Devil is in the Edges: Learning Semantic Boundaries from Noisy Annotations

David Acuna, Amlan Kar, Sanja Fidler
CVPR19(Oral)

Sanghyeon Lee

14 Dec. 2020

Introduction

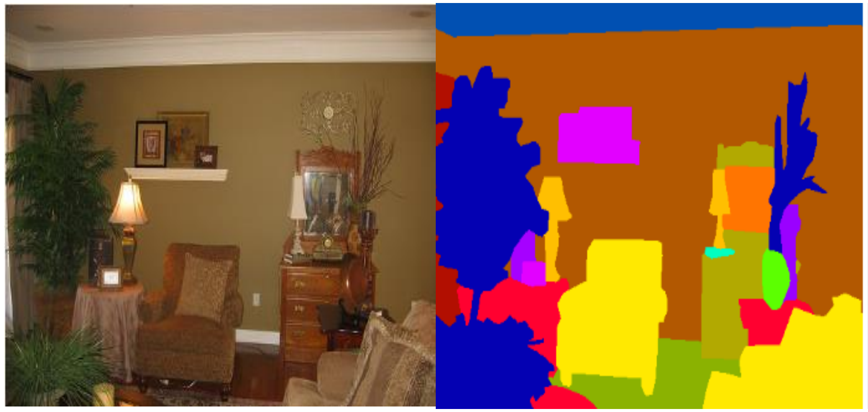
Precise annotations are laborious to get ➔ Datasets consist of a significant level of label noise

1. Boundary thinning layer (Motivated by Canny edge)
2. Active alignment (Level set optimization)

PASCAL VOC

COCO

PASCAL Context



ADE20K

Introduction

Common 'derivative' filters

Sobel $\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$ $\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$

Scharr $\begin{bmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{bmatrix}$ $\begin{bmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix}$

Prewitt $\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$ $\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

Roberts $\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

Canny edge detector (Canny,1986)

1. Smoothing (parameter: σ)

$$G(x) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} \exp(-\frac{x^2}{2\sigma^2})$$

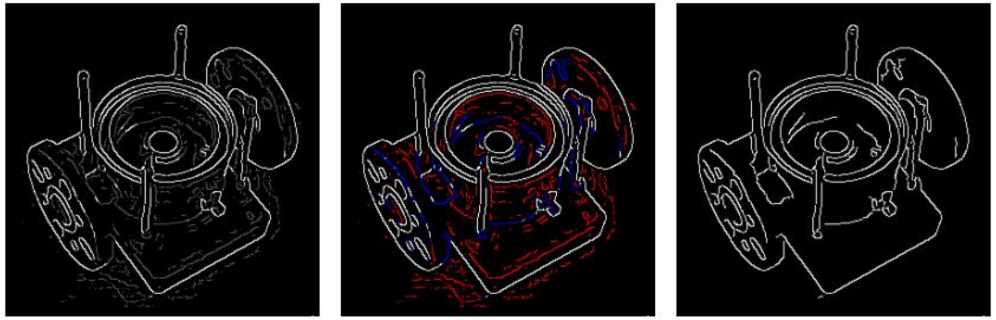
$$B = \frac{1}{159} \cdot \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & \boxed{15} & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

2. Gradient magnitudes

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]^T \qquad \nabla f_x = \underbrace{\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}}_{\text{Sobel}} = \underbrace{\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}}_{\text{weighted average and scaling}} \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$$

$$|\nabla f| = \sqrt{\nabla f_x^2 + \nabla f_y^2}$$

(a) Original (b) Smoothed (c) Gradient magnitudes (d) Edges after non-maximum suppression



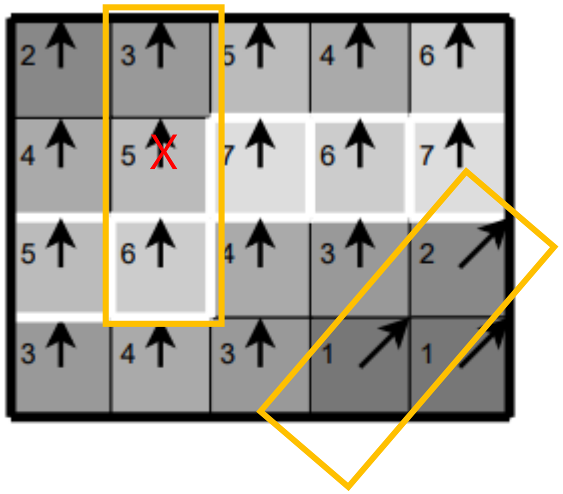
(e) Double thresholding (f) Edge tracking by hysteresis (g) Final output

All steps of the edge detector

Introduction

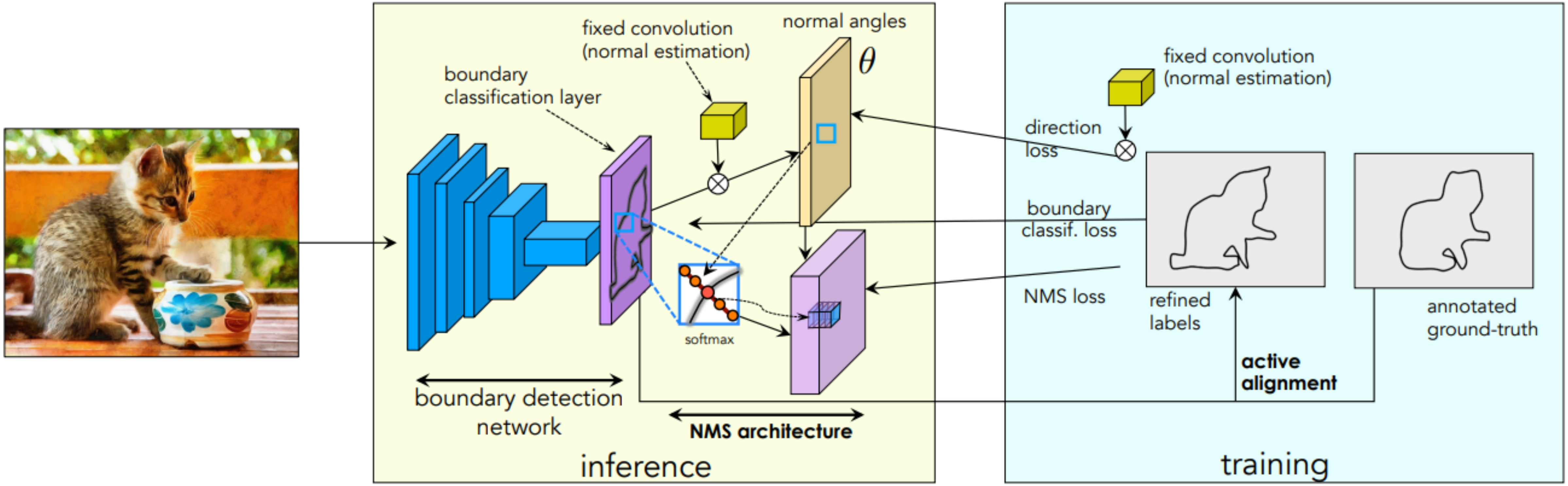
Canny edge detector (Canny,1986)

3. Non-maximum suppression
- Preserving local maxima in the gradient image and deleting everything else
1) Round the gradient direction θ to nearest 45◦ , corresponding to the use of an 8-connected neighborhood
2) Compare the edge strength of the current pixel with the edge strength of the pixel in the positive and negative gradient direction
3) If the edge strength of the current pixel is largest; preserve the value of the edge strength. If not, suppress (i.e. remove) the value



4. Double thresholding and Edge tracking by hysteresis
(parameter: t_h, t_l)

Methods



1. Semantic-Aware Edge-Detection

$$L_{BCE}(\theta) = -\sum_k logP(y_k|x;\theta) = -\sum_k \sum_m \{\beta y_k^m log f_k(m|x;\theta) + (1-\beta)(1-y_k^m) log(1 - f_k(m|x;\theta))$$

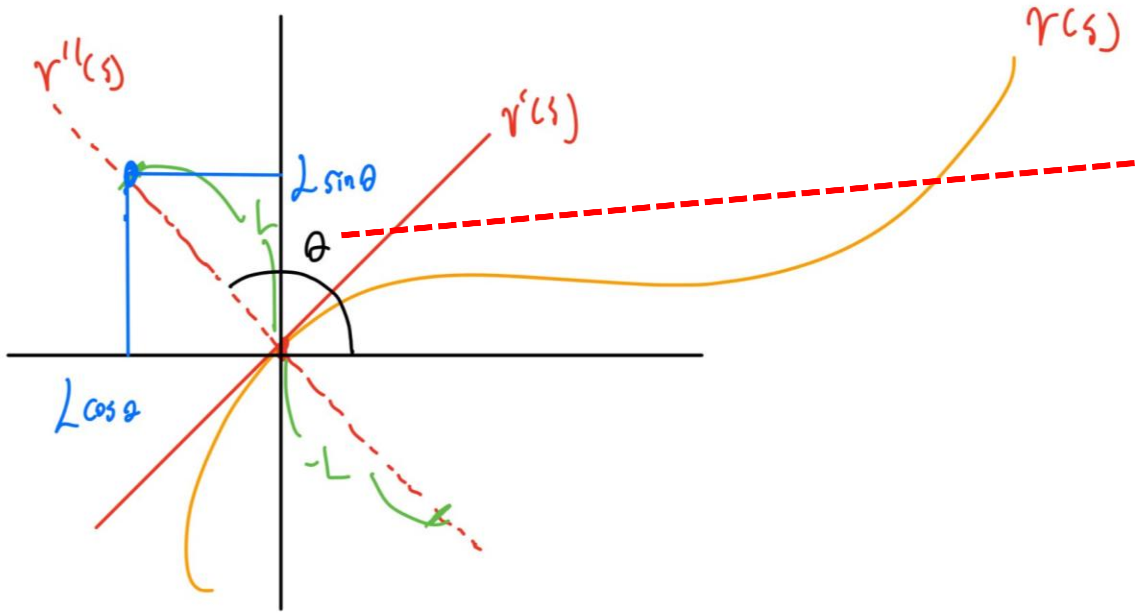
where, $y_k^m \in (0,1)$, $\beta = \frac{\#Boundary\ pixel}{\#Non{-}Boundary\ pixel}$, **m: pixel, k: class,** f_k**: model**

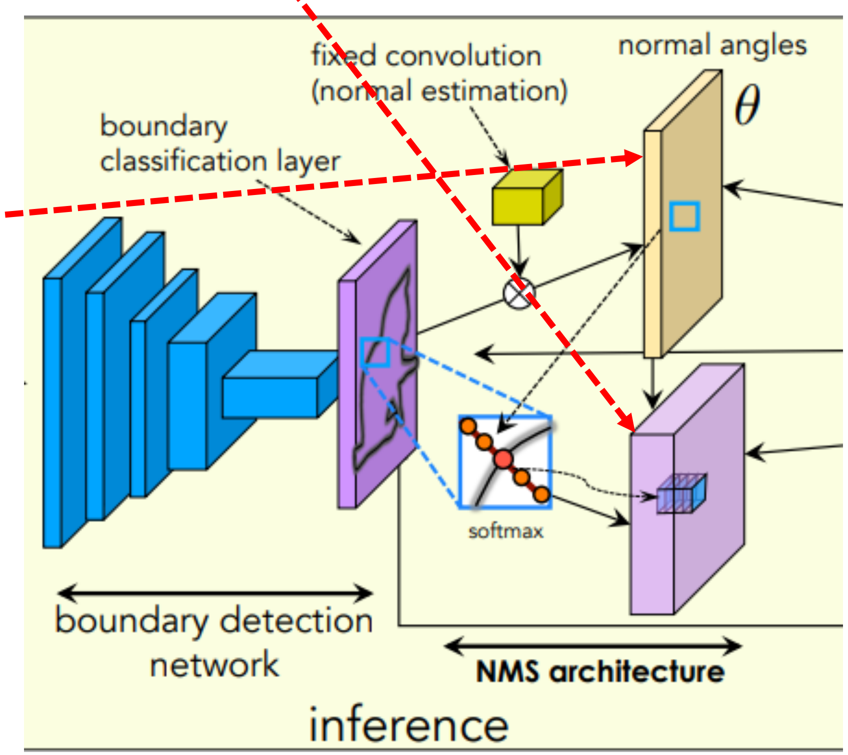
Methods

2. Semantic Boundary Thinning Layer (NMS loss)

$$L_{nms}(\theta) = -\sum_{k}\sum_{p} \log h_k(p|x;\theta)\,,\; h_k(p|x;\theta) = \frac{\exp(f_k(p|x,\theta)/\tau)}{\sum_{t=-L}^{L} \exp(f_k(p_t|x,\theta)/\tau)}$$

$$p_t = (p_{t,x}, p_{t,y}) = p + (t\cos\overrightarrow{d_p}, t\sin\overrightarrow{d_p})$$



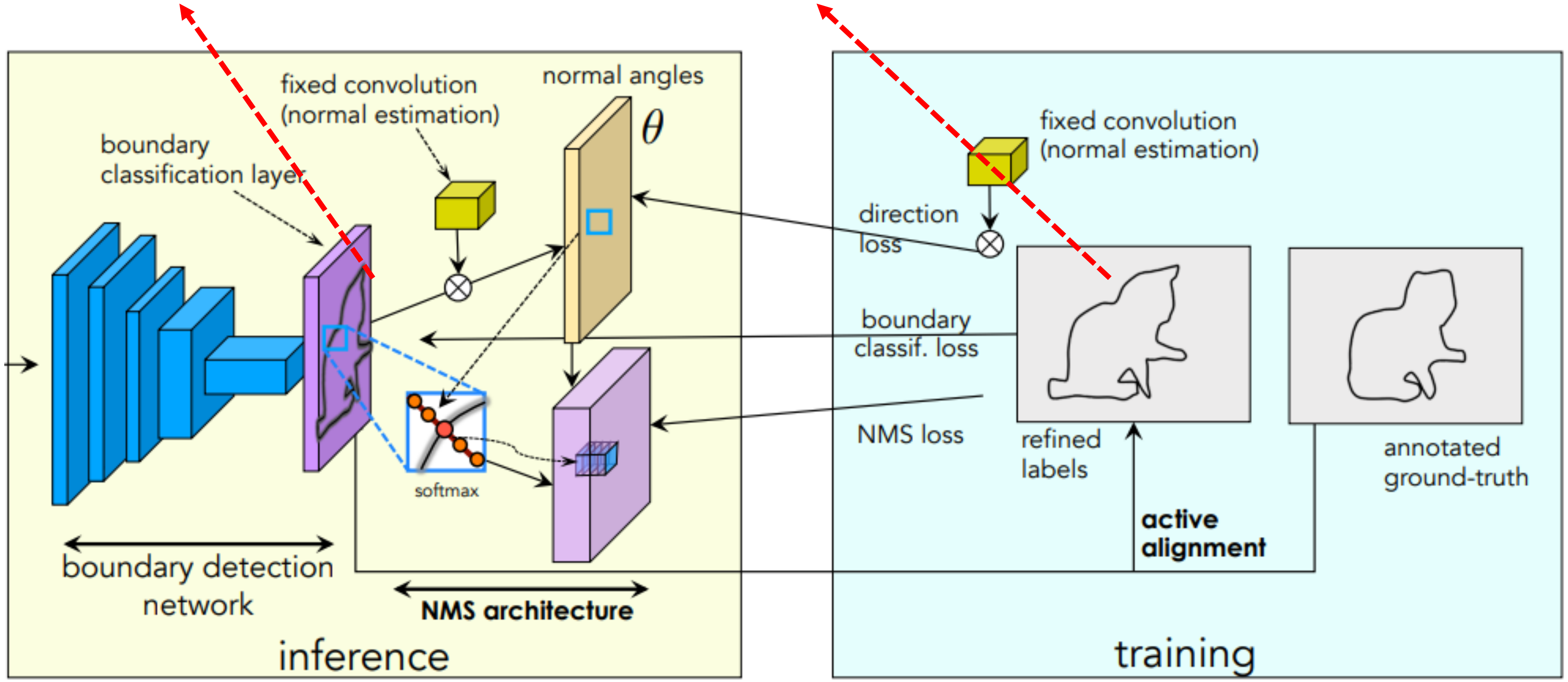


Methods

2. Semantic Boundary Thinning Layer (Direction loss)

$$L_{dir}(\theta) = -\sum_{k}\sum_{p} ||cos^{-1} < \overrightarrow{d_p}, \overrightarrow{e_p}(\theta) > ||,$$

$$\overrightarrow{e_p}(\theta): from\ the\ predicted\ boundary, \overrightarrow{d_p}: from\ the\ GT\ boundary$$



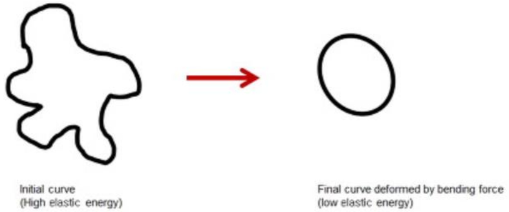
Methods

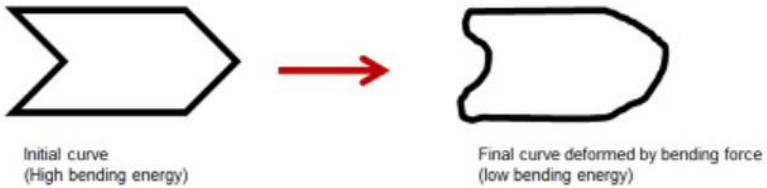
3. Active Alignment

3.1 Active Contour Model (Snake)

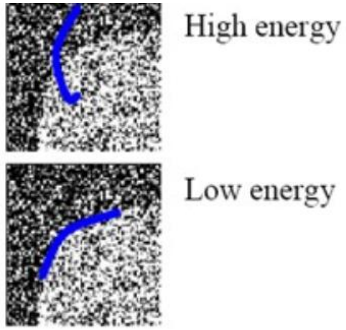
$$\mathbf{argmin}_{v(s)} E(v(s)) = \int E_{int}(v(s)) + E_{image}(v(s)) + E_{reg}(v(s))ds$$

$$E_{\text{int}} = \frac{\alpha(s)|v_s(s)|^2 + \beta(s)|v_{ss}(s)|^2}{2}$$





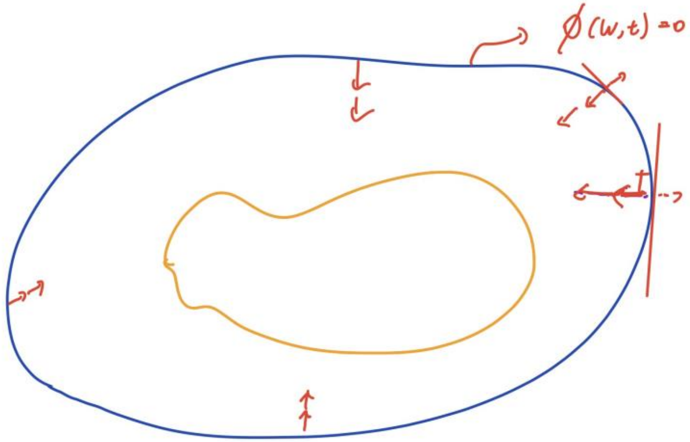
$$E_{image} = w_1 I(x, y) - \mathbf{w_2}|\nabla \mathbf{I}(\mathbf{x}, \mathbf{y})|^2$$



3.2Level set method

$C = \{w = (x, y) \in R^2 | \phi(w, t) = 0\}, C: Curve, \phi: level\ set\ function\ s.t$

We calculate the velocity of a curve along the normal vector



Experiments

1. Per category performance in the re-annotated(Feng et al. CVPR'17) SBD set

Metric	Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mean
MF (ODS)	CASENet	74.84	60.17	73.71	47.68	66.69	78.59	66.66	76.23	47.17	69.35	36.23	75.88	72.45	61.78	73.10	43.01	71.23	48.82	71.87	54.93	63.52
	CASENet-S	76.26	62.88	75.77	51.66	66.73	79.78	70.32	78.90	49.72	69.55	39.84	77.25	74.29	65.39	75.35	47.85	72.03	51.39	73.13	57.35	65.77
	SEAL	78.41	66.32	76.83	52.18	67.52	79.93	69.71	**79.37**	49.45	72.52	41.38	78.12	74.57	65.98	76.47	49.98	72.78	**52.10**	74.05	58.16	66.79
	Ours (NMS Loss)	78.96	66.20	77.53	**54.76**	69.42	**81.77**	**71.38**	78.28	**52.01**	**74.10**	42.79	79.18	76.57	66.71	77.71	49.70	74.99	50.54	**75.50**	59.32	67.87
	Ours (NMS Loss + AAlign)	**80.15**	**67.80**	**77.69**	54.26	**69.54**	81.48	71.34	78.97	51.76	73.61	**42.82**	**79.80**	**76.44**	**67.68**	**78.16**	**50.43**	**75.06**	50.99	75.31	**59.66**	**68.15**
AP	CASENet	50.53	44.88	41.69	28.92	42.97	54.46	47.39	58.28	35.53	45.61	25.22	56.39	48.45	42.79	55.38	27.31	48.69	39.88	45.05	34.77	43.71
	CASENet-S	67.64	53.10	69.79	40.51	62.52	73.49	63.10	75.26	39.96	60.74	30.43	72.28	65.15	56.57	70.80	33.91	61.92	45.09	67.87	48.93	57.95
	SEAL	74.24	57.45	72.72	42.52	65.39	74.50	65.52	**77.93**	40.92	65.76	**33.36**	76.31	68.85	58.31	73.76	38.87	66.31	**46.93**	69.40	**51.40**	61.02
	Ours (NMS Loss)	75.85	59.65	**74.29**	**43.68**	65.65	**77.63**	67.22	76.63	**42.33**	**70.67**	31.23	77.66	74.59	61.04	77.44	38.28	69.53	40.84	**71.69**	50.39	62.32
	Ours (NMS Loss + AAlign)	**76.74**	**60.94**	73.92	43.13	**66.48**	77.09	**67.80**	77.50	42.09	70.05	32.11	**78.42**	**74.77**	**61.28**	**77.52**	**39.02**	**68.51**	41.46	71.62	51.04	**62.57**

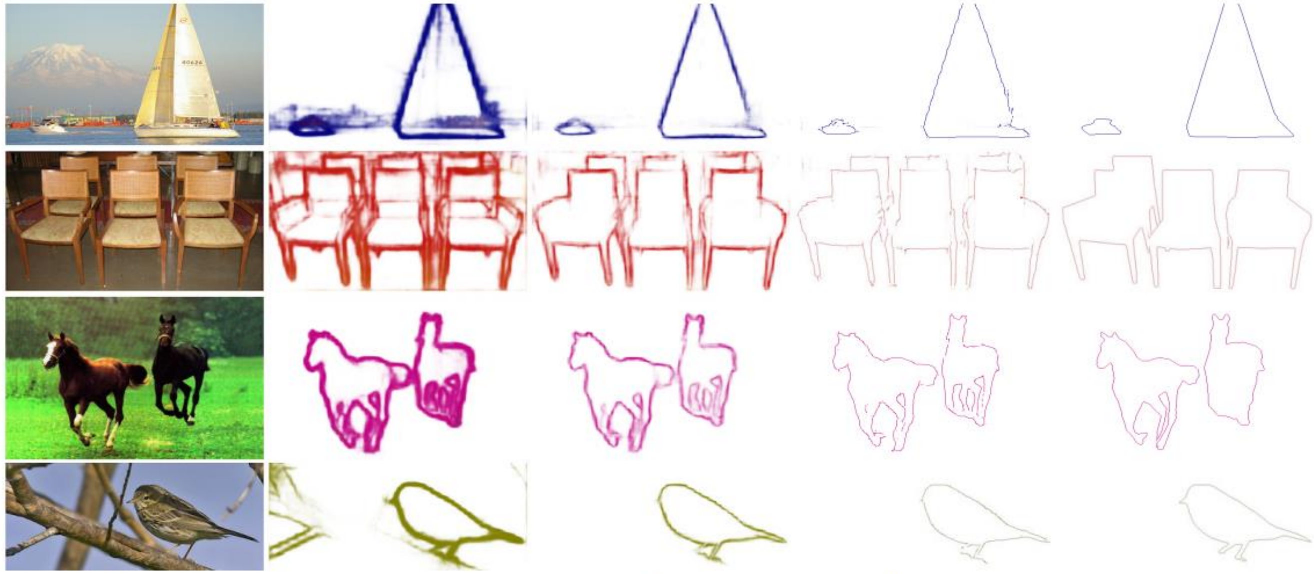
2. Performance of the Active Aliment

Metric	Method	Test NMS	Or. Test Set	Re-annot. Test Set
MF (ODS)	CASENet		62.21	63.52
	Ours (CASENet)		63.20	64.03
	Ours (CASENet)	✓	64.84	66.58
	+ NMS Layer		64.15	64.99
	+ NMS Layer	✓	**65.93**	67.87
	+ Active Align	✓	64.83	**68.15**
AP	CASENet		42.99	43.71
	Ours (CASENet)		34.60	45.60
	Ours (CASENet)	✓	44.83	60.48
	+ NMS Layer		53.67	54.18
	+ NMS Layer	✓	**60.10**	62.32
	+ Active Align	✓	57.98	**62.57**

Table 3: Effect of the NMS Loss and Active Alignment on the SBD dataset. Score (%) represents mean over all classes.

Experiments

3. Qualitative Results and Coarse-to-fine Segmentation Score



(a) Image (b) CASENet (c) Ours (d) +Thinning Layer (e) Ground-truth

Figure 3: Qualitative Results on the SBD Dataset.

Figure 4: Active Alignment. From Left-to-right (GT, Refined).

Figure 5: Comparison of our boundaries vs those obtained from DeepLab v3+'s segmentation masks. We perform 4.2% better at the strictest regime.

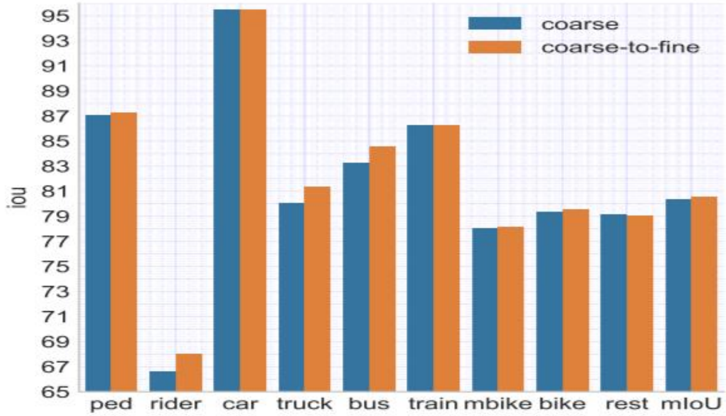


Figure 6: Semantic Segmentation on Cityscapes val: Performance of DeepLab V3+ when trained with fine data and (blue) vanilla train_extra set, (orange) our refined data (8 object classes) from train_extra. We see improvement of more than 1.2 IoU % in rider, truck and bus.

Thank you