# Street Parking Presence Inference from Street-Level Imagery via Multi-Cue Detection and Geo-Aggregation

Chirag Jain (9087168606) Ritik Singh (9087047321)
University of Wisconsin-Madison

April 2026

**Note on writing assistance.** This report was prepared with the assistance of Claude/ChatGPT for drafting, restructuring, and language refinement. All content, results, experimental details, and observations were provided by the authors and manually reviewed and verified.

## 1 Introduction

This project studies **street parking presence inference from street-level imagery**. The core idea is to detect image-level parking-related cues such as parking signs, parking meters, and curb-related evidence, and then aggregate these cues across nearby views to infer whether a street segment supports on-street parking. This framing is motivated by a practical difficulty of the task: parking cues are often **small, sparse, occluded, and viewpoint-dependent**, so single-image predictions are brittle. A negative result from one image may simply mean that the relevant cue is outside the field of view or too small to be detected reliably.

Our experiments focus on three visual cues:

- **Parking signs**, modeled as a supervised object detection task on MTSD.
- **Parking meters**, evaluated in a zero-shot cross-dataset setting using COCO-pretrained detection on Mapillary Vistas.
- **Curb structure and curb color**, modeled through binary curb segmentation followed by heuristic color analysis.

Among these, the parking-sign detector emerges as the strongest explicit cue. Parking meters provide some transferable signal but are noisy. Curb color is visually meaningful in some cases, but it is sparse and difficult to recover robustly, especially when segmentation masks are thin or contaminated by nearby road markings.

The final project result is not just a set of independent detectors. The detectors are used as image-level components inside a segment-level aggregation system. This is important because the true downstream task is closer to: *does this local street segment contain parking-related evidence?* rather than: *does this single image contain a visible sign?* The aggregation experiments show that pooling evidence across multiple views substantially improves recall compared with single-image inference, while retaining useful precision.

The overall result is a multi-cue system in which:

- parking signs provide the strongest explicit parking evidence,
- parking meters provide a weak but non-random auxiliary signal,
- curb segmentation is feasible,
- curb color is possible to estimate conservatively, but should be treated as a sparse auxiliary cue rather than a primary standalone signal,
- segment-level aggregation substantially improves recovery of parking-related evidence when cues are sparse or only visible from some viewpoints.

# 2 Datasets and Label Mapping

## 2.1 Parking Sign Dataset Construction

We selected the **Mapillary Traffic Sign Dataset (MTSD)** as the main dataset for the parking-sign cue. MTSD provides a large number of traffic sign annotations, but it is not directly packaged for parking inference. We therefore constructed a binary parking-sign detection dataset by mapping multiple parking-related sign variants (`information-parking-g1, information-parking-g5`) into a single class called `parking_sign`.

The final mapping includes multiple no-parking, no-stopping, parking information, tow-away, and parking-restriction sign variants. We excluded the generic `other-sign` label because it has no consistent visual identity and would introduce substantial noise. We also filtered out annotations marked as ambiguous, occluded, out-of-frame, or dummy.

After preprocessing, our local MTSD setup contained:

- 52,453 images on disk
- 41,909 annotation files
- split files listing 36,589 train, 5,320 validation, and 10,544 test image IDs

In practice, our experiments rely on the **validation split** because the available labeled annotations correspond to train/validation data. The test split IDs exist in the released split files, but it is not directly usable for our current local evaluation pipeline because of unavailable annotations.

## 2.2 Exploratory Data Analysis

We ran an exploratory analysis script to better understand class balance and object scale. The most important findings are:

- Total raw labeled objects: 206,386
- Total clean objects after filtering: 63,806
- Positive images after filtering: 4,446 ($\approx 10.6\%$)
- Negative images after filtering: 24,390
- Effective negative-to-positive ratio: $\approx 5.5 : 1$
- Total positive parking-sign instances: 5,983

- Tiny signs ($< 0.1\%$ of image area): 48,317 ($\approx 75.7\%$)
- Median relative object area: 0.031% of the image

These statistics show that the challenge is not only class imbalance, but also that most signs are **very small objects** in large street-view images.

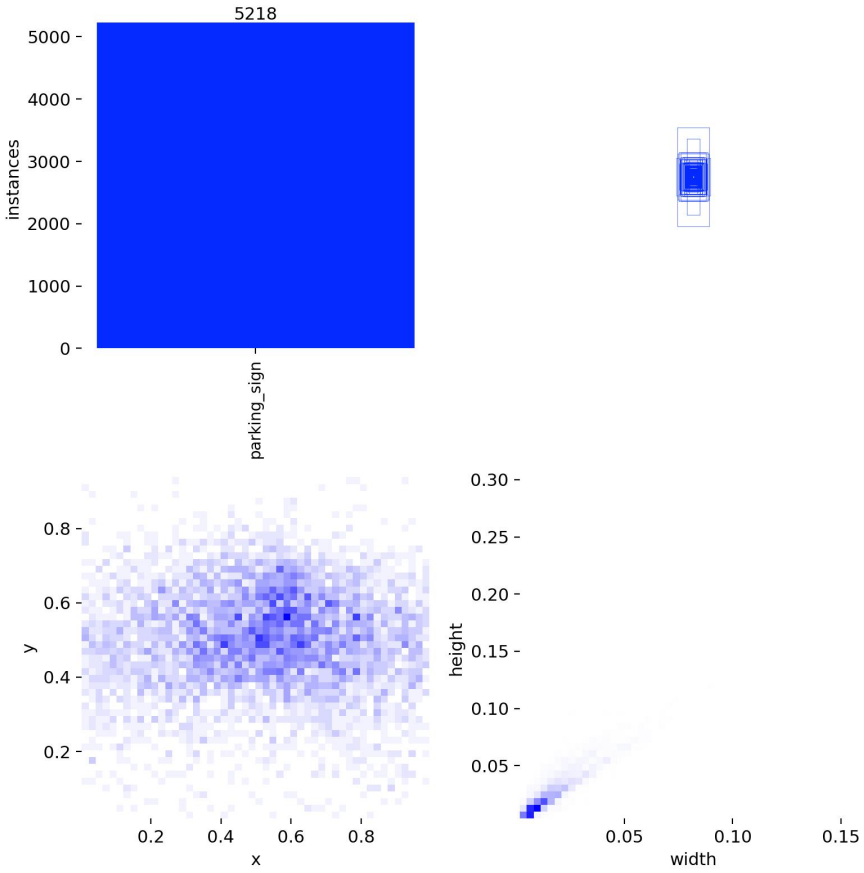



Figure 1: Label distribution and bounding-box size statistics from the processed parking-sign dataset. Most parking signs are very small relative to the image, which makes the task a difficult small-object detection problem.

We also analyzed **Mapillary Vistas v2.0** to understand what parking-related cues are actually available beyond traffic signs. Vistas contains 123 unique classes, including explicit parking-related objects such as:

- `object-parking-meter` (839 instances),
- `object-traffic-sign-information-parking` (3,418 instances),
- `construction-flat-parking` (3,355 instances),
- `construction-flat-parking-aisle` (247 instances).

At the same time, the dataset contains much richer curb-related structure:

- `construction-barrier-curb` (59,767 instances),
- `construction-flat-curb-cut` (17,582 instances).

This led to an important insight: **explicit parking objects in Vistas are relatively sparse, while curb-related scene structure is much more abundant**. This makes curb modeling more attractive than parking-meter modeling as a non-sign cue, although curb color remains harder to interpret reliably.

# 3 Methodology

## 3.1 Parking Sign Detection

We implemented a full training pipeline using YOLOv8m. The model was trained with the following main configuration:

- Model: YOLOv8m
- Image size: 640
- Epochs: 50
- Batch size: 32
- Class loss weight: 2.0
- Augmentations: mosaic, mixup, copy-paste

Training was completed in two phases because the Kaggle session crashed after epoch 24 and had to be resumed. Even though the process was interrupted, the final training curves were smooth and consistent, suggesting that the resumed training behaved as expected.

From Figure 2, several trends are clear:

- Training and validation losses decrease steadily.
- Precision stabilizes in the mid-0.6 range.
- Recall improves more gradually, reflecting the difficulty of recovering small distant signs.
- mAP@50 and mAP@50–95 continue improving through the later epochs, although with diminishing returns near the end.

This suggests that the model is learning useful signal and not obviously overfitting, but that the task remains challenging.

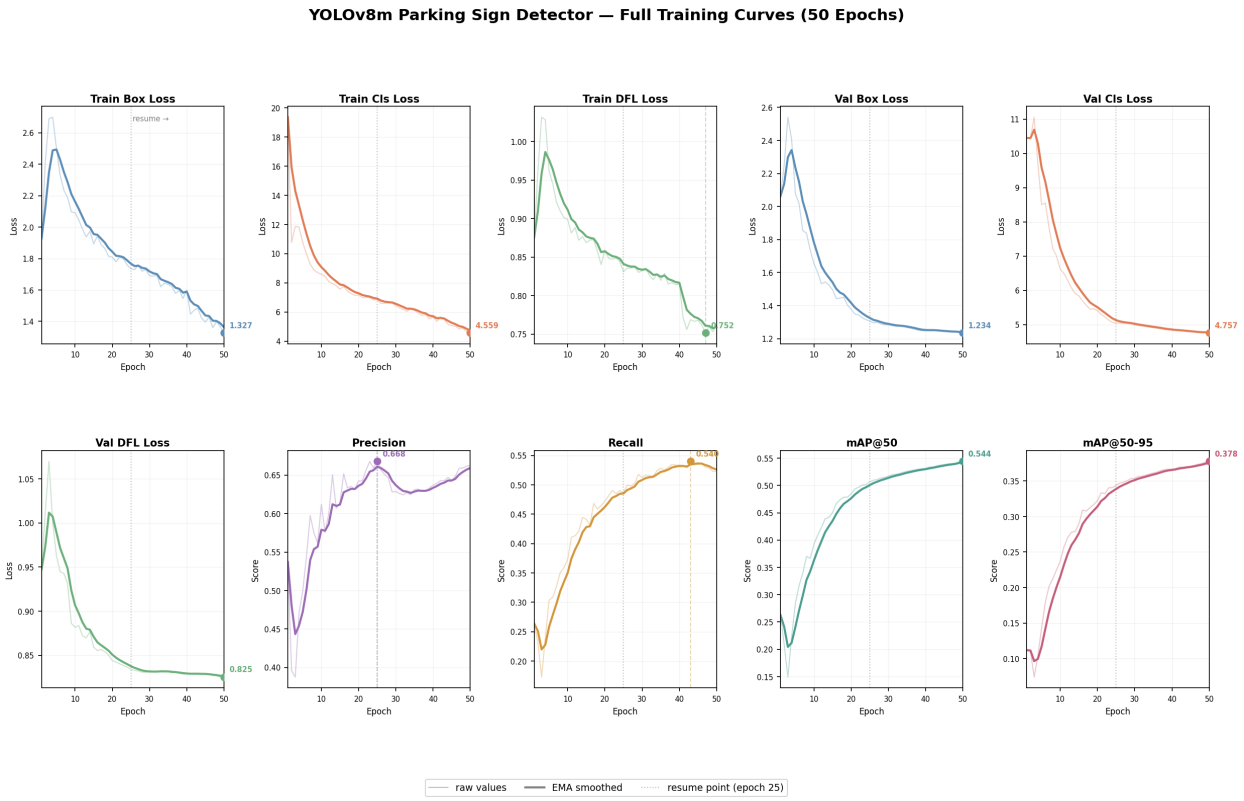



Figure 2: Combined training and validation curves over 50 epochs. The run was completed in two stages due to infrastructure interruptions, but the losses and validation metrics show stable convergence.

## 3.2 Image-Level Evaluation

A major methodological decision was to build a custom **image-level evaluation script** in addition to standard YOLO validation. This was necessary because the downstream task is not only bounding-box localization; we ultimately care about whether an image contains parking-related evidence that can later be aggregated over a street segment.

Our evaluation therefore has two parts:

1. **Box-level evaluation:** standard YOLO detection metrics such as mAP, precision, and recall.
2. **Image-level evaluation:** for each image, we take the maximum detection confidence across all predicted boxes and convert it into a binary parking-presence score. This allows us to compute precision, recall, F1, and AUROC at the image level.

This image-level evaluation aligns much more closely with the eventual segment-level inference objective.

## 3.3 Parking Meter Detection

For parking meters, we used a **zero-shot cross-dataset evaluation**. We took a COCO-pretrained detector, restricted it to the parking-meter class, and evaluated it against Mapillary Vistas parking-meter annotations. Ground-truth parking-meter polygons were converted into bounding boxes so that we could compute both image-level and box-level metrics.

### 3.4 Curb Segmentation and Color Analysis

Because curb-related structure is abundant in Mapillary Vistas, we explored curb as an additional non-sign cue. Unlike parking signs and meters, curbs are naturally better represented as a **segmentation** problem rather than object detection, since they are long, thin, and irregular structures rather than compact objects.

We therefore formulated curb detection as a **binary semantic segmentation problem** using the Mapillary Vistas class `construction-barrier-curb` as foreground and all other pixels as background. We trained a U-Net-based segmentation model and found that it can localize curb regions reasonably well, but the predicted masks are often **thin and fragmented**, especially when curbs are distant, visually smooth, or weakly separated from the road surface.

After obtaining curb masks, we implemented a second stage for **curb color analysis**. The goal of this stage was not to recover exact municipal parking rules, but rather to extract coarse visual categories such as red, yellow, green, white, gray, and unknown. In practice, this turned out to be substantially harder than mask extraction itself, because color estimation is highly sensitive to **mask purity**. If non-curb pixels such as crosswalk markings, lane paint, or asphalt are included in the mask, the resulting color distribution becomes ambiguous.

To reduce this contamination, we refined the color pipeline in two ways:

- We restricted color extraction to **boundary pixels** derived from the predicted mask, since the curb itself is most naturally expressed as a thin boundary rather than a filled region.
- We introduced a **confidence-margin rule**: a color prediction is accepted only if the top color score is both sufficiently large and sufficiently separated from the second-best score. Otherwise, the result is labeled as `unknown`.

This makes the curb color output intentionally conservative, which is preferable for downstream cue fusion and easier to justify in qualitative analysis.

### 3.5 Segment-Level Aggregation Methodology

The final stage of the system aggregates image-level cue predictions into a segment-level score. This section is included in methodology, rather than only in results, because aggregation changes the unit of inference. The earlier detectors answer: *what cue is visible in this image?* The aggregation stage answers: *does this local segment contain enough parking-related evidence across multiple views?*

For each image $i$ in a five-image segment, we record three model-derived scores:

- $s_i$: maximum parking-sign confidence in the image,
- $m_i$: maximum parking-meter confidence in the image,
- $c_i$: positive curb-color score, defined as the maximum of the yellow, green, and white curb-color scores. Red is excluded because red curbs usually indicate no parking. In a more complete rule-aware system, red could be modeled as negative evidence, but in this project we exclude it from the positive curb score to keep the aggregation rule simple and conservative.

We experimented with max, mean, count-above-threshold, and noisy-OR features. The primary reported aggregation rule uses a weighted max score:

$$S_{segment} = \max\left(\max_i s_i,\ 0.6\max_i m_i,\ 0.4\max_i c_i\right).$$

The choice of max pooling is motivated by the multiple-instance nature of the task. A street segment can be positive even if only one of its views contains a visible parking sign or meter. Averaging across views would dilute exactly the sparse evidence we are trying to recover. Max pooling instead asks whether any view contains strong evidence.

The weights are simple reliability-based heuristics rather than learned parameters. Parking signs are left unscaled because they are the most direct and best-performing cue. Parking meters are informative but noisy in the zero-shot setting, so their contribution is down-weighted to 0.6. Curb color is more indirect and depends heavily on segmentation quality, so it is down-weighted to 0.4. The purpose of this weighting is not to optimize a learned fusion model, but to encode the relative trustworthiness observed in the preceding experiments: sign evidence is primary, meter evidence is useful but weaker, and curb color is auxiliary.

# 4 Quantitative Results

## 4.1 Parking Sign Detection Results

Our final validation results from the best checkpoint after 50 epochs are shown in Table 1.

Table 1: Main validation results for the parking sign detector.

| **Metric** | **Value** |
|---|---|
| mAP@50 | 0.5487 |
| mAP@50–95 | 0.3824 |
| Precision (box-level) | 0.6616 |
| Recall (box-level) | 0.5373 |
| Best image-level threshold | 0.15 |
| Best image-level F1 | 0.6673 |
| Image-level AUROC | 0.8310 |

The box-level metrics show that the model is learning to localize parking signs reasonably well, while the image-level F1 and AUROC show that it can separate positive and negative images with useful reliability.

## 4.2 Threshold Analysis

We also swept the confidence threshold to study the precision-recall trade-off at the image level. Selected results are shown in Table 2.

Table 2: Image-level threshold sweep on the validation set.

| **Threshold** | **Precision** | **Recall** | **F1** | **Accuracy** |
|---|---|---|---|---|
| 0.15 | 0.6913 | 0.6448 | **0.6673** | 0.9299 |
| 0.20 | 0.7328 | 0.6052 | 0.6629 | 0.9329 |
| 0.30 | 0.7565 | 0.5517 | 0.6381 | 0.9318 |
| 0.50 | 0.8262 | 0.4672 | 0.5969 | 0.9312 |
| 0.70 | 0.9150 | 0.3155 | 0.4692 | 0.9222 |

The best operating threshold is around 0.15. This is relatively low, which indicates that many useful detections are not extremely high-confidence. If a very strict confidence threshold is used, recall drops sharply and many parking cues are lost. This is exactly the kind of behavior that makes aggregation valuable: several weak or partial detections across nearby images may still provide strong segment-level evidence.

### 4.3 Training Progress Over Time

To show intermediate progress clearly, Table 3 summarizes key checkpoints during training.

Table 3: Progression of validation performance across checkpoints.

| Checkpoint | mAP@50 | Image-level F1 | AUROC |
|---|---|---|---|
| Epoch 20 | 0.4835 | 0.6430 | 0.8088 |
| Epoch 25 | 0.5065 | 0.6535 | 0.8226 |
| Epoch 50 | **0.5487** | **0.6673** | **0.8310** |

The model improved steadily throughout training. However, the gains from epoch 25 to epoch 50 were smaller than the gains earlier in training, suggesting that the detector is approaching a plateau. This indicates that future gains are more likely to come from aggregation and additional cues than from extensive further tuning of the sign detector alone.

### 4.4 Zero-Shot Parking-Meter Results

In addition to the trained parking-sign baseline, we performed a preliminary zero-shot experiment for parking-meter detection. We used a COCO-pretrained YOLO11x model and evaluated it on Mapillary Vistas validation images, using `object-parking-meter` as ground truth. The full validation set contained:

- 2,000 validation images,
- 50 positive images containing at least one parking meter,
- 1,950 negative images.

Before evaluating on the full validation set, we first ran a positives-only sweep to understand the effect of inference resolution and confidence threshold. That sweep showed that larger input resolution substantially improves recall, which is consistent with the qualitative observation that parking meters are often very small in street-view imagery. Based on this analysis, we selected `imgsz=1280` for the full run.

The final full-validation results are shown in Table 4.

Table 4: Zero-shot parking-meter evaluation on Mapillary Vistas validation set using COCO-pretrained YOLO11x.

| imgsz | conf | Img P | Img R | Img F1 | Box P | Box R | Box F1 |
|---|---|---|---|---|---|---|---|
| 1280 | 0.05 | 0.109 | 0.520 | 0.181 | 0.0469 | 0.168 | 0.0734 |
| 1280 | 0.10 | **0.158** | 0.480 | **0.238** | **0.0758** | 0.158 | **0.1020** |

For the better setting, `imgsz=1280, conf=0.10`, the detailed counts were:

- image-level: TP = 24, FP = 128, TN = 1822, FN = 26
- box-level: TP = 15, FP = 183, FN = 80

These results lead to three main conclusions:

- **Zero-shot transfer is real but limited.** The detector recovers nearly half of positive parking-meter images at the image level, so it is not behaving randomly.
- **Precision is poor.** The detector produces many false positives, especially on pole-like objects and other narrow vertical street furniture.
- **Parking meters are therefore a weak auxiliary cue.** They may still contribute useful evidence in a multi-cue system, but they are not reliable enough to serve as a primary cue by themselves.

Between the two settings, `conf=0.10` gives the better overall trade-off because it substantially reduces false positives while only slightly reducing recall. This makes it the more reasonable operating point if parking-meter scores are later incorporated into cue fusion.

### 4.5 Curb Segmentation and Color Results

We trained a curb segmentation model on Mapillary Vistas using `construction-barrier-curb` as foreground in a binary segmentation setting. The model was trained for **20 epochs**, and the best validation checkpoint occurred at **epoch 17**, where it achieved a validation Dice score of **0.5184** and a validation IoU of **0.4238**. The final epoch produced a validation Dice score of **0.5047**, which suggests that the model had largely stabilized and that later epochs provided only marginal gains.

The training dynamics are shown in Figure 3, Figure 4, and Figure 5. Several trends are visible:

- Training and validation losses decrease rapidly in the earlier epochs and then fluctuate within a narrower range.
- Validation Dice and IoU improve substantially during the first half of training and then begin to plateau.
- The best validation checkpoint occurs before the final epoch, indicating that additional training does not fundamentally change the learned representation.

This behavior matches the qualitative observations: the model learns to localize curb boundaries reasonably well, but the remaining errors are largely due to the intrinsic difficulty of thin-structure segmentation rather than undertraining.

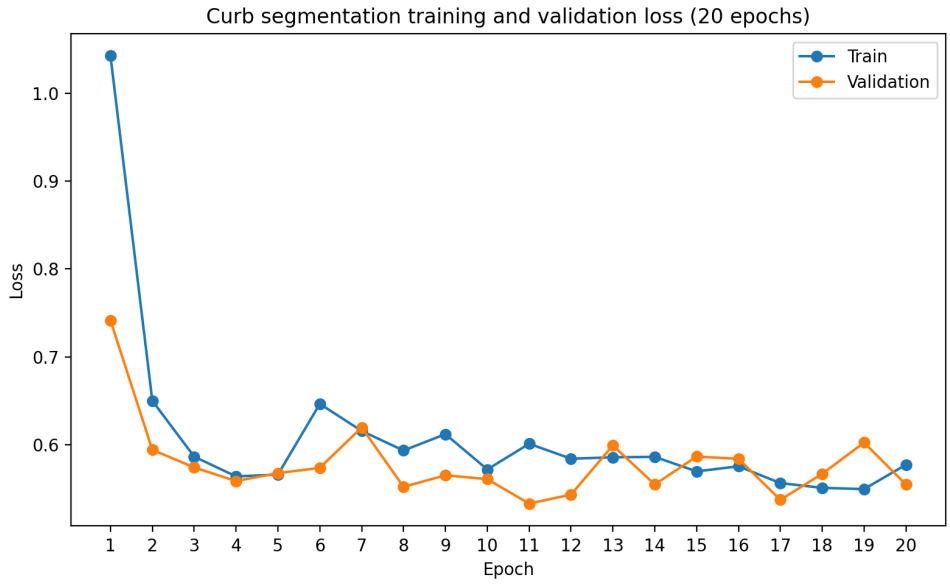



Figure 3: Training and validation loss for the curb segmentation model over 20 epochs.

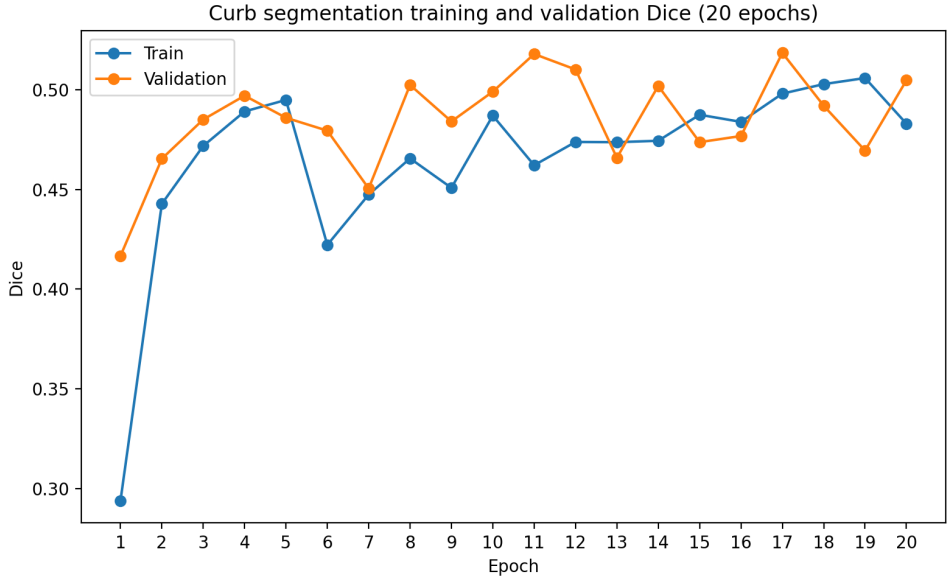



Figure 4: Training and validation Dice score for the curb segmentation model over 20 epochs. The best validation Dice occurs at epoch 17.

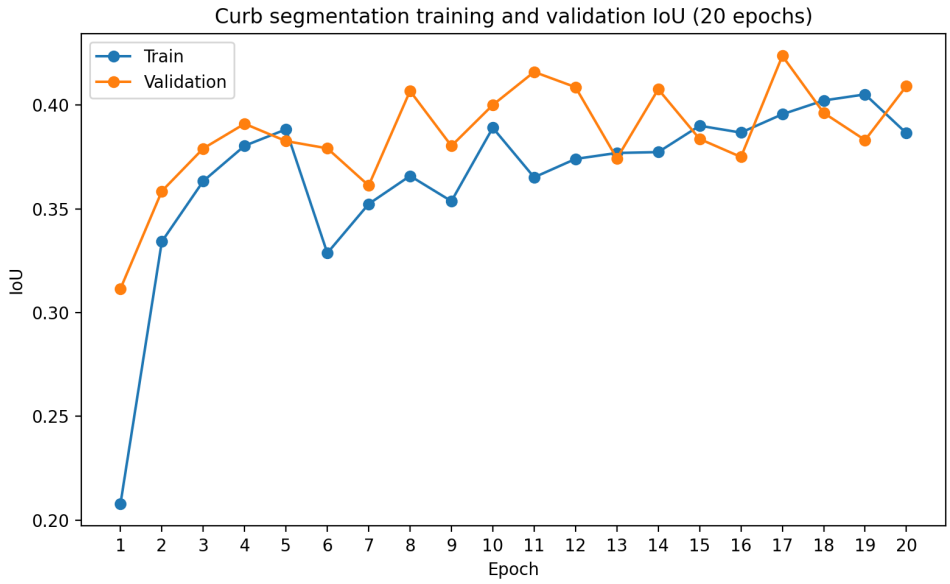



Figure 5: Training and validation IoU for the curb segmentation model over 20 epochs.

Using the curb segmentation output, we then ran curb color inference on the **2,000-image validation split**. The segmentation model was first used to predict curb masks, and then an HSV-based color analysis stage was applied to the predicted curb boundaries. The final color assignment was restricted to the set {`red`, `yellow`, `green`, `white`, `gray`, `unknown`}, with a conservative fallback to `unknown` whenever the evidence was weak or ambiguous.

The final validation-set distribution of dominant curb-color predictions is shown in Table 5.

Table 5: Validation-set distribution of dominant curb-color predictions from the curb analysis pipeline.

| **Dominant color** | **Count** | **Fraction** |
|---|---|---|
| Unknown | 937 | 46.85% |
| Gray | 835 | 41.75% |
| Yellow | 190 | 9.50% |
| Green | 21 | 1.05% |
| White | 13 | 0.65% |
| Red | 4 | 0.20% |

These numbers support three main conclusions. First, the curb segmentation model recovers enough structure for downstream color analysis on a substantial fraction of images, even though the masks are often thin and fragmented. Second, **strong painted curb colors are relatively sparse**, while gray and unknown dominate, which is consistent with the fact that many curbs are unpainted, weakly painted, or visually ambiguous. Third, conservative uncertainty handling is necessary: forcing a hard color prediction in every case would introduce many incorrect labels.

An earlier version of the color pipeline used all predicted mask pixels for color extraction. That approach produced noticeably more contamination from nearby white road markings and crosswalks. After switching to boundary-based color extraction, the number of white predictions dropped substantially while gray became more common, which suggests that the revised pipeline

is more accurately capturing curb surface color rather than nearby painted road elements. The resulting system therefore trades some recall for better precision and interpretability.

### 4.6 Segment-Level Synthetic Aggregation Results

The segment-level aggregation experiment is the main system-level result of the project. The synthetic pseudo-segment benchmark contains 225 five-image segments: 80 negative segments and 145 positive segments. Positive segments were constructed to include at least one parking-related cue, while negative segments contained no selected strong cue. This construction tests whether a segment-level rule can recover sparse evidence distributed across multiple views.

The final cue-pool sizes used to build this dataset were:

- sign-positive images: 4,446,
- meter-positive images: 50,
- strong curb-color images: 228,
- none/neutral images: 41,185.

The meter pool is especially small. This directly shaped the final segment distribution: we reduced meter-heavy segment types to avoid reusing the same meter examples too frequently. This is important because repeated reuse would make the evaluation less diverse and would overstate the amount of meter evidence available in the data.

Table 6: Final synthetic pseudo-segment distribution. Each segment contains five images. The skewed distribution reflects cue sparsity, with more negative and sign-only segments and fewer meter-heavy combinations.

| **Segment type** | **Number of segments** | **Segment label** |
|---|---|---|
| None | 80 | 0 |
| Sign only | 50 | 1 |
| Meter only | 15 | 1 |
| Curb color only | 25 | 1 |
| Sign + meter | 15 | 1 |
| Sign + curb color | 20 | 1 |
| Meter + curb color | 10 | 1 |
| Sign + meter + curb color | 10 | 1 |
| **Total** | **225** | – |

The synthetic dataset should be interpreted with the correct scope. It does not claim that randomly combined images are actual road-neighboring views. Rather, it is a controlled multiple-instance benchmark for the aggregation mechanism. This is still useful because it isolates a key property of the task: a segment may be positive even if only one of several views contains visible evidence.

We compare aggregation against a fair single-image per-segment baseline. The baseline uses one selected image from each segment and applies only the sign score. This simulates a single-view deployment setting. The aggregation method uses all five images and combines sign, meter, and curb evidence using the weighted max rule described earlier.

Table 7: Synthetic pseudo-segment aggregation results. Aggregation substantially improves recall and F1 compared with a single-image per-segment baseline.

| Method | Threshold | Precision | Recall | F1 |
|---|---|---|---|---|
| Single-image baseline | 0.05 | 0.784 | 0.200 | 0.319 |
| Segment aggregation | 0.15 | 0.753 | 0.924 | **0.830** |

The result is strong and directly supports the project hypothesis. The single-image baseline has high precision but extremely low recall: when it detects a sign, the prediction is often meaningful, but it misses most positive segments because the selected view often lacks visible evidence. Aggregation makes the opposite trade-off. Precision drops only slightly, from 0.784 to 0.753, but recall increases from 0.200 to 0.924. This means aggregation is mainly solving the false-negative problem caused by sparse cue visibility. Precision decreases slightly because aggregation treats any strong cue from any view as sufficient evidence, which increases true positives but also introduces some false positives from noisy auxiliary cues such as meters and curb color.

A threshold sweep of the aggregated score is shown in Table 8. The best F1 occurs at threshold 0.15, the same general operating region as the image-level sign detector. This reinforces the observation that low-confidence detections should not be discarded too aggressively. In this task, weak cues can become reliable when pooled over several views.

Table 8: Threshold sweep for the synthetic segment-level aggregation score.

| Threshold | Precision | Recall | F1 | AUROC |
|---|---|---|---|---|
| 0.05 | 0.671 | 0.972 | 0.794 | 0.820 |
| 0.10 | 0.711 | 0.966 | 0.819 | 0.820 |
| 0.15 | 0.753 | 0.924 | **0.830** | 0.820 |
| 0.20 | 0.774 | 0.876 | 0.822 | 0.820 |
| 0.25 | 0.800 | 0.800 | 0.800 | 0.820 |
| 0.30 | 0.823 | 0.738 | 0.778 | 0.820 |
| 0.40 | 0.878 | 0.545 | 0.672 | 0.820 |
| 0.50 | 0.946 | 0.483 | 0.639 | 0.820 |

The practical interpretation is that segment-level aggregation is not merely a post-processing trick. It changes the operating regime of the system. Single-image inference asks each view to independently contain enough evidence. Aggregation lets the system use the best available evidence from the local context. This is much closer to the real-world structure of the task.

### 4.7 Manual Real-World Segment Validation

To test whether the same behavior appears outside the synthetic pseudo-segment setting, we manually collected a small real-world dataset of six street segments. Each segment contains five nearby views from the same local curbside context, collected using Google Maps/Street View-style imagery. The dataset contains only 30 images, so it is not intended as a statistically meaningful benchmark. Its purpose is qualitative validation: to check whether the aggregation logic behaves sensibly on real geographically coherent examples.

Manual construction is expensive for this project because each segment requires several decisions: identifying a street with visible parking-related evidence, moving through nearby views,

saving images consistently, ensuring that the views correspond to the same local curbside context, and writing notes about which cues are visible. We therefore collected six examples covering different cue configurations rather than attempting to create a large manual benchmark.

Table 9: Manually collected real-world segment examples. These examples are used for qualitative validation rather than as a large-scale benchmark.

| **Segment** | **Label** | **Observed cue pattern** |
|---|---|---|
| seg_000 | Positive | Multiple parking signs visible across the five views, including a faint sign in one view. This tests whether aggregation increases confidence when the sign detector already has evidence. |
| seg_001 | Positive | Meter-only segment. Parking meters are visible in a subset of views, while sign evidence is absent. This tests whether the meter cue can rescue a segment missed by a sign-only baseline. |
| seg_002 | Positive | Mixed cue segment containing a parking meter, parking sign evidence, and yellow curb color. This tests whether heterogeneous cues can support the same segment-level decision. |
| seg_003 | Positive | Yellow curb-color-only segment. This is a weak-cue case because there is no strong sign or meter evidence. |
| seg_004 | Positive | Yellow curb color plus one visible parking meter. This tests complementary cue fusion between meter and curb evidence. |
| seg_005 | Positive | Difficult failure case. The visible sign is not part of the detector's training distribution and resembles a non-standard/storefront sign more than the mapped MTSD parking-sign classes. |

The manual results are shown in Table 10. Aggregation correctly identifies five of the six positive examples, while the single-image sign baseline only succeeds on seg_000. This matches the synthetic result qualitatively: aggregation improves recall by allowing evidence to come from any view and from auxiliary cues.

Table 10: Manual real-world segment results. Scores are maximum segment-level cue scores. The combined score uses the same weighted max rule as the synthetic aggregation experiment.

| **Segment** | **Single** | **Sign max** | **Meter max** | **Curb max** | **Combined** | **Pred.** |
|---|---|---|---|---|---|---|
| seg_000 | 0.431 | 0.529 | 0.394 | 0.634 | 0.529 | 1 |
| seg_001 | 0.000 | 0.000 | 0.722 | 0.359 | 0.433 | 1 |
| seg_002 | 0.000 | 0.395 | 0.227 | 0.426 | 0.395 | 1 |
| seg_003 | 0.000 | 0.148 | 0.000 | 0.407 | 0.163 | 1 |
| seg_004 | 0.000 | 0.000 | 0.708 | 0.539 | 0.425 | 1 |
| seg_005 | 0.000 | 0.000 | 0.000 | 0.093 | 0.037 | 0 |

Several observations are important. First, seg_001 is a clean meter-only success case. The sign detector contributes no signal, so a sign-only system would fail completely. The meter detector produces a strong score, and after the 0.6 down-weighting the segment still crosses the threshold. This validates the idea that parking meters are weak as a standalone detector but useful as an auxiliary cue.

Second, seg_003 demonstrates the role and limitation of curb color. It is recovered, but only barely: the combined score is 0.163, just above the 0.15 threshold. This is the desired behavior. Curb color is meaningful but indirect, so it should not dominate the final decision unless there is enough evidence. The low margin reflects uncertainty rather than overconfidence.

Third, seg_004 shows complementary cue fusion. The sign detector contributes no evidence, but the meter and curb modules both produce useful signals. The segment is correctly recovered because aggregation combines these auxiliary cues instead of requiring a sign detection.

Finally, seg_005 is a useful failure case. At first, this looked like an image-quality issue, but closer inspection showed a more meaningful failure mode: the visible sign is outside the detector's training distribution. The system was trained on mapped MTSD parking-regulation signs, but the visible sign resembles a non-standard storefront or local sign. Since no view contains a sign matching the trained detector's visual vocabulary, and because there are no strong meter or curb backup cues, aggregation cannot recover the segment. This illustrates a key limitation: aggregation can compensate for sparse visibility, but it cannot compensate for a detector that lacks the relevant visual concept.

Figures 6–9 show annotated outputs for selected manual segments. Each segment is visualized using multiple nearby views with overlaid detections for parking signs, parking meters, and curb segmentation. These examples highlight how cues are distributed across views and demonstrate how aggregation leverages complementary evidence to recover parking presence even when individual views contain weak or incomplete signals.

(a) View 0

(b) View 1

(c) View 2

(d) View 3

(e) View 4

Figure 6: Manual segment seg_001. This meter-only segment demonstrates the value of auxiliary cues. The sign detector does not fire, but the parking-meter cue is strong enough for segment-level aggregation to correctly classify the segment as positive.

(a) View 0

(b) View 1

(c) View 2

(d) View 3

(e) View 4

Figure 7: Manual segment seg_003. This curb-color-only example is a borderline but successful case. The low margin is appropriate because curb color is an indirect cue and is intentionally down-weighted.

(a) View 0

(b) View 1

(c) View 2

(d) View 3

(e) View 4

Figure 8: Manual segment seg_004. This example combines meter and curb-color evidence. The sign detector contributes no signal, but aggregation succeeds by using complementary auxiliary cues.

(a) View 0

(b) View 1

(c) View 2

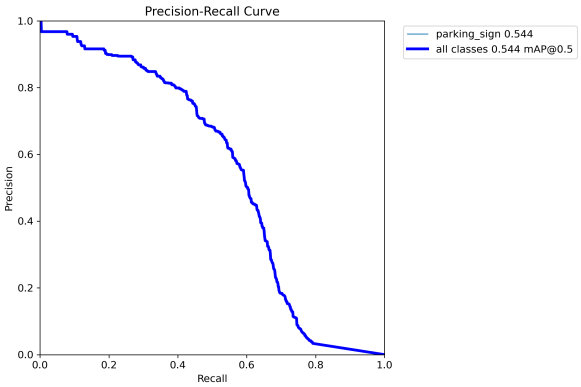
(d) View 3

(e) View 4

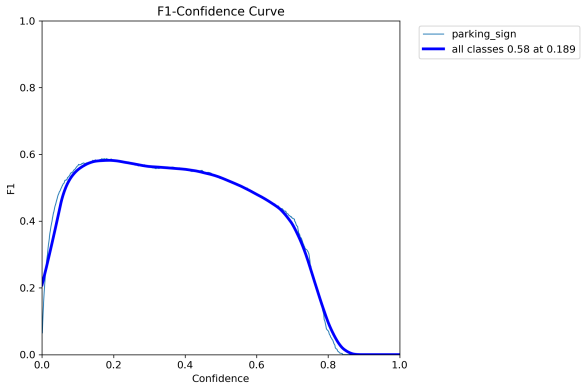
Figure 9: Manual segment seg_005. This failure case shows a limitation of the system: the visible sign is outside the training distribution, and there are no strong auxiliary cues. Aggregation cannot recover missing visual concepts.

## 5 Additional Validation Plots

The standard YOLO validation plots are also informative and are included for completeness.




(a) Precision–recall curve

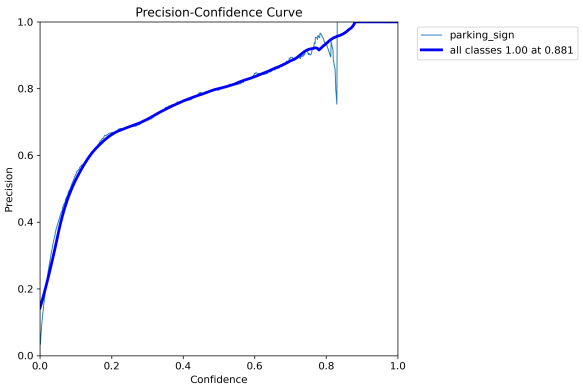



(b) F1–confidence curve

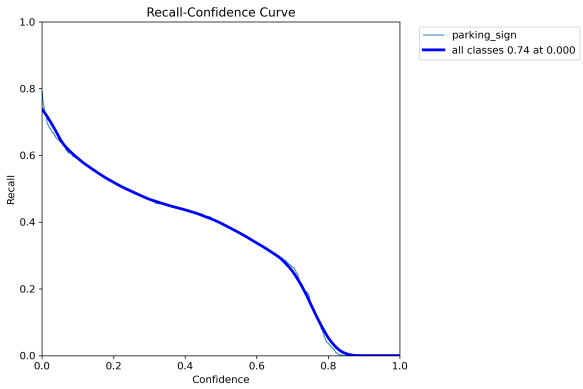
Figure 10: Detection performance curves from YOLO validation.

Figure 10 supports several observations:

- The precision–recall curve confirms moderate but useful detection performance, with mAP@50 around 0.54.
- The F1–confidence curve shows that the detector works best at a relatively low confidence threshold, again indicating that retaining weaker detections is important.




(a) Precision–confidence curve




(b) Recall–confidence curve

Figure 11: Precision and recall behavior as the confidence threshold changes.

Figure 11 shows that precision increases as the threshold becomes stricter, but recall drops rapidly. This is another sign that strict single-image decision rules are not ideal for this problem setting.

# 6 Qualitative Findings and Error Analysis

A useful part of the project is not just the final numbers, but also the qualitative analysis of *why* the models succeed or fail.

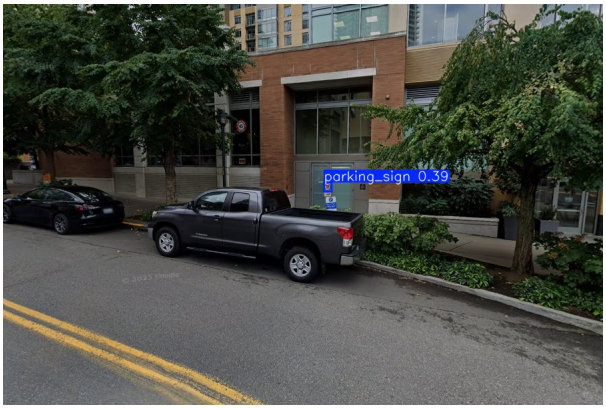
## 6.1 Resolution and Scale Sensitivity for Parking Signs

One of the clearest findings from our experiments is that the parking-sign detector is strongly limited by object scale. We tested the same scene at different effective scales and observed the following behavior:

- Figure 12a shows that at **640** resolution on the zoomed-out image, the model missed both parking signs.
- Figure 12b shows that at **960** resolution on the same zoomed-out image, the model detected one of the signs.
- Figure 12c shows that at **1280** resolution on the same zoomed-out image, the model detected both signs.
- Figure 12d shows that on a manually zoomed-in crop, the model detected the signs more reliably even at lower inference size (imgsz=640).

This is a strong confirmation of the dataset analysis: the detector is **resolution-limited rather than concept-limited**. In other words, the model has learned what parking signs look like, but in wide street scenes the signs are often too small after image resizing for reliable detection.

(a) Original Image at imgsz=640 (miss)




(b) Original Image at imgsz=960 (partial detection)


(c) Original Image at imgsz=1280 (both signs detected)


(d) Original Image Manually Zoomed at imgsz=640 (Full detection)

Figure 12: Detection results for a Seattle image showing the same scene under different effective scales. Detection success improves as the parking signs occupy more pixels in the model input.

This finding directly supports the project motivation. If a single view is zoomed out, the cue may be missed; if another nearby view captures the sign more closely, the cue may become detectable.

### 6.2 Partial Localization of Composite Signs

Another recurring observation is that the detector often boxes only the most salient sub-part of a parking sign assembly, such as the blue "P" symbol or a no-parking icon, rather than the full stacked signboard including time restrictions.

This behavior is not surprising for two reasons:

1. MTSD is a traffic-sign dataset, so annotations are naturally sign-centric rather than designed for full sign-assembly understanding.

2. Text-heavy restriction plates are smaller and more variable than the main symbol panel, making them much harder to learn.

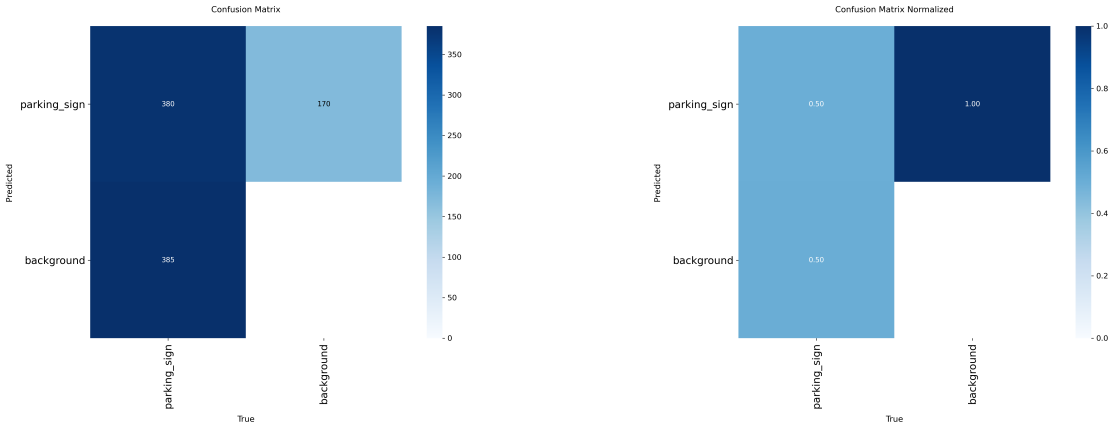
For the present task, this is acceptable because the main requirement is **parking-cue presence detection**, not full OCR-based rule parsing. However, it is an important limitation to state clearly: the detector is suitable for cue detection, but not yet for complete parking-rule understanding.



Figure 13: Qualitative examples from manually collected images. These illustrate successful detections and also the tendency to localize the most visually salient sub-sign rather than the full sign assembly (for example, the image on the right where only the P sign is detected).

### 6.3 Confusion Matrix Interpretation

We also inspected the validation confusion matrices produced by YOLO. They show that the detector still produces a meaningful number of false negatives, which is consistent with the moderate recall values reported earlier. This again suggests that future gains are unlikely to come only from tightening the detector threshold; instead, combining evidence across views is a more promising direction.




(a) Raw confusion matrix

(b) Normalized confusion matrix

Figure 14: Confusion matrices from YOLO validation.

### 6.4 Resolution Sensitivity and Object Scale Analysis

During qualitative evaluation, we also observed a non-intuitive behavior related to input image resolution. In certain cases, reducing the input resolution (for example, from 640 to 160) improved the detection of distant parking signs.

**Observation.** Some parking signs that were not detected at higher resolutions became detectable when the input resolution was reduced. An example is shown in Figure 15, where the same image produces different detection results depending on the input resolution.

**Explanation.** Object detection models such as YOLO operate on fixed-size inputs and rely on hierarchical feature maps to detect objects at different scales. Detection performance is therefore strongly dependent on the relative size of the object within the resized image.

At higher resolutions, distant parking signs occupy only a small number of pixels relative to the full image. After multiple downsampling operations in the network, these objects become extremely small in deeper feature maps, making them difficult to detect. In contrast, when the input image is resized to a smaller resolution, the background is compressed and the object occupies a larger relative portion of the image. This effectively increases the prominence of the object in the feature maps, making detection easier.

**Dataset-Induced Bias.** This behavior is consistent with our dataset analysis. As shown in Figure 16, many parking signs in the training data are small and low-resolution, particularly in wide street-view images. Consequently, the model learns to detect parking signs at specific scales that are most common in the dataset.

**Implications.** These observations highlight that the model is not inherently learning "small images," but rather learning to detect objects at specific relative scales. This has several practical implications:

- Detection performance is sensitive to object scale and image resolution.
- Text-heavy signs are particularly affected due to their dependence on fine-grained visual details.


Figure 15: Effect of input resolution on detection. Left: Higher resolution (imgsz=1280) input where the parking sign is missed. Right: Lower resolution (imgsz=160) input where the same sign is detected.

- A single fixed input resolution may not be sufficient for robust detection across all scenarios.

**Practical Considerations.** To address these limitations, several improvements can be considered:

- Using higher-resolution inference (e.g., 960 or 1280) to better capture small objects.
- Applying multi-scale inference to improve robustness across object sizes.
- Using tiled or patch-based inference for better small object detection.
- Incorporating OCR-based methods in future work for better handling of text-heavy parking signs.

Overall, this experiment highlights the importance of object scale in detection performance and provides valuable insight into the limitations of purely visual detection approaches for parking sign understanding.

### 6.5 Qualitative Parking-Meter Findings

The zero-shot parking-meter experiment also produced useful qualitative insight. On close, clear images, the COCO-pretrained detector can correctly identify parking meters. However, the model struggles much more in realistic street-view scenes where parking meters are small relative to the

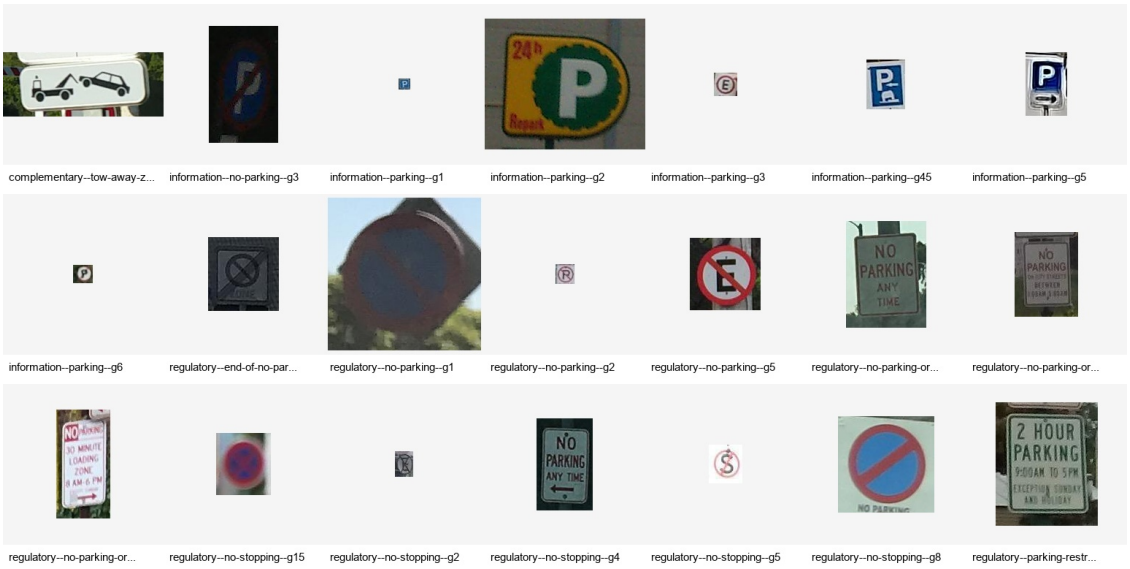



Figure 16: Sample parking sign crops from different categories in the training dataset. The large variation in size, appearance, and text content contributes to scale sensitivity and detection bias.

full image, partially occluded, or visually similar to signposts and other narrow vertical street furniture.

This behavior matches the quantitative results in Table 4. The image-level recall shows that some transferable signal is present, but the low precision indicates that the model frequently mistakes pole-like objects for parking meters. In other words, the detector is not useless, but it is not yet reliable enough to stand on its own.

Taken together, the qualitative meter results match the quantitative findings: parking-meter detection is useful as a **partial cue**, but currently too noisy to be relied on by itself.


Figure 17: Example qualitative parking-meter detection from a manually collected image. The COCO-pretrained YOLO11x model successfully detects the parking meter in a close-view nighttime scene.

### 6.6 Qualitative Curb Segmentation Findings

The curb segmentation model produces a useful but imperfect signal. In many scenes, it correctly identifies the curb boundary and captures enough structure for downstream analysis. However, unlike parking signs or meters, curb regions are long, thin, and often visually weak. As a result, the predicted masks are frequently **sparse and fragmented** rather than dense, continuous regions.

This behavior is not necessarily a failure for our use case. Since the downstream goal is to recover a coarse curb-related cue rather than a pixel-perfect geometric model, even partial mask recovery can be sufficient if the predicted pixels are located on the true curb boundary.

At the same time, the segmentation model exhibits a recurring qualitative failure mode on **painted or low-elevation curbs**. In these cases, the curb can look visually similar to flat road paint or lane markings rather than a structurally distinct boundary. This suggests that the model relies strongly on geometric and shading cues, and is less robust when the curb is visually smooth, uniformly painted, or weakly separated from adjacent surfaces.

Figure 19 shows the painted curb is missed because it lacks strong geometric separation from the road surface. This failure is likely influenced by dataset bias, as many training examples emphasize raised curbs with clear boundary structure. As a result, the model relies heavily on geometric edge cues rather than learning a fully semantic notion of curb appearance. When the curb is flat,

painted, or visually similar to road markings, especially in zoomed-in views where contextual cues are reduced, the model struggles to distinguish it from surrounding surfaces.

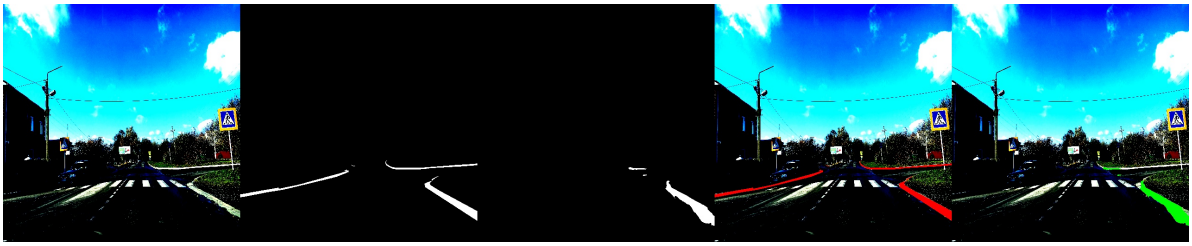


Figure 18: Example qualitative curb segmentation result. The model captures the curb location, but the prediction remains thin and fragmented rather than forming a dense continuous region.


Figure 19: Failure case for curb segmentation.

### 6.7 Qualitative Curb Color Findings

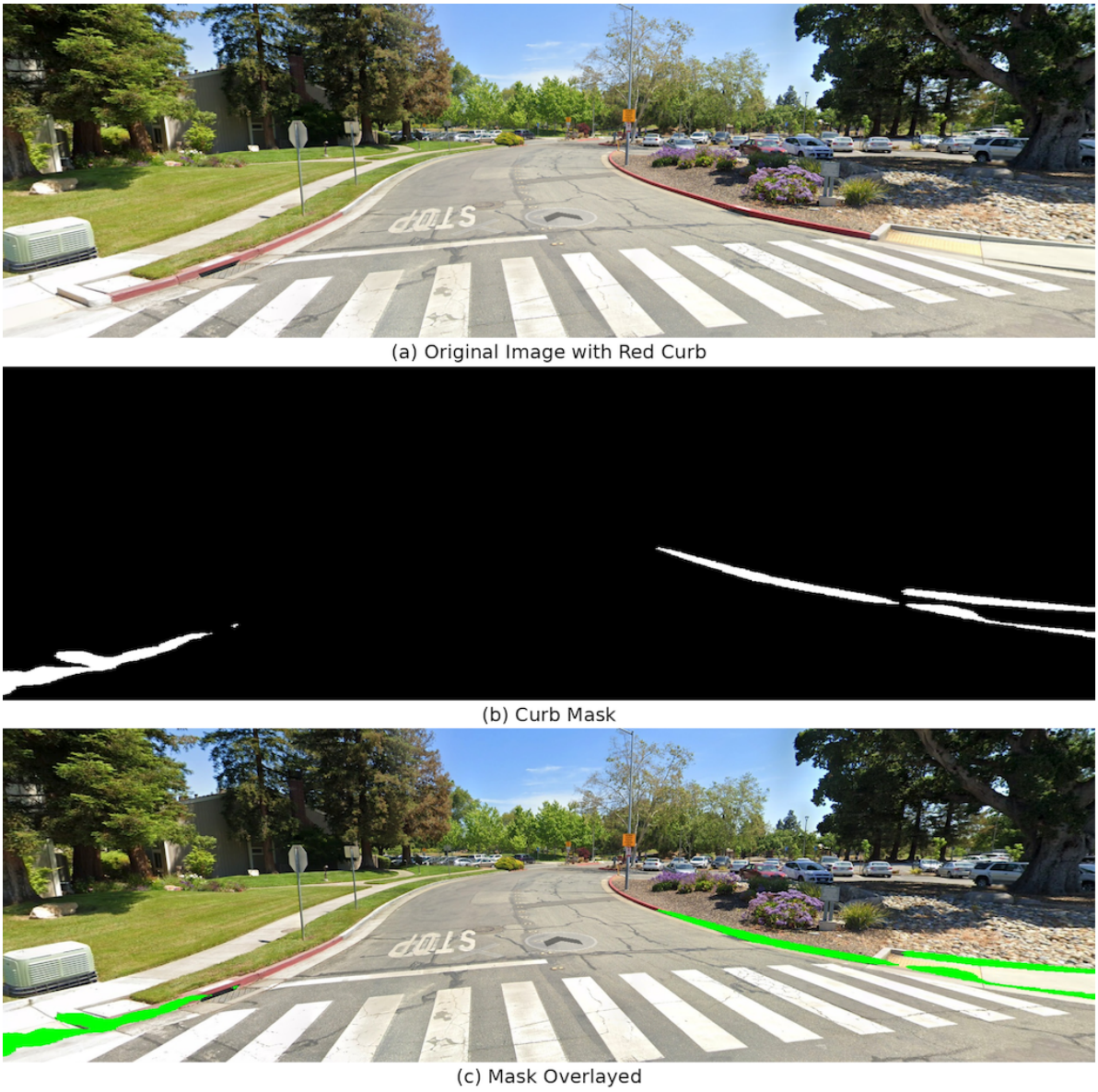
The curb color stage produced some of the most informative qualitative findings in the project. A straightforward initial approach would be to classify curb color using all pixels inside the predicted curb mask. In practice, this led to substantial contamination from nearby structures such as crosswalk stripes, painted road markings, and adjacent asphalt. In particular, white road markings frequently caused the system to over-predict `white` even when the curb itself was not white.

To address this, we changed the pipeline to use **boundary-based color extraction**. Instead of taking all predicted mask pixels, we compute the edges of the predicted curb mask and use only those boundary pixels for HSV-based color analysis. This significantly reduces contamination from nearby structures such as crosswalk markings and lane paint, and produces more conservative but more reliable curb color predictions.

We also found that color prediction requires explicit **uncertainty handling**. In many realistic scenes, the curb mask contains a mixture of red curb paint, white crosswalk markings, gray asphalt, and lighting variation. In these cases, the color distribution becomes multi-modal rather than clearly dominated by a single class. Instead of forcing a hard label, we introduced a confidence-margin rule: a color is accepted only if it has both sufficient absolute confidence and a sufficient margin

over the second-best color. Otherwise, the prediction is labeled as `unknown`. This made the final output more conservative, but also more trustworthy.

A representative failure case is shown in Figure 20. The curb is visibly red, but the predicted mask also overlaps with nearby white and gray regions. As a result, the extracted color distribution becomes ambiguous and the final output is correctly labeled as `unknown`. This is a useful project-level lesson: for curb color, **precision matters more than forcing coverage**, because an incorrect hard color prediction would be more harmful to downstream reasoning than an explicit uncertainty label.




(a) Original Image with Red Curb

(b) Curb Mask

(c) Mask Overlayed

Figure 20: Failure case for curb color inference. The curb itself is red, but the mask overlaps with nearby white and gray structures, creating a multi-modal color distribution. The conservative decision rule labels this case as `unknown` rather than forcing an incorrect dominant color.

# 7 Practical Challenges and Limitations

Several practical difficulties influenced the experiments and shaped the final system design.

## 7.1 Infrastructure and Storage Constraints

The datasets and preprocessing artifacts are relatively large. Working with tens of gigabytes of images and annotations created repeated issues across local storage, Google Colab, and remote execution environments. MTSD was 36 GB and Vistas was 32.6 GB. In particular:

- Google Colab storage limits made it difficult to keep the full dataset and processed outputs in one place.
- We initially explored multiple environments, including local setup and cloud-based options.
- We attempted to use CloudLab, but acquiring GPU instances was difficult. Even when a short A30 allocation was obtained, the combination of large data transfer and missing GPU driver issues made the setup impractical.
- Kaggle itself introduced interruptions: the sign training run crashed after epoch 24 and had to be resumed separately.

## 7.2 Evaluation on Local Hardware

When we first tried full-image inference for the validation split on a local Apple Silicon machine, the process was killed during prediction because the script attempted to pass the entire validation set at once. We fixed this by changing the evaluation code to use chunked prediction. This was an important engineering improvement because it made local validation feasible and reproducible.

Parking-meter evaluation introduced additional engineering difficulties. Full zero-shot evaluation with YOLO11x at high inference resolution was slow locally, and moving the evaluation to Kaggle required extra work because of GPU/runtime compatibility issues. We eventually ran the full validation experiment by restructuring the evaluation code into chunked batches on Kaggle.

## 7.3 Task Difficulty

The dataset contains a large number of negatives and, more importantly, extremely small positive objects. This made it clear that the hardest part of the sign-detection problem is not simply class imbalance, but small-object detection under varied viewpoint and scale. The qualitative experiments with zoom and inference resolution strongly confirmed this.

The same issue reappears in the parking-meter experiments. Many zero-shot failures were caused by the meters being too small, too far away, or visually confusable with generic poles and other narrow street furniture.

A related but distinct difficulty appeared in the curb experiments. Although curb annotations are much more abundant than parking-meter annotations, curb *color* is not directly supervised and is much harder to recover robustly than curb presence. The main challenge was not simply detecting curb pixels, but ensuring that the recovered pixels actually corresponded to the curb itself rather than adjacent crosswalk paint, road markings, or asphalt. This made the curb module less of a pure segmentation problem and more of a joint segmentation-and-representation problem, where uncertainty handling became an essential part of the final design.

### 7.4 Limitations of the Aggregation Evaluation

The segment-level aggregation experiments are useful but should be interpreted with the right scope. The synthetic pseudo-segment benchmark tests the mathematical and systems behavior of aggregation under sparse cue visibility, but it does not prove full geographic street-segment reasoning because the synthetic segments are assembled from images across datasets rather than from a road network. This was a deliberate design choice: the available public datasets provide image-level annotations for signs, meters, and curbs, but not clean, labeled street-segment parking ground truth.

The manually collected six-segment dataset addresses this limitation qualitatively by using real nearby views from actual street locations. However, it is intentionally small because manual collection is slow: each example requires finding a suitable street, capturing multiple nearby views, ensuring that the images correspond to the same local curbside context, and writing notes about the visible cue pattern. Therefore, the manual examples are best understood as real-world validation cases rather than a statistically significant benchmark.

The aggregation rule itself is also heuristic. The weights for signs, meters, and curb color are based on observed cue reliability rather than learned calibration. This is appropriate for the current project stage because the goal is to demonstrate the value of aggregation, but a future system should learn cue weights on a larger georeferenced validation set and should explicitly model spatial consistency, road side, and distance along the street.

## 8 Conclusion

This project studies street parking presence inference from street-level imagery through a multi-cue pipeline built around parking signs, parking meters, curb structure, curb color, and segment-level aggregation. The strongest single cue remains the supervised parking-sign detector trained on MTSD, with final validation performance of **mAP@50 = 0.5487**, **mAP@50–95 = 0.3824**, **image-level F1 = 0.6673**, and **AUROC = 0.8310**. These results show that sign detection is a useful baseline, but the threshold and qualitative analyses also show that single-image sign detection is limited by small-object scale, viewpoint, and sign appearance variation.

The parking-meter and curb experiments refine the role of auxiliary cues. The COCO-pretrained parking-meter detector transfers partially to Mapillary Vistas, but low precision and many pole-like false positives mean that parking meters should not be treated as a primary standalone cue. Curb segmentation is feasible and can recover useful structure, but curb color inference must be conservative because strong painted curb colors are sparse and color estimates are sensitive to mask contamination. Boundary-based color extraction and confidence-margin rules make the curb-color output more interpretable, but the resulting signal remains auxiliary.

The final aggregation experiments provide the main systems-level result. On a controlled synthetic pseudo-segment benchmark with 225 five-image segments, segment-level aggregation improved F1 from **0.319** for a single-image per-segment baseline to **0.830**. The gain is primarily driven by recall: the baseline misses most positive segments because a single selected view often does not contain the relevant cue, while aggregation can recover evidence from any of the five views. This directly supports the original motivation of the project: street parking evidence is sparse and viewpoint-dependent, so the segment should be inferred from multiple views rather than one image.

The manually collected six-segment real-world dataset provides qualitative validation of the same idea. Aggregation correctly recovered five of six positive examples, including meter-only, curb-only, and mixed-cue segments that the single-image sign baseline missed. The one failure case

is also informative: the visible sign was outside the detector's training distribution, showing that aggregation can compensate for sparse visibility but not for missing visual concepts.

Overall, the project demonstrates a complete pipeline from cue detection to segment-level inference. The results support a practical conclusion: parking-sign detection is a strong starting point, but robust street parking inference requires multi-view aggregation and auxiliary cues. Future work should replace synthetic pseudo-segments with a larger georeferenced segment dataset, learn cue-fusion weights from validation data, incorporate road-side geometry, and extend sign understanding toward OCR and rule interpretation.