Semantic Segmentation

Danna Gurari
University of Colorado Boulder
Fall 2024

https://dannagurari.colorado.edu/course/recent-advances-in-computer-vision-fall-2024/

Review

- Last lecture:
  - Scene Classification Problem and Applications
  - Scene Classification Datasets and Evaluation Metrics
  - Scene Classification Models: Deep Features
  - Attribute Classification: Problem, Applications, and Datasets

- Assignments (Canvas)
  - Reading assignment was due earlier today
  - Next reading assignment due Monday
  - Project proposal due in one week

- Questions?

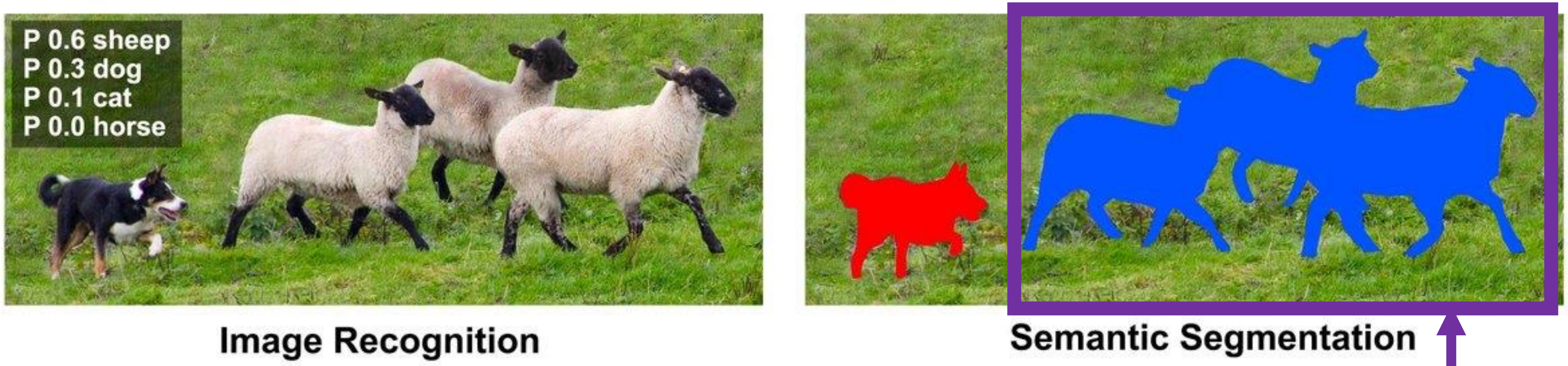
Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Today's Scope: Localize Pixels for Each Category



Note: instances of the same category are NOT separated

https://ai-pool.com/d/could-you-explain-me-how-instance-segmentation-works

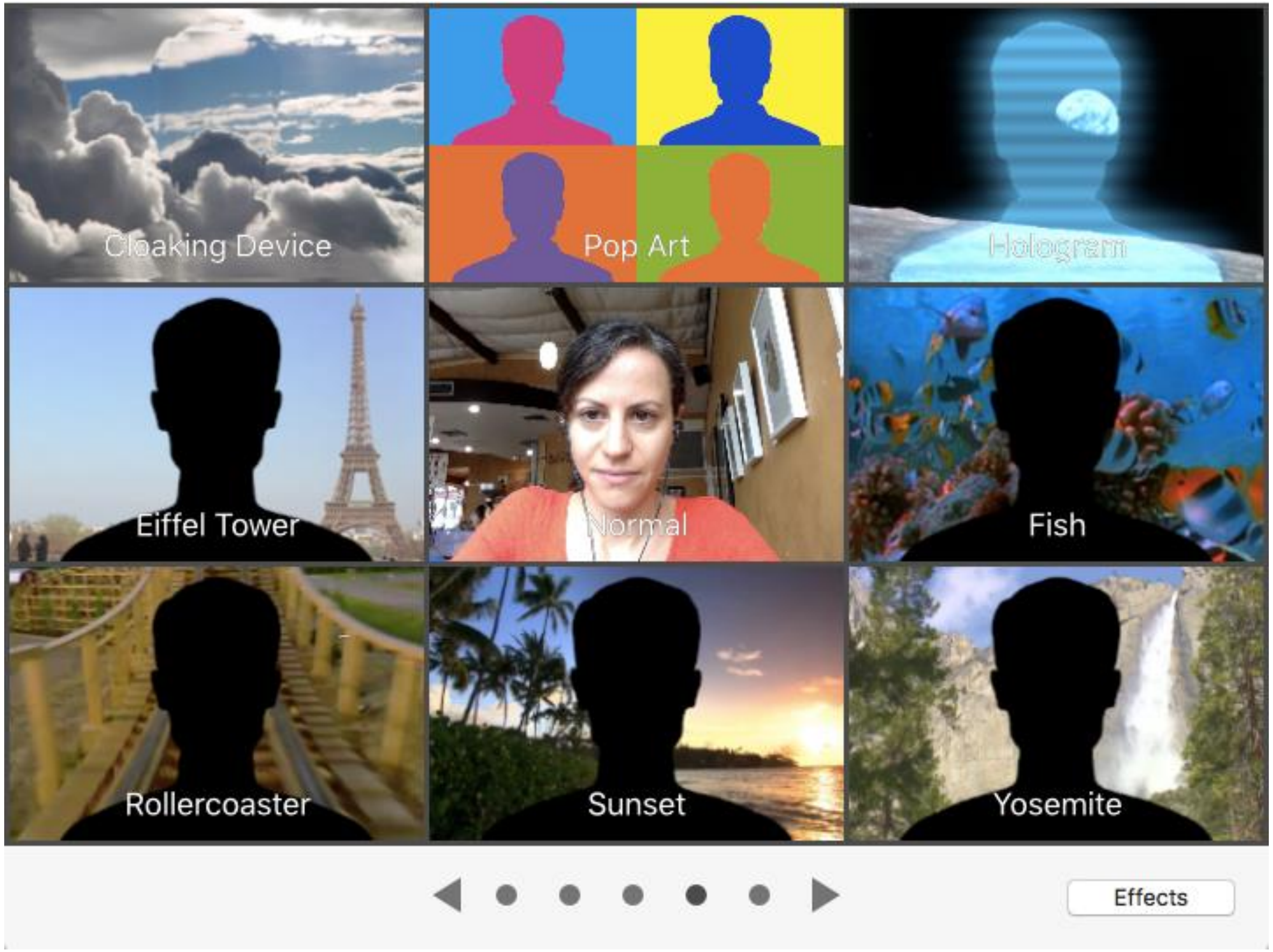
Remodeling Inspiration

(a) Target photo

(b) Retextured

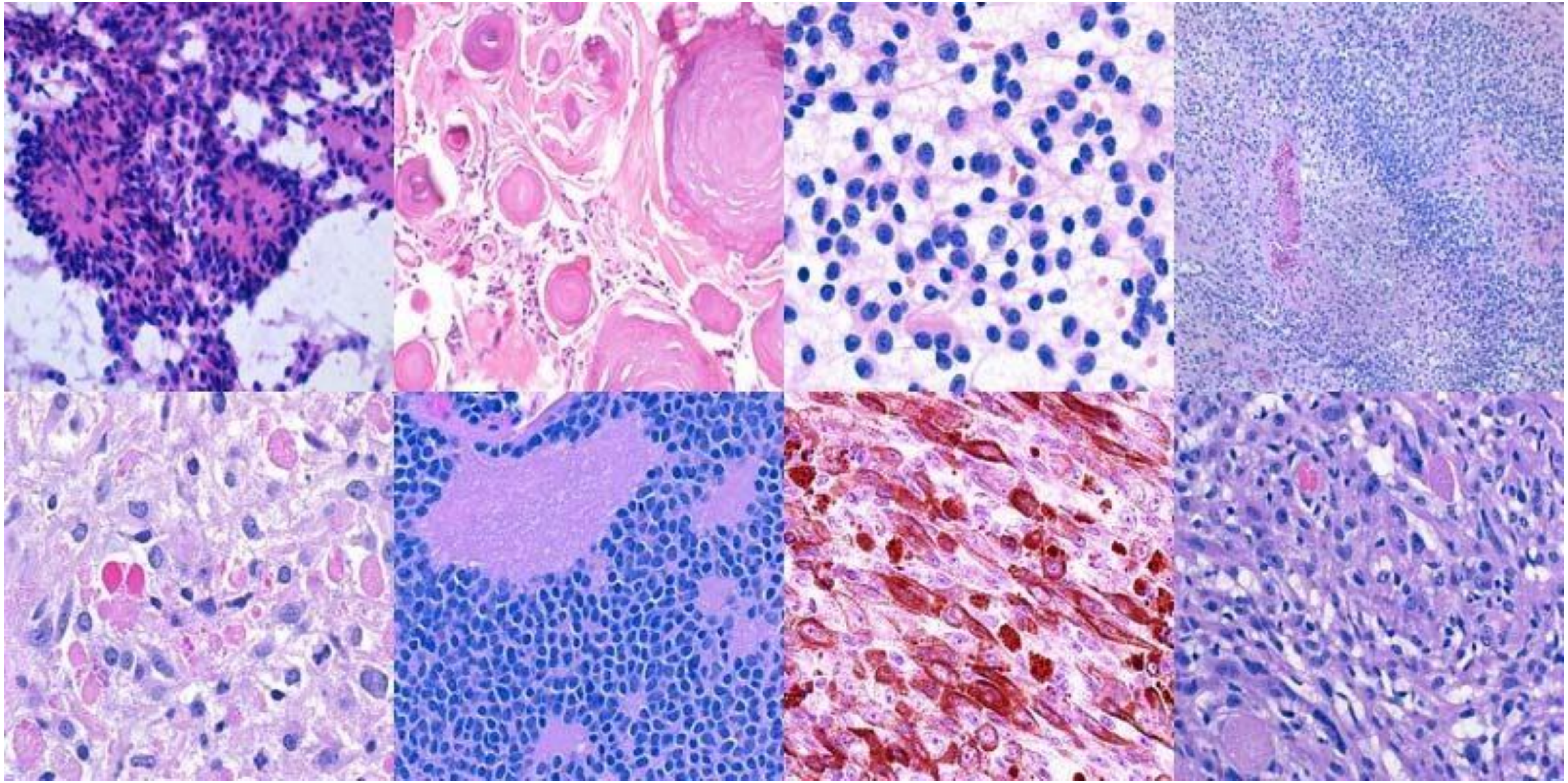
Bell et al; SIGGRAPH; 2013

Rotoscoping (many examples on Wikipedia)



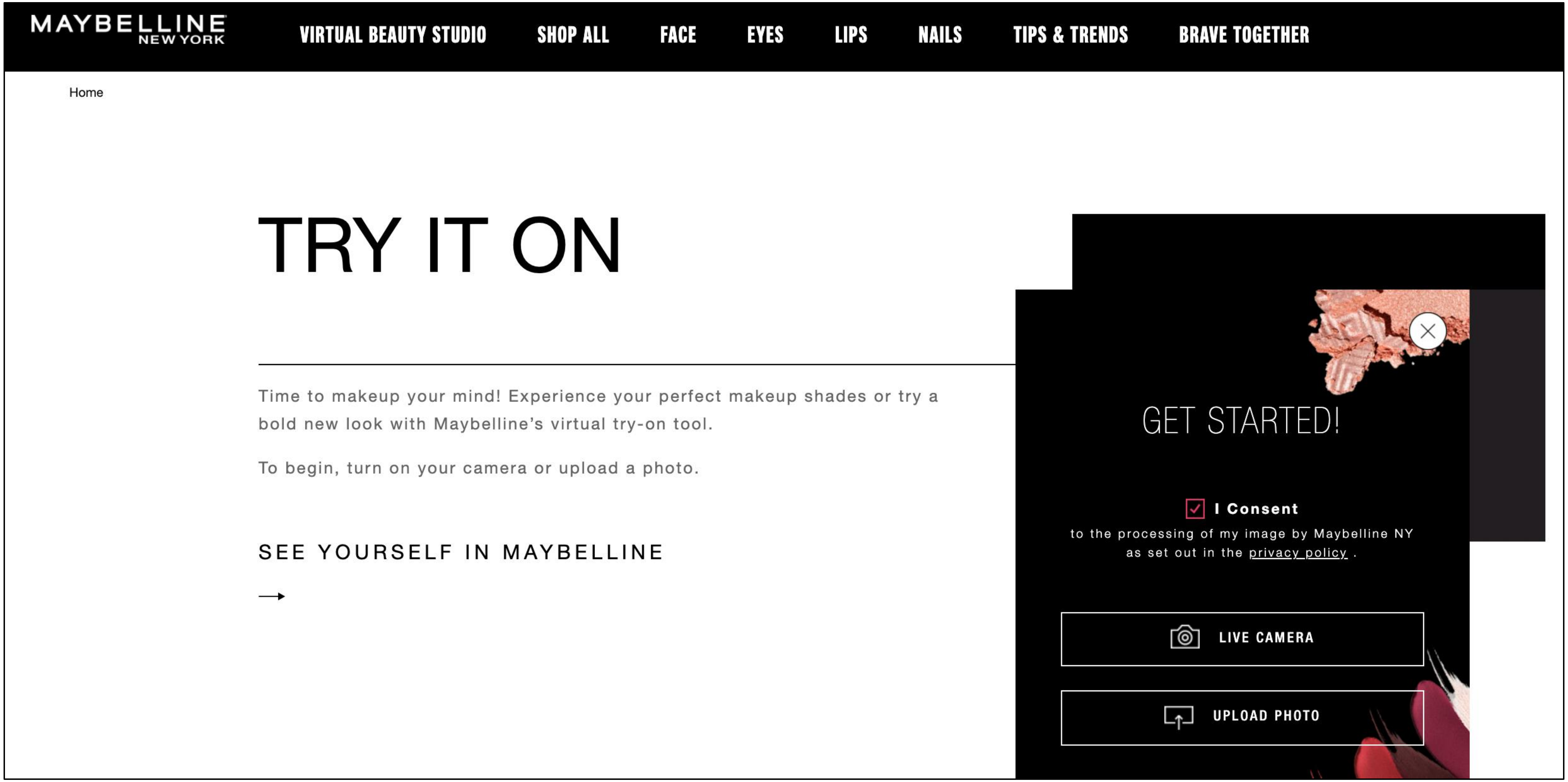
https://www.starnow.co.uk/ahmedmohammed1/photos/4650871/before-and-after-rotoscopinggreen-screening

Disease Diagnosis; e.g., PathAI



https://pathology.jhu.edu/brain-tumor/grading-classification

Face Makeover



Demo: https://www.maybelline.com/virtual-try-on-makeup-tools

Self-Driving Vehicles

https://www.inc.com/kevin-j-ryan/self-driving-cars-powered-by-people-playing-games-mighty-ai.html

Can you think of any other potential applications?

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Datasets

1945 — 1957 — 1966 — 1983 — 1987 — 1990 — 2009 — 2017



CVPR (1983), ICCV (1987), ECCV (1990)

e.g., VOC (2009), ADE20k (2017)

	VOC	ADE20k
# Categories:	21	3,169
# Images:	1112 train/val	25,210

Trend: build bigger datasets

VOC

1. Category Selection

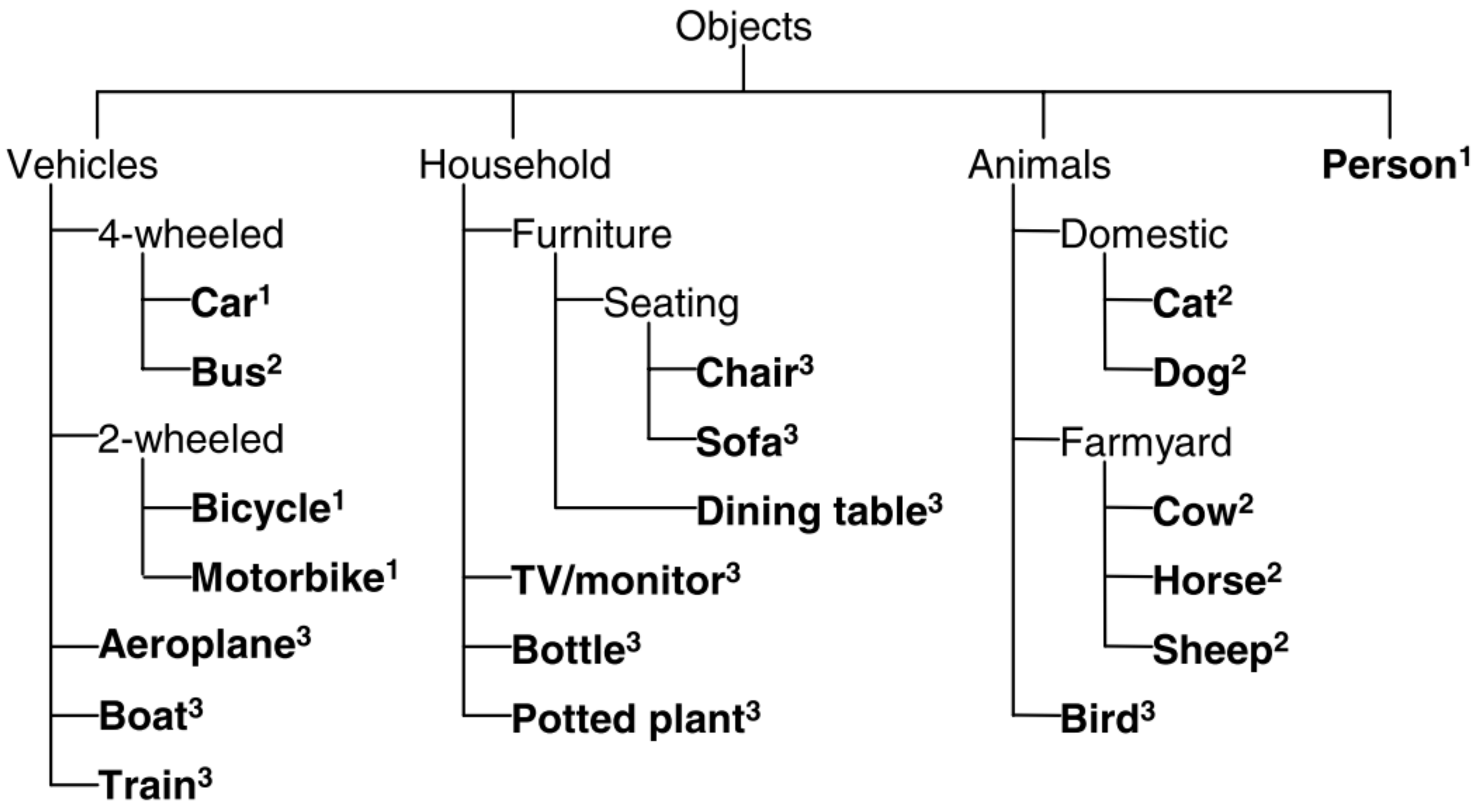
- 20 categories chosen:

1) Initial 4 categories stem from existing dataset

2) 2006: added 6 classes

3) 2007: added 10 classes

- Categories added for more generalization and finer-grained coverage



(superscript indicates year of inclusion in the challenge: 2005^{1}, 2006^{2}, 2007^{3})

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV 2010

VOC

1. Category Selection

- 20 categories chosen:

1) Initial 4 categories stem from existing dataset

2) 2006: added 6 classes

3) 2007: added 10 classes

- Categories added for more generalization and finer-grained coverage

2. Image Collection

- 500,000 images retrieved from Flickr with many search terms

(search terms per category)

– **aeroplane**, airplane, plane, biplane, monoplane, aviator, bomber, hydroplane, airliner, aircraft, fighter, airport, hangar, jet, boeing, fuselage, wing, propellor, flying
– **bicycle**, bike, cycle, cyclist, pedal, tandem, saddle, wheel, cycling, ride, wheelie
– **bird**, birdie, birdwatching, nest, sea, aviary, birdcage, bird feeder, bird table
– **boat** ship, barge, ferry, canoe, boating, craft, liner, cruise, sailing, rowing, watercraft, regatta, racing, marina, beach, water, canal, river, stream, lake, yacht
– **bottle**, cork, wine, beer, champagne, ketchup, squash, soda, coke, lemonade, dinner, lunch, breakfast
– **bus**, omnibus, coach, shuttle, jitney, double-decker, motorbus, school bus, depot, terminal, station, terminus, passenger, route
– **car**, automobile, cruiser, motorcar, vehicle, hatchback, saloon, convertible, limousine, motor, race, traffic, trip, rally, city, street, road, lane, village, town, centre, shopping, downtown, suburban
– **cat**, feline, pussy, mew, kitten, tabby, tortoiseshell, ginger, stray
– **chair**, seat, rocker, rocking, deck, swivel, camp, chaise, office, studio, armchair, recliner, sitting, lounge, living room, sitting room
– **cow**, beef, heifer, moo, dairy, milk, milking, farm
– **dog**, hound, bark, kennel, heel, bitch, canine, puppy, hunter, collar, leash
– **horse**, gallop, jump, buck, equine, foal, cavalry, saddle, canter, buggy, mare, neigh, dressage, trial, racehorse, steeplechase, thoroughbred, cart, equestrian, paddock, stable, farrier
– **motorbike**, motorcycle, minibike, moped, dirt, pillion, biker, trials, motorcycling, motorcyclist, engine, motocross, scramble, sidecar, scooter, trail
– **person**, people, family, father, mother, brother, sister, aunt, uncle, grandmother, grandma, grandfather, grandpa, grandson, granddaughter, niece, nephew, cousin
– **sheep**, ram, fold, fleece, shear, baa, bleat, lamb, ewe, wool, flock
– **sofa**, chesterfield, settee, divan, couch, bolster
– **table**, dining, cafe, restaurant, kitchen, banquet, party, meal
– **potted plant**, pot plant, plant, patio, windowsill, window sill, yard, greenhouse, glass house, basket, cutting, pot, cooking, grow
– **train**, express, locomotive, freight, commuter, platform, subway, underground, steam, railway, railroad, rail, tube, underground, track, carriage, coach, metro, sleeper, railcar, buffet, cabin, level crossing
– **tv/monitor**, television, plasma, flatscreen, flat screen, lcd, crt, watching, dvd, desktop, computer, computer monitor, PC, console, game

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV 2010

VOC

1. Category Selection

- 20 categories chosen:

1) Initial 4 categories stem from existing dataset

2) 2006: added 6 classes

3) 2007: added 10 classes

- Categories added for more generalization and finer-grained coverage

→

2. Image Collection

- 500,000 images retrieved from Flickr with many search terms

→

3. Image Verification + Image Annotation

- University of Leeds annotation parties to recruit annotators annually

- Annotation guidelines & real-time assistance: detections subsequently refined to segmentations

- Post-hoc correction/feedback about the number and kind of errors made

- Annotations for each object class merged and another class added for background

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV 2010

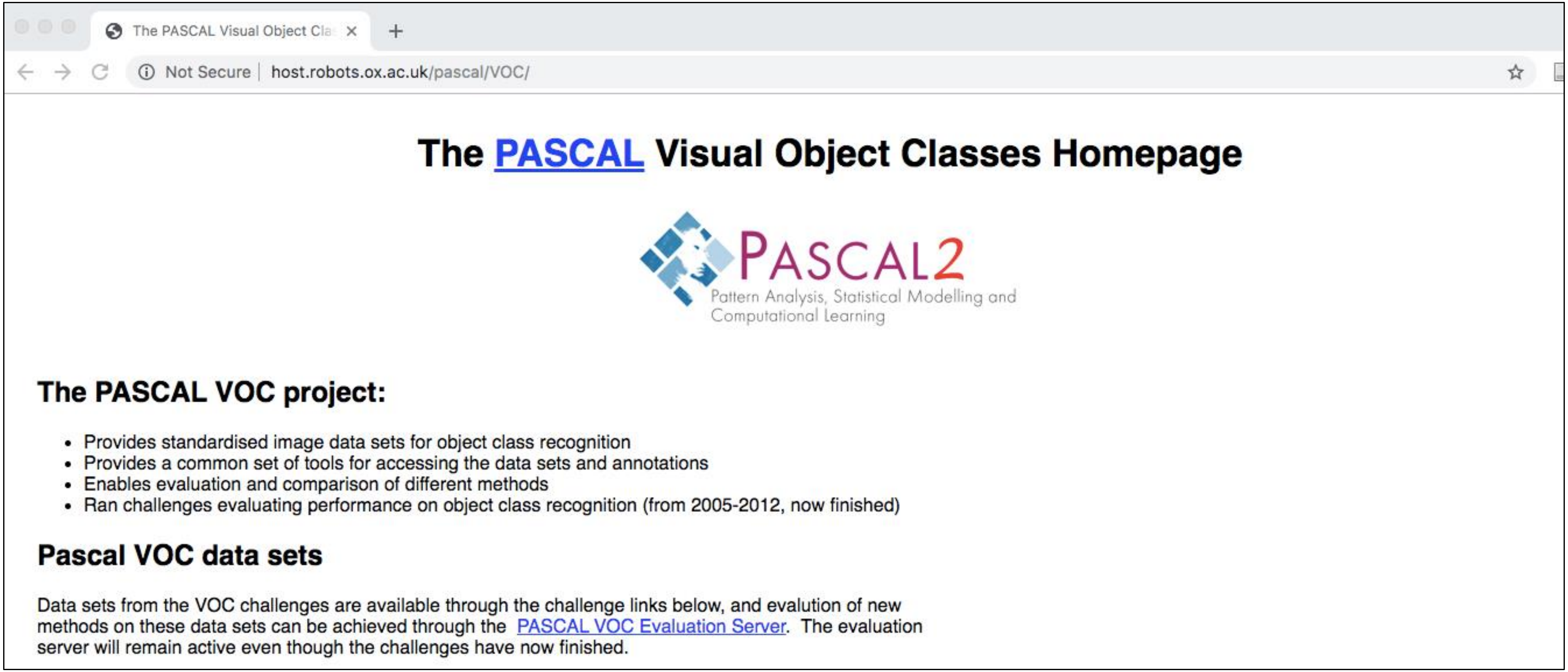
VOC: Datasets Evolved

The table below gives a brief summary of the main stages of the VOC development.

Year	Statistics	New developments	Notes
2005	Only 4 classes: bicycles, cars, motorbikes, people. Train/validation/test: 1578 images containing 2209 annotated objects.	Two competitions: classification and detection	Images were largely taken from exising public datasets, and were not as challenging as the flickr images subsequently used. This dataset is obsolete.
2006	10 classes: bicycle, bus, car, cat, cow, dog, horse, motorbike, person, sheep. Train/validation/test: 2618 images containing 4754 annotated objects.	Images from flickr and from Microsoft Research Cambridge (MSRC) dataset	The MSRC images were easier than flickr as the photos often concentrated on the object of interest. This dataset is obsolete.

http://host.robots.ox.ac.uk/pascal/VOC/

VOC Annual Workshop



The PASCAL Visual Object Cla
host.robots.ox.ac.uk/pascal/VOC/

The PASCAL Visual Object Classes Homepage

PASCAL2
Pattern Analysis, Statistical Modelling and Computational Learning

The PASCAL VOC project:

- Provides standardised image data sets for object class recognition
- Provides a common set of tools for accessing the data sets and annotations
- Enables evaluation and comparison of different methods
- Ran challenges evaluating performance on object class recognition (from 2005-2012, now finished)

Pascal VOC data sets

Data sets from the VOC challenges are available through the challenge links below, and evalution of new methods on these data sets can be achieved through the PASCAL VOC Evaluation Server. The evaluation server will remain active even though the challenges have now finished.

http://host.robots.ox.ac.uk/pascal/VOC/

VOC: Boundary Accuracy Heuristic

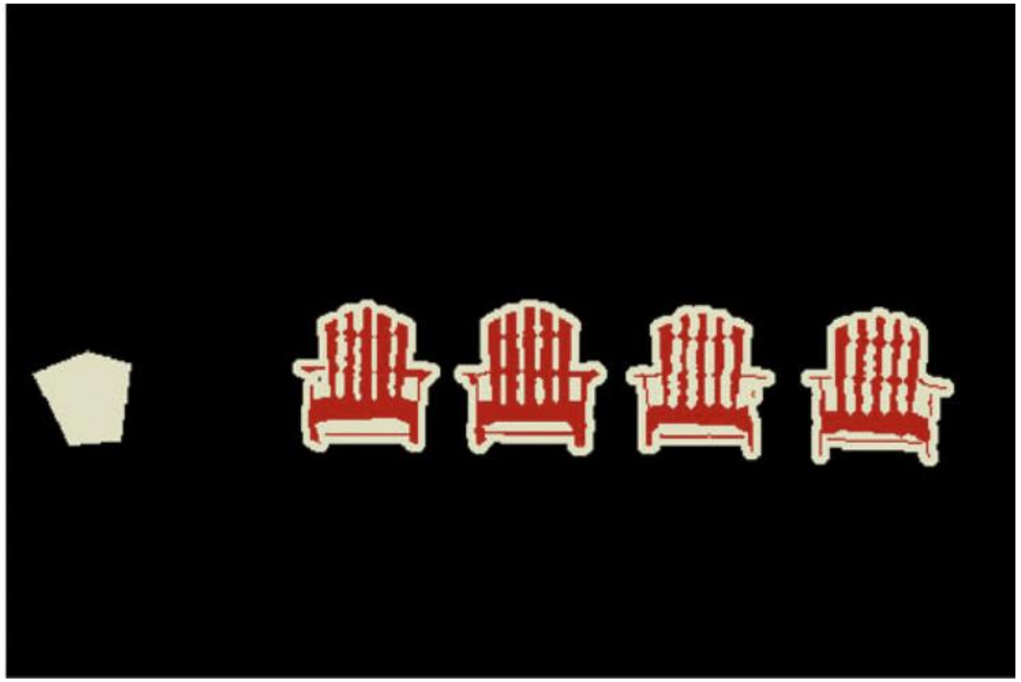
Image

Class segmentation

"To give high accuracy but to keep the annotation time short enough to provide a large image set, a border area of 5 pixels width was allowed around each object where the pixels were labelled neither object nor background."

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV 2010

What is a Limitation of Datasets Built Around Specific Categories (e.g., Objects)?

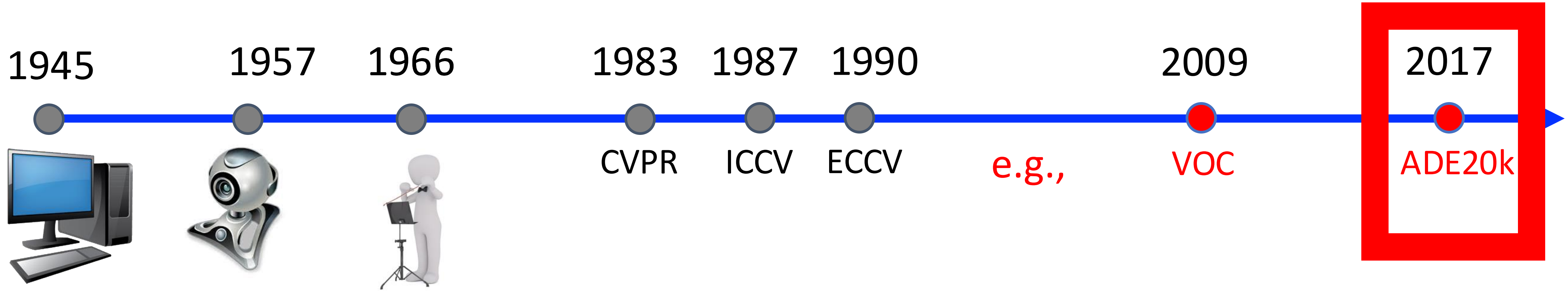


Most pixels are labeled as `background'!

Lacks knowledge anything else is in the scene, such as a house, trees or flowers!

Mark Everingham et al. The PASCAL Visual Object Classes Challenge: A Retrospective. IJCV 2015

Datasets


1945
1957
1966
1983
1987
1990
2009
2017
CVPR
ICCV
ECCV
e.g.,
VOC
ADE20k

ADE20K

1. Image Collection

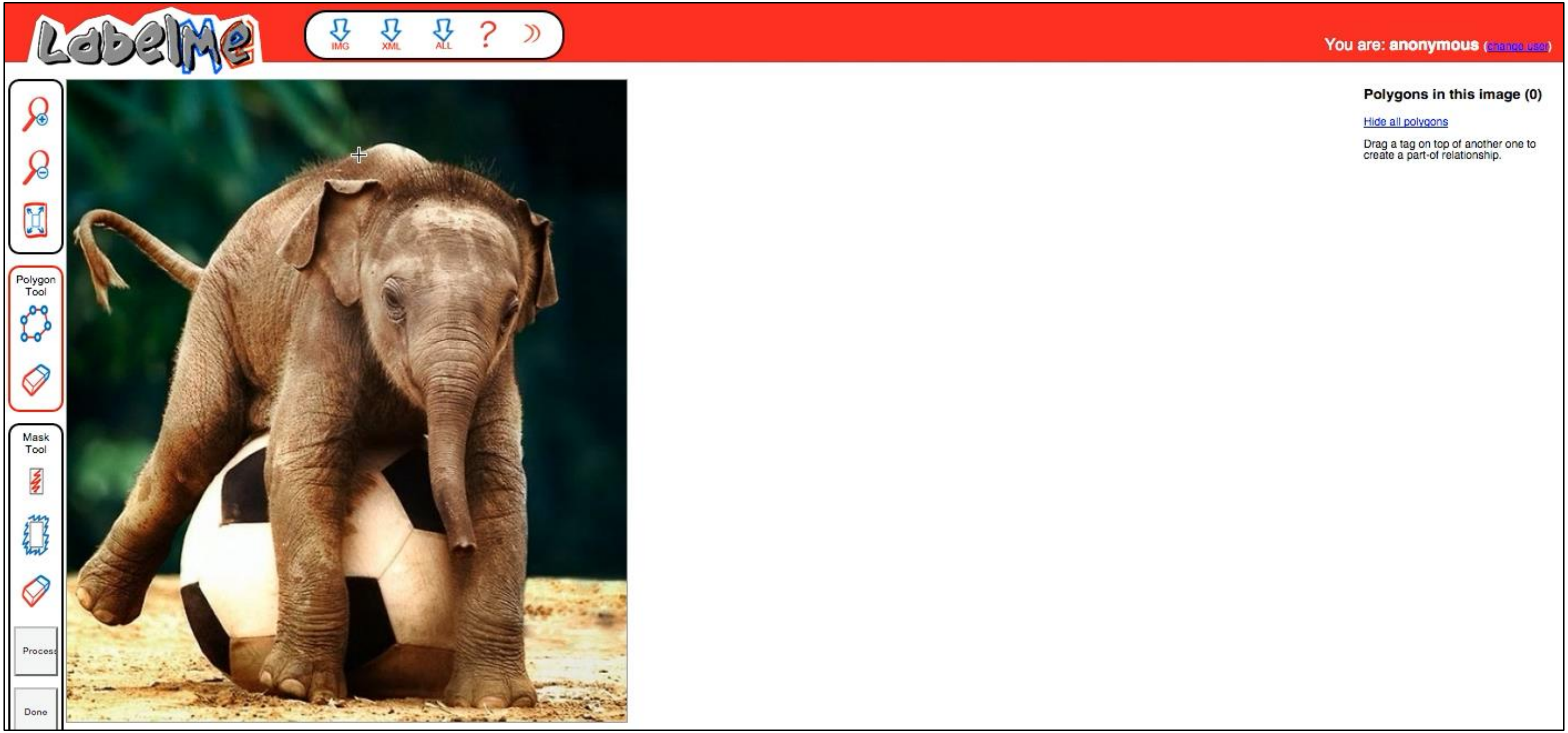
- 25,210 images collected from existing datasets (SUN, Places, and LabelMe)

- Selected to capture all scene categories defined in SUN

2. Region Localization and Category Assignment

- A single person annotated all images into three types and kept adding new categories as they were observed: (1) objects, (2) object parts, and (3) attributes (e.g., occluded)

ADE20K: User Annotation Tool



ADE20K: User Annotation Tool

- ceiling
- wall
- wall
- window (arch)
  - ↳ pane (glass)
  - ↳ figurine
- door frame
- double door
  - ↳ door
    - ↳ handle
- tray
- figurine
- refrigerator (crop)
- cabinet
  - ↳ door
    - ↳ knob
  - ↳ door
    - ↳ knob
  - ↳ door
    - ↳ knob
- jar
- cabinet
  - ↳ door
    - ↳ knob
- cabinet
  - ↳ door
    - ↳ knob
  - ↳ door
    - ↳ knob
- cabinet
  - ↳ door
    - ↳ knob
- microwave
  - ↳ door
    - ↳ window
  - ↳ button (door release)
- outlet
- pot
- sink
  - ↳ faucet
- soap dispenser
- spice rack
- coffee maker
- knife set
- knife set
- range
  - ↳ button panel
    - ↳ dial
    - ↳ dial
    - ↳ dial
    - ↳ dial
    - ↳ screen time
  - ↳ stove
    - ↳ burner
    - ↳ burner
    - ↳ burner
    - ↳ burner
  - ↳ oven
    - ↳ door
      - ↳ handle
- toaster
- blender
- pot
- box
- worktop
- cabinet
  - ↳ drawer
    - ↳ knob
- jar
- salt cellar
- worktop
- paper towels
- dishwasher
- cabinet
- cabinet
- bottle rack
- napkin rack
- kitchen island
- glass (wine)
- glass (wine)
- coasters
- bowl
- bowl
- trash can
- dog dish
- dog dish
- chair
  - ↳ back
  - ↳ seat (fabric)
  - ↳ leg
  - ↳ leg
  - ↳ leg
- chair
  - ↳ back
  - ↳ seat (fabric)
  - ↳ leg
  - ↳ leg
  - ↳ leg
- chair
  - ↳ back
  - ↳ seat (fabric)
  - ↳ leg
  - ↳ leg
  - ↳ leg
  - ↳ leg
- side table (crop)
- rug
- sofa (crop)
- cushion
- cushion
- cushion
- floor (tile)
- carpet
- bowl
- light switch
- picture (map)

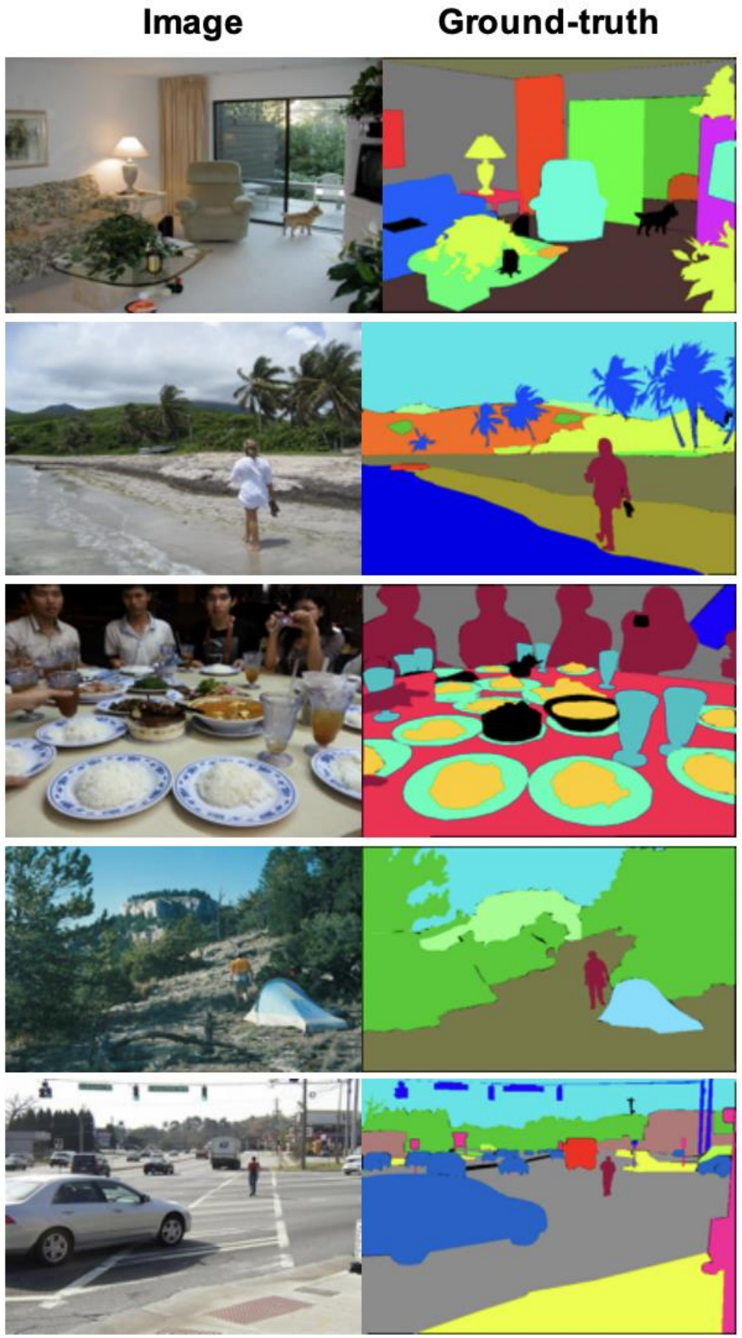
Bolei Zhou et al. Scene Parsing through ADE20K Dataset. CVPR 2017

ADE20K

- Includes:

- “things”: objects that can easily be labeled; e.g., person, chair

- “stuff”: objects with no clear boundaries; e.g., sky, grass

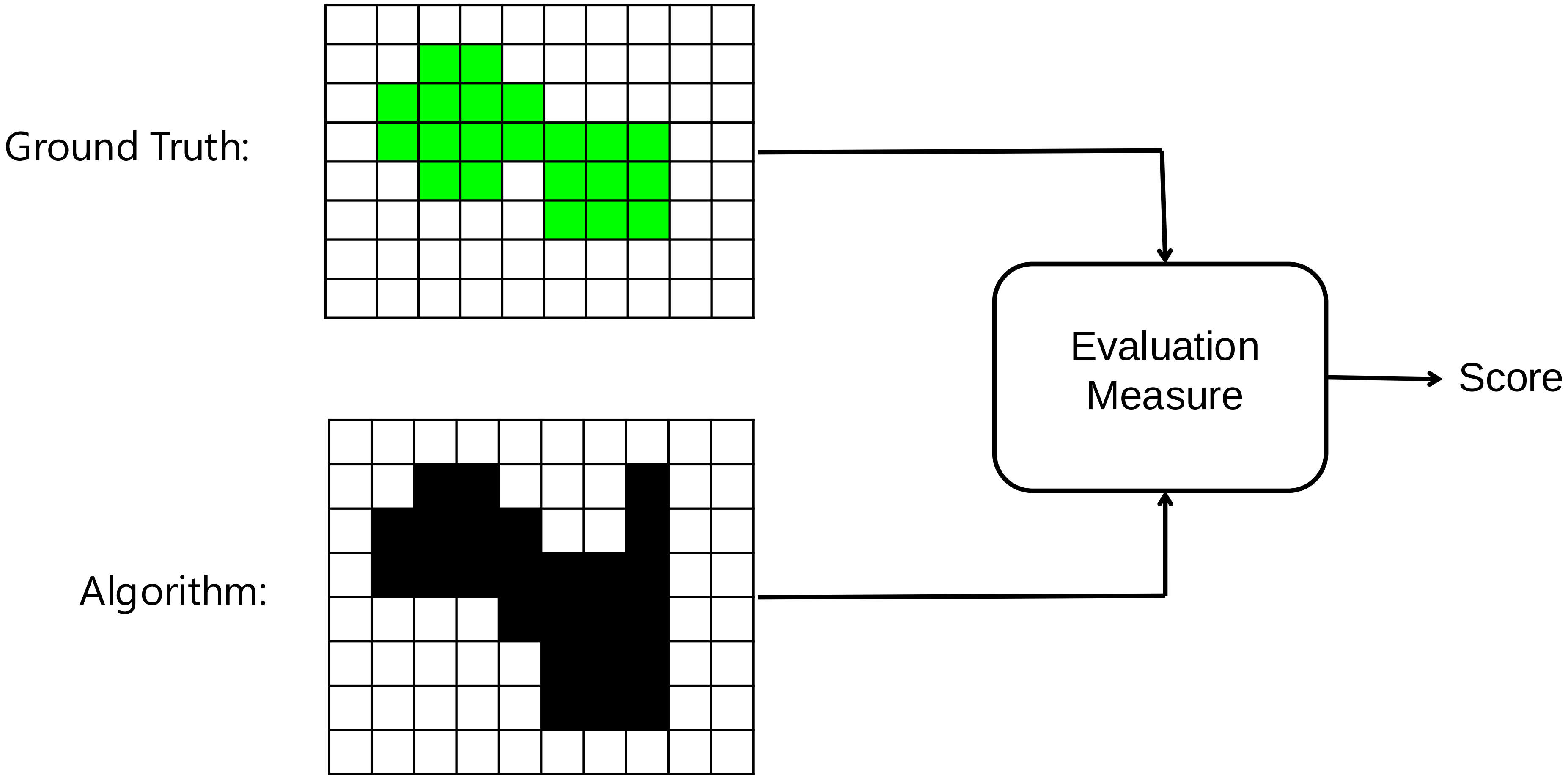


Bolei Zhou et al. Scene Parsing through ADE20K Dataset. CVPR 2017

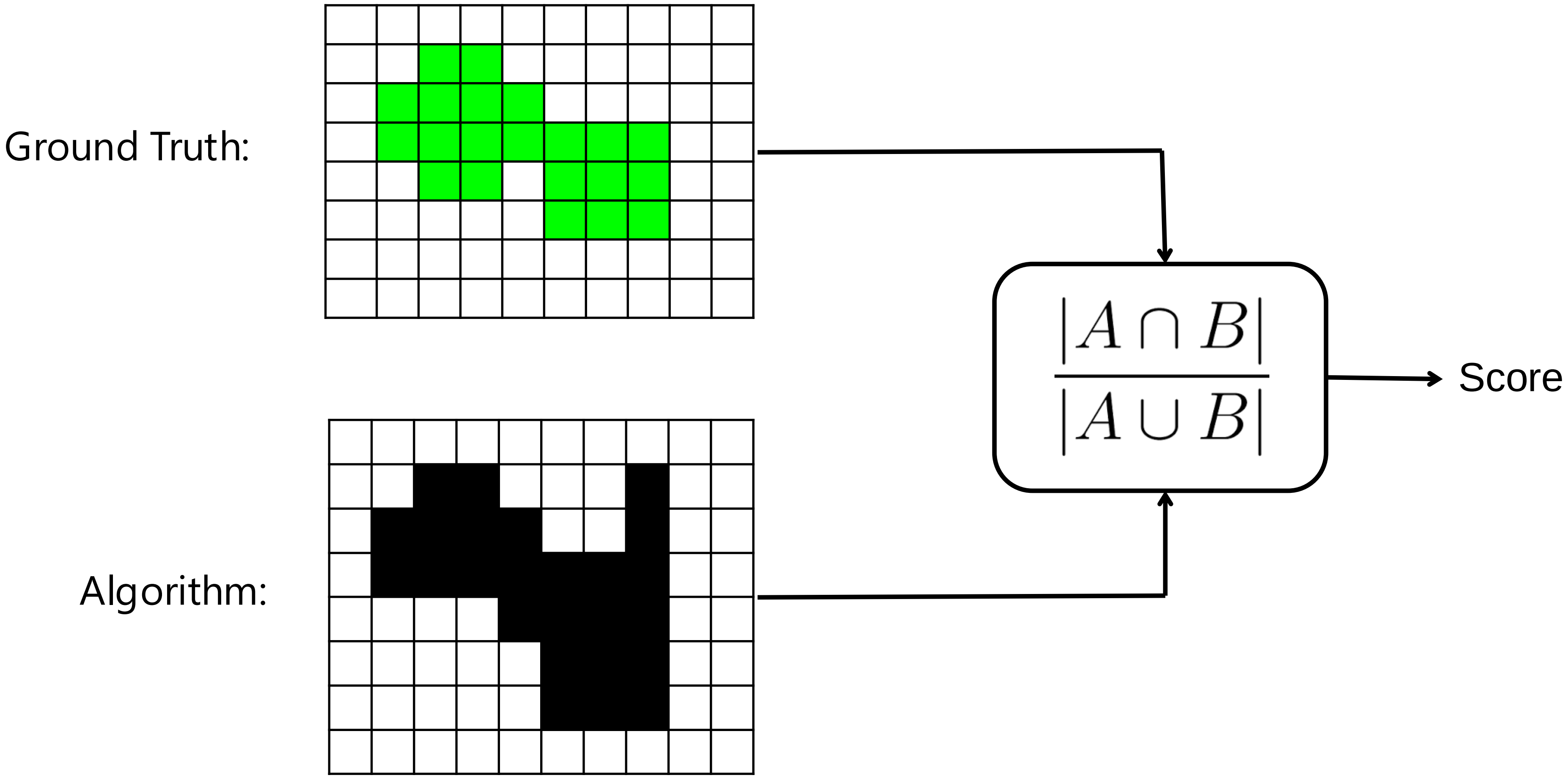
Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

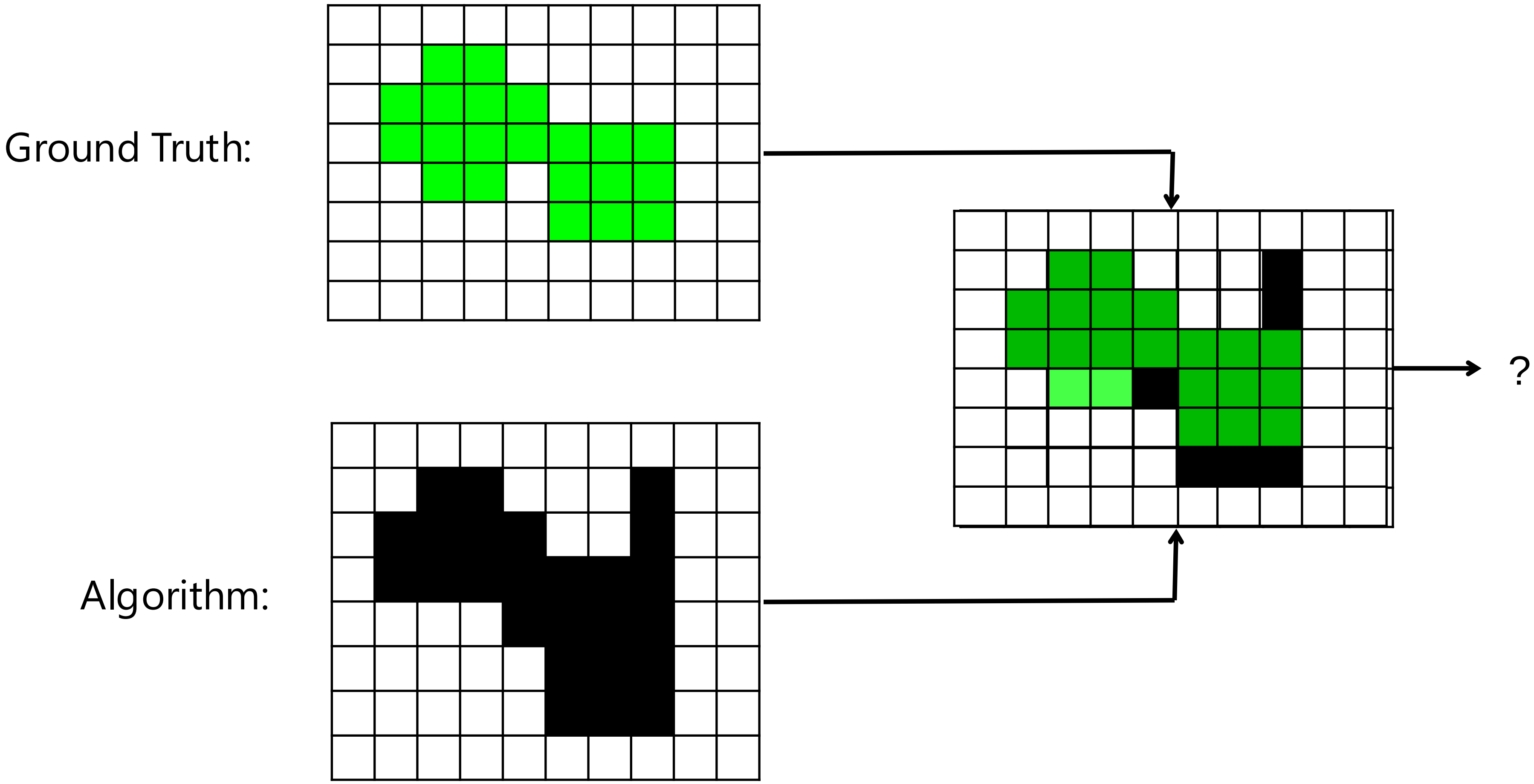
Evaluation Metric



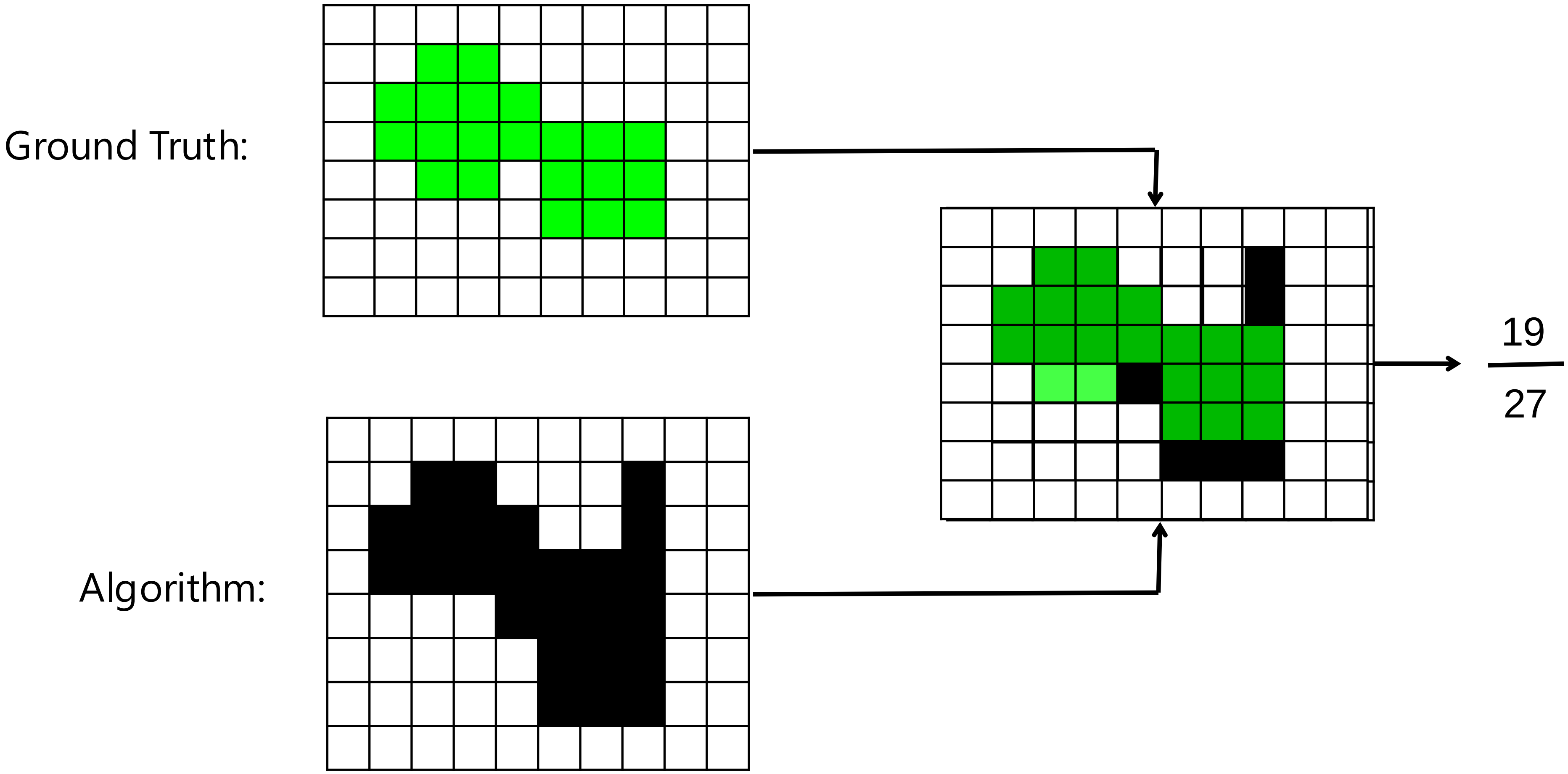
Recall: IoU Metric



Recall: IoU Metric



Recall: IoU Metric



Mean IoU (mIoU)

- Mean IoU score over all categories

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Why Fully Convolutional Network?

Named after the proposed technique that excludes fully connected layers:

> Jonathon Long, Evan Shelhamer, and Trevor Darrell. “Fully Convolutional Networks for Semantic Segmentation.” CVPR 2015.

Key Novelties of Fully Convolutional Networks

First work for pixelwise prediction to:

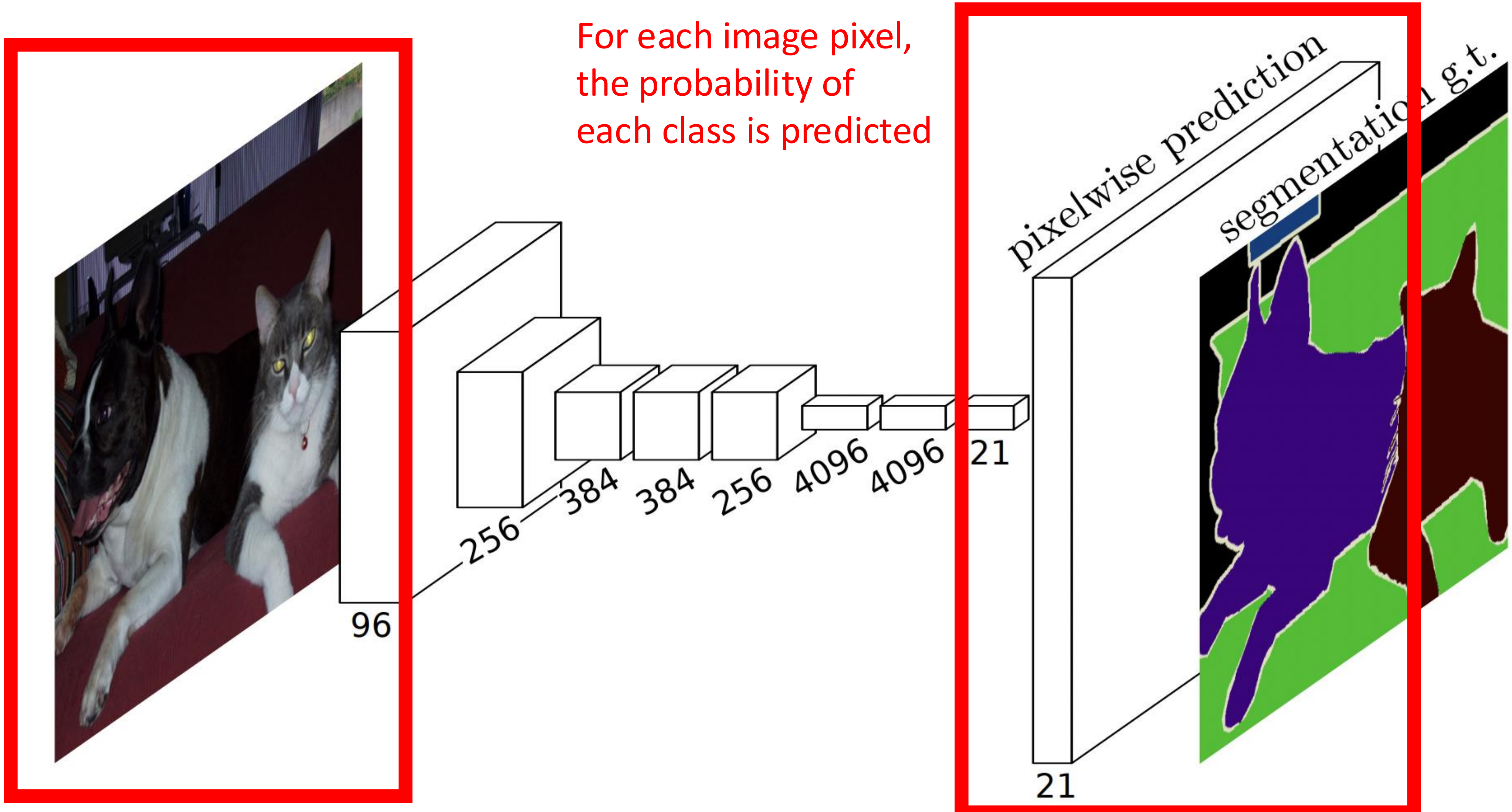
1. Train fully convolutional networks end-to-end

2. Use supervised pre-training (recall, ViT benefited from this as well)

Architecture

Input: RGB image of ANY size

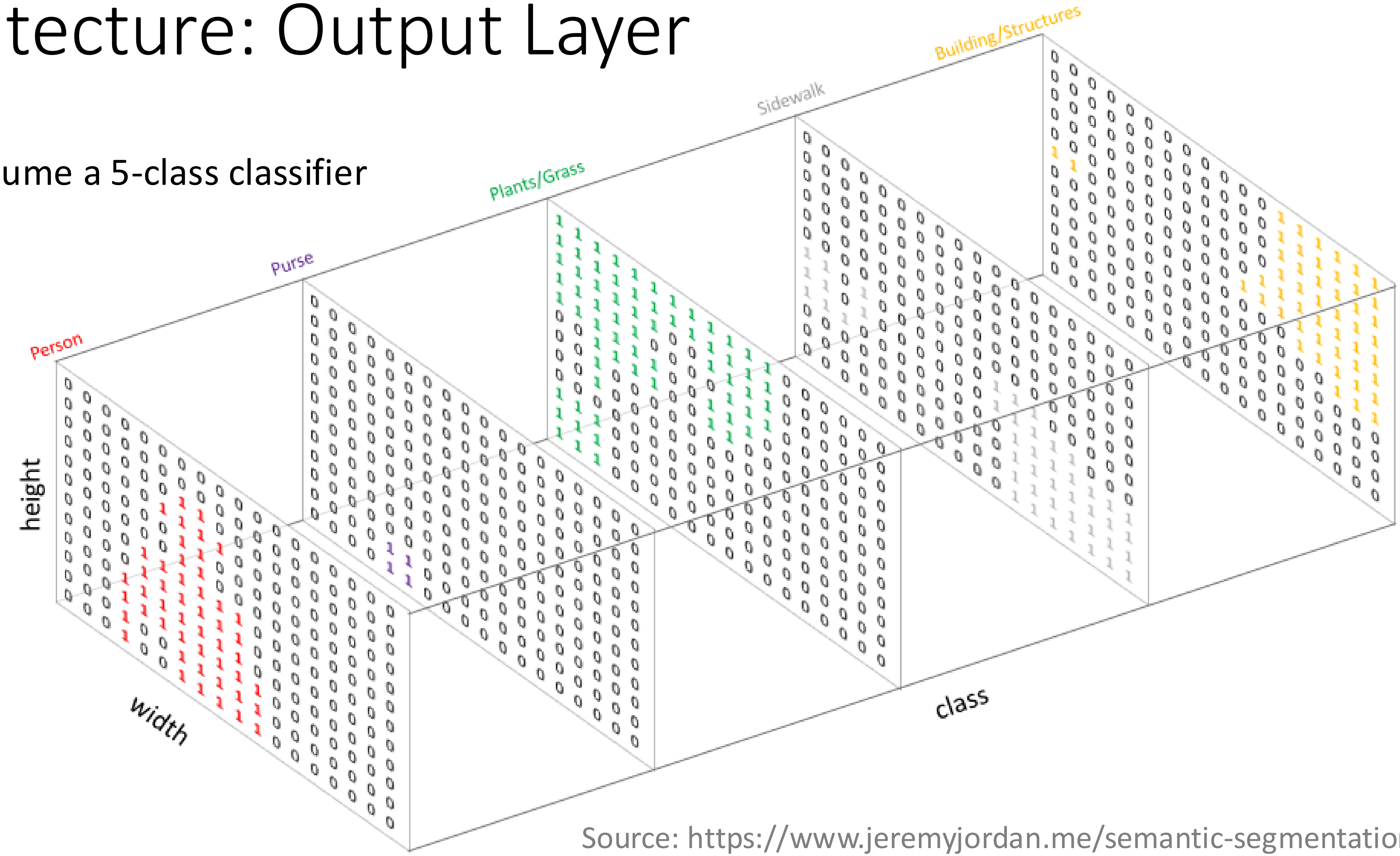
Output: Image of same size as input



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture: Output Layer

- e.g., assume a 5-class classifier



Source: https://www.jeremyjordan.me/semantic-segmentation/

Architecture: Output Layer

- e.g., assume a 5-class classifier; output 1-hot encoding collapsed into single mask image

0: Background/Unknown
1: Person
2: Purse
3: Plants/Grass
4: Sidewalk
5: Building/Structures

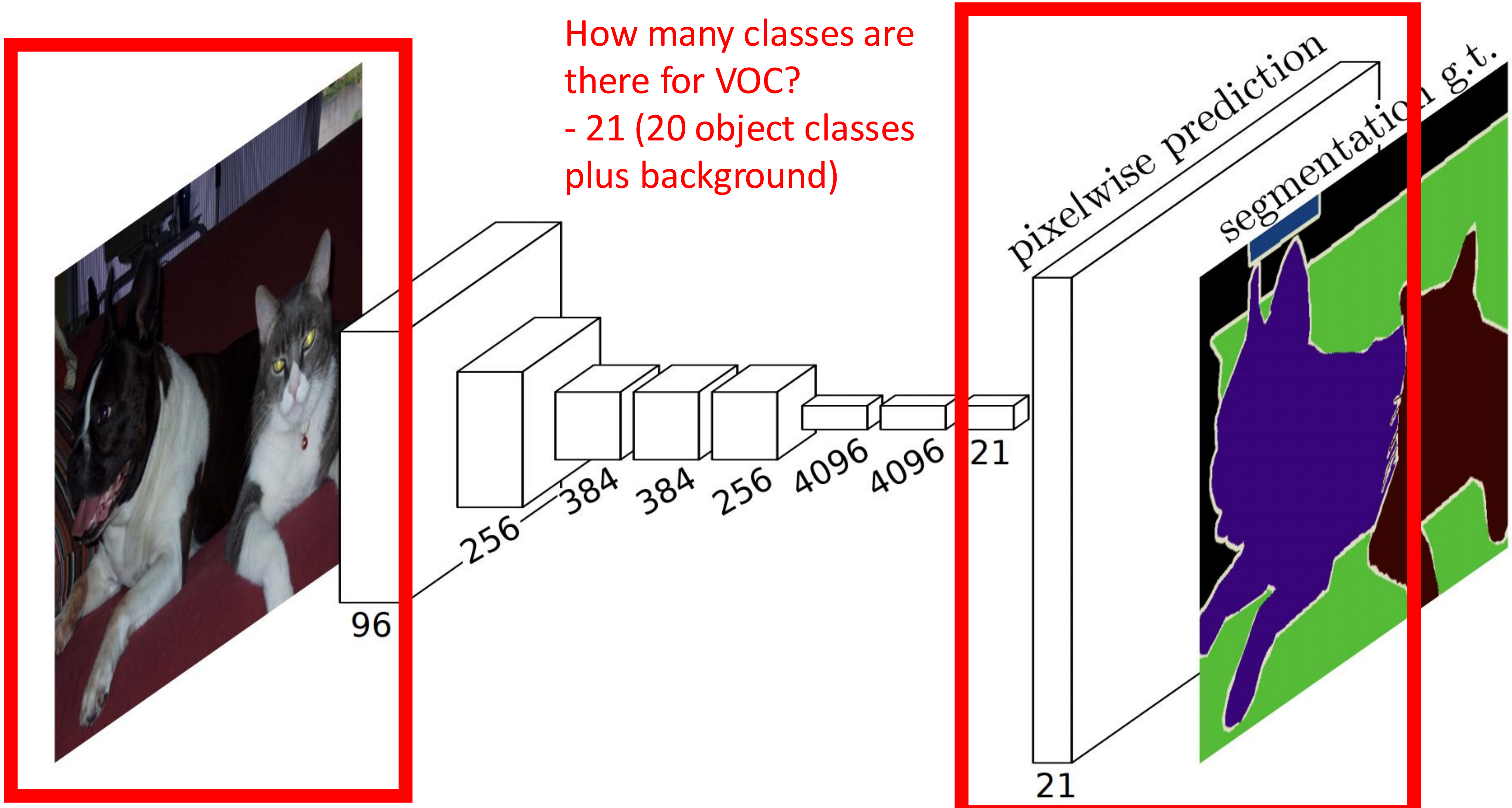
Source: https://www.jeremyjordan.me/semantic-segmentation/

Input: RGB image of ANY size

Output: Image of same size as input

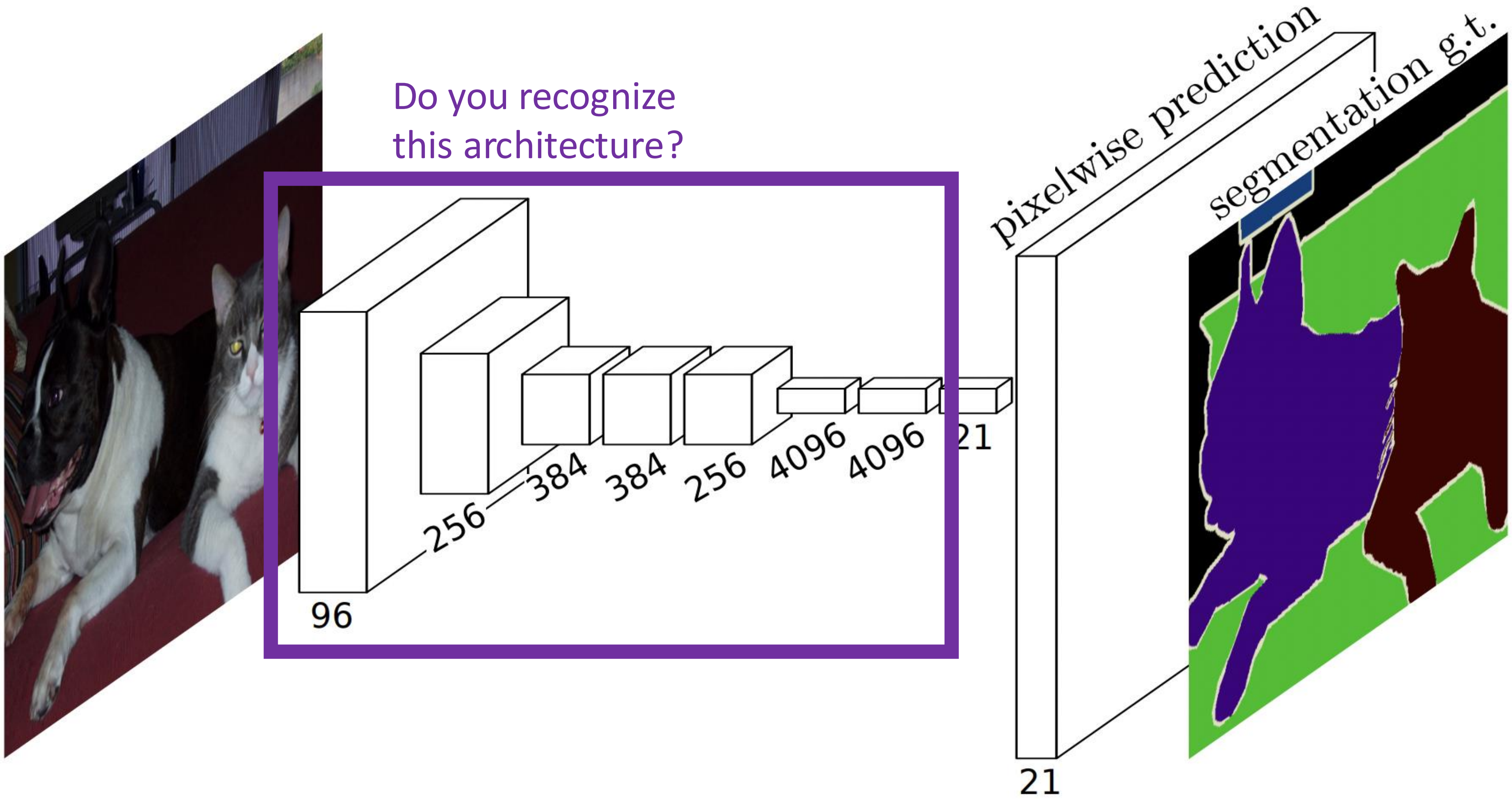
Architecture

How many classes are there for VOC?
- 21 (20 object classes plus background)



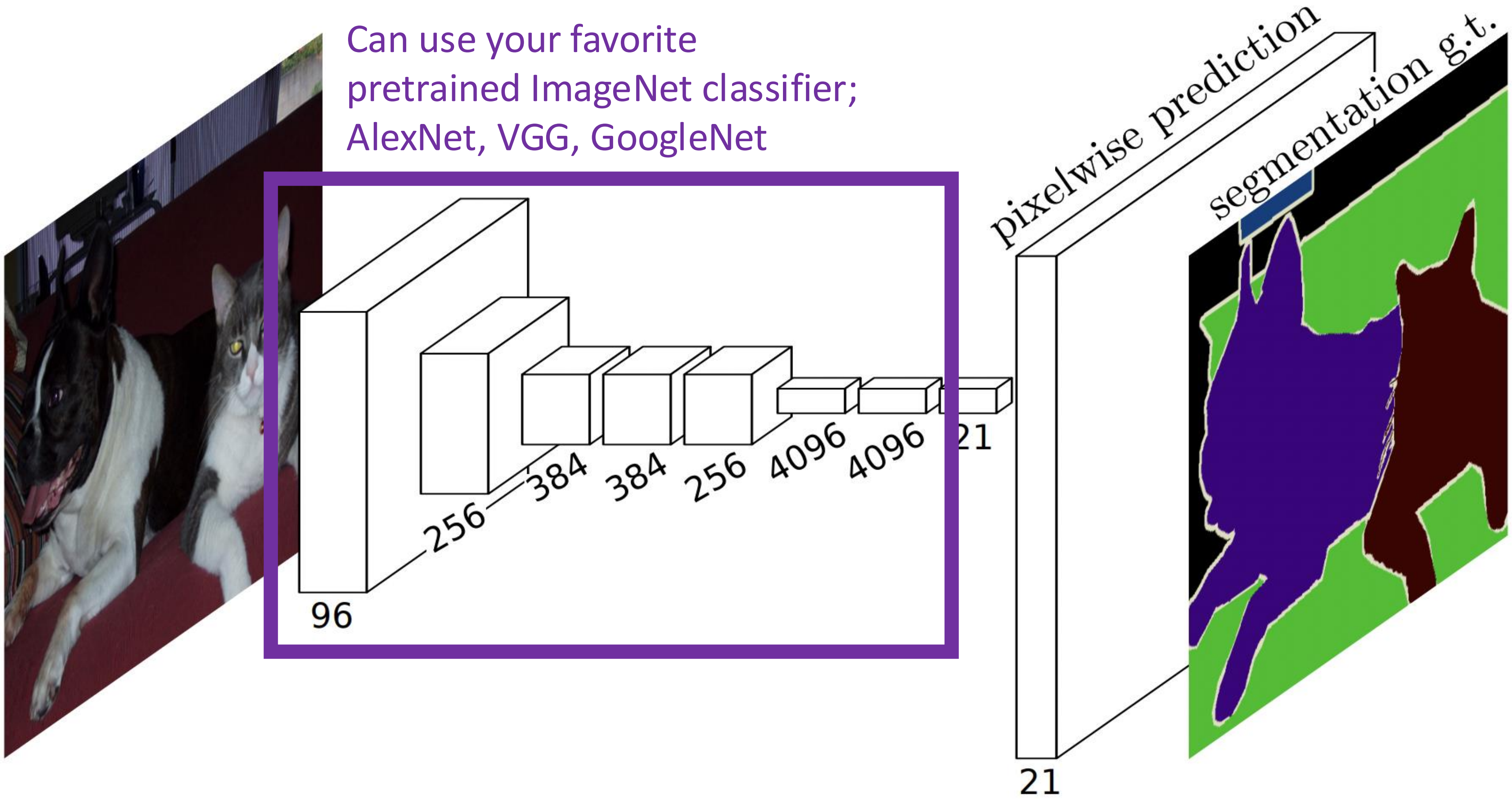
Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture

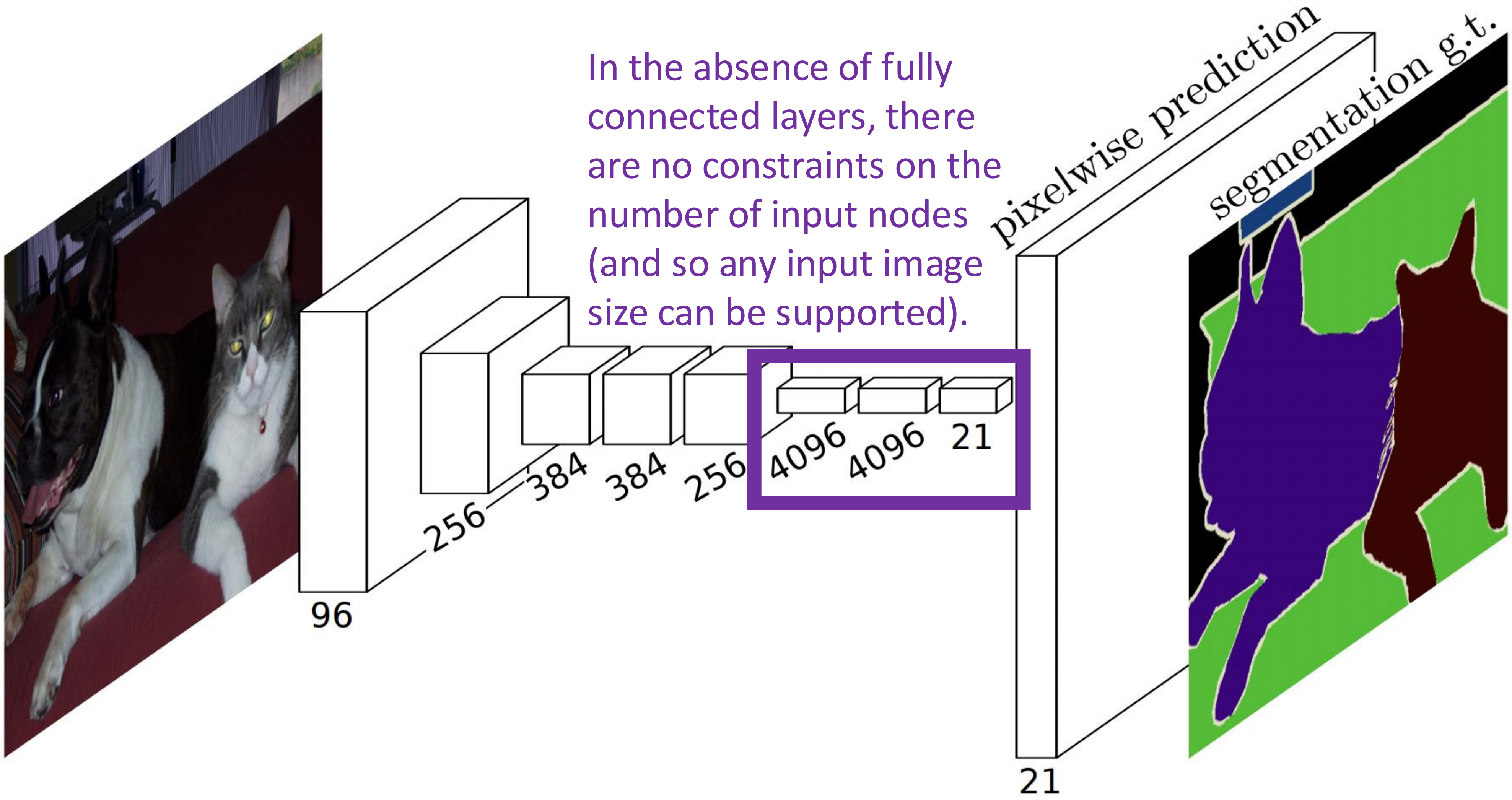


Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture

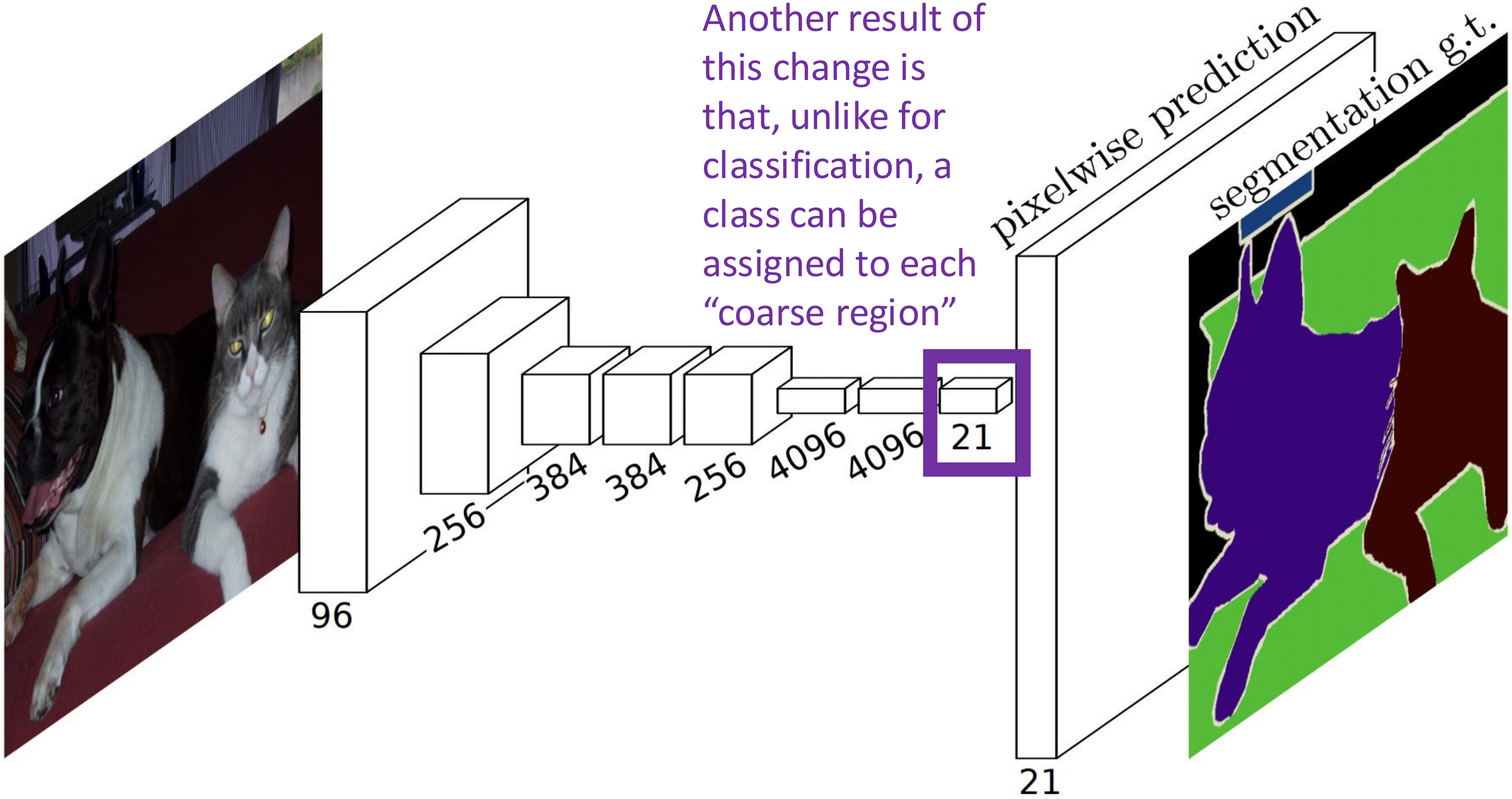
To make architecture fully convolutional, fully connected layers are converted to convolutional layers.

In the absence of fully connected layers, there are no constraints on the number of input nodes (and so any input image size can be supported).



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture

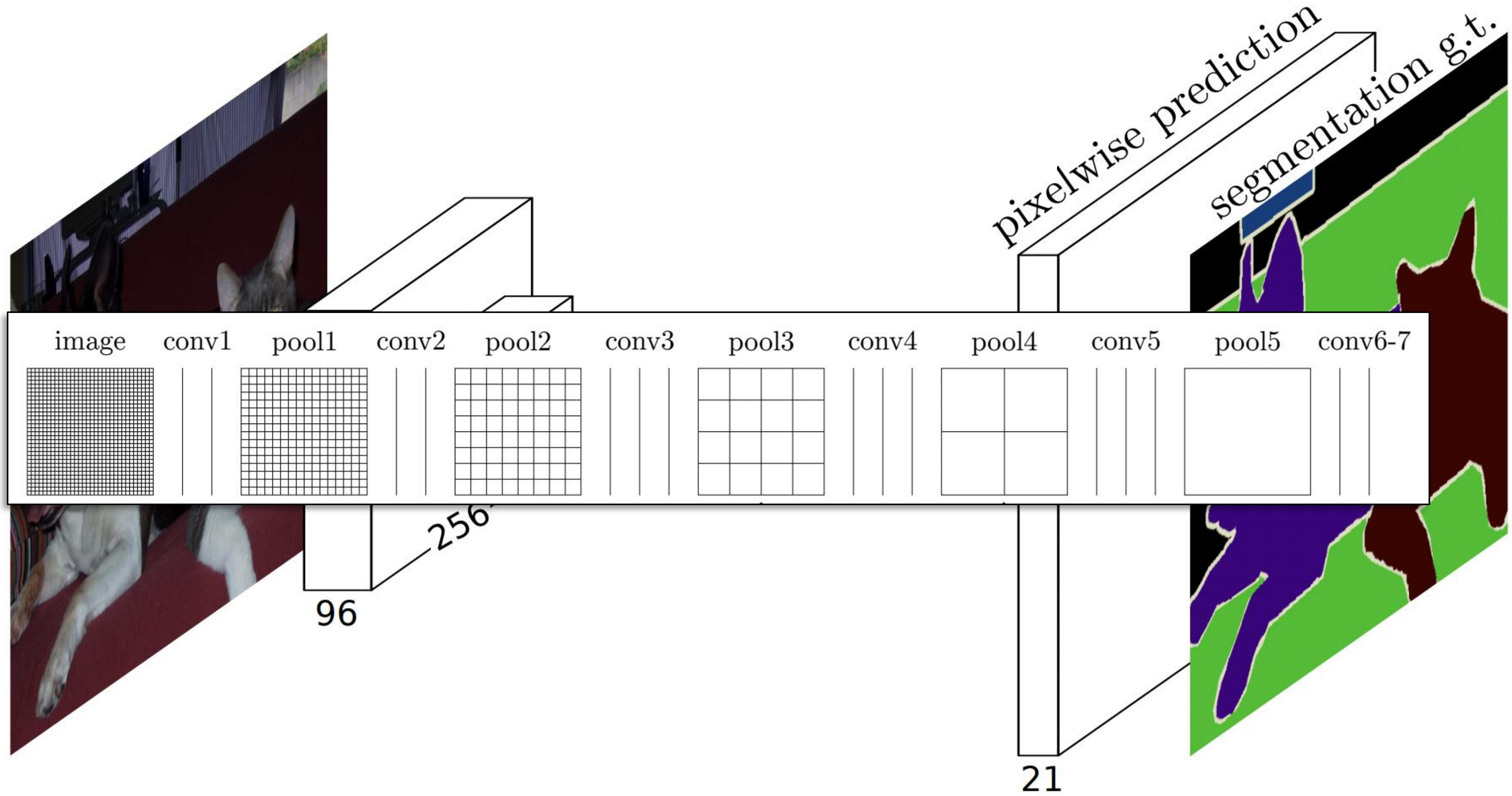


Another result of this change is that, unlike for classification, a class can be assigned to each "coarse region"

Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture: Coarse Region Classification (Recall Intuition)

Using VGG16 instead:

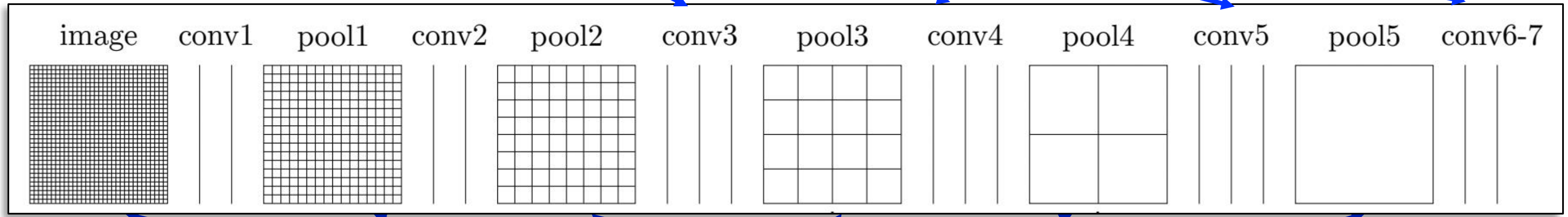


Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture: Coarse Region Classification (Recall Intuition)

Each line represents a convolutional layer

Using VGG16 instead:

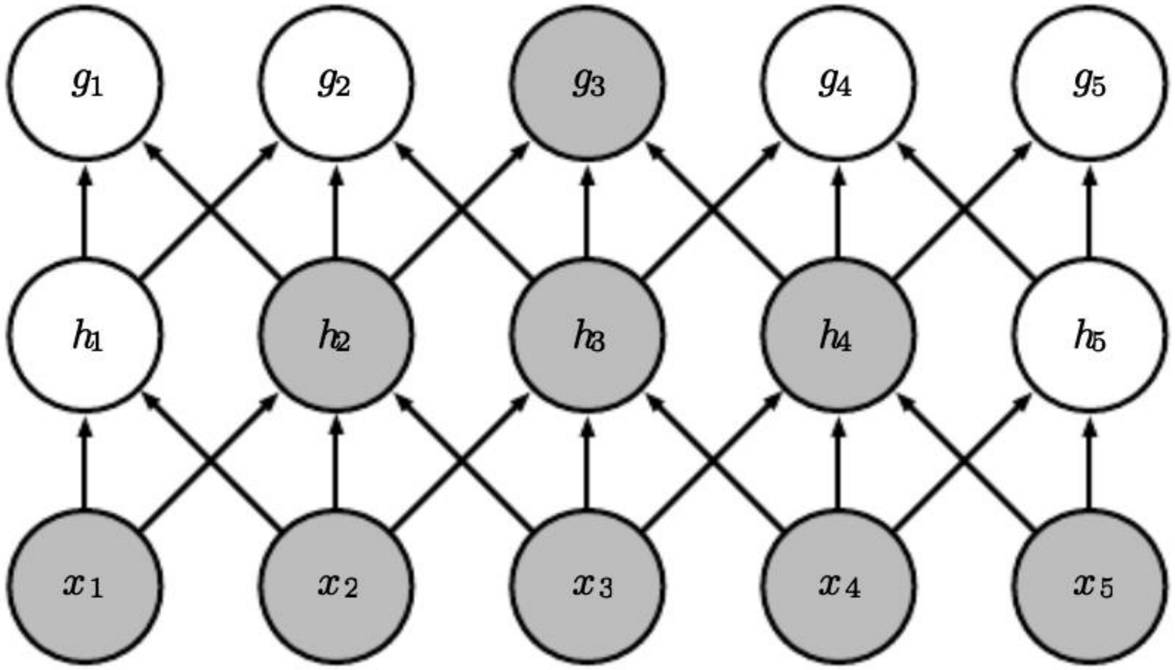


Grids reflect relative spatial coarseness at each layer

Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

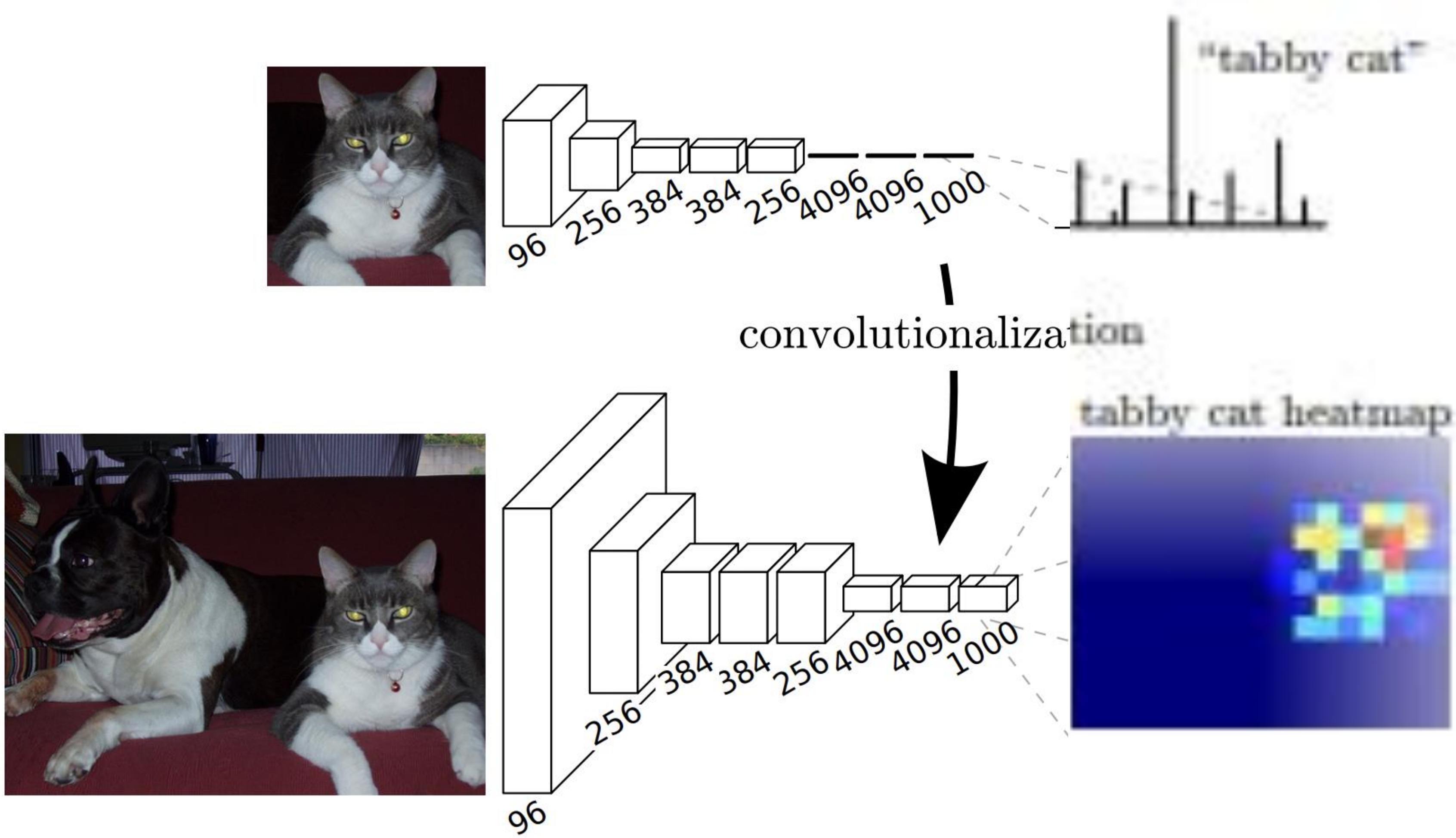
Architecture: Coarse Region Classification (Recall Intuition)

Stacking many convolutional layers leads to learning patterns in increasingly **larger regions of the input (e.g., pixel) space.**



https://www.deeplearningbook.org/contents/convnets.html

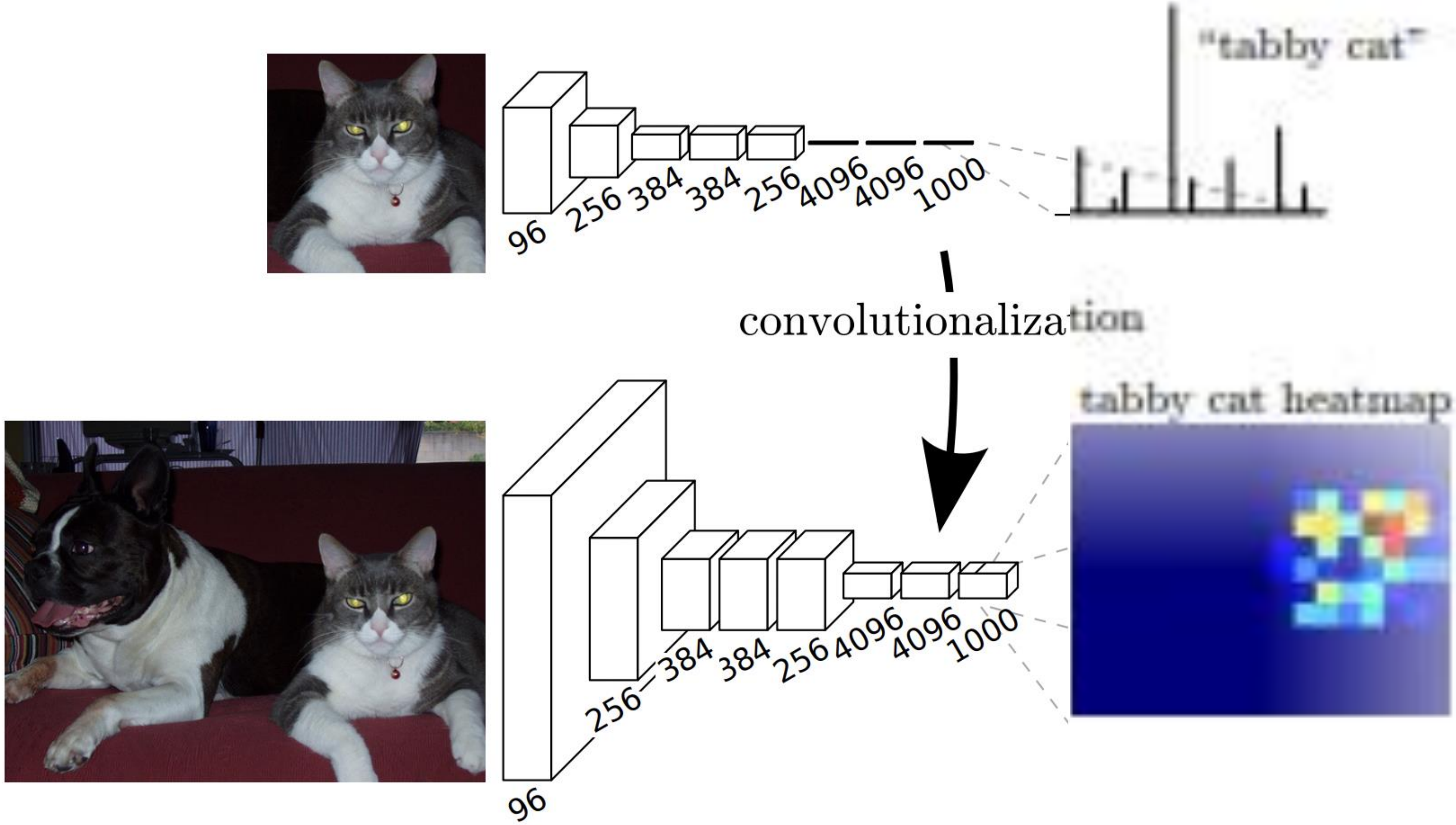
Architecture: Fully vs Convolution Layers



Each slice indicates the likelihood each pixel in the coarse region belongs to the class identified by the filter

Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

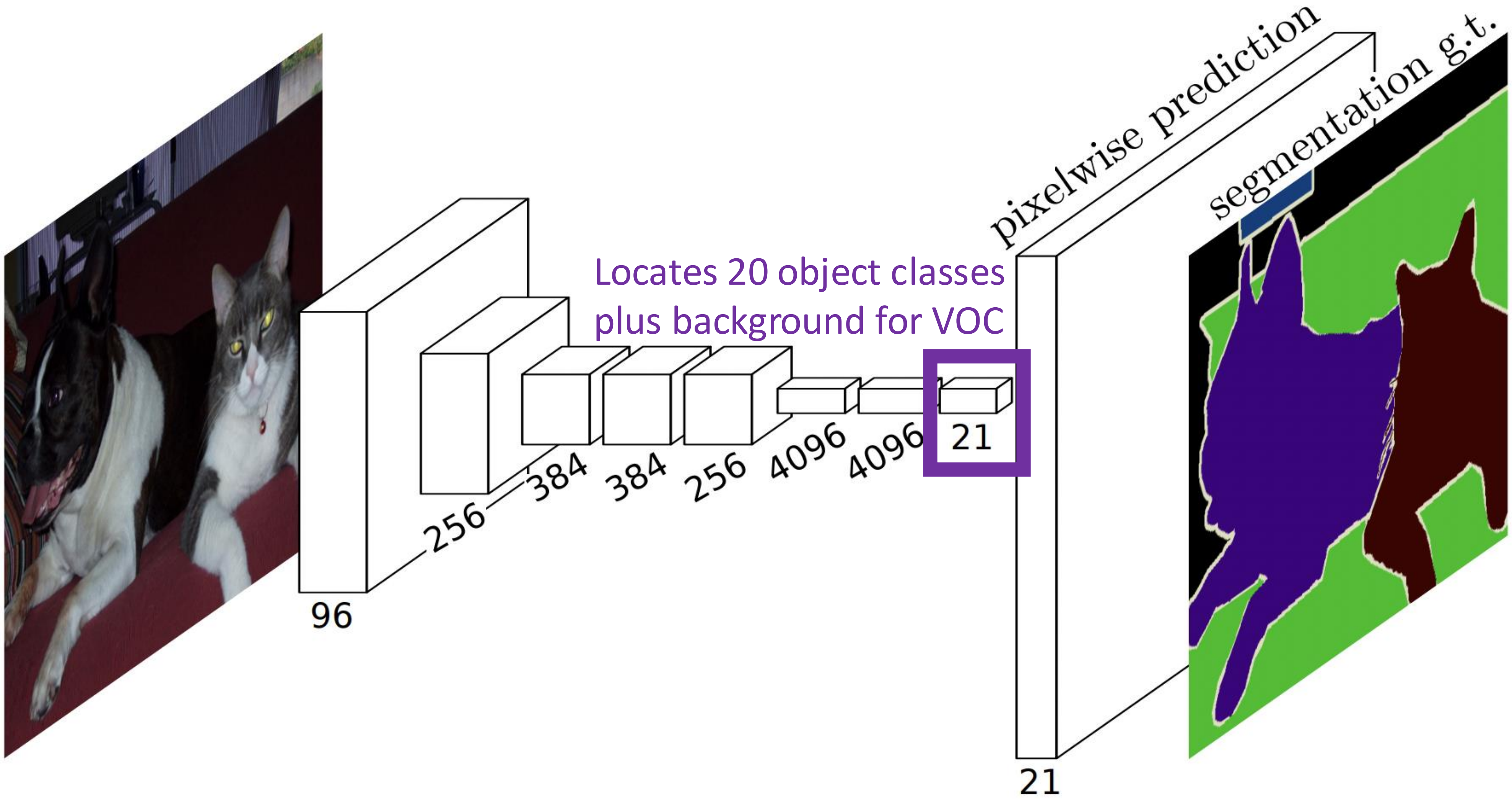
Architecture: Fully vs Convolution Layers



If convolutionizing ImageNet trained classifiers, how many classes would be predicted for each coarse region?

Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

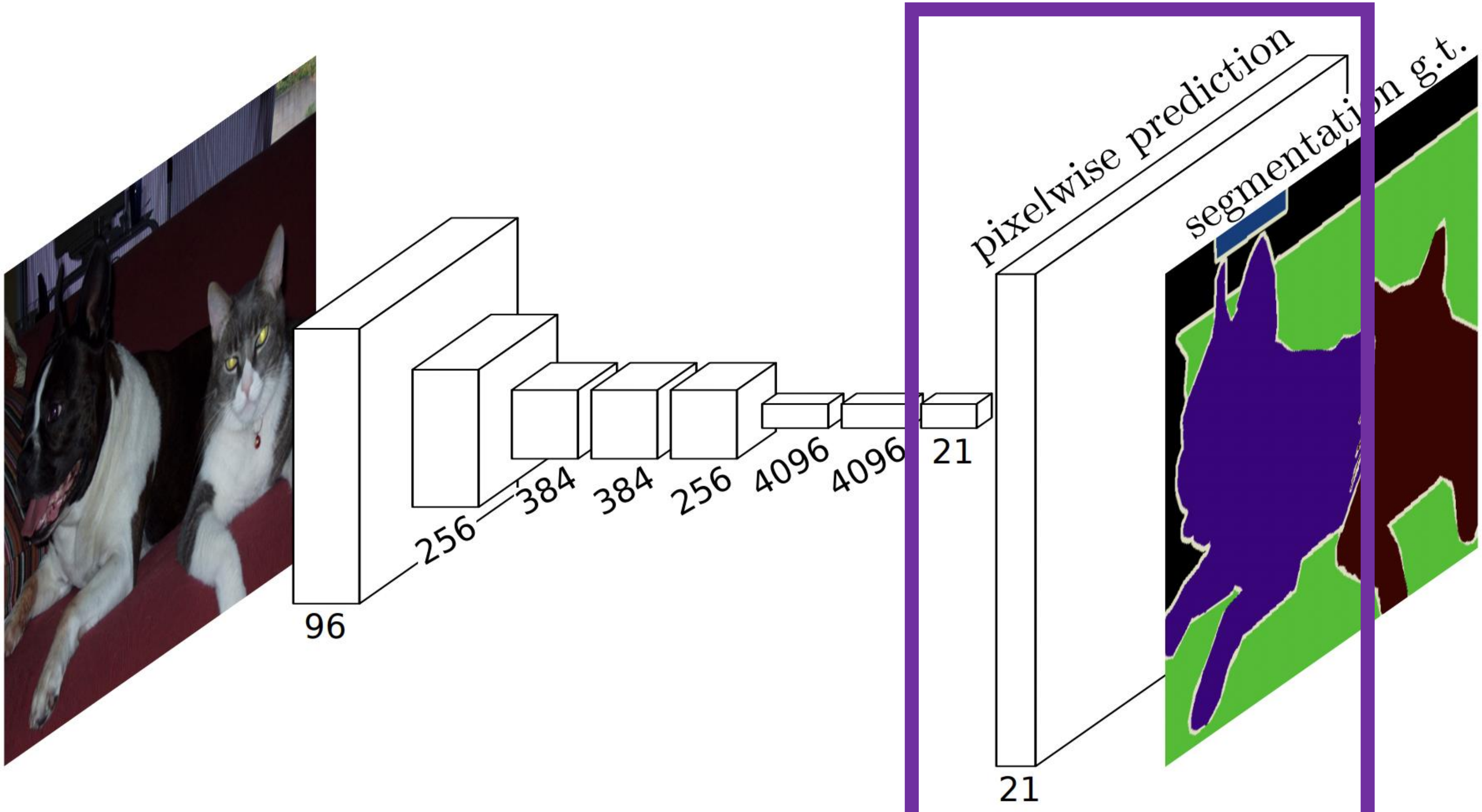
Architecture: Coarse Region Classification



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

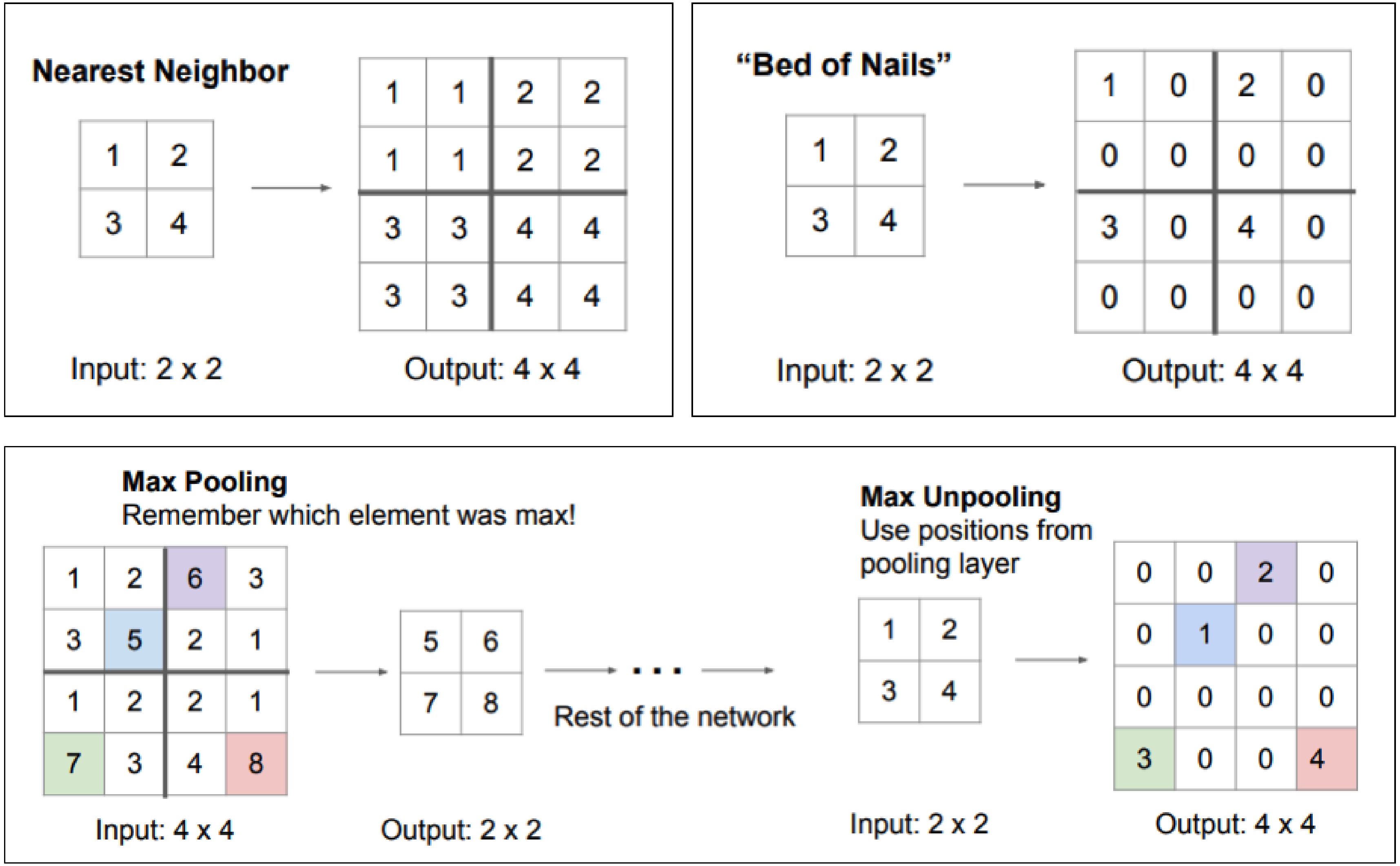
Architecture

Challenge: how to decode from coarse region classifications to per pixel classification?



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

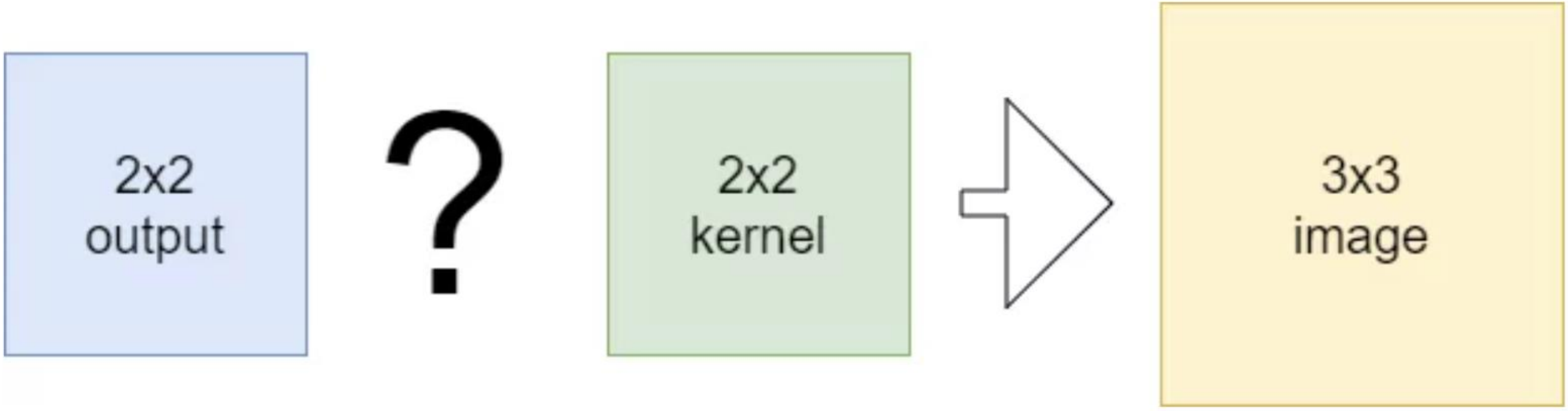
Architecture: Upsampling (Many Approaches)



Source: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

Architecture: Upsampling (Transposed Convolutional Layer)

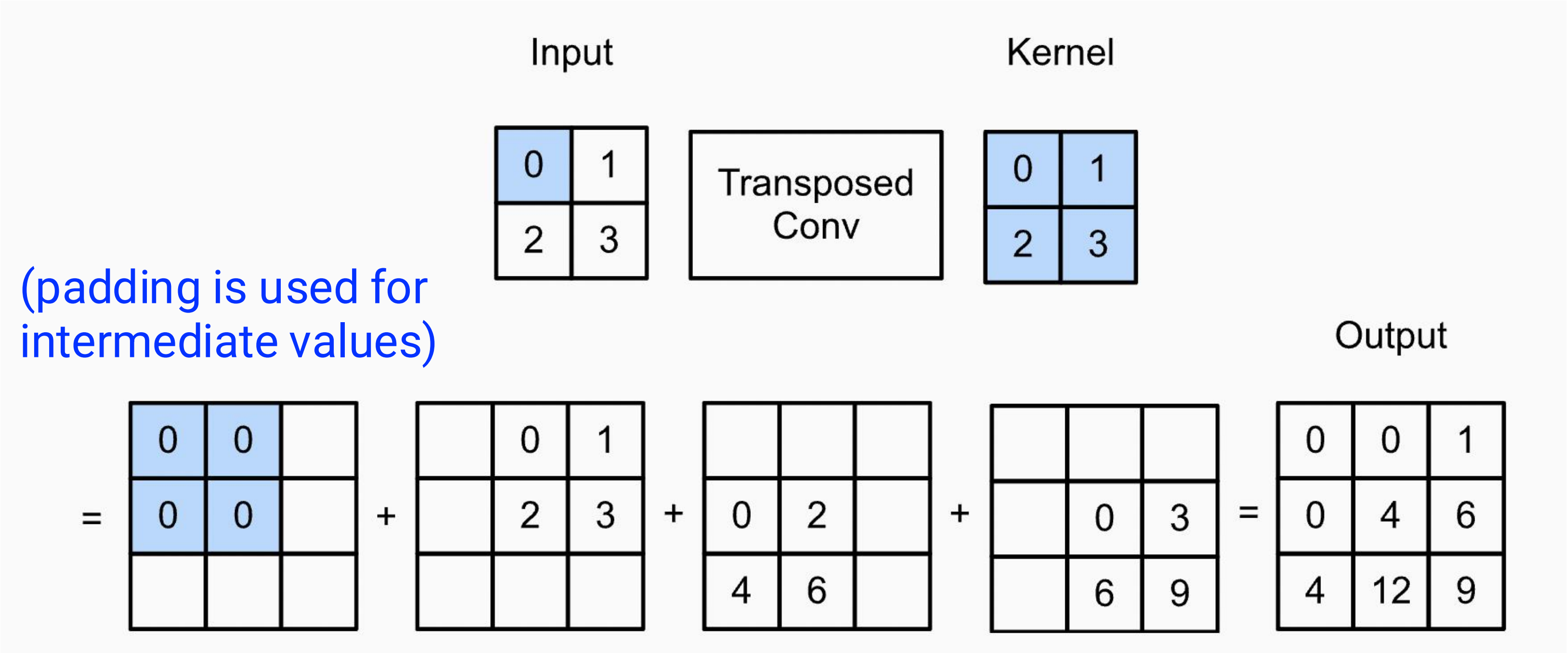
- **Idea**: learn convolutional filters to upsample the coarse image with fractional sized steps
- Also called “fractional convolutional layer”, “backward convolution”, and, incorrectly, ”deconvolution layer”, there are many implementations



https://www.machinecurve.com/index.php/2019/09/29/understanding-transposed-convolutions/#the-goal-reconstructing-the-original-input

Architecture: Upsampling (Transposed Convolutional Layer)

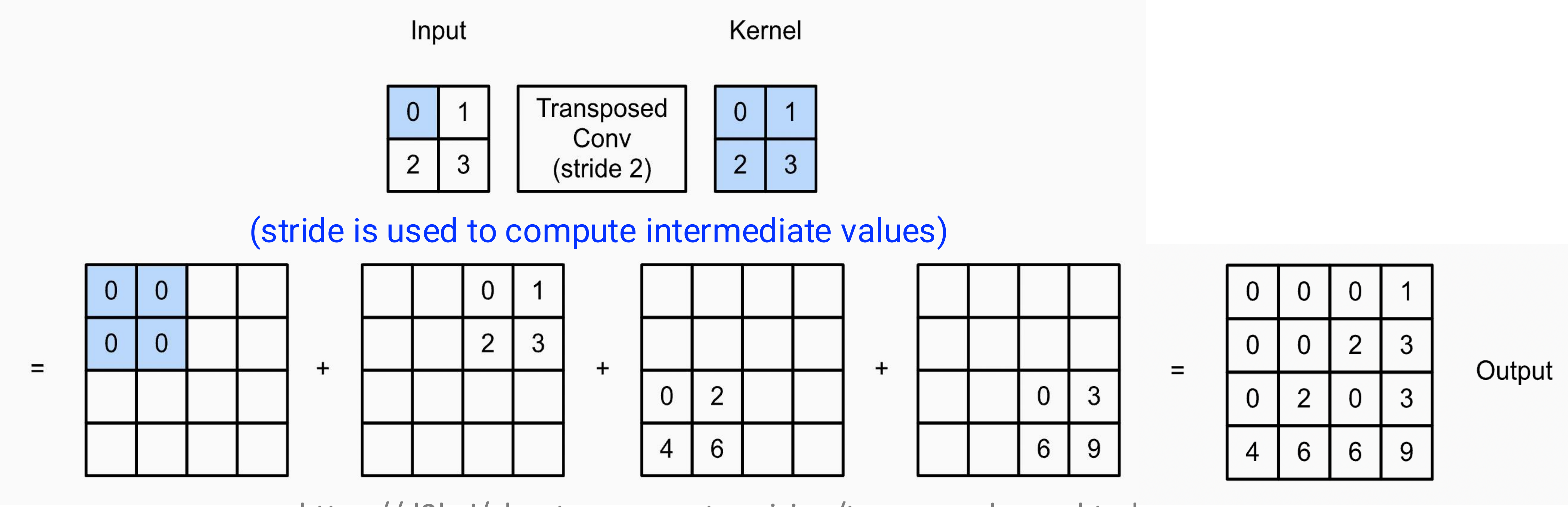
- **Idea**: learn convolutional filters to upsample the coarse image with fractional sized steps
- Also called “fractional convolutional layer”, “backward convolution”, and, incorrectly, ”deconvolution layer”, there are many implementations



https://d2l.ai/chapter_computer-vision/transposed-conv.html

Architecture: Upsampling (Transposed Convolutional Layer)

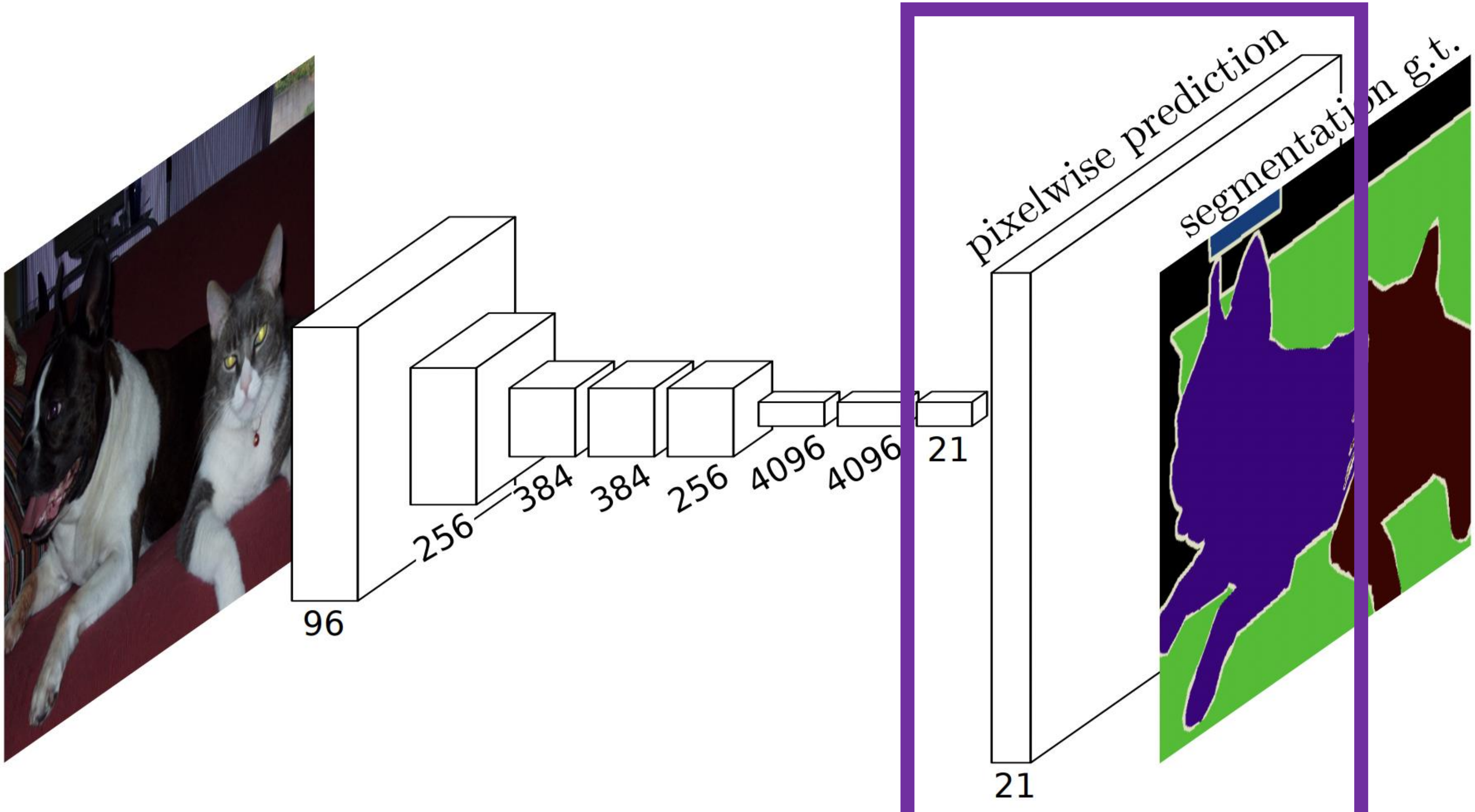
- **Idea**: learn convolutional filters to upsample the coarse image with fractional sized steps
- Also called "fractional convolutional layer", "backward convolution", and, incorrectly, "deconvolution layer", there are many implementations



https://d2l.ai/chapter_computer-vision/transposed-conv.html

Architecture

Next challenge: how to decode a **highly detailed** per pixel classification from the coarse region classifications?

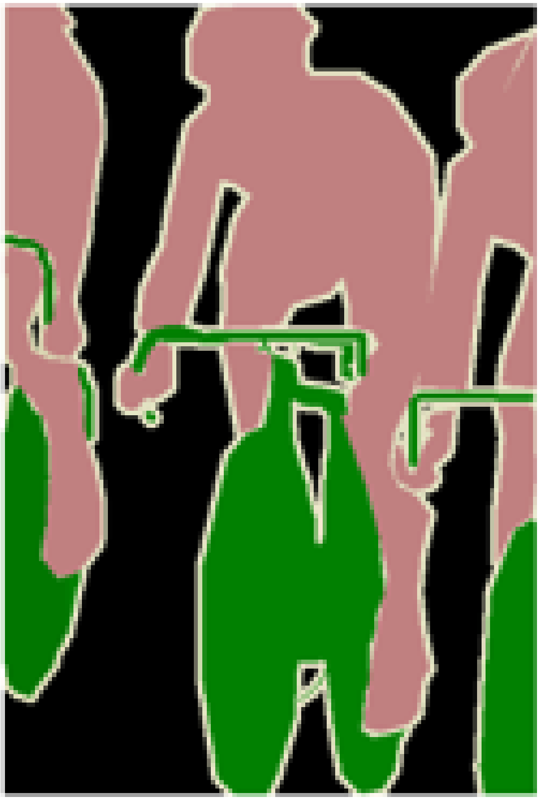


Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

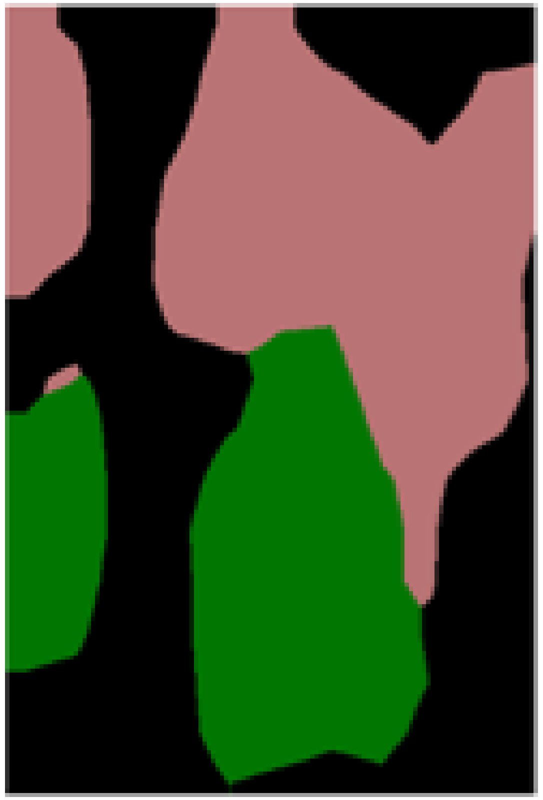
Architecture: Results

Next challenge: how to decode a **highly detailed** per pixel classification from the coarse region classifications?

Ground truth target

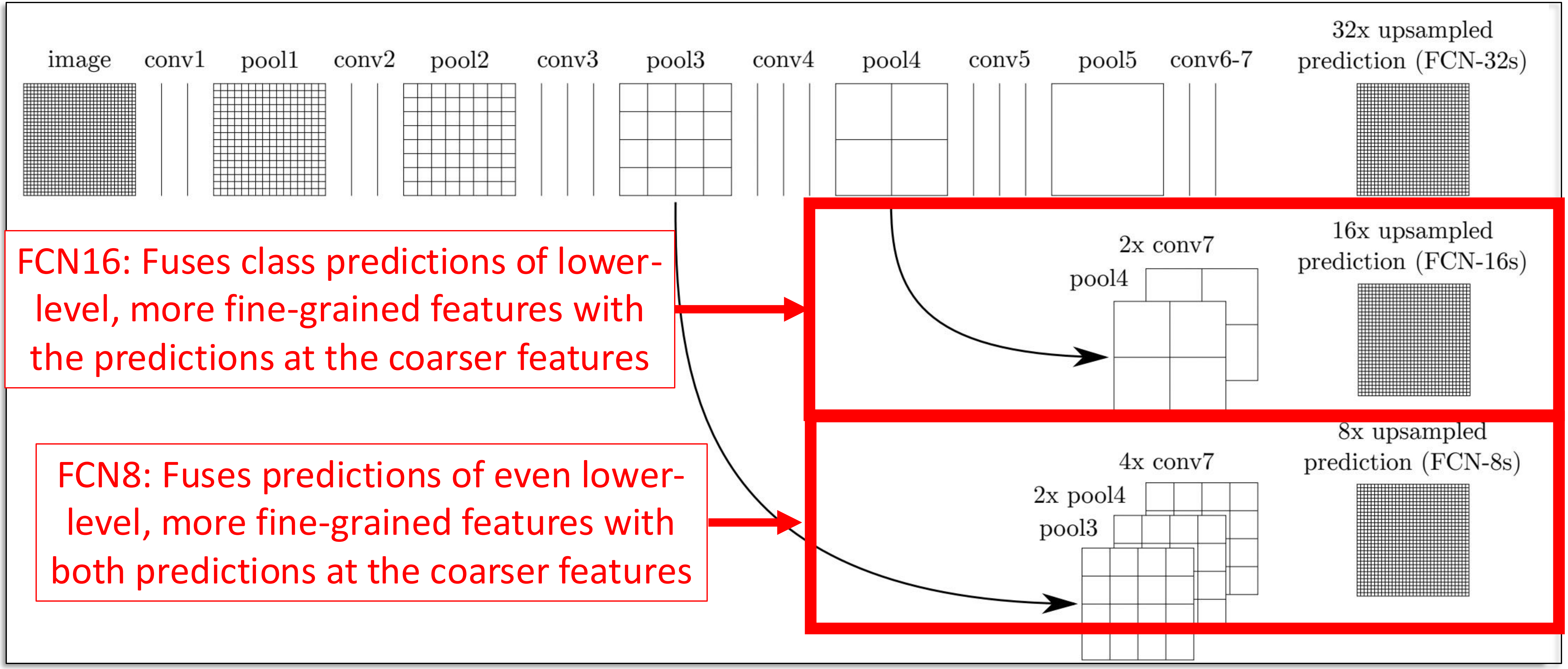


Predicted segmentation

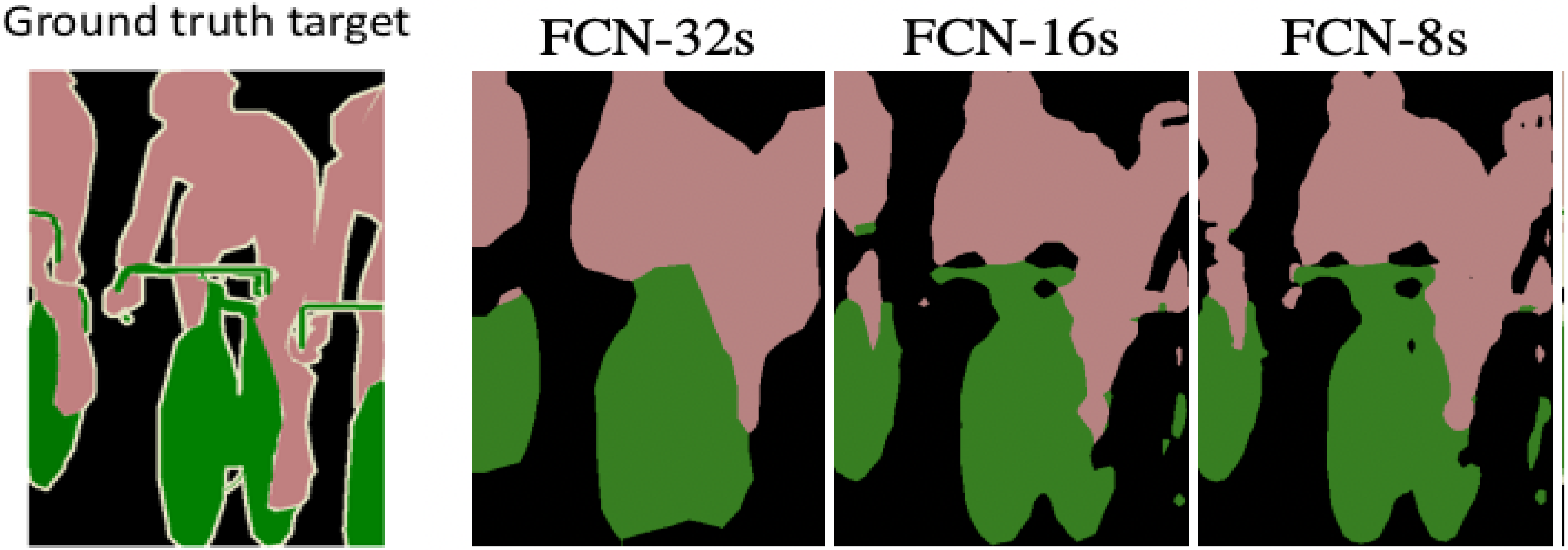


https://www.jeremyjordan.me/semantic-segmentation/

Architecture: Update to Use Skip Connections



Architecture: Results

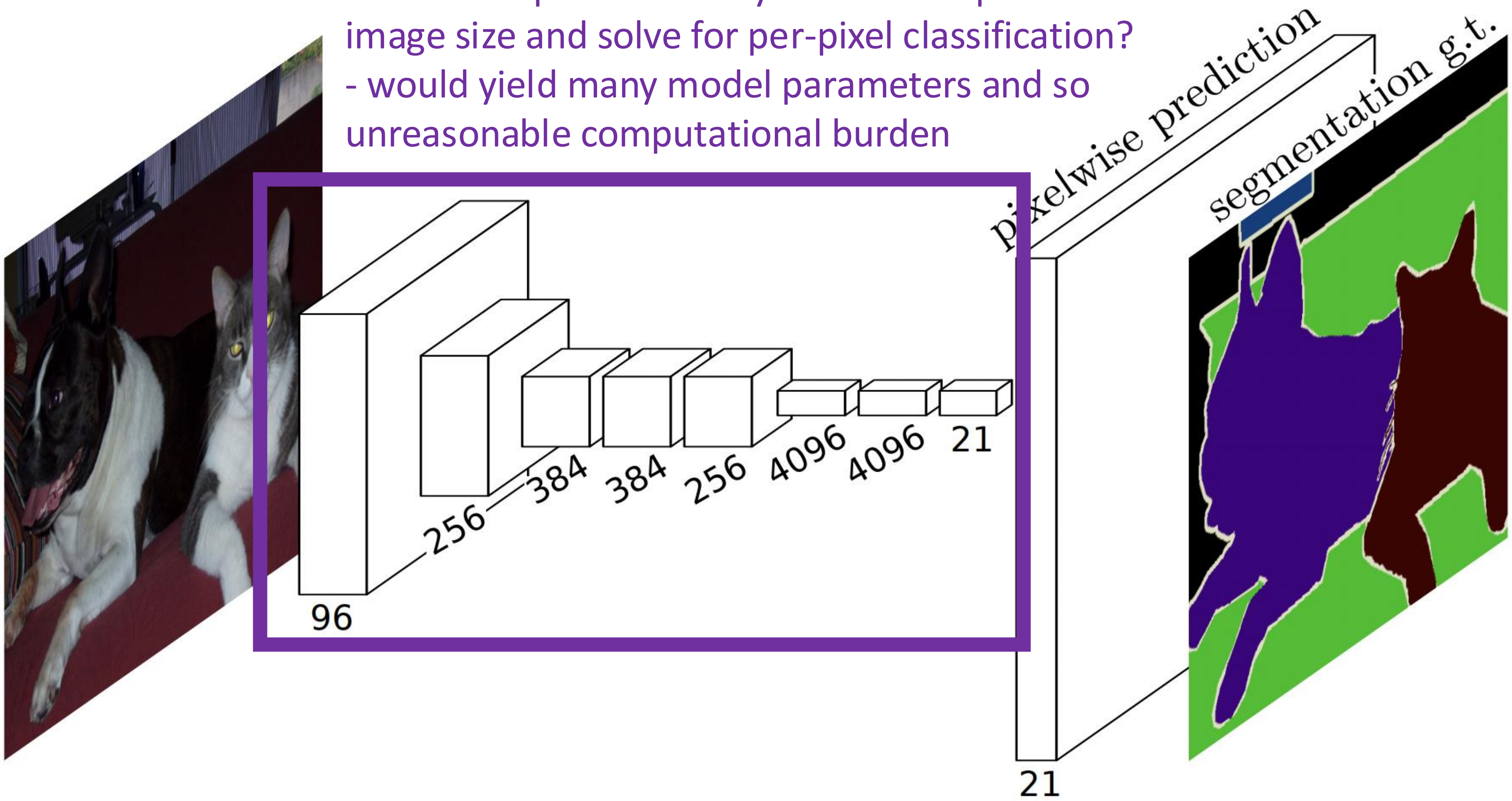


Skip connections support capturing finer-grained details while retaining correct semantic information!

https://www.jeremyjordan.me/semantic-segmentation/

Architecture: Upsampling + Skip Connections

Seems complicated… why not instead preserve the image size and solve for per-pixel classification?
- would yield many model parameters and so unreasonable computational burden

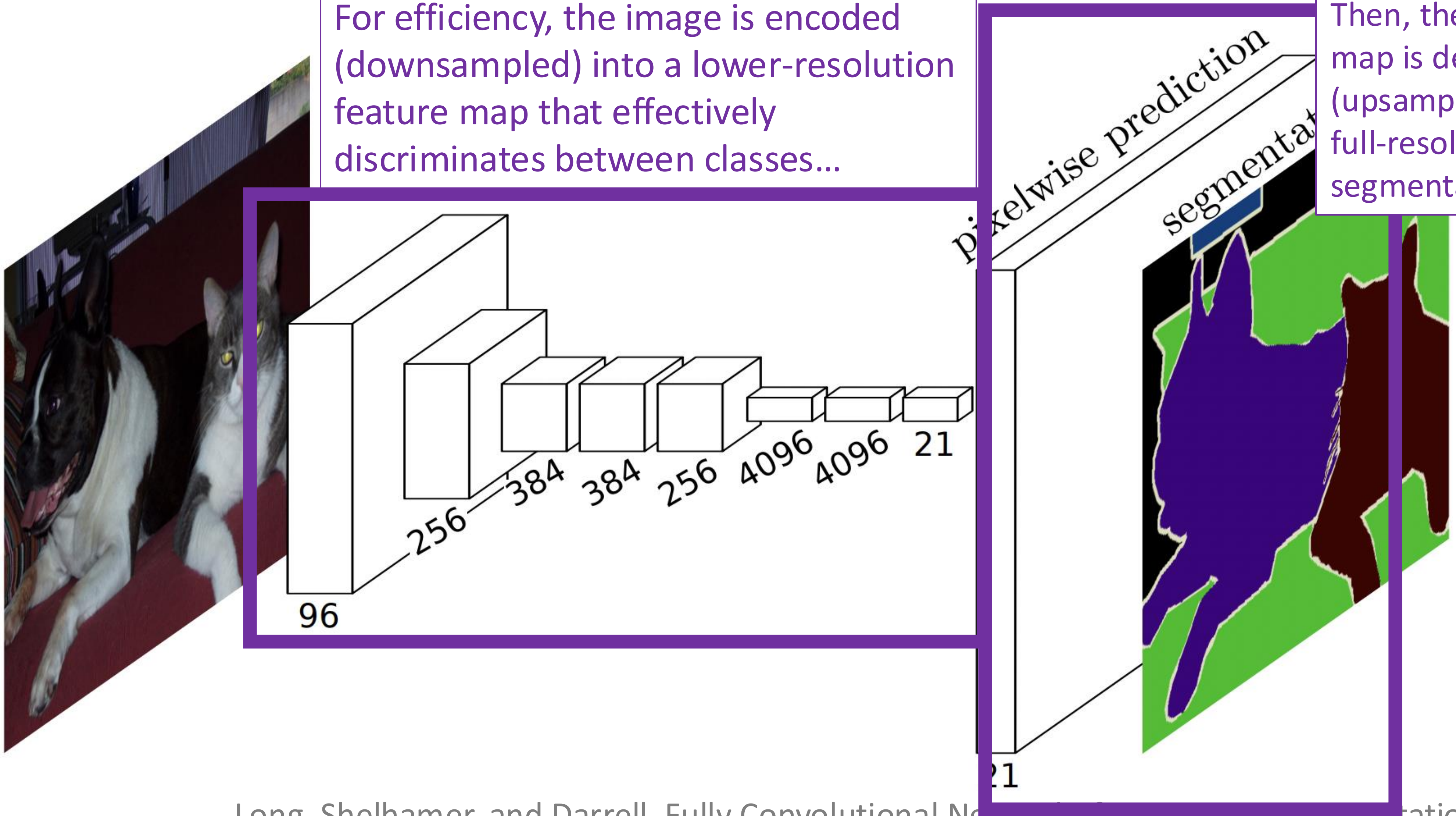


Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Architecture: Encoder Decoder Architecture

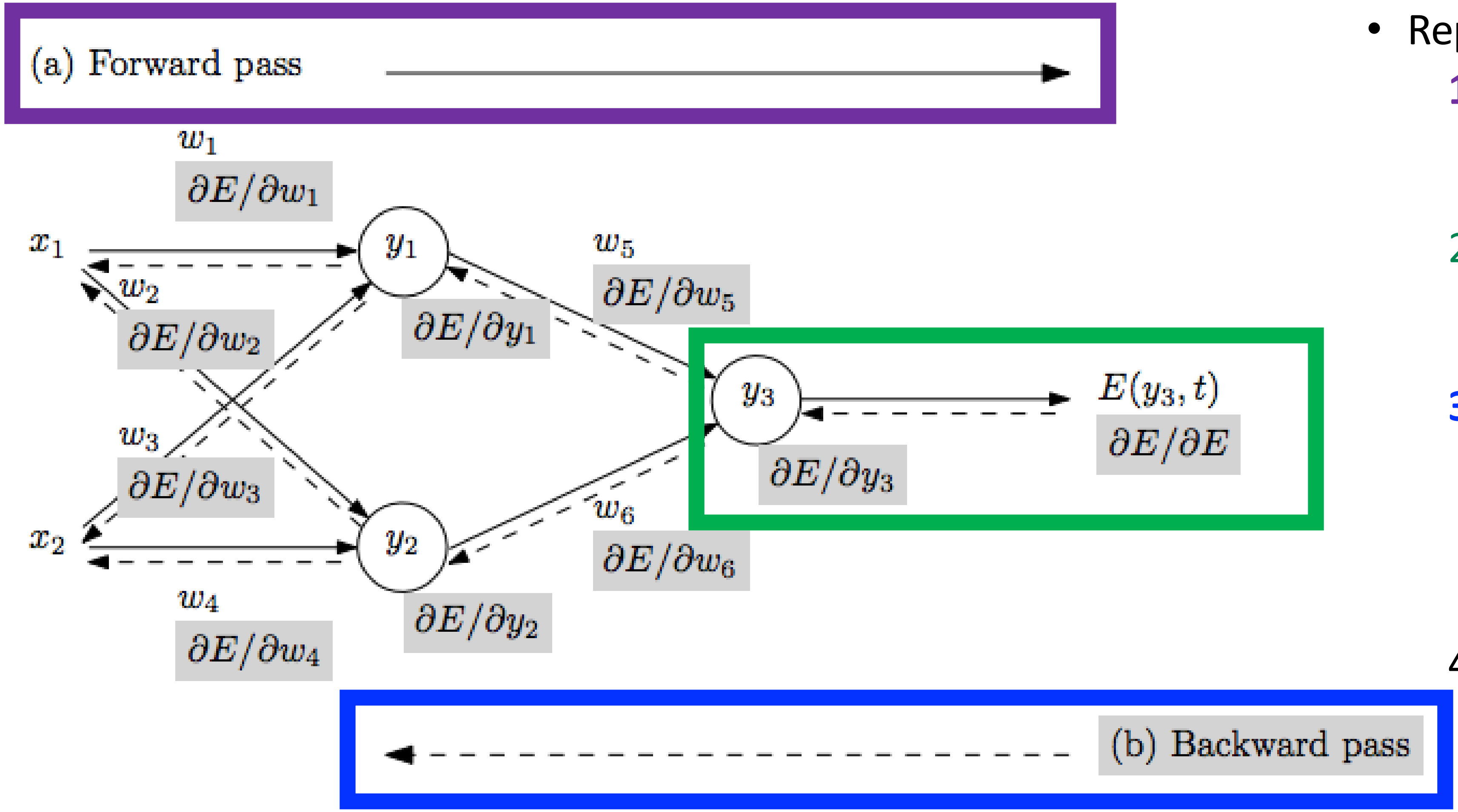
For efficiency, the image is encoded (downsampled) into a lower-resolution feature map that effectively discriminates between classes…

Then, the feature map is decoded (upsampled) into a full-resolution segmentation map.



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Training: Took 3 days on 1 GPU



- Repeat until stopping criterion met:
  1. **Forward pass**: propagate training data through model to make prediction
  2. Quantify the dissatisfaction with a model's results on the training data
  3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
  4. Update each parameter using calculated gradients

Figure from: Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, Jeffrey Mark Siskind; Automatic Differentiation in Machine Learning: a Survey; 2018

Training: How Neural Networks Learn

Sum across all pixels the distance between predicted and true distributions using cross entropy loss

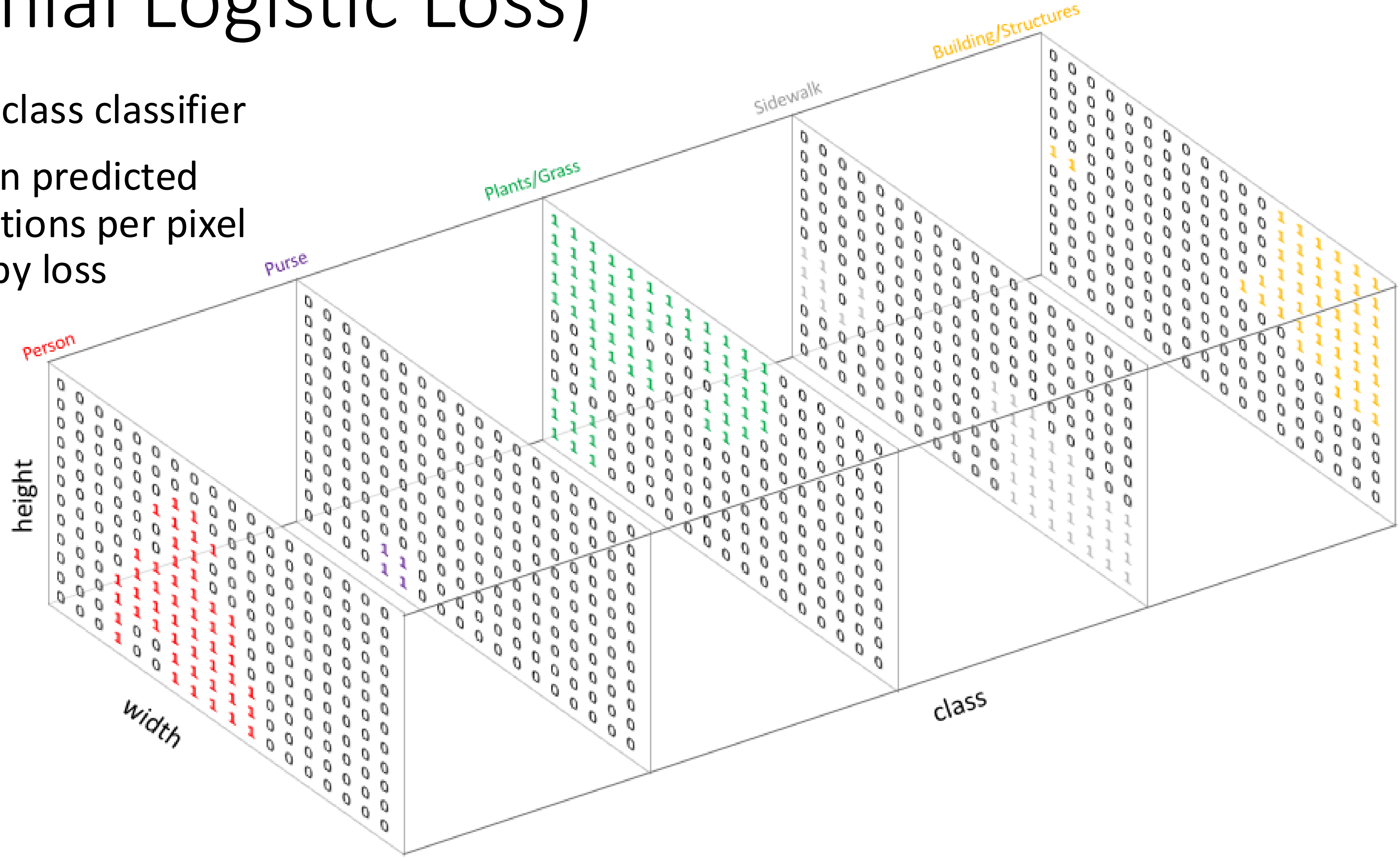
Sum of gradients for all pixels (acts like a minibatch)

- Repeat until stopping criterion met:
  1. **Forward pass**: propagate training data through model to make prediction
  2. Quantify the dissatisfaction with a model's results on the training data
  3. **Backward pass**: using predicted output, calculate gradients backward to assign blame to each model parameter
  4. Update each parameter using calculated gradients

Figure from: Atilim Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, Jeffrey Mark Siskind; Automatic Differentiation in Machine Learning: a Survey; 2018

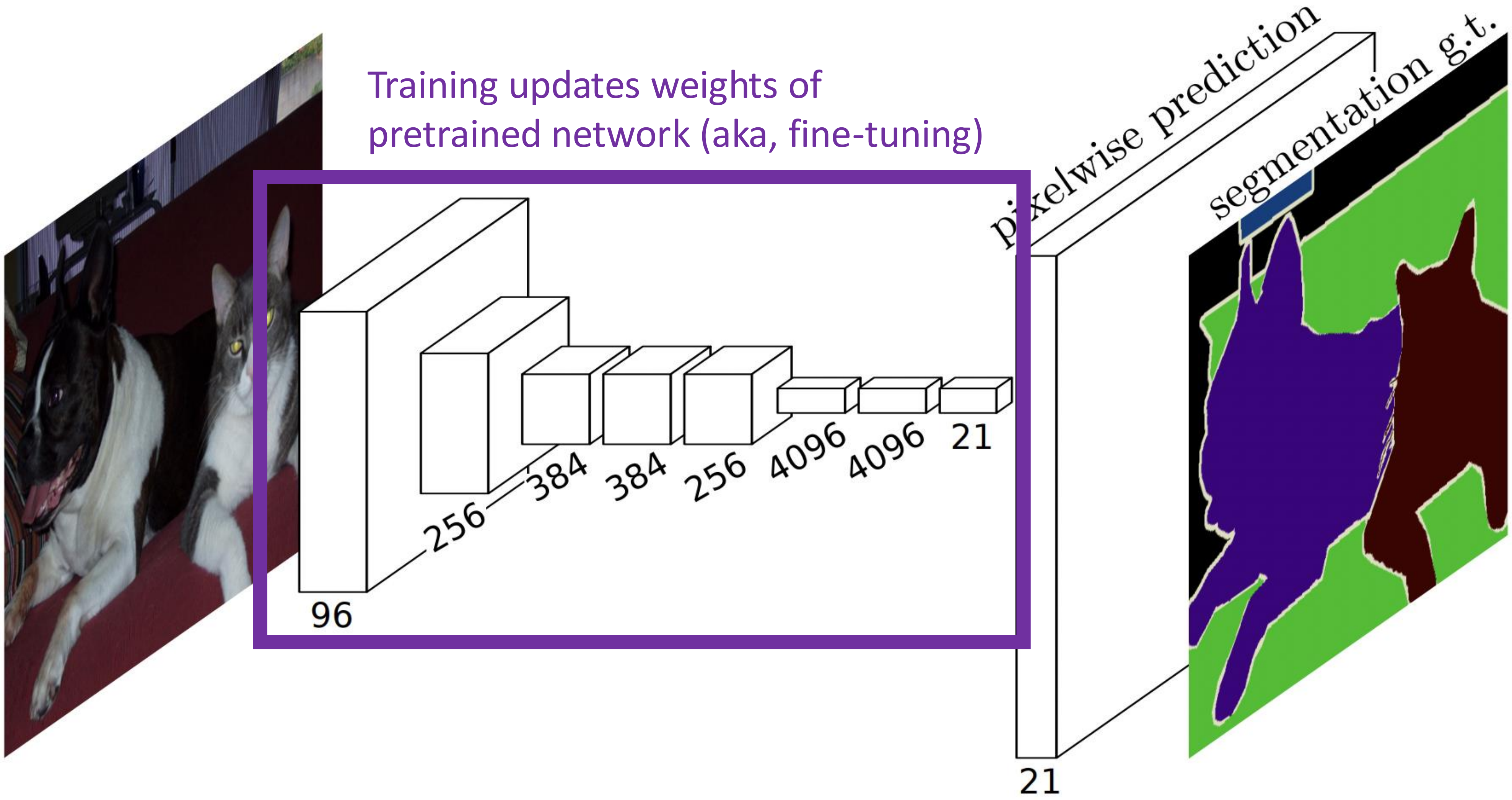
Training: Cross Entropy Loss (Multinomial Logistic Loss)

- e.g., assume a 5-class classifier
- Distance between predicted and true distributions per pixel with cross entropy loss



Architecture: Algorithm Training

Training updates weights of
pretrained network (aka, fine-tuning)



Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Results

	mean IU VOC2011 test	mean IU VOC2012 test	inference time
R-CNN [12]	47.9	-	-
SDS [16]	52.6	51.6	$\sim$ 50 s
FCN-8s	**62.7**	**62.2**	$\sim$ **175 ms**

Compared to existing methods, produces better results at a faster speed!

Long, Shelhamer, and Darrell. Fully Convolutional Networks for Semantic Segmentation. CVPR 2015

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Why Swin Transformer?

Named after the proposed technique: Shifted Windows

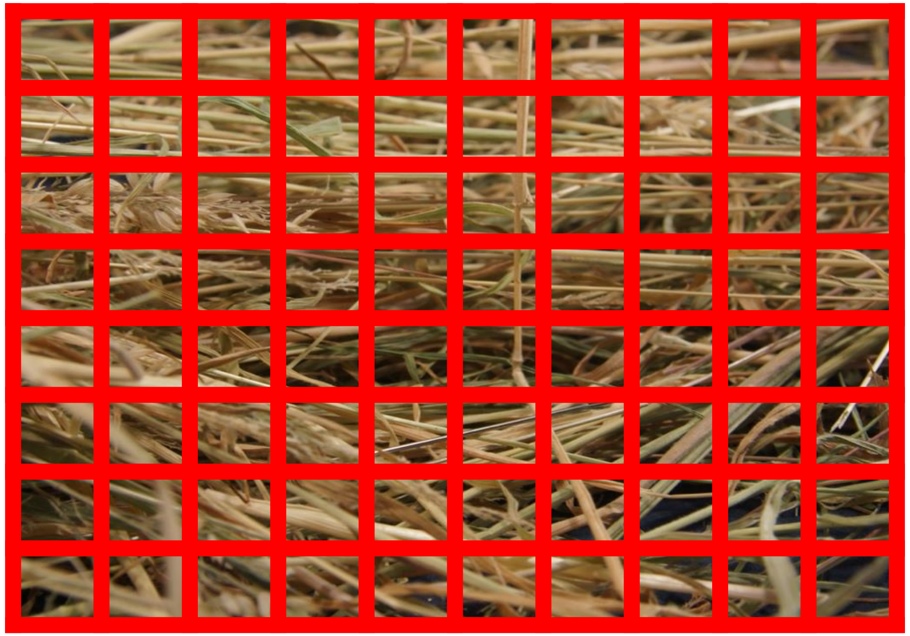
Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021.

Novelty

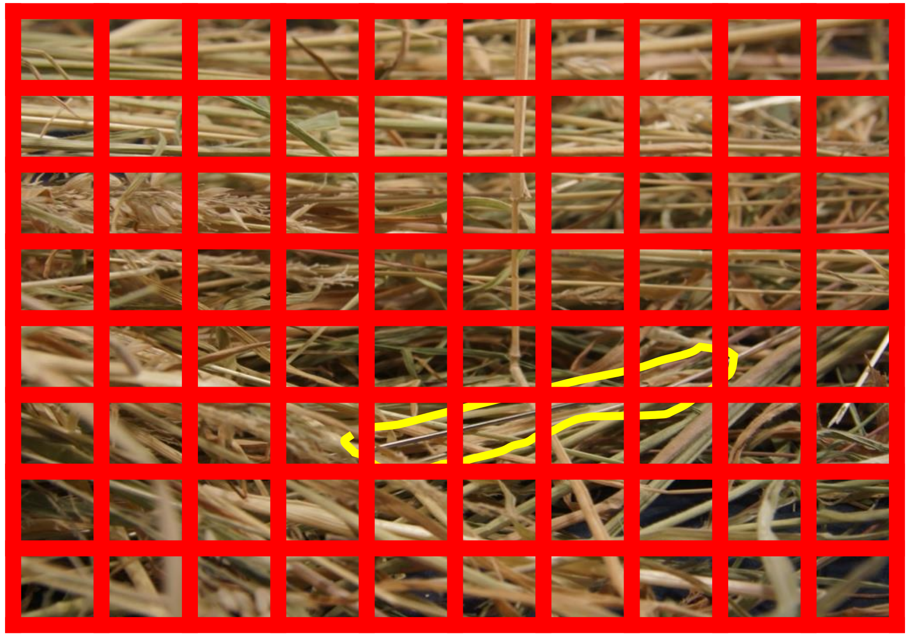
- Demonstrates a transformer “backbone” can generalize to diverse vision tasks, with state-of-the-art results for semantic segmentation and object detection (aka – dense prediction problems) as well as strong results for image classification

Why ViT Is Inadequate for Dense Prediction

Image classification
- What image label is predicted?
- "Big" patches are sufficient



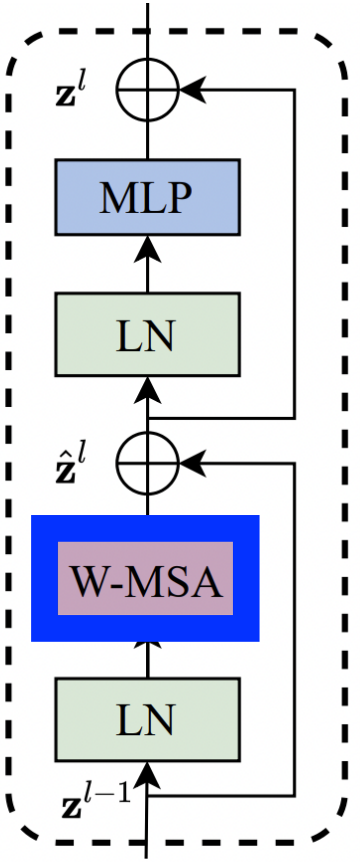
Object detection/Semantic segmentation
- What pixel label(s) are predicted?
- "Big" patches may be insufficient



Issue: quadratic expense of self-attention necessitated 16 x 16 patches, but this can be too large for pixel-level predictions (e.g., locating needle in a haystack)

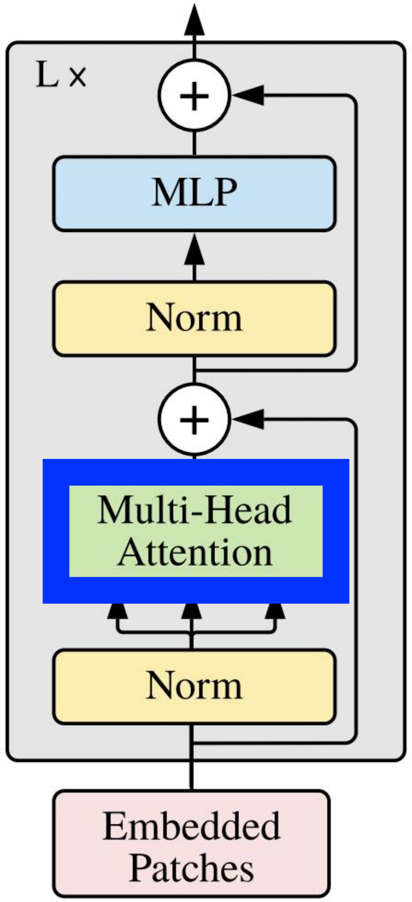
Key Idea of Swin: Modify Self-Attention Module

Swin Transformer



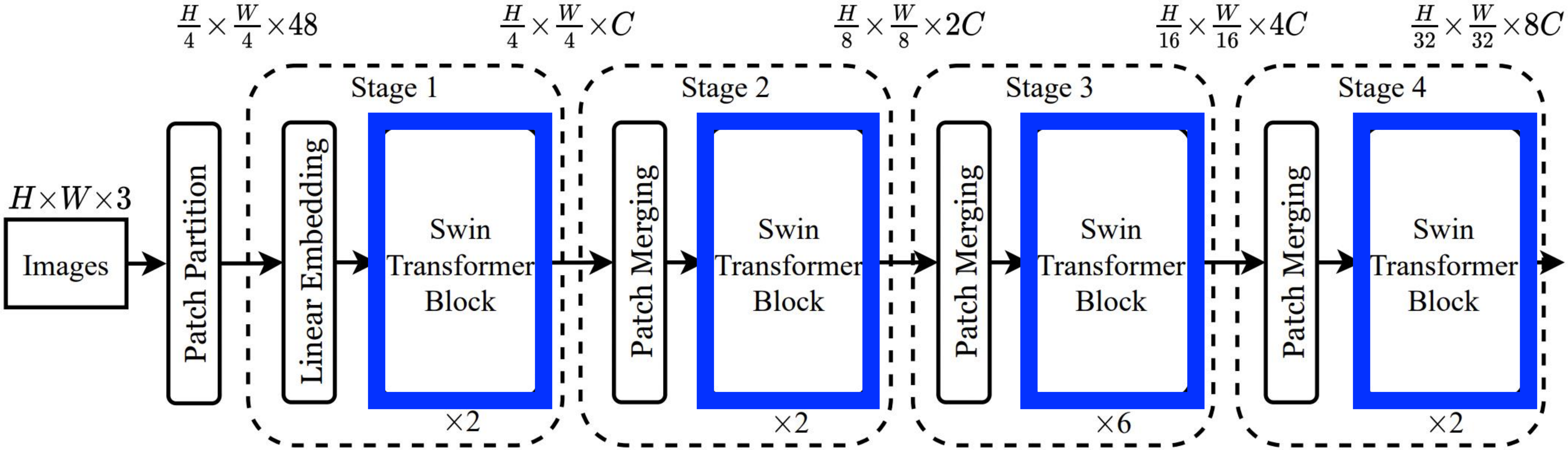
Liu et al. ICCV 2021.

ViT



Dosovitskiy et al. ICLR 2021.

Architecture

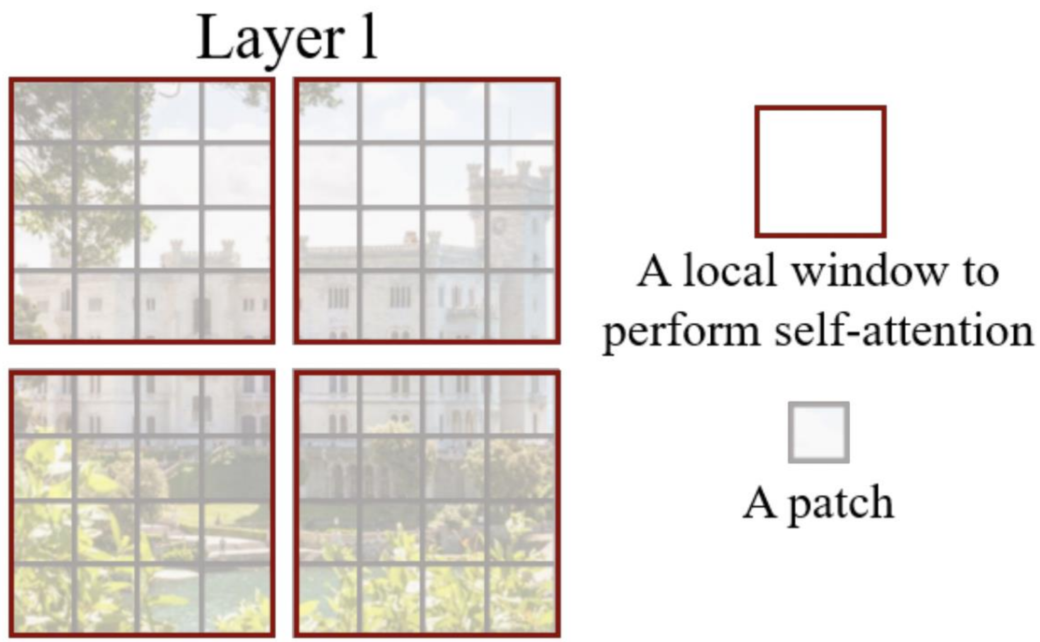


Contains a series of modified self-attention modules

Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Key Idea: Modified Self-Attention Module

Applies self-attention only between the fixed number of patches in each window to capture fine-grained details (i.e., limited to local context)



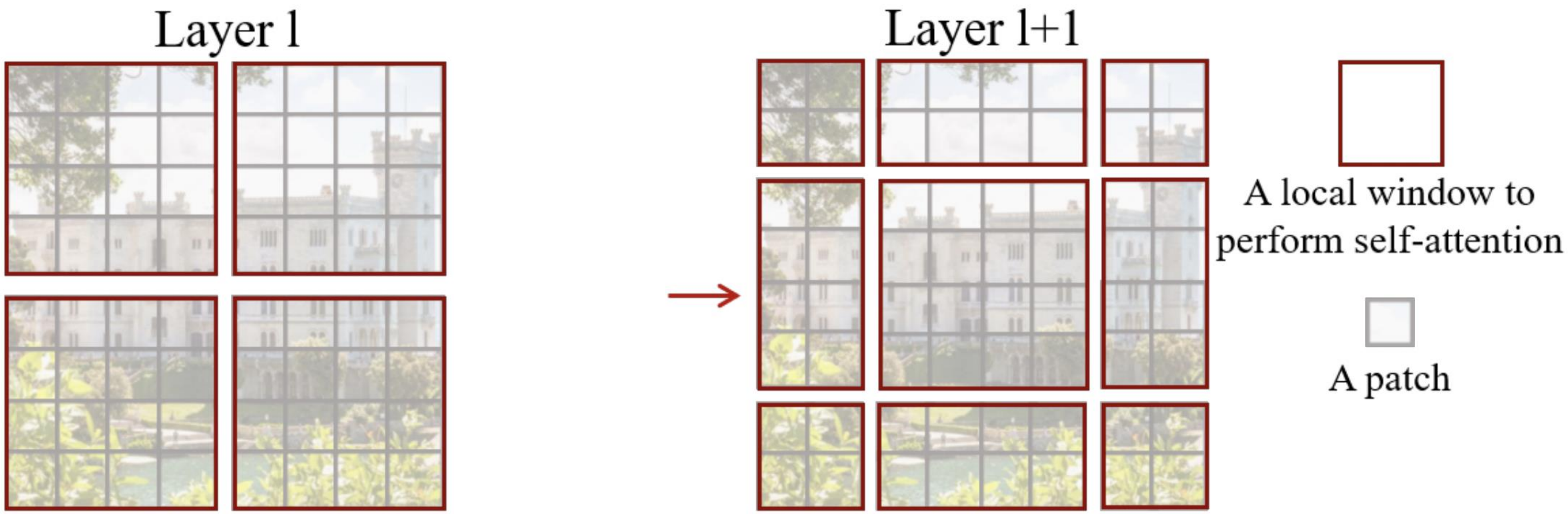
What is the computational complexity?
- Linear based on fixed patch number chosen per window rather than quadratic based on number of input patches

Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Key Idea: Modified Self-Attention Module

Applies self-attention only between the fixed number of patches in each window to capture fine-grained details (i.e., limited to local context)

In each subsequent layer, windows shifted to infuse global context by enabling communication between previously non-communicative neighboring patches



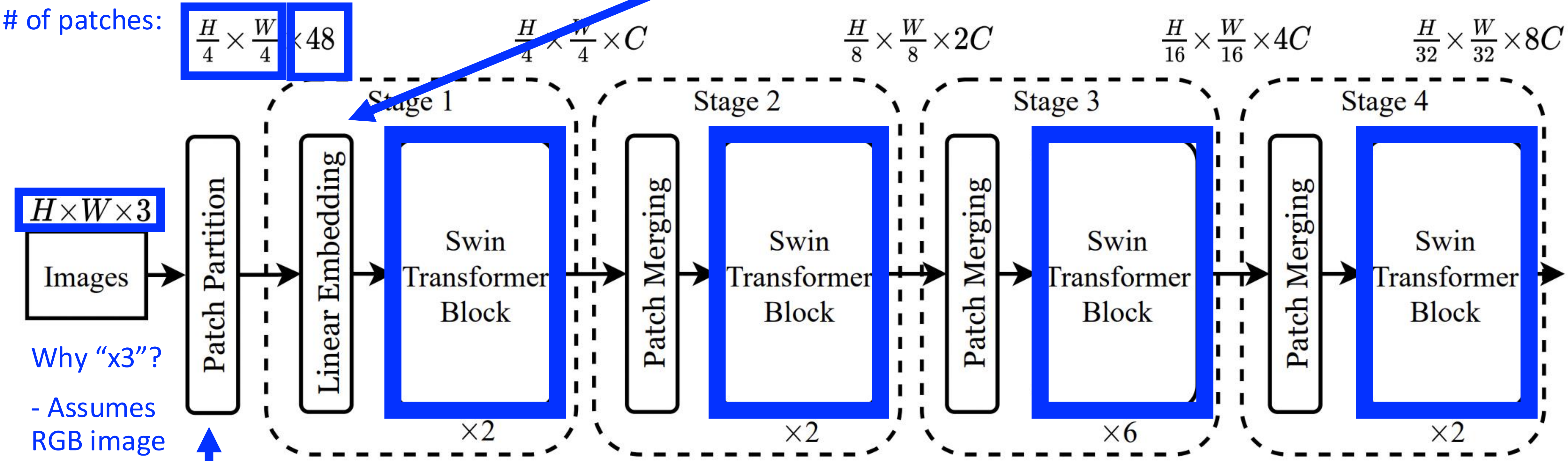
Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Architecture

Each patch is 4 x 4 (x3);
Smaller than 16 x 16 (x3)

Input linearly converted to a target (smaller) size;
e.g., 96 and 192

# of patches:



Why "x3"?

- Assumes RGB image

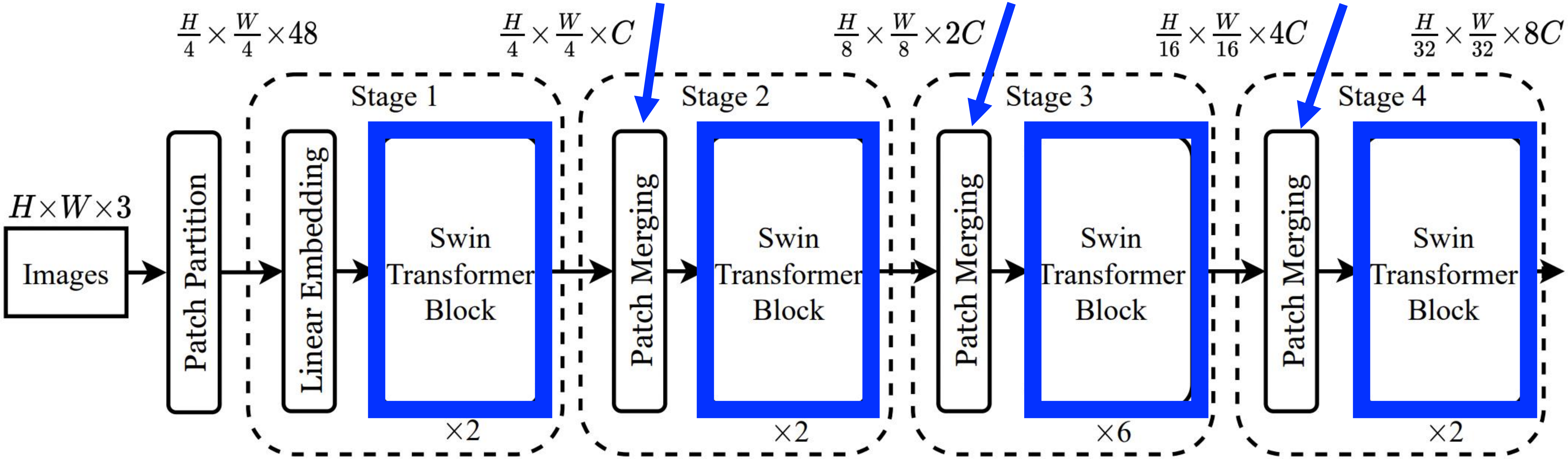
Contains a series of modified self-attention modules

How many image pixels are in each image patch?

Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Architecture

Neighboring patches merged into increasingly bigger patches (mimics convolutional layers); this hierarchical design also increases global context to better support visual content at different scales! (*output feature maps match resolution of common CNNs, e.g., VGG & ResNet*)



Contains a series of modified self-attention modules at different resolutions

Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Dense Prediction: State-of-the Art Results

Four object detection algorithms tested on COCO 2017 with three “backbone” sources:
- ResNe(X)t
- DeiT
- Swin: was consistently top-performer

UperNet semantic segmentation algorithm tested on ADE20K with two “backbone” sources:
- DeiT
- Swin: was consistently top-performer

Liu et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV 2021

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)

Semantic Segmentation: Today's Topics

- Motivation
- Datasets
- Evaluation metric
- Fully convolutional network
- Swin transformer
- Discussion (chosen by YOU ☺)


The End