Image Segmentation

Visual Understanding

What you will learn in this course

Low-Level Vision

Image Processing:
Photometric Image formation
Point Operations
Histogram Equalization
Linear & Non-Linear filters
Convolution & Cross correlation

Feature Detection:
Edge detection
Corner detection
Line detection

Feature Description:
SIFT
Feature Matching
Image Stitching
Panorama

Geometric Vision

Transformation and Camera Mode:
Geometric Image Formation
Pinhole Camera Model

Camera Calibration:
Estimating intrinsics
Estimating extrinsics

Visual Understanding

Object Detection & Tracking:
Single-stage detectors
Two-stage detectors
YOLO
Object tracking

Image Segmentation:
Traditional Image Segmentation
Learning-based Image Segmentation

Machine Learning for CV

Machine Learning for Classification:
k-NN
Linear Classifier

Deep Learning:
Optimization
Neural Network
CNN
CNN architectures:

Generative & Geometric Learning

Generative Models:
AE, VAE, GAN

3D Vision and 3D Deep Learning:
3D shape representations
Geometric deep learning overview
3D Deep Learning models : pointnet

What we will learn today

- Classical Image Segmentations
  - k-Mean Clustering
  - Mean Shift
  - Graph Based Segmentation
- Learning-based Image Segmentation
  - Semantic Segmentation
  - Instance Segmentation
  - Panoptic Segmentation

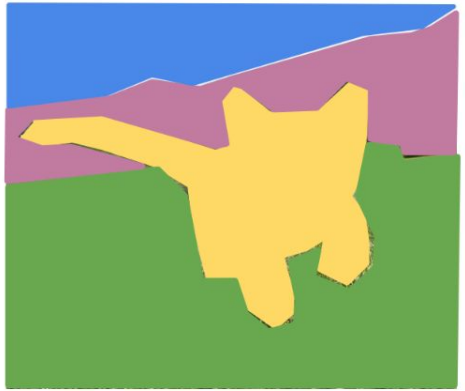
Computer Vision Tasks

Classification

CAT

No spatial extent

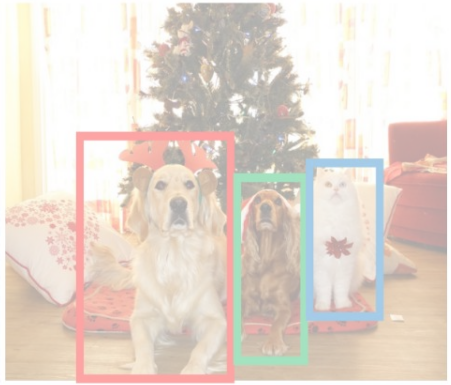
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

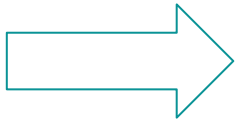
Multiple objects

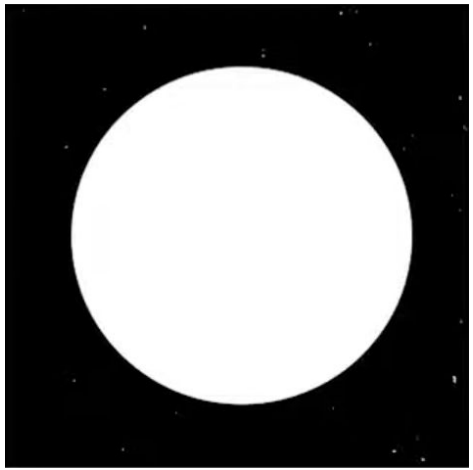
Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

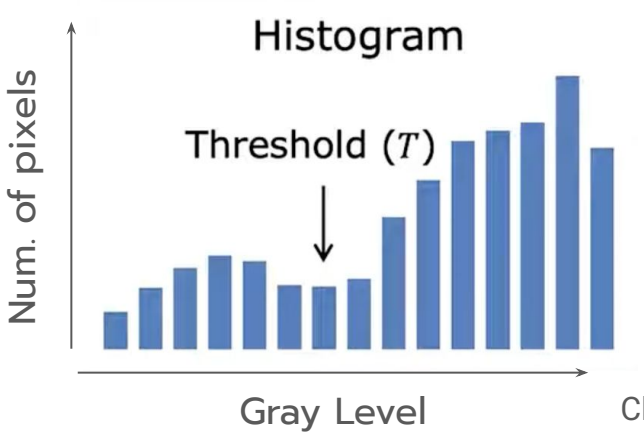
What is Image Segmentation ?

Simple image segmentation: Binary Segmentation

Gray Image
g(x,y).







What is Image Segmentation ?

How can we segment the image into meaningful regions

Segmentation and Grouping in Humans

Gestalt Psychology

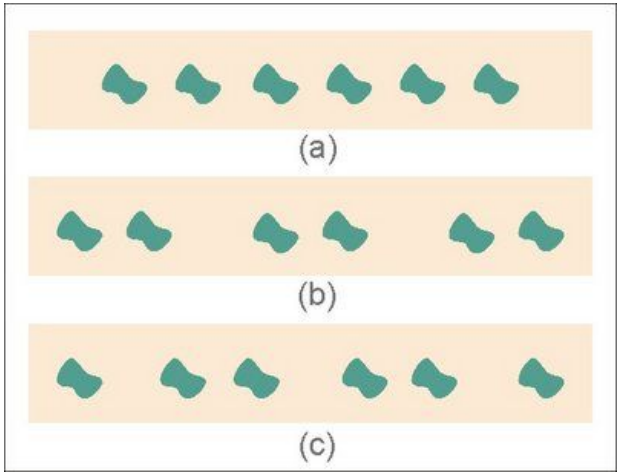
When we look at the world, we usually perceive complex scenes composed of many **groups** of objects on some background, with the objects themselves consisting of **parts**, which may be composed of smaller parts, etc

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

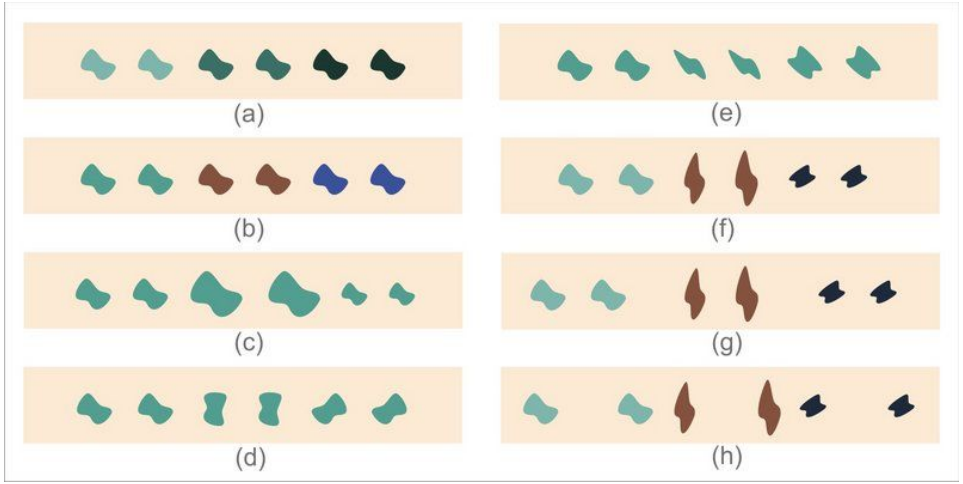
Segmentation and Grouping in Humans

Gestalt Psychology

Proximity Principle



Similarity Principle

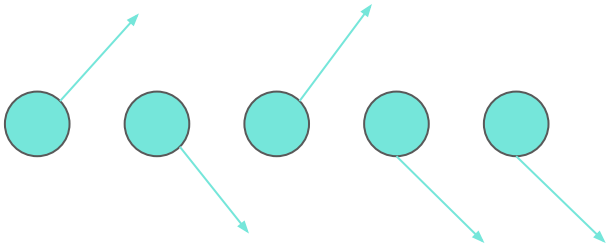


Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Segmentation and Grouping in Humans

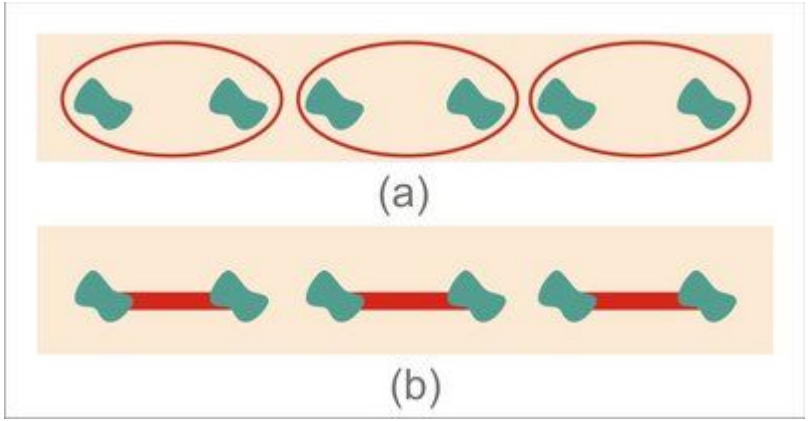
Gestalt Psychology

Common fate



Objects with similar motion or change in appearance are grouped together.

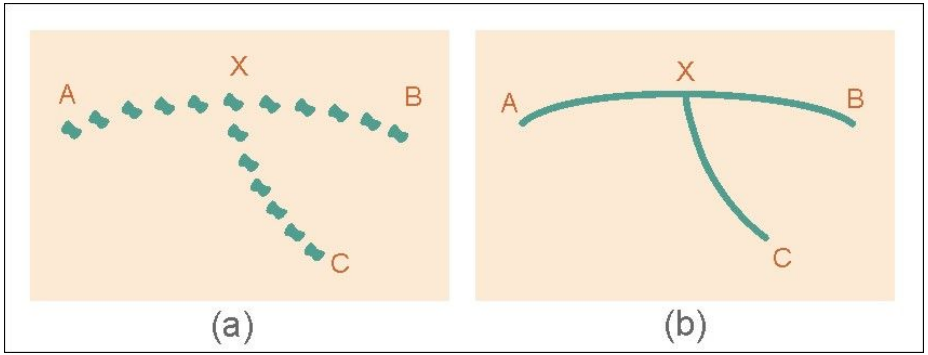
Common region/Connectivity Principle



Segmentation and Grouping in Humans

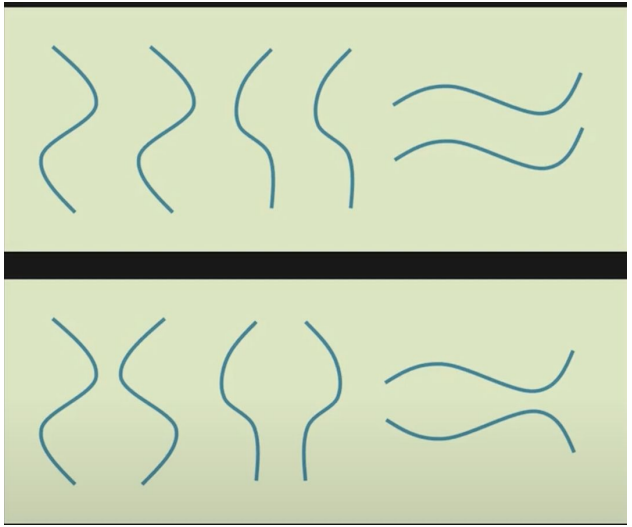
Gestalt Psychology

Continuity Principle



Features on a continuous curve are grouped together.

Symmetry Principle



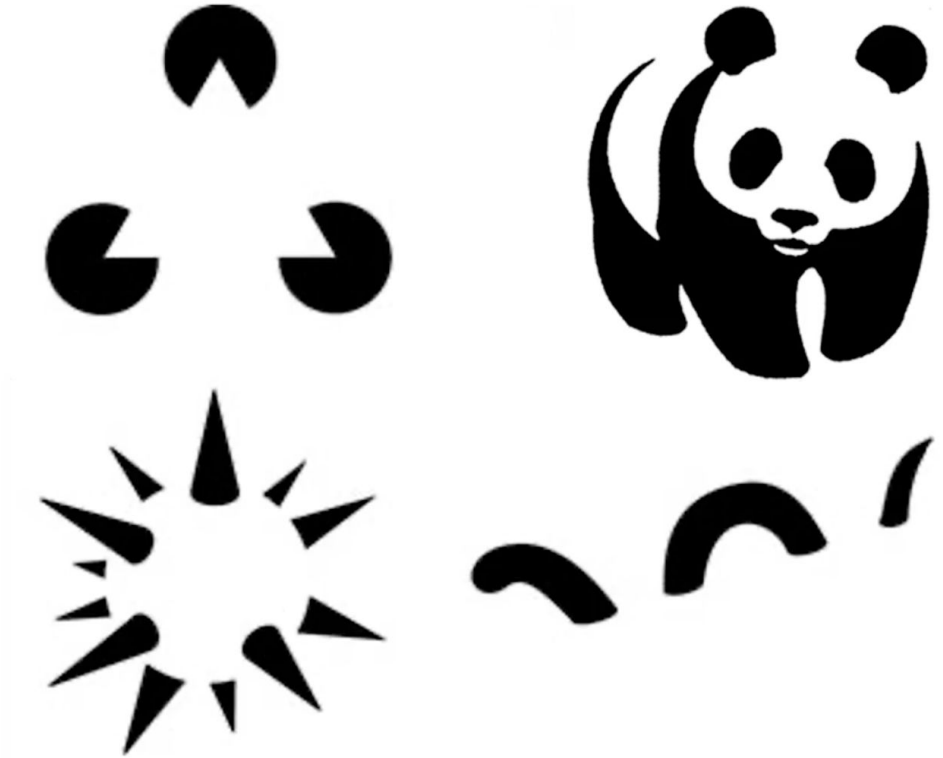
Parallel and symmetrical features are grouped together.

Segmentation and Grouping in Humans

Gestalt Psychology

Illusory Contours

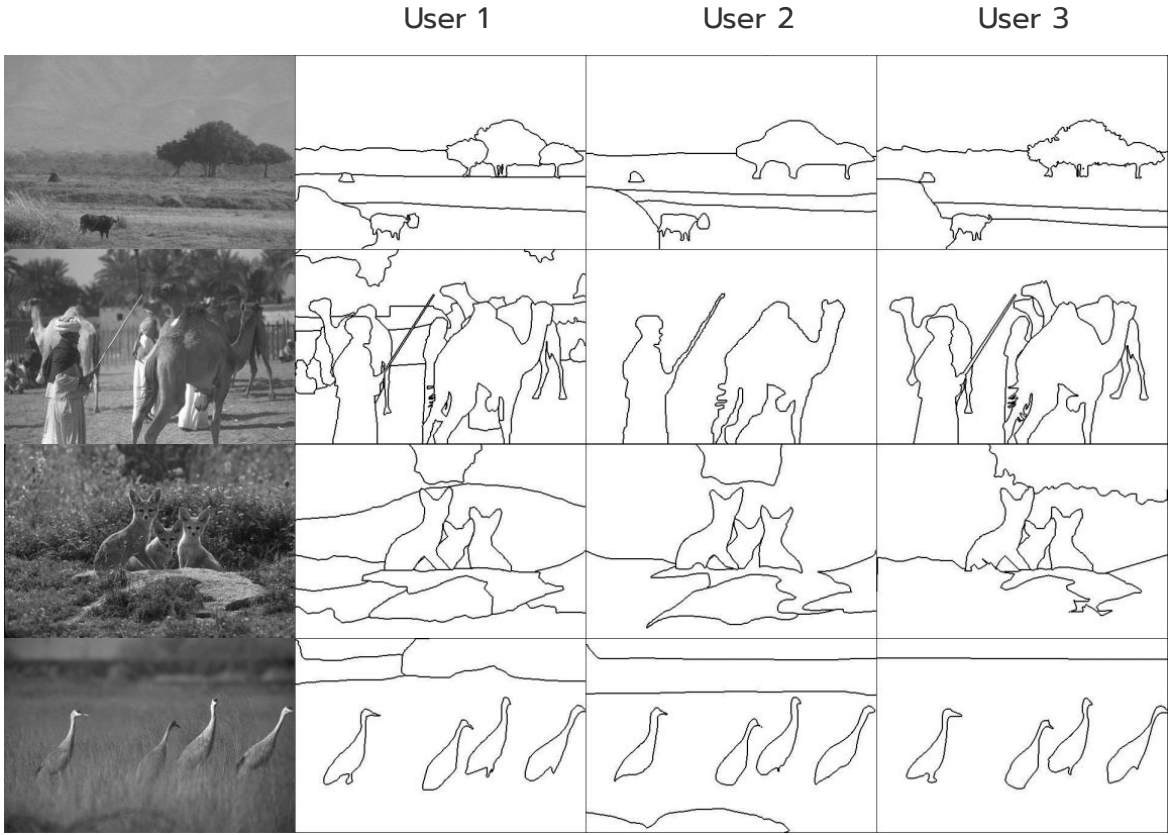
Illusory or subjective contours are perceived



Segmentation and Grouping in Humans

Segmentation by Humans

Segmentation is highly **subjective**.



Source: A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics, D. Martin

Segmentation Strategies

Segmentation is intuitive for us.
Very hard to translate these intuitions to an algorithm.

Top down segmentation:

- pixels belong together because they are from the same object.
- This is a hard problem, we do not know how to solve this yet to some extent.

Bottom up:

- pixels belong together because they look similar

What is Segmentation ?

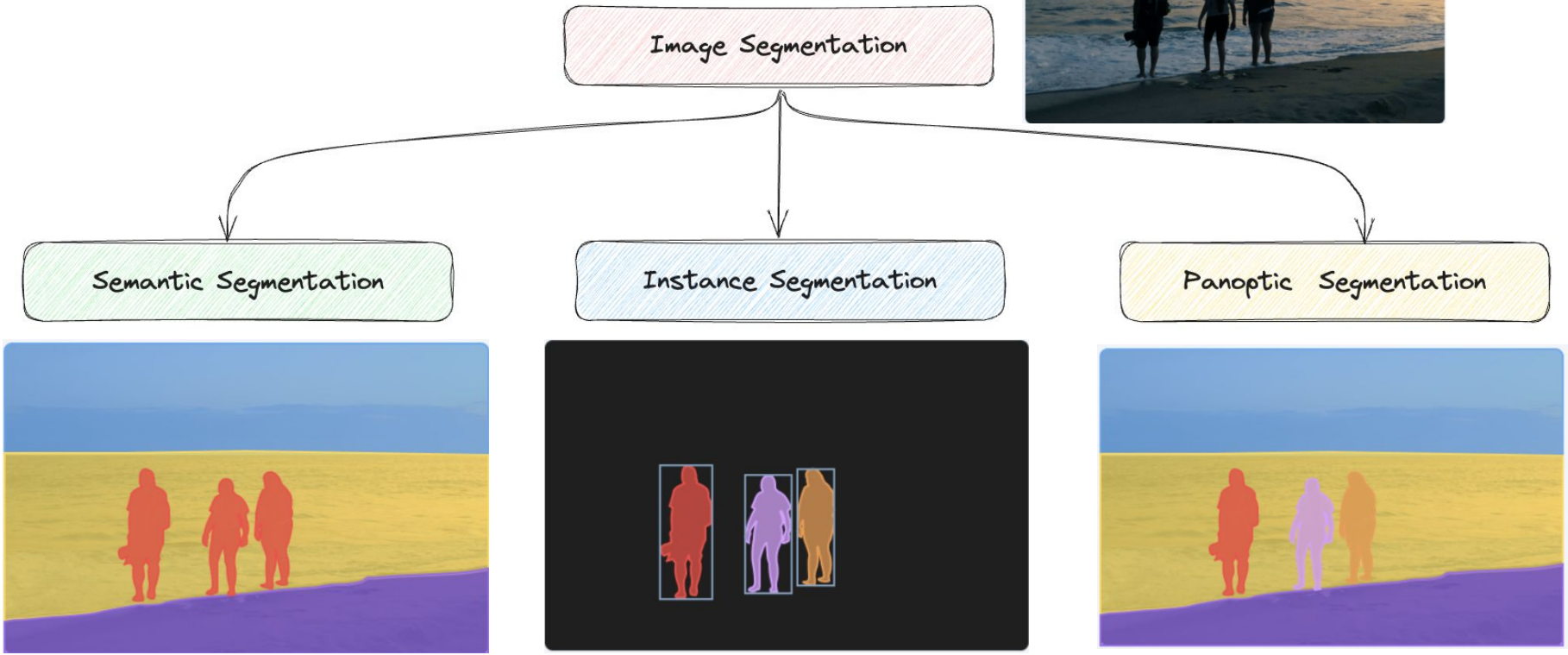


Image Segmentation Techniques

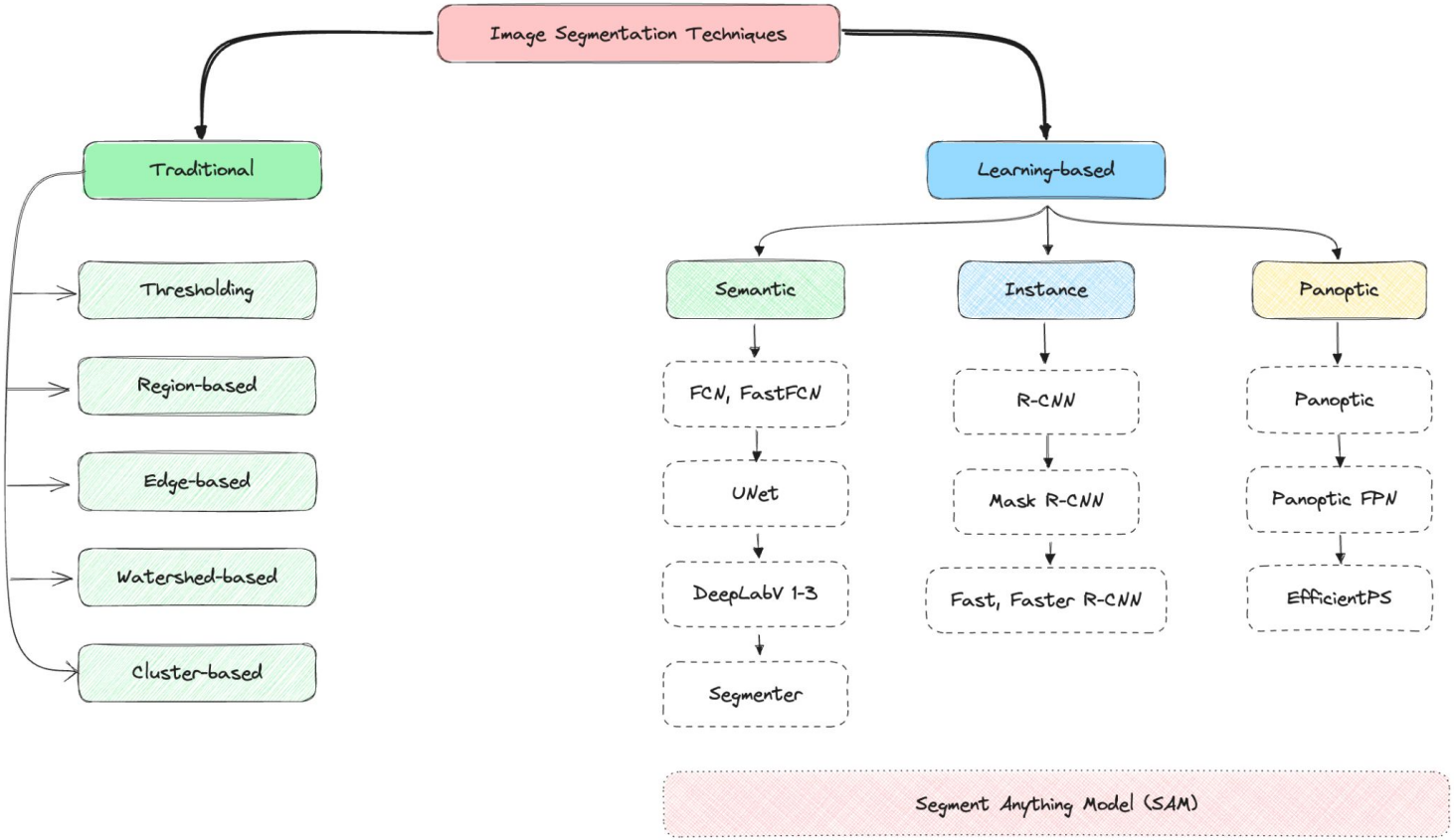


Image Segmentation

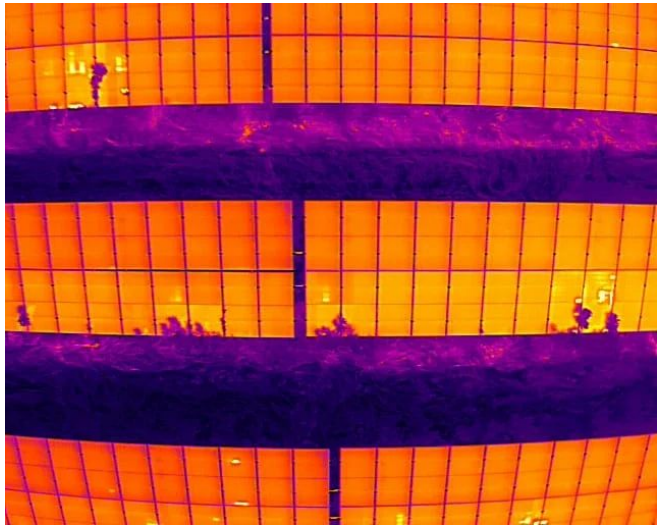
Classical

Classical Image Segmentation Techniques

Why should we care about Classical methods?

World largest Floating hybrid solar farm.
Sirindhorn Dam in Ubon Ratchathani

Why should we care about Classical methods?



Cherdsak Kingkan

Why should we care about Classical methods?

Techniques that are used

- Image restoration - Lens distortion
- Image stitching
- Image Segmentation

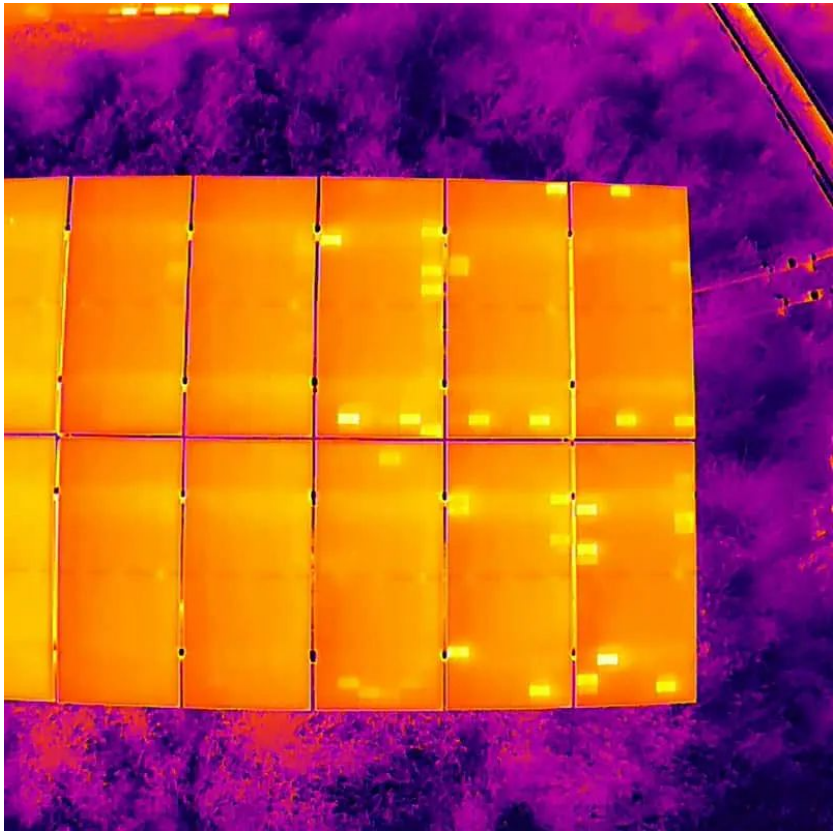
not work on deep learning

can extract the heat inside each bounding box

Thermal camera by drone

Graph based segmentation method

not all problems can be solved by deep learning.



Segmentation as Clustering

Visual Characteristics of Pixels

Visual similarity can be based on:

- Brightness
- Color
- Position

Direct measurements

- Depth
- Motion
- Texture
- Material

Can be computed

- Etc...

Segmentation as Clustering

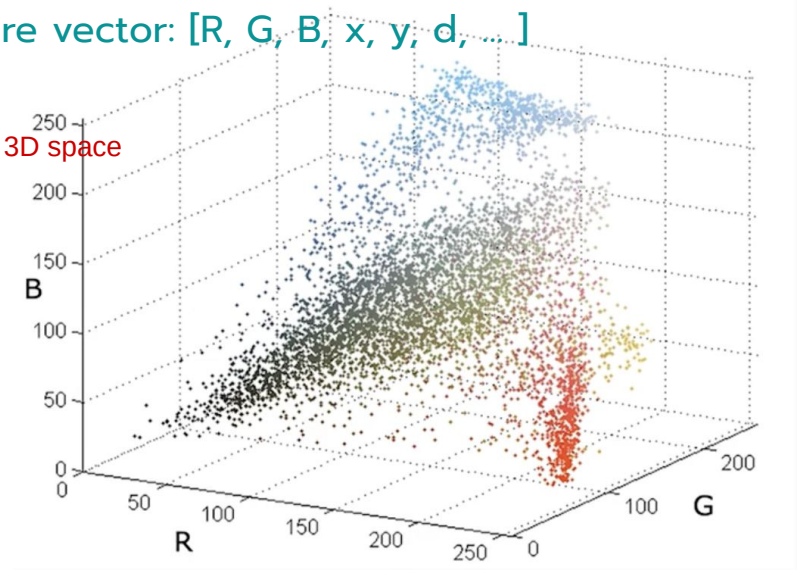
Pixels in Euclidean Space

Euclidean Space: Generalization of 3D Cartesian space to higher dimensions.

Input Image

Pixels as feature vector: [R, G, B, x, y, d, …]

can project any to 3D space



Pixel RGB Color Distribution

(Color of feature point ≡ Color of image pixel

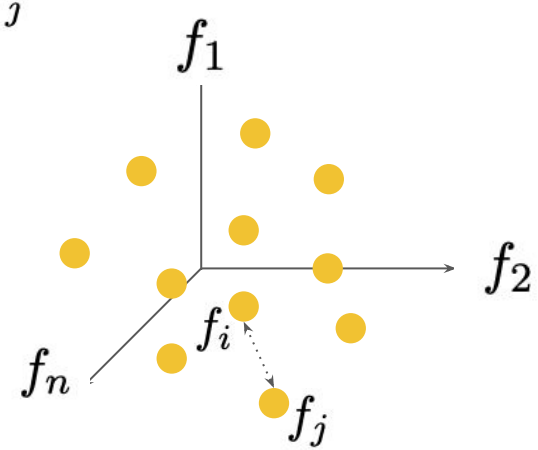
Segmentation as Clustering

Pixels in Euclidean Space

Let i and j be two pixels whose features are f_i and f_j

$\mathcal{L}^2$ Distance between f_i and f_j :

$$S(f_i, f_j) = \sqrt{\sum_k (f_{ik} - f_{jk})^2}$$

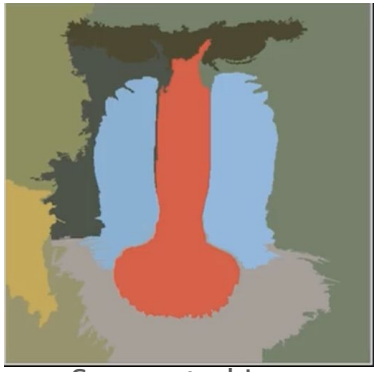


Smaller the Distance, Greater the Similarity

Segmentation as Clustering

Clustering Similar Pixels

Input Image

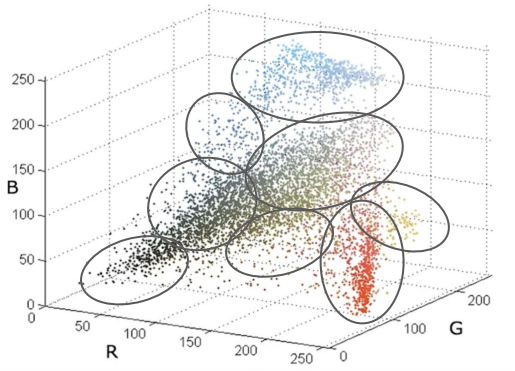


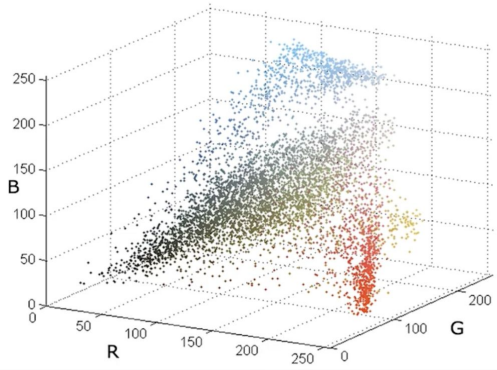
Segmented Image

which pixel belong to which cluster

Pixel RGB Color Distribution

(Color of feature point ≡ Color of image pixel





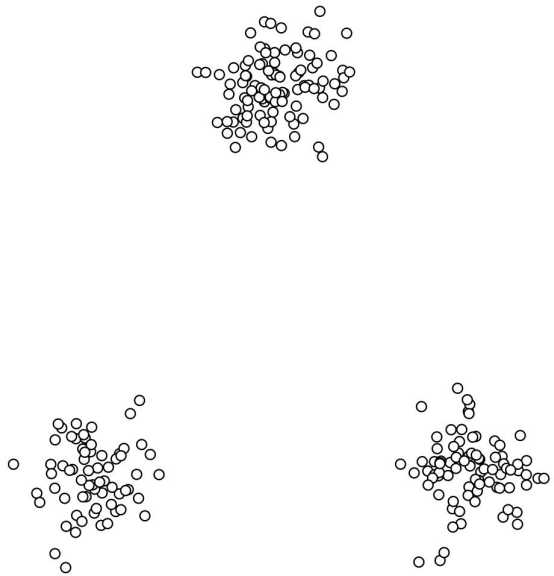
Color-Coded Distribution

(Color of feature point ≡ Color of image pixel

k-Means Segmentation

3-Mean Clustering Example

Problem: Segment the given pixel feature distribution into 3 clusters.



each of these dot are the pixel we are projecting from the image

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

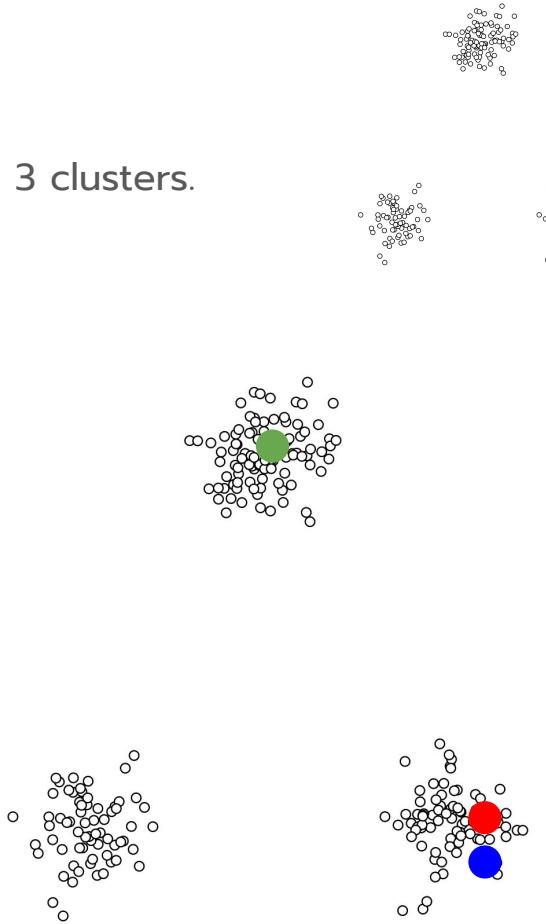
k-Means Segmentation

3-Mean Clustering Example

Problem: Segment the given pixel feature distribution into 3 clusters.

Step 1: Randomly generate the initial centroids **(means)** of the 3 clusters.

3 segments > initial centroid



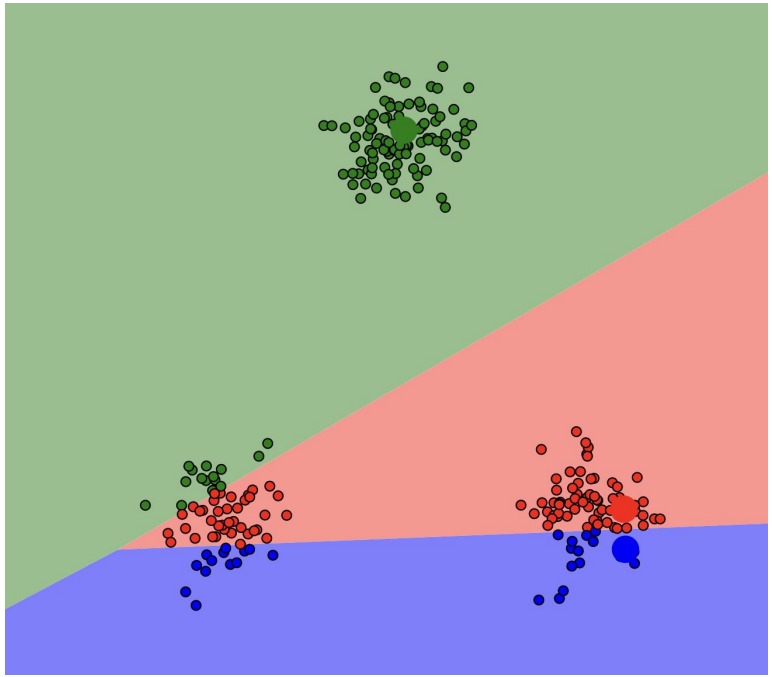
k-Means Segmentation

3-Mean Clustering Example

Problem: Segment the given pixel feature distribution into 3 clusters.

Step 1: Randomly generate the initial centroids **(means)** of the 3 clusters.

Step 2: Then create 3 clusters by assigning each feature point to the nearest mean.

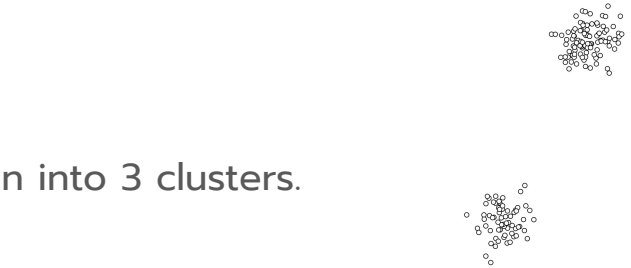


Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

3-Mean Clustering Example

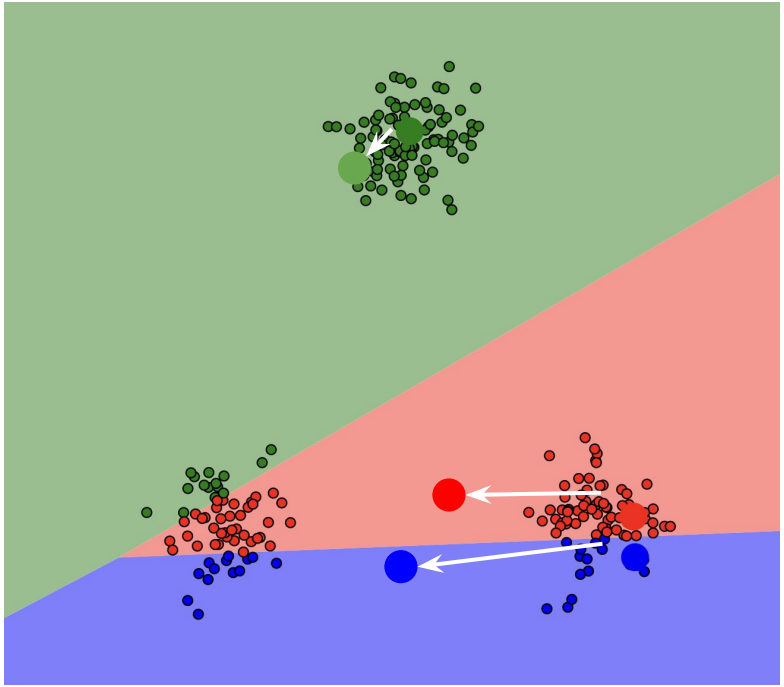
Problem: Segment the given pixel feature distribution into 3 clusters.



Step 1: Randomly generate the initial centroids **(means)** of the 3 clusters.

Step 2: Then create 3 clusters by assigning each feature point to the nearest mean.

Step 3: Recompute the mean of each cluster and assign each feature point to the new nearest mean.



Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

3-Mean Clustering Example

Problem: Segment the given pixel feature distribution into 3 clusters.

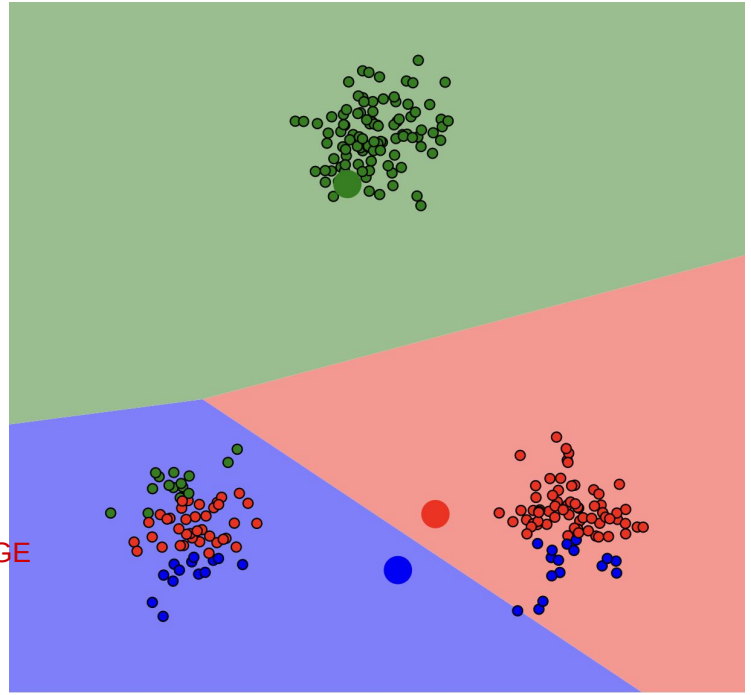
Step 1: Randomly generate the initial centroids **(means)** of the 3 clusters.

Step 2: Then create 3 clusters by assigning each feature point to the nearest mean.

Step 3: Recompute the mean of each cluster and assign each feature point to the new nearest mean.

Step 4: Repeat steps 2 and 3 until convergence.

reassign each AND EVERY PT INTO THE NEW MEANS UNTIL WE CONVERGE

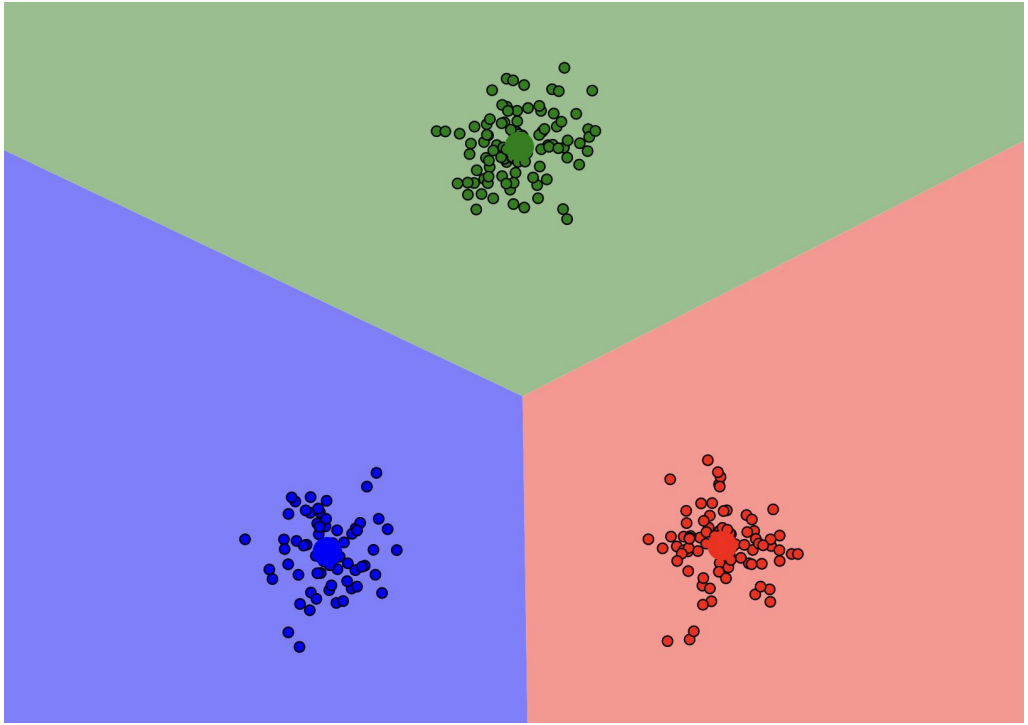


Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

3-Mean Clustering Example

Problem: Segment the given pixel feature distribution into 3 clusters.



Demo:
https://www.naftaliharris.com/blog/visualizing-k-means-clustering/

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

k-Mean Clustering

Given: Image with *N* pixels and number of clusters *k*.

Task: Find the *k* clusters. Pseudo code

Clustering:

1. Pick *k* points randomly as the initial centroids (means) { $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_k$ } of the *k* clusters in feature space.
2. For each pixel $\mathbf{x}_j$ find nearest cluster mean $\mathbf{m}_i$ to pixel's feature $\mathbf{f}_j$ and assign pixel to cluster i.
3. Recompute mean for each cluster using its assigned pixels.
4. If changes in all *k* means is less than a threshold ε, stop. Else go to step 2.

means move less than 0.5 we stop computing

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

k-Mean Initialization Methods

Method 1: select "k" random feature points as initial centroid. If two point are very close, resample.

Method 2: select "k" uniformly distributed means within the rage of the distribution.

Method 3 (Preferred): perform k-mean clustering on subset of pixels and use the result as the initial means.

Innstead of randomly slect

you uniformly select the initial means

K-Means clustering

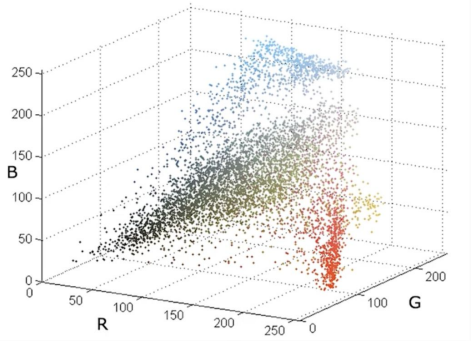
Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

k-Mean Clustering Results (k = 2)

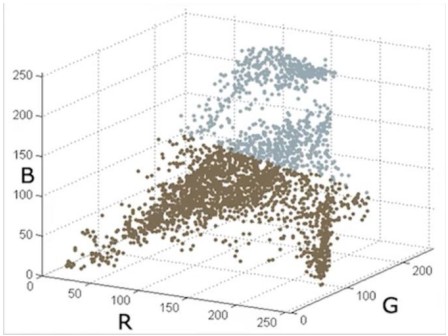
Input Image

Segmented Image
(k = 2; {R,G,B} - space)



Pixel RGB Color Distribution

(Color of feature point ≡ Color of image pixel



Color-Coded Distribution

(Color of feature point ≡ Color of image pixel

increasing no of cluster

much better segmentation image

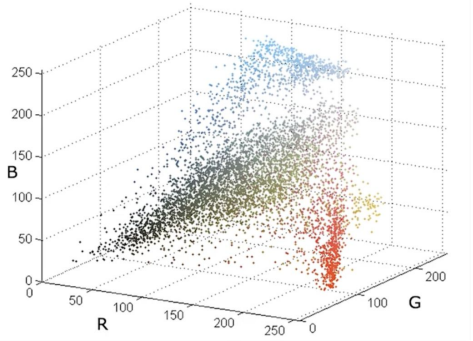
Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

k-Mean Clustering Results (k = 8)

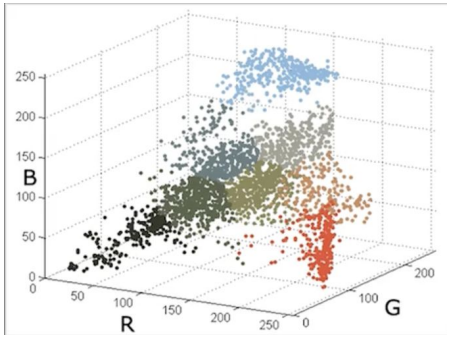
Input Image

Segmented Image
(k = 8; {R,G,B} - space)



Pixel RGB Color Distribution

(Color of feature point ≡ Color of image pixel



Color-Coded Distribution

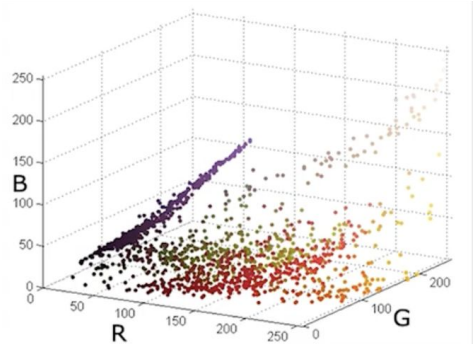
(Color of feature point ≡ Color of image pixel

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

k-Means Segmentation

k-Mean Clustering Results (k = 16)

Input Image



Pixel RGB Color Distribution

(Color of feature point ≡ Color of image pixel

can solve by increasing the features onto a 5 dimensional features with the same k=16

multiple disjoint

Segmented Image
(k = 16; {R,G,B} - space)

Note: Disjoint regions could belong to a single cluster.

higher no dimension of features

x , y depth

Segmented Image
(k = 16; {R,G,B,x,y} - space)

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Cherdsak Kingkan

k-Means Segmentation

Pros:

- Simple, fast to compute
- Converge to local minimum of within-cluster squared error

Cons:

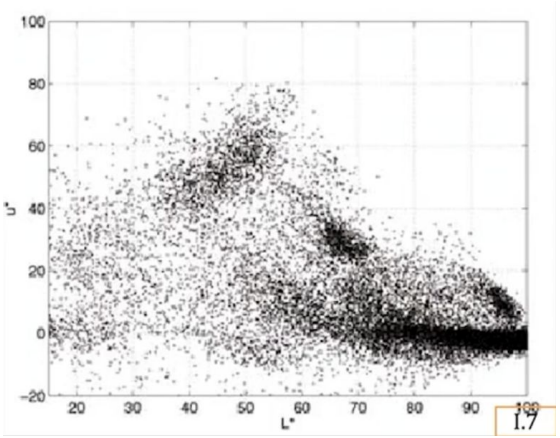
- Need to pick the number of cluster *k*
- Sensitive to initial center
- Sensitive to outlier

pick the number of k

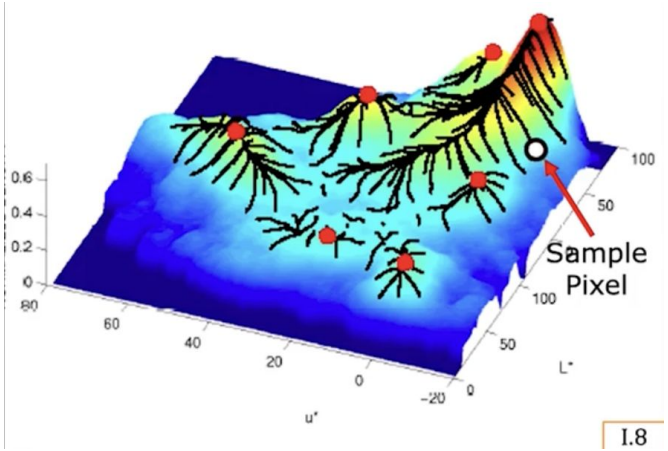
cannot run the algorithms

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Mean Shift Segmentation



Pixel Feature Distribution



Normalized Density

number of Pixel
have certain feature

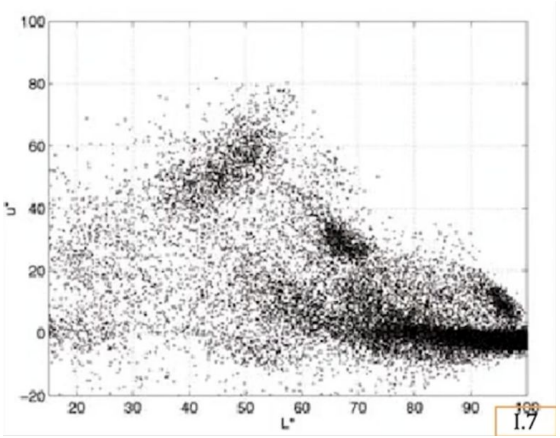
- Each hill represents a cluster.
- Peak (mode) of hills represents “center” of cluster.
- Each pixel climbs the steepest hill within its neighborhood
- Pixel assigned to the hill (cluster) it climbs.

which pixel has the same mode or mean (high frequency / occurency appears

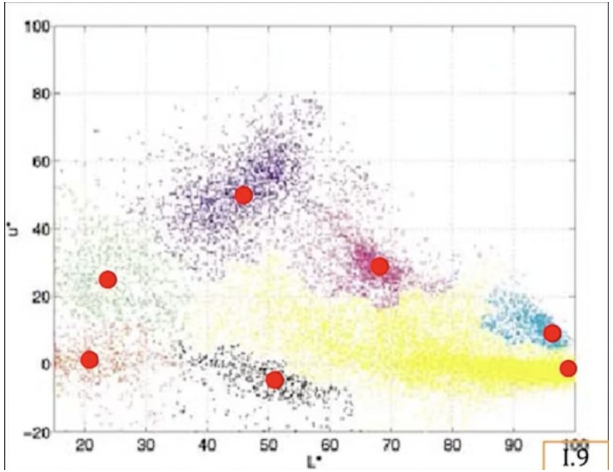
find whether which pixel belong to which hill (and belong to which cluster)

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Mean Shift Segmentation



Pixel Feature Distribution



Labeled Clusters and their centers

densely area in each group

- Each hill represents a cluster.
- Peak (mode) of hills represents “center” of cluster.
- Each pixel climbs the steepest hill within its neighborhood
- Pixel assigned to the hill (cluster) it climbs.

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space

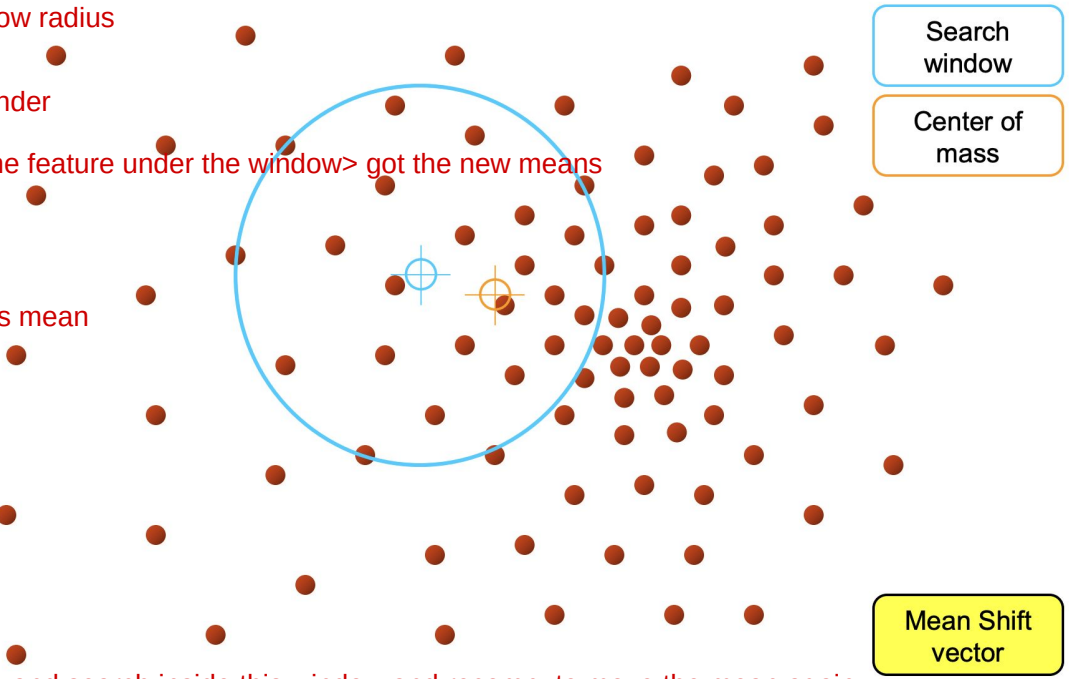
random selecting

search window- a window radius

how many point fall under

recompute mean of the feature under the window> got the new means

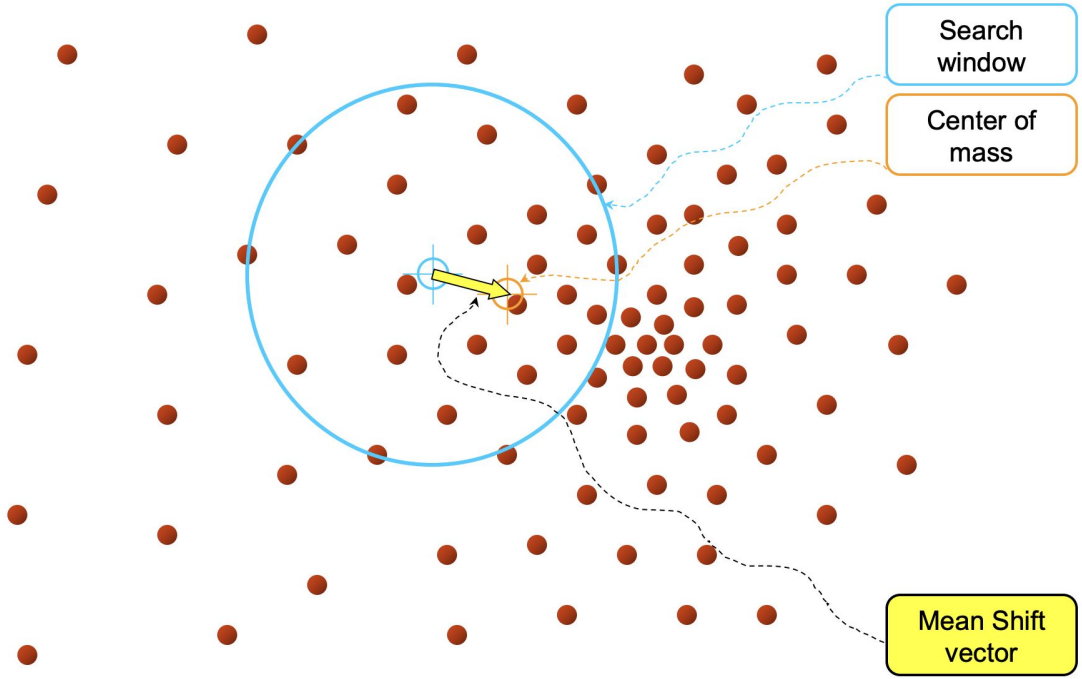
move from the previous mean



keep repeating this process and search inside this window and recompute move the mean again

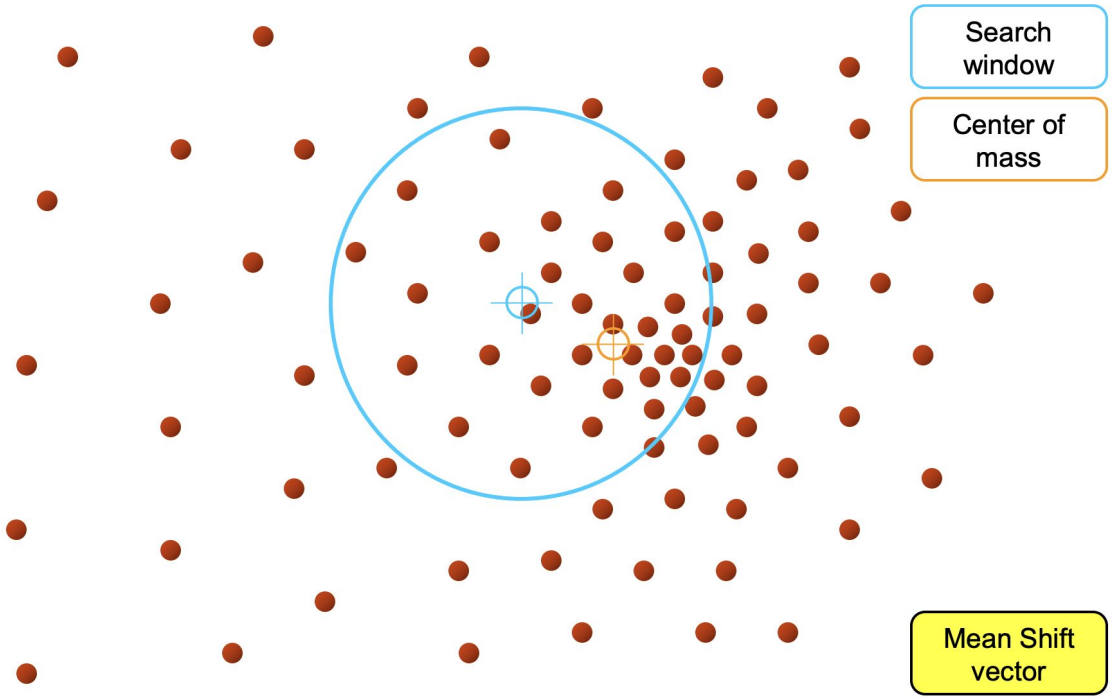
Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space



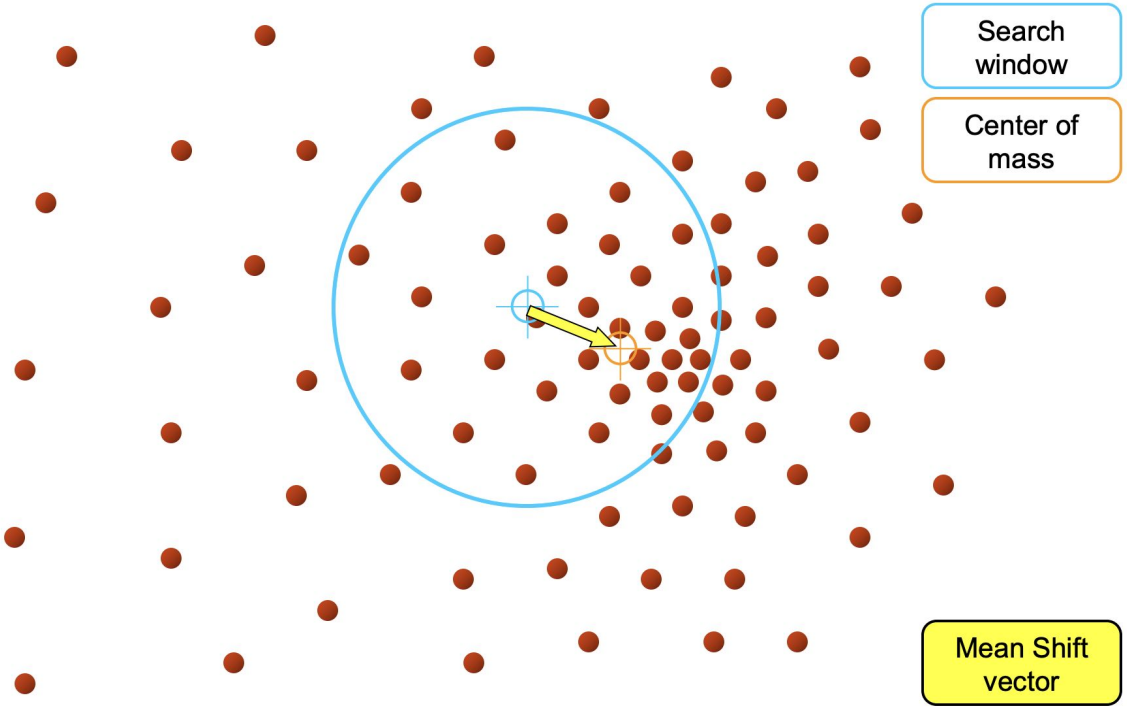
Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space



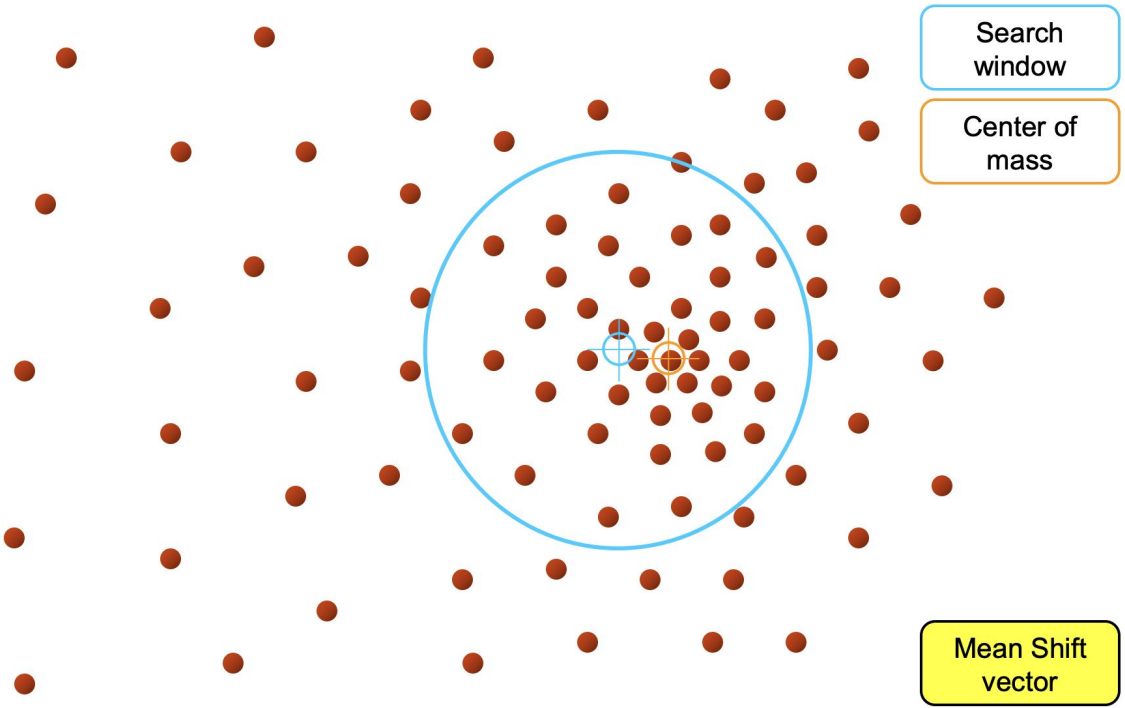
Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space



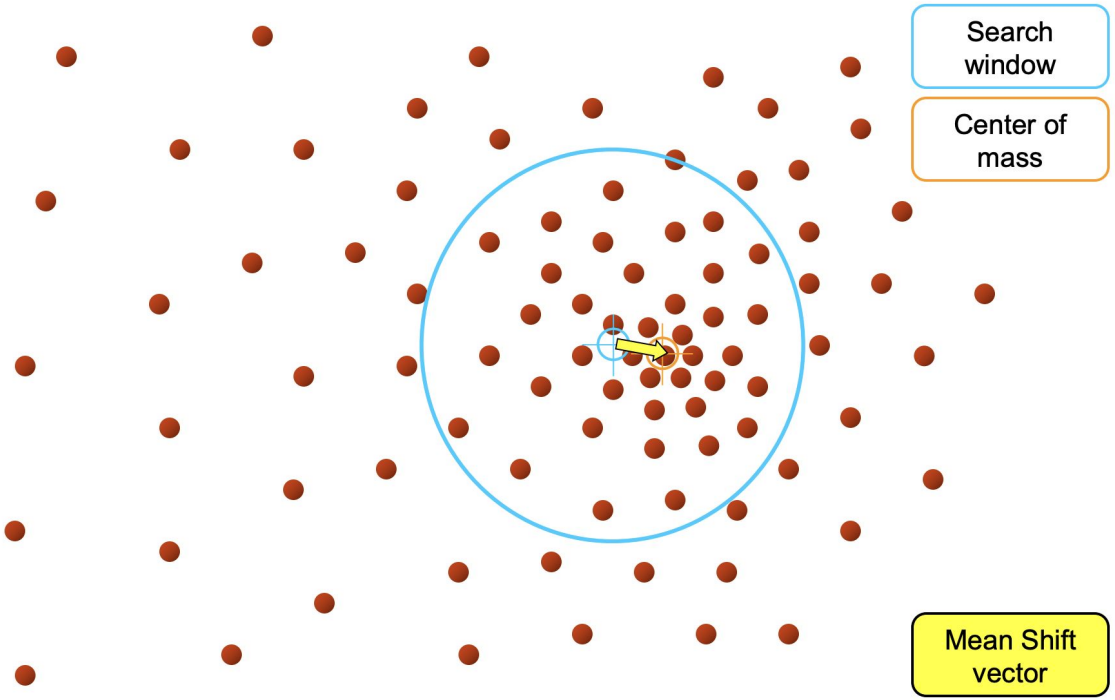
Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space



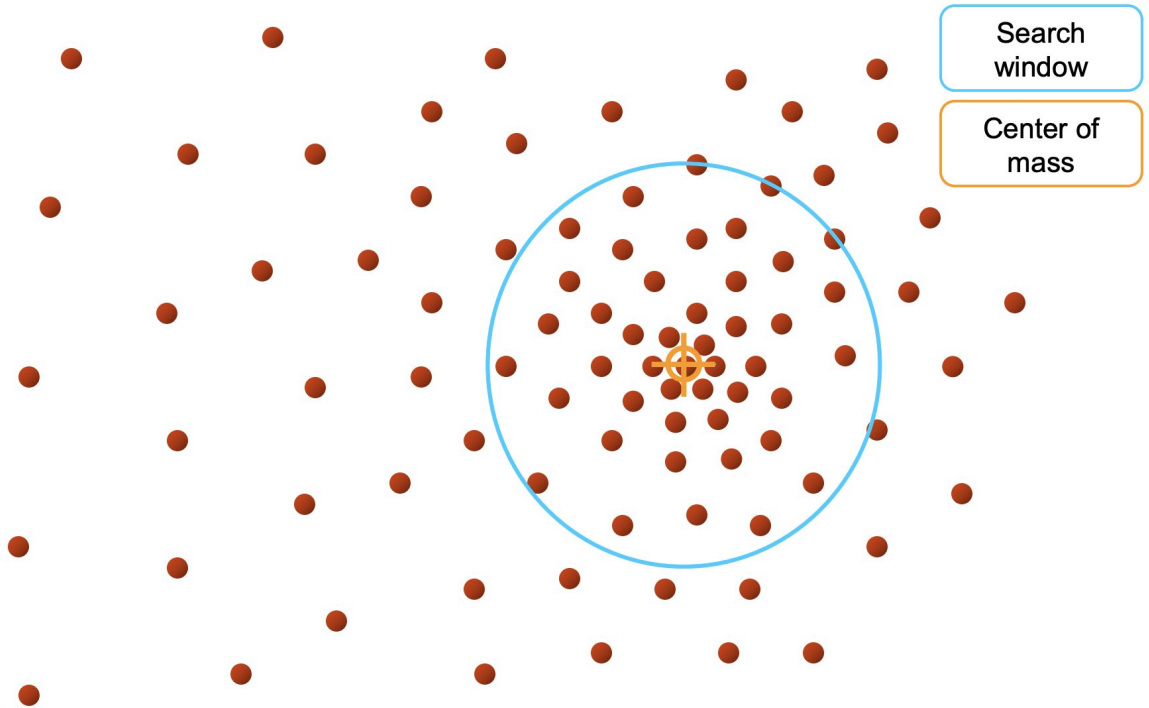
Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space



Mean Shift Segmentation

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space

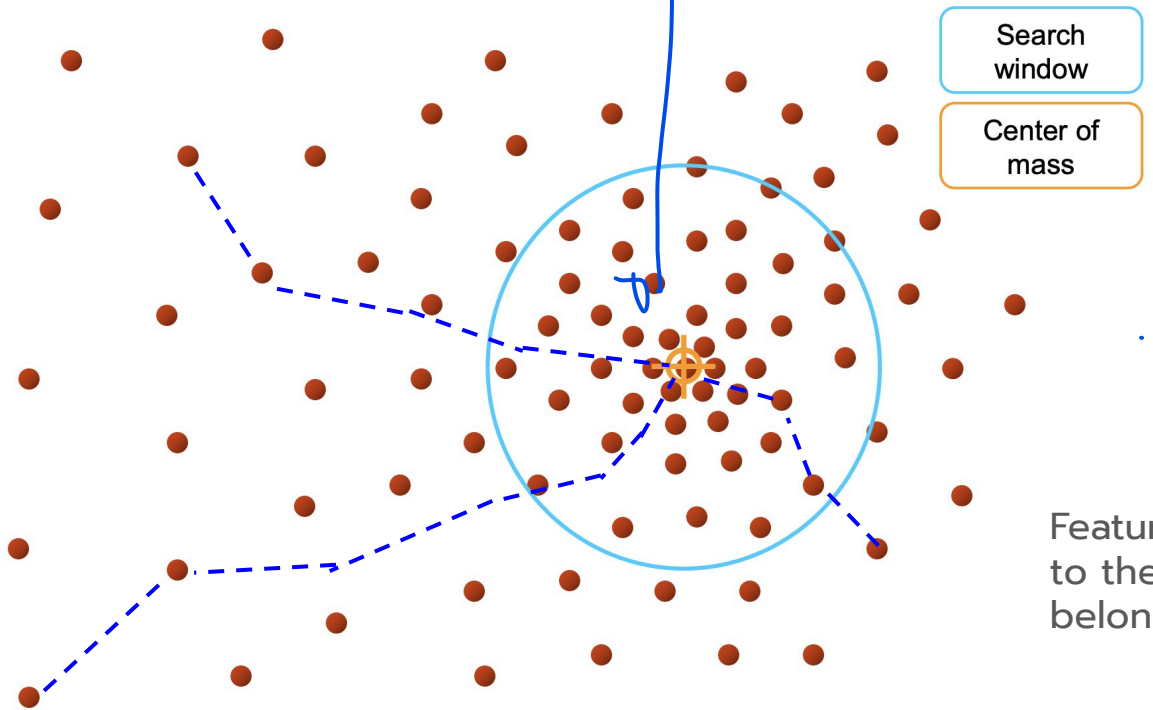


Mean Shift Segmentation

hill / mode of this group clim=mbing tothe same mode belongs to the same cluster (MEAN SHIFT> WE recomputer we shift to the new mean value)

The mean shift algorithm seeks **modes** or **local maxima** of density in the feature space

no stops moving



Features that **converge** to the same mode belong to same cluster

Mean Shift Segmentation

Given: Distribution of *N* pixels in feature space

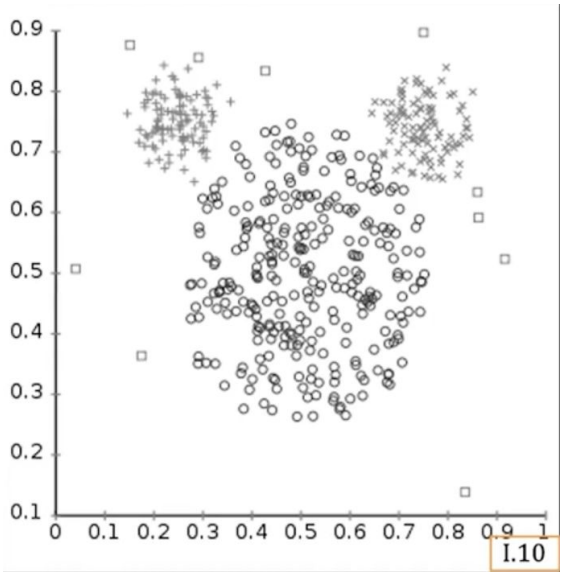
Task: Find mode (clusters) of the distribution

Clustering:

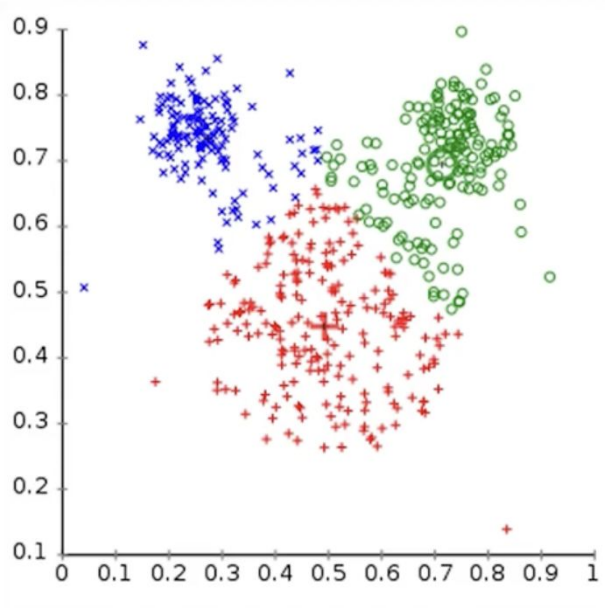
1. Set $\mathbf{m}_i = \mathbf{f}_i$ as initial mean for each pixel i.
2. Repeat the following for each mean $\mathbf{m}_i$:
   a. Place a window of size *W* around $\mathbf{m}_i$
   b. Compute centroid **m** within the window, Set $\mathbf{m}_i = \mathbf{m}$
   c. Stop if shift in mean $\mathbf{m}_i$ is less than a threshold ε.

   $\mathbf{m}_i$ is the mode
3. Label all pixels that have same mode as belonging to same cluster.

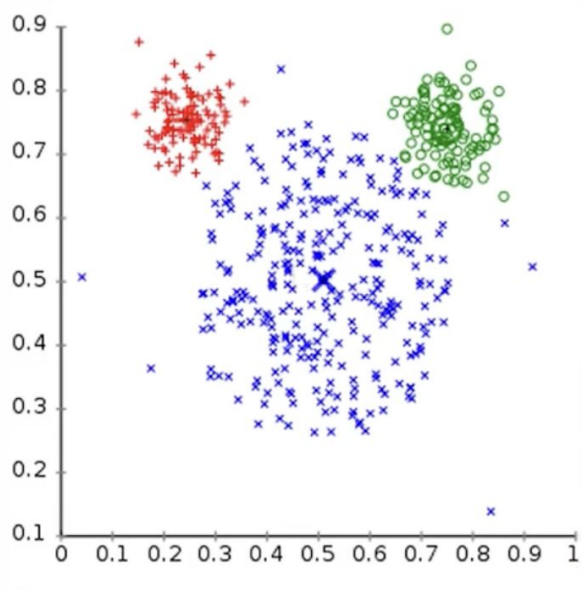
k-Mean vs. Mean Shift



Original Data



k-Means (k = 3)



Mean Shift

k-Mean vs. Mean Shift

Input Image

k-Means (k =16)

Mean Shift (W=21)

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Mean Shift Segmentation

Pros:

- No initialization required
- Good general-practice segmentation
- Flexible in number and shape of regions
- Robust to outliers

no need to specify the number of cluster we need (in k means need)

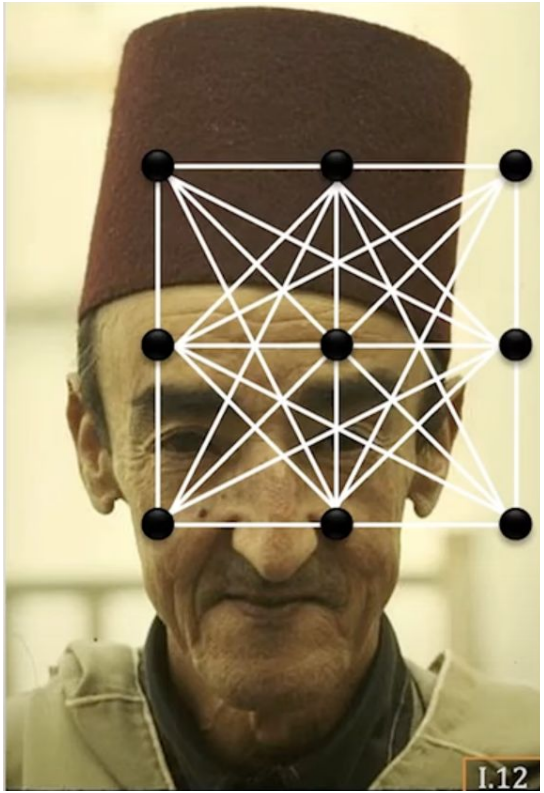
method how may hills are there (represents the output expected)

Cons:

- Computationally expensive
- Selection of window size
- Does not scale well with dimension of feature space

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Graph Based Segmentation



Images as Graphs:

- A vertex for each pixel.
- An edge between each pair of pixels.
- Graph notation: *G = (V, E)* where
  - V = set of vertices
  - E = set of edges
- Each edges is weighted by the affinity or **similarity** between its two vertices.

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Graph Based Segmentation

Measure Affinity

Let i and j be two pixels whose features are $\mathbf{f_i}$ **and** $\mathbf{f_j}$
(Features can be brightness value, color– RGB, L*u*v; texton histogram, etc)

Pixel Dissimilarity: $$S(f_i, f_j) = \sqrt{\left(\sum_k (f_{ik} - f_{jk})^2\right)}$$

Pixel Affinity: $$w(i,j) = A(f_i, f_j) = e^{\{\frac{-1}{2\sigma^2} S(f_i, f_j)\}}$$

weight in the edge
we have to compuite the weight for every image inside our image

Smaller the Dissimilarity, Larger the Affinity

segmentation only the vertices that are very close to each other

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Graph Based Segmentation

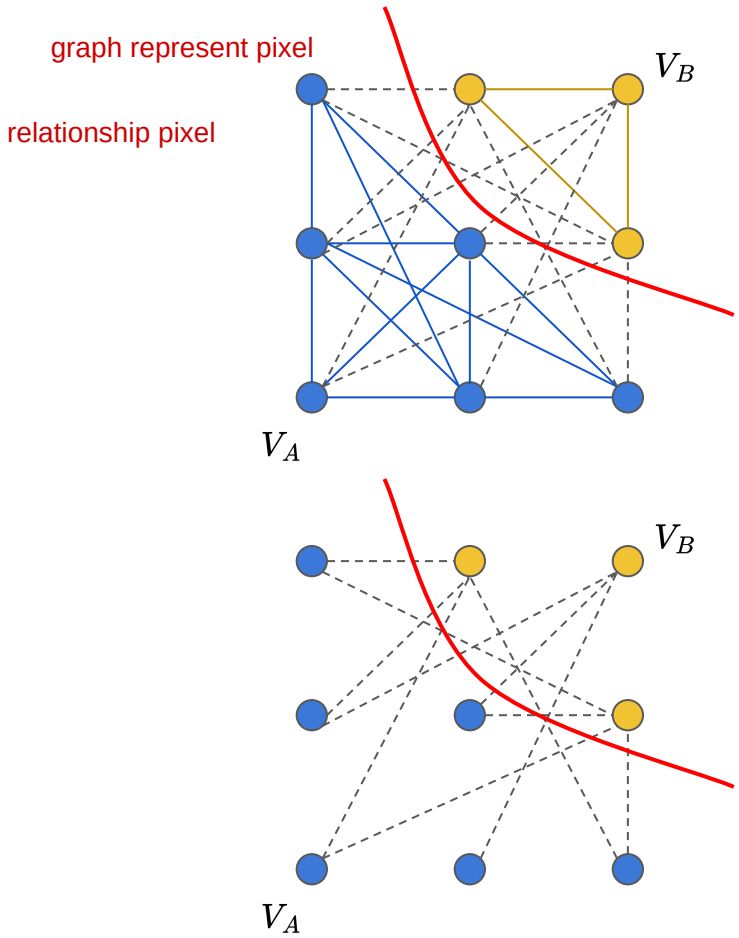
Segmentation by graph cut

Cut $C = (V_A, V_B)$ is a partition of vertices *V* of a graph *G = (V, E)* into two **disjoint subset** V_A and V_B

Cut-Set: Set of edges whose vertices are in different subsets of partition.

Cost of Cut : Sum of weights of cut-set edges.

$$cut(V_A, V_B) = \sum_{u \in V_A, v \in V_B} w(u, v)$$



Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Graph Based Segmentation

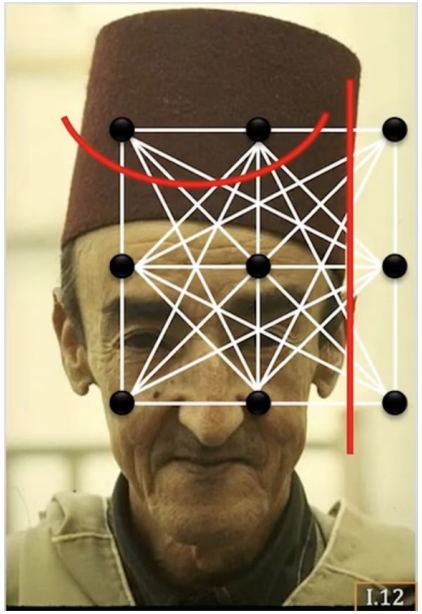
Criteria for Graph Cut

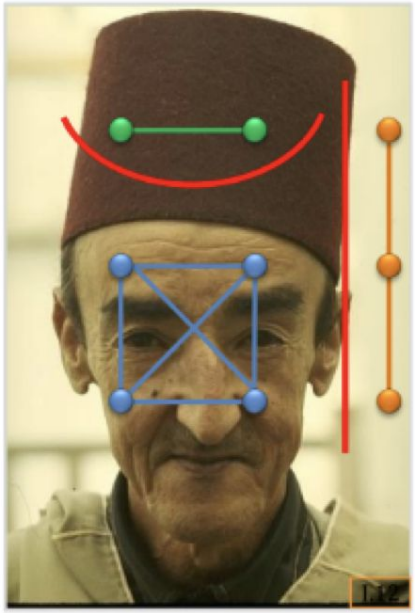
- A pair of vertices (pixels) **within** a subgraph have **high affinity**.
- A pair of vertices **from two different** subgraph have **low affinity**.

That is, **minimize the cost of cut**.

Also called **Min-Cut**.

Each subgraph is an image segment

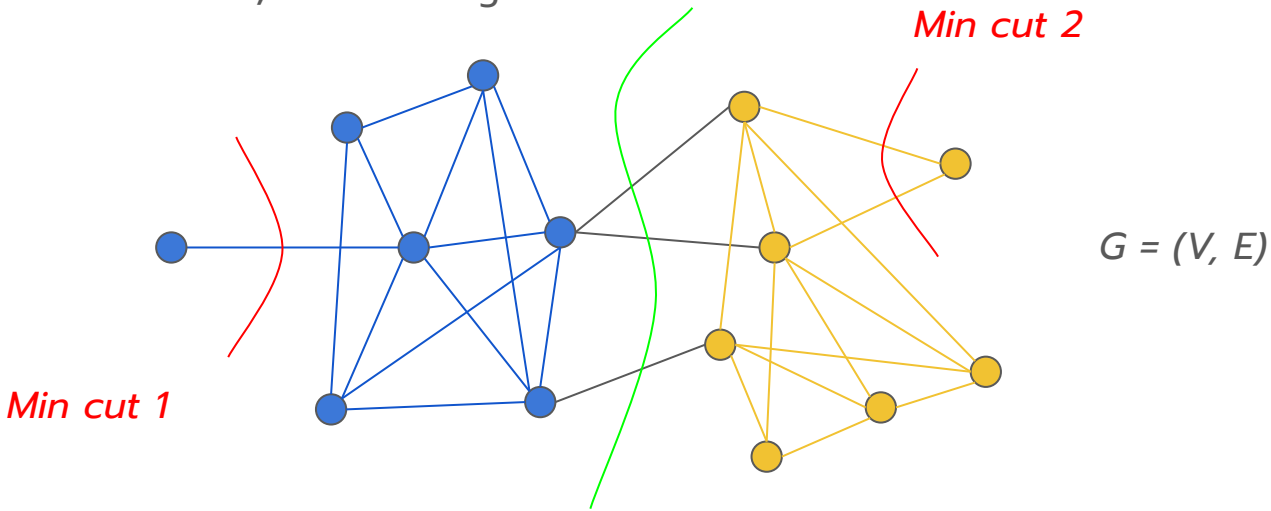




Graph Based Segmentation

Problem with Min-Cut

There is a bias to cut small, isolated segment.



$G = (V, E)$

Solution: **Normalize Cut** to favor larger subgraph.

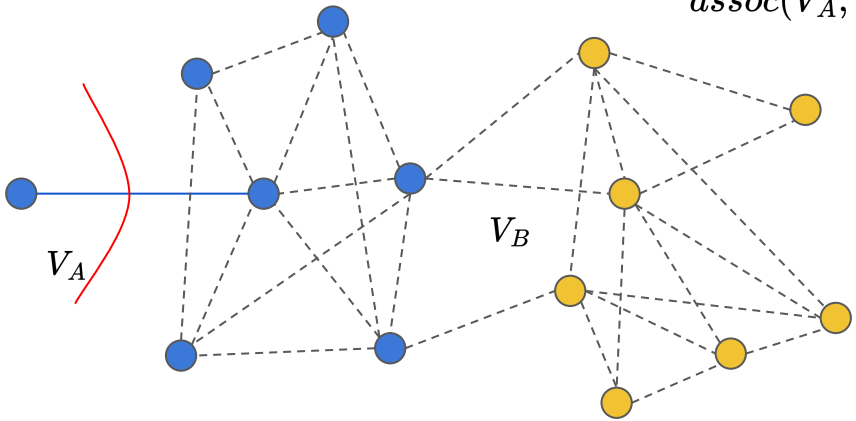
Graph Based Segmentation

Measure of Subgraph Size

within the subgraph find association

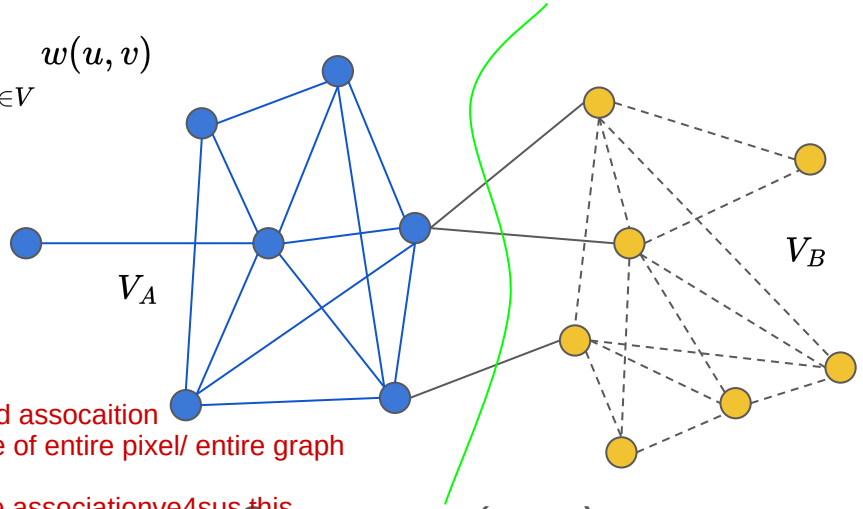
Compute how strongly vertices V_A are associated with vertices V

$$assoc(V_A, V) = \sum_{u \in V_A, v \in V} w(u, v)$$



each edege has w > Va is a subgraph

Weak assoc(V_A , V)



first find assocaition
v=node of entire pixel/ entire graph

find the associationve4sus this

Strong assoc(V_A , V)

weak assoc versus strong assoc

assoc ( ) is the sum of the weights of the solid edges.

Graph Based Segmentation

Normalize Cut (NCut)

Minimize Cost of Normalize Cut during Partition

$$NCut(V_A, V_B) = \frac{cut(V_A, V_B)}{assoc(V_A, V)} + \frac{cut(V_A, V_B)}{assoc(V_B, V)}$$

instead of just looking at the weight, find whether we have strong or weak association

Minimizing NCut has no known polynomial time solution.

It is **NP-Complete**.

cannot do in polynomial times
directly estimate it / approximate in certain times

if high association (a group of graph have) make entire equation smaller

Fast eigenvector-based approximation exist [Shi 2000]

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Graph Based Segmentation

Normalize Cut (NCut) : Results

Pixel Feature: {Brightness, Location}

Adopted from: Shree K. Nayar, Monograph FPCV-1-4, First Principles of Computer Vision, Columbia University, New York, Mar. 2022

Image Segmentation

Learning-based

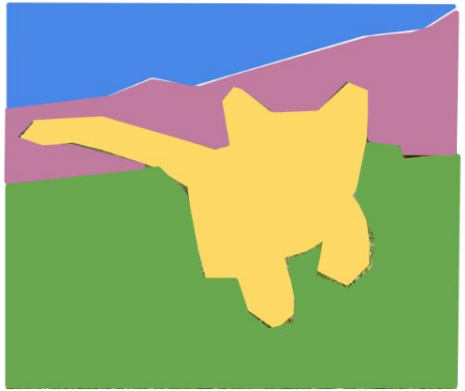
Computer Vision Tasks

Classification

CAT

No spatial extent

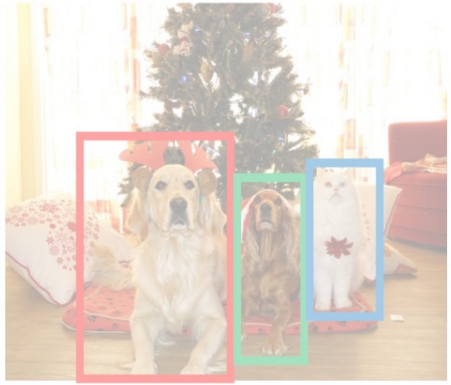
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



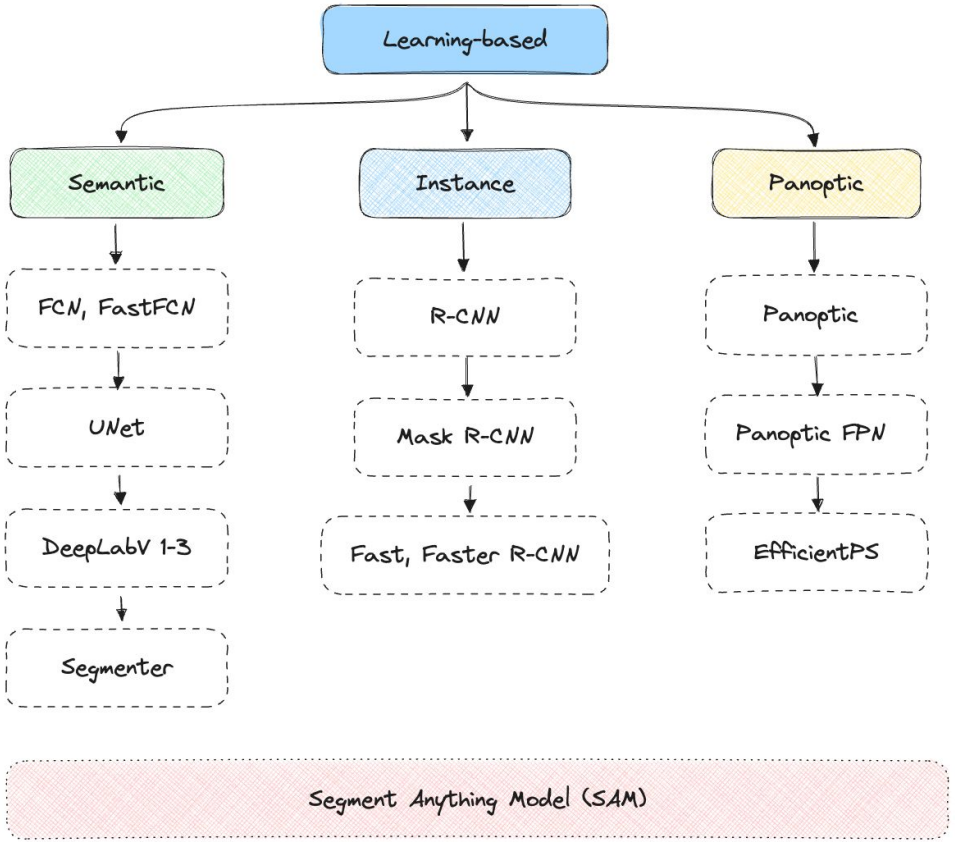
DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

Multiple objects

Learning-based Segmentation



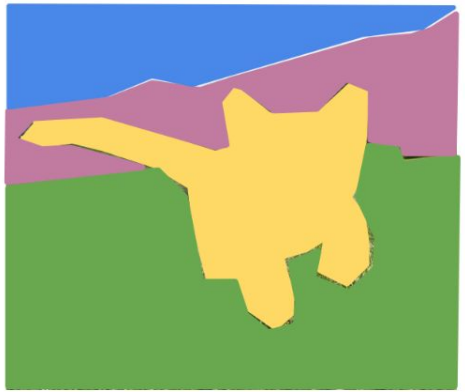
Computer Vision Tasks

Classification

CAT

No spatial extent

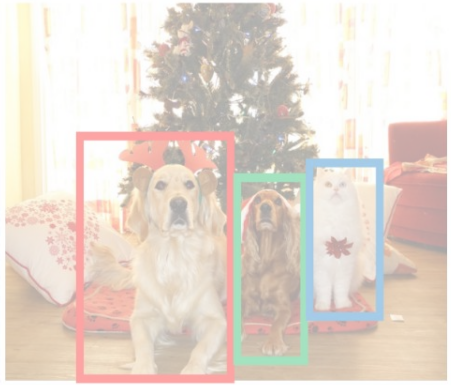
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

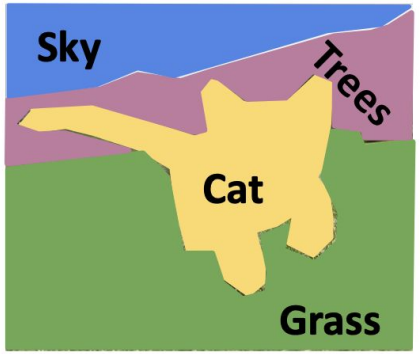
Multiple objects

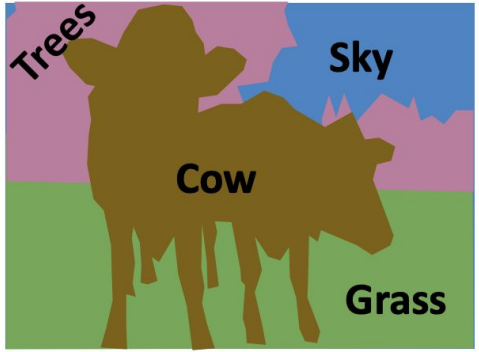
Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Label each pixel in the image with a category label.

Do not differentiate instances, **only care about pixels**.



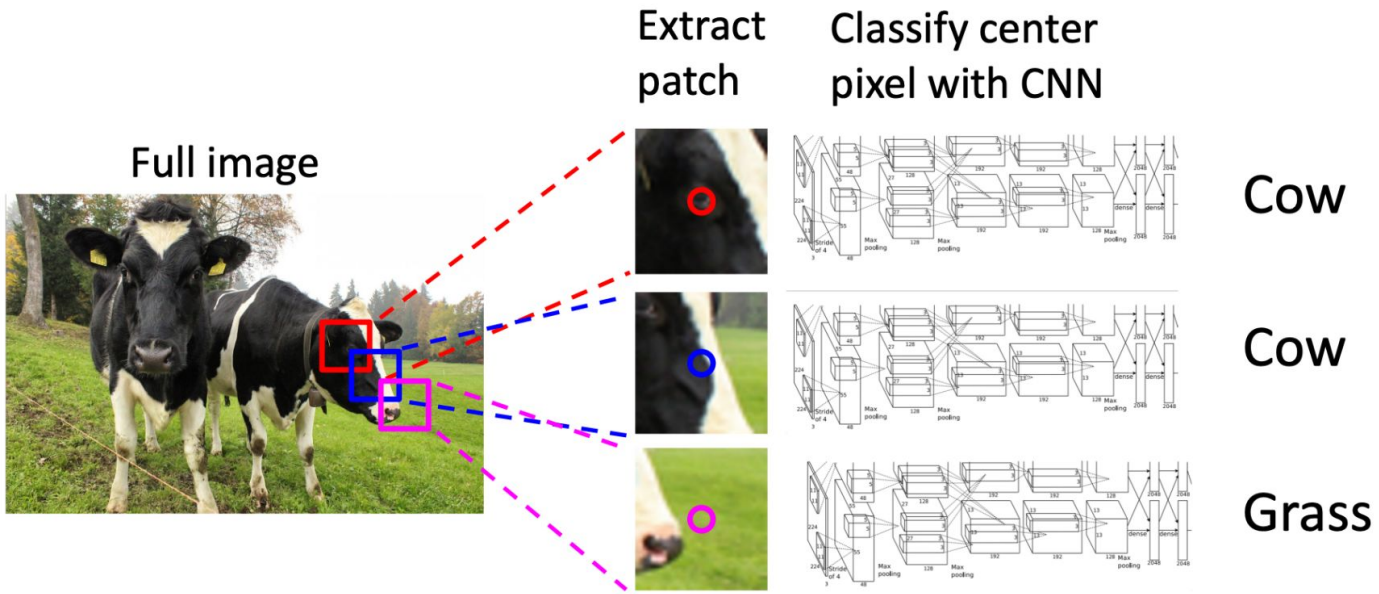


Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Sliding window

Problem: Very inefficient! Not reusing shared features between overlapping patches.



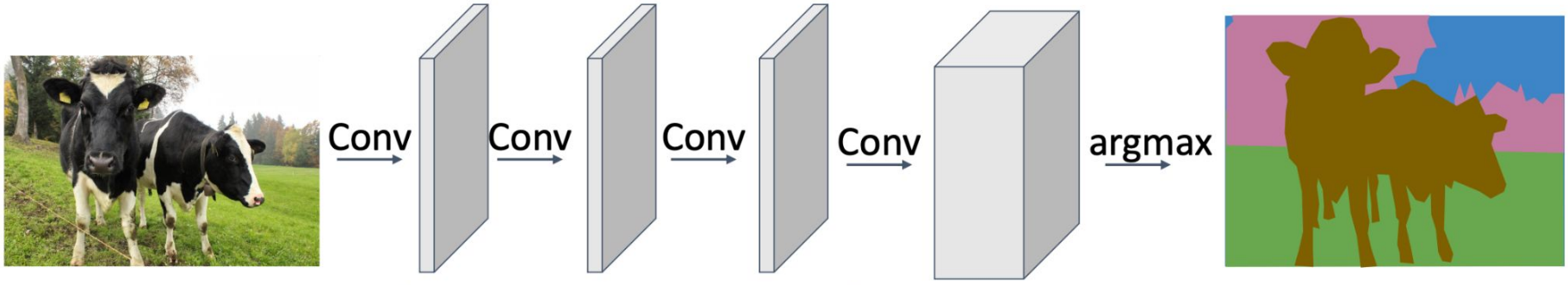
Farabet et al, "Learning Hierarchical Features for Scene Labeling," TPAMI 2013
Pinheiro and Collobert, "Recurrent Convolutional Neural Networks for Scene Labeling", ICML 2014

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Fully Convolutional Network

Design a network as a bunch of convolutional layers to make predictions for pixels all at once!



Loss function: Per-Pixel cross-entropy

Long et al, "Fully convolutional networks for semantic segmentation", CVPR 2015

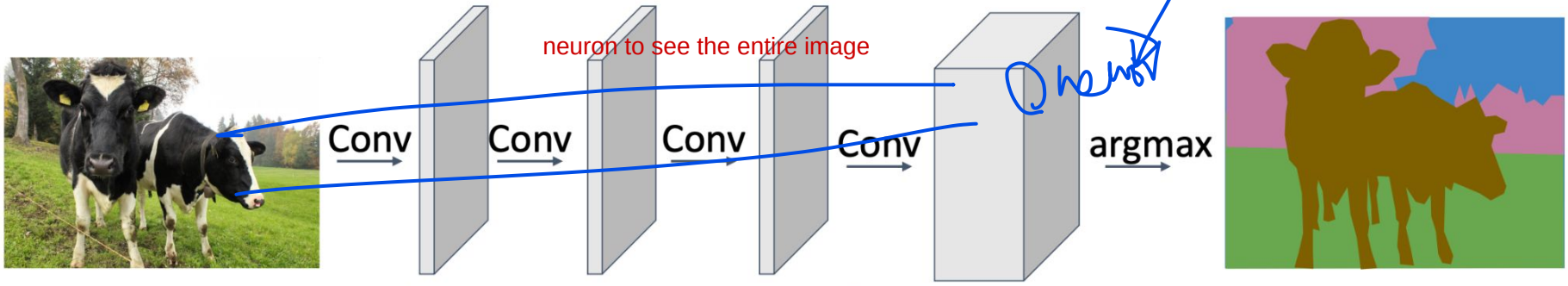
Semantic Segmentation

Fully Convolutional Network

Problem 2: : Effective receptive field size is linear in number of conv layers: With L 3x3 conv layers, receptive field is 1+2L, that is, we need many conv layers to get large receptive field size.

Design a network as a bunch of convolutional layers to make predictions for pixels all at once!

local area that one particular neuron can see : Receptive field



Input:
3 x H x W

Convolutions:
D x H x W

Scores:
C x H x W

Predictions:
H x W

Problem 1: convolutions at original image resolution will be very expensive.

doing downsampling > reduce the size of input

Loss function: Per-Pixel cross-entropy

Long et al, "Fully convolutional networks for semantic segmentation", CVPR 2015

Semantic Segmentation

Fully Convolutional Network

Design a network as a bunch of convolutional layers, with **downsampling** and **upsampling** inside the network!

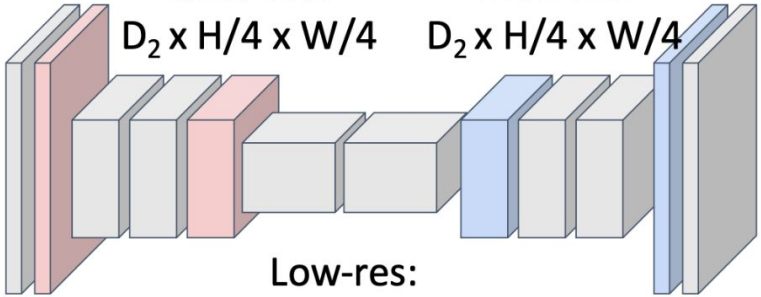
classification

Downsampling:
Pooling, strided convolution

Upsampling:
???

so that the resolutio betwen the input an doutput matches

Input:
3 x H x W



High-res:
D1 x H/2 x W/2

High-res:
D1 x H/2 x W/2

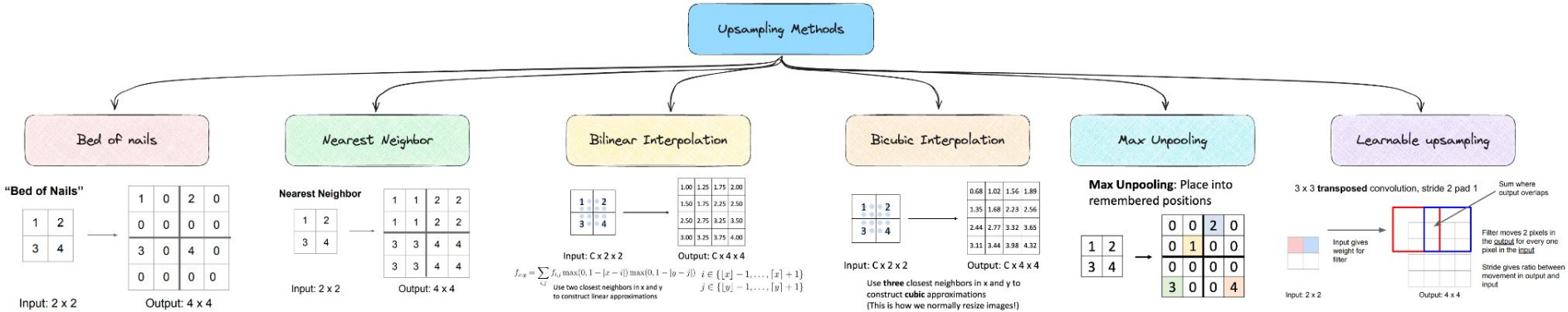
Predictions:
H x W

downsampling > colukution with/ w.o padding / stride

Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015
Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

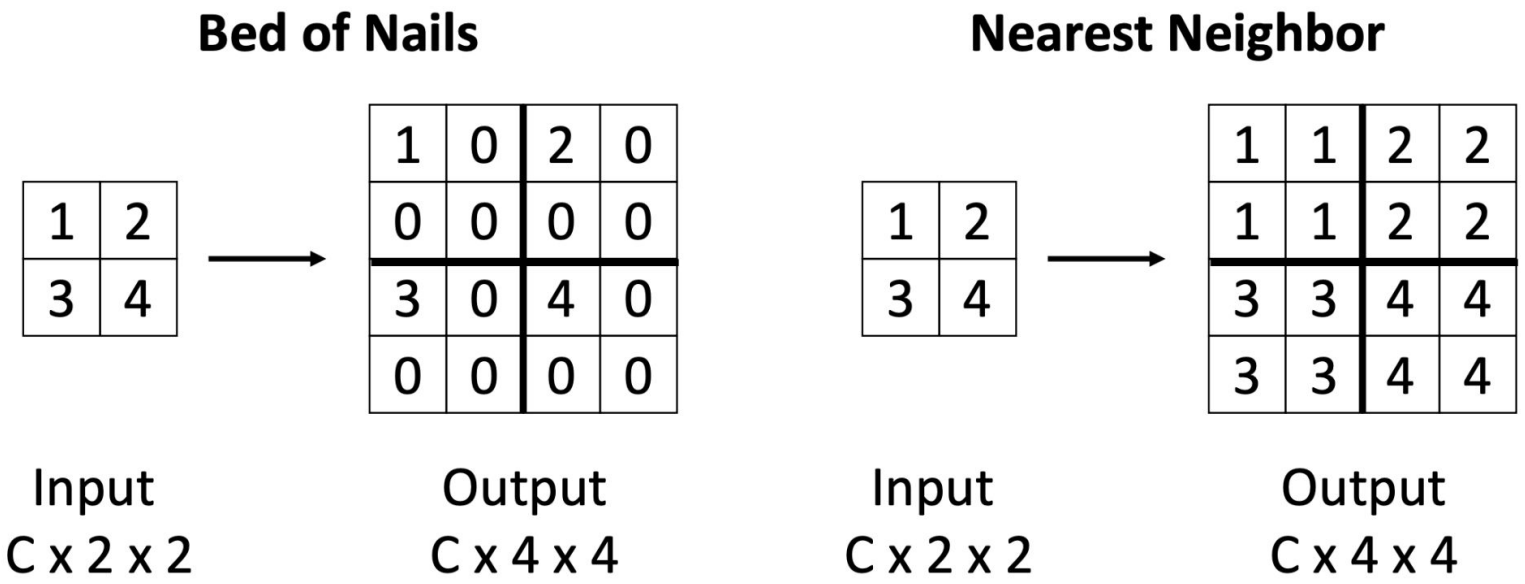
Semantic Segmentation

Upsampling Methods



Semantic Segmentation

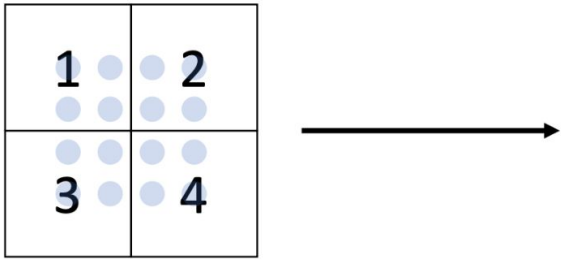
Unpooling



Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Bilinear Interpolation



1.00	1.25	1.75	2.00
1.50	1.75	2.25	2.50
2.50	2.75	3.25	3.50
3.00	3.25	3.75	4.00

Input: C x 2 x 2

Output: C x 4 x 4

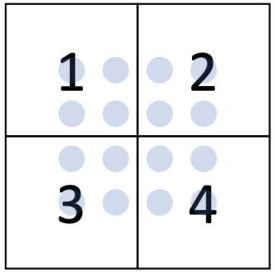
$$f_{x,y} = \sum_{i,j} f_{i,j} \max(0, 1 - |x - i|) \max(0, 1 - |y - j|) \quad i \in \{\lfloor x \rfloor - 1, \ldots, \lceil x \rceil + 1\} \quad j \in \{\lfloor y \rfloor - 1, \ldots, \lceil y \rceil + 1\}$$

Use two closest neighbors in x and y to construct linear approximations

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Bicubic Interpolation



0.68	1.02	1.56	1.89
1.35	1.68	2.23	2.56
2.44	2.77	3.32	3.65
3.11	3.44	3.98	4.32

Input: C x 2 x 2

Output: C x 4 x 4

Use **three** closest neighbors in x and y to construct cubic approximations
(This is how we normally resize images!)

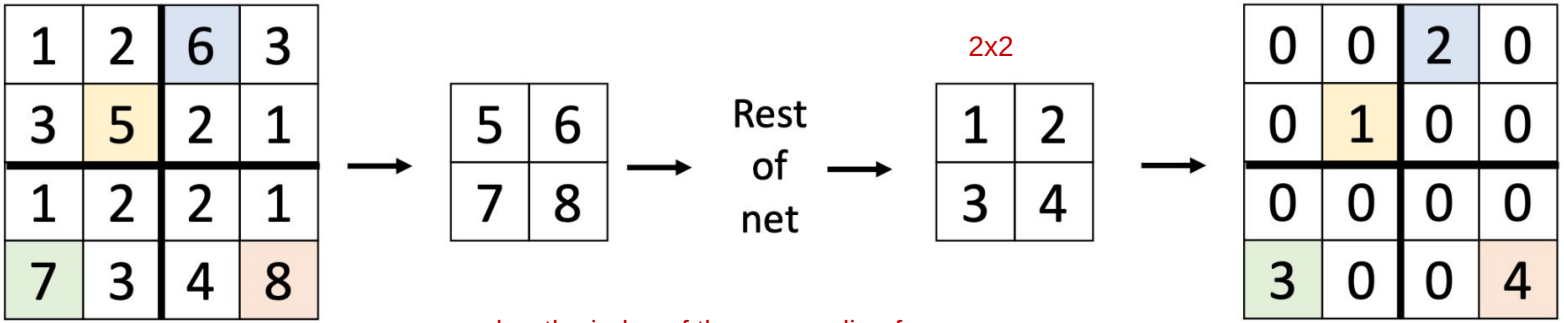
Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Max-pooling

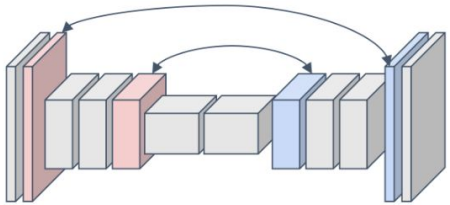
Max Pooling: Remember which position had the max

Max Unpooling: Place into remembered positions



remember the index of the maxpooling from

4x 4



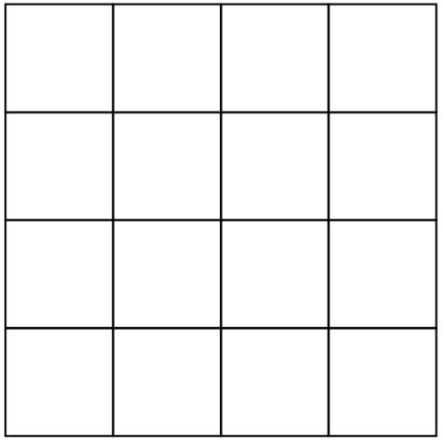
Pair each downsampling layer with an upsampling layer

Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

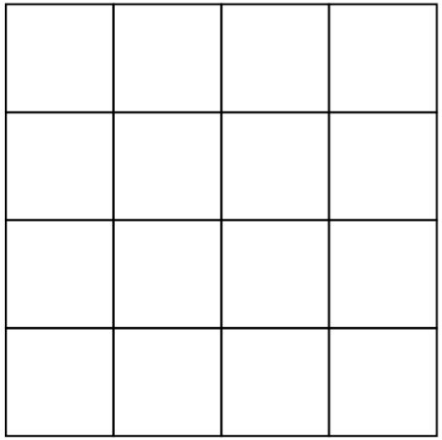
Semantic Segmentation

Transposed Convolution

Recall: Normal 3 x 3 convolution, stride 1, pad 1



Input: 4 x 4



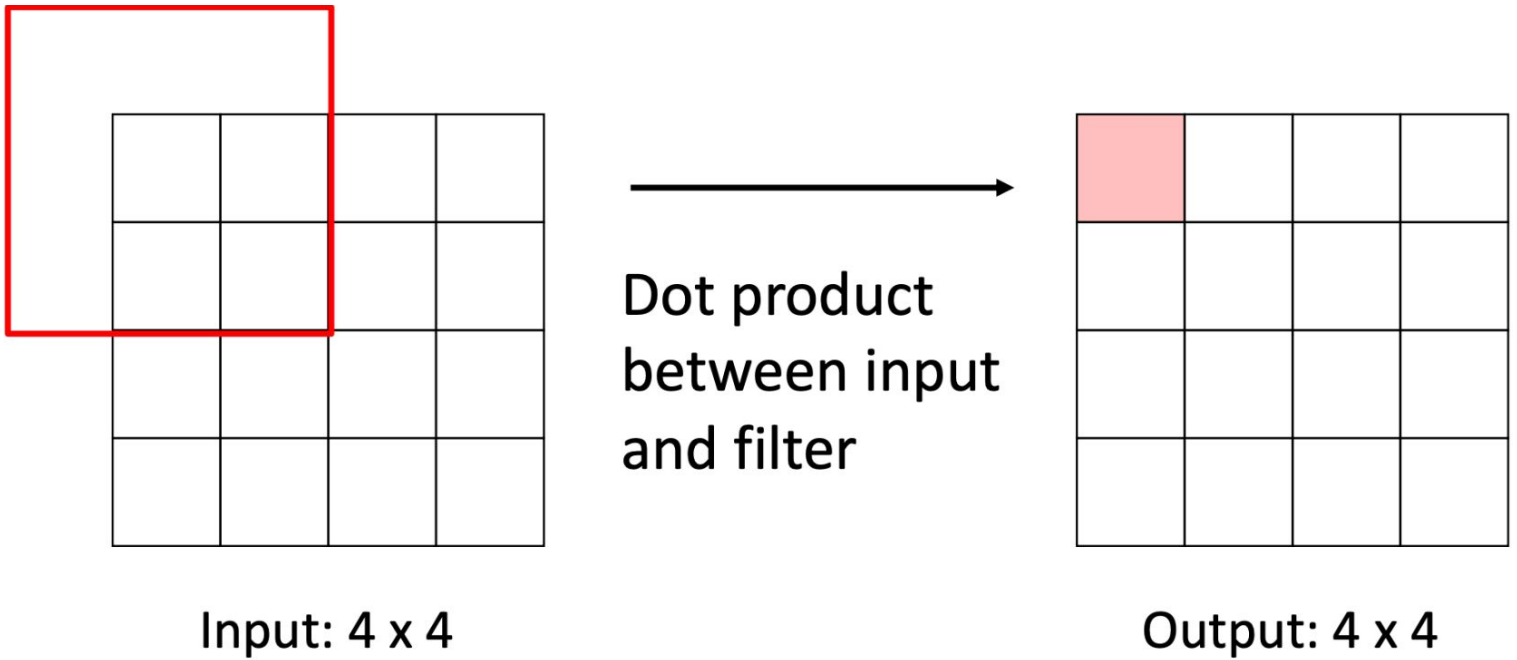
Output: 4 x 4

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Transposed Convolution

Recall: Normal 3 x 3 convolution, stride 1, pad 1

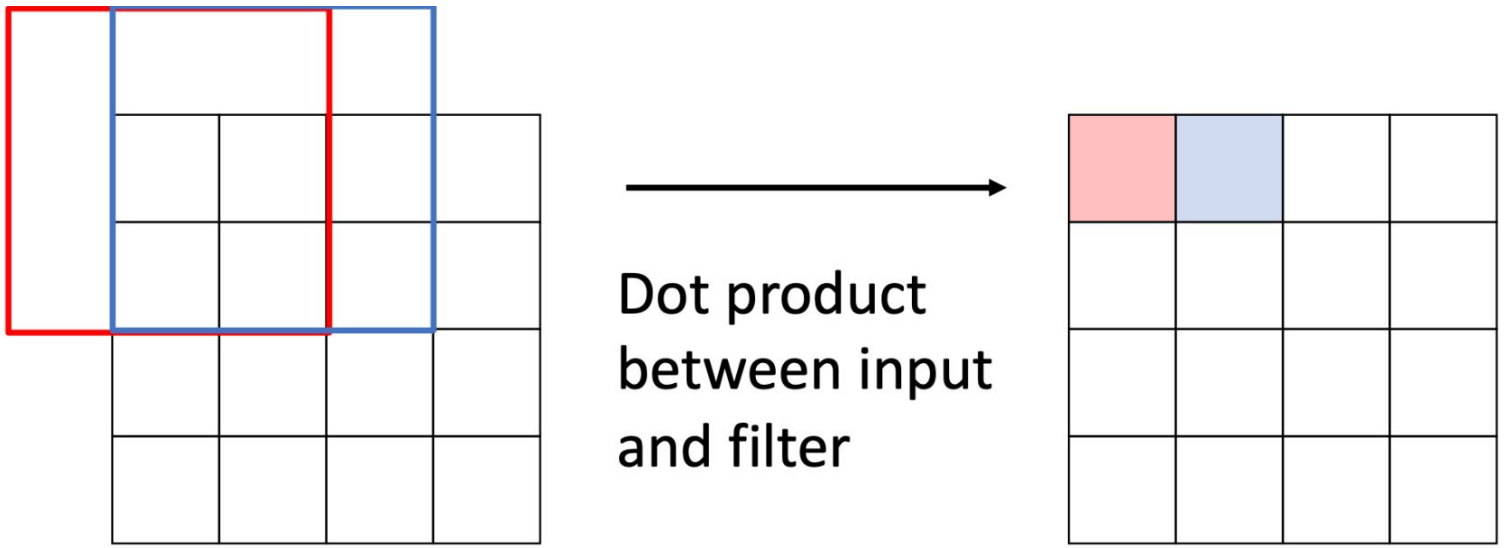


Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Transposed Convolution

Recall: Normal 3 x 3 convolution, stride 1, pad 1

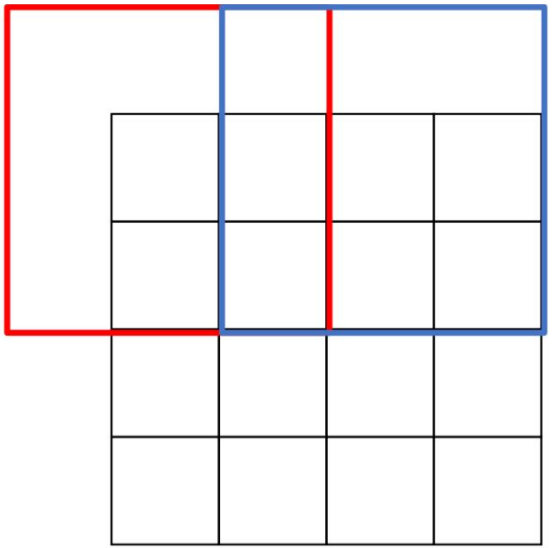


Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

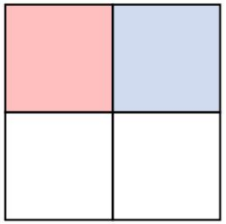
Transposed Convolution

Recall: Normal 3 x 3 convolution, **stride 2**, pad 1



Convolution with stride > 1 is “Learnable Downsampling”
Can we use stride < 1 for “Learnable Upsampling”?

Dot product between input and filter



Input: 4 x 4

Output: 2 x 2

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

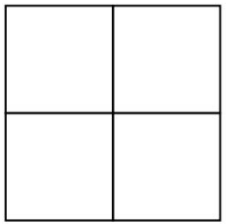
Semantic Segmentation

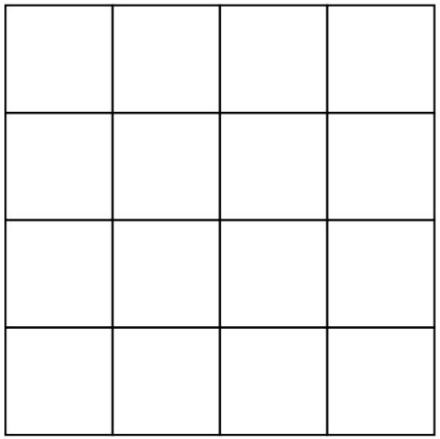
Transposed Convolution

opposite of convulution

3 x 3 **convolution transpose,** stride 2

unpooling upsampling





Input: 2 x 2

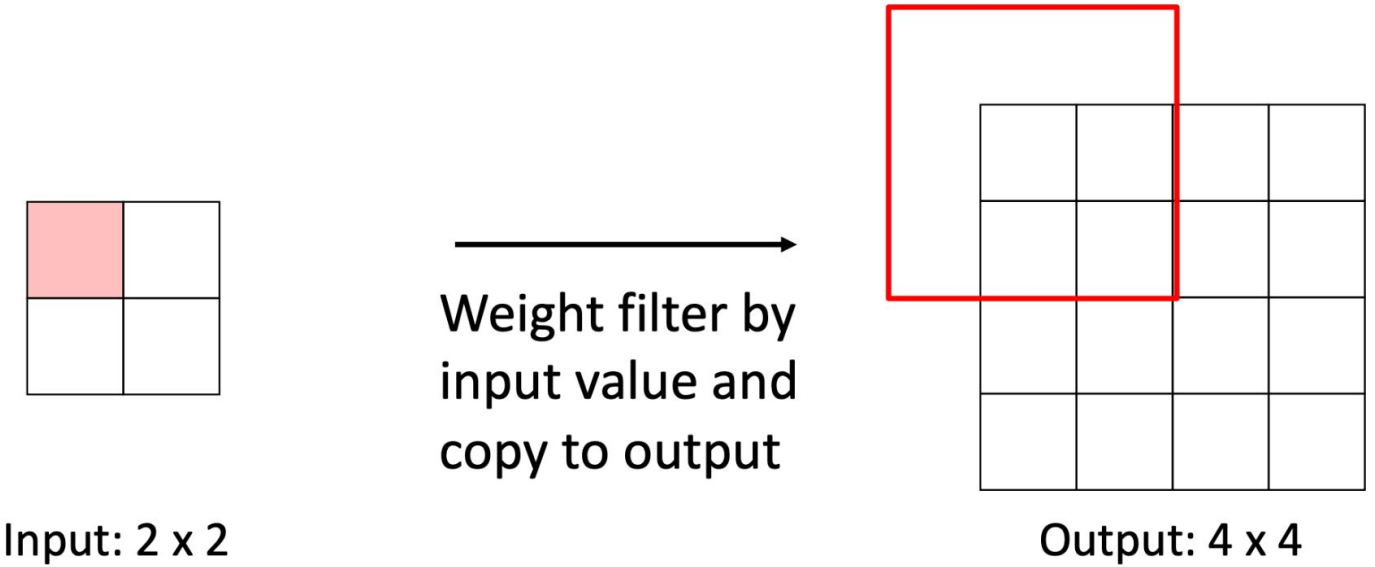
Output: 4 x 4

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Transposed Convolution

3 x 3 **convolution transpose,** stride 2



Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

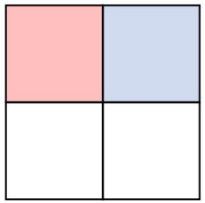
Semantic Segmentation

Transposed Convolution

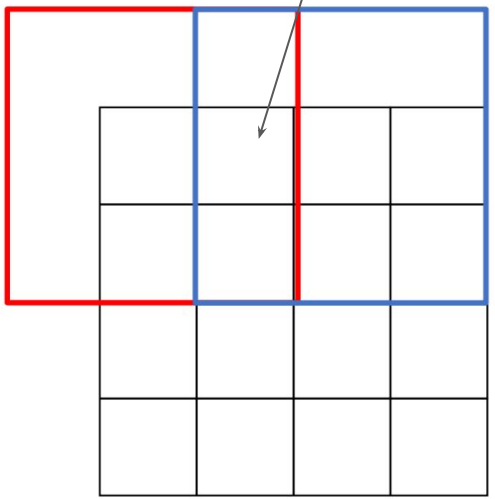
3 x 3 **convolution transpose,** stride 2

Sum where output overlaps

Filter moves 2 pixels in output for every 1 pixel in input



Weight filter by input value and copy to output



Input: 2 x 2

Output: 4 x 4

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

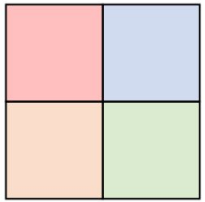
Semantic Segmentation

Transposed Convolution

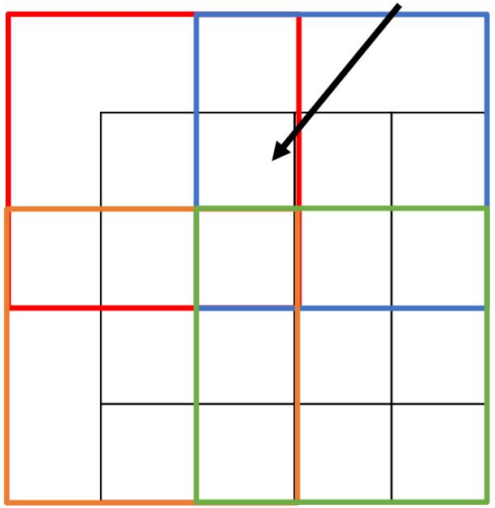
3 x 3 **convolution transpose,** stride 2

Sum where output overlaps

This gives 5x5 output – need to trim one pixel from top and left to give 4x4 output



Weight filter by input value and copy to output

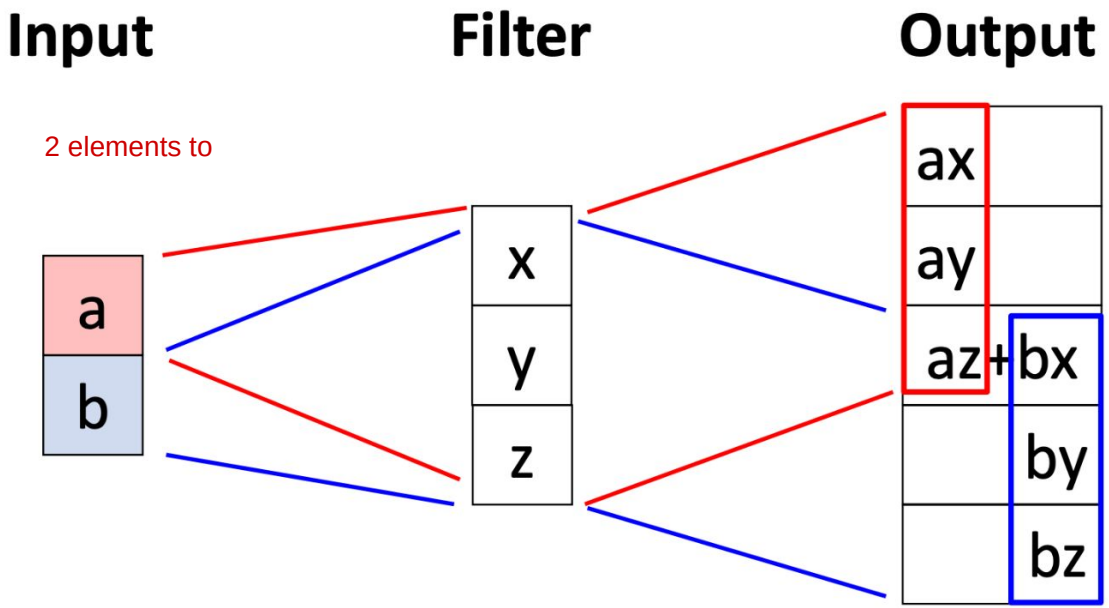


Input: 2 x 2

Output: 4 x 4

Semantic Segmentation

Transposed Convolution: 1D Example



Output has copies of filter weighted by input

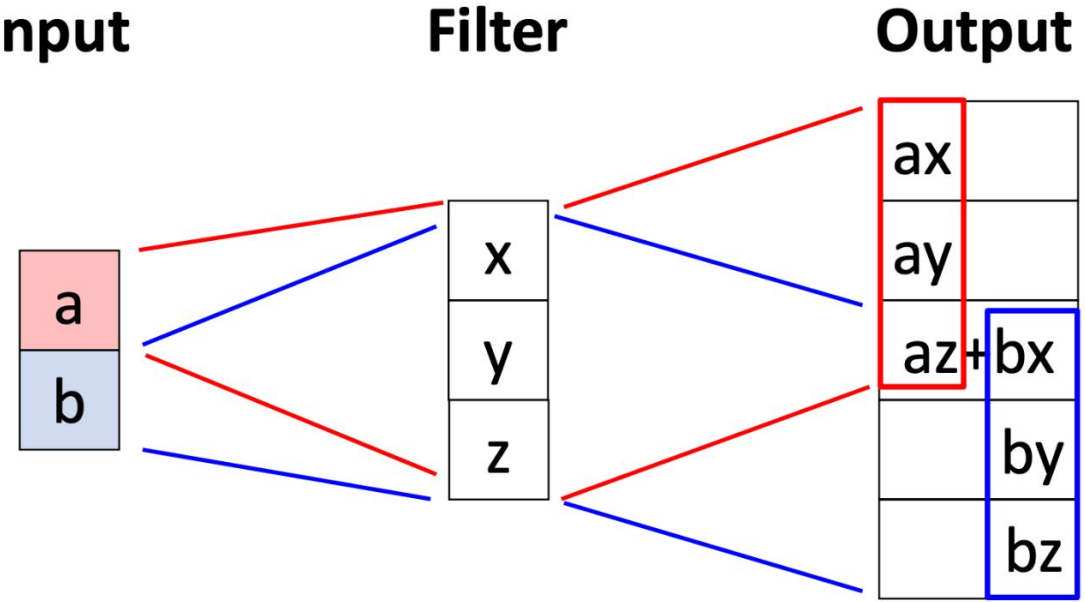
Stride 2: Move 2 pixels output for each pixel in input

Sum at overlaps

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

Transposed Convolution: 1D Example



This has many names:

- Deconvolution (bad)!
- Upconvolution
- Fractionally strided convolution
- Backward strided convolution
- Transposed Convolution (best name)

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

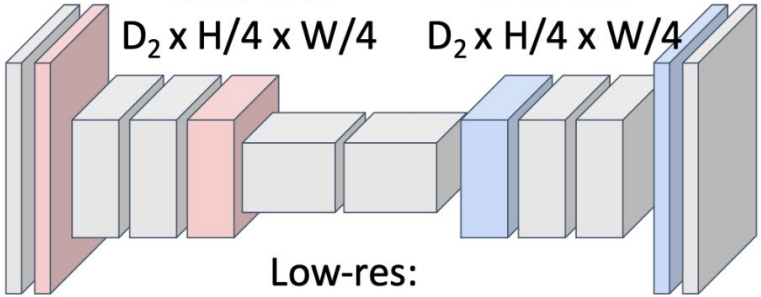
Fully Convolutional Network

Design a network as a bunch of convolutional layers, with **downsampling** and **upsampling** inside the network!

Downsampling:
Pooling, strided convolution

Upsampling:
Interpolation, transposed conv.

Input:
3 x H x W



Predictions:
H x W

Long, Shelhamer, and Darrell, "Fully Convolutional Networks for Semantic Segmentation", CVPR 2015
Noh et al, "Learning Deconvolution Network for Semantic Segmentation", ICCV 2015

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Semantic Segmentation

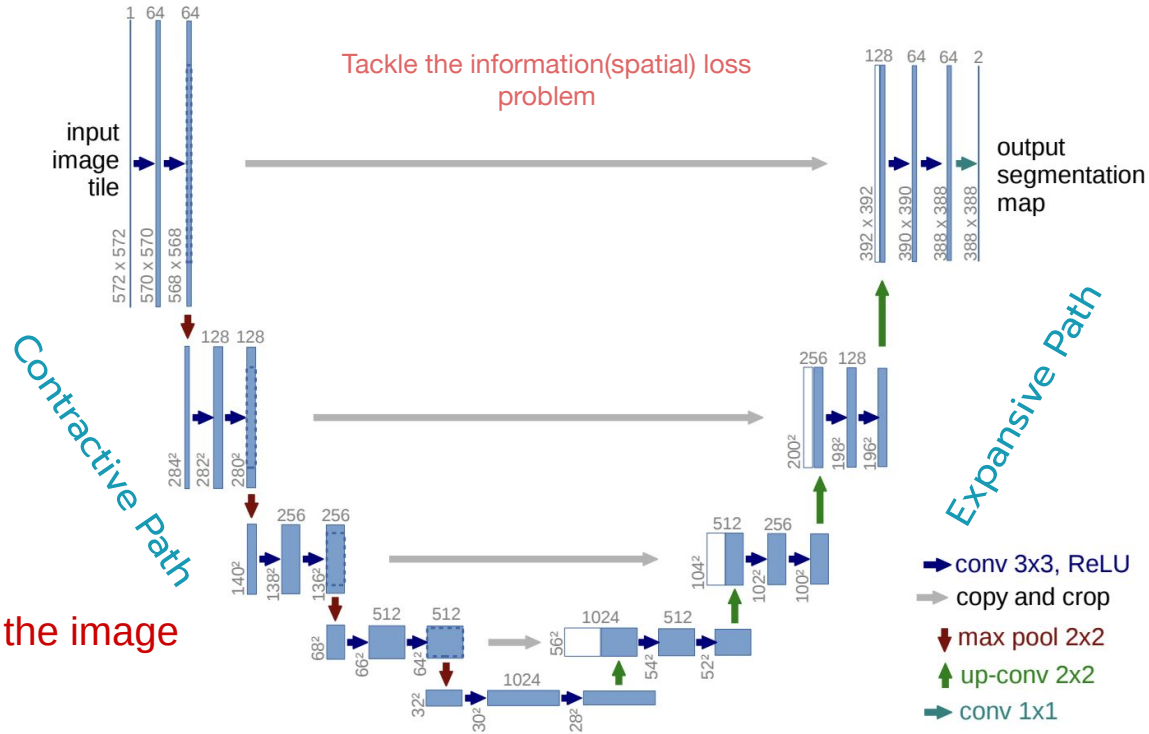
UNet

U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger, Philipp Fischer, and Thomas Brox

Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany

Medical Image Segmentation



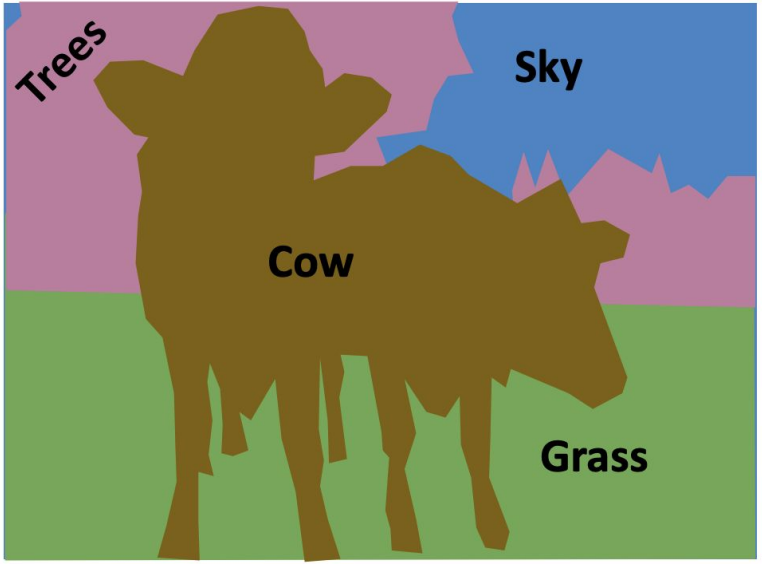
Pixel-wise Cross Entropy loss

what is inside the image

Computer Vision Tasks

Object Detection: Detects individual object instances, but only gives box

Semantic Segmentation: Gives per-pixel labels, but merges instances



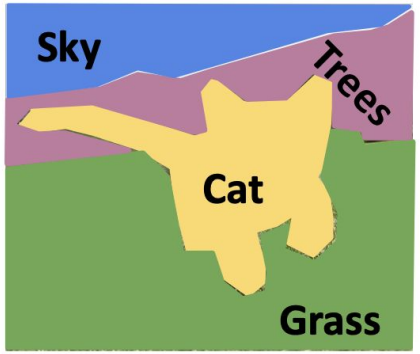
Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

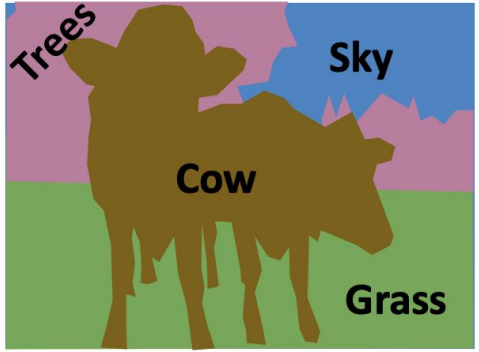
Computer Vision Tasks

Object Categories

Things: Object categories that can be separated into object instances (e.g. cats, cars, person)

Stuffs: Object categories that cannot be separated into instances (e.g. sky, grass, water, trees)



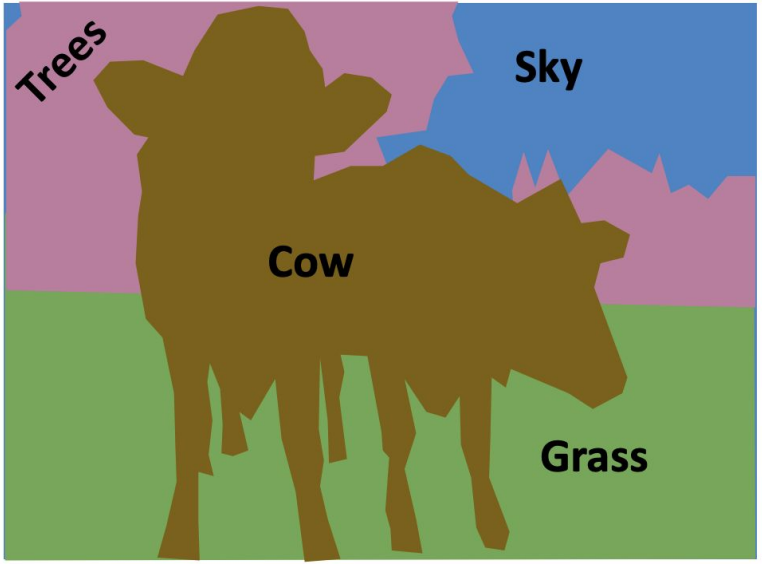


Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Computer Vision Tasks

Object Detection: Detects individual object instances, but only gives box **(Only things!)**

Semantic Segmentation: Gives per-pixel labels, but merges instances **(Both things and Stuff)**



Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

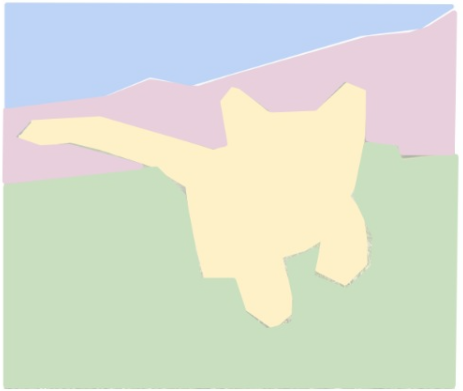
Computer Vision Tasks

Classification

CAT

No spatial extent

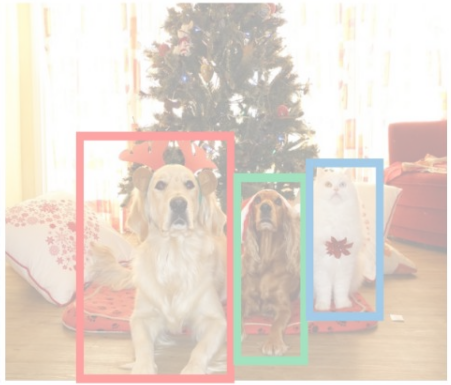
Semantic Segmentation



GRASS, CAT, TREE, SKY

No objects, just pixels

Object Detection



DOG, DOG, CAT

Instance Segmentation

DOG, DOG, CAT

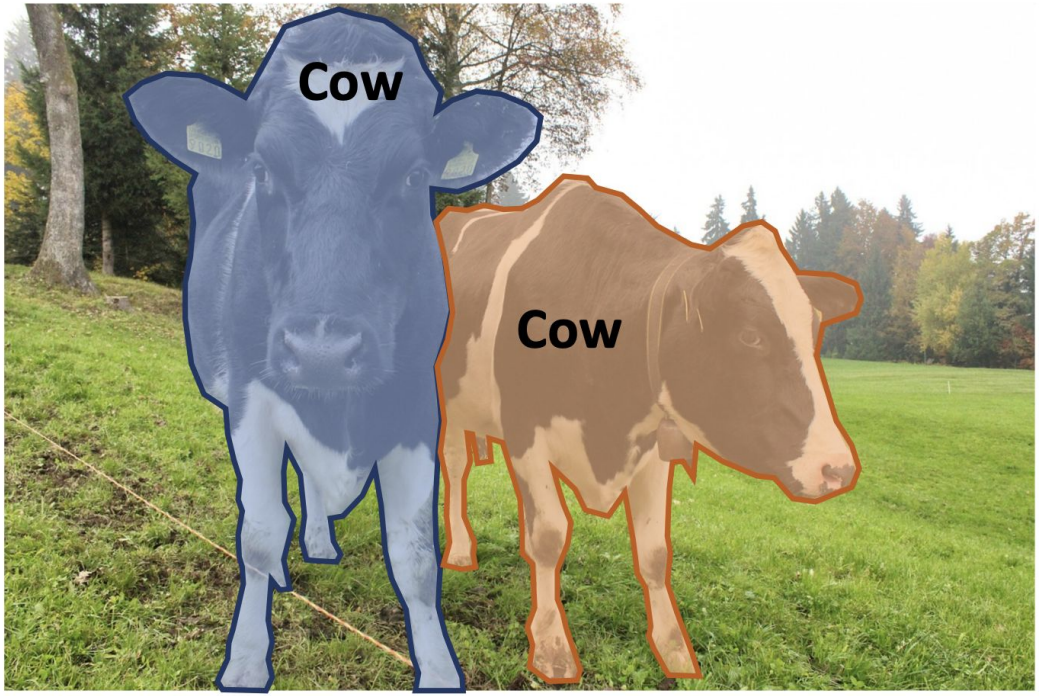
Multiple objects

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Computer Vision Tasks

Instance Segmentation

Instance Segmentation:
Detect all objects in the image, and identify the pixels that belong to each object **(Only things!)**

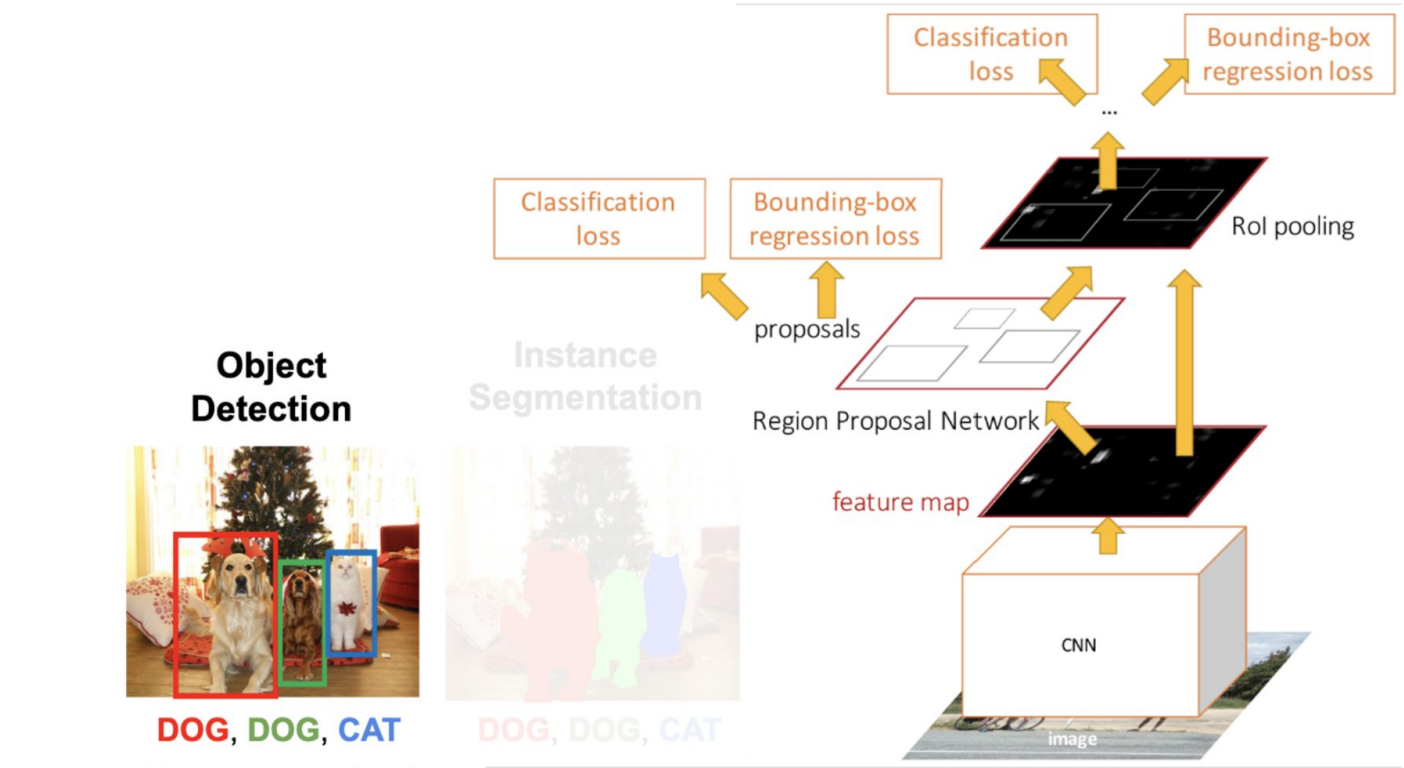


Approach: Perform object detection, then predict a segmentation mask for each object!

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Instance Segmentation

Recall Faster R-CNN for Object Detection

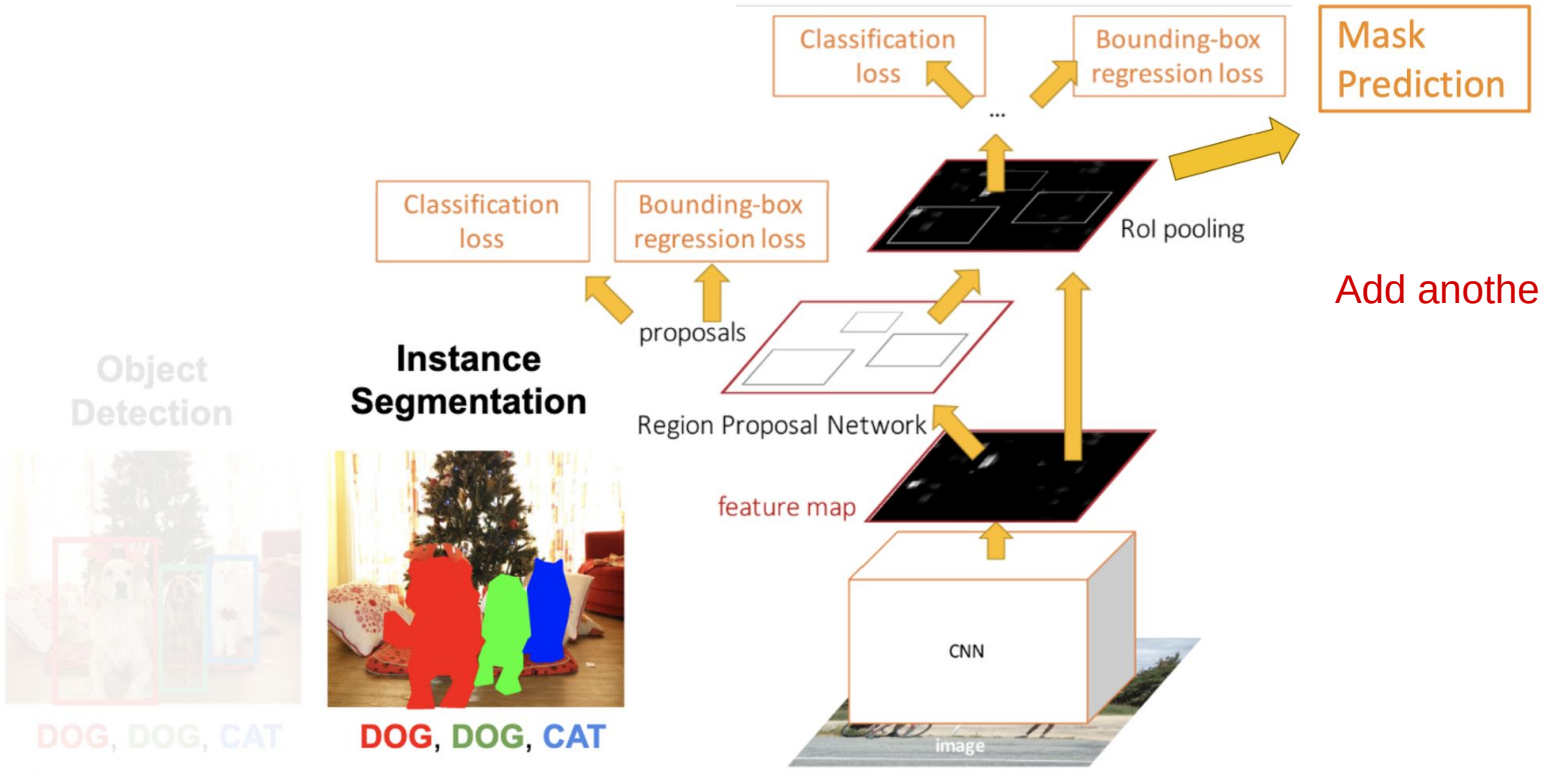


Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NeurIPS 2015

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Instance Segmentation

Mask R-CNN



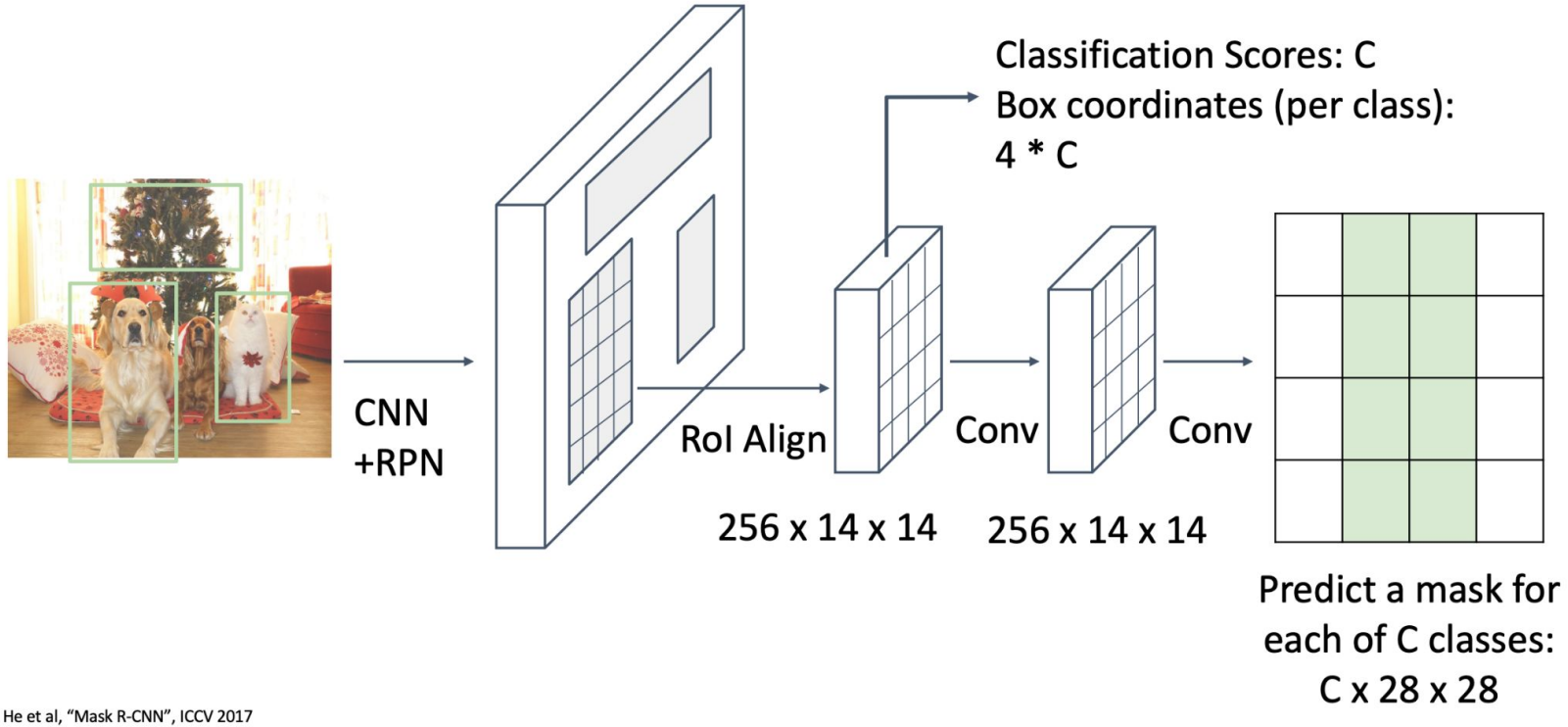
Add another head

He et al, "Mask R-CNN", ICCV 2017

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Instance Segmentation

Mask R-CNN



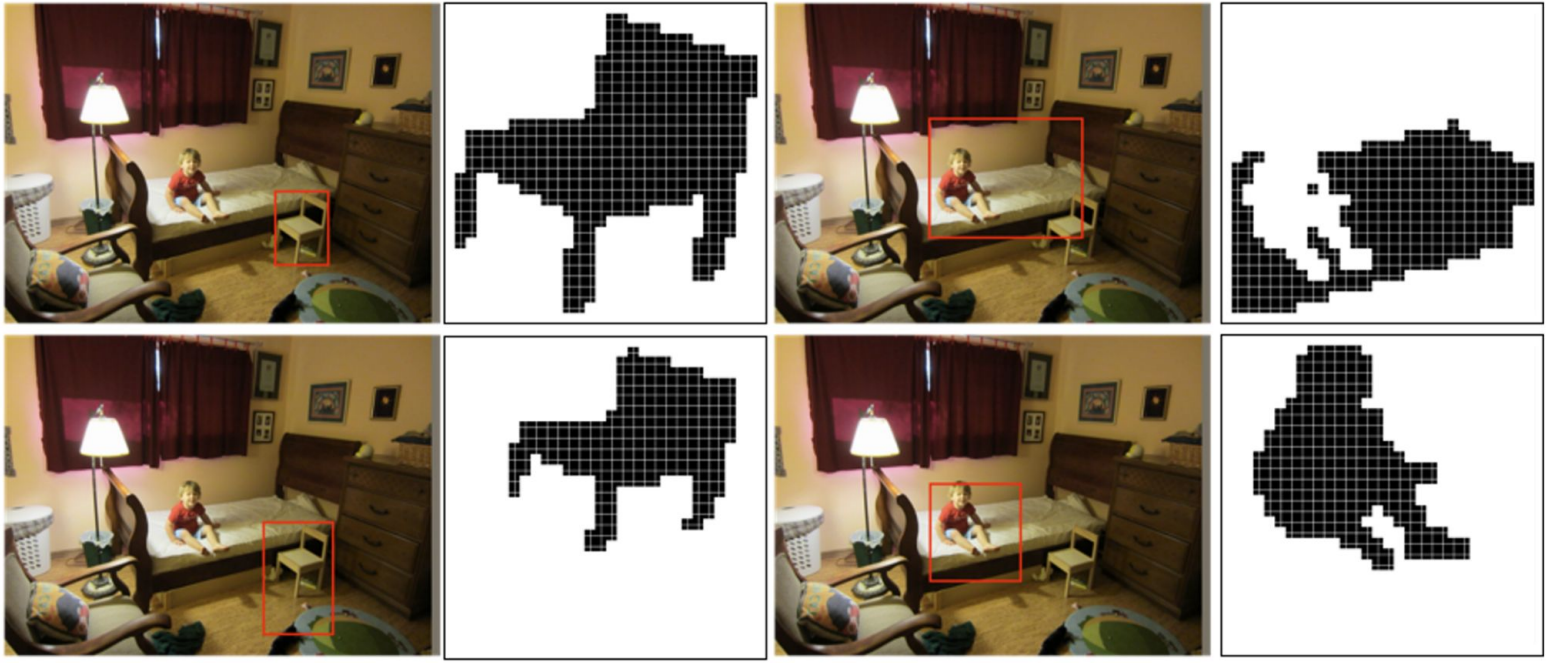
He et al, "Mask R-CNN", ICCV 2017

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Instance Segmentation

Mask R-CNN: Example Training Targets

Mask Prediction head of the model



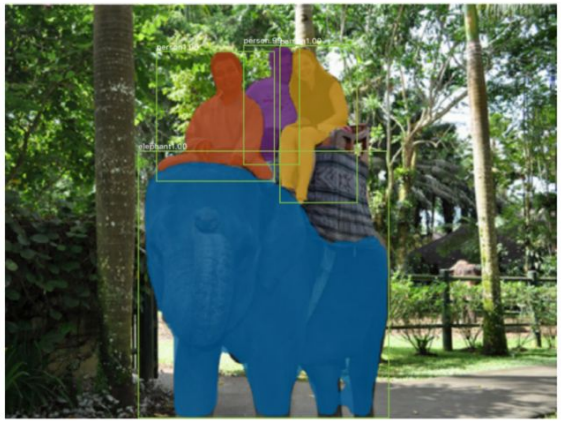
Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

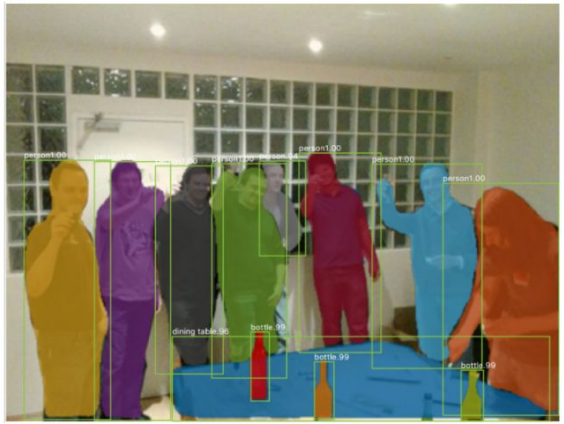
Instance Segmentation

Mask R-CNN: Results

number of objects

number of classes



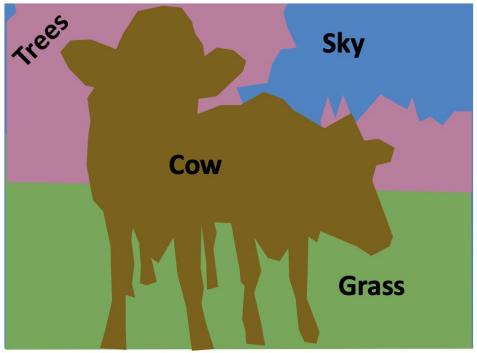


Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

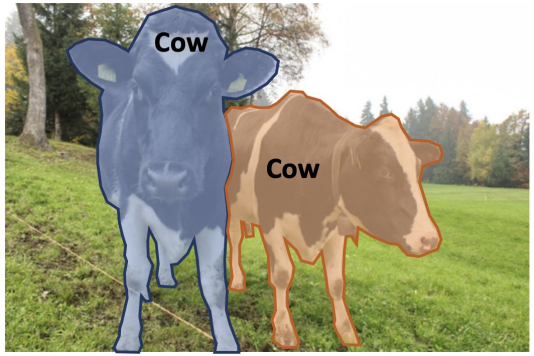
Panoptic Segmentation

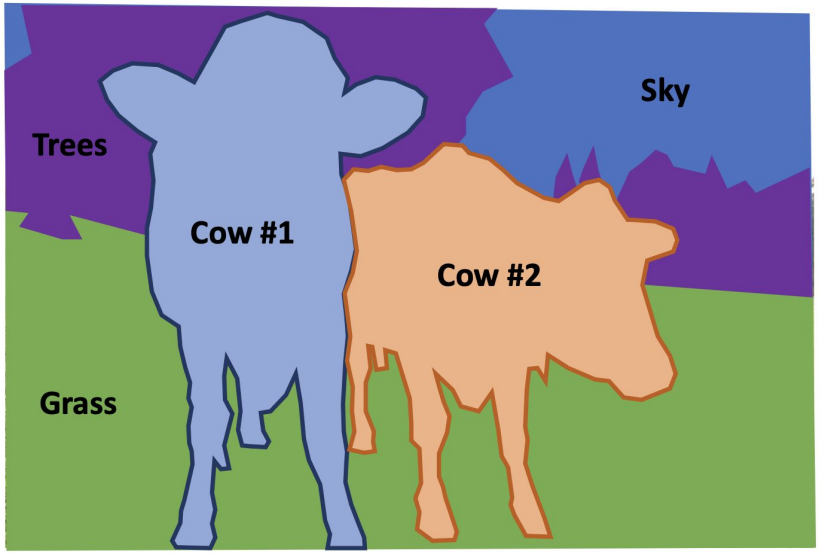
Panoptic Segmentation: : Label all pixels in the image (both things and stuff). For "thing" categories also separate into instances.

Semantic Segmentation: : Identify both things and stuff, but doesn't separate instances



Instance Segmentation: : Separate object instances, but only things





Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/

Panoptic Segmentation

Kirillov et al, "Panoptic Feature Pyramid Networks", CVPR 2019

Slide adopted from: Justin Johnson, EECS 498-007 / 598-005: Deep Learning for Computer Vision, University of Michigan, https://web.eecs.umich.edu/~justincj/teaching/eecs498/WI2022/