Ahmedabad University

28/02/2025

CSE641: Computer Vision : Modern Methods and Applications

Group: RCNN   Project no.:13

Weekly Report 2

Name	**Enrollment number**
Sloka Thakkar	AU2240103
Daksh Shah	AU2240207
Shalvi Modi	AU2240215

Summary:

YOLOv11 (You Only Look Once version 11) is a state-of-the-art object detection model designed for high accuracy and fast inference. It builds on previous YOLO versions with several key architectural improvements:

1. **Transformer-Based Modules**: YOLOv11 integrates vision transformers (ViTs) into its backbone network, allowing the model to capture long-range dependencies and global context information more effectively than traditional convolutional layers.
2. **Dynamic Head Structure**: The model introduces dynamic convolution heads, which adapt their structure during training based on the complexity of detected objects. This helps in better feature refinement and improves detection accuracy.
3. **Adaptive Anchor Boxes**: Instead of relying on fixed anchor box sizes, YOLOv11 uses adaptive anchor boxes that automatically adjust based on object size distribution in the dataset, reducing localization errors.
4. **Multi-Scale Feature Extraction**: The model uses multiple feature pyramid layers to extract information from different spatial scales, improving performance on small and large objects.
5. **Improved Non-Maximum Suppression (NMS)**: The model applies a more refined NMS technique to filter out duplicate detections, which reduces false positives and improves precision.
6. **Efficiency and Speed**: Despite its advanced features, YOLOv11 maintains a balance between accuracy and inference speed, making it suitable for real-time applications.

Task completed:

- Implemented YOLOv11 model on the annotated dataset with bounding boxes applied to mark the crocodile dorsal scute patterns.
- Trained YOLOv11 on the dataset to detect and localize crocodiles efficiently.
- Compared YOLOv11's performance with YOLOv5l to observe improvements in detection accuracy and reduced false positives.

Pseudo code:

```
Step 1: Preprocessing
Input: Image Dataset with Bounding Box Annotations
Output: Preprocessed Images with Bounding Boxes

Function Preprocess_Images(image_dataset):
    - Resize images to (640x640)
    - Normalize pixel values [0, 1]
    - Apply Data Augmentation (Flip, Rotation, Brightness Adjustment)
    - Split Dataset into Training, Validation, and Test sets
Return Preprocessed Images

Step 2: Model Initialization
Input: Preprocessed Dataset, Model Configuration
Output: Initialized YOLOv11 Model

Function Initialize_Model():
    - Load Pretrained Backbone Network (ConvNeXt or Swin Transformer)
    - Add Transformer-based Feature Extractor
    - Add Detection Head with Dynamic Convolution Layers
    - Apply Adaptive Anchor Box Mechanism
Return Model

Step 3: Training the Model
Input: Preprocessed Dataset, Model
Output: Trained Model with Optimized Weights

Function Train_Model(model, dataset, epochs):
    For epoch in range(1, epochs):
        For image, label in dataset:
```

```
        - Forward Pass
            Feature_Map = model.Backbone(image)
            Predictions = model.Detection_Head(Feature_Map)
        - Loss Calculation
            Localization_Loss = Smooth_L1_Loss(Predicted_BBox, Ground_Truth_BBox)
            Classification_Loss = CrossEntropyLoss(Predicted_Class, Ground_Truth_Class)
            Total_Loss = Localization_Loss + Classification_Loss
        - Backward Propagation
            Update Weights using Adam Optimizer
        Save Best Weights
Return Trained Model

Step 4: Inference
Input: Test Image, Trained Model
Output: Detected Objects with Bounding Boxes

Function Inference(model, test_image):
    - Preprocess Test Image
    - Forward Pass through the model
    - Apply Non-Maximum Suppression (NMS)
    - Draw Bounding Boxes with Confidence Score > Threshold
Return Detections

Step 5: Evaluation
Input: Model Predictions, Ground Truth
Output: Accuracy, Precision, Recall, F1-Score

Function Evaluate_Model(predictions, ground_truth):
    - Compute Intersection over Union (IoU)
    - Calculate Precision, Recall, F1-Score
Return Evaluation Metrics
```

Goals for Next week:

- To train YOLO on the training dataset.