DeepStrip: High Resolution Boundary Refinement (Supplementary Material)

A. Appendix

A.1. Baseline Details

In Section 4 of the paper, we compare our method to prior approaches. Here we give specific details on how we applied each prior work.

• **Bilinear Upsampling**: Directly bilinearly upsampling the low resolution mask to high resolution. The hard mask is obtained by the optimal threshold the from soft mask and the boundary is obtained by taking the gradient of the upsampled hard mask.

• **Grabcut [7]**: We apply grabcut given the upsampled low resolution, and use the boundary mask for evaluation.

• **Dense CRF [5]**: A non-learning approach based on conditional random field of the nearby pixels. Given the upsampled mask and High Resolution (HR) image, we apply dense CRF to refine the mask. The boundary mask is obtained from the gradient of the predicted mask.

• **Bilateral Solver [2]**: A edge-aware smoothing algorithm with fast and robust optimization. We use the publicly released code for evaluation. We provide the upsampled mask as the reference image. The hard mask is obtained by the optimal threshold from the soft mask.

• **JBU [4]**: A Joint Bilateral Upsampling (JBU) algorithm which upsamples the source image taking into account the reference image jointly. We use the contributed opencv function for evaluation. We take the Low Resolution (LR) image as the source image and jointly upsample to obtain the output. The hard mask is obtained by the optimal threshold from the soft mask.

• **Deep GF [9]**: A learnable guided filtering approach that performs pixel-wise image prediction. We use the released code for evaluation [1] and we use radius 1 for testing.

• **Guided Filtering [3]**: The original guided filter approach. We use the built-in opencv function for evaluation.

• **Curve-GCN [6]**: A GCN based approach which aims to predict the control points of the contour and fit curve to obtain the final boundary. Instead of random initialization, we provide the upsampled contours as initialization to train the network. The input size is 512×512 and HR prediction is made by upsampling from LR prediction as the whole boundary region is required for prediction.

• **DELSE [8]**: A level-set based approach with extreme points as initialization. We use the released code for evaluation. Since a ground truth hi-res mask is not available at inference time, instead of extracting extreme points from ground truth mask, we use the upsampled LR mask to extract extreme points for evaluation. The input dimension is 1024×1024 and predictions on PixaHR are made in low resolution and upsampled to original resolution. We report the optimal threshold for evaluation. The original DELSE setting with ground truth extreme points is also shown in Table 1.

• **STEAL [1]**: A semantic boundary refinement approach which adds a thinning layer and active alignment to refine boundaries from coarse to fine. We use the public released code and model [2] and we follow the default patch-by-patch testing with patch size 512 for evaluation.

• **U-Net Boundary**: Since it is difficult to implement in the whole image the boundary distance and C0 continuity loss, which are applied in strip domain, we only apply common edge detection loss as in [10]. As a result, the predicted boundaries are thick with high recall and low precision in Boundary-based F score. (See results in Figure 4 and Figure 3)

A.2. Additional Ablation Analysis

We provide additional ablation results in Table 1. To determine the width of the strip, besides multiplying the pixel number in LR mask with the scale factor, we slightly increase the width further by a factor from 1 to 2. Comparing among Ours 1, Ours 1.5 and Ours 2, the performance changes by a small margin under different factors. We report Ours 1.5 in the main result as a trade off between performance and computation. Additionally, since our strip reconstruction step uses image gradient as part of energy function, we conduct experiment which only uses image gradient to find the minimum path in the strip reconstruction step. As shown in Table 1, the performance degrades by a large margin if we only use gradient (Strip + gradient) because spurious boundaries will be included, indicating the effectiveness of our learning based approach.

[1] https://github.com/wuhuikai/DeepGuidedFilter

[2] https://github.com/nv-tlabs/STEAL

Dataset	DAVIS 2016	PixaHR 16×
Metrics	F(0 pix)	F(1 pix)
DELSE original	0.275	0.082
Strip + gradient	0.165	0.295
Our 1	0.414	0.392
Our 2	0.416	0.415
Ours 1.5	0.423	0.396

Table 1. Ablation analysis on two datasets. Each entry is the boundary-based F score tested on individual dataset.

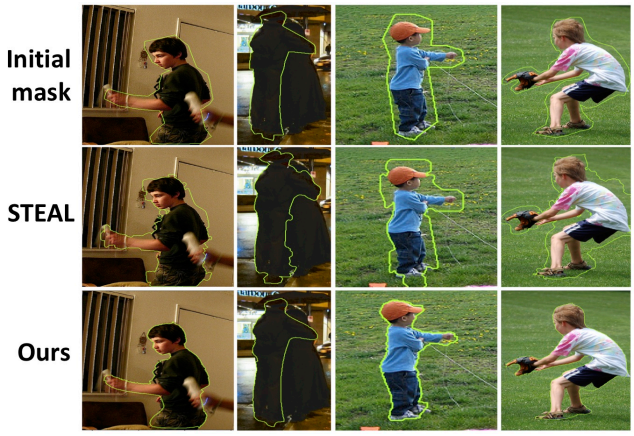

Figure 1. Less accurate examples from COCO.

A.3. Qualitative Results of Loss Function

Figure 2 compares the results with different losses. With only weighted $l1$ and dice loss, spurious boundaries are not suppressed and thus false positive exists. With the introduction of selection layer, the network select target boundaries from all potential ones so that spurious boundaries get ignored. Additionally, a closer prediction is observed with boundary distance loss. Lastly, with the introduction of C0 continuity and matching loss, a better result is obtained.

A.4. Additional Qualitative results

Figure 3 and Figure 4 show the results among baselines in multiple regions. It is clear that our method achieves more accurate results than the baselines. In particular, our approach have smoother boundaries than U-Net boundary and less false positive than DELSE and bilateral solver. More visualization examples are displayed in Figure 5, Figure 6 and Figure 7. Less accurate initial mask results on COCO is shown in Figure 1.

References

[1] David Acuna, Amlan Kar, and Sanja Fidler. Devil is in the edges: Learning semantic boundaries from noisy annotations. In *CVPR*, 2019. 1

[2] Jonathan T Barron and Ben Poole. The fast bilateral solver. In *ECCV*, 2016. 1

[3] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *TPAMI*, 2012. 1

[4] Johannes Kopf, Michael F Cohen, Dani Lischinski, and Matt Uyttendaele. Joint bilateral upsampling. In *ToG*, 2007. 1

[5] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NeurIPS*, 2011. 1

[6] Huan Ling, Jun Gao, Amlan Kar, Wenzheng Chen, and Sanja Fidler. Fast interactive object annotation with curve-gcn. In *CVPR*, 2019. 1

[7] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *TOG*, 2004. 1

[8] Zian Wang, David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler. Object instance annotation with deep extreme level set evolution. In *CVPR*, 2019. 1

[9] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *CVPR*, 2018. 1

[10] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *CVPR*, 2015. 1

Figure 2. Visualization of loss components. Rows from top to bottom are original images, U-Net strip + dice which use weighted $l1$ and dice loss, U-Net strip + dice + selection, U-Net strip + dice + selection + SD, Ours and Ground truth.

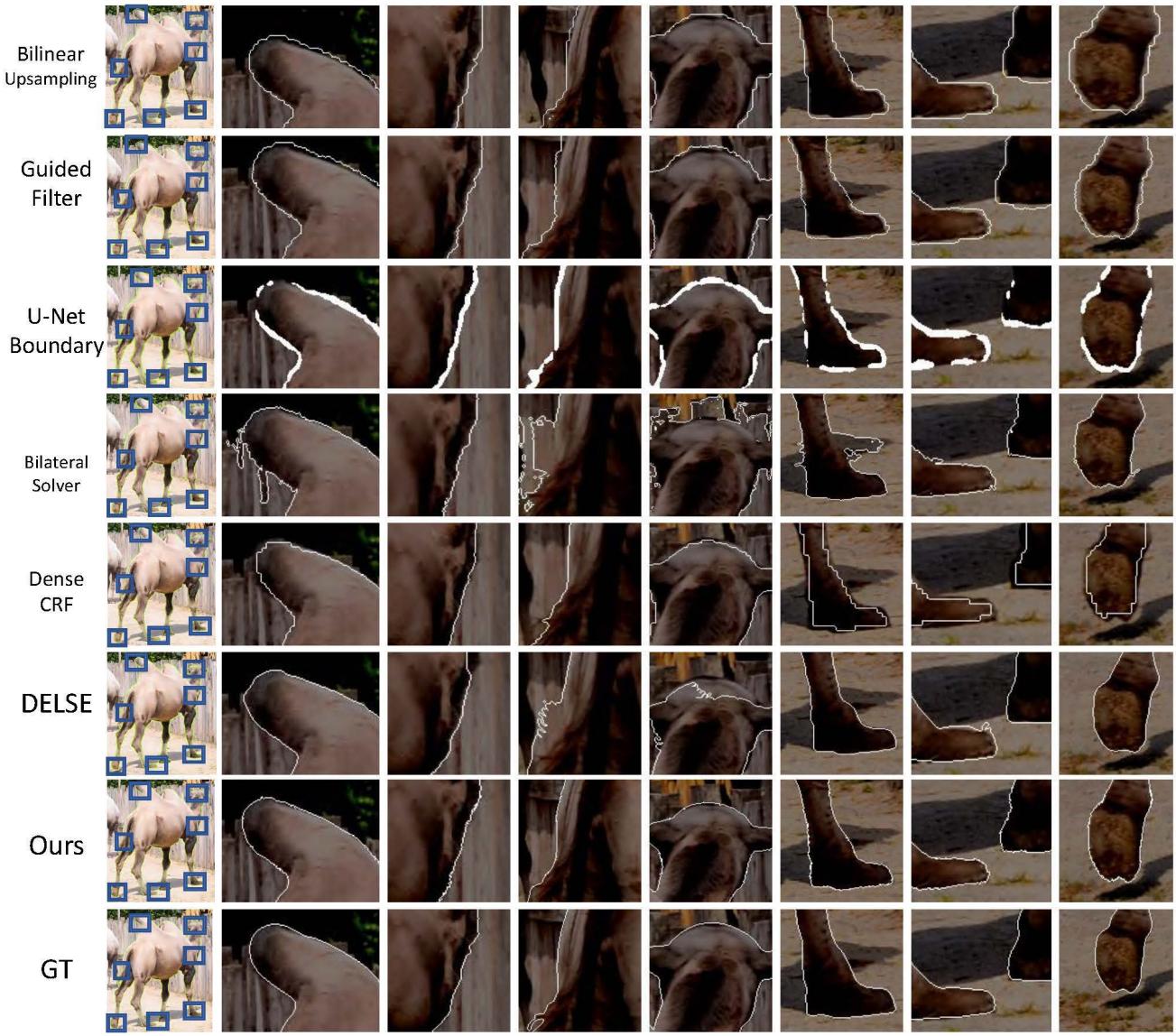

Figure 3. Multi-region visualization on DAVIS 2016. The first column shows the whole image and the rest columns are the enlarged box regions. Boundaries are highlighted in white. Notice that our approach has closer prediction than methods like bilinear upsampling and guided filter, has less spurious boundaries than DELSE and bilateral solver, and thinner boundaries than U-Net Boundary.

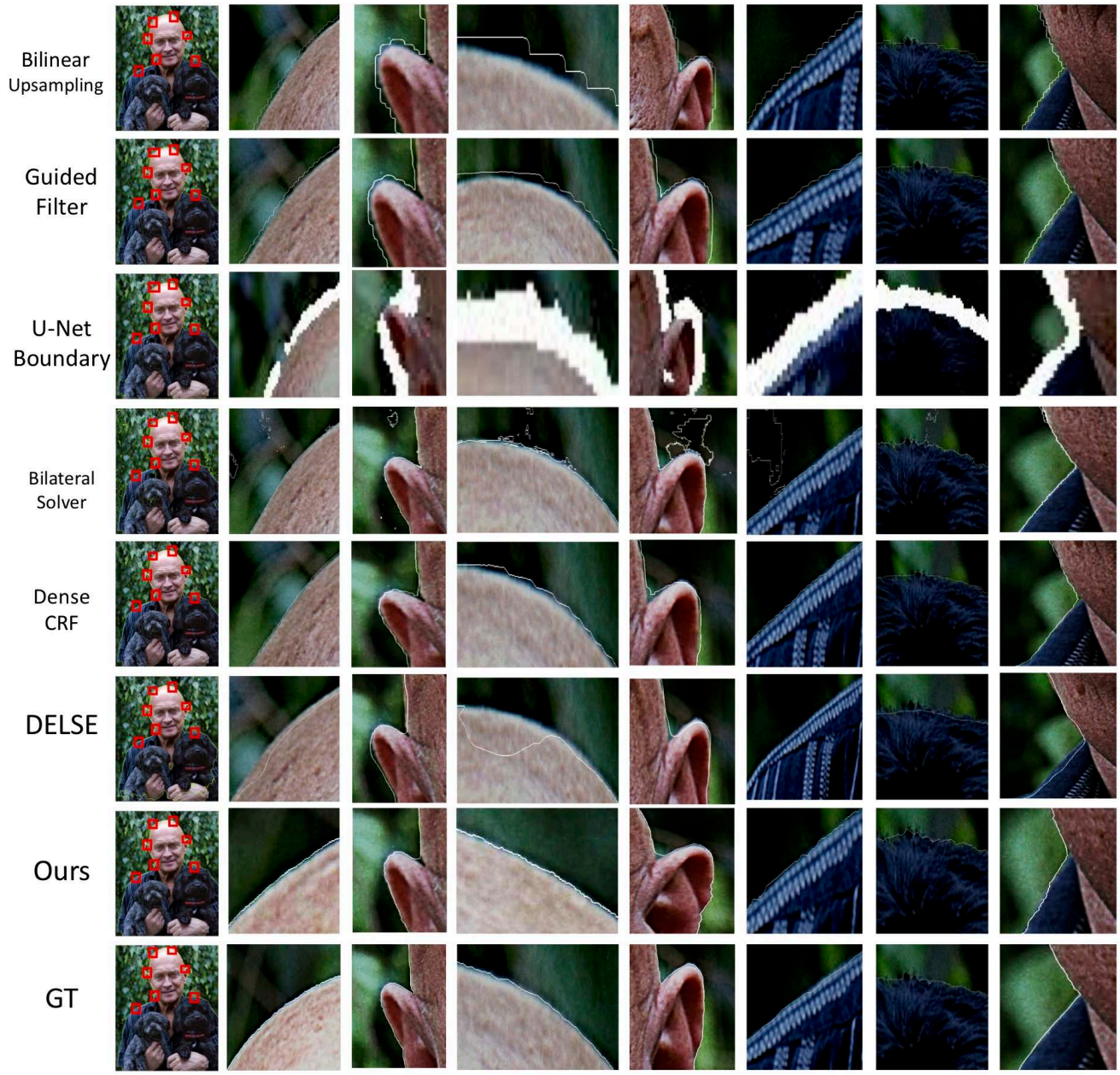

Figure 4. Multi-region visualization on PixaHR. The first column shows the whole image and the rest columns are the enlarged box regions. Boundaries are highlighted in white. Notice that our approach has closer prediction than methods like bilinear upsampling and guided filter, has less spurious boundaries than DELSE and bilateral solver, and thinner boundaries than U-Net Boundary.

Figure 5. Additional visualization on DAVIS 2016. We first show the whole boundary visualization and then show the enlarged box region. The boundaries in the enlarged regions are displayed in white. Notice that for complicated topology, our approach still has better result than the baselines.

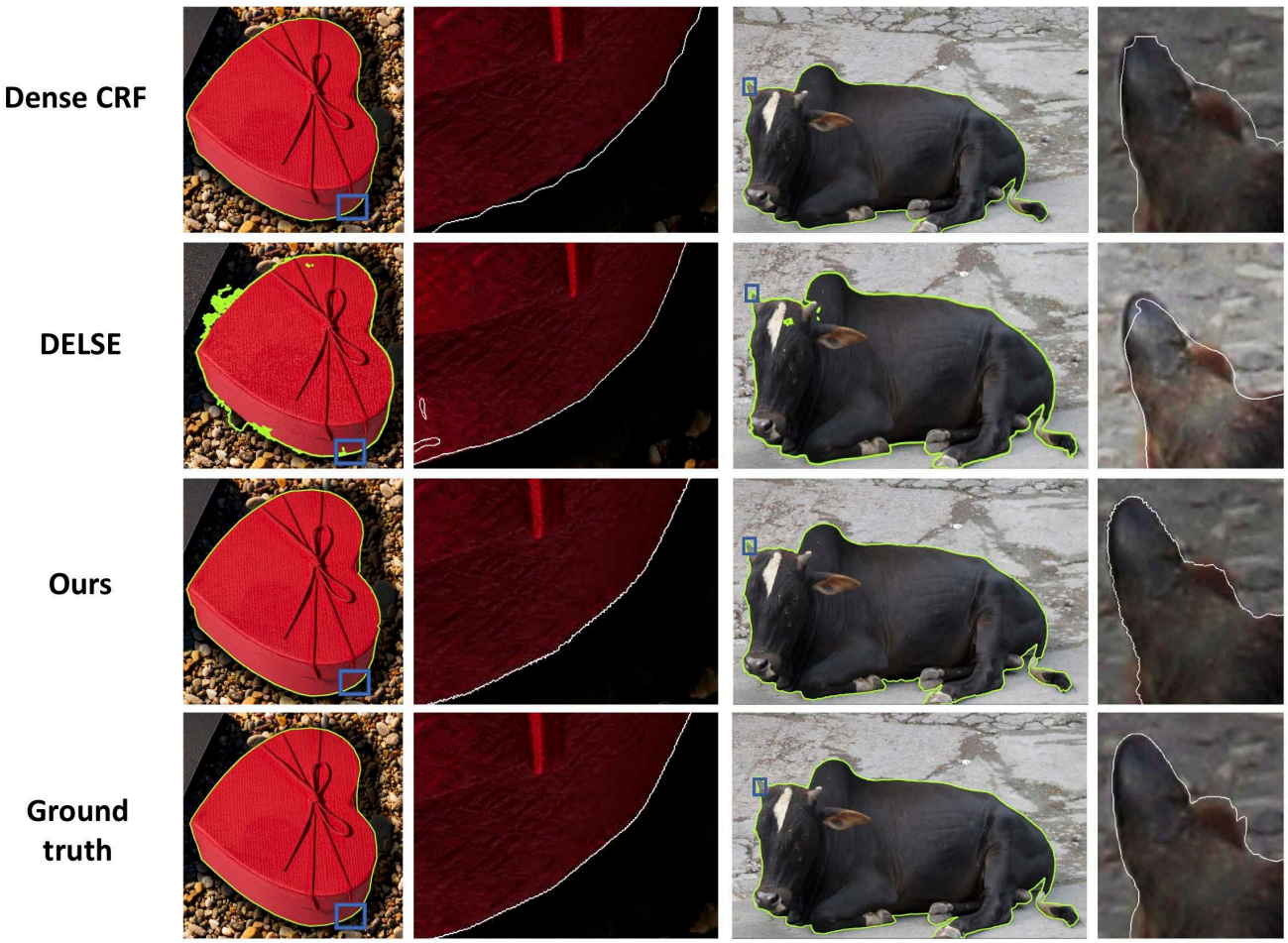

Figure 6. Additional visualization on PixaHR 32×. We first show the whole boundary visualization and then show the enlarged box region. The boundaries in the enlarged regions are displayed in white. Notice that our approach makes smoother prediction than dense CRF and less false positive than DELSE.

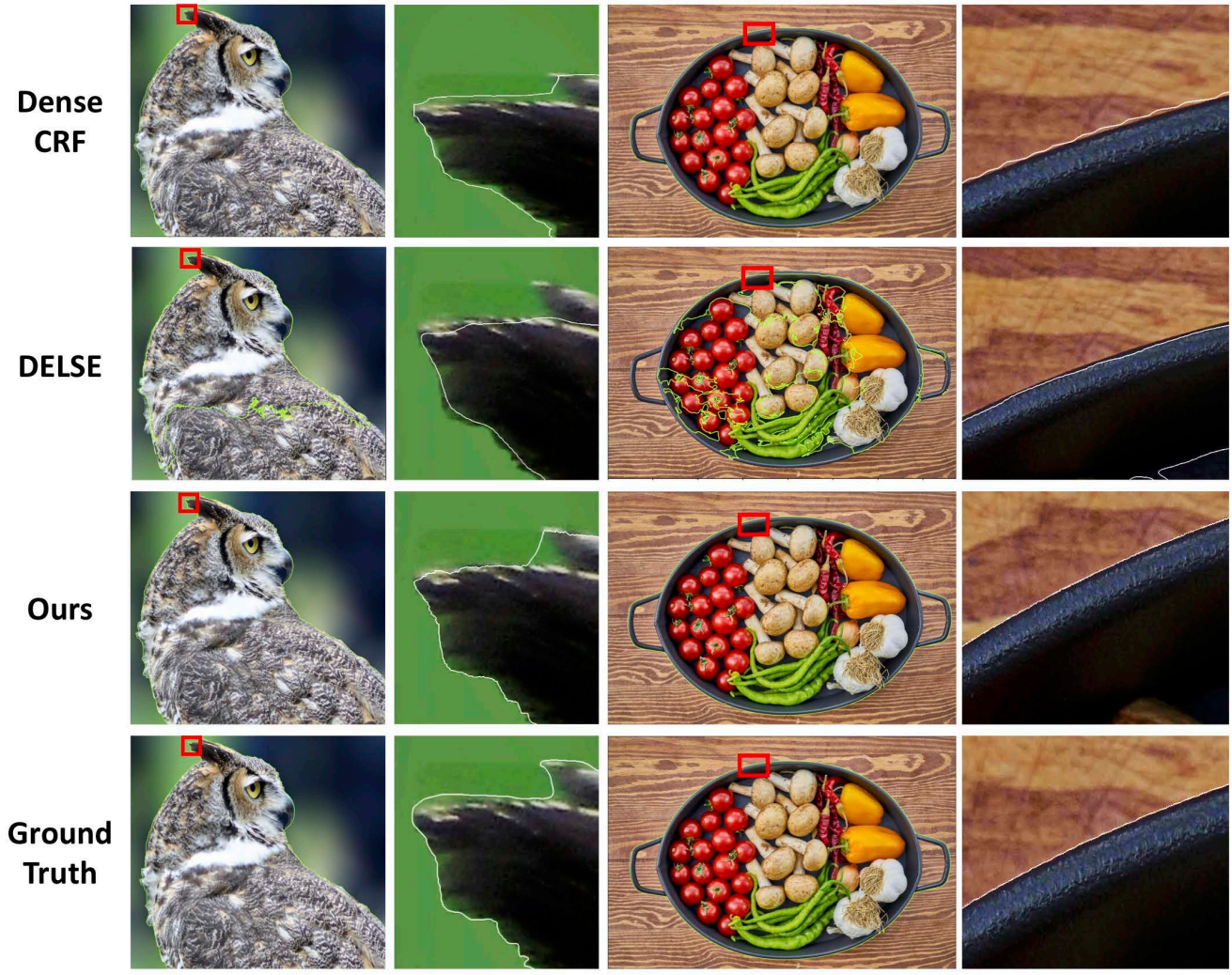

Figure 7. Additional visualization on PixaHR 16×. We first show the whole boundary visualization and then show the enlarged box region. The boundaries in the enlarged regions are displayed in white. Notice that our approach makes smoother prediction than dense CRF and less false positive than DELSE.