Image segmentation

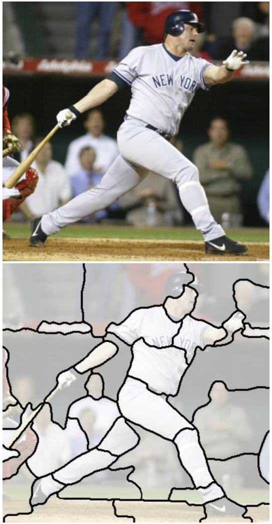
Vlad Shakhuro

13 November 2025

Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

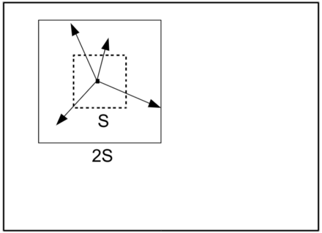
Superpixels (visual segmentation or oversegmentation)



Regions of image. Desired properties:

- homogeneous
- compact
- uniformly distributed over the image
- large enough to be informative
- have boundaries aligned with object boundaries
- superpixel is fully contained in one object mask
- small object are described with whole superpixels
- easily computable

SLIC (Simple Linear Iterative Clustering)



k-means with
bounded comparisons

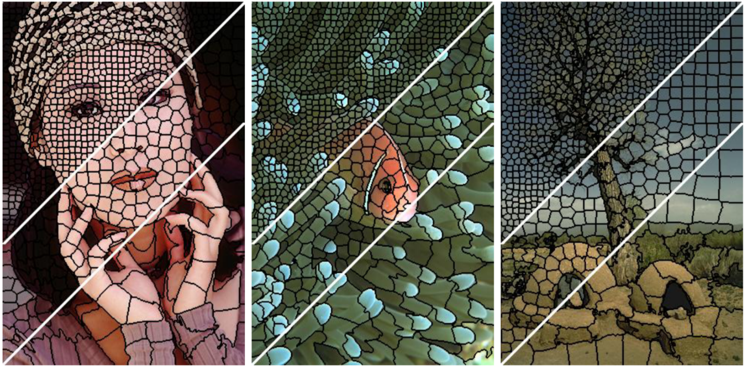
Initialize clusters at regular grid S

Run bounded k-means:

1. Compute distances between clusters and pixels in 2S×2S area. Use CIELAB and (x, y) coordinates as feature vectors
2. Recompute clusters and amount of change (L_1 distance between old and new clusters)

Achanta et al. SLIC Superpixels. EPFL Tech Report 2010
Achanta et al. SLIC superpixels compared to state-of-the-art superpixel methods. TPAMI 2012

SLIC results



SLIC comparison

Efficient Graph-Based

TurboPixel

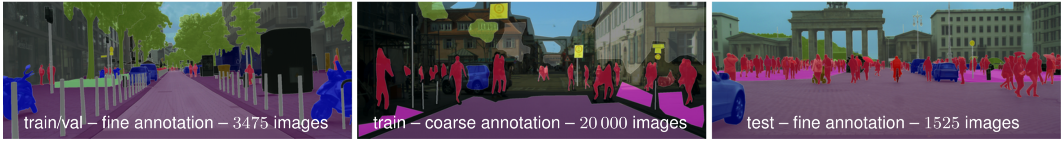
QuickShift

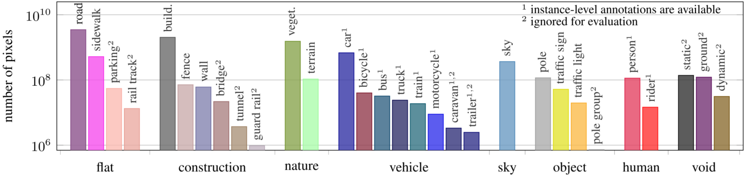
SLIC

Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

Cityscapes

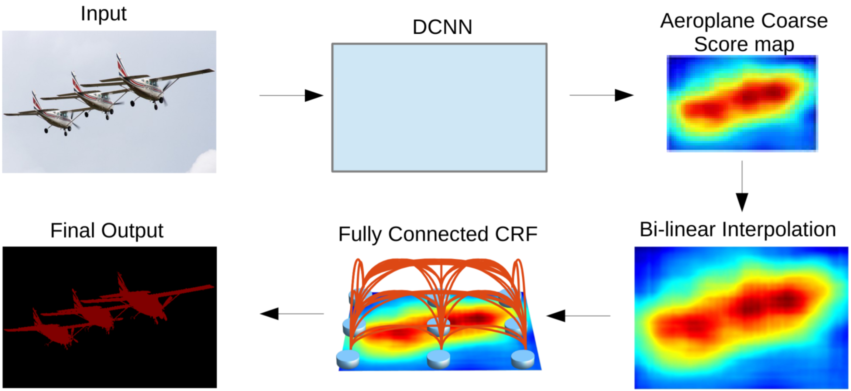




- images from a car from several german cities
- 30 object classes
- 5k finely labelled images
- 20k coarsely labelled images

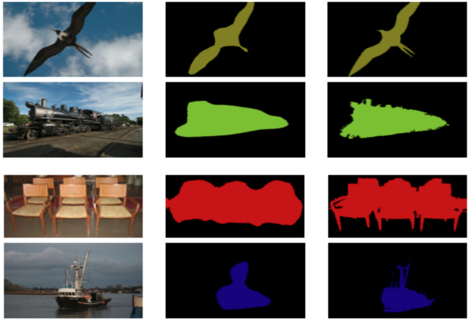
Cordts et al. The Cityscapes Dataset for Semantic Urban Scene Understanding. CVPR 2016

DeepLab



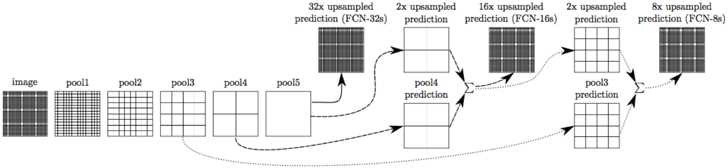
Chen et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. TPAMI 2016

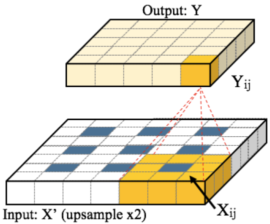
DeepLab

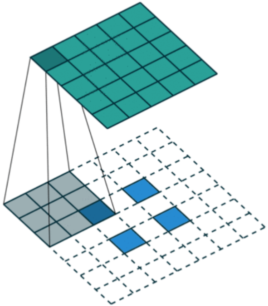


Chen et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. TPAMI 2016

Fully Convolutional Networks

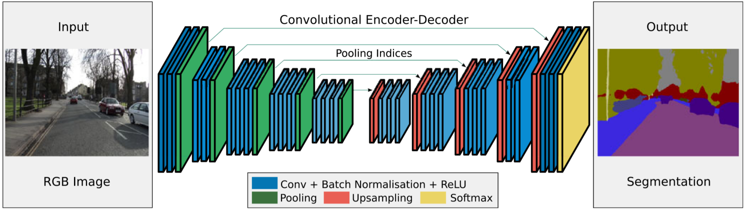


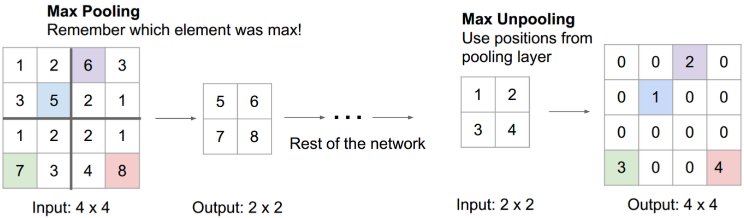




Long et al. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

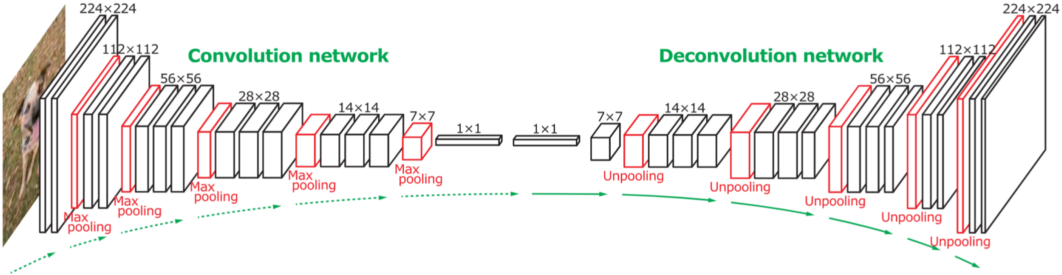
Segnet with unpooling





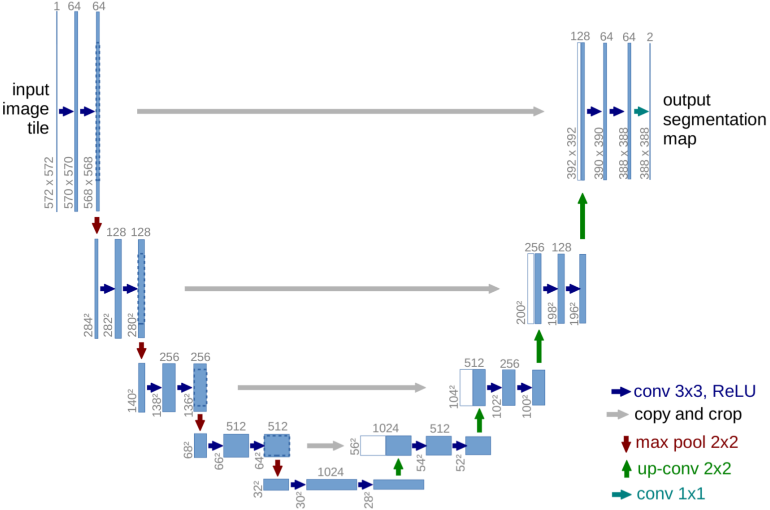
Badrinarayanan et al. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. TPAMI 2017

DeconvNet



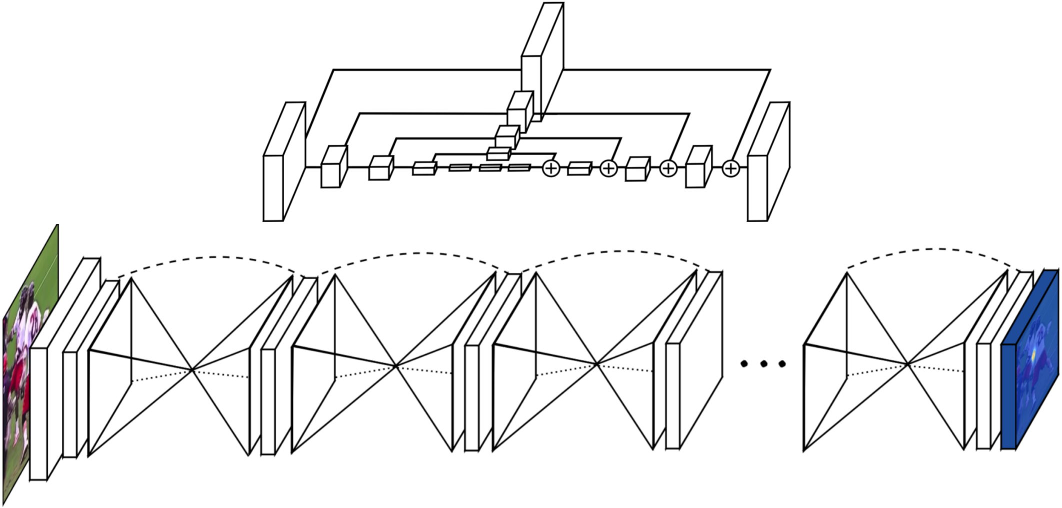
Noh et al. Learning Deconvolution Network for Semantic Segmentation. ICCV 2015

U-Net



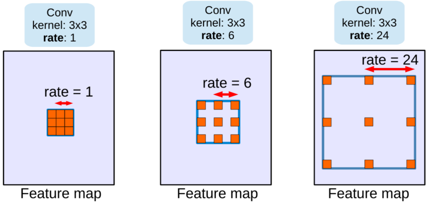
Ronneberger et al. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

Hourglass networks



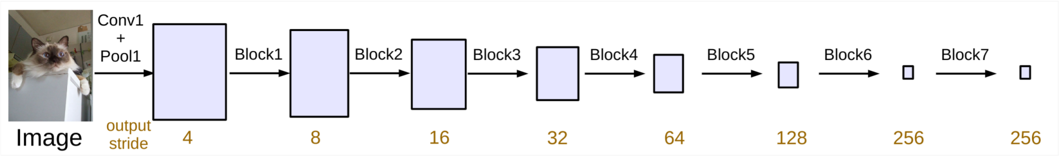
Newell et al. Stacked Hourglass Networks for Human Pose Estimation. ECCV 2016

Atrous convolutions

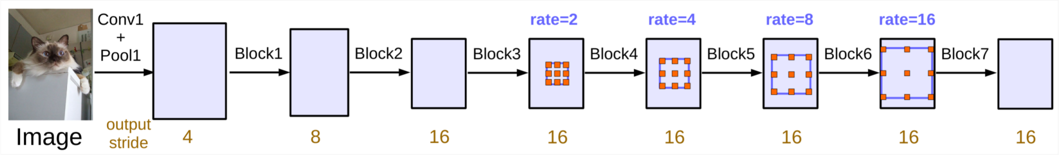


Chen et al. Rethinking Atrous Convolution for Semantic Image Segmentation. 2017

Atrous convolutions



(a) Going deeper without atrous convolution.

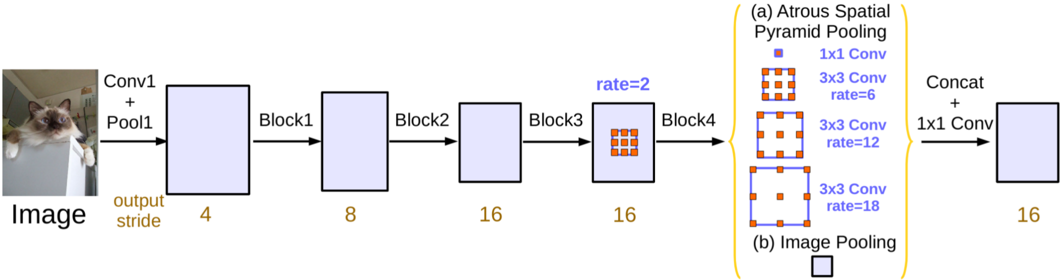


(b) Going deeper with atrous convolution. Atrous convolution with $rate > 1$ is applied after block3 when $output\_stride = 16$.

Figure 3. Cascaded modules without and with atrous convolution.

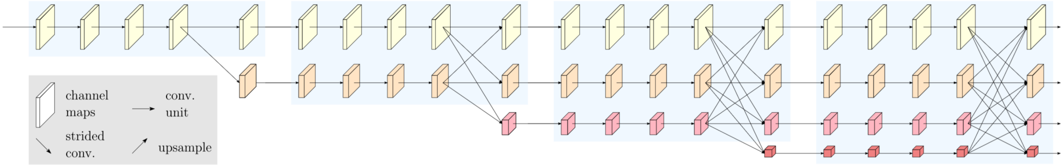
Chen et al. Rethinking Atrous Convolution for Semantic Image Segmentation. 2017

Atrous convolutions



Chen et al. Rethinking Atrous Convolution for Semantic Image Segmentation. 2017

HRNet



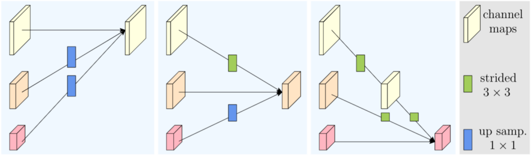
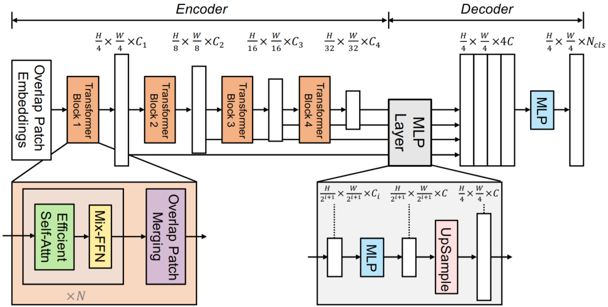


Fig. 3. Illustrating how the fusion module aggregates the information for high, medium and low resolutions from left to right, respectively. Right legend: strided 3×3 = stride-2 3×3 convolution, up samp. 1×1 = bilinear upsampling followed by a 1×1 convolution.

Wang et al. Deep High-Resolution Representation Learning for Visual Recognition. TPAMI 2020

SegFormer



Efficient SA:

$$SA = softmax(qk^T/\sqrt{D_h})v$$

$$k = Reshape(\frac{N}{R}, C \cdot R)(k)$$
$$k = Linear(C \cdot R, C)(k)$$

Mix-FFN:

$$MLP(Conv_{3\times 3}(MLP(x))) + x$$

Xie et al. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. NeurIPS 2021

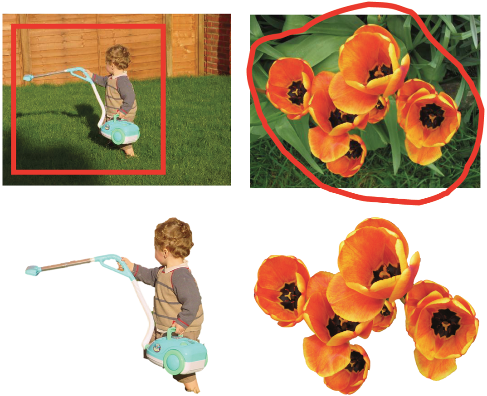
SegFormer

	Output Size	Layer Name	Mix Transformer					
			B0	B1	B2	B3	B4	B5
Stage 1	$\frac{H}{4} \times \frac{W}{4}$	Overlapping Patch Embedding	$K_1 = 7;\ S_1 = 4;\ P_1 = 3$					
			$C_1 = 32$	$C_1 = 64$				
		Transformer Encoder	$R_1 = 8$ $N_1 = 1$ $E_1 = 8$ $L_1 = 2$	$R_1 = 8$ $N_1 = 1$ $E_1 = 8$ $L_1 = 2$	$R_1 = 8$ $N_1 = 1$ $E_1 = 8$ $L_1 = 3$	$R_1 = 8$ $N_1 = 1$ $E_1 = 8$ $L_1 = 3$	$R_1 = 8$ $N_1 = 1$ $E_1 = 8$ $L_1 = 3$	$R_1 = 8$ $N_1 = 1$ $E_1 = 4$ $L_1 = 3$
Stage 2	$\frac{H}{8} \times \frac{W}{8}$	Overlapping Patch Embedding	$K_2 = 3;\ S_2 = 2;\ P_2 = 1$					
			$C_2 = 64$	$C_2 = 128$				
		Transformer Encoder	$R_2 = 4$ $N_2 = 2$ $E_2 = 8$ $L_2 = 2$	$R_2 = 4$ $N_2 = 2$ $E_2 = 8$ $L_2 = 2$	$R_2 = 4$ $N_2 = 2$ $E_2 = 8$ $L_2 = 3$	$R_2 = 4$ $N_2 = 2$ $E_2 = 8$ $L_2 = 3$	$R_2 = 4$ $N_2 = 2$ $E_2 = 8$ $L_2 = 8$	$R_2 = 4$ $N_2 = 2$ $E_2 = 4$ $L_2 = 6$
Stage 3	$\frac{H}{16} \times \frac{W}{16}$	Overlapping Patch Embedding	$K_3 = 3;\ S_3 = 2;\ P_3 = 1$					
			$C_3 = 160$	$C_3 = 320$				
		Transformer Encoder	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 2$	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 2$	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 6$	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 18$	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 27$	$R_3 = 2$ $N_3 = 5$ $E_3 = 4$ $L_3 = 40$
Stage 4	$\frac{H}{32} \times \frac{W}{32}$	Overlapping Patch Embedding	$K_4 = 3;\ S_4 = 2;\ P_4 = 1$					
			$C_4 = 256$	$C_4 = 512$				
		Transformer Encoder	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 2$	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 2$	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 3$	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 3$	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 3$	$R_4 = 1$ $N_4 = 8$ $E_4 = 4$ $L_4 = 3$

Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

Interactive segmentation

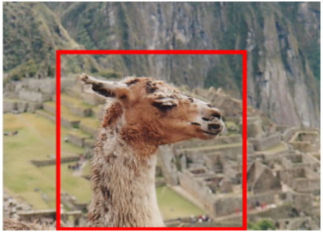


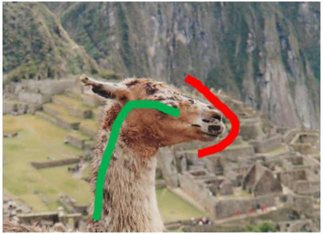
Applications:

- stickers
- inpainting
- fast labelling

UI types


clicks

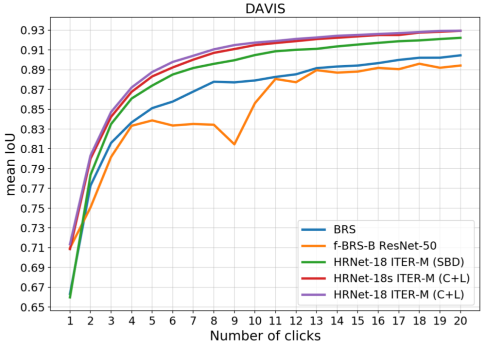

bbox


strokes


contour

Datasets and metrics

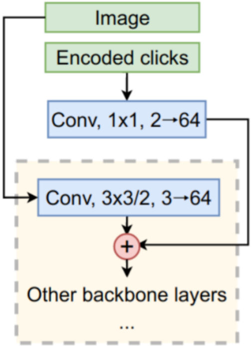
Berkeley — 50 images
GrabCut — 100 images
DAVIS — 345 images
SBD — 2857 images, 6671 masks



NoC@0.9 — average number of clicks to reach IoU 0.9

#images ⩾ 20 — number of images with IoU < 0.9 withing 20 clicks

RITM



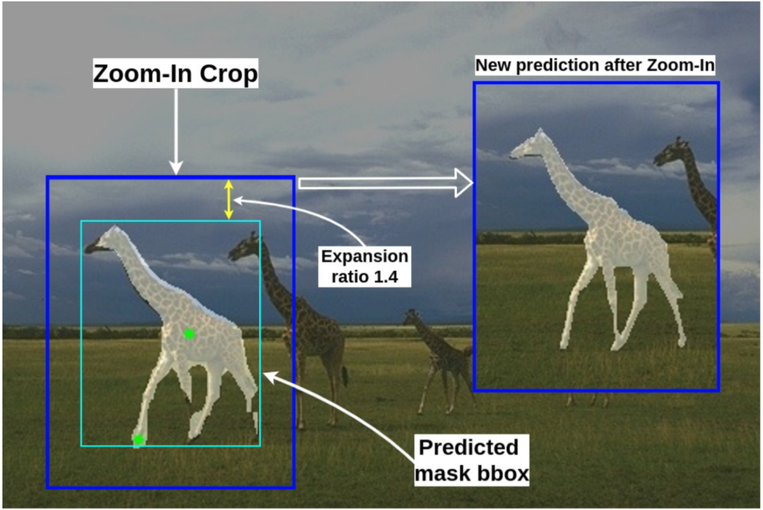
Conv1S

Key ideas:

- click encoding
- iterative training
- using mask from previous step
- usage of modern dataset (COCO+LVIS) for training

Sofiiuk et al. Reviving Iterative Training with Mask Guidance for Interactive Segmentation. ICIP 2022

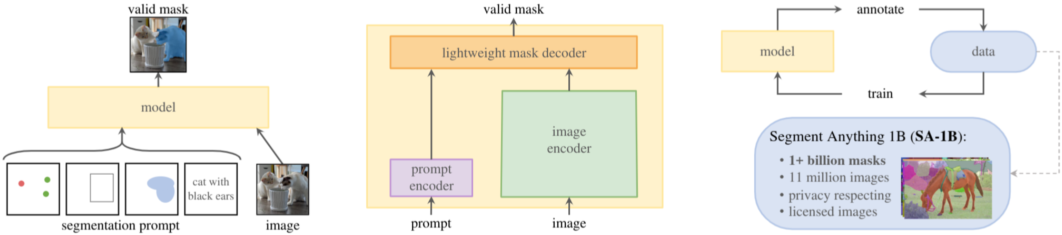
Zoom-In



Sofiiuk et al. f-BRS: Rethinking Backpropagating Refinement for Interactive Segmentation. CVPR 2020

RITM examples

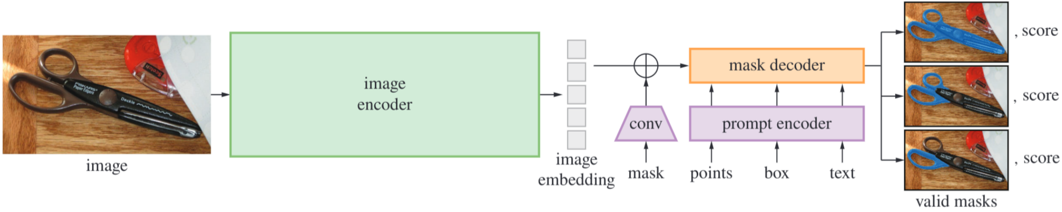
SegmentAnything



(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**) (c) **Data**: data engine (top) & dataset (bottom)

Kirillov et al. Segment Anything. ICCV 2023

SegmentAnything

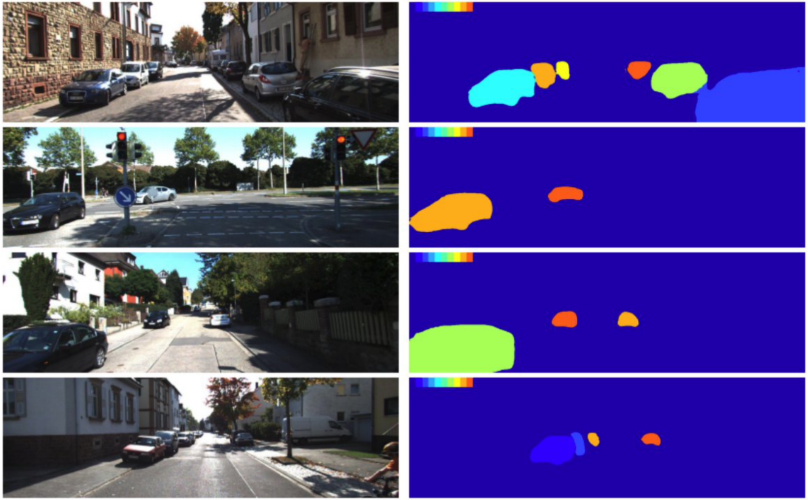


Kirillov et al. Segment Anything. ICCV 2023

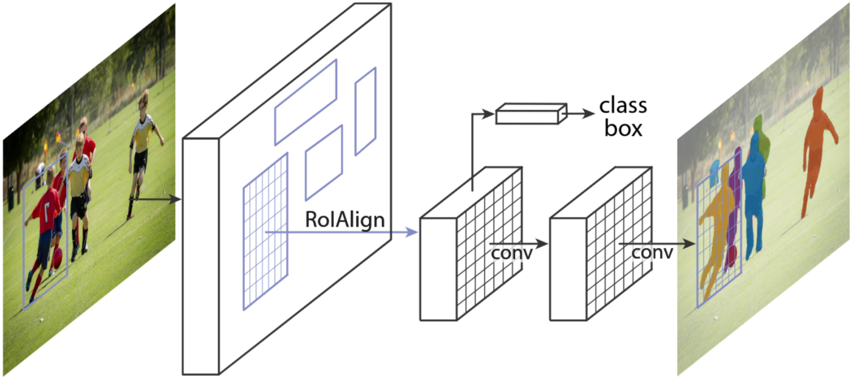
Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

Instance segmentation



Mask R-CNN



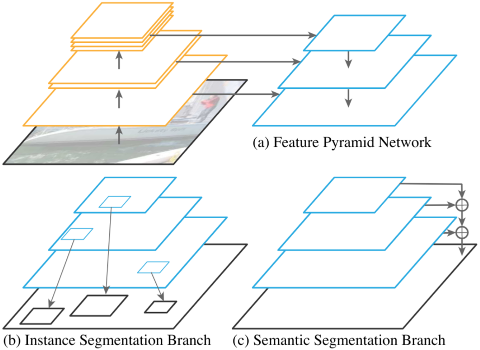
He et al. Mask R-CNN. ICCV 2017

Mask R-CNN results

Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

Panoptic Feature Pyramid Networks



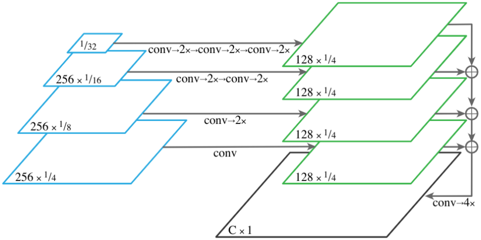
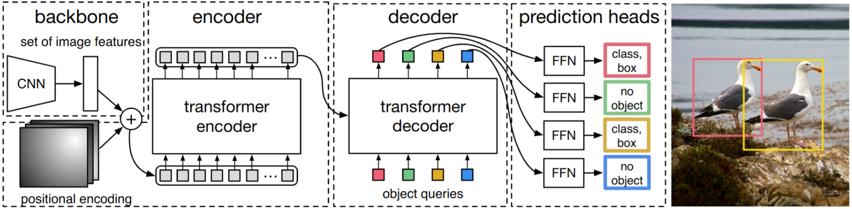
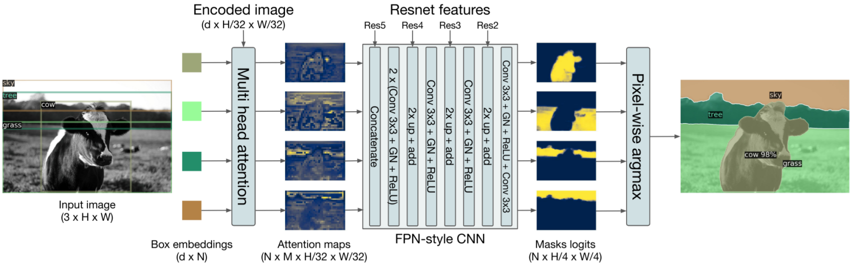


Figure 3: **Semantic segmentation branch.** Each FPN level (left) is upsampled by convolutions and bilinear upsampling until it reaches 1/4 scale (right), theses outputs are then summed and finally transformed into a pixel-wise output.

Kirillov et al. Panoptic Feature Pyramid Networks. CVPR 2019

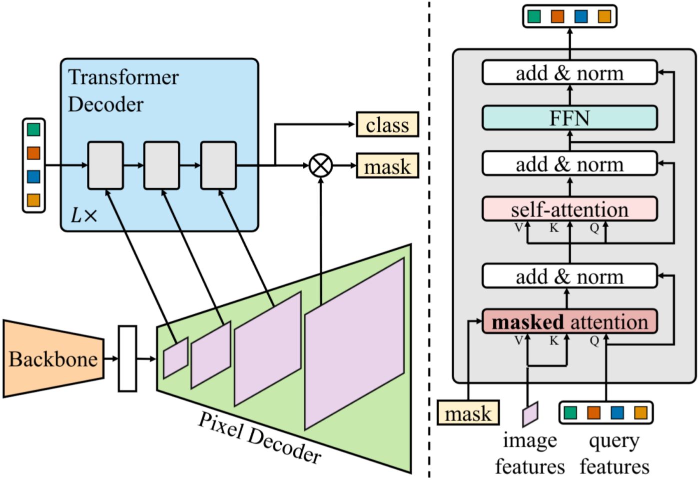
DETR





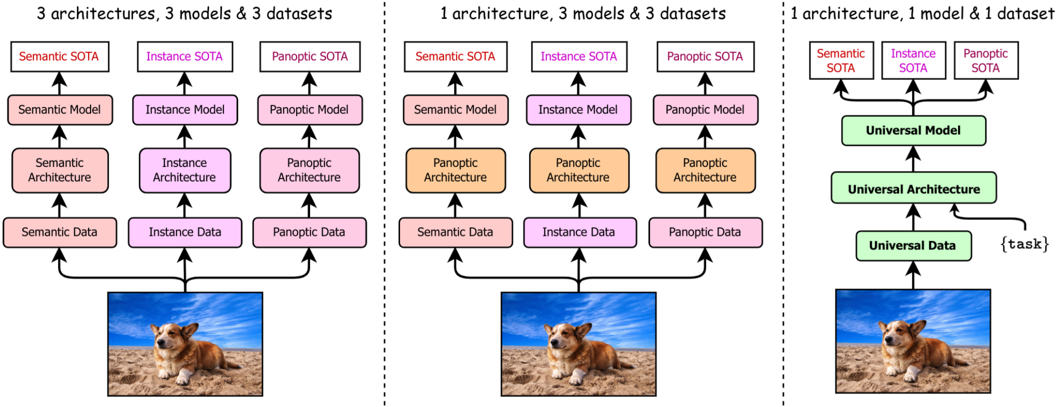
Carion et al. End-to-End Object Detection with Transformers. ECCV 2020

Mask2Former



Cheng et al. Masked-attention Mask Transformer for Universal Image Segmentation. CVPR 2022

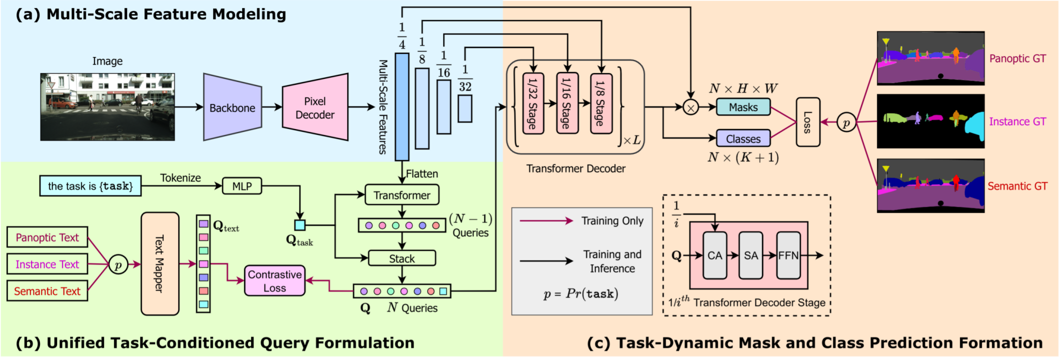
OneFormer



(a) Specialized Architectures, Models & Datasets
(b) Panoptic Architecture BUT Specialized Models & Datasets
(c) Universal Architecture, Model and Dataset

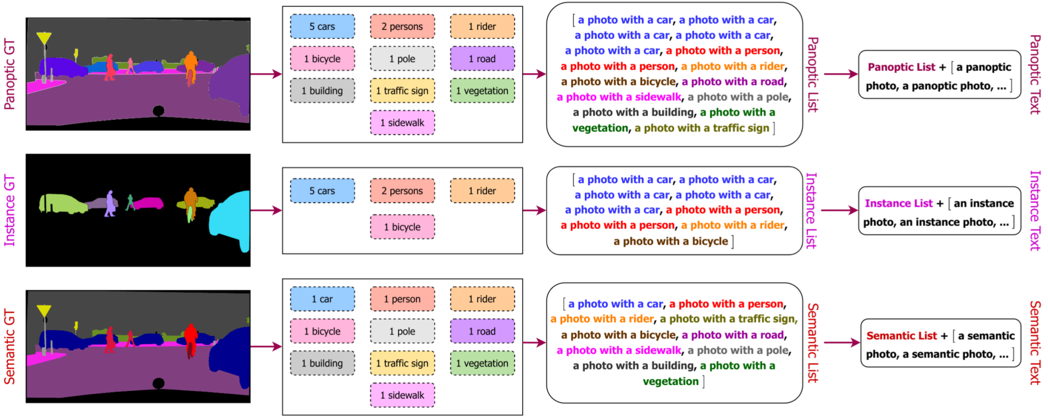
Jain et al. OneFormer: One Transformer to Rule Universal Image Segmentation. CVPR 2023

OneFormer



Jain et al. OneFormer: One Transformer to Rule Universal Image Segmentation. CVPR 2023

OneFormer



(a) Task based GT Label

(b) Extract number of binary masks for each class

(c) Form a list with text for each mask's class type

(d) Pad extracted Text to obtain a list of length N_{text}

Jain et al. OneFormer: One Transformer to Rule Universal Image Segmentation. CVPR 2023

OneFormer

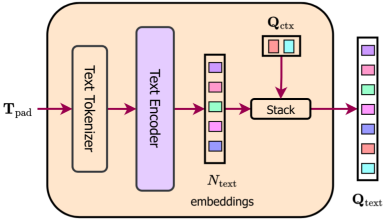


Figure 4. **Text Mapper.** We tokenize and then encode the input text list ($\mathbf{T}_{\text{pad}}$) using a 6-layer transformer text encoder [49, 57] to obtain a set of N_{text} embeddings. We concatenate a set of N_{ctx} learnable embeddings to the encoded representations to obtain the final N text queries ($\mathbf{Q}_{\text{text}}$). The N text queries stand for a text-based representation of the objects present in an image.

$$\mathcal{L}_{\mathbf{Q}\rightarrow\mathbf{Q}_{\text{text}}} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp(q_i^{obj}\odot q_i^{txt}/\tau)}{\sum_{j=1}^{B}\exp(q_i^{obj}\odot q_j^{txt}/\tau)},$$

$$\mathcal{L}_{\mathbf{Q}_{\text{text}}\rightarrow\mathbf{Q}} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\exp(q_i^{txt}\odot q_i^{obj}/\tau)}{\sum_{j=1}^{B}\exp(q_i^{txt}\odot q_j^{obj}/\tau)}$$

$$\mathcal{L}_{\mathbf{Q}\leftrightarrow\mathbf{Q}_{\text{text}}} = \mathcal{L}_{\mathbf{Q}\rightarrow\mathbf{Q}_{\text{text}}} + \mathcal{L}_{\mathbf{Q}_{\text{text}}\rightarrow\mathbf{Q}}$$

Jain et al. OneFormer: One Transformer to Rule Universal Image Segmentation. CVPR 2023

OneFormer

Method	Backbone	#Params	#FLOPs	#Queries	Crop Size	Iters	PQ	AP	mIoU (s.s.)	mIoU (m.s.)
Individual Training										
CMT-DeepLab‡ [59]	MaX-S† [50]	—	—	—	1025×2049	60k	64.6	—	81.4	—
Axial-DeepLab-L‡ [51]	Axial ResNet-L† [51]	45M	687G	—	1025×2049	60k	63.9	35.8	81.0	81.5
Axial-DeepLab-XL‡ [51]	Axial ResNet-XL† [51]	173M	2447G	—	1025×2049	60k	64.4	36.7	80.6	81.1
Panoptic-DeepLab‡ [11]	SWideRNet† [8]	536M	10365G	—	1025×2049	60k	66.4	40.1	82.2	82.9
Mask2Former-Panoptic [12]	Swin-L† [38]	216M	514G	200	512×1024	90k	66.6	43.6	82.9	—
Mask2Former-Instance [12]	Swin-L† [38]	216M	507G	200	512×1024	90k	—	43.7	—	—
Mask2Former-Semantic [12]	Swin-L† [38]	215M	494G	100	512×1024	90k	—	—	83.3	84.3
kMaX-DeepLab‡ [60]	ConvNeXt-L† [39]	232M	1673G	256	1025×2049	60k	68.4	44.0	83.5	—
Joint Training										
OneFormer	Swin-L† [38]	219M	543G	250	512×1024	90k	**67.2**	**45.6**	83.0	**84.4**
OneFormer	ConvNeXt-L† [39]	220M	497G	250	512×1024	90k	**68.5**	**46.5**	83.0	84.0
OneFormer	ConvNeXt-XL† [39]	372M	775G	250	512×1024	90k	**68.4**	**46.7**	**83.6**	**84.6**
OneFormer	DiNAT-L† [21]	223M	450G	250	512×1024	90k	**67.6**	**45.6**	83.1	84.0

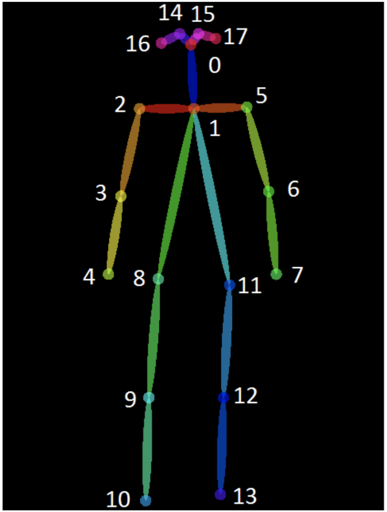
Table 2. **SOTA Comparison on Cityscapes val set.** †: backbones pretrained on ImageNet-22K; ‡: trained with batch size 32, *: hidden dimension 1024. OneFormer outperforms the individually trained Mask2Former [12] models. Mask2Former's performance with 250 queries is not listed, as its performance degrades with 250 queries. We compute FLOPs using the corresponding crop size.

Jain et al. OneFormer: One Transformer to Rule Universal Image Segmentation. CVPR 2023

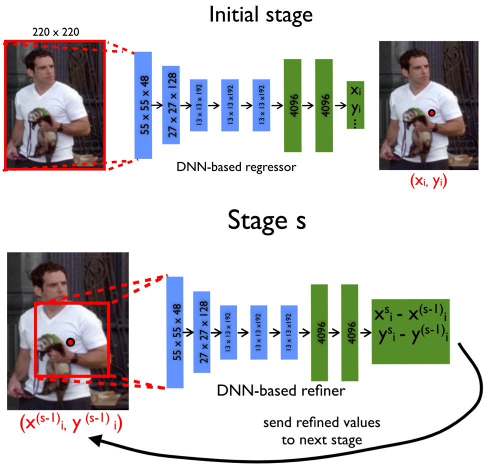
Outline

1. Superpixels
2. Semantic
3. Interactive
4. Instance
5. Panoptic
6. Human pose estimation

Human pose estimation

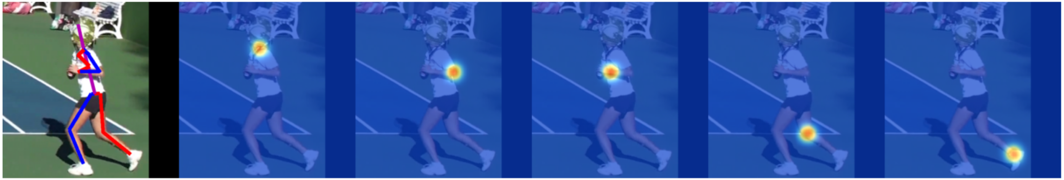


Regressing joint positions



Toshev, Szegedy. DeepPose: Human Pose Estimation via Deep Neural Networks. CVPR 2014

Predicting joint using heatmaps

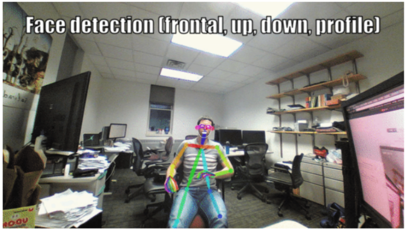


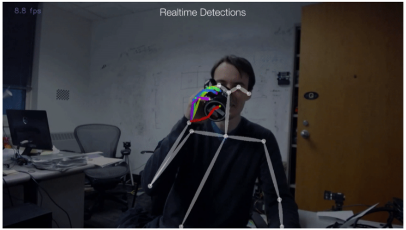
OpenPose

(a) Input Image (b) Part Confidence Maps (c) Part Affinity Fields (d) Bipartite Matching (e) Parsing Results

Cao et al. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. CVPR 2017

OpenPose





Cao et al. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. CVPR 2017

Conclusion

We reviewed following topics:

- superpixel computation with SLIC algorithm
- various methods for semantic segmentation
- several modern methods for click-based interactive segmentation
- instance segmentation using Mask R-CNN
- several modern methods for panoptic segmentation
- human pose estimation via segmentation