# Few-Shot Object Detection with Meta-Learning: A Comparative Study

1. Introduction

In this project, I implemented and compared two object detection models within a few-shot learning scenario using the PASCAL VOC dataset. The primary goal was to explore the benefits of applying meta-learning techniques to object detection when only limited labeled data is available.

Specifically, I trained:

A baseline model using traditional fine-tuning and a linear classification head

A meta-learning-inspired variant using a cosine similarity head to improve generalization

This comparison sheds light on how feature normalization and metric-based learning enhance performance in data-scarce conditions an increasingly common challenge in real-world AI systems.

## Architecture

Backbone (ResNet-50 + FPN): extracts feature maps from input images

Region Proposal Network (RPN): generates object proposals

ROI Head: classifies regions and refines box coordinates

During few-shot fine-tuning, we freeze the backbone and RPN, and train only the classification and box regression heads.

### Baseline Detector

Uses a standard linear classifier head (dot-product followed by softmax)

Trained with SGD on 1-shot examples

No meta-learning involved

### Meta-Style Detector

Replaces the linear head with a cosine similarity head

Encourages generalization to novel classes by comparing ROI features to class prototypes

Inspired by TFA (ICML 2020) ablation and meta-learning literature

## 2. Motivation

Few-shot learning simulates realistic constraints where collecting and annotating large datasets is impractical or impossible. Meta-learning is one solution that helps models adapt quickly to new tasks with minimal supervision.

By comparing both approaches, this project aims to understand:

How well a standard detector generalizes under 1-shot supervision

Whether introducing a metric-learning head can lead to measurable gains

## 3. Dataset Setup

The models were trained and evaluated using the PASCAL VOC 2007 and 2012 datasets. Here's how the data was prepared:

Training set: Generated from VOC trainval (2007 + 2012) using only 1 image per class (1-shot)

Test set: Standard VOC 2007 test split

## 4. Methodology

4.1 Baseline Detector

Architecture: Faster R-CNN with ResNet-50-FPN

Head: Standard linear classifier on ROI features

Training strategy: Fine-tuning the classifier head while freezing the backbone

### 4.2 Meta-Learning Variant

Architecture: Same base (Faster R-CNN + ResNet-50-FPN)

Head: Cosine similarity classifier with feature normalization

Why cosine? This approach computes classification logits based on cosine similarity between ROI features and class weights  a common technique in few-shot literature (e.g. TFA, DeFRCN). It improves generalization by aligning features in embedding space.

### 4.3 Training

Both models trained for ~800 iterations with a batch size of 16

Learning rate: 0.02 (higher than default, since only classifier head is trained)

Input augmentations: Resizing with variable short side

## 5. Evaluation

Models were evaluated using mean Average Precision at IoU = 0.5 (mAP@50), both per class and overall.

### 5.1 Metrics (VOC 1-shot, Split 1)

| Model | Novel AP@50 |
|---|---|
| Baseline | 56.5% |
| Cosine Head | 64.2% |

The cosine similarity head consistently outperformed the baseline, especially on novel classes where data is sparse. This confirms the hypothesis that meta-learning-based strategies are beneficial in low-data regimes.

## 6. Visual Results

To illustrate the difference qualitatively, I ran both models on the same unseen test image.

Baseline Prediction  Cosine Head Prediction

The cosine model showed tighter bounding boxes and fewer false positives, especially for small or overlapping objects.

## 7. Code Design & Clean Submission

To ensure clarity and reproducibility:

I trimmed the FsDet codebase to include only the necessary components (fsdet/, tools/, datasets/, demo/)

I removed all large files (.pth, VOC images, logs)

All scripts  are provided

Evaluation results are provided

Visualizations are provided

## 8. Conclusions

This project demonstrates that even simple meta-learning techniques such as replacing the linear classifier with a cosine similarity head  can yield significant gains in few-shot object detection.

When annotation is expensive or data is limited, integrating metric-based learning into detection architectures becomes a practical and powerful tool.

## 9. Future Work

With more time, I would explore:

Multi-shot settings (5-shot, 10-shot)

Deeper meta-learning techniques like support-query episodic training

Applying this framework to other datasets (e.g., COCO few-shot, LVIS)

### Appendix

Framework: PyTorch 2.2.2, Detectron2 0.6

GPU: NVIDIA RTX 3090, CUDA 12.1

Reference base: FsDet (TFA, ICML 2020)