Lane Detection using Deep Layer Aggregation

ECE 285 - Spring 2020
Group 3: Hala Abualsaud, Sean Liu, David Lu, Kenny Situ
Mentor: Akshay Rangesh
Motivation

- Lane detection is one of the most important components to developing any intelligent transportation system
- Required to steer and guide the vehicle
- Help vehicles interact safely on the road
Expected Goal

Improve lane detection performance by combining a state-of-the-art segmentation backbone network architecture with various loss functions.
Objectives

1. Use Deep Layer Aggregation (DLA) as backbone network
   a. DLA-34

2. Evaluate performance of different loss functions
   a. Binary Cross-Entropy loss
   b. Binary Cross-Entropy loss with Self Attention Distillation (SAD) loss
Novelty

- Utilization of Deep Layer Aggregation (DLA) for the purpose of lane detection
- Utilization of SAD loss metric to train DLA
Related Research

- Conventional methods of lane detection utilizes computer vision techniques and rely on heuristics and hand crafted features.

- Current approaches are camera based systems and rely on deep learning models; it works by extracting features in an end-to-end manner.

- Another approach to the lane detection task is instance segmentation.
Datasets, Software, Libraries, and Compute Resources

**Dataset**
- TuSimple

**Models**
- DLA-34
- ENet (for comparison)
- ResNet-18 (for comparison)
- ResNet-34 (for comparison)

**Language / Libraries**
- Python
- PyTorch
- Numpy

**Compute Resources**
- Datahub
  - Provided 1 GPU and 4 CPUs with 32 GB RAM
Workflow Diagram

1. TuSimple Dataloader
2. Backbone Architecture (DLA-34)
3. Build Losses
4. Compare DLA to Prior Work
5. Binary Cross-Entropy
6. Binary Cross-Entropy + Self Attention Distillation

- Architecture Improvements
- Train our Networks on the Dataset
Methodology

1. Build the backbone architecture with dataset (DLA-34)
2. Build our loss (SAD with Binary Cross-Entropy)
3. Compare prior work with our implementation (DLA-SAD)
4. Test whether SAD loss improves the state-of-the-art DLA network
5. Implement changes in architecture to improve accuracy/run-time
Methodology

Deep Layer Aggregation (DLA)

a. Iterative Deep Aggregation (IDA)
b. Hierarchical Deep Aggregation (HDA)
1. **Iterative Deep Aggregation (IDA)**

\[ I(x_1, \ldots, x_n) = \begin{cases} 
  x_1 & \text{if } n = 1 \\
  I(N(x_1, x_2), \ldots, x_n) & \text{otherwise,}
\end{cases} \]

2. **Hierarchical Deep Aggregation (HDA)**
Methodology

Self Attention Distillation loss

a. Add an attention generator after each hierarchical deep aggregation

b. Implemented as distillation loss term in overall loss function

c. Total loss consists of segmentation loss added to existence loss (binary cross entropy) and SAD loss

\[
\mathcal{L}_{\text{distill}}(A_m, A_{m+1}) = \sum_{m=1}^{M-1} \mathcal{L}_d(\Psi(A_m), \Psi(A_{m+1}))
\]

\[
\mathcal{L} = \mathcal{L}_{\text{seg}}(s, \hat{s}) + \alpha \mathcal{L}_{\text{IOU}}(s, \hat{s}) + \beta \mathcal{L}_{\text{exist}}(b, \hat{b}) + \gamma \mathcal{L}_{\text{distill}}(A_m, A_{m+1})
\]
Experiment Details (Parameters)

- Learning rate = .005
- Number of epochs = 100
- Early Stopping
- Loss:
  - Binary Cross-Entropy
  - Optimizer: Adam
  - Momentum = 0.9
  - Weight decay = 0.005
  - Batch size = 3
Training Process

TuSimple Dataloader & Data Pre-processing → Backbone (DLA-34) → Sigmoid Activation Function → Calculate BCE Loss and Accuracy → Backprop Loss
Loss & Accuracy Plots

**Losses vs Epoch**
- **train loss**
- **val loss**

**Accuracy vs Epoch**
- **train accuracy**
- **val accuracy**
Preliminary Results

Test Image #1 Ground Truth Heat Map

Test Image #1 Generated Heat Map
Individual Workload

• Researched current state-of-the-art and decided technical approach (all)

• Completed dataloader for TuSimple dataset (Kenny)

• Integrated DLA backbone architecture to train with TuSimple dataset (all)

• Trained model on TuSimple dataset using Binary Cross-Entropy loss (all)
Future Work

1. Improve current results from DLA baseline network
2. Train network with SAD loss
3. Compare DLA baseline with DLA-SAD and prior work
4. Post-process model outputs
Milestones & Timeline

Week 10

Train DLA Baseline

Week 11

Train with SAD Loss

Continuous Improvement

Report Writing
Thank you!
Questions?