Image Processing and Analysis

Semantic image segmentation using deep learning techniques

Oge Marques, PhD
Professor
College of Engineering and Computer Science
Florida Atlantic University

Outline

1. Motivation
2. Terminology
3. Figures of merit
4. Segmentation techniques (before deep learning)
5. Segmentation-related apps in MATLAB
6. Datasets, benchmarks, challenges
7. Semantic image segmentation using deep learning in MATLAB
8. Hands-on assignment

What will we learn?

- What is image segmentation…?
- Where is it used?
- What are the main types of image segmentation techniques?
- How can image segmentation algorithms be evaluated / benchmarked?
- What are the main deep learning architectures used for image segmentation today?
- How can I segment images using MATLAB?

Motivation

Why segment?

Introduction

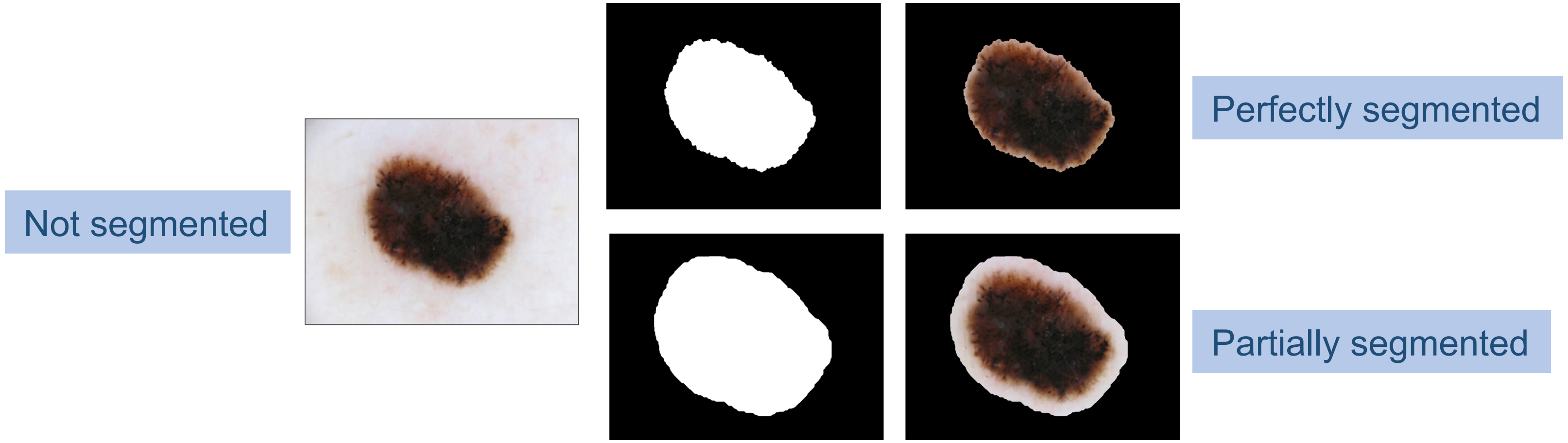
- Segmentation: the process of partitioning an image into a set of non-overlapping regions whose union is the entire image.
  - These regions should ideally correspond to objects and their meaningful parts, and background.

- Most image segmentation algorithms are based on one of two basic properties that can be extracted from pixel values -- discontinuity and similarity -- or a combination of them.

- Segmentation is one of the most crucial tasks in image processing and computer vision.
  - And it's still an open research problem.

Motivation

- Objects and (back)ground
  - There are many computer (and human) vision tasks where the focus is on the main object(s) in the scene, regardless of the background.
  - In these cases, we must segment ("extract") the object from the background before performing other tasks (such as measuring the objects properties or classifying it).
- Example: skin lesion classification



[Source:International Skin Imaging Collaboration Archive]

J. Burdick, **O. Marques**, J. Weinthal, and B. Furht, "Rethinking Skin Lesion Segmentation in a Convolutional Classifier", *Journal of Digital Imaging* (2017) https://doi.org/10.1007/s10278-017-0026-y

Why Image Segmentation Matters

- Several algorithms and techniques for image segmentation have been developed over the years using *domain-specific knowledge* to effectively solve segmentation problems in that *specific application area*.

- These applications include medical imaging, automated driving, video surveillance, machine vision, and many others.

Source: https://www.mathworks.com/discovery/image-segmentation.html

Terminology

Terminology

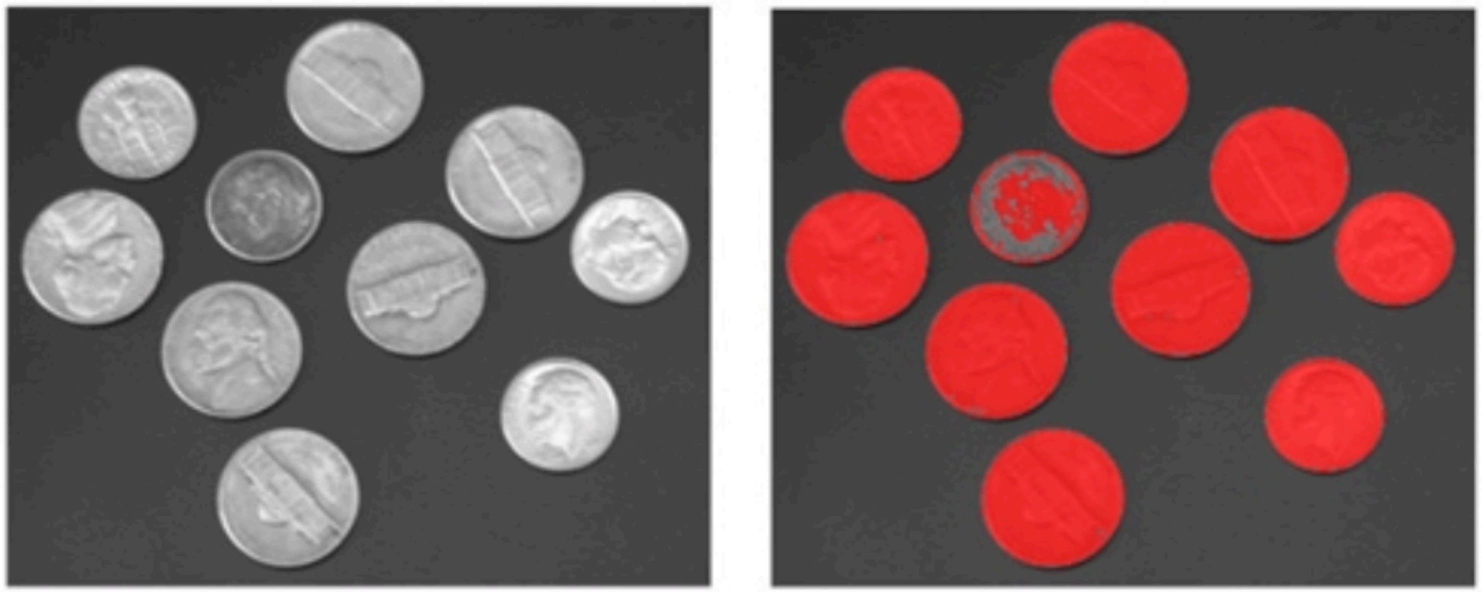
1. **Image segmentation**:
   process of automatically labeling each pixel in an image as belonging to *foreground* or *background*.

2. **Semantic segmentation**:
   connected labeled pixels map to object classes.

3. **Instance segmentation**:
   connected labeled pixels correspond to individual instances of a class.

Terminology

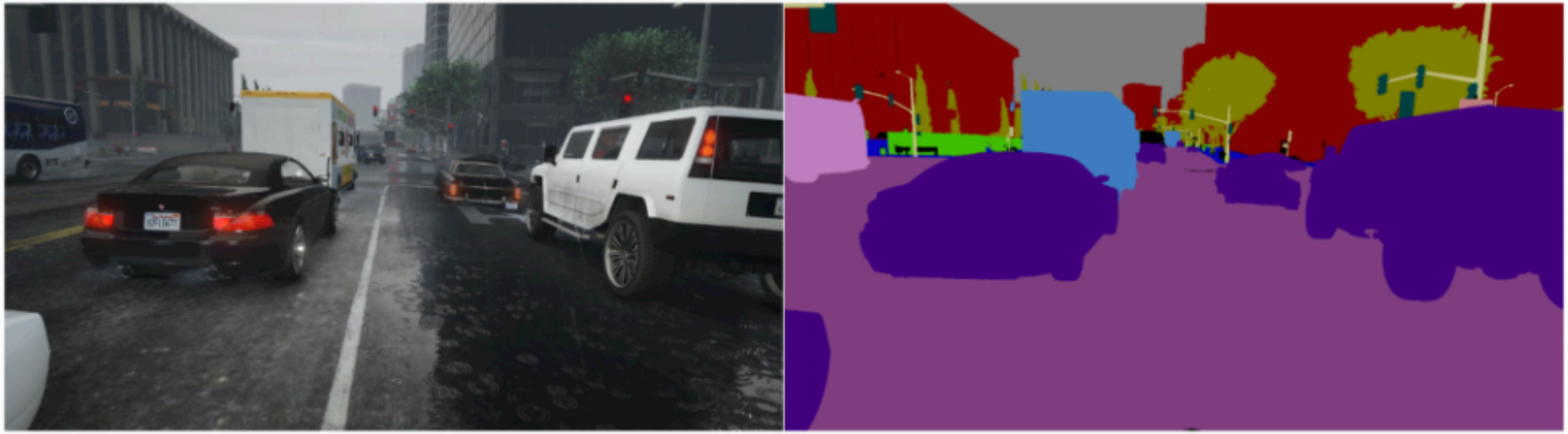
Image segmentation: *foreground* (color-coded in red) vs. *background*.



Source: https://www.mathworks.com/discovery/image-segmentation.html

Terminology

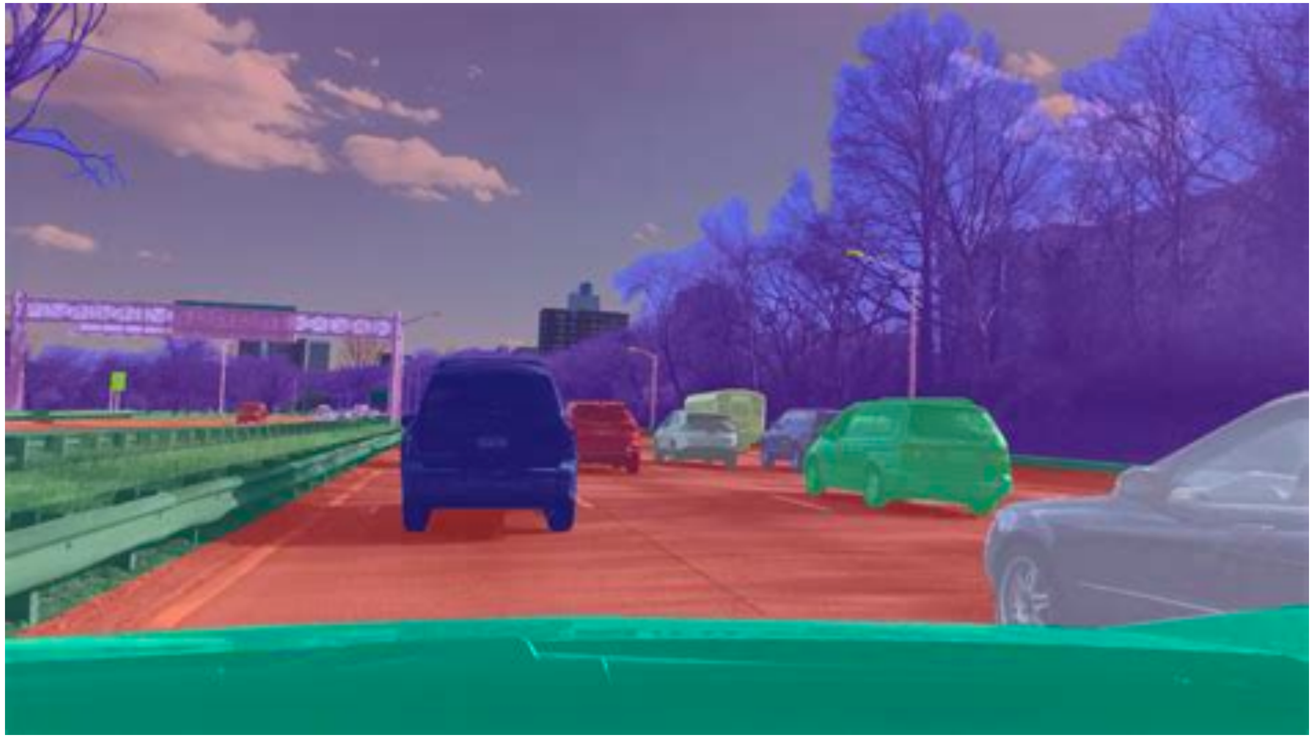
Semantic segmentation: different *classes* are labeled differently (car=purple, truck=light blue, bus = pink, etc.).



Source: https://deepdrive.pl/datasets/

Terminology

Instance segmentation: individual *instances* of a class (in this case, *car*) are labeled differently.



Source: https://deepdrive.pl/bdd100k-dataset/

Figures of merit

Evaluating Segmentation Accuracy

Steps:

1. Choose figure of merit
   - Jaccard
   - Dice
   - BF score
2. Select ground truth (mask)
3. Compute figure of merit (see MATLAB functions on next slide)
   - A value of 1 is ideal (perfect match, pixel by pixel)

Evaluating segmentation accuracy

jaccard	Jaccard similarity coefficient https://www.mathworks.com/help/images/ref/jaccard.html
dice	Sørensen-Dice similarity coefficient https://www.mathworks.com/help/images/ref/dice.html
bfscore	Contour matching score https://www.mathworks.com/help/images/ref/bfscore.html

Summary of image segmentation techniques

Image segmentation

- There is no underlying theory of image segmentation, only *ad hoc* methods, whose performance is often evaluated indirectly, based on the performance of the larger system to which they belong.

- Image segmentation techniques can vary widely according to:
  - **type of image** (e.g., binary, gray, color, multispectral)
  - choice of **mathematical framework** (e.g., morphology, image statistics, graph theory)
  - type of **features** (e.g., intensity, color, texture, motion) and
  - **approach** (e.g., top-down, bottom-up, graph-based).

Source: Marques (2011) “Practical Image and Video Processing Using MATLAB”, Wiley.

Categories

- There is no universally accepted taxonomy for classification of image segmentation algorithms.

- Image segmentation algorithms usually fall into one of these categories (names of MATLAB library functions):
  - **Threshold-based techniques**: `graythresh, multithresh, otsuthresh, adaptthresh`
  - **Region growing / flood fill**: `grayconnected, imseggeodesic, imsegfmm, gradientweight, graydiffweight`
  - **Active contours / snakes**: `activecontour`
  - **Clustering based**: `imsegkmeans, imsegkmeans3`
  - **Morphological**: `watershed`
  - **Superpixel segmentation**: `superpixels, superpixels3`
  - **Graph-based**: `lazysnapping, grabcut`

Segmentation techniques in MATLAB

https://www.mathworks.com/help/images/image-segmentation.html

graythresh	Global image threshold using Otsu's method
multithresh	Multilevel image thresholds using Otsu's method
otsuthresh	Global histogram threshold using Otsu's method
adaptthresh	Adaptive image threshold using local first-order statistics
grayconnected	Select contiguous image region with similar gray values using flood-fill technique
watershed	Watershed transform
activecontour	Segment image into foreground and background using active contours (snakes) region growing technique
lazysnapping	Segment image into foreground and background using graph-based segmentation
grabcut	Segment image into foreground and background using iterative graph-based segmentation
imseggeodesic	Segment image into two or three regions using geodesic distance-based color segmentation
imsegfmm	Binary image segmentation using fast marching method
gradientweight	Calculate weights for image pixels based on image gradient
graydiffweight	Calculate weights for image pixels based on grayscale intensity difference
imsegkmeans	K-means clustering based image segmentation
imsegkmeans3	K-means clustering based volume segmentation
superpixels	2-D superpixel oversegmentation of images
superpixels3	3-D superpixel oversegmentation of 3-D image

Segmentation: Threshold-based techniques

- Thresholding:
  "The conversion of an image with many gray levels into another image with fewer gray levels, usually only two."
  - This conversion is usually performed by comparing each pixel intensity against a reference value (*threshold*) and replacing the pixel with a value that means 'white' or 'black' depending on the outcome of the comparison.

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{otherwise} \end{cases}$$

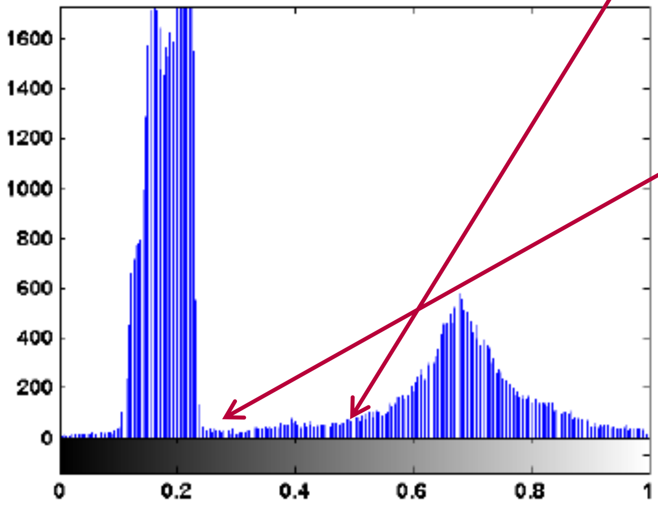
Source: Marques (2011) "Practical Image and Video Processing Using MATLAB", Wiley.

Threshold-based techniques: variants

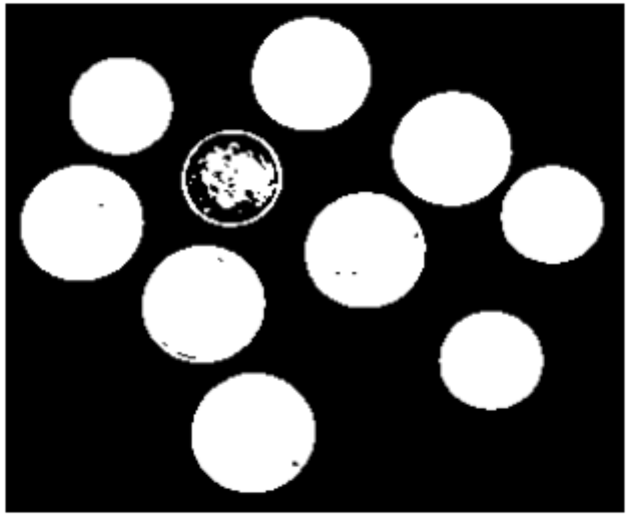
- Variants
  - Global thresholding (`graythresh, otsuthresh`)
    - Computes a single value of *T* for the whole image using Otsu's method, which chooses a threshold that minimizes the intraclass variance of the thresholded black and white pixels. The resulting value can be used to binarize the image using `imbinarize`.
  - Multilevel thresholding (`multithresh`)
    - Computes multiple values of *T* for the whole image using Otsu's method, which can be used to requantize the image using `imquantize`.
  - Adaptive thresholding (`adaptthresh`)
    - Computes a locally adaptive threshold, based on the local mean intensity (first-order statistics) in the neighborhood of each pixel.

Threshold-based techniques: examples

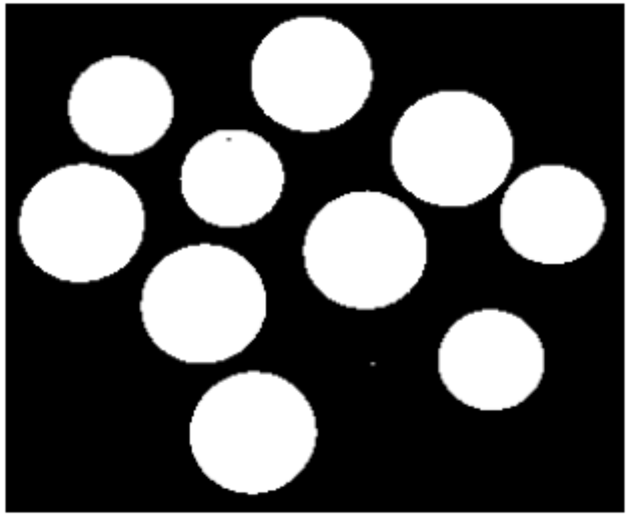
Different values of T lead to different results



T = 0.4947



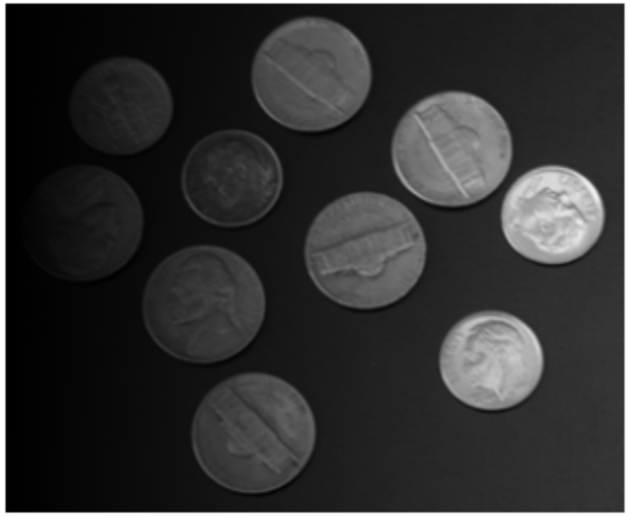
T = 0.25

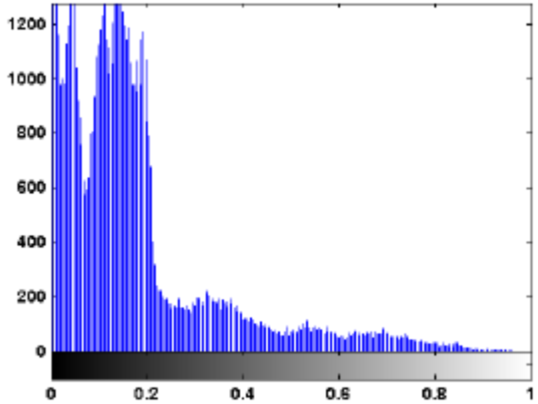


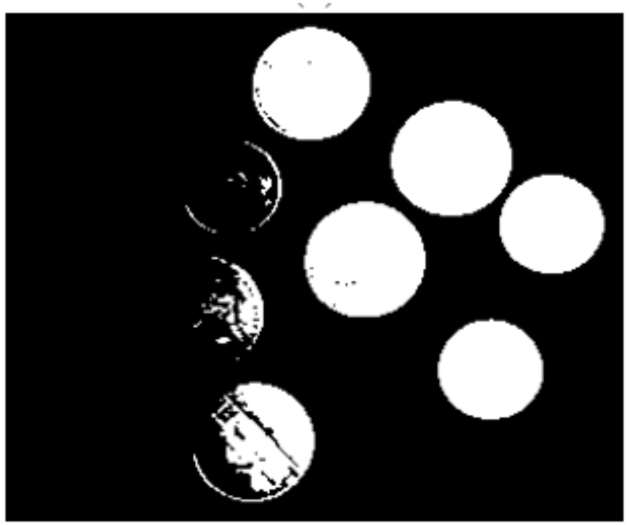
Source: Marques (2011) “Practical Image and Video Processing Using MATLAB”, Wiley.

Threshold-based techniques: examples

Global thresholding is sensitive to uneven illumination



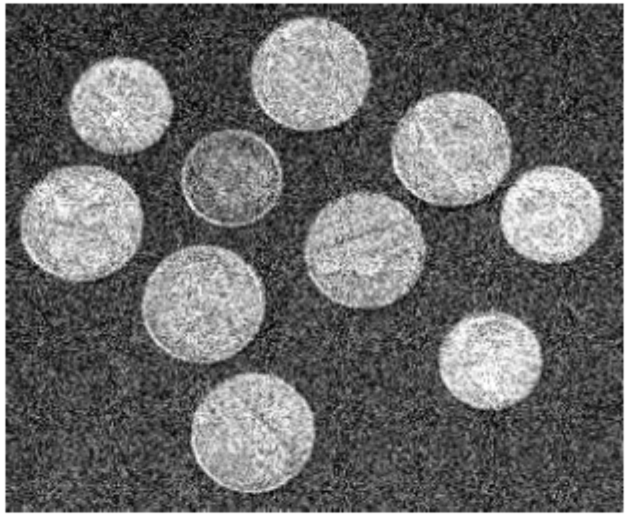


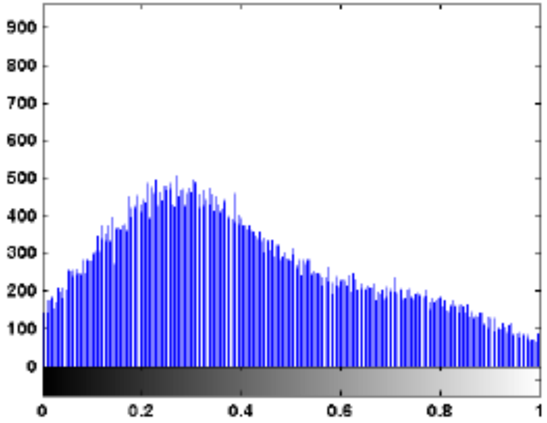


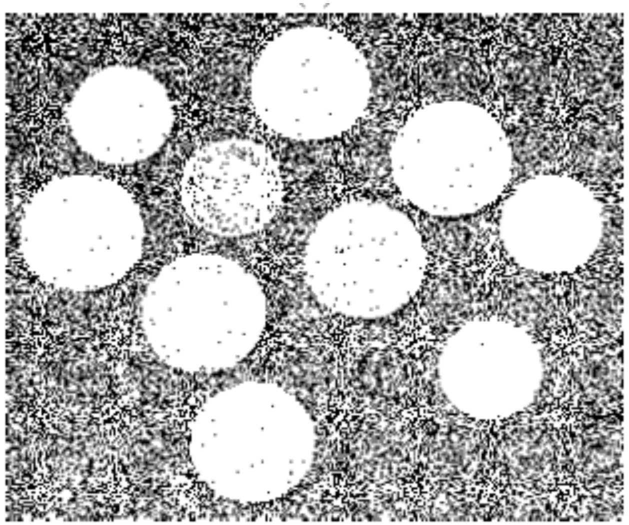
Source: Marques (2011) “Practical Image and Video Processing Using MATLAB”, Wiley.

Threshold-based techniques: examples

Global thresholding
is sensitive to noise







Source: Marques (2011) “Practical Image and Video Processing Using MATLAB”, Wiley.

Segmentation: Region growing techniques

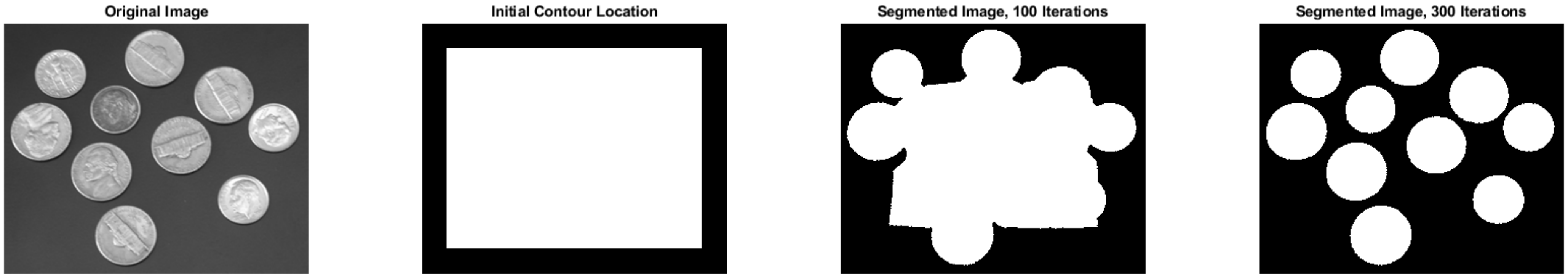
- Region growing is a simple pixel-based image segmentation method that starts from a pixel location (the seed) and grows a region around it according to certain criteria.
  - Essentially, “keep growing until the pixels within the connected region are similar to each other and all the surrounding pixels are sufficiently different – according to some definition of similarity”.

- Implementation options (using MATLAB):
  - `grayconnected` : classic flood fill algorithm
  - `imseggeodesic` : geodesic-distance-based algorithm
  - `imsegfmm` : fast-marching method (Sethian, 1999)
  - `gradientweight`, `graydiffweight` : auxiliary functions to compute weights for image pixels based on image gradient (i.e., grayscale intensity difference)

Source: https://www.mathworks.com/discovery/image-segmentation.html

Segmentation: Active contours (snakes)

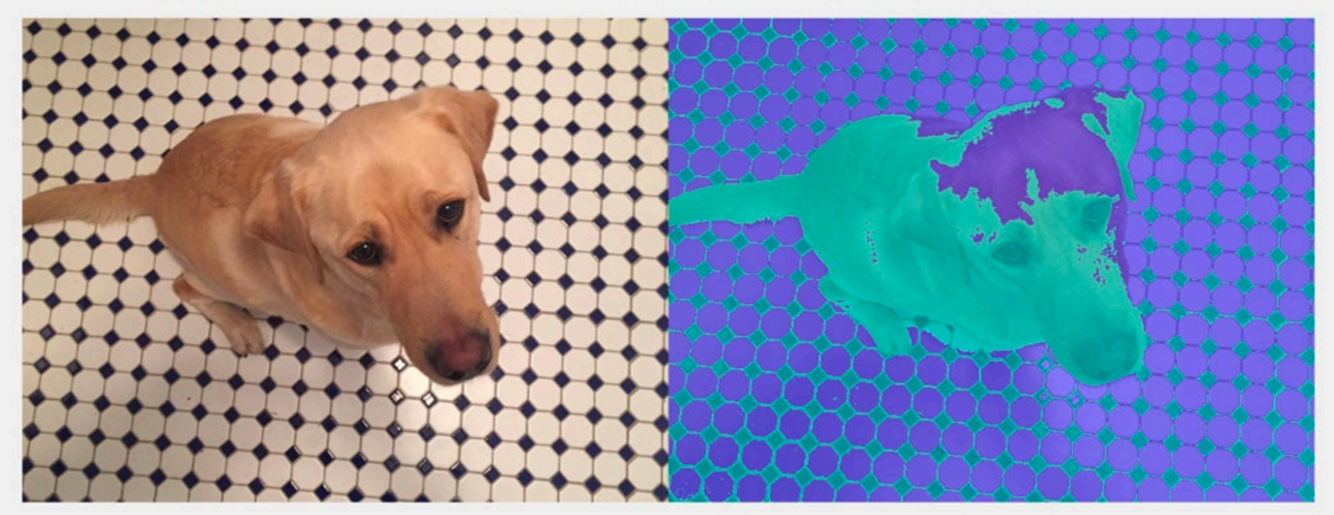
- Iterative region-growing image segmentation algorithm.
  - Using the active contour algorithm, you specify initial curves on an image and then use the `activecontour` function to evolve the curves towards object boundaries.



Source: https://www.mathworks.com/help/images/ref/activecontour.html

Segmentation: clustering-based techniques

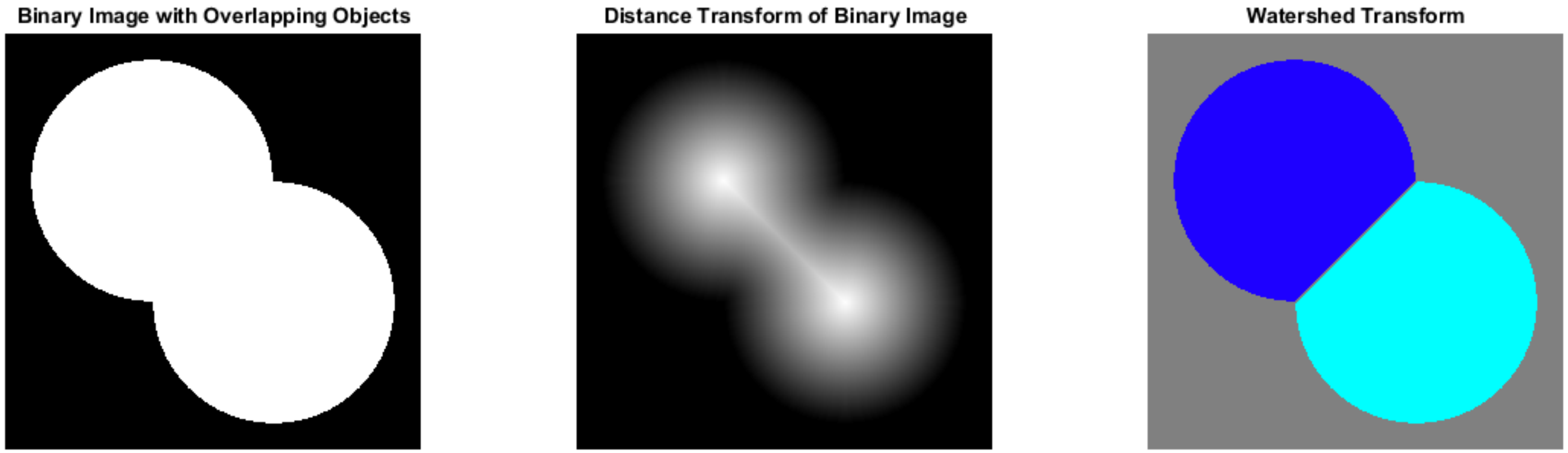
- Basic idea: treat segmentation as a clustering problem, in which pixels with similar intensity levels are labeled as belonging to the same cluster.
- Popular algorithm: k-means clustering.
  - In MATLAB: `imsegkmeans` (2D images) and `imsegkmeans3` (3D volumes)
- Major shortcoming: it disregards connectivity information.



Source: https://www.mathworks.com/discovery/image-segmentation.html

Segmentation: morphological techniques

- Uses the watershed transform, which finds "catchment basins" or "watershed ridge lines" in an image by treating it as a surface where light pixels represent high elevations and dark pixels represent low elevations.
- The watershed transform can be used to segment contiguous regions of interest into distinct objects.
  - **In MATLAB:** `watershed`

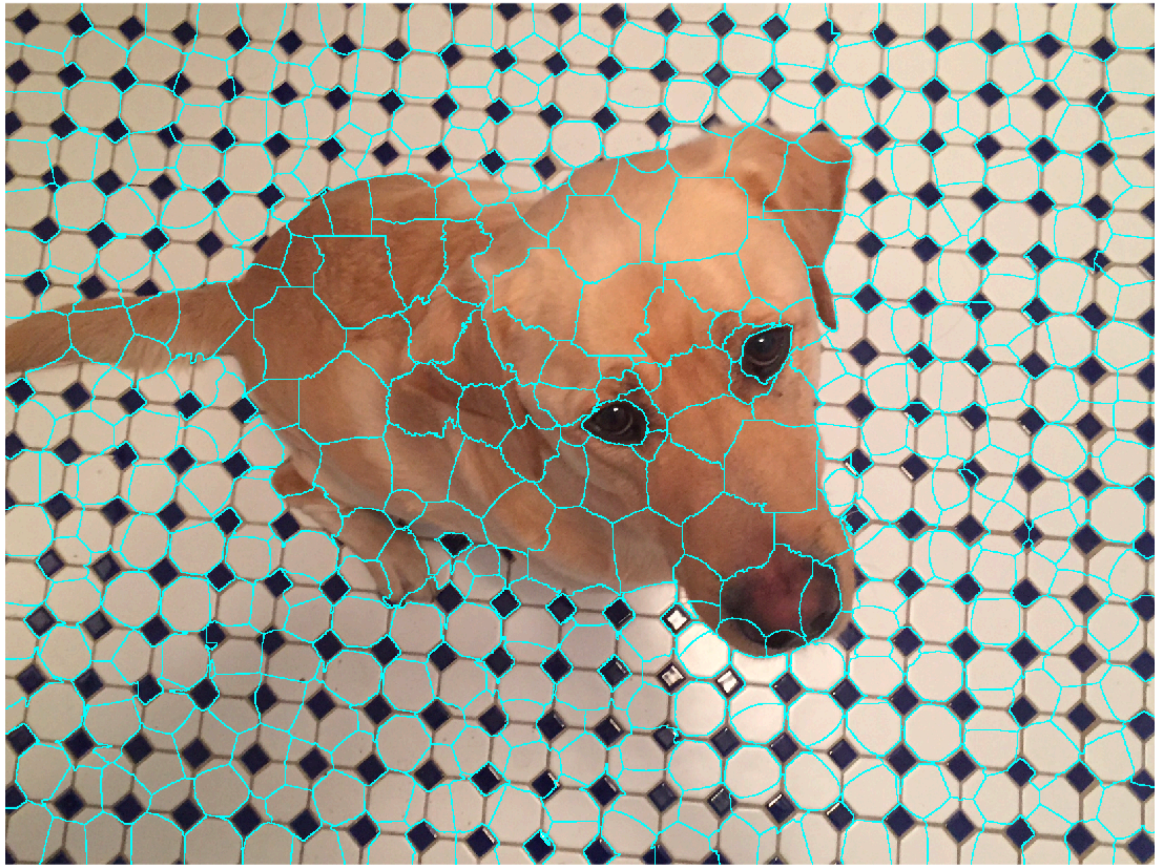


Source: https://www.mathworks.com/help/images/ref/watershed.html

Segmentation: superpixel techniques

- Basic idea: group pixels into regions with similar values.
    - Using these regions in image processing operations, such as segmentation, can reduce the complexity of these operations. simple linear iterative clustering (SLIC) algorithm.

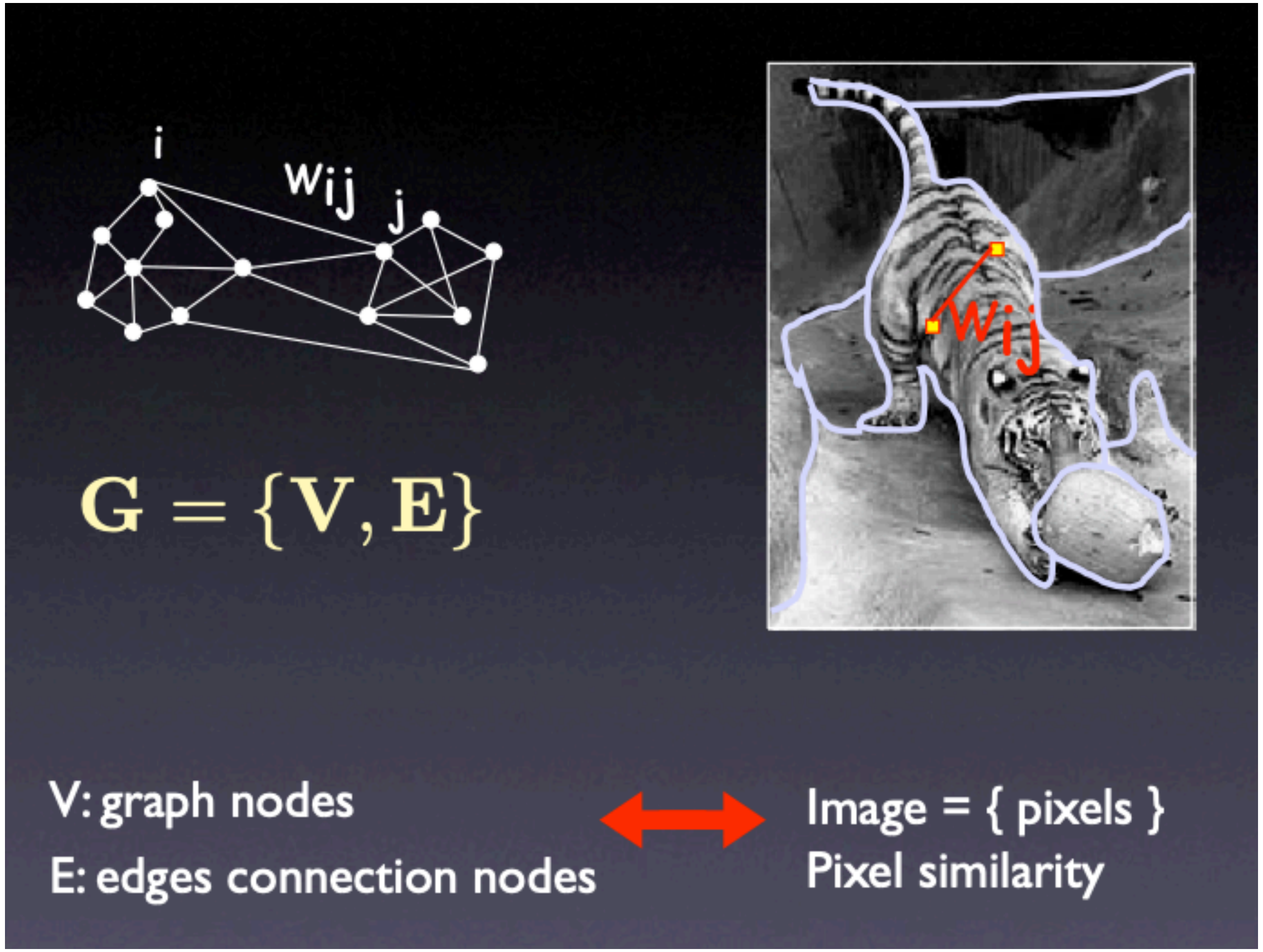
- **In MATLAB:**
  `superpixels` (2D images) and
  `superpixels3` (3D volumes)



Source: https://www.mathworks.com/help/images/ref/superpixels.html

Segmentation: graph-based techniques

- **Basic idea**:
  treat segmentation as a graph cut problem, in which pixels are the graph nodes and pixel similarity is treated as the edges connecting the nodes.

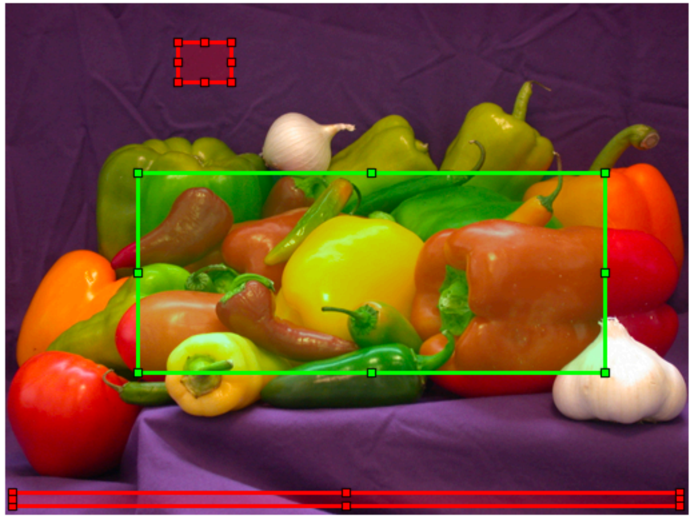


Source: https://www.cis.upenn.edu/~jshi/GraphTutorial/Tutorial-ImageSegmentationGraph-cut1-Shi.pdf

Segmentation: graph-based techniques

- In MATLAB: `lazysnapping`

- Example: background removal



Source: https://www.mathworks.com/help/images/ref/lazysnapping.html

Image segmentation apps in MATLAB

Image Segmentation with MATLAB: beyond library function calls

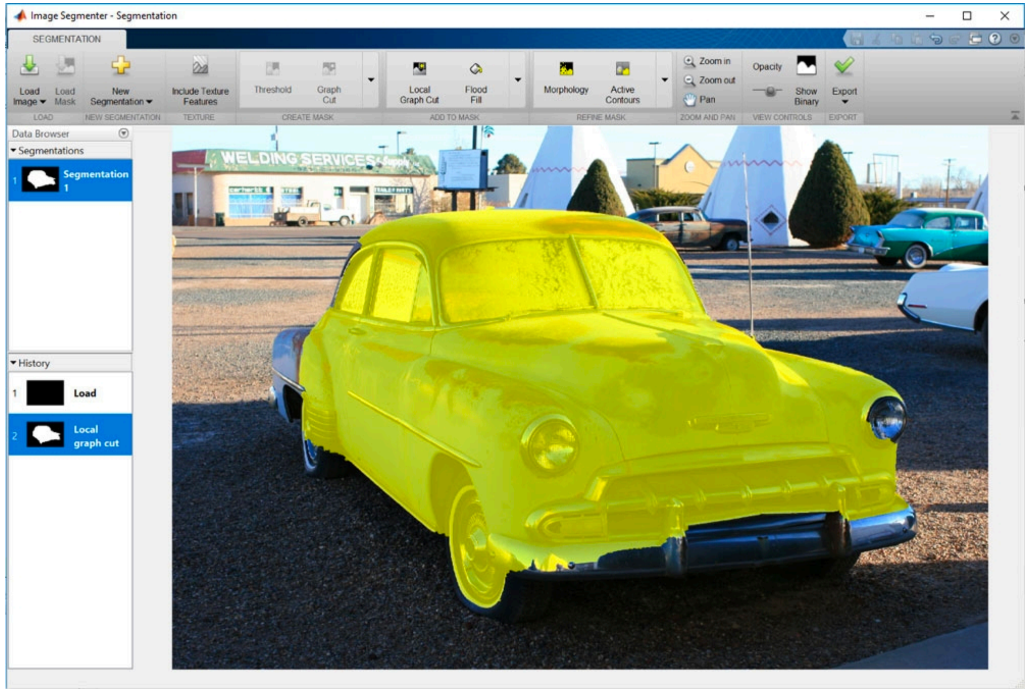
MATLAB has interactive GUI-based apps for:

- Image segmentation (Image Segmenter)
- Image thresholding (Color Thresholder)
- Image labeling, i.e. ground truth creations (Image Labeler)

Source: https://www.mathworks.com/discovery/image-segmentation.html

Image Segmenter App

Interactive app that lets you try several methods to create segmentation masks, using automatic algorithms (e.g., flood fill), semi-automatic techniques (e.g., graph cut), and manual techniques (e.g., drawing ROIs). You can also refine masks using morphology or an iterative approach such as active contours (*snakes*).
Resulting masks (and the code used to create them) can be exported to the workspace.



Learn more at:

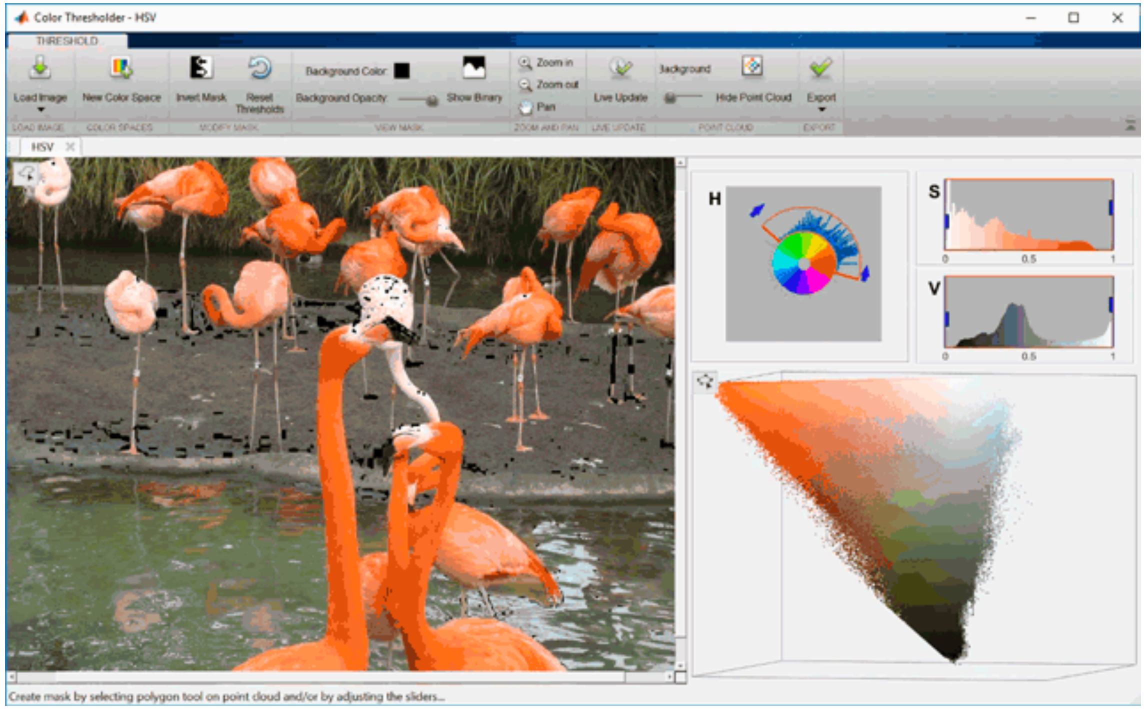
https://www.mathworks.com/help/images/image-segmentation-using-the-image-segmenter-app.html

Source: https://www.mathworks.com/discovery/image-segmentation.html

Color Thresholder App

Interactive app that lets you apply thresholding to color images by manipulating the color of the images interactively, based on different color spaces.

Useful for creating segmentation masks for color images.

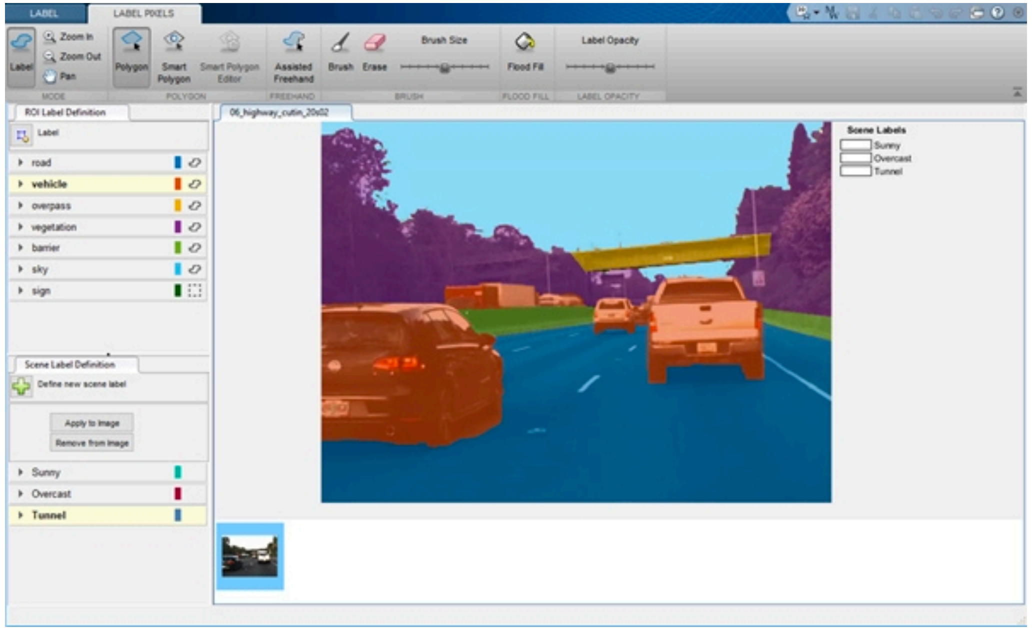


Learn more at:
https://www.mathworks.com/help/images/ref/colorthresholder-app.html

Source: https://www.mathworks.com/discovery/image-segmentation.html

Image Labeler App

Allows you to interactively label your ground truth data.
For segmentation tasks, labeling can be done at a pixel level.
Supports built-in labeling automation algorithms and allows you to use your own.



Check out (review) “Assignment 0: Create your own dataset of labeled images”

Learn more at:
https://www.mathworks.com/help/vision/ref/imagelabeler-app.html

Source: https://www.mathworks.com/discovery/image-segmentation.html

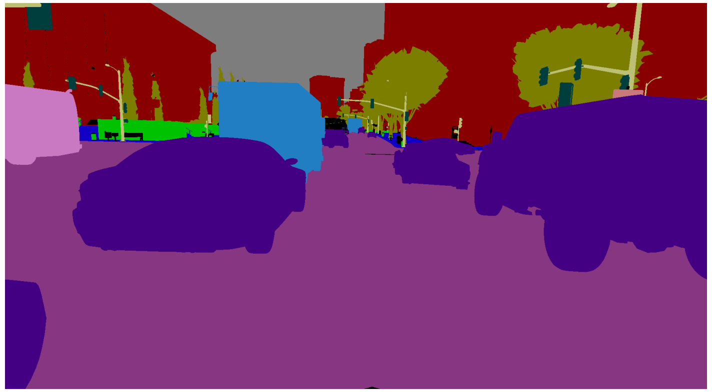
Datasets, benchmarks, challenges

PASCAL-context dataset

- This dataset is a set of additional annotations for PASCAL VOC 2010.
- It goes beyond the original PASCAL semantic segmentation task by providing annotations for the whole scene.
- It contains 400+ labels.
- https://cs.stanford.edu/~roozbeh/pascal-context/

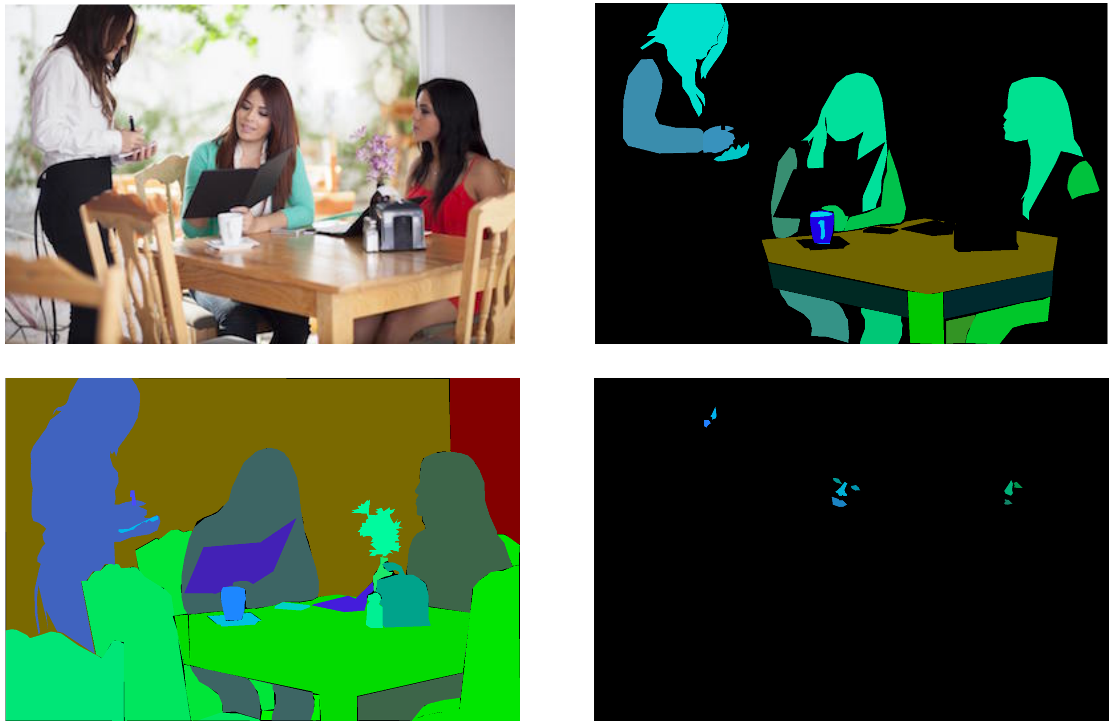
Playing for Data: Ground Truth from Computer Games

- The datasets consists of 24966 densely labelled frames, split into 10 parts for convenience.
- The class labels are compatible with the CamVid and CityScapes datasets.
- https://download.visinf.tu-darmstadt.de/data/from_games/



MIT ADE20K

- 22,000+ images separated by scene category (same scene categories than the Places Database).
- For each image, the object and part segmentations are stored in two different files.
- All object and part instances are annotated separately.
- https://groups.csail.mit.edu/vision/datasets/ADE20K/

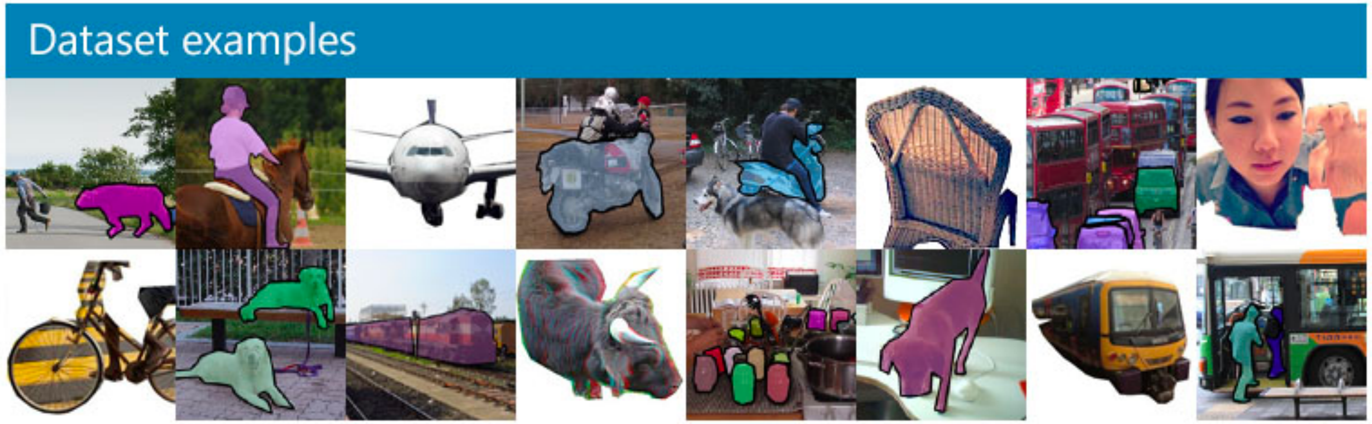


Stanford Background Dataset (SBD)

- Introduced in *Gould et al. (ICCV 2009)* for evaluating methods for geometric and semantic scene understanding.
- The dataset contains 715 images chosen from existing public datasets: LabelMe, MSRC, PASCAL VOC and Geometric Context.
- Selection criteria were for the images to be of outdoor scenes, have approximately 320-by-240 pixels, contain at least one foreground object, and have the horizon position within the image (it need not be visible).
- http://dags.stanford.edu/projects/scenedataset.html

Microsoft COCO (Common Objects in Context)

- COCO is a large-scale object detection, segmentation, and captioning dataset.
- Features:
    - Object segmentation
    - Recognition in context
    - Superpixel stuff segmentation
    - 330K images (>200K labeled)
    - 1.5 million object instances
    - 80 object categories
    - 91 stuff categories
    - 5 captions per image
    - 250,000 people with keypoints
- https://cocodataset.org/



Semantic image segmentation using deep learning in MATLAB

Semantic Image Segmentation using Deep Learning in MATLAB

Basic steps:

1. Label training images with Image Labeler app (or search Web for labeled data)
2. Create two datastores
   - ImageDatastore
   - PixelLabelDatastore
3. Partition datastores into training and test sets
4. Import a pretrained CNN model and modify it to become an encoder-decoder network (or create your own network from scratch)
5. Train and evaluate your network.

Source: https://www.mathworks.com/videos/semantic-segmentation-overview-1510858047780.html

Semantic Image Segmentation using Deep Learning in MATLAB

Step 1: Label your data

- **Define your problem**.
  - What is the context?
  - How do images look like? Where do they come from?
  - What are the relevant portions within the images? How should they be labeled?
  - What is the goal of applying semantic segmentation techniques to those images?
  - What will the semantic segmentation results be used for?
  - How can you measure the quality of the solution?

- **Collect useful images** and organize them in a folder.

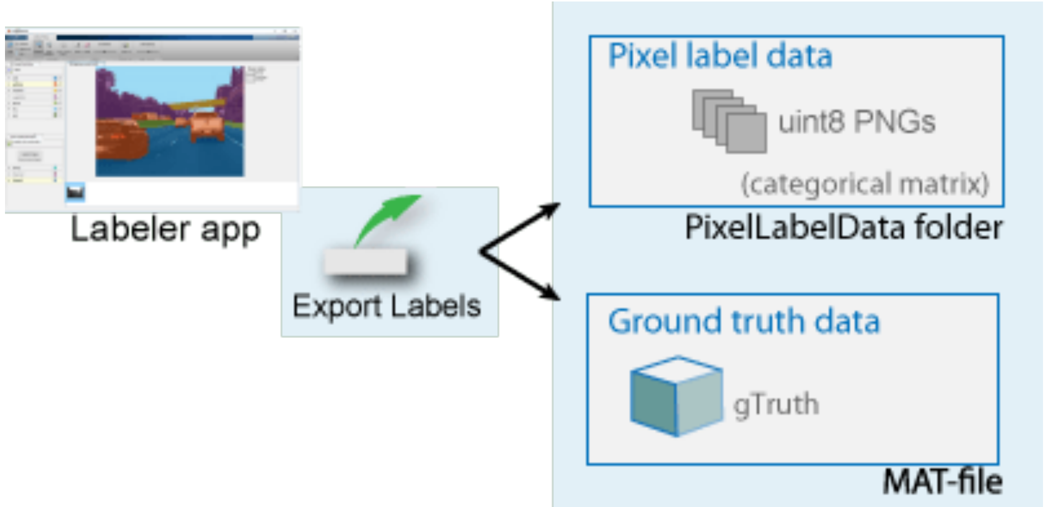
- **Prepare a list of labels** that will be used to label selected regions of interest (ROIs) within each image.

Semantic Image Segmentation using Deep Learning in MATLAB

Step 1: Label your data (cont'd)

- **Use MATLAB Image Labeler app to generate pixel-level masks for the regions of interest (ROIs)**

- **Export the labels to the MATLAB workspace**.
  This will trigger the creation of a collection of mask images (in the *PixelLabelData* folder) and a ground truth MAT-file.

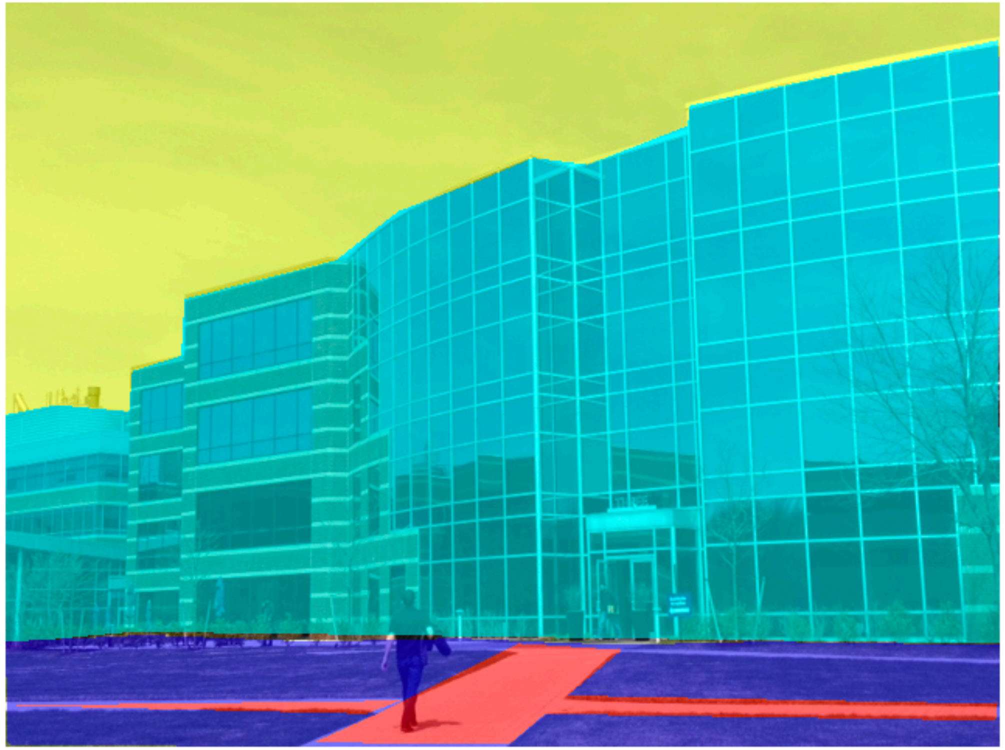


Semantic Image Segmentation using Deep Learning in MATLAB

Step 2: Create two datastores

- ImageDatastore
  - Contains the original images
- PixelLabelDatastore
  - Contains the labeled images (masks)

- Example of masks (sky, building, grass, sidewalk) overlapped on input image



Step 3: Partition datastores into training and test sets

Semantic Image Segmentation using Deep Learning in MATLAB

Step 4: Import a pretrained CNN model and modify it to become an encoder-decoder network (or create your own network from scratch)

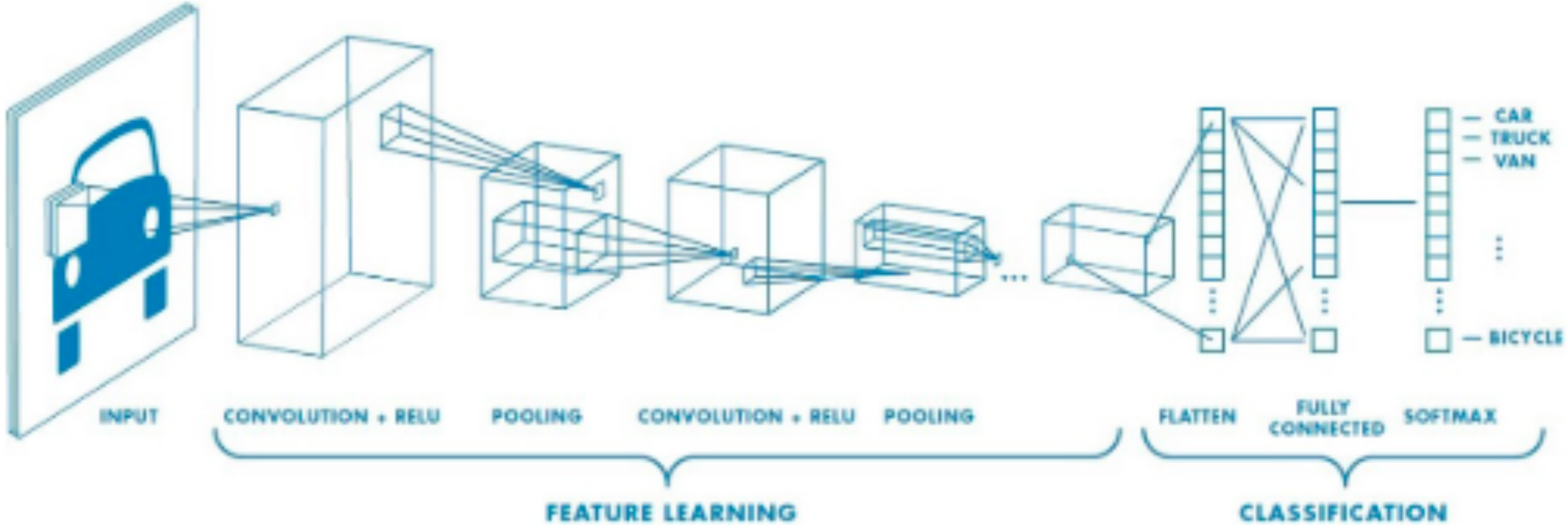
- Commonly used encoder-decoder networks (and associated base models):
  - SegNet (Vanilla CNN, VGG16, Resnet-50, MobileNet)
  - U-Net (Vanilla CNN, VGG16, Resnet-50, MobileNet)
  - Fully Convolutional Networks
    - FCN8 (Vanilla CNN, VGG16)
    - FCN32 (VGG16, Resnet-50, MobileNet)
  - Pyramid Scene Parsing – PSPNet (Vanilla CNN, VGG16, Resnet-50)
  - Deeplab v3+ (Resnet-18, Resnet-50, MobileNet)

Step 5: Train and evaluate your network.

Source: https://github.com/divamgupta/image-segmentation-keras

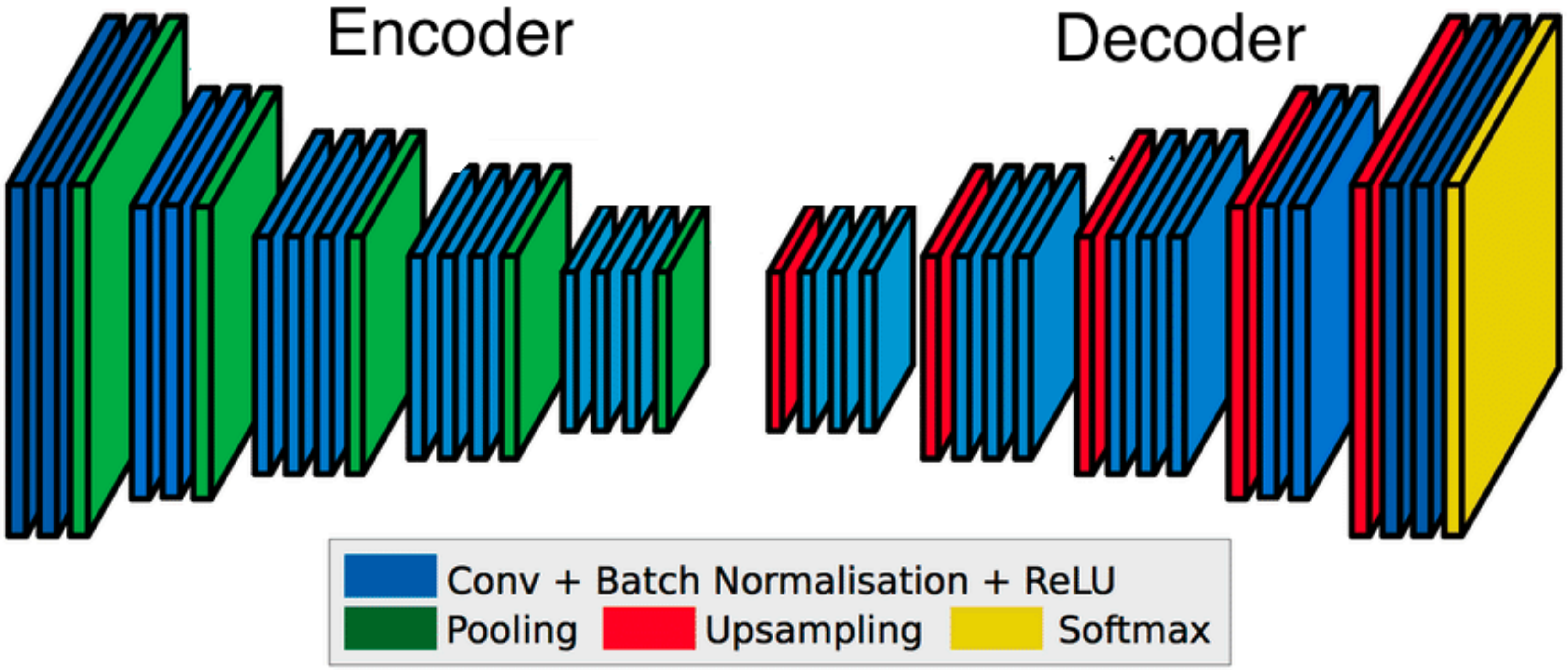
Sidenote: CNNs, encoder-decoder, and transfer learning

- Traditional CNN



Sidenote: CNNs, encoder-decoder, and transfer learning

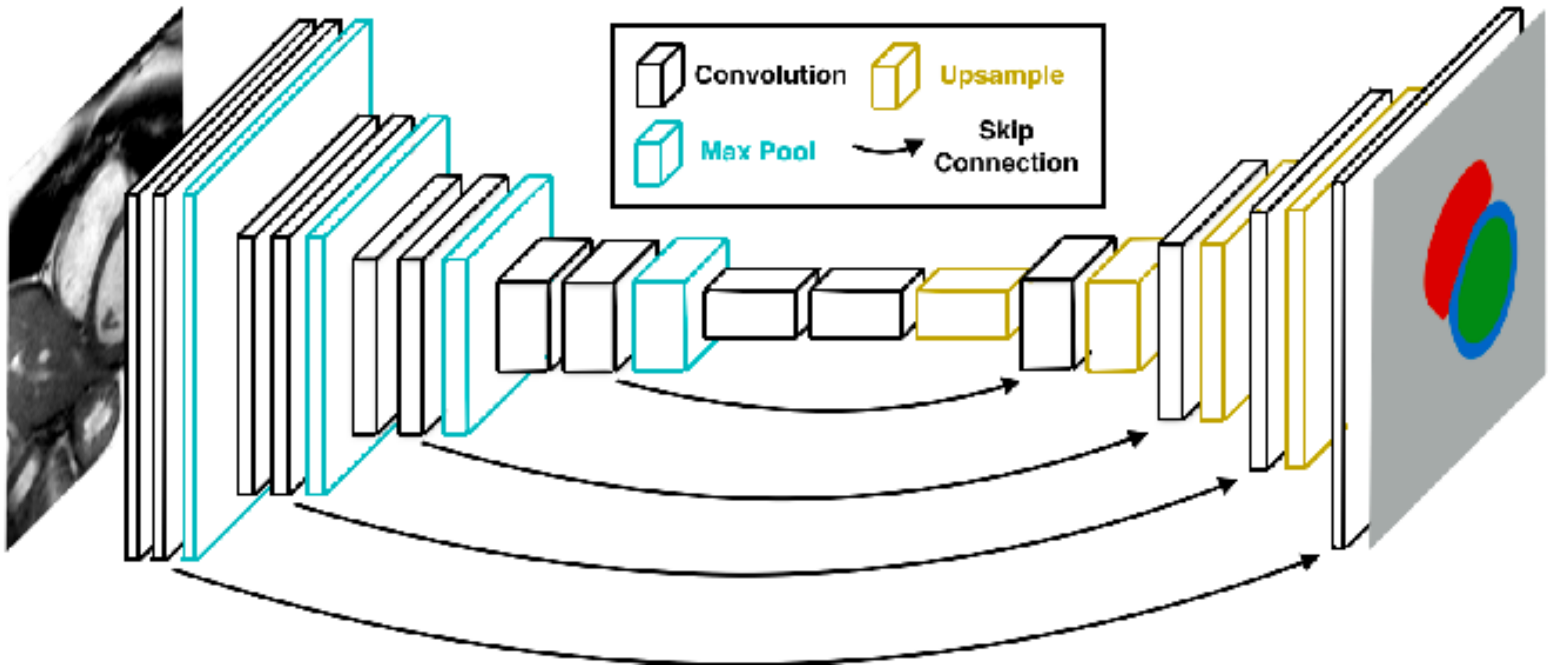
- Encoder-decoder architecture



Source: https://www.semanticscholar.org/paper/SSeg-LSTM%3A-Semantic-Scene-Segmentation-for-Syed-Morris/b5e4abb8c1bebf7d531202e57e431c638dd1c4af

Sidenote: CNNs, encoder-decoder, and transfer learning

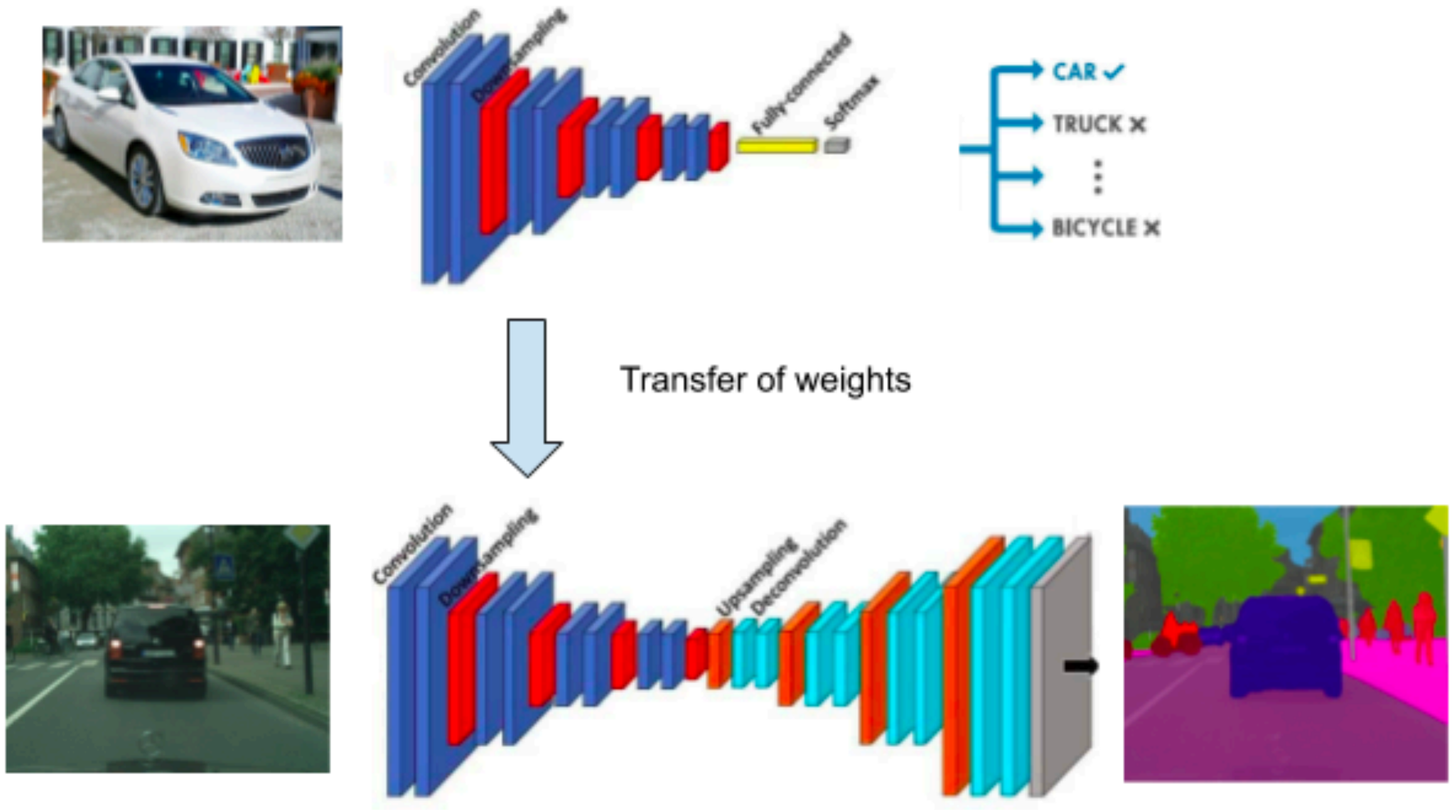
- Encoder-decoder with skip connections



Source: https://www.groundai.com/project/fastventricle-cardiac-segmentation-with-enet/

Sidenote: CNNs, encoder-decoder, and transfer learning

- Transfer Learning



Source: https://divamgupta.com/image-segmentation/2019/06/06/deep-learning-semantic-segmentation-keras.html

Semantic Image Segmentation using Deep Learning in MATLAB

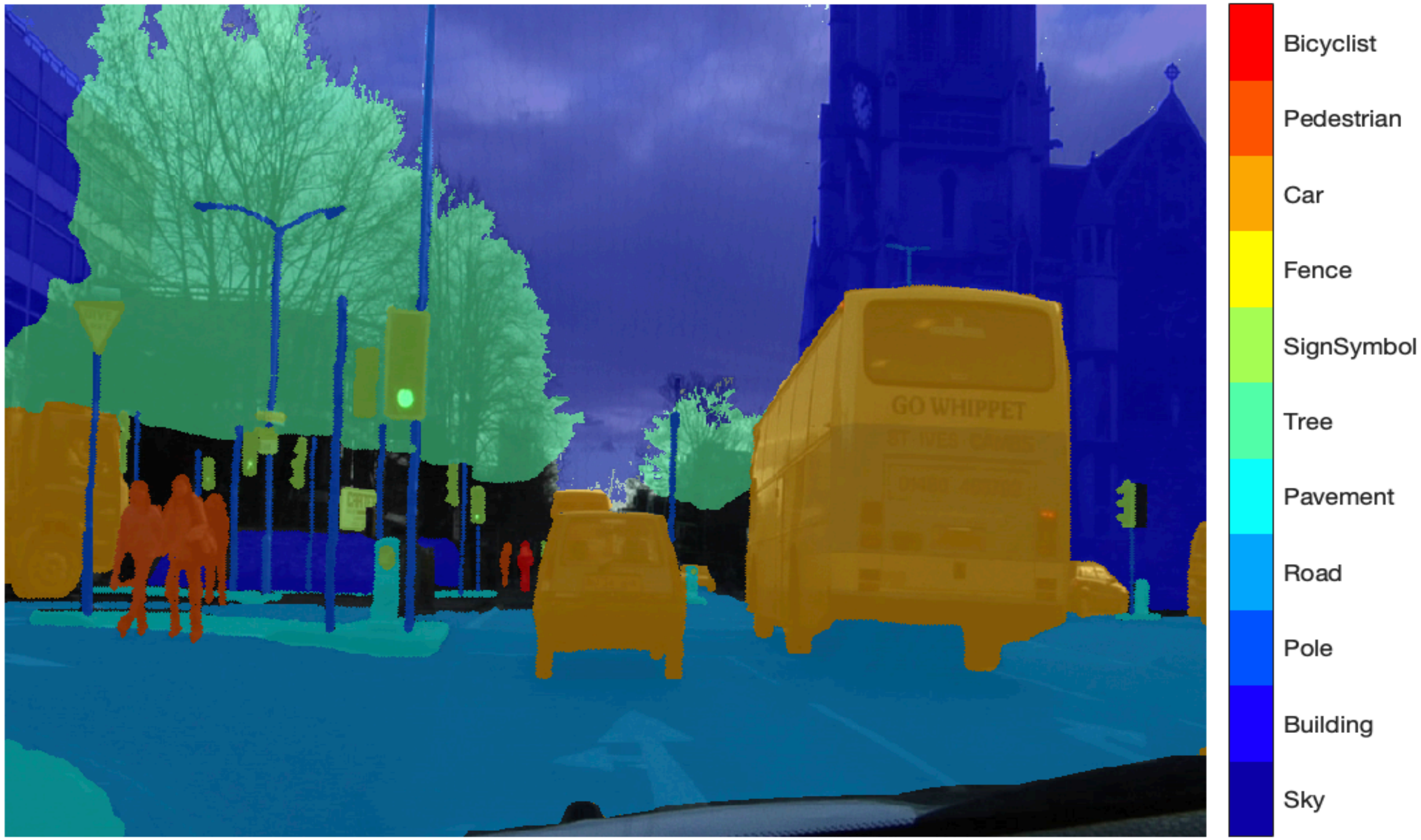
Key MATLAB functions:

- `semanticseg`
  Takes an image and a pretrained network as input and returns semantic segmentation of the input image.

- `evaluateSemanticSegmentation`
  Evaluates semantic segmentation data set against ground truth.

Hands on!

MATLAB Assignment:
Semantic image segmentation using deep learning (semantic_segmentation_dl.mlx)



MATLAB Assignment:
Semantic image segmentation using deep learning (semantic_segmentation_dl.mlx)

- Part 1:
  - Basic workflow for semantic image segmentation using a convolution-deconvolution neural network *created by you and trained from scratch*.
  - Segmentation metrics: global accuracy, class accuracy, IoU, weighted IoU, and BF score.
- Part 2:
  - Workflow for semantic image segmentation using a *pretrained* network and a *realistic* public dataset (CamVid).
  - Data augmentation.