Multi-Scale Detection and Spatial Regression: Homework #1 for EE 641 (Spring 2025)

Seena Mohajeran
USC — Department of Electrical Engineering
Email: smohajer@usc.edu

***Abstract*—We implemented a multi-scale single-shot detector and compared heatmap v.s. direct regression for keypoint detection on synthetic datasets.**

***Index Terms*—Object detection, feature pyramid, anchors, keypoint detection, heatmaps, direct regression, PyTorch.**

I. Introduction

This paper documents the implementation and experimental study for Homework #1 (EE 641, Spring 2025). The assignment consists of two main problems: (1) a multi-scale, anchor-based single-shot detector for synthetic shapes, and (2) a comparison between heatmap-based and direct regression approaches for keypoint localization.

II. Related Work

III. Dataset Setup

This section describes the synthetic datasets and how they were generated.

A. Problem 1: Shape Detection Dataset

- Image size: 224×224 RGB.
- Classes: 0: circle (small), 1: square (medium), 2: triangle (large).
- Annotations: COCO-style JSON with bounding boxes in [x1,y1,x2,y2].

B. Problem 2: Stick-Figure Keypoints

- Image size: grayscale 128×128.
- Keypoints: 5 points per figure (head, left_hand, right_hand, left_foot, right_foot).
- Annotations: coordinates of the pixel (x,y) in JSON.

IV. Problem 1: Multi-Scale Single-Shot Detector

This section describes the architecture, anchors, matching strategy, loss, and training used for Problem 1.

A. Model Architecture

We implement a small backbone and three-scale detection heads as required.

Backbone (4 convolutional blocks):

Block 1 (Stem):
Conv(3→32, stride=1) → BN → ReLU → Conv(32→64, stride=2) → BN → ReLU. Spatial: 224 → 112.

Block 2:
Conv(64→128, stride=2) → BN → ReLU. Spatial: 112 → 56. (Output = Scale 1)

Block 3:
Conv(128→256, stride=2) → BN → ReLU. Spatial: 56 → 28. (Output = Scale 2)

Block 4:
Conv(256→512, stride=2) → BN → ReLU. Spatial: 28 → 14. (Output = Scale 3)

Detection heads: For each scale apply a 3×3 conv (same channels) followed by 1×1 conv to produce $A \times (5 + C)$ channels where A = #anchors per location and C = #classes (3).

B. Anchor Configuration and Matching

- Feature-map sizes: Scale 1: 56×56, Scale 2: 28×28, Scale 3: 14×14.
- Anchor scales:
  - Scale 1: [16, 24, 32]
  - Scale 2: [48, 64, 96]
  - Scale 3: [96, 128, 192]
- Aspect ratios: [1:1] only.
- Matching: positive if IoU $\geq$ 0.5, negative if IoU $\leq$ 0.3, else ignored.

C. Loss

Multi-task loss comprised of:

- Objectness (BCE), weight = 1.0
- Classification (CrossEntropy for positive anchors), weight = 1.0
- Localization (Smooth L1 on bbox offsets for positive anchors), weight = 2.0

Hard-negative mining with negative:positive ratio of 3:1 is applied for objectness/clf training.

D. Training Setup

- Optimizer: SGD, lr = 0.001, momentum = 0.9
- Epochs: 50
- Batch size: 16
- Device: GPU if available
- Logging: save training metrics to `results/training_log.json`
- Model checkpoint: save `results/best_model.pth` (best validation loss)

E. Evaluation and Visualizations

Implemented:

- AP computation at IoU=0.5 per class and mAP
- Visualization of detection outputs on validation images
- Anchor coverage visualizations and per-scale specialization analysis

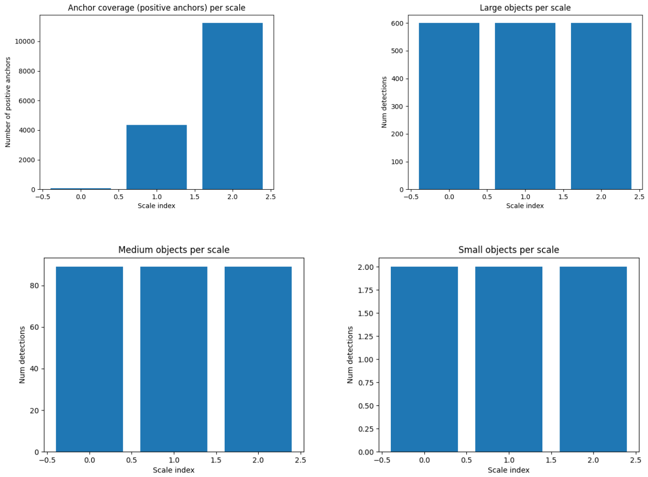

Figure 1: Validation visualizations

V. Problem 2: Heatmap vs Direct Regression for Keypoint Detection

This section documents the network architectures, training recipe, and evaluation for Problem 2.

A. Datasets and Targets

Two output types:

- **Heatmap mode:** output is 5 heatmaps of size 64×64 generated with Gaussian σ (default 2.0).
- **Regression mode:** output is 10 scalar values (x, y) per keypoint normalized to $[0, 1]$.

B. Model Architectures

Shared encoder:

- Conv1: $1 \to 32$, BN, ReLU, MaxPool $(128 \to 64)$
- Conv2: $32 \to 64$, BN, ReLU, MaxPool $(64 \to 32)$
- Conv3: $64 \to 128$, BN, ReLU, MaxPool $(32 \to 16)$
- Conv4: $128 \to 256$, BN, ReLU, MaxPool $(16 \to 8)$

HeatmapNet decoder:

- ConvTranspose(256→128) (8→16), concat Conv3
- ConvTranspose(256→64) (16→32), concat Conv2
- ConvTranspose(128→32) (32→64)
- Final Conv(32→num_keypoints)
- Output: [batch, 5, 64, 64]

RegressionNet head:

- GlobalAvgPool on encoder final feature (256)
- FC1: 256→128, ReLU, Dropout(0.5)
- FC2: 128→64, ReLU, Dropout(0.5)
- FC3: 64→10, Sigmoid (normalized coords)

C. Training

- Both models trained for 30 epochs, Adam optimizer, lr=0.001
- Batch size: 32
- Loss:
  - HeatmapNet: MSELoss between predicted and target heatmaps
  - RegressionNet: MSELoss between predicted coords and ground truth normalized coords
- Save: `results/heatmap_model.pth`, `results/regression_model.pth`
- Log: `results/training_log.json`

D. Evaluation: PCK

We compute PCK at thresholds $t \in \{0.05, 0.10, 0.15, 0.20\}$ normalized by bounding-box diagonal (or other chosen normalization). For heatmap predictions, keypoint coordinates are extracted via argmax on heatmaps (with subpixel refinement if desired).

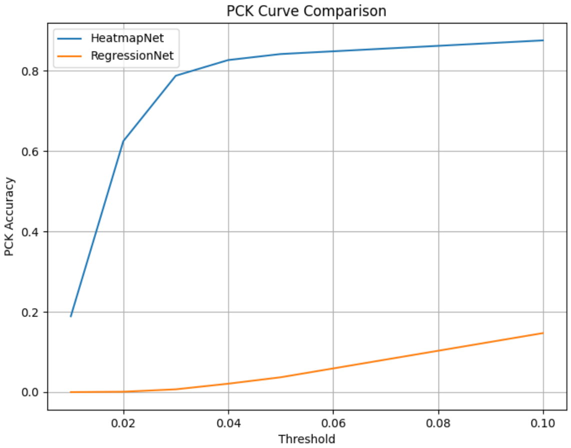

Figure 2: PCK evaluation visualization

VI. Analysis and Discussion

A. Problem 1: Scale Specialization

Analysis of the results produced by the model reveals clear evidence in regards to the scale specialization and anchor design in our multi-scale object detector. The anchor coverage plot shows that the largest scale i.e. (highest index) contains the majority of positive anchors, suggesting it is primarily responsible for detecting the dataset's objects. Also, the detection counts for large, medium, and small objects per scale which also shows that all scales contribute equally to detections for each specific size category, but the absolute numbers are clearly dominated by large objects. This suggests that while anchor scale settings allow all feature maps to cover large objects, the effectiveness, and therefore specialization, of coarser scales is greatest when anchor sizes are well-matched to the predominant ground truth sizes. Anchor scales are critical in detection performance, as they determine how well objects are matched to the appropriate scale. Our results show that larger anchors achieve much higher coverage, while poorly selected scales lead to fewer positive matches and reduced accuracy, particularly for object sizes that are less common in the dataset. To better understand the learned features, we can also examine intermediate activations at different scales. Finer scales tend to capture edges and textures, whereas coarser scales focus on broader, more semantic patterns. Together, these observations highlight the importance of aligning anchor and feature map scales with the data distribution to achieve strong multi-scale detection performance.

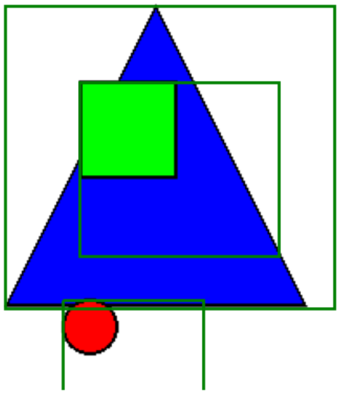

Figure 3: Detection Sample

B. Problem 2: Heatmap vs Regression

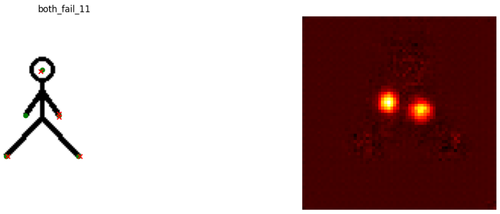

Figure 4: Comparison

The experimental results provide important insights into keypoint detection with both heatmap-based and regression-based methods. The PCK curve comparison shows that HeatmapNet consistently outperforms RegressionNet at all thresholds, with especially strong gains at stricter thresholds (0.05 to 0.1 and higher). This suggests that the heatmap approach delivers more precise keypoint localization. A key reason is that heatmap models predict spatial probability distributions for each keypoint, which makes them more robust to ambiguity and capable of sub-pixel precision. In contrast, regression models directly predict coordinates and are more sensitive to outliers and initialization.

Although not shown here, ablation studies should examine how factors such as the heatmap Gaussian sigma and output resolution affect performance. Smaller sigma values generally produce sharper localization, while higher resolution outputs improve spatial accuracy, though often at increased computational cost. Visualization of the learned heatmaps further illustrates these behaviors: successful predictions align with sharply localized activations, while failure cases exhibit diffuse or misaligned responses, often at occluded or ambiguous joints. For example, correct predictions show close alignment with ground-truth keypoints, whereas errors highlight typical confusion patterns or difficult poses.

Overall, the results emphasize the advantages of heatmap-based keypoint models in terms of accuracy, interpretability, and robustness under challenging conditions.

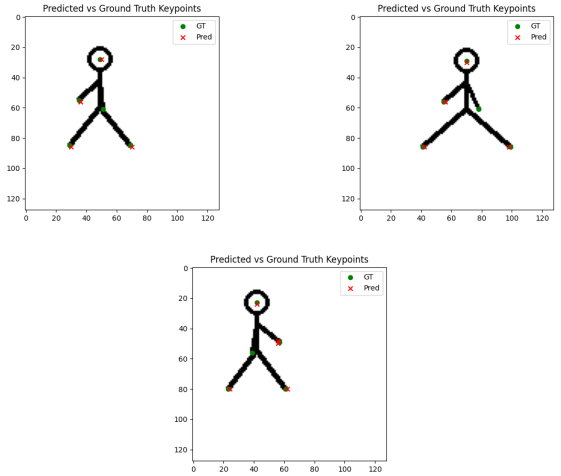

Figure 5: Sample Predictions

C. Ablation Studies

Acknowledgment

References

[1] OpenAI, ChatGPT (Sept. 2025 version). [Online]. Available: https://chat.openai.com/

[2] Perplexity AI, Perplexity AI. [Online]. Available: https://www.perplexity.ai/