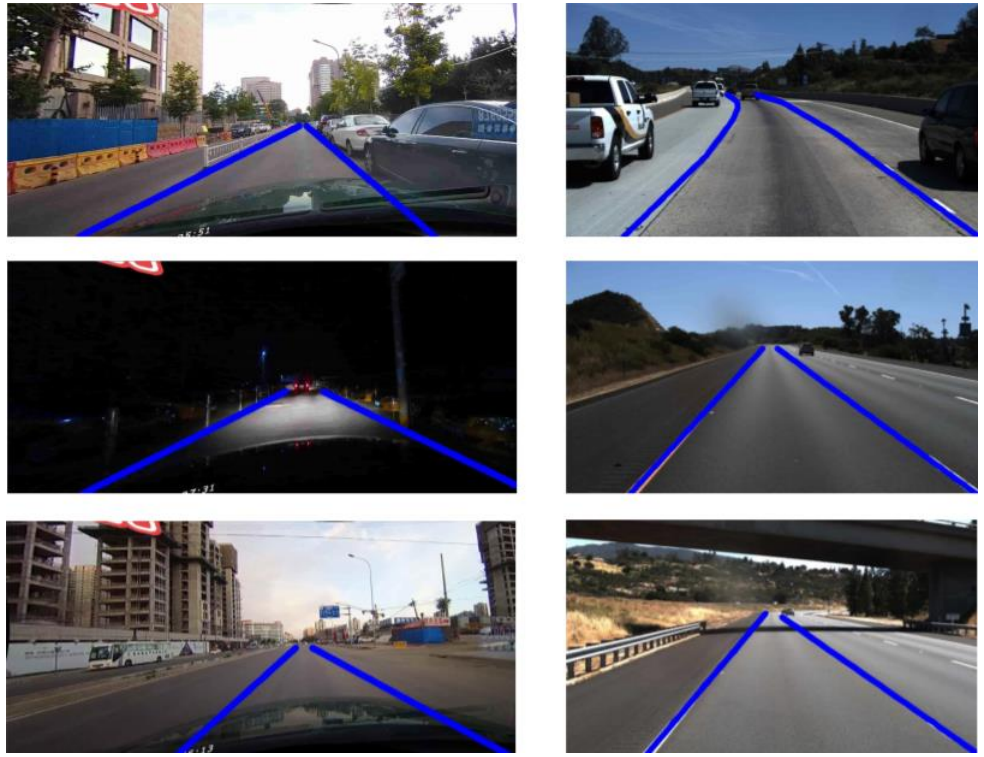
RONELD: Robust Neural Network Output Enhancement for Active Lane Detection

A.I. Aviation Lab Pte. Ltd
Singapore

ZHE MING CHNG, JOSEPH MUN HUNG LEW, JIMMY ADDISON LEE

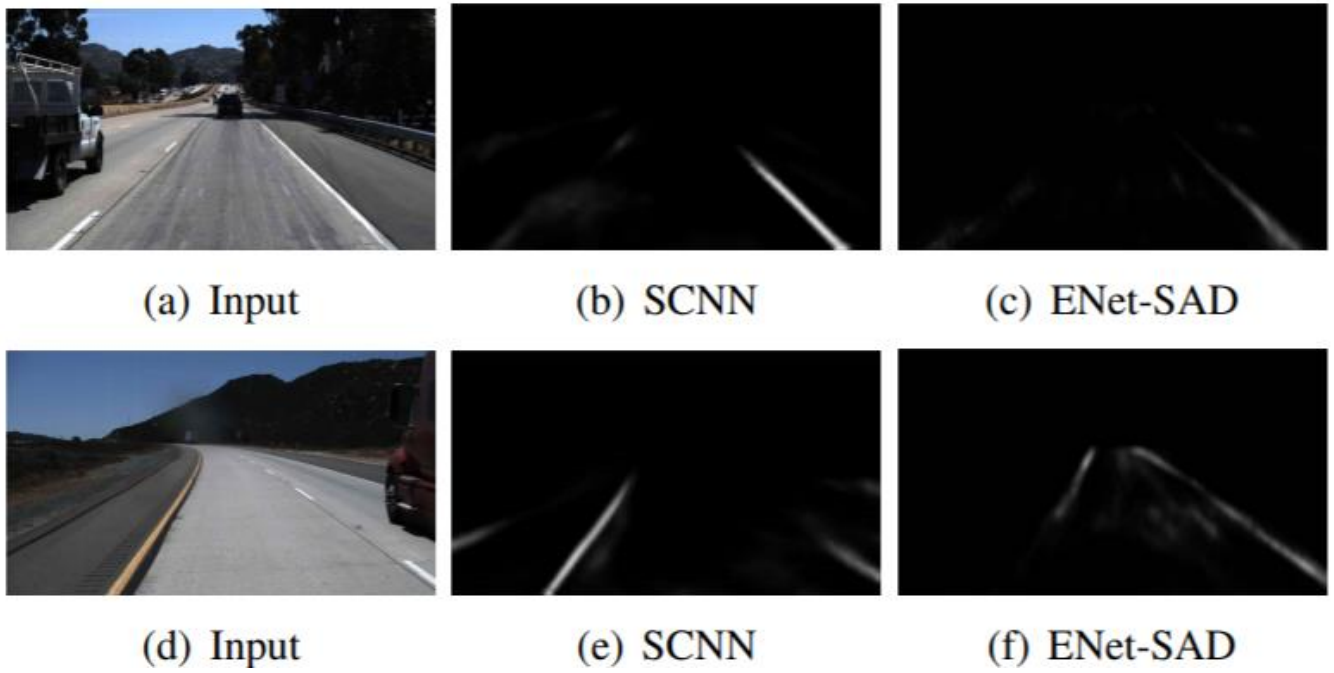
Problem Overview

Active lane detection: identifying the lane that the vehicle is currently travelling on



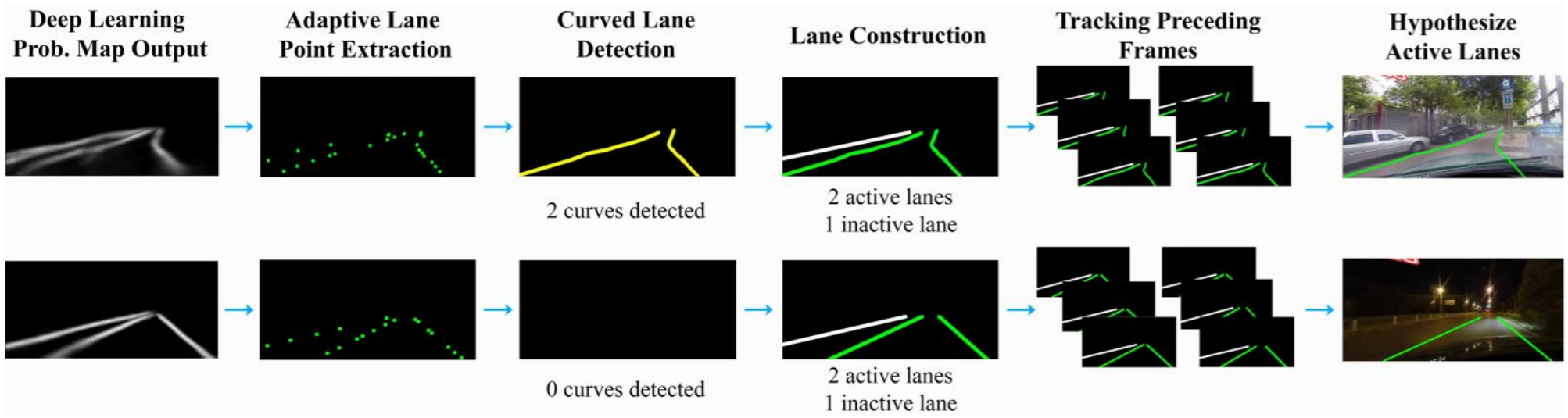
Motivation

Poor performance of deep learning models on unseen datasets

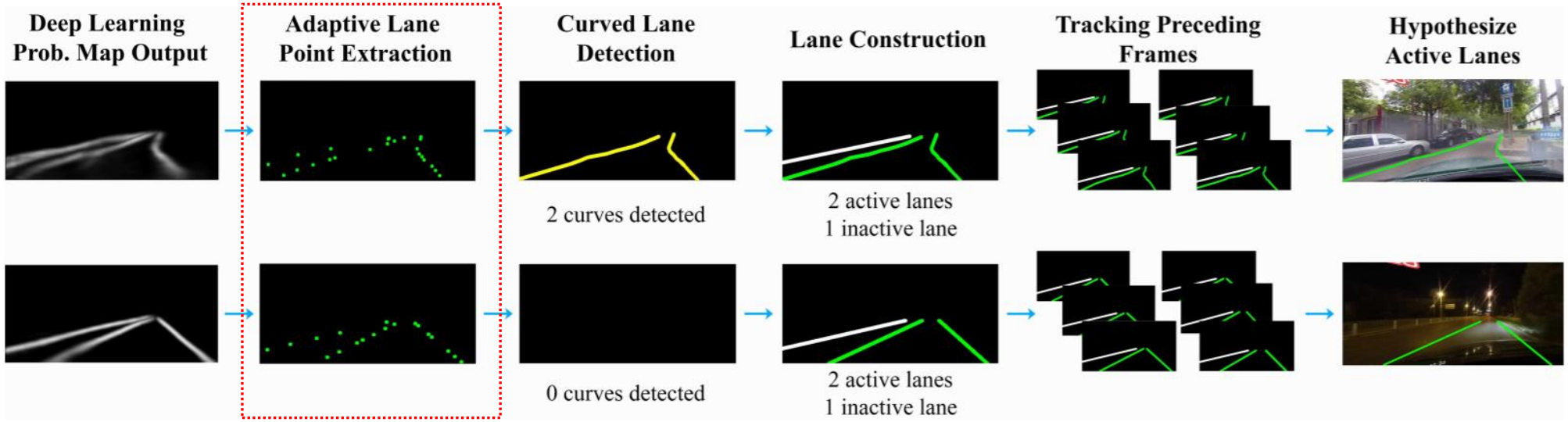


CULane-trained models on TuSimple datasets

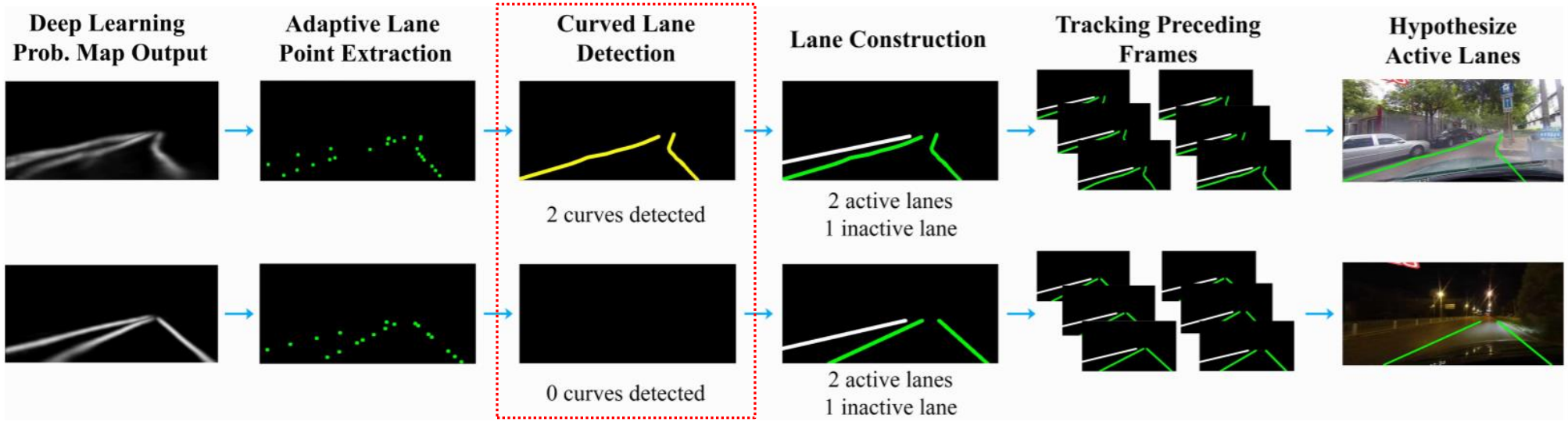
Method


Deep Learning Prob. Map Output
Adaptive Lane Point Extraction
Curved Lane Detection
Lane Construction
Tracking Preceding Frames
Hypothesize Active Lanes
2 curves detected
2 active lanes
1 inactive lane
0 curves detected
2 active lanes
1 inactive lane

Method


Deep Learning Prob. Map Output
Adaptive Lane Point Extraction
Curved Lane Detection
Lane Construction
Tracking Preceding Frames
Hypothesize Active Lanes
2 curves detected
2 active lanes
1 inactive lane
0 curves detected
2 active lanes
1 inactive lane

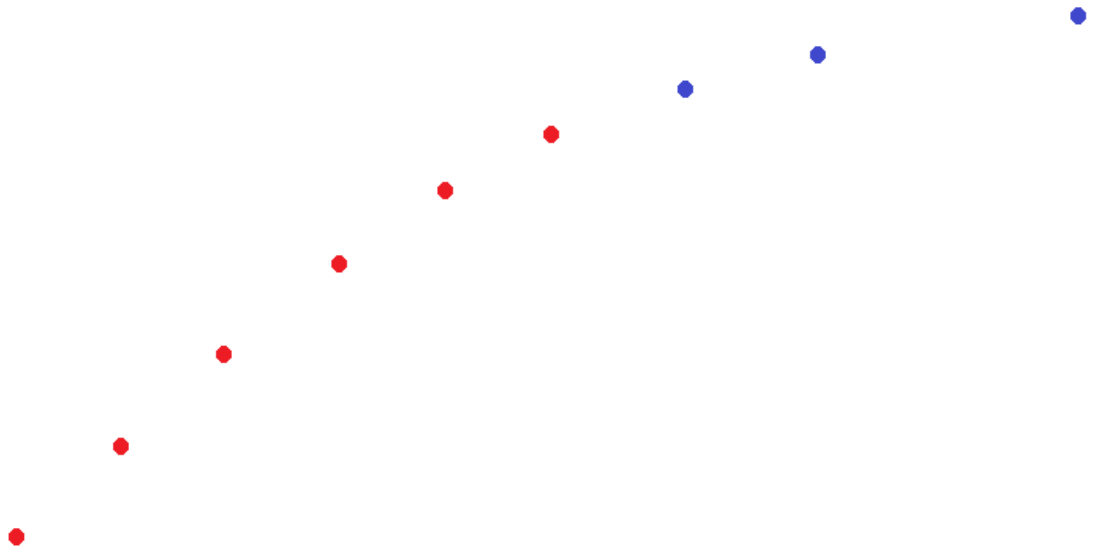
Method


Deep Learning Prob. Map Output
Adaptive Lane Point Extraction
Curved Lane Detection
Lane Construction
Tracking Preceding Frames
Hypothesize Active Lanes
2 curves detected
2 active lanes
1 inactive lane
0 curves detected
2 active lanes
1 inactive lane

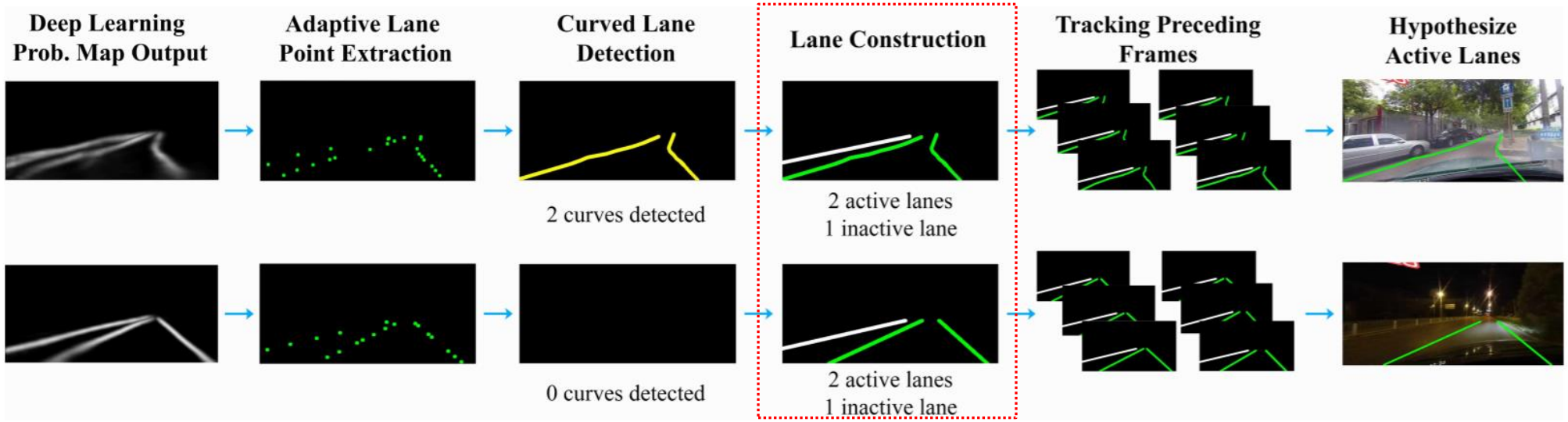
Method

- Curved lane detection
  - Minimum length: 9 points

$$r^2 = \frac{[\mathrm{Cov}(X,Y)]^2}{\mathrm{Var}(X)\mathrm{Var}(Y)}$$



Method


Deep Learning Prob. Map Output
Adaptive Lane Point Extraction
Curved Lane Detection
Lane Construction
Tracking Preceding Frames
Hypothesize Active Lanes
2 curves detected
2 active lanes
1 inactive lane
0 curves detected
2 active lanes
1 inactive lane

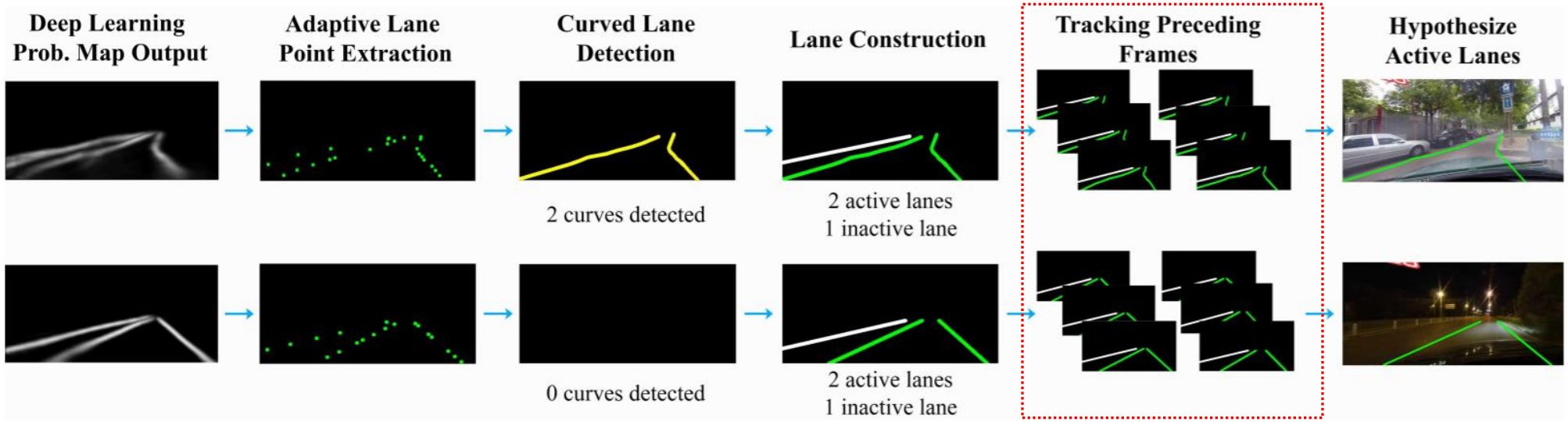
Method

- Lane construction
  - For curved lanes, we use quadratic splines to connect the lane points
  - For straight lanes, we use weighted least squares linear regression

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} ||\boldsymbol{C}^{\frac{1}{2}}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})||^2 = (\boldsymbol{X}^T\boldsymbol{C}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{C}\boldsymbol{y}.$$

$$\boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_m \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}, \boldsymbol{C} = \begin{pmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & c_m \end{pmatrix},$$

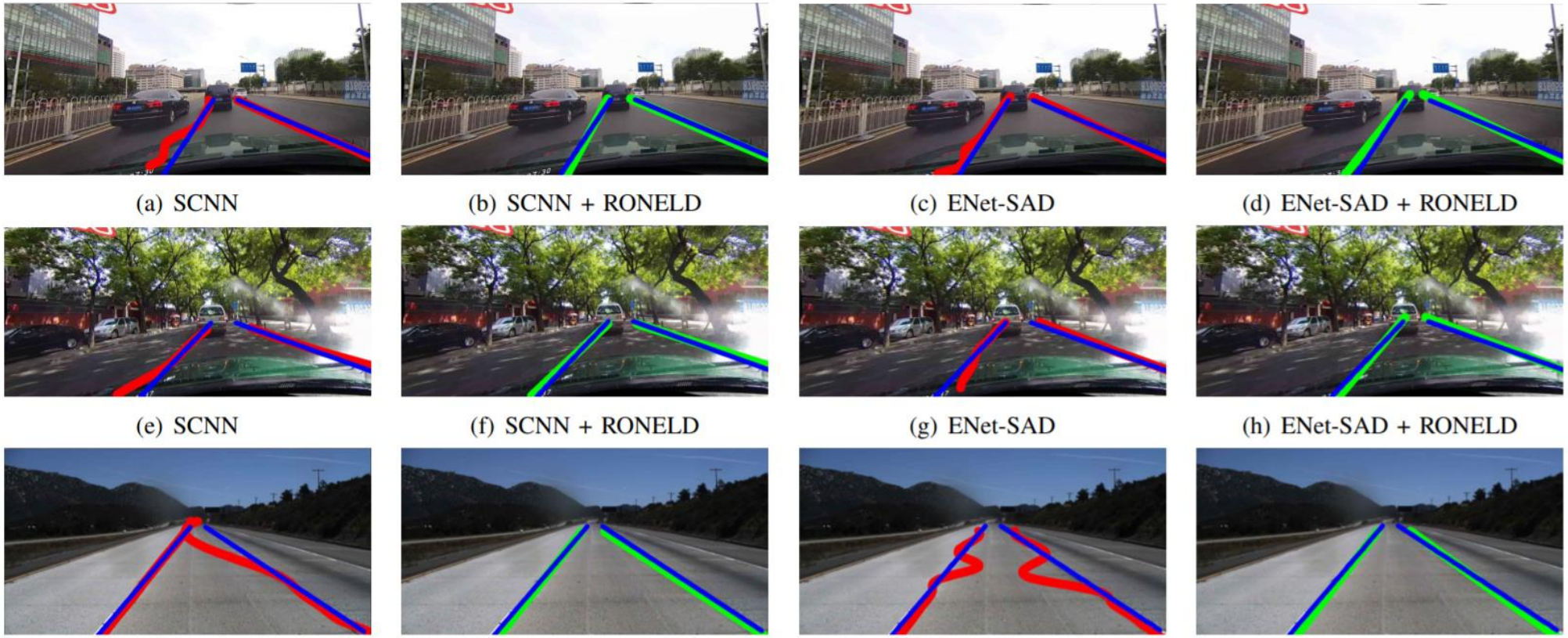
Method


Deep Learning Prob. Map Output
Adaptive Lane Point Extraction
Curved Lane Detection
Lane Construction
Tracking Preceding Frames
Hypothesize Active Lanes
2 curves detected
2 active lanes
1 inactive lane
0 curves detected
2 active lanes
1 inactive lane

Method

- Tracking preceding frames
- Weight of lane is dependent on:
  - Number of points in lane marking
  - Confidence of points in lane marking
  - Whether the lane is an identified active lane
  - Number of frames where lane was missing

Results



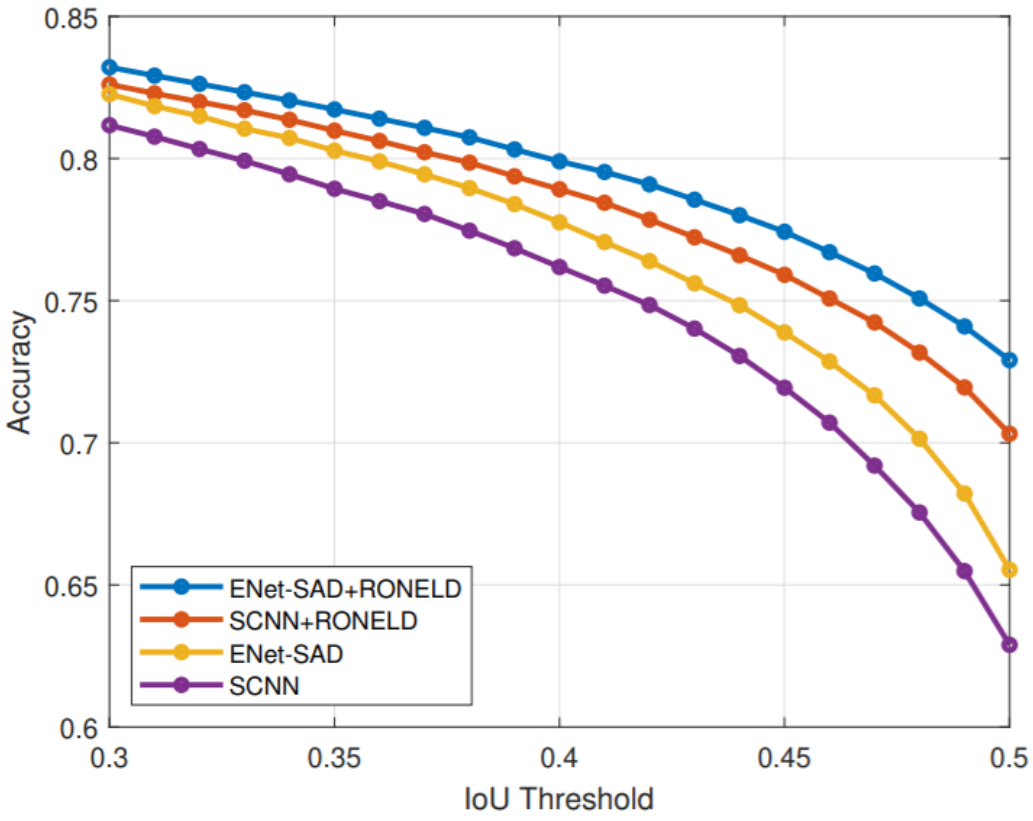
(a) SCNN (b) SCNN + RONELD (c) ENet-SAD (d) ENet-SAD + RONELD

(e) SCNN (f) SCNN + RONELD (g) ENet-SAD (h) ENet-SAD + RONELD

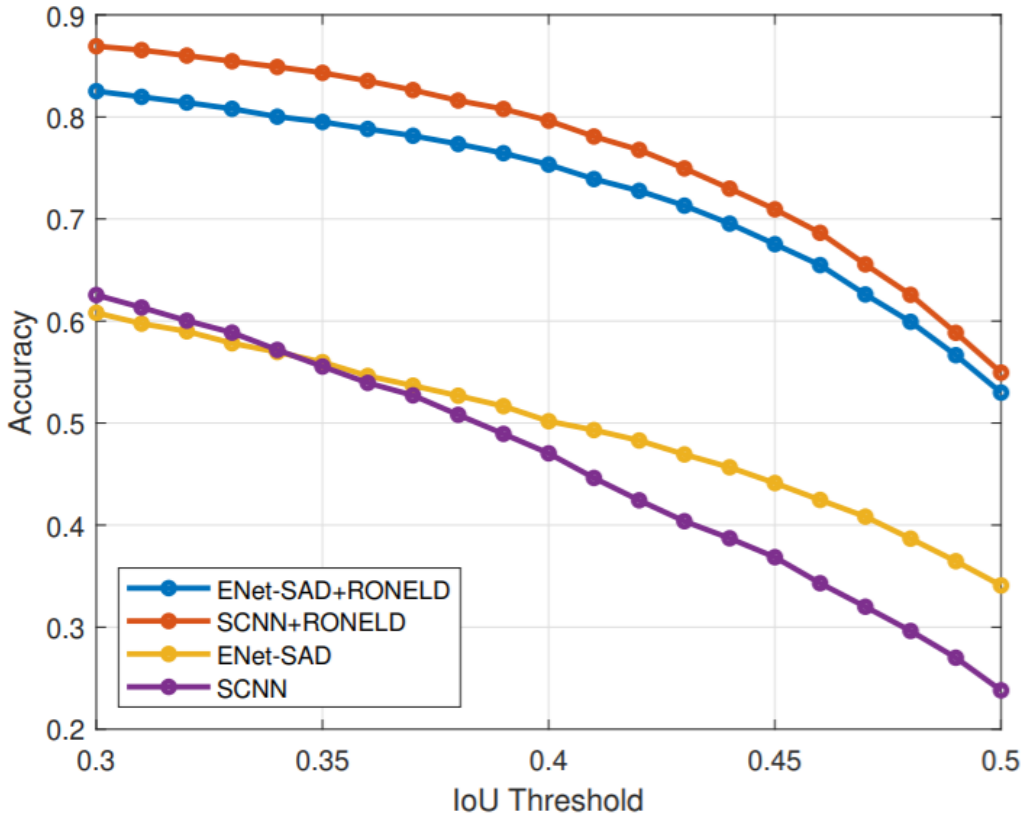
(i) SCNN (j) SCNN + RONELD (k) ENet-SAD (l) ENet-SAD + RONELD

Results

CULane



TuSimple



Limitations

- Added runtime to the lane detection model

TABLE V
AVERAGE RUNTIME OF SCNN + RONELD AND ENET-SAD + RONELD ON THE CULANE AND TUSIMPLE TEST SETS (IN MILLISECONDS) USING A PYTHON 3 + NUMBA [35] IMPLEMENTATION.

Dataset	SCNN + RONELD	ENet-SAD + RONELD
CULane	5.68	6.29
TuSimple	2.80	3.55
Mean	4.24	4.92

Conclusion

- Current state-of-the-art deep learning lane detection models work well on test sets that are similar to their train sets but do not work well when datasets start to differ significantly
- RONELD enhances the output of deep learning lane detection models to achieve higher accuracy through a four step method: adaptive lane point extraction, curved lane detection, lane construction, and tracking preceding frames

Code: github.com/czming/RONELD-Lane-Detection

Email: zchng3@gatech.edu