1. Appendix

1.1. Inference

In the inference phase, the four branches of the network output, including the confidence $M_{conf}$, the embedding $M_{emb}$, the offset $M_{off}$, and the lane height $M_Z$, are synthesized into the instance-level lanes by a fast unsupervised clustering method which we refer to mean-shift. As shown in Alg 1. This process consists of four steps: 1) Filtering the positive points of confidence mask $M_{conf}$ by the $S_{threshold}$ to obtain the $E_{list}$. 2) Clustering points of $M_{emb}$ to obtain the clustering points of each lane $R_{point}$ and cluster centers of each lane $R_{center}$ by inter-class distance $D_{gap}$. 3) Adding offset $M_{off}$ and lane height $M_Z$ to points in each lane to obtain 3D lanes $R_{lines}$. 4) Fitting the key points of the lanes $R_{lines}$ to yield the lane equations $R_{fit}$.

1.2. Visualization

We analyse the comparison between BEV-LaneDet and PersFormer [1] in different scenarios by visualization. In Figure 1, the first column is the input images; the second column is the results of PersFormer on BEV; the third column is the results of our method on BEV; the fourth column is the results of PersFormer in 3D space; the fifth column is the results of our method in 3D space. The visualization results demonstrate that our method is more stable and accurate in different scenarios. At the same time, our method is more suitable to represent the diversity of lane structures compared with PersFormer.

References


---

**Algorithm 1** Post-processing algorithm of our method

**Input:** $M_{conf}, M_{emb}, M_{off}, M_{Z} \leftarrow \text{Model}(I_c)$

**Output:** 3D lanes after fitting, $R_{fit}$:

1. Filtering the point of confidence mask by $S_{threshold}$ to get the $E_{list}$:
   2. for $x = 0; x < M_{conf}.cols; x += 1$
   3. for $y = 0; y < M_{conf}.rows; y += 1$
   4. if $M_{conf}[x,y] >= S_{threshold}$ then
   5. $E_{list}.append([x,y,M_{emb}(:,y,z)])$;
   6. end if
   7. end for
   8. end for

9. Clustering points to get the $R_{point}$ and $R_{center}$ by $D_{gap}$:
   10. for $i = 0; i < E_{list}.length; i += 1$
   11. $x, y, value = E_{list}[i]$;
   12. $min_{gap} = D_{gap} + 1$
   13. $min_{cid} = -1$
   14. for $j = 0; j < R_{center}.length; j += 1$
   15. $center_{id}, (center, num) = R_{center}[j]$;
   16. $diff = \text{Euclidean}(value, center)$;
   17. if $diff < min_{gap}$ then
   18. $min_{gap} = diff$
   19. $min_{cid} = center_{id}$
   20. end if
   21. end for
   22. if $min_{gap} < D_{gap}$ then
   23. $R_{point}.append([x,y, min_{cid}])$
   24. $center, num = R_{center}[min_{cid}]$
   25. $R_{center}[min_{cid}] = ([center \times num + value]/(num + 1), num + 1)$
   26. else
   27. $R_{center}.append([value, 1])$
   28. $R_{point}.append([x,y, R_{center}.length - 1])$
   29. end if
   30. end for
   31. Adding offset to points in each lane to get $R_{lanes}$:
   32. for $k = 0; k < R_{point}.length; k += 1$
   33. $x, y, id = R_{point}[k]$
   34. $off_{yx} = M_{off}[:,y,x]$
   35. $z = M_{Z}[y,x]$
   36. $R_{lanes}[id].append([x,y + off_{yx}, z])$
   37. end for
   38. $R_{fit} = \text{FitFunc}(R_{lanes})$.

*Corresponding author
Figure 1. Qualitative results of PersFormer [1] and BEV-LaneDet in different scenarios of the OpenLane dataset. First row: Curve; second row: Merge&Split; third row: Up&Down; fourth row: Night; fifth row: Intersection; sixth row: Backlight.