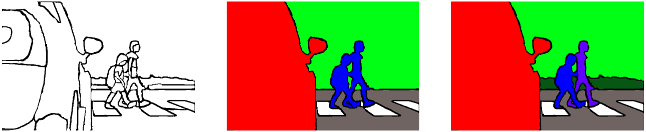
UE CSC_5RO13
Computer Vision with Deep Learning
Semantic Segmentation

Antoine Manzanera - ENSTA



Segmentation: What? Why?

- Segmentation: *Partition* an image into consistent *segments* / *regions* in terms of:
  - colour
  - texture
  - objects
  - foreground *vs* background
  - things and stuff
- Fundamental Computer Vision task for:
  - Object detection, Pose estimation, Action recognition,...
  - Obstacle avoidance, Navigable surface detection,...
  - Virtual background, Augmented Reality,...
  - Remote sensing, Medical images,...

Lecture outline

Outline

Segmentation did not start with SegNet!

Segmentation has been a cardinal task of Computer Vision since the very beginning! Thousands of papers were published on the subject before 2010, with a huge variety of approaches. Those methods did not pretend to address semantic segmentation, but aimed to reduce the content of the image to a partition in significant regions, by grouping pixels according to two criteria:

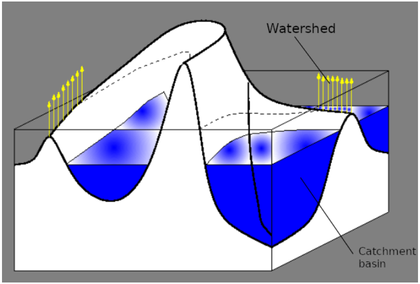
- *Appearance* consistency: Pixels in a same region should have close colours or textures.
- *Geometric* consistency: The region should be regular and not too large.

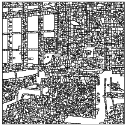
Morphological Watersheds

The morphological watershed is a well founded segmentation algorithm, based on a topographic model of the image gradient, filled by water immersion.
Shape and size of regions (catchment basins) can be controlled by:

- Morphological filtering
- Marking of relevant regions.

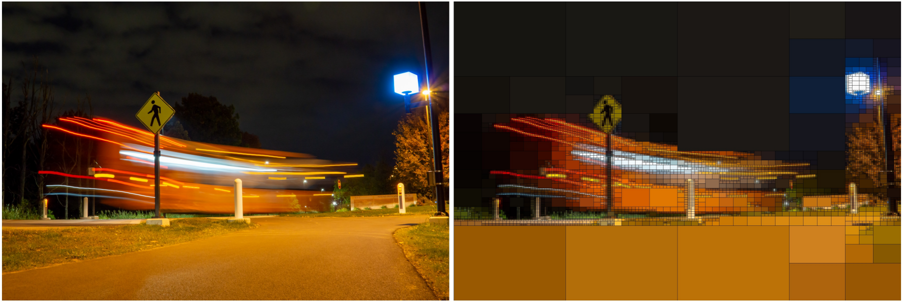
[Vincent91]





Divide-and-Conquer methods

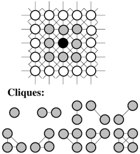
Divide-and-Conquer methods first split the image into atomic regions (e.g. pixels), then recursively merge the regions (e.g. following a dyadic pyramid process) based on similarity criteria.



[image from jrtechs.net]

Markovian image segmentation

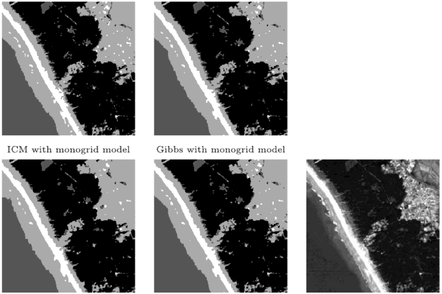
Markov Random Fields are a well founded framework for image segmentation, based on miminising the energy of a Gibbs field defined over the cliques (fully connected subgraphs) of the regular graph formed by the image:



$$P(X=\omega) = \frac{1}{Z} \exp\left(-\sum_{c\in\mathcal{C}} V_c(\omega)\right)$$

$$\max_{\omega} P(X=\omega) = \min_{\omega} \sum_{c\in\mathcal{C}} V_c(\omega)$$

[Kato12]



ICM with monogrid model — Gibbs with monogrid model

ICM with multiscale model — Gibbs with multiscale model — Original image

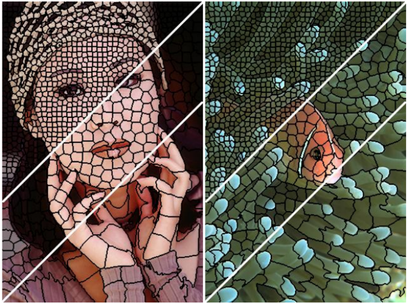
Combining with clustering

Image Segmentation can be combined with clustering algorithms calculated in the tonal (gray level, colour) space, or in some transformed (latent) space:

- K-means clustering
- Histogram segmentation
- Meanshift mode tracking
- Bayesian classification
- .../...

Superpixels and RAGs

Superpixel algorithms are popular non-semantic segmentation methods that allow to reduce in a flexible way the volume of data while keeping a - relatively - regular graph topology. Superpixel graphs can then be used as inputs of convolutional or transformer based neural networks.

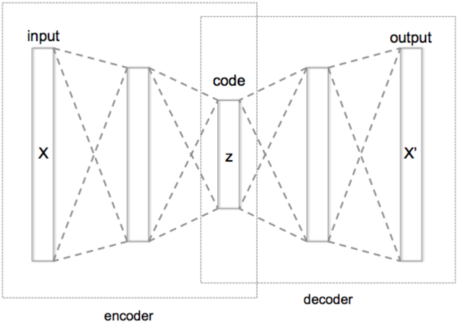


[Achanta12]

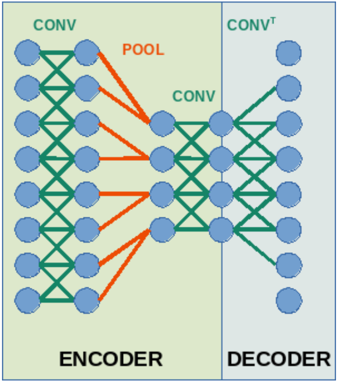
Outline

It starts from Autoencoders...

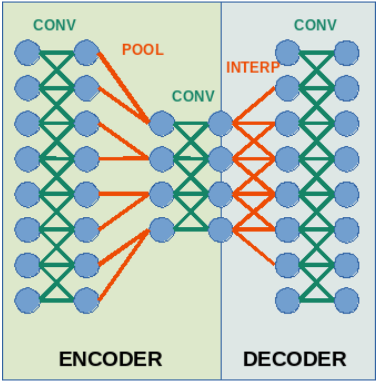
Since a segmentation algorithm is expected to provide a label (class) for each pixel of the input image, the architecture of a neural network trained for image segmentation follows the structure of an autoencoder:



How does the decoder increase the resolution? (1/2)

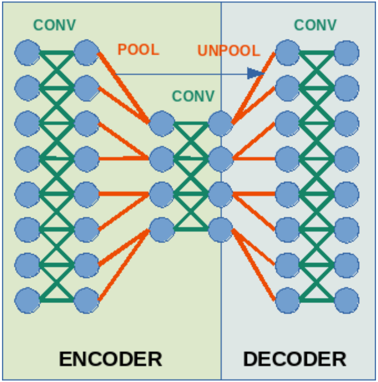


Transposed Convolution

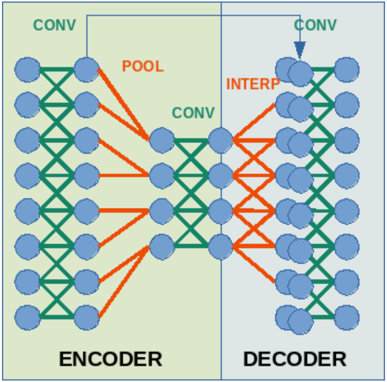


Interpolation

How does the decoder increase the resolution? (2/2)



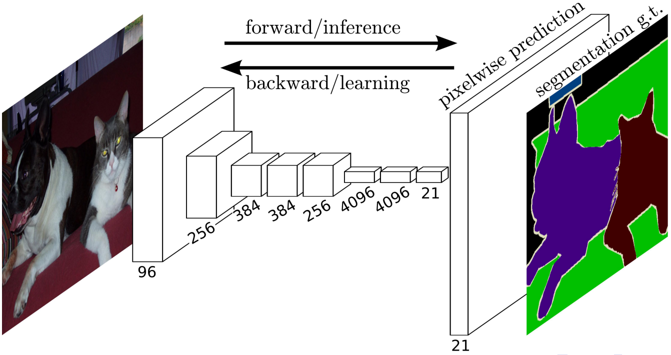
Unpooling



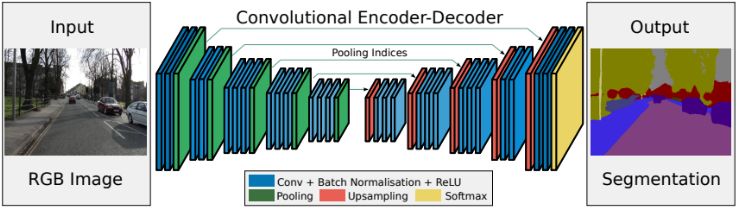
Skip connections

FCNSeg [Shelhamer15]

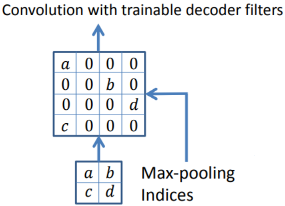
FCNSeg is a Fully Convolutional Network (FCN) that ends with a large transposed convolution layer which produces a coarse segmentation map, which is then upsampled using different methods:



SegNet [Badrinarayanan15]

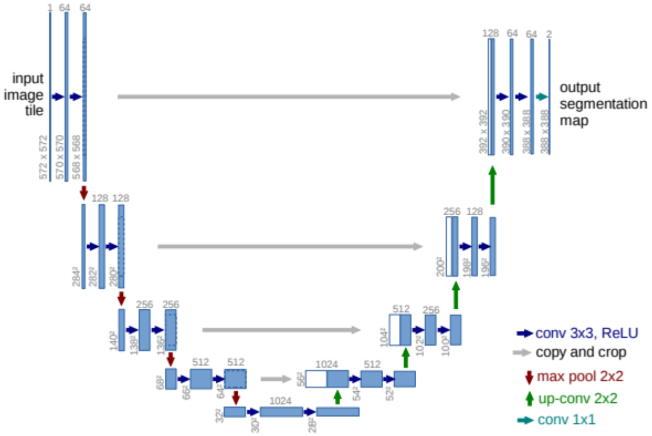


SegNet is a symmetrical FCN based on an encoder-decoder structure without skip connections but with particular max-unpooling layers:

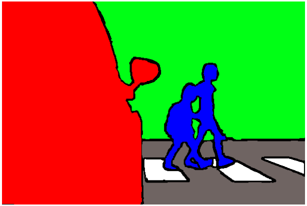


U-Net [Ronneberger15]

U-Net is another FCN that promotes a higher resolution of the features (and then segmentation) maps by using skip connections:



Output encoding and loss functions?



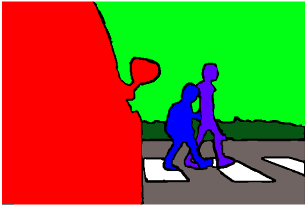
In the case of semantic segmentation, the last layer is a *softmax* function that encodes, for each pixel $p \in \mathcal{P}$, a probability distribution among classes $i \in \mathcal{C}$: $\hat{y}(p)_i = \frac{e^{\lambda(p)_i}}{\sum_{j \in \mathcal{C}} e^{\lambda(p)_j}}$

Akin to classification, the typical loss function for segmentation is the sum over pixels of the cross entropy:

$$\mathcal{L}_{\text{seg}}(\hat{y}, y) = -\sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{C}} \omega_i y(p)_i \log(\hat{y}(p)_i)$$

(the weights ω_i can be adjusted to account for disbalanced classes in the training set).

Output encoding and loss functions?



In the case of instance segmentation, in addition to softmax, an instance label $k \in \mathcal{K}$ has to be predicted by the network for each pixel.

The instance-level loss function is then typically summed over the different predicted instances:

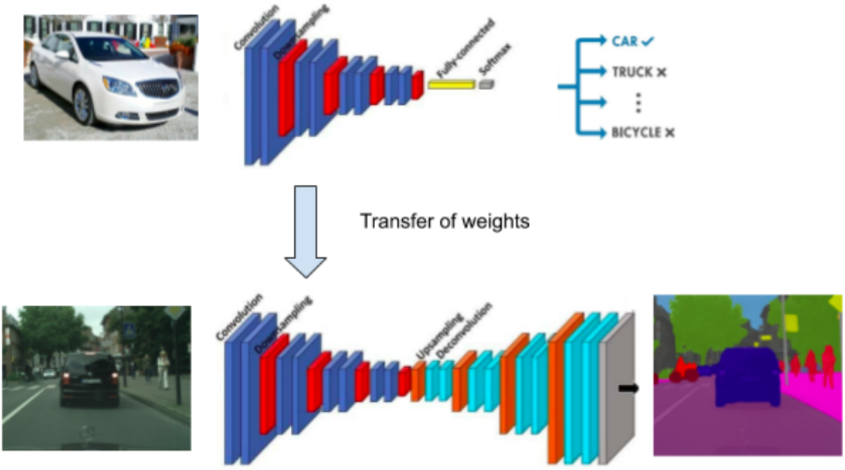
$$\mathcal{L}_{\text{inst}}(\hat{y}, y) = -\sum_{p \in \mathcal{P}} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{C}} \omega_i y(p)_i^k \log(\hat{y}(p)_i^k)$$

(the weights ω_i can be adjusted to account for disbalanced classes in the training set).

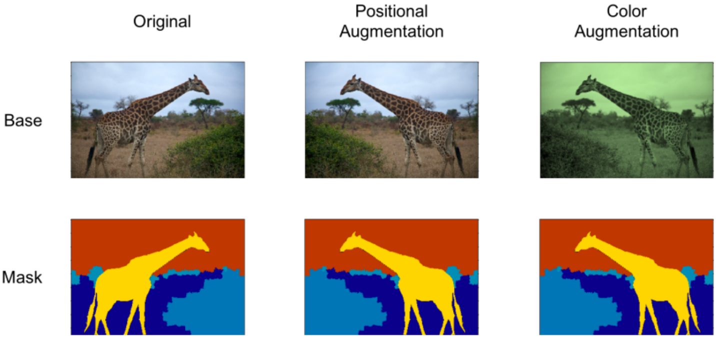
Training Segmentation Networks

- Segmentation CNN are Fully Convolutional, then applicable on any size images, but take care of the receptor fields, that determine the scale and then the semantic level of the representation.
- Like auto-encoders and their variations (denoising or restoring networks), the loss functions are relatively straightforward.
- But unlike auto-encoders and their variations, self-supervision is hard to design, and ground truth annotations hard to obtain.
- For supervised approaches, ground truth annotations are typically obtained by:
  - Manual annotations using e.g. CVAT or LabelMe (Pascal VOC, Cityscapes,...)
  - Image synthesis tools (SynthIA, GTA5,...)

Using pre-trained encoders [from medium.com/@VK]



Using data augmentation [from mxnet.apache.org]



Outline

Back to the principle: From NL-Means to Self-Attention

- Self-attention layers (Transformers) overcome the limitation of local computation (temporal causality / spatial dependence), by allowing the interaction - in one single layer - of very distant elements in the input data.
- In the same way as Neural Networks adopted convolution as a fundamental primitive in CNN to generalise local operations through learned kernels, self-attention layers generalise Non-Local operations by learning both similarity functions (which pixels will most interact), and the associated weights.

From NL-Means to Self-Attention

NL-Means :

$$y_i = \frac{1}{\pi(i)} \sum_j w(i,j)\, x_j$$

inputs

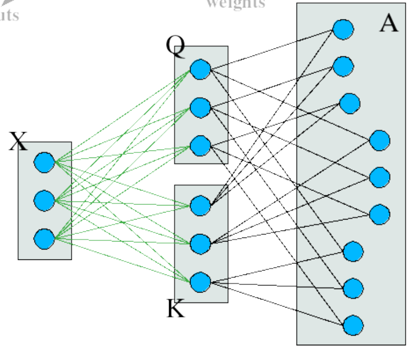
weights

outputs

Self-attention (transformer) :

$$w(i,j) \approx A_{ij} = q_i k_j = (W^q X)_i . (W^k X)_j$$

Learned weights



From NL-Means to Self-Attention

NL-Means :

$$y_i = \frac{1}{\pi(i)} \sum_j w(i,j)\, x_j$$

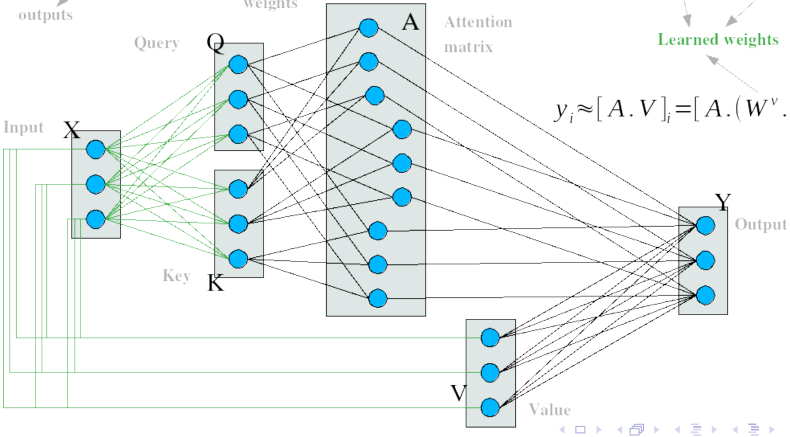
outputs, weights, inputs

Self-attention (transformer) :

$$w(i,j) \approx A_{ij} = q_i k_j = (W^q X)_i . (W^k X)_j$$

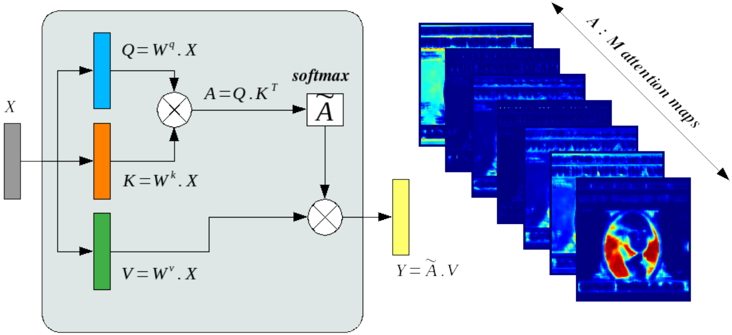
Learned weights

$$y_i \approx [A.V]_i = [A.(W^v . X)]_i$$



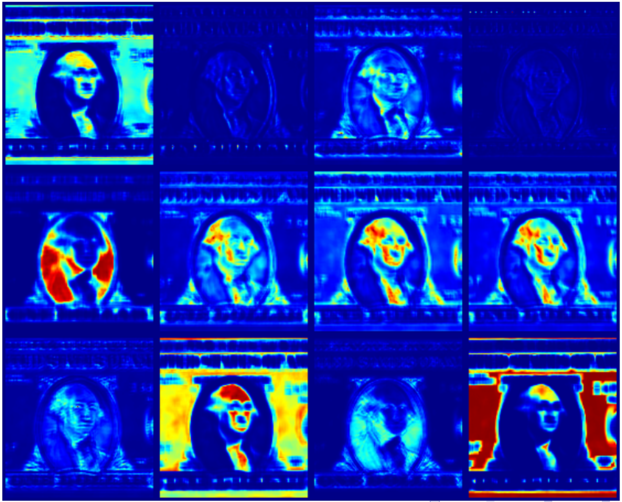
...as an end-to-end version

In end-to-end version, X and Y are 2 images of size N (= number of pixels!), W_q, W_k and W_v are learned weight matrices of sizes $N \times N$, $M \times N$, and $M \times N$ respectively, and the attention matrix A has size $N \times M$.



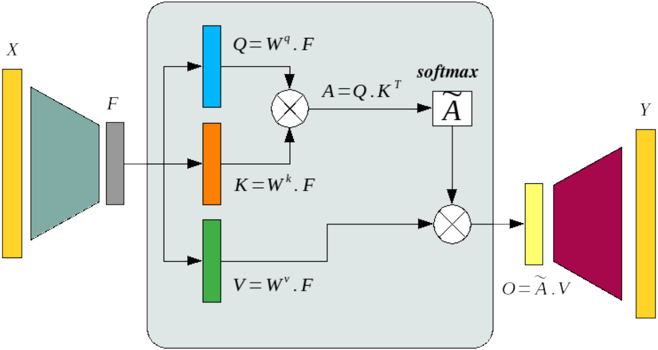
Self-Attention Module

Example of Attention maps for Denoising



...as a module version

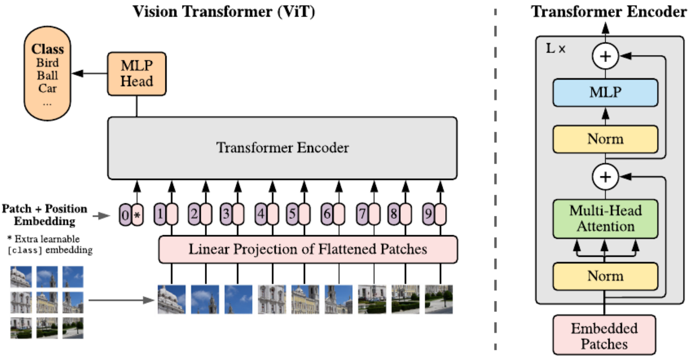
For images, self-attention modules are generaly applied on smaller images (patches), on smaller feature maps, on patch or region (superpixels!) embeddings...



Self-Attention Module

Vision Transformer ViT [Dosovitskiy21]

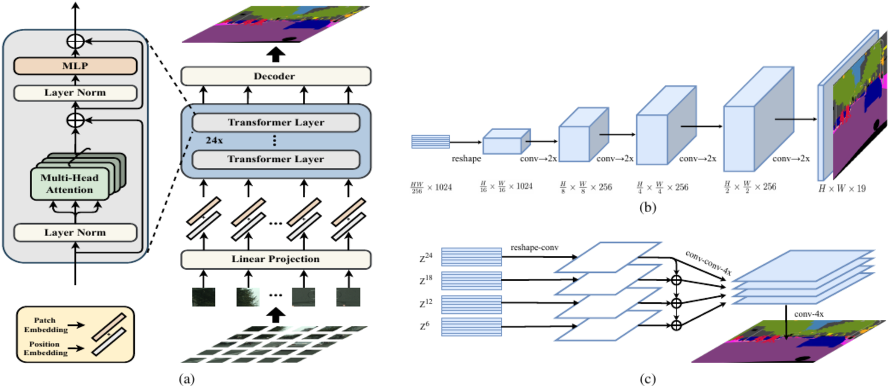
ViT is used as a module in most modern (including foundation) models, in particular for segmentation. It is based on applying the transformer to a (learned) linear projection of image patches:



Properties of ViT [Dosovitskiy21]

- Since the self-attention module is composed of three fully connected layers, the transformer originally *ignores the order* between the components of its input, and then the *spatial relations*, that play a vital role for images.
- To overcome this weakness, ViT joins to each patch embedding, a *vector encoding its relative position* in the image (positional encoding).
- Similar to multiple channels in CNN, a Transformer layer generally has several self-attention modules (multi-head attention), that allow to *encode different concepts* that are useful and complementary for a given task.

SEgmentation TRansformer [Zheng21]



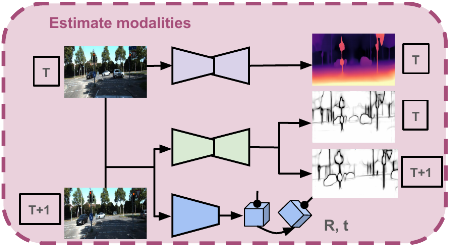
Outline

How to get rid of supervision?

- Foundation models leverage semi supervised learning based on prompt engineering, either visual (points, bounding boxes, free curves,...) or textual (using multimodal models).
- Trained under such framework, Segment Anything Model **[Kirillov23]** shows impressive zero-shot performance that in turn, allows to build a huge densely annotated image segmentation dataset, likely to improve supervised models, and so on...
- Self supervised segmentation is only emerging; it is based on auxiliary tasks that can be learned autonomously, and that provide objective semantic clues.

Towards fully self supervised segmentation [Hariat24]

As examples, depth (distance to the focal plane) and motion (optical flow) can both be learned in a self supervised way, and provide physical clues to separate objects or surfaces:



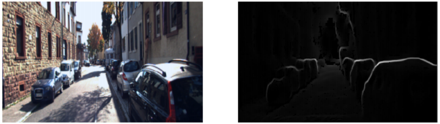
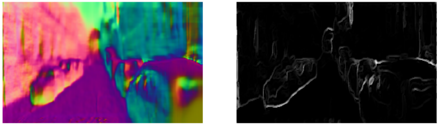


Image (left) and Laplacian activation of depth (right).



Normal (left) and gradient activation of normal (right).

Conclusion and take-away messages

- State-of-the-Art Semantic (and Instance) Segmentation models exploit the most powerful encoders, trained for Classification task on the largest existing datasets.
- Decoders are mostly trained in a fully supervised manner for Segmentation, using a variety of strategies combining interpolation, learned upsampling kernels, skip connection, multi-layer aggregation,...
- Foundation models provide Zero-shot Segmentation, based on large pre-trained encoders and prompt based weak supervision.
- Fully self supervised segmentation is emerging, by leveraging physics based auxiliary tasks.

Bibliography - Before DNN

[Vincent91] L. Vincent and P. Soille
Watersheds in digital spaces: an efficient algorithm based on immersion simulations
IEEE Trans. on Pattern Analysis and Machine Intelligence, 13(6), pp 583-598, 1991.

[Kato12] Z. Kato and J. Zerubia
Markov Random Fields in Image Segmentation
Now Editor, World Scientific, 2012, Foundation and Trends in Signal Processing.

[Achanta12] R. Achanta et al.
SLIC Superpixels Compared to State-of-the-art Superpixel Methods
IEEE Trans. on Pattern Analysis and Machine Intelligence, 34(11), pp 2274-2282, 2012.

Bibliography - CNN

- **[Shelhamer15]** E. Shelhamer, J. Long and T. Darrell
  Fully Convolutional Networks for Semantic Segmentation
  CVPR 2015
- **[Badrinarayanan15]** V. Badrinarayanan, A. Handa and R. Cipolla
  SegNet: A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling
  ArXiv, abs/1505.07293, 2015.
- **[Ronneberger15]** O. Ronneberger, P. Fischer and Th. Brox
  UNet: Convolutional Networks for Biomedical Image Segmentation
  MICCAI 2015

Bibliography - Transformers

- **[Dosovitskiy21]** A. Dosovitskiy et al.
An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale
ICLR 2021

- **[Zheng21]** S. Zheng et al.
Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers
CVPR 2021, pp. 6881-6890

- **[Kirillov23]** A. Kirillov et al.
Segment Anything
International Conference on Computer Vision (ICCV), 2023, pp. 3992-4003