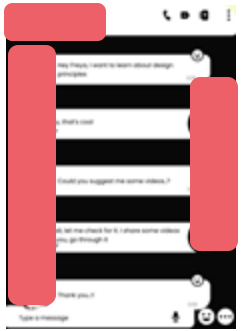


**Computer Vision**

Detect Private Information



Anonymizing private information



# Object Detection

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Goal:** answer "**what** objects are **where?**"

# Object Detection



**Goal:** answer "**what** objects are **where?**"

coordinates of the objects + Confidence level

# Object Detection



**Goal:** answer "**what** objects are **where?**"

**Algorithms:**

- Histogram of Oriented Gradients(HOG)
- Region-based Convolutional Neural Networks (R-CNN)
- Region-based Fully Convolutional Network (R-FCN)
- YOLO (You Only Look Once)

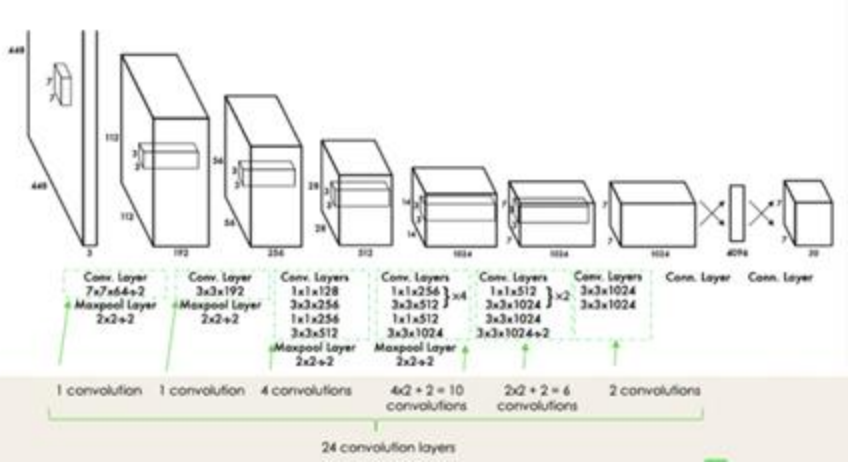
02

# Methodology

# Model Overview: YOLOv8



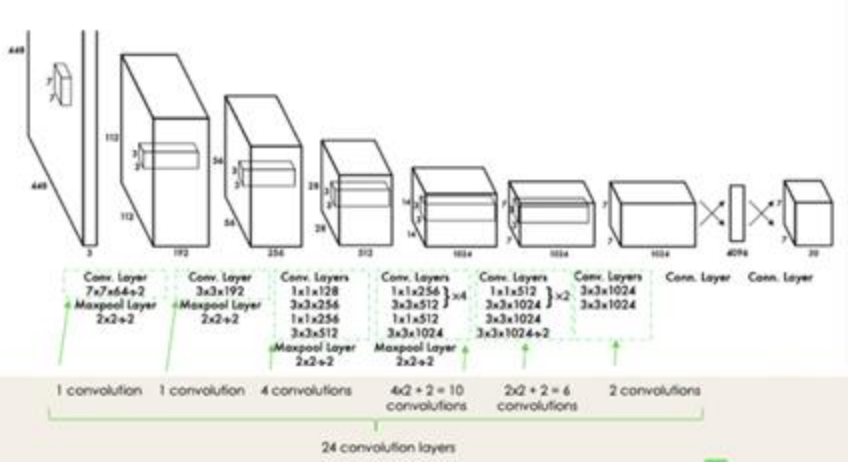
**You Only Look Once (YOLO):** state-of-the-art, real-time object detection algorithm firstly introduced in 2015



# Model Overview: YOLOv8

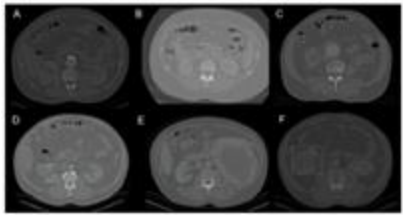


**You Only Look Once (YOLO):** state-of-the-art, real-time object detection algorithm firstly introduced in 2015




**Advantages:**

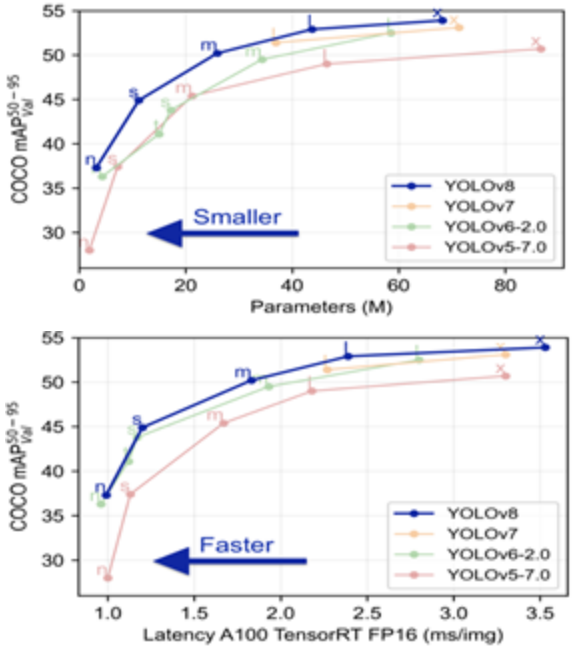
- Speed
- Detection accuracy
- Good generalization
- Open-source
- Broad scope of applications



https://www.datacamp.com/blog/yolo-object-detection-explained

# Model Overview: YOLOv8

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|




| Model | Filenames | Task | Inference | Validation | Training | Export |
|---|---|---|---|---|---|---|
| YOLOv8 | yolov8n.pt<br>yolov8s.pt<br>yolov8m.pt<br>yolov8l.pt<br>yolov8x.pt | Detection | ✅ | ✅ | ✅ | ✅ |
| YOLOv8-seg | yolov8n-seg.pt<br>yolov8s-seg.pt<br>yolov8m-seg.pt<br>yolov8l-seg.pt<br>yolov8x-seg.pt | Instance Segmentation | ✅ | ✅ | ✅ | ✅ |
| YOLOv8-pose | yolov8n-pose.pt<br>yolov8s-pose.pt<br>yolov8m-pose.pt<br>yolov8l-pose.pt<br>yolov8x-pose.pt<br>yolov8x-pose-p6.pt | Pose/Keypoints | ✅ | ✅ | ✅ | ✅ |
| YOLOv8-obb | yolov8n-obb.pt<br>yolov8s-obb.pt<br>yolov8m-obb.pt<br>yolov8l-obb.pt<br>yolov8x-obb.pt | Oriented Detection | ✅ | ✅ | ✅ | ✅ |
| YOLOv8-cls | yolov8n-cls.pt<br>yolov8s-cls.pt<br>yolov8m-cls.pt<br>yolov8l-cls.pt<br>yolov8x-cls.pt | Classification | ✅ | ✅ | ✅ | ✅ |

https://docs.ultralytics.com/models/

# Loss Function



**Box Loss:**
Measures how accurately the model locates objects within their bounding boxes.

7.5

**Classification Loss:**
Ensures objects are correctly classified according to their labels.

0.5

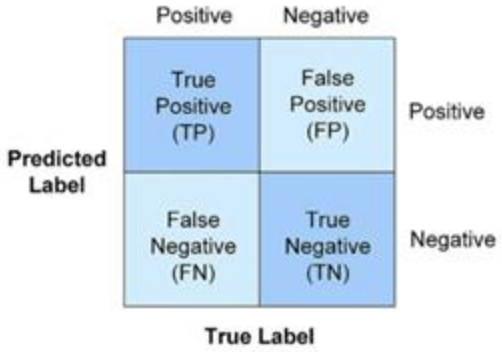
**Distribution Focal Loss**: Addresses class imbalance within the object detection process.

1.5

**Total Loss**

# Related Metrics





# Related Metrics



| | | Positive | Negative |
|---|---|---|---|
| Predicted Label | Positive | True Positive (TP) | False Positive (FP) |
| | Negative | False Negative (FN) | True Negative (TN) |

True Label

$$\text{recall} = \frac{TP}{TP + FN} \qquad \text{precision} = \frac{TP}{TP + FP}$$

$$F_1 = \frac{2 * (\text{precision} * \text{recall})}{(\text{precision} + \text{recall})}$$

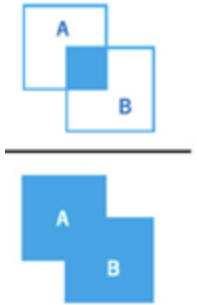
- **Recall:** The ability of the model to identify all instances of objects in the images.
- **Precision:** The accuracy of the detected objects, indicating how many detections were correct.

# Related Metrics



- **Intersection over Union (IoU)**

$$IoU = \frac{(\text{Area of Intersection})}{(\text{Area of Union})} = \frac{|A \cap B|}{|A \cup B|}$$

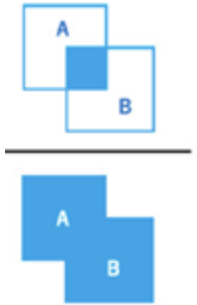


# Related Metrics



- **Intersection over Union (IoU)**

$$IoU = \frac{(\text{Area of Intersection})}{(\text{Area of Union})} = \frac{|A \cap B|}{|A \cup B|}$$




- **mAP50:** It's a measure of the model's accuracy considering only the "easy" detections.
- **mAP50-95:** It gives a comprehensive view of the model's performance across different levels of detection difficulty.

03

# Experiment

# Experiment Setup



- Datasets:
  - 130 WeChat **screenshots** (93 for training, 25 for validation, 12 for testing)

# Experiment Setup



- Datasets:

```
data
│
└───images
│   ├───train
│   ├───val
│   └───test
│
└───labels
    ├───train
    ├───val
    └───test
```

  - 130 WeChat **screenshots** (93 for training, 25 for validation, 12 for testing)

# Experiment Setup



- Datasets:

```
data
│
└───images
│   ├───train
│   ├───val
│   └───test
│
└───labels
    ├───train
    ├───val
    └───test
```

  - 130 WeChat **screenshots** (93 for training, 25 for validation, 12 for testing)
  - 510 instances of **personal information**
  - manually **annotated** through Computer Vision Annotation Tool (CVAT)

# Experiment Setup



- Datasets:

# Model Training



**Environment:** colab

**Training Settings:**

batch size = 16, learning rate = 0.01 ~ 0.01

momentum = 0.937, and weight decay = 0.0005.

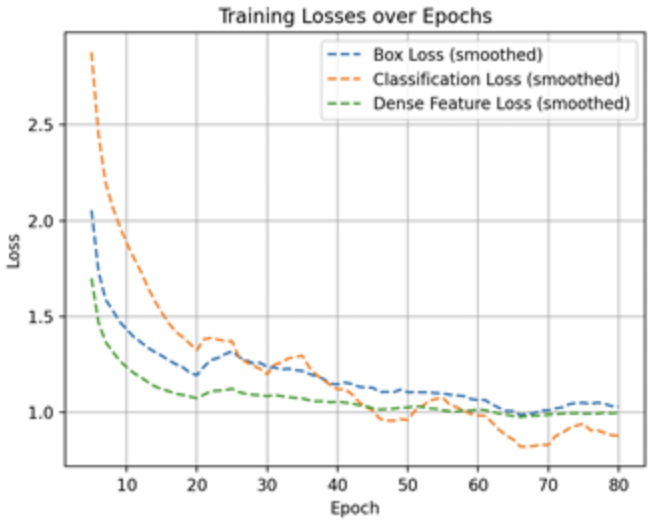
**Augmentation Settings and Hyperparameters:**

hue, saturation, brightness, rotation, translation, scaling, and shearing

*See YOLO document: https://docs.ultralytics.com/modes/train/#train-settings*

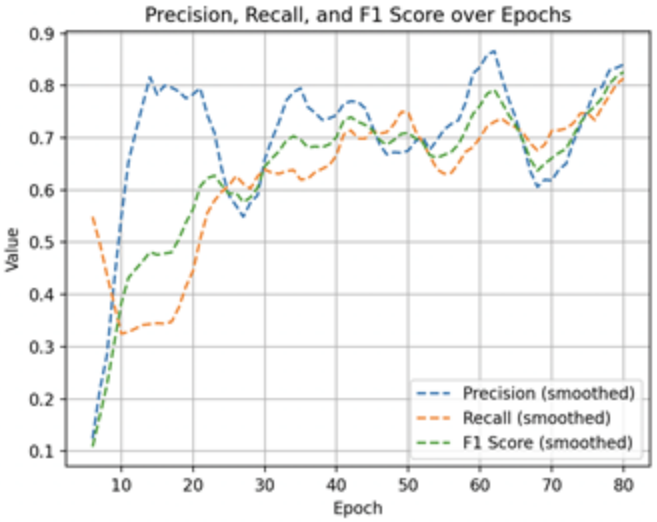
# Model Training

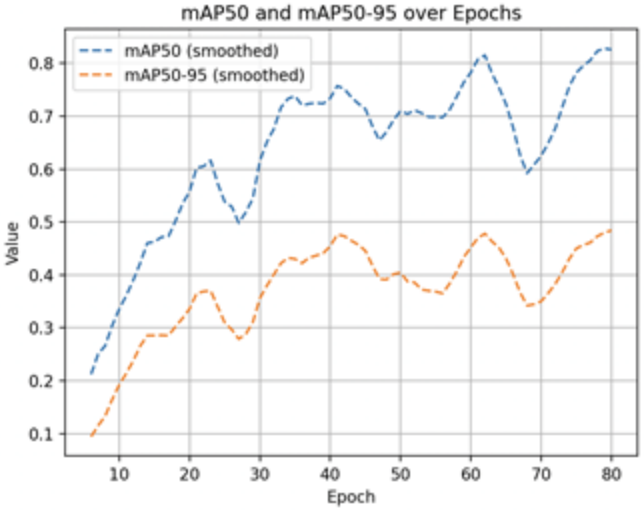
| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

# Model Training





# Performance Evaluation

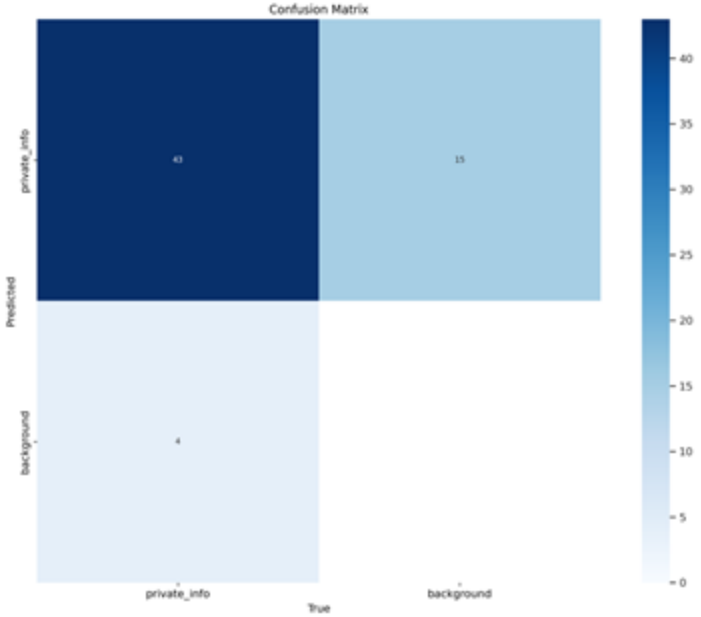


- Tested on 12 images with 47 instances

# Performance Evaluation

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

- Tested on 12 images with 47 instances
- **Precision: 96.8%**
- **Recall: 85.1%**
- **mAP50: 95.2%**
- **mAP50-95: 65.8%**



# Performance Evaluation



- Example Ground Truth:

# Performance Evaluation



- Example Ground Truth:


Example Test Predictions:

04

# Summary

# Summary

| Introduction | Methodology | Experiment | Summary |
| --- | --- | --- | --- |

**Deep Learning for Privacy:**
Project employs YOLOv8 to boost privacy in screenshot sharing.

# Summary

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Deep Learning for Privacy:** Project employs YOLOv8 to boost privacy in screenshot sharing.


**Self-Annotated Dataset:** Trained with over 500 personally annotated instances.

# Summary

| Introduction | Methodology | Experiment | Summary |
|---|---|---|---|

**Deep Learning for Privacy:** Project employs YOLOv8 to boost privacy in screenshot sharing.


**Self-Annotated Dataset:** Trained with over 500 personally annotated instances.

**Promising Results:** Model demonstrates strong performance, indicating potential for wider platform application.

# Summary



- **Challenges and Limitation:**
  - **Annotation Consistency:** standardize the labels precision/location for elements
  - **Dataset Limitations:** inherent privacy concerns restricting the training

# Summary



- **Challenges and Limitation:**
  - **Annotation Consistency:** standardize the labels precision/location for elements
  - **Dataset Limitations:** inherent privacy concerns restricting the training

- **Future Direction:**
  - **Model and Network Diversity:** explore other architectures/models
  - **Dataset Expansion:** include screenshots from other applications/different devices
  - **Pipeline Completion:** integrating the trained model into a complete pipeline to also blur/block identified info

# Thanks