Interactive Object Segmentation with Inside-Outside Guidance

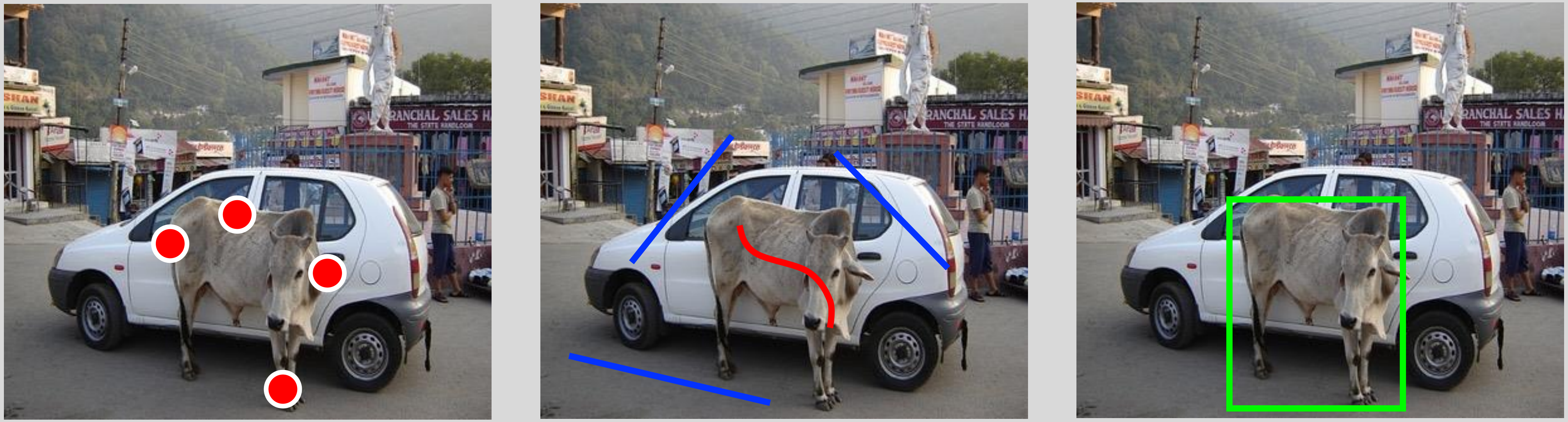
Shiyin Zhang[1], Jun Hao Liew[2], Yunchao Wei[3], Shikui Wei[1], Yao Zhao[1]
[1]Beijing Jiaotong University [2]National University of Singapore [3]University of Technology Sydney

- Manual data annotation is costly and time-consuming

Polygon
≈60s per instance

Segmentation target

Interactive Object Segmentation



Sparse clicks | Scribbles | Bounding box

- Extraction of target object given some user inputs (*e.g.* points, scribbles, bounding box)

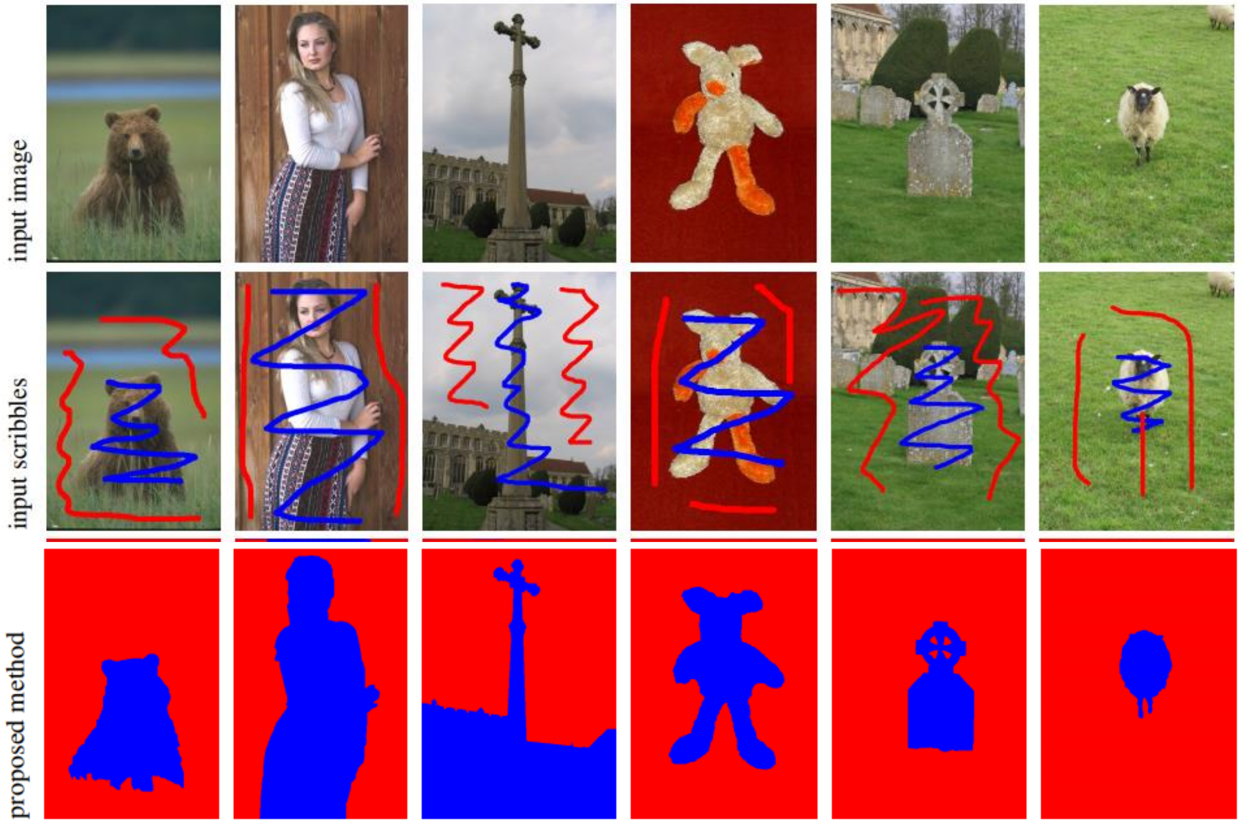
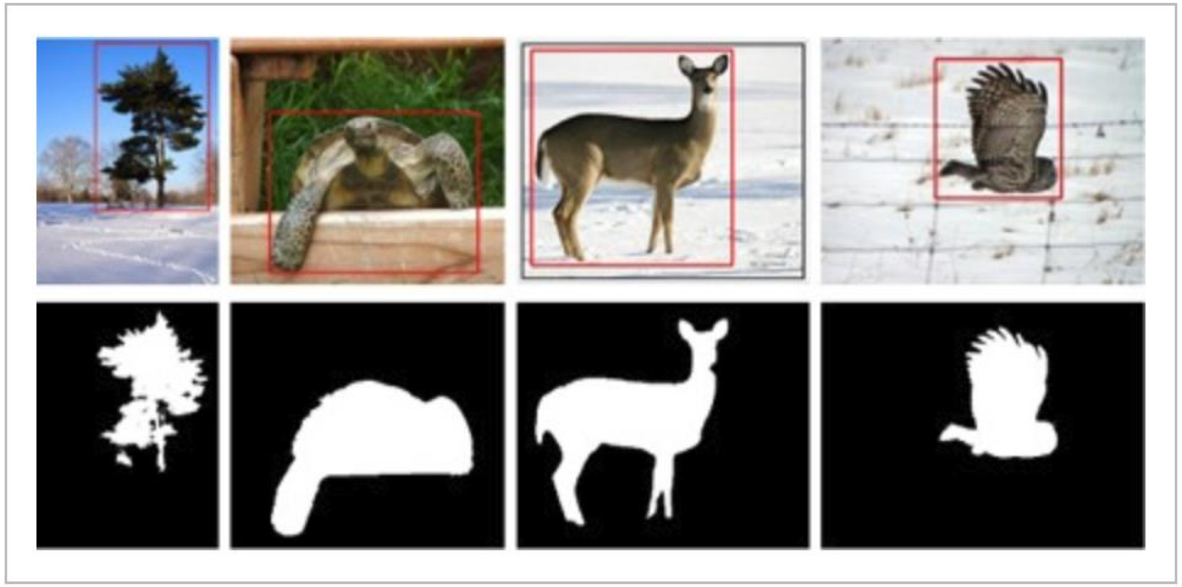
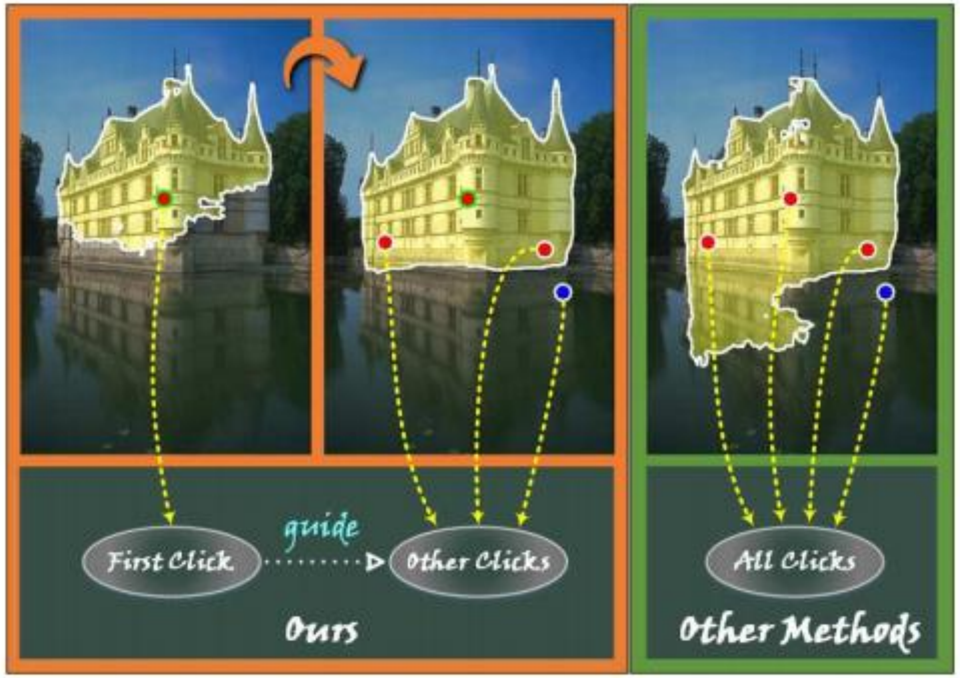


Figure 6. Illustrative results on user scribbles with large amount of errors.

Error-tolerant Scribbles Based Interactive Image Segmentation.cvpr2014



Densecut: Densely connected crfs for realtime grabcut. Comput. Graph. Forum.2015



Interactive Image Segmentation with First Click Attention.cvpr2020

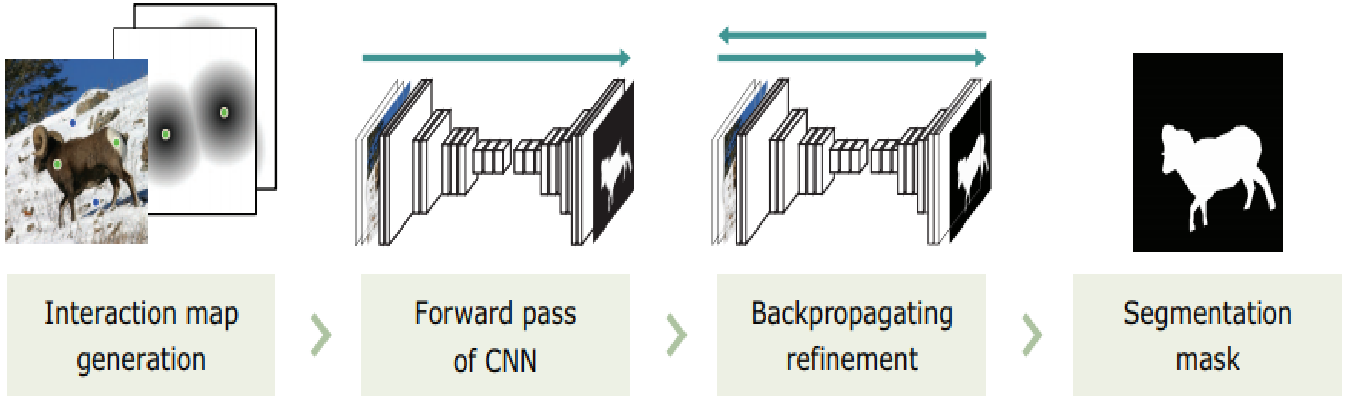


Figure 1. Overview of the proposed algorithm: we perform this segmentation process again when a user provides a new annotation.

Interactive Image Segmentation via Backpropagating Refinement Scheme.cvpr2020

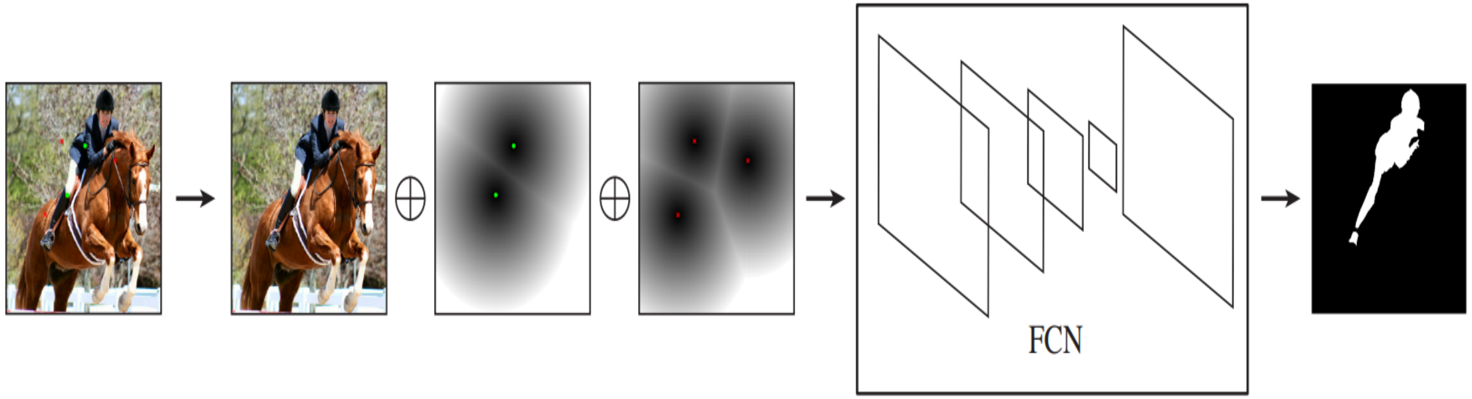
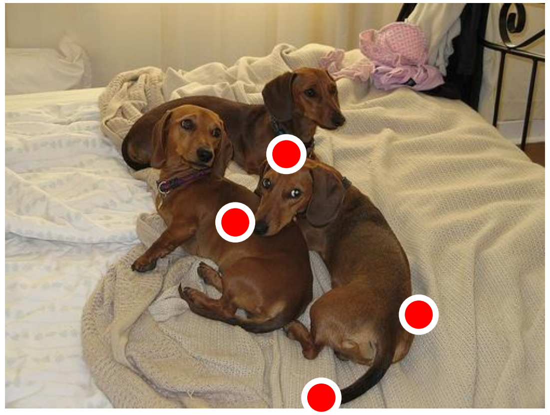


Figure 1: The framework of learning our FCN models. Given an input image and user interactions, our algorithm first transforms positive and negative clicks (denoted as green dots and red crosses respectively) into two separate channels, which are then concatenated (denoted as ⊕) with the image's RGB channels to compose an input pair to the FCN models. The corresponding output is the ground truth mask of the selected object.

Deep Interactive Object Selection.cvpr2016

Existing State-of-the-Art Method

- DEXTR (Deep Extreme Cut)
  - Take 4 extreme points (top, bottom, leftmost and rightmost pixels) as inputs
  - Problems:
    - *Confusing annotation*:
      - **Multiple extreme points appear at similar location**
      - **Unrelated object lying inside the target object**



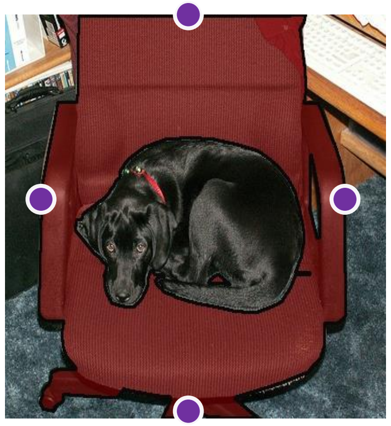
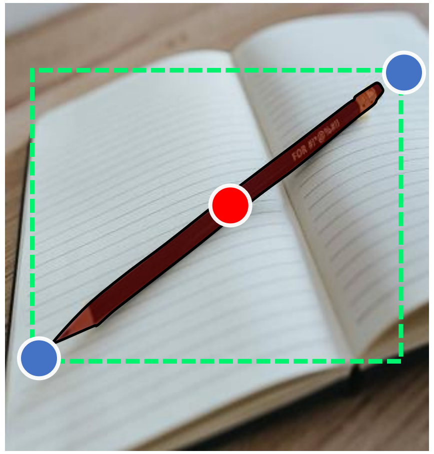
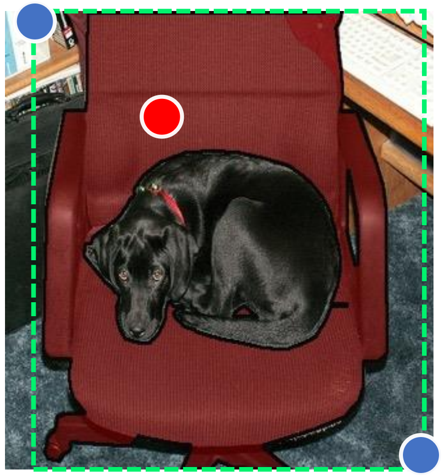


Figure credit: Maninis et al. "Deep Extreme Cut: From Extreme Points to Object Segmentation", CVPR 2018.

Inside-Outside Guidance (IOG)

- Inside-Outside Guidance (3 clicks)
  - *Inside guidance* (1 click)
    - Interior point located roughly at the object center
    - Disambiguate the segmentation target
  - *Outside guidance* (2 clicks)
    - 2 corner clicks of a box enclosing the object
    - Indicate the background region
    - The remaining 2 corners can be inferred automatically





Clicking Paradigm

- Steps:
  (1) Click on a corner point
  (2) Click on the symmetrical corner
  (3) Click on the object center

The vertical and horizontal guided lines
are used to make the box visible

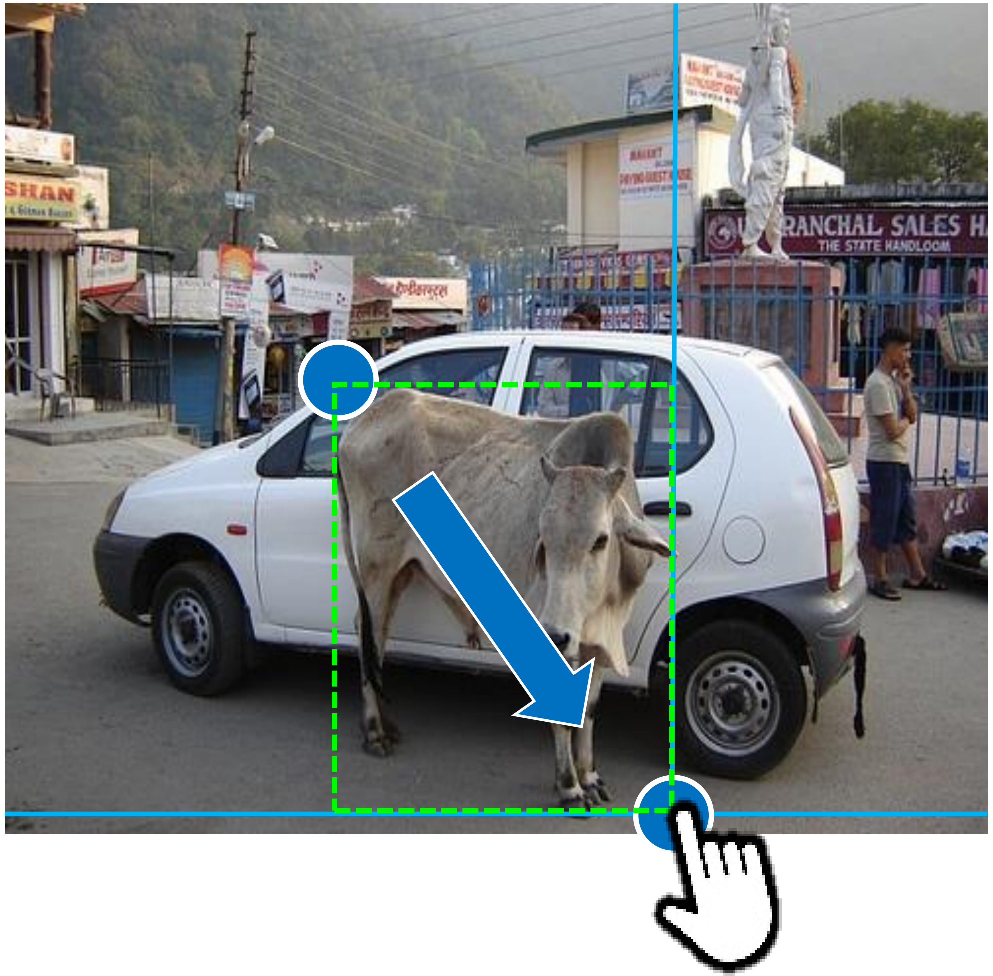
Clicking Paradigm

- Steps:
  (1) Click on a corner point
  (2) Click on the symmetrical corner
  (3) Click on the object center

The vertical and horizontal guided lines are used to make the box visible

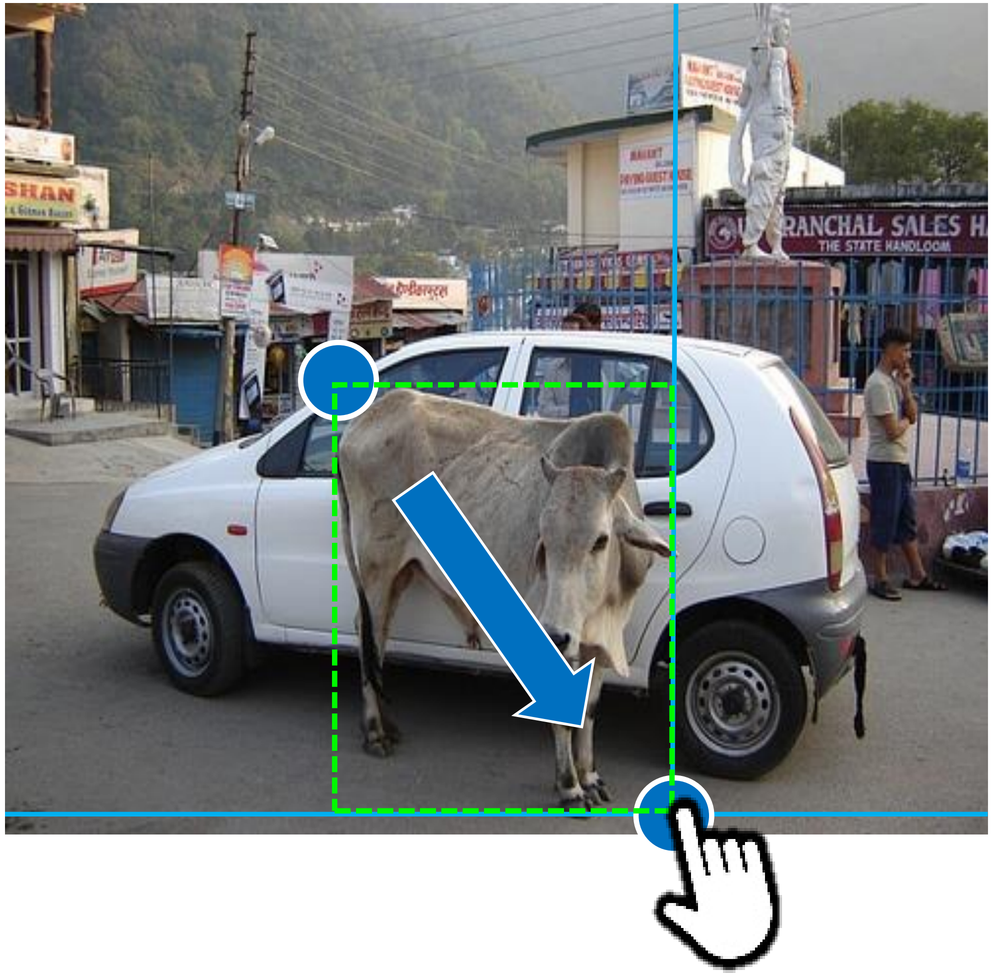
Clicking Paradigm

- Steps:
  (1) Click on a corner point
  (2) Click on the symmetrical corner
  (3) Click on the object center



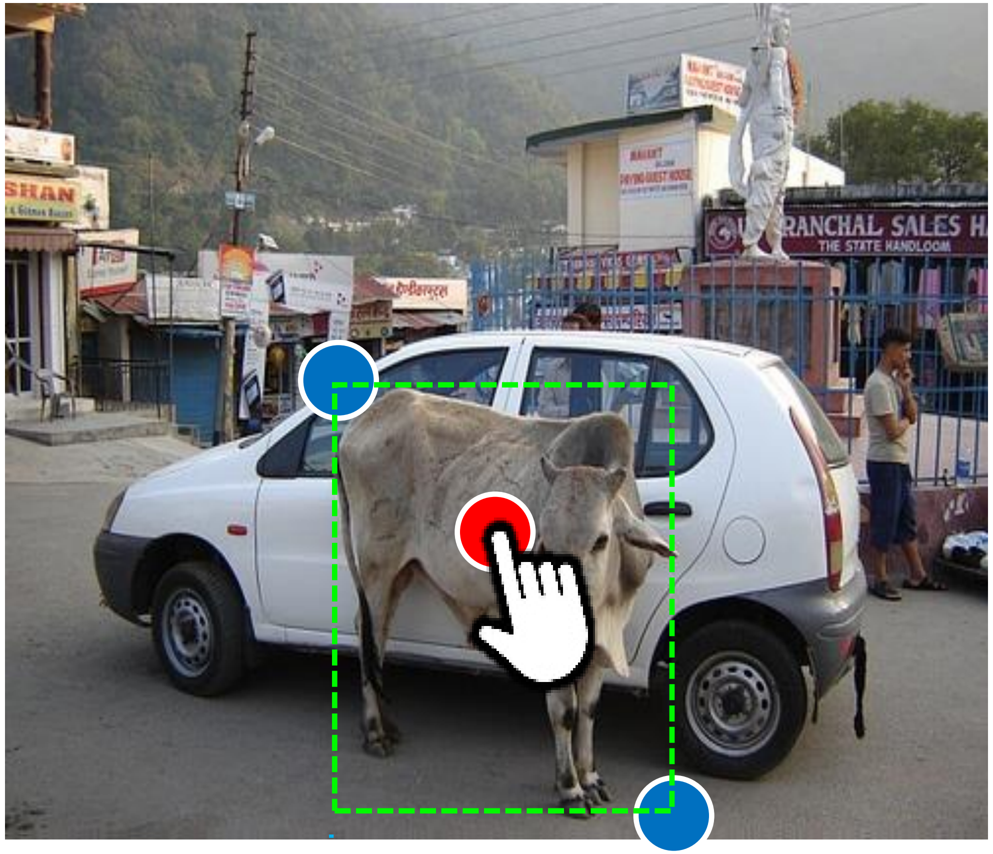
Clicking Paradigm

- Steps:
  1. (1) Click on a corner point
  2. (2) Click on the symmetrical corner
  3. (3) Click on the object center



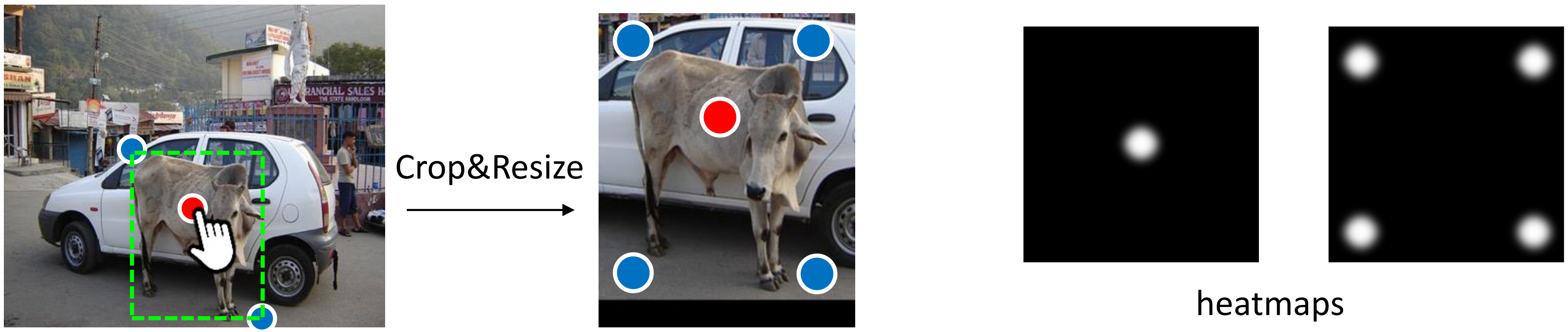
Clicking Paradigm

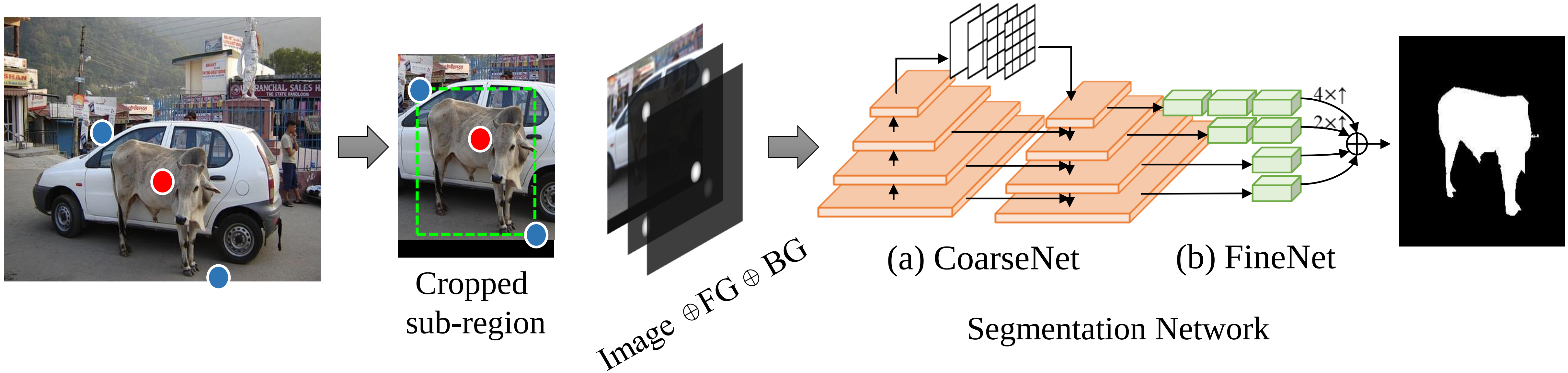
- Steps:
  (1) Click on a corner point
  (2) Click on the symmetrical corner
  (3) Click on the object center



Input Representation

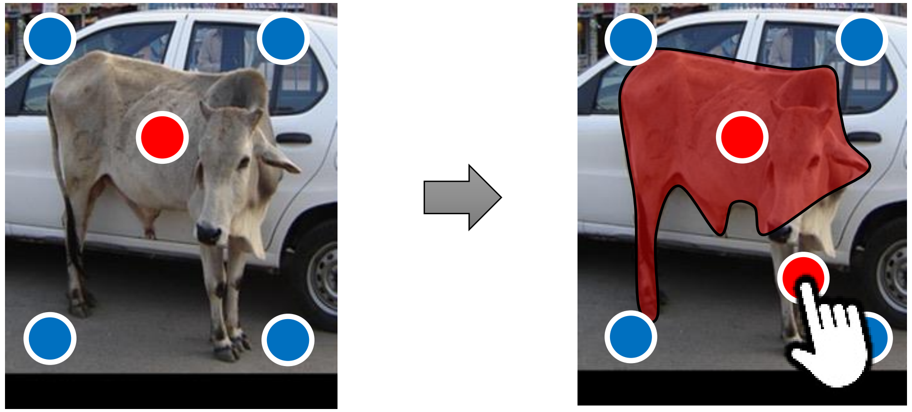
- Follow the practice of DEXTR
  - Enlarge the bounding box by 10 pixels to include context
  - Crop and resize the inputs to 512x512
- Input representation
  - 2 separate Gaussian heatmaps for the inside and outside clicks

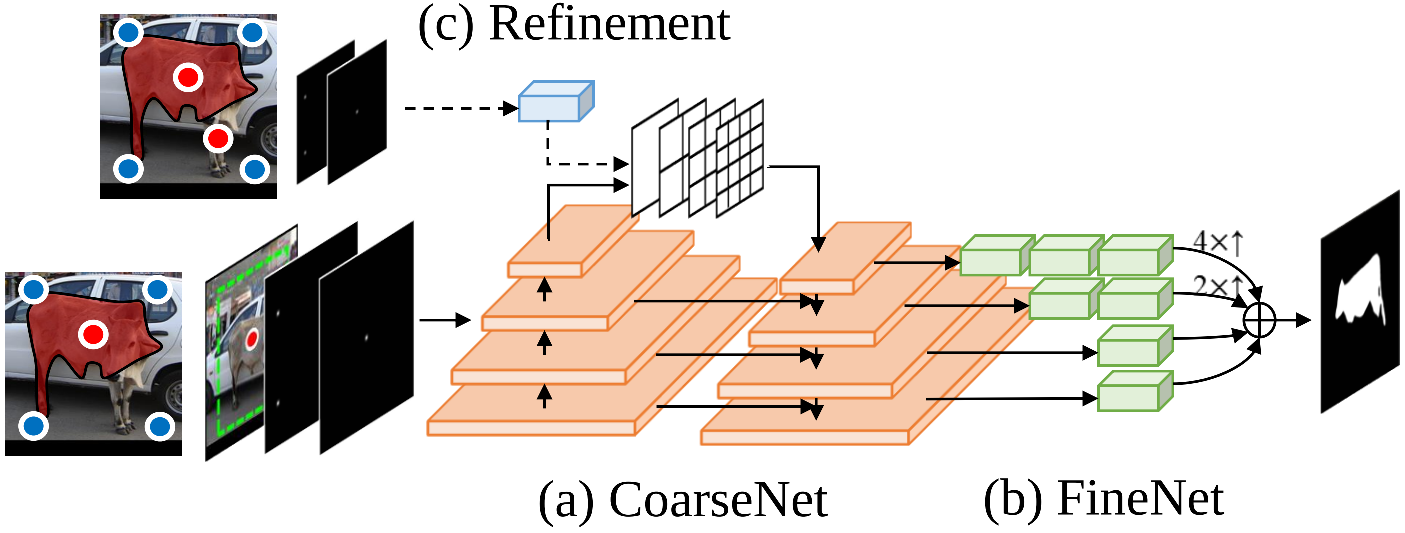


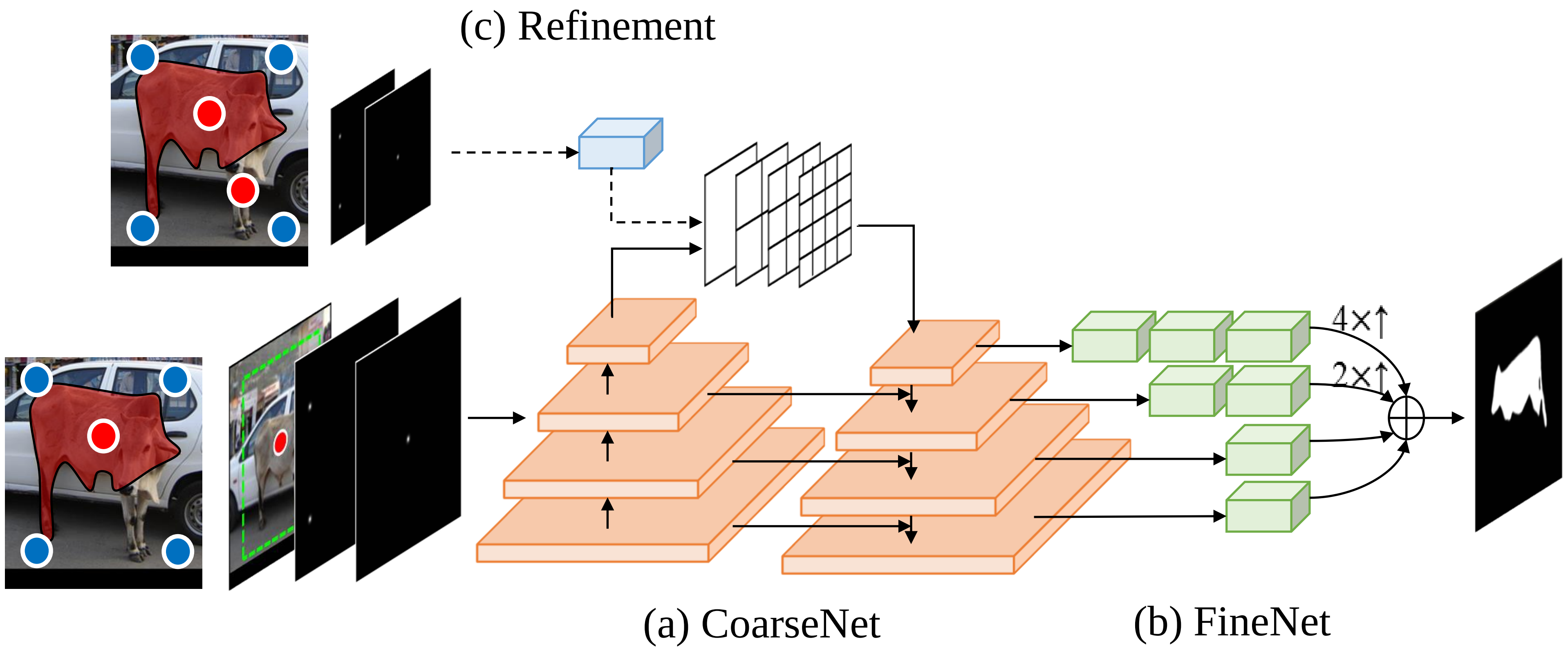

Cropped
sub-region
Image ⊕FG ⊕ BG
4×↑
2×↑
(a) CoarseNet
(b) FineNet
Segmentation Network

Beyond Three Clicks

- Our IOG naturally supports interactive adding of new clicks
- Add a lightweight branch to accept additional inputs
- Train with iterative training strategy

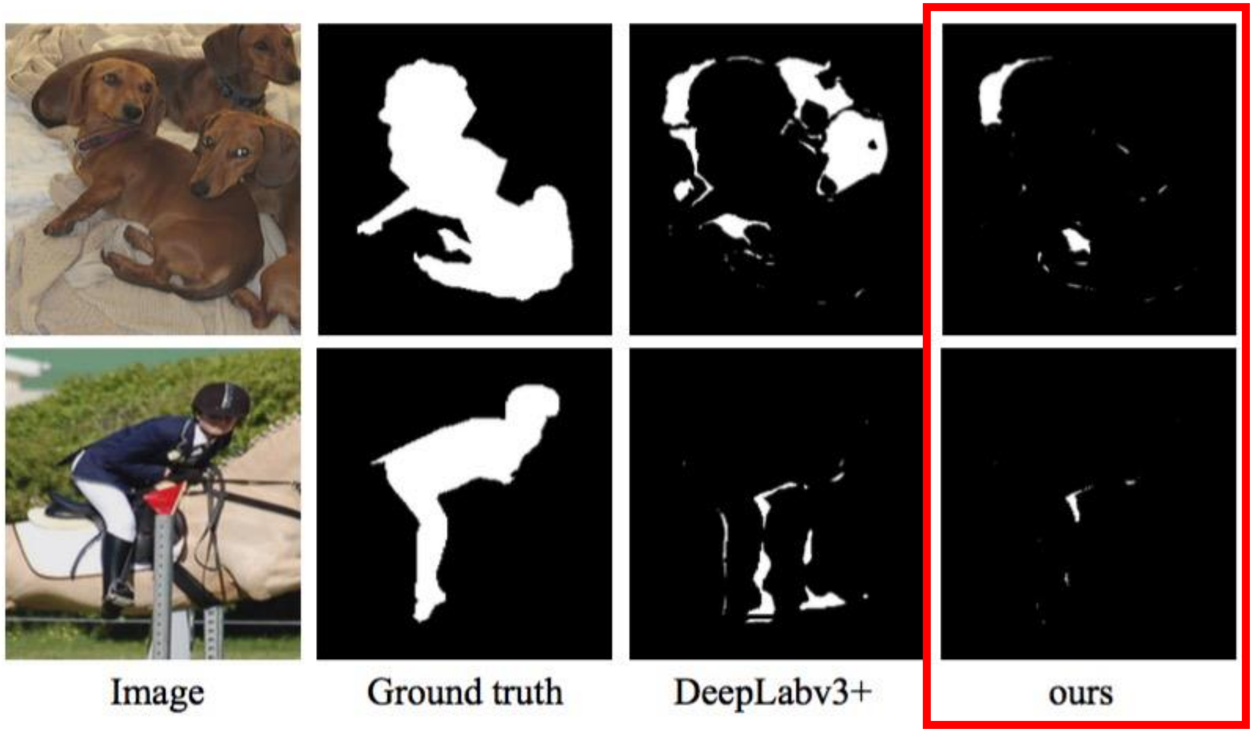




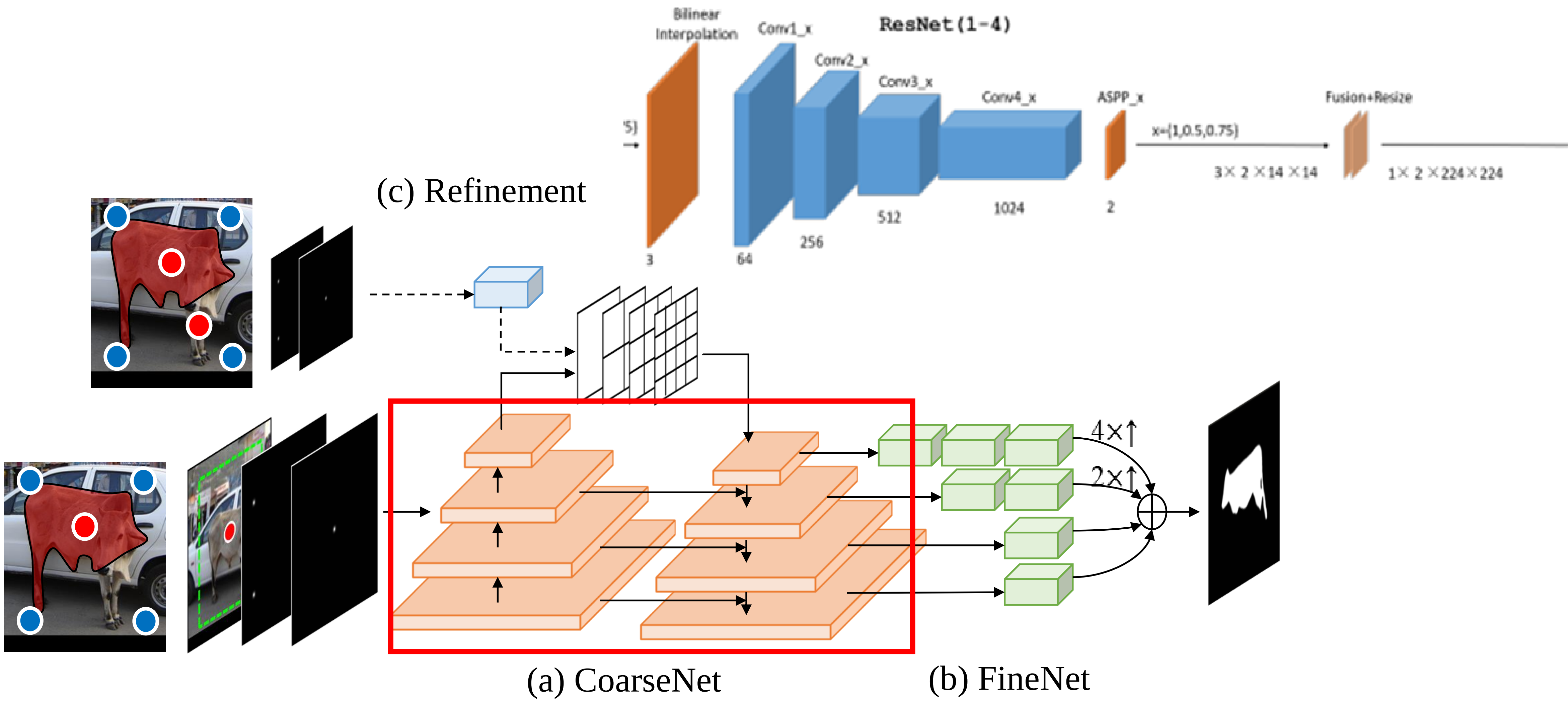

(c) Refinement
4×↑
2×↑
(a) CoarseNet
(b) FineNet

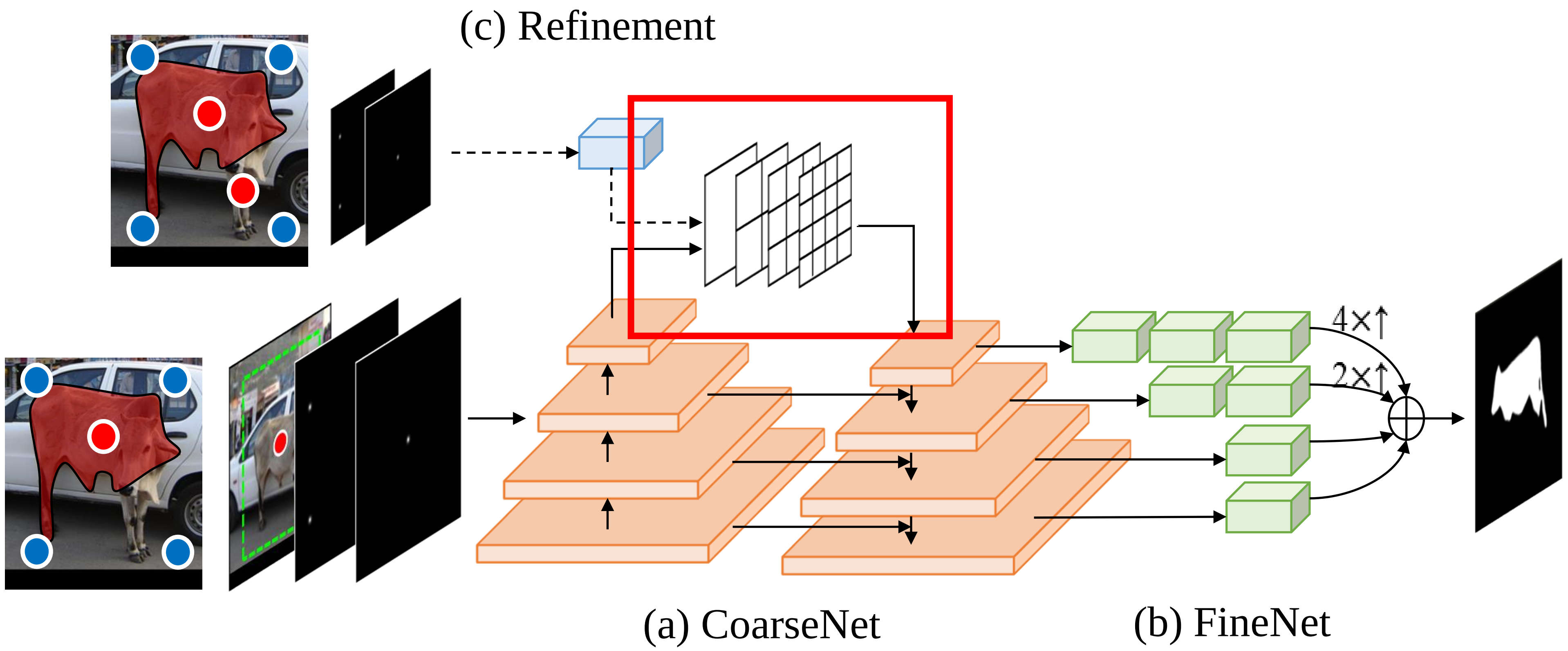
Segmentation Network

- Segmentation errors mostly occur around the object boundaries
- Use a coarse-to-fine network structure

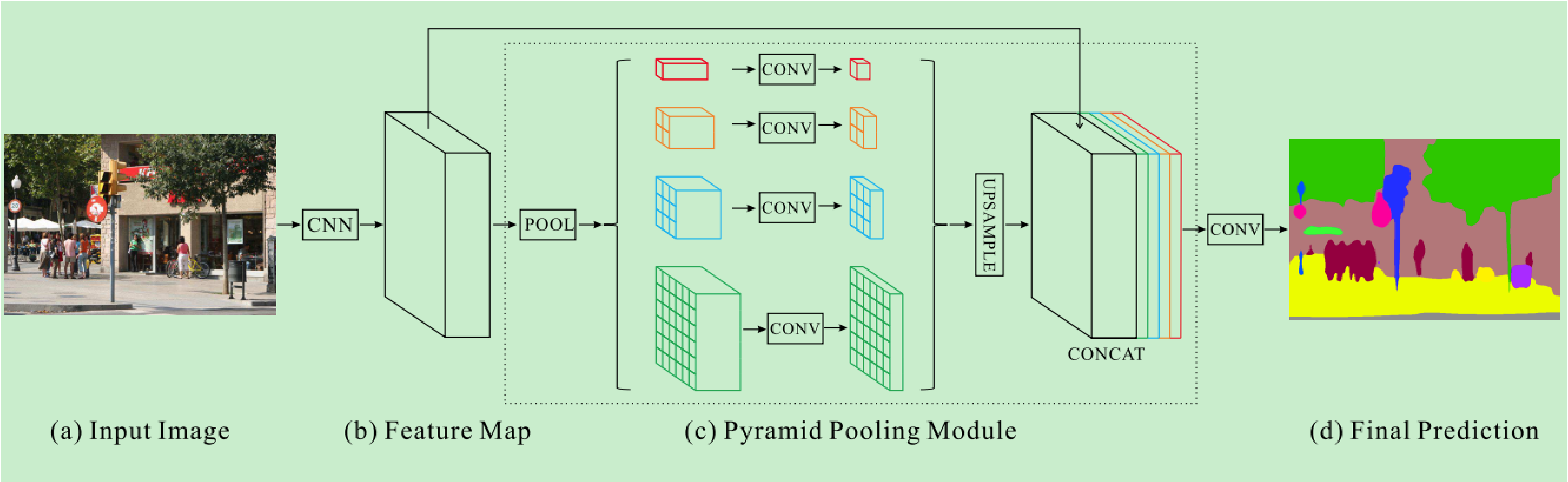


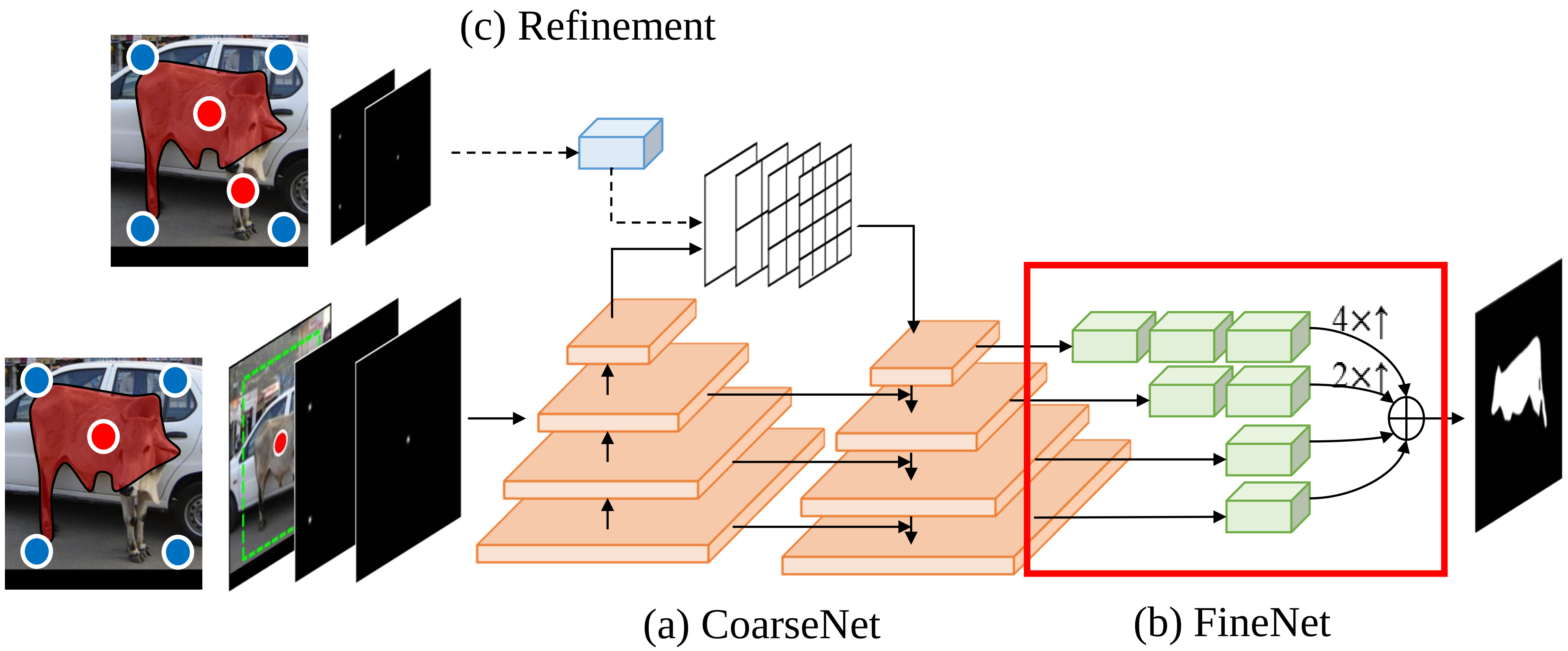
The coarse-to-fine structure is similar to : Yilun Chen et al. "Cascaded pyramid network for multi-person pose estimation", CVPR 2018.


Bilinear Interpolation
Conv1_x
ResNet(1-4)
Conv2_x
Conv3_x
Conv4_x
ASPP_x
Fusion+Resize
x={1,0.5,0.75}
3× 2 ×14 ×14
1× 2 ×224×224
3
64
256
512
1024
2
(c) Refinement
4×↑
2×↑
(a) CoarseNet
(b) FineNet



(c) Refinement
4×↑
2×↑
(a) CoarseNet
(b) FineNet

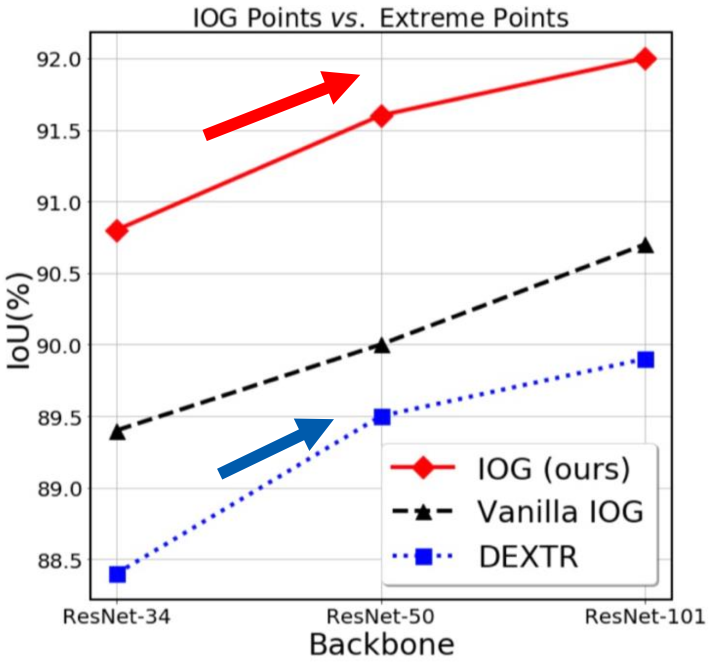

CNN
POOL
CONV
CONV
CONV
CONV
UPSAMPLE
CONCAT
CONV
(a) Input Image
(b) Feature Map
(c) Pyramid Pooling Module
(d) Final Prediction



(c) Refinement
4×↑
2×↑
(a) CoarseNet
(b) FineNet

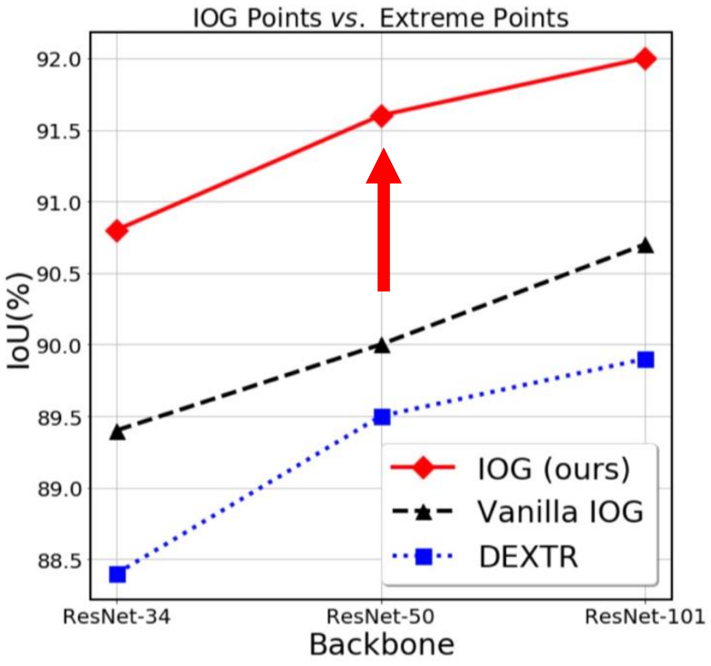
IOG vs. Extreme Clicks

- Observation:
  (1) IOG is more effective than extreme points across different backbone



IOG vs. Extreme Clicks

- Observation:
  (1) IOG is more effective than extreme points across different backbone
  (2) Using a coarse-to-fine network structure further improves the performance



Comparison with SOTA

Methods	Number of Clicks		IoU(%) @ 4 clicks	
	PASCAL@85%	GrabCut@90%	PASCAL	GrabCut
Graph cut [5]	> 20	> 20	41.1	59.3
Random walker [23]	16.1	15	55.1	56.9
Geodesic matting [2]	> 20	> 20	45.9	55.6
iFCN [66]	8.7	7.5	75.2	84.0
RIS-Net [38]	5.7	6	80.7	85.0
DEXTR [46]	4	4	91.5	94.4
Li et al. [37]	-	4.79	-	-
ITIS [45]	3.4	5.7	-	-
FCTSFN [28]	4.58	3.76	-	-
IOG-ResNet101 (**ours**)	3	3	93.2*	96.3*
IOG-ResNet101 (**ours**)	4	4	94.4	96.9

Table 1. Comparison with the state-of-the-art methods on PASCAL and GrabCut in terms of the number of clicks to reach a certain IoU and in terms of quality at 4 clicks. *denotes the IoU of our IOG given only 3 clicks.

Generalization capability

on unseen classes and across different datasets

- 1. In domain
- 2. Cross domain
  - Object categories
  - Stuff categories

1. In domain

- on ***unseen*** categories

	Train	Test	DEXTR [46]	ours
Unseen Classes	PASCAL	COCO MVal (seen)	79.9%	81.7%
	PASCAL	COCO MVal (unseen)	80.3%	82.1%
Generalization	PASCAL	COCO MVal	80.1%	81.9%
	COCO	COCO MVal	82.1%	85.2%
	COCO	PASCAL	87.8%	91.6%
	PASCAL	PASCAL	89.8%	93.2%

Table 2. Comparison in terms of generalization ability between the state-of-the-art DEXTR and our IOG.

2. cross domain——Object categories

Methods	Train	Finetune	Backbone	#Clicks	IoU
Curve-GCN [42]	Cityscapes	N.A.	ResNet-50	2	76.3
Curve-GCN [42]	Cityscapes	N.A.	ResNet-50	2.4	77.6
Curve-GCN [42]	Cityscapes	N.A.	ResNet-50	3.6	80.2
DEXTR [42]	Cityscapes	N.A.	ResNet-101	4	79.4
IOG (ours)	PASCAL	✗	ResNet-50	3	77.9
IOG (ours)	PASCAL	✓	ResNet-50	3	82.2
IOG (ours)	PASCAL	✓	ResNet-101	3	82.7
IOG (ours)	COCO	✓	ResNet-101	3	**83.8**

Table 4. **Cross domain analysis on Cityscapes [17].** "Fine-tune" indicates that the method is fine-tuned on a small set of the Cityscapes dataset (10%).

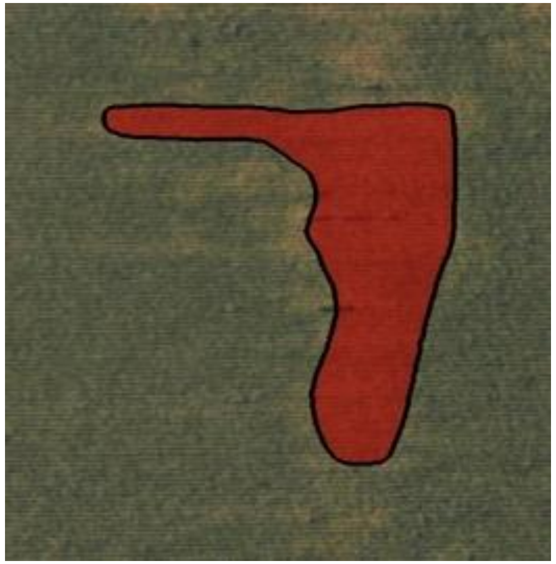
Methods	Train	Finetune	Backbone	#Clicks	IoU
Curve-GCN [42]	CityScapes	✗	ResNet-50	2	68.3
Curve-GCN [42]	CityScapes	✓	ResNet-50	2	78.2
IOG (ours)	PASCAL	✗	ResNet-50	3	90.7
IOG (ours)	PASCAL	✓	ResNet-50	3	92.8
IOG (ours)	PASCAL	✓	ResNet-101	3	93.6
IOG (ours)	COCO	✓	ResNet-101	3	**94**

Table 5. **Cross domain analysis on Rooftop [57].** Even without fine-tuning, our method already outperforms Curve-GCN with fine-tuning, showing the strong generalization of our approach.

Methods	Train	Finetune	Backbone	#Clicks	IoU
Curve-GCN [42]	CityScapes	✗	ResNet-50	2	60.9
IOG (ours)	PASCAL	✗	ResNet-50	3	81.4
IOG (ours)	PASCAL	✗	ResNet-101	3	**83.7**

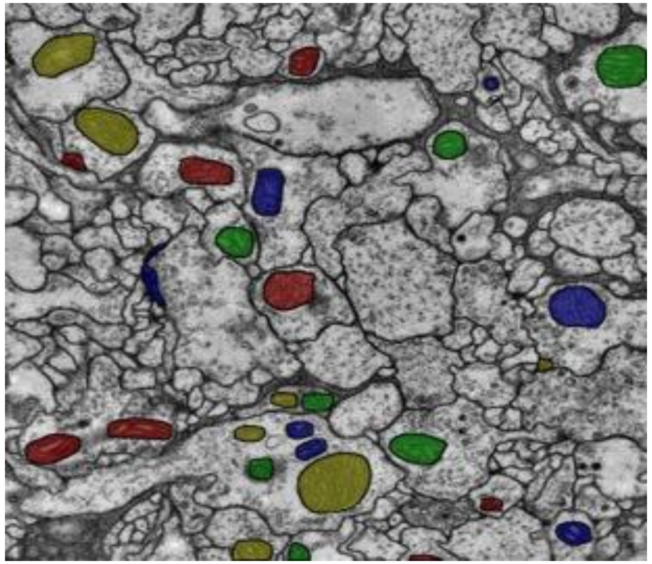
Table 6. **Cross domain analysis on ssTEM [22].** Note that ssTEM does not have a training split, therefore we do not perform fine-tuning on this dataset.

Cityscapes



Agriculture-Vision

Rooftop

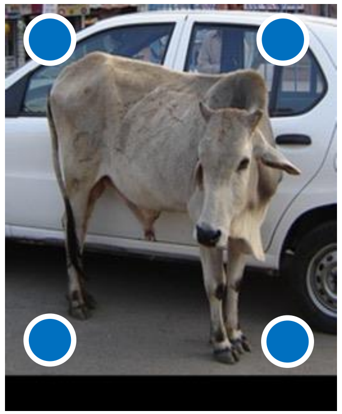


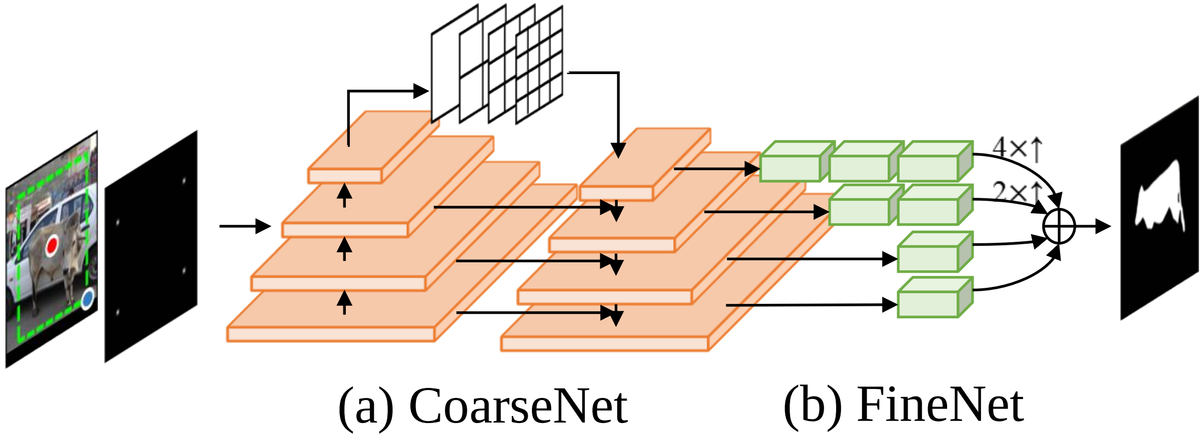
ssTEM

2. cross domain——Stuff categories

Extension (Automated Mode)

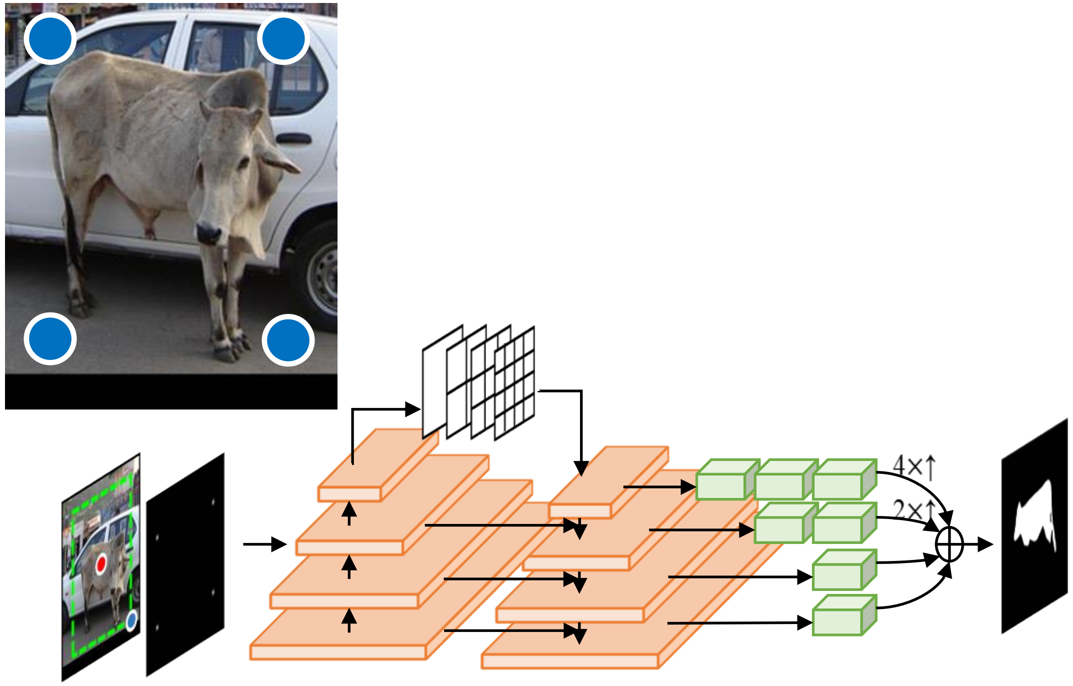
- *Without* user interaction, our IOG can still harvest high quality masks from off-the-shelf datasets with *box annotations* (e.g. ImageNet)



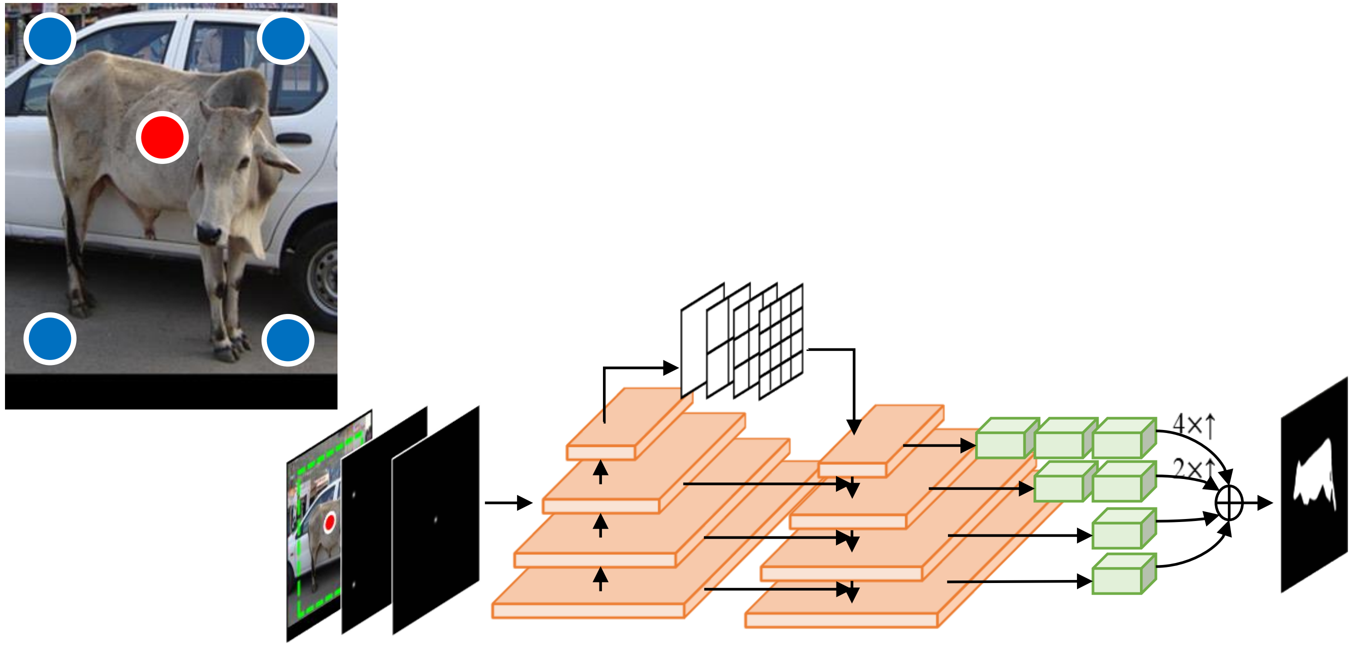


Extension (Automated Mode)

- Solution: Two-stage Training:
  - (S1) Train a network that takes box as inputs to produce segmentation
  - (S2) Infer interior clicks from the masks produced in S1 and apply IOG



(a) CoarseNet (b) FineNet



(a) CoarseNet (b) FineNet

Method	Backbone	Train	IoU
(A) Crop	ResNet-50	PASCAL-1k	87.5
(B) Geo	ResNet-50	PASCAL-1k	89.5
(C) Sim	ResNet-50	PASCAL-1k	86.1
(D) Outside only	ResNet-50	PASCAL-1k	89.5
(D) Outside only	ResNet-101	PASCAL-10k	90.9
(E) 2-stage	ResNet-101	PASCAL-10k	91.1

Table 8. **Extension to dataset with box annotations only.** All the results are reported on PASCAL *val* using box annotations only.

PIXEL-IMAGENET

https://github.com/shiyinzhang/Pixel-ImageNet

Characteristics

- #Classes: 1000
- #Instance: >600K

Possible Applications

- Image classification
- Instance segmentation
- Semantic segmentation
- Salient object detection
- …. and more

Conclusions

- Propose IOG:
  - Requires only three points (an inside point and two outside points)
  - Supports additional points for further correction
  - Performs well across different datasets and domains
- Contribute Pixel-ImageNet:
  - A large volumes of high-quality pixel-level dataset
  - Offer unparalleled opportunities to researchers in the computer vision community