COMP9517
Computer Vision

2024 Term 2 Week 8

Dr Dong Gong

UNSW SYDNEY

Deep Learning II

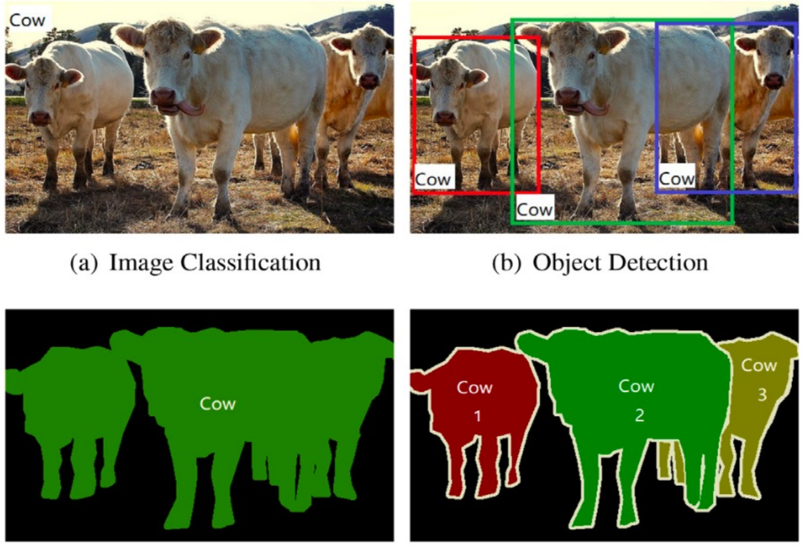
Semantic Segmentation, Instance Segmentation and Video Understanding using CNNs

Outline

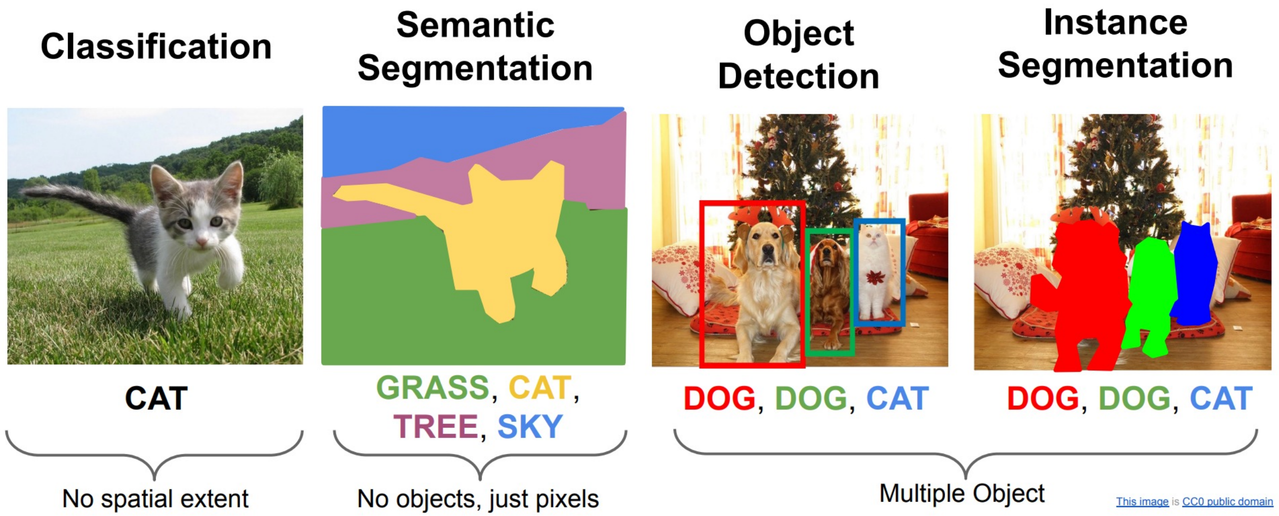
- Computer Vision tasks
- Semantic Segmentation
  - Sliding Window
  - Fully Convolutional Networks (FCNs)
  - U-Net
  - U-Net variants
- Instance Segmentation
  - Mask R-CNN
- Video understanding
  - Challenges in processing videos
  - Video datasets
  - C3D: Learning spatiotemporal features with 3D CNN
  - Two-stream network for video classification

Vision tasks

- Image classification: Assigning a label or class to an image
- Object detection: Locate the presence of objects with a bounding box and class of the located objects in an image
- Semantic segmentation: Label every pixel (pixel-wise classification)
- Instance segmentation: Differentiate instances

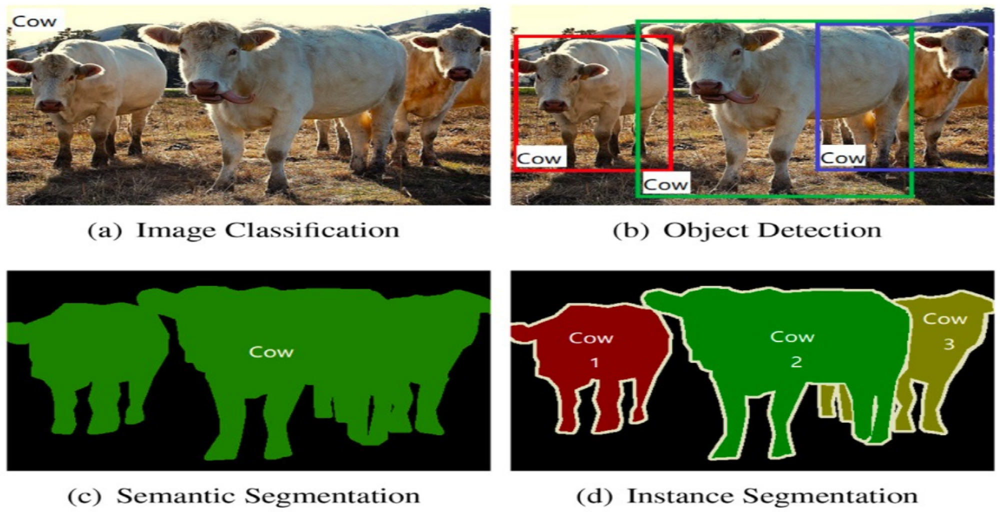


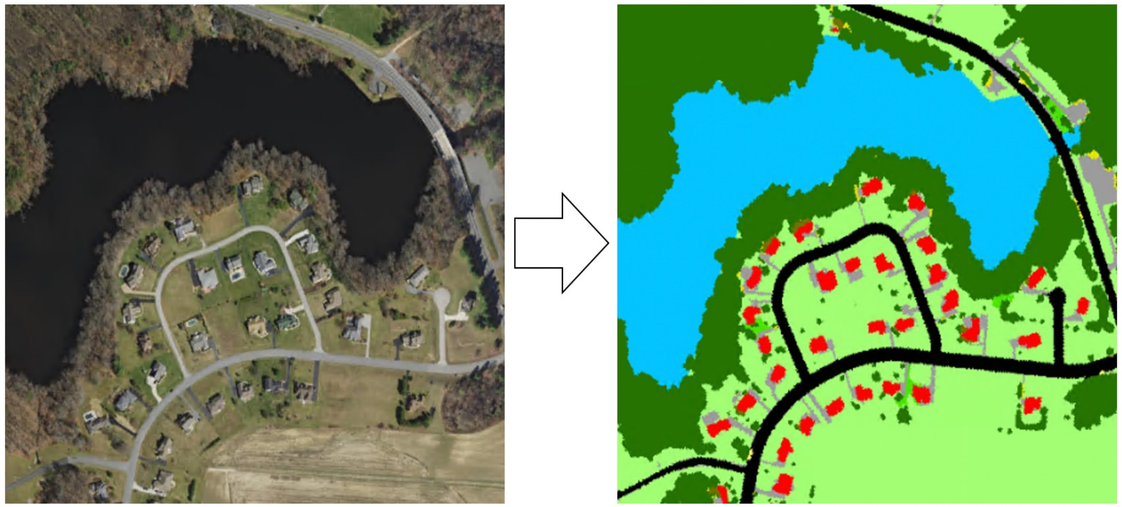
(a) Image Classification (b) Object Detection
(c) Semantic Segmentation (d) Instance Segmentation



Semantic Segmentation

- Classify each pixel in an image





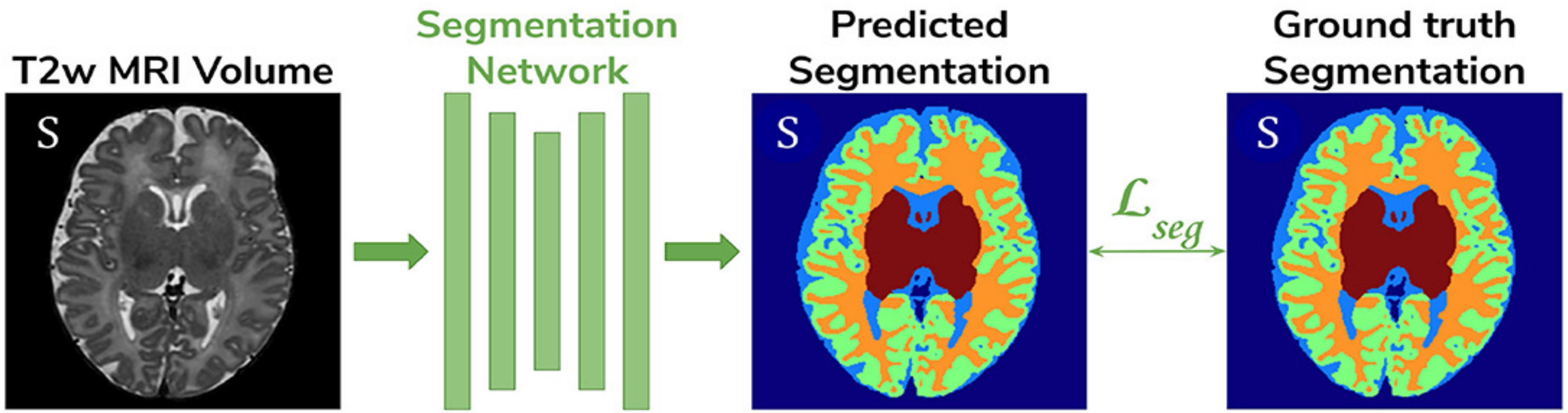


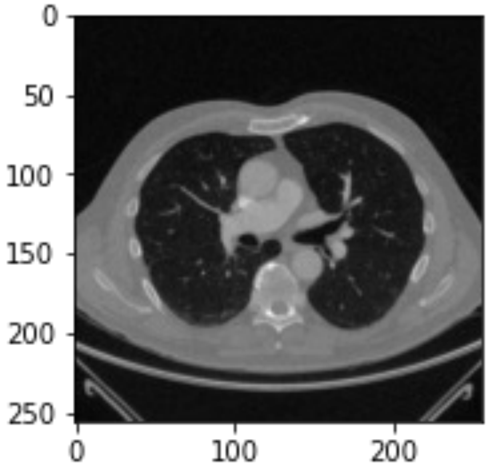
Image Credit: Creative Commons Licenses

How to train semantic segmentation network?

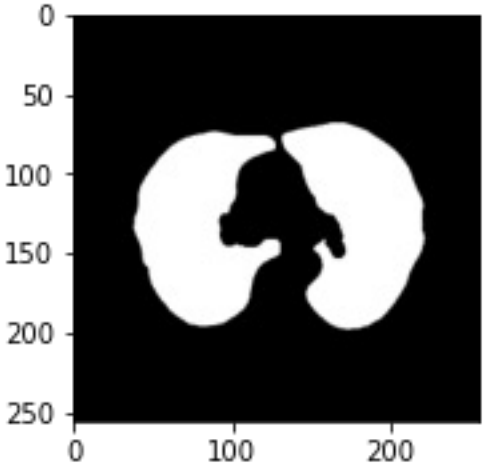
- For each image, annotated mask or ground-truth mask is given

Training

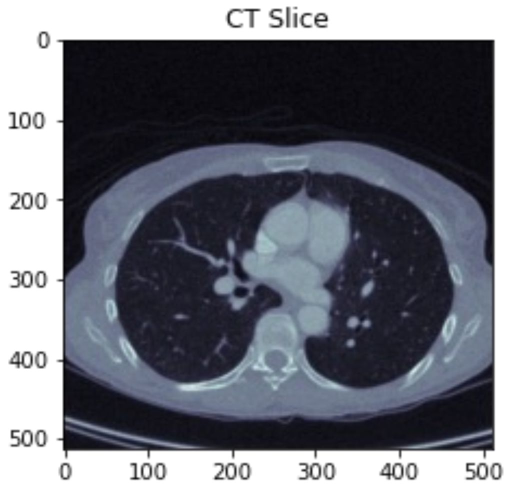
Input CT Image



Annotated Mask



Test



Segmentation Model

Label every pixel

Sliding Window approach

➢ Classify individual pixels

Image Credit: Creative Commons Licenses

Sliding Window approach

- Classifying individual pixel is not a good idea
  - No context!
- How can we include neighbourhood context to classify individual pixel?

Sliding Window approach

➢ Idea: Extract “patches” from entire image, classify centre pixel based on the neighbouring context

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

Image Credit: Creative Commons Licenses

Sliding Window approach

➢ Limitations: 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

Semantic Segmentation using Convolution

- Idea: Encode the entire image with "Conv Net", and do semantic segmentation
- Problem: Semantic segmentation requires the output size to be the same as input size (see below).

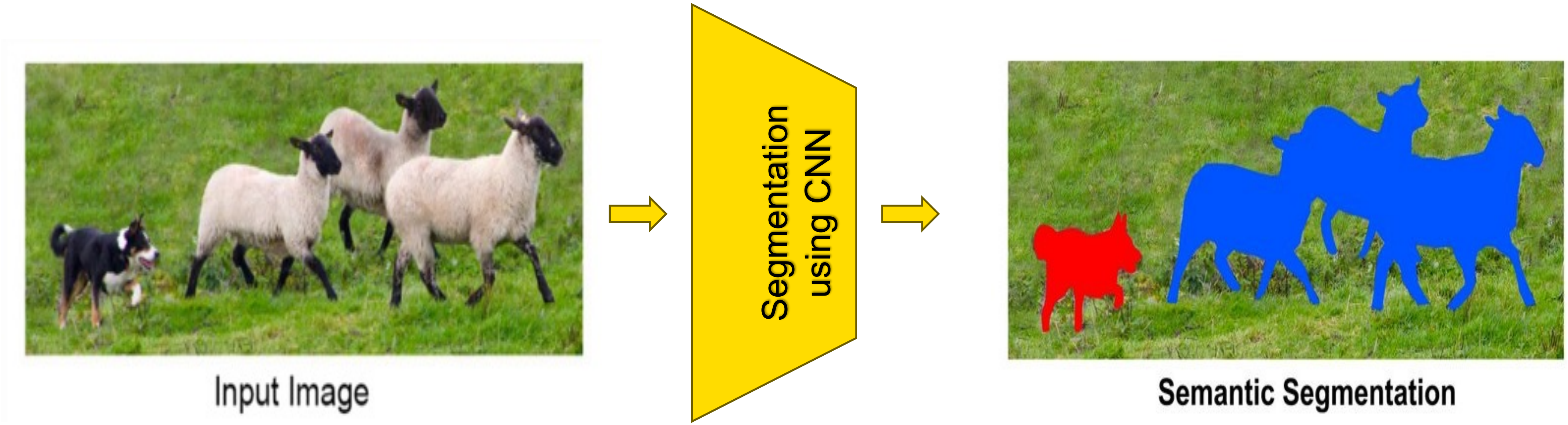
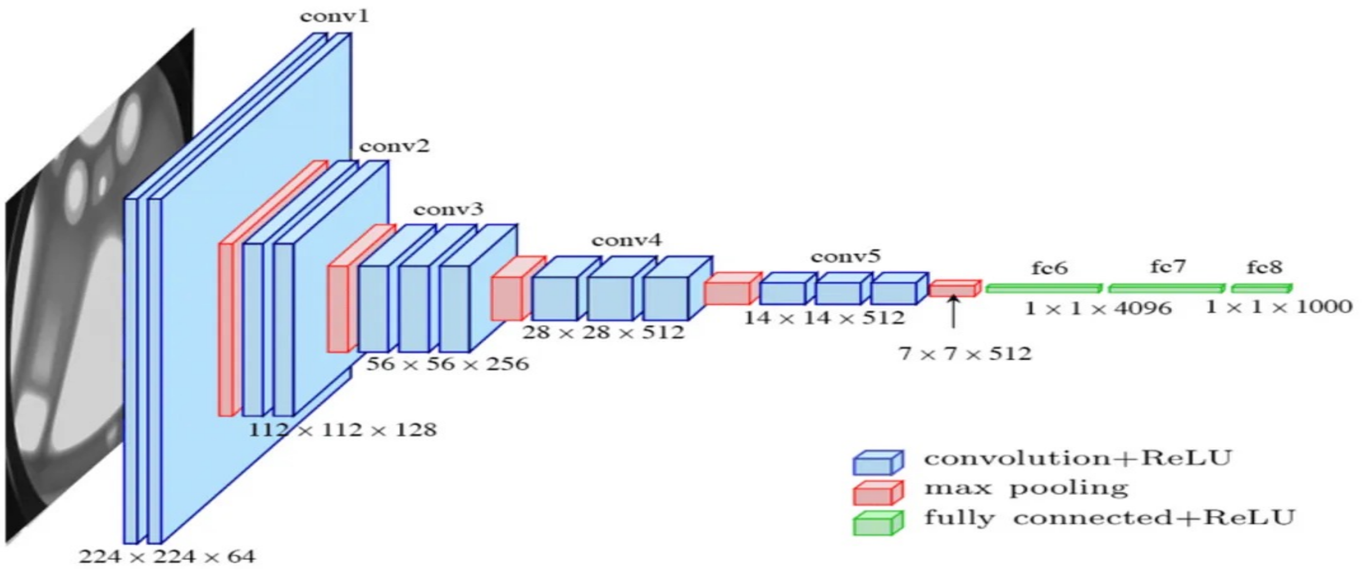


Image Credit: Creative Commons Licenses

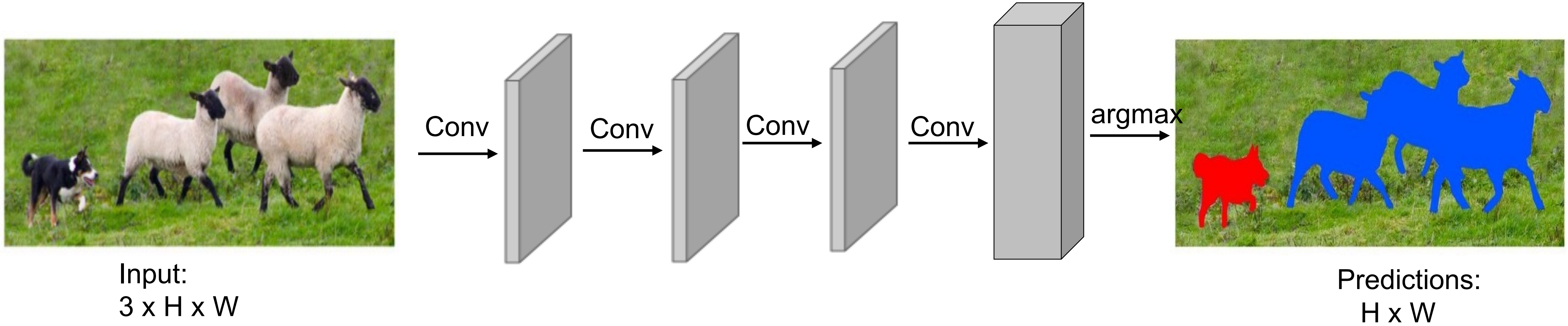
Semantic Segmentation using Convolution

- However, CNN classification architectures reduces spatial size of features as they go deeper (due to downsampling)



Fully Convolutional Networks for Semantic Segmentation

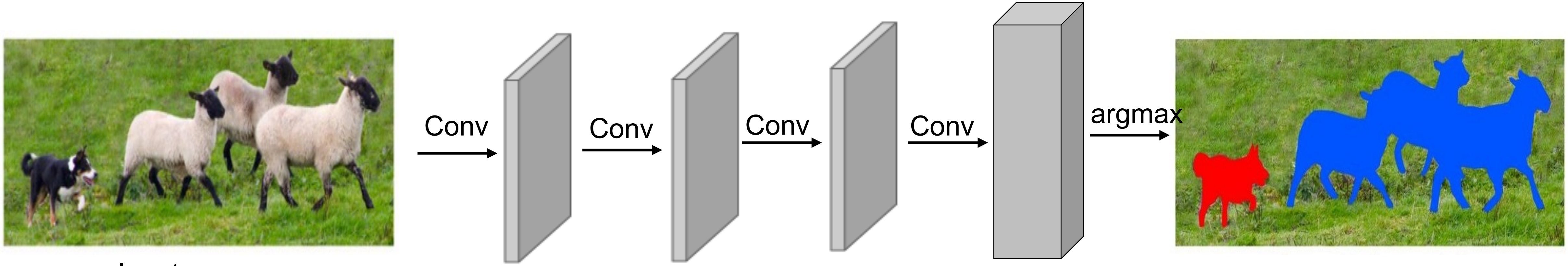
- Design a network with only convolutional layers without downsampling operators to make prediction map of same size as of input image



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

Fully Convolutional Networks for Semantic Segmentation

- Design a network with only convolutional layers without downsampling operators to make prediction map of same size as of input image



Input:
3 x H x W

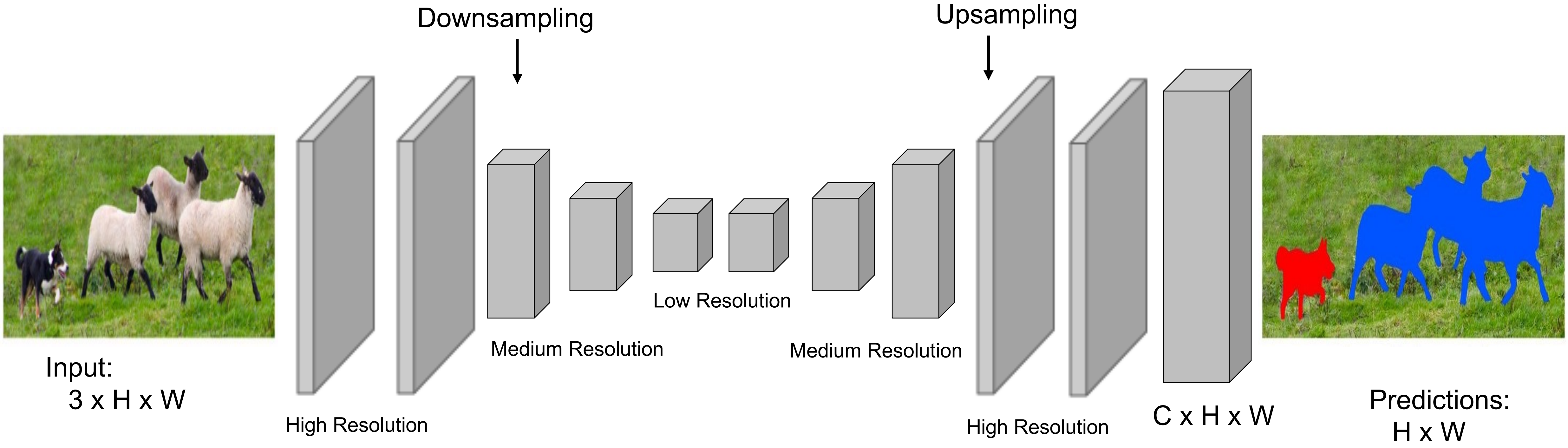
Predictions:
H x W

Problem:
Convolutions at original image resolution will be very expensive

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

Fully Convolutional Networks for Semantic Segmentation

➢ Design a network having convolutional layers, with downsampling and upsampling inside the network (learning in an end-to-end manner)

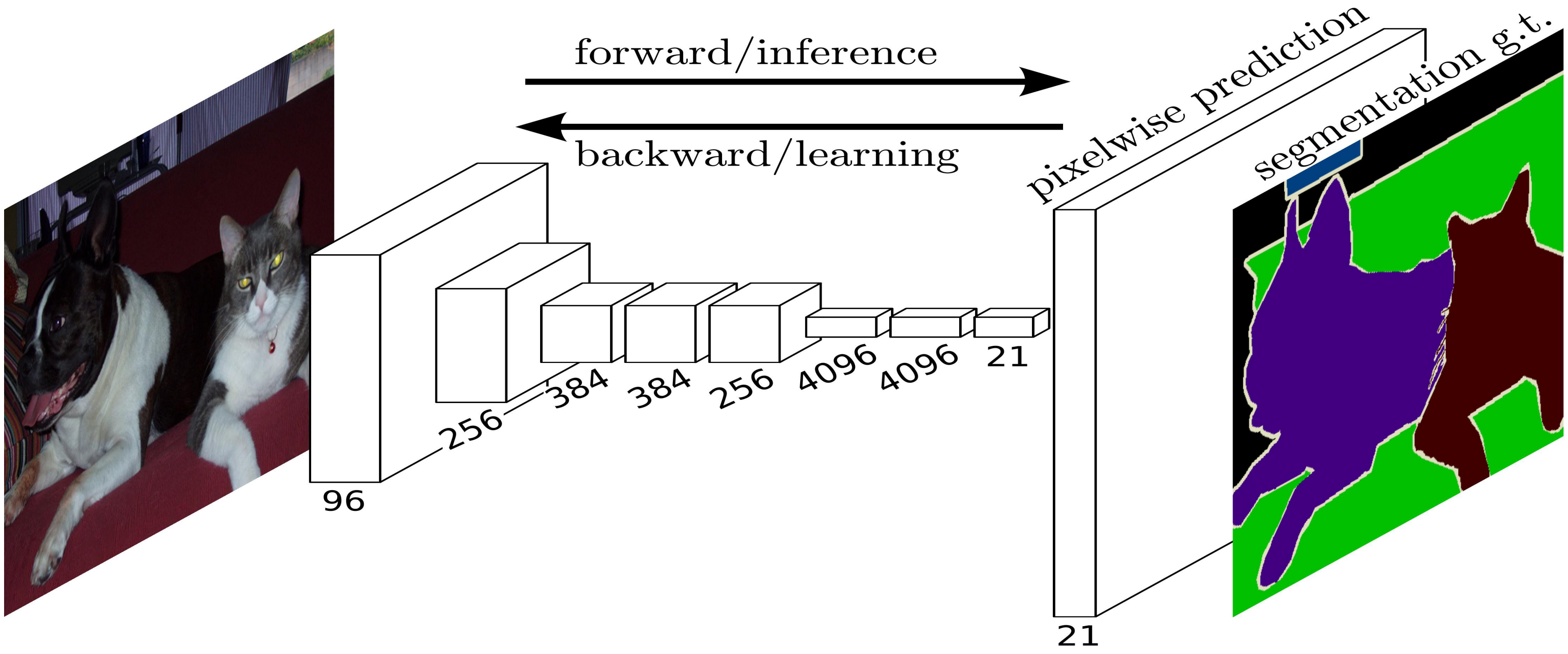


Convolutions at original image resolution will be very expensive

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

Fully Convolutional Networks for Semantic Segmentation

➢ Design a network with only convolutional layers without downsampling operators to make prediction map of same size as of input image



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

In-network Upsampling: Unpooling

- ➢ Abstract feature maps are upsampled to make their spatial dimensions equal to the input image

Max-Pooling

1	2	6	3
3	5	2	1
1	2	2	1
7	3	4	8

Input: 4 x 4

→

5	6
7	8

Output: 2 x 2

Nearest Neighbour

1	2
3	4

Input: 2 x 2

→

1	1	2	1
1	1	2	2
3	3	4	4
3	3	4	4

Output: 4 x 4

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

In-network Upsampling: Unpooling

➢ Abstract feature maps are upsampled to make their spatial dimensions equal to the input image

Max-Pooling

1	2	6	3
3	5	2	1
1	2	2	1
7	3	4	8

Input: 4 x 4

→

5	6
7	8

Output: 2 x 2

Unpooling

1	2
3	4

Input: 2 x 2

→

1	0	2	0
0	0	0	0
3	0	4	0
0	0	0	0

Output: 4 x 4

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

In-network Upsampling: Max Unpooling

➢ Abstract feature maps are upsampled to make their spatial dimensions equal to the input image

Max-Pooling

Remember the position of the max element

1	2	6	3
3	5	2	1
1	2	2	1
7	3	4	8

→

5	6
7	8

→ ------------- →

Rest of the network

1	2
3	4

→

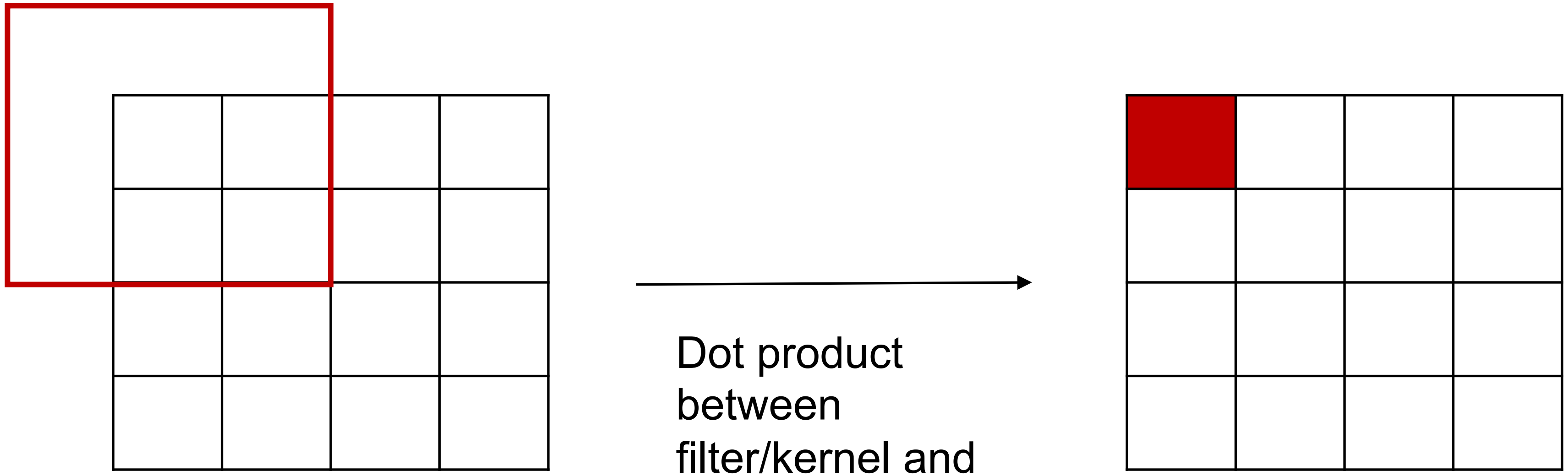
Max-Unpooling

Use position from pooling layer

0	0	2	0
0	1	0	0
0	0	0	0
3	0	0	4

Learning Upsampling: Transpose Convolution

Recall: Typical 3 x 3 convolution; stride 1, padding 1



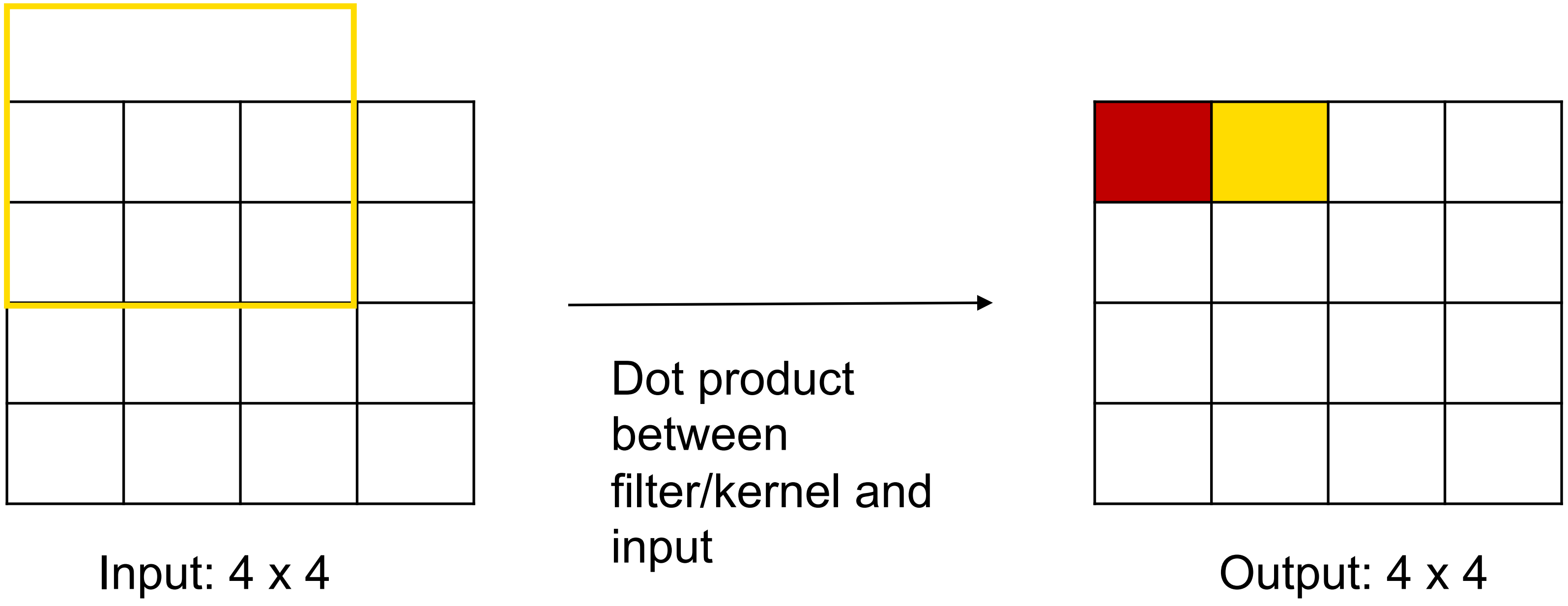
Input: 4 x 4

Dot product between filter/kernel and input

Output: 4 x 4

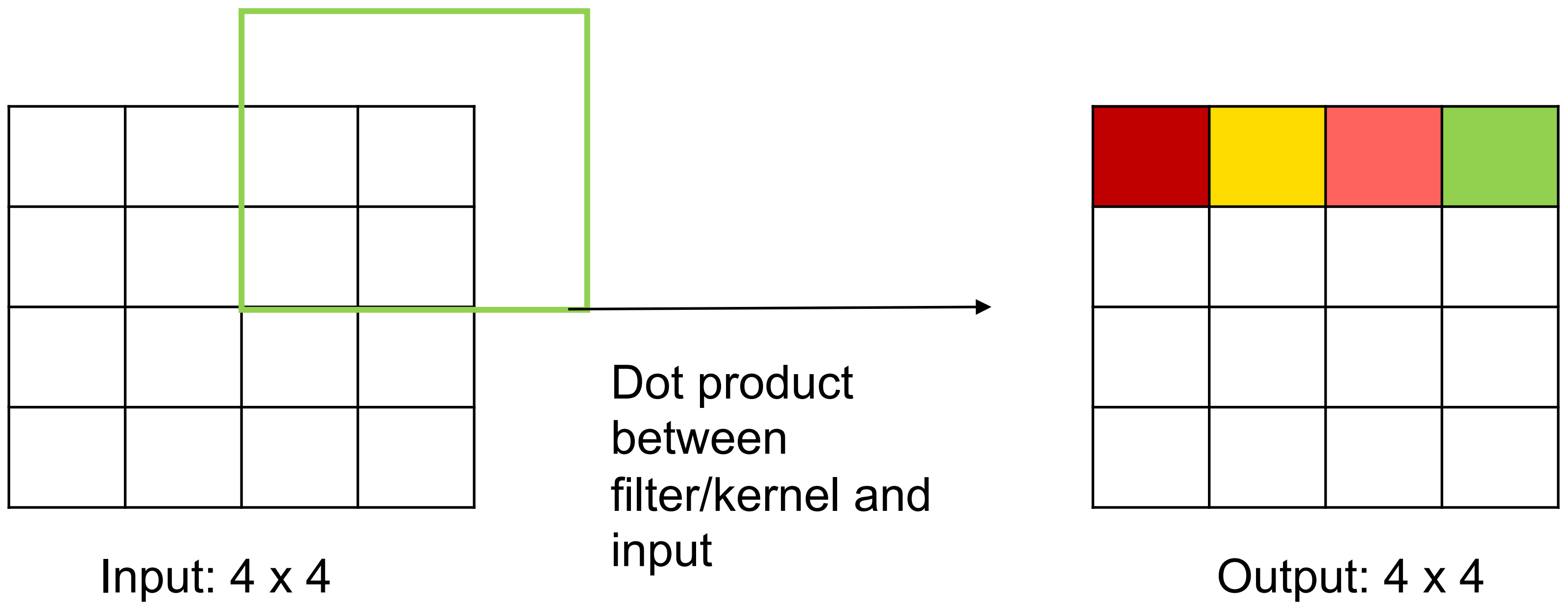
Learning Upsampling: Transpose Convolution

Recall: Typical 3 x 3 convolution; stride 1, padding 1



Learning Upsampling: Transpose Convolution

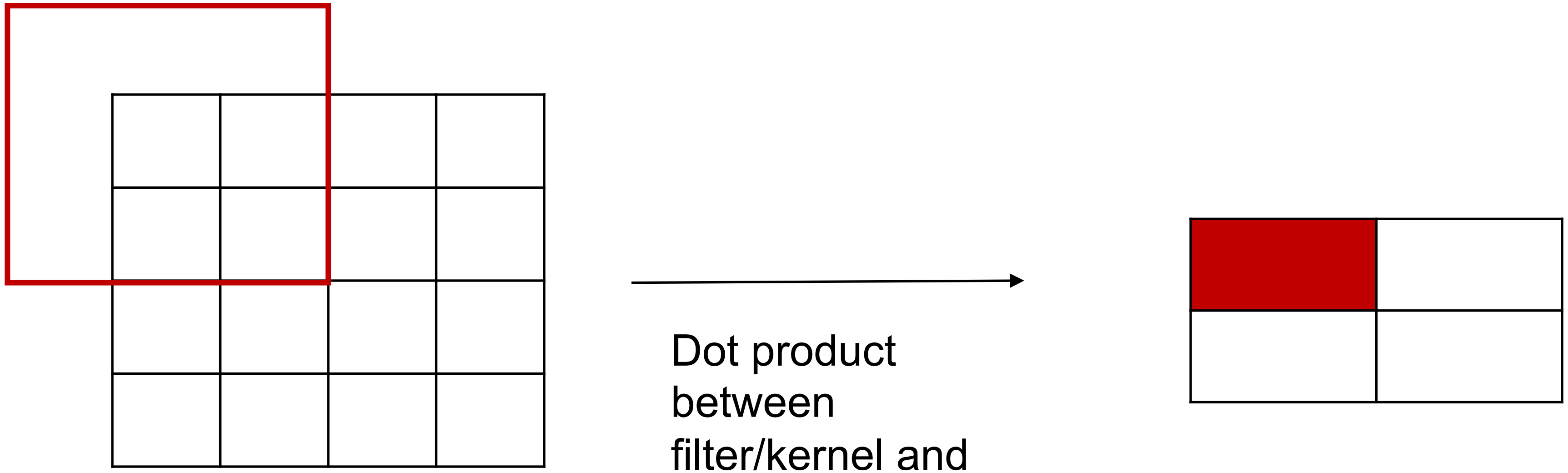
Recall: Typical 3 x 3 convolution; stride 1, padding 1



Learning Upsampling: Transpose Convolution

Recall: Stride Convolution

Recall: Typical 3 x 3 convolution; stride 2, padding 1



Input: 4 x 4

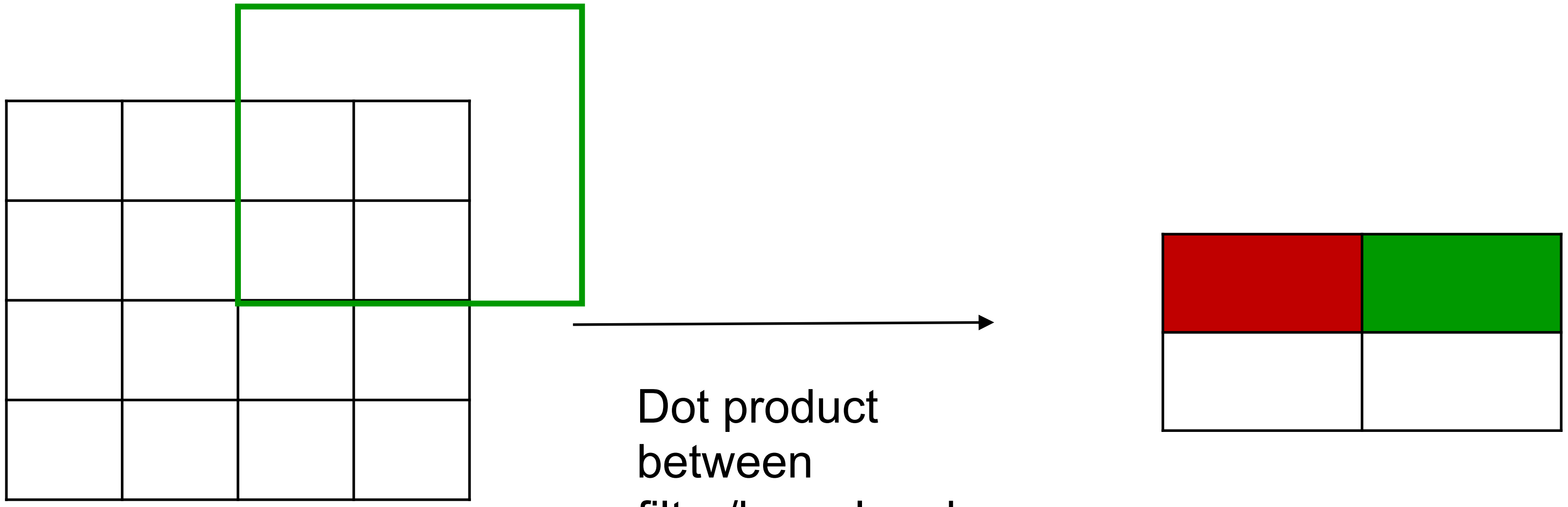
Dot product between filter/kernel and input

Output: 2 x 2

Learning Upsampling: Transpose Convolution

Recall: Stride Convolution

Recall: Typical 3 x 3 convolution; <u>stride 2</u>, padding 1



Input: 4 x 4

Dot product between filter/kernel and input

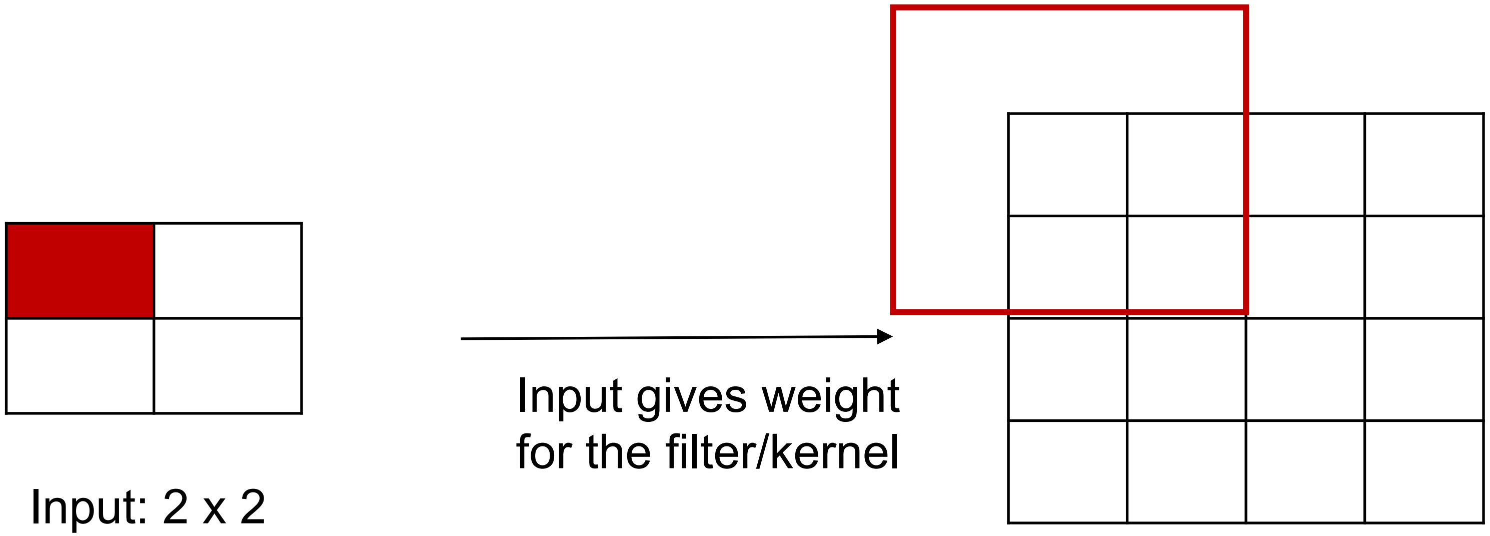
Output: 2 x 2

Filter/Kernel moves 2 pixels in the input for every one pixel in the output.

Stride gives ratio between movement in the input and the output.

Learning Upsampling: Transpose Convolution

3 x 3 Transpose Convolution; stride 2, padding 1



Input: 2 x 2

Output: 4 x 4

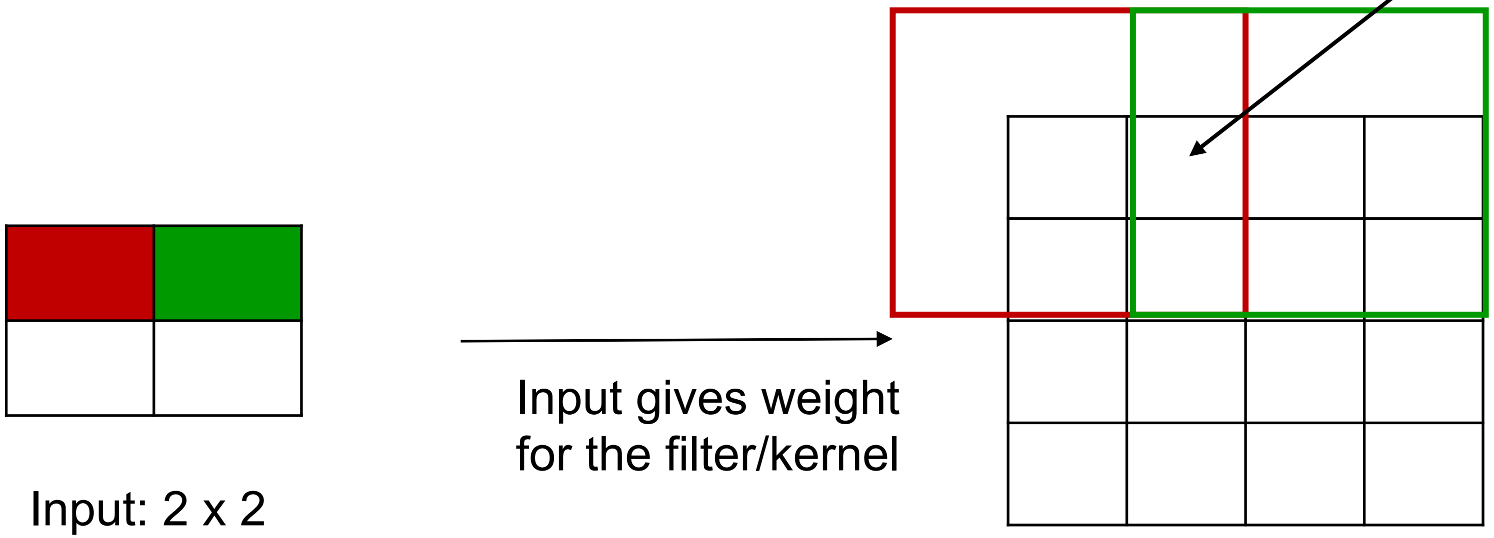
Filter/Kernel moves 2 pixels in the output for every one pixel in the input.

Stride gives ratio between movement in the output and the input.

Learning Upsampling: Transpose Convolution

3 x 3 Transpose Convolution; stride 2, padding 1

Sum where output overlaps



Input: 2 x 2

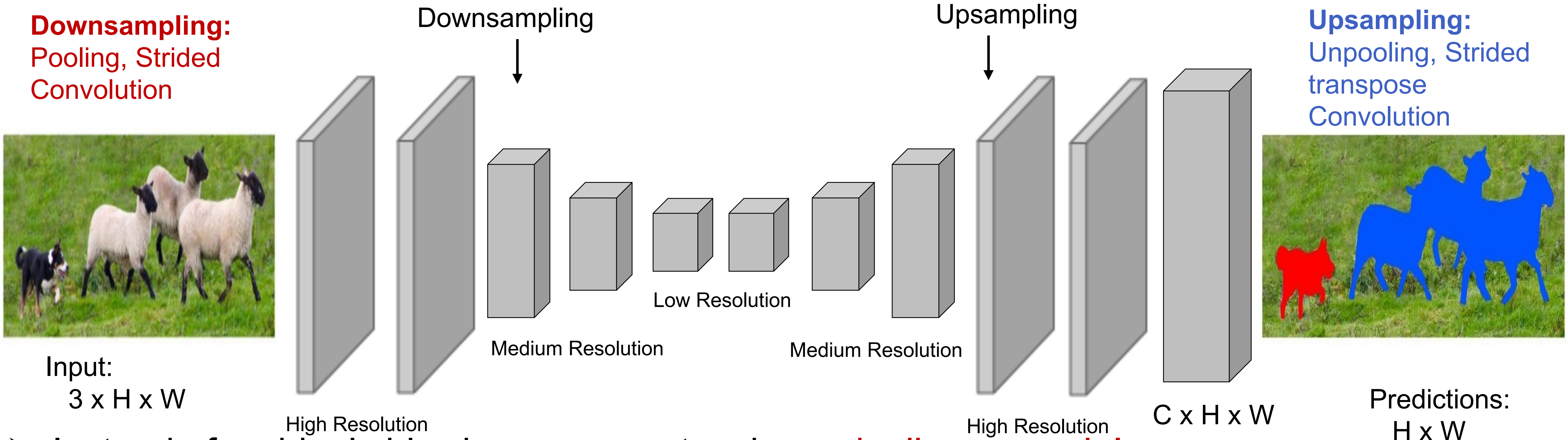
Output: 4 x 4

Filter/Kernel moves 2 pixels in the output for every one pixel in the input.

Stride gives ratio between movement in the output and the input.

Fully Convolutional Networks for Semantic Segmentation

- Design a network having convolutional layers, with downsampling and upsampling inside the network (learning in an end-to-end manner)

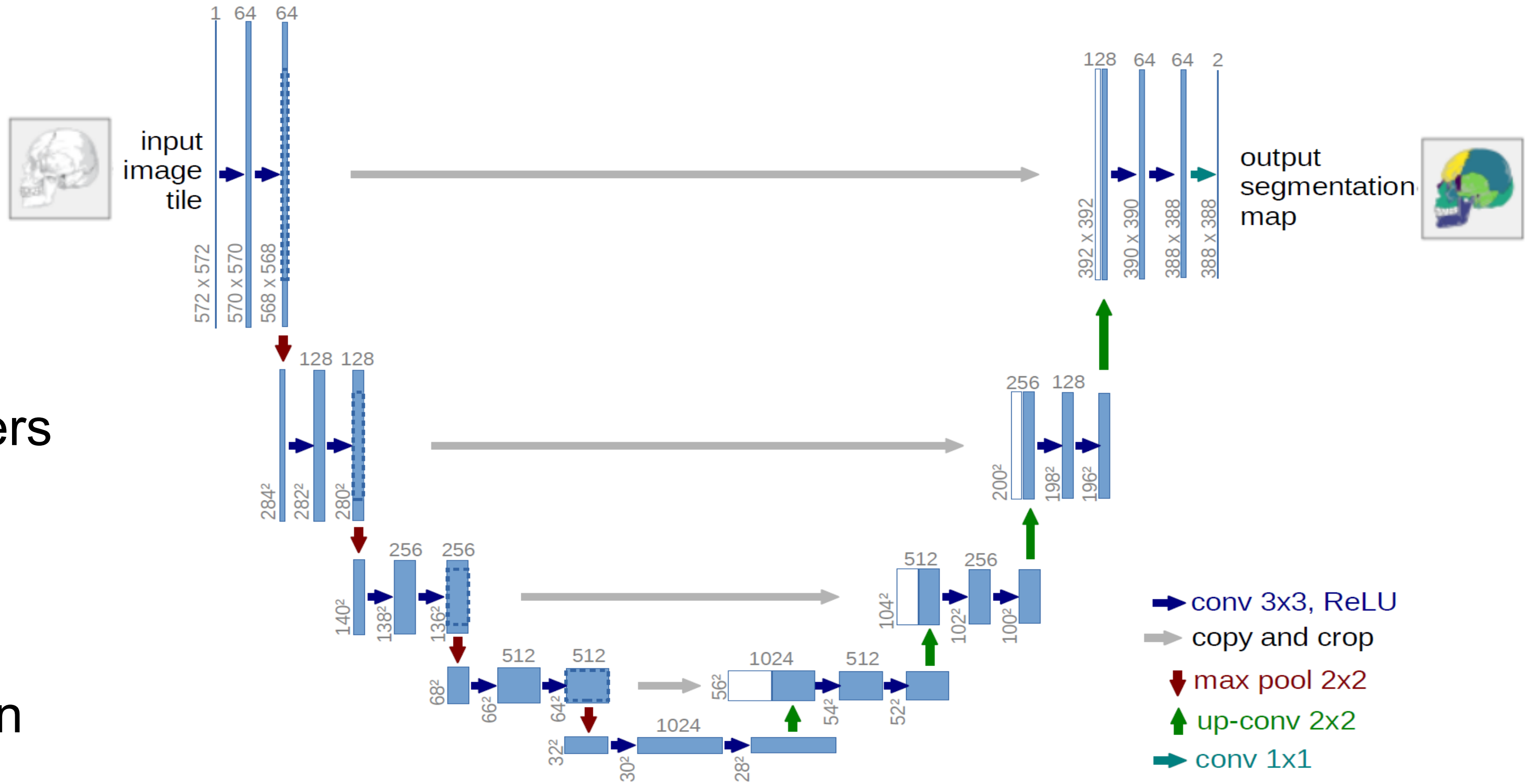


- Instead of suddenly blowing up our network, gradually upsample!
- Learning the upsampling with convolutions!

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

U-Net

- Combines all the previous improvements but also add skip-connections.

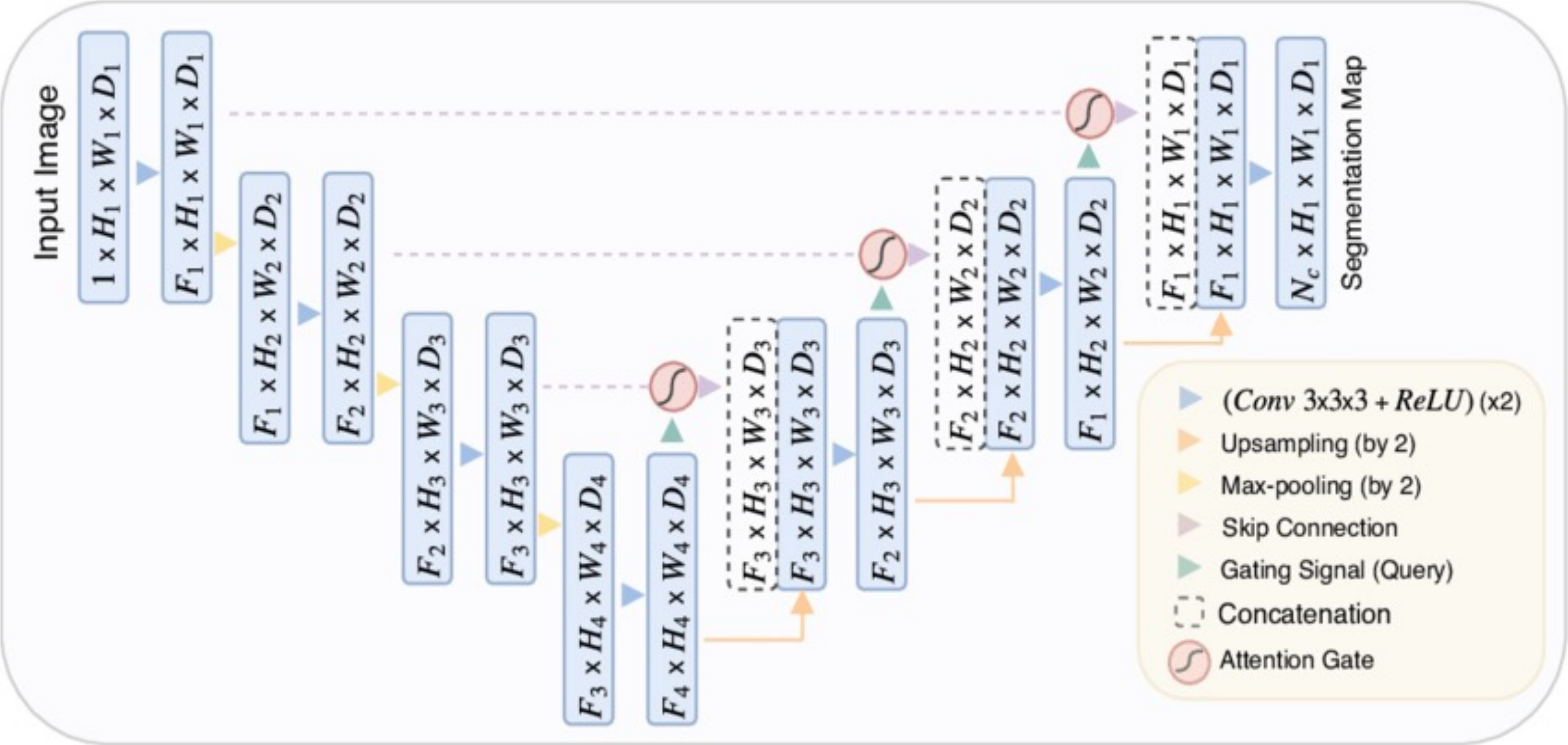


- Skip connections allow outputs from previous layers to feed in directly as input to later layers.
- U-Net learns segmentation in an end-to-end manner.

Ronneberger et al. (2015). U-net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015

U-Net variants

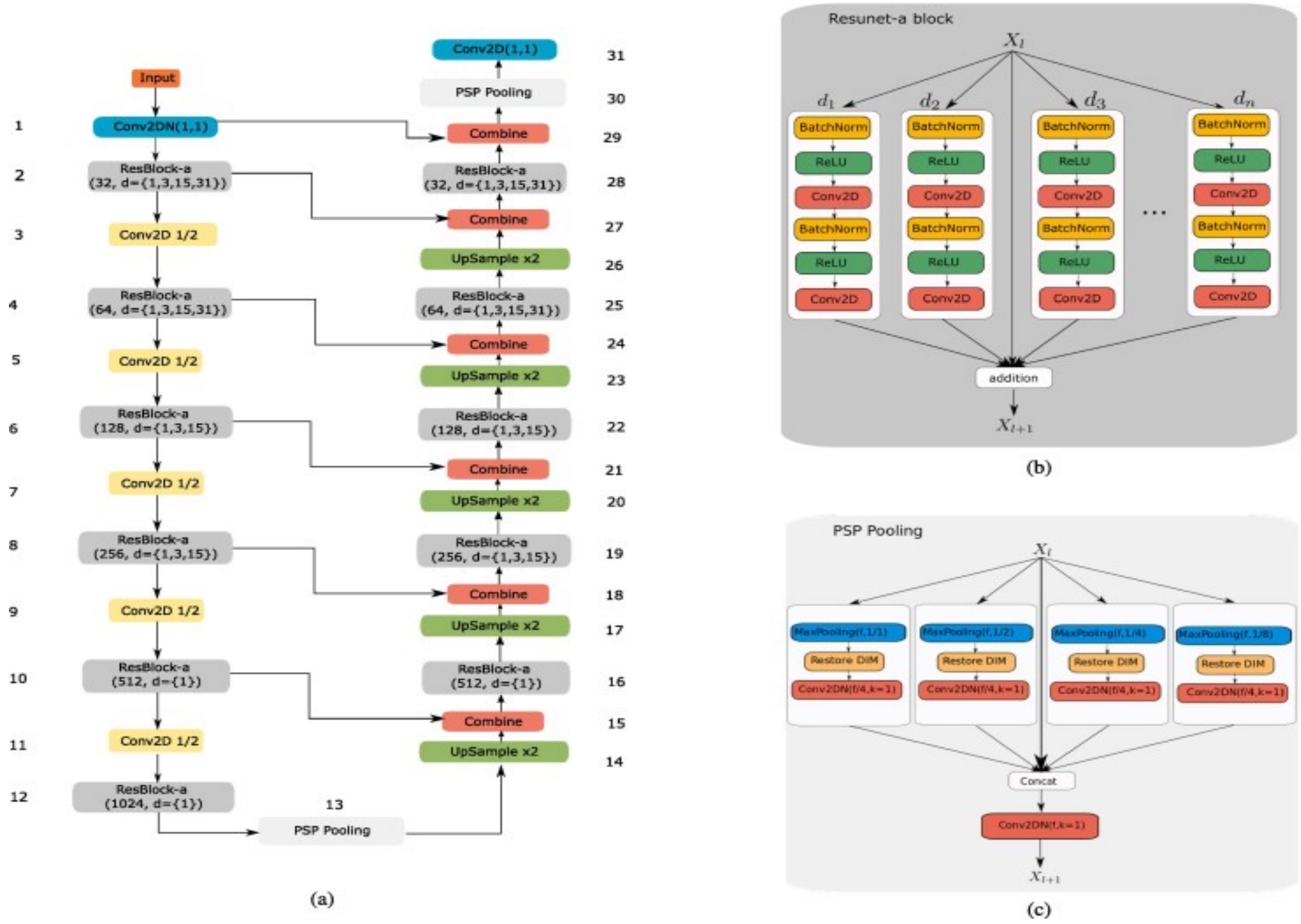
- Attention U-Net



Oktay et al., (2018). "Attention U-Net: Learning where to look for the Pancreas", MIDL 2018

U-Net variants

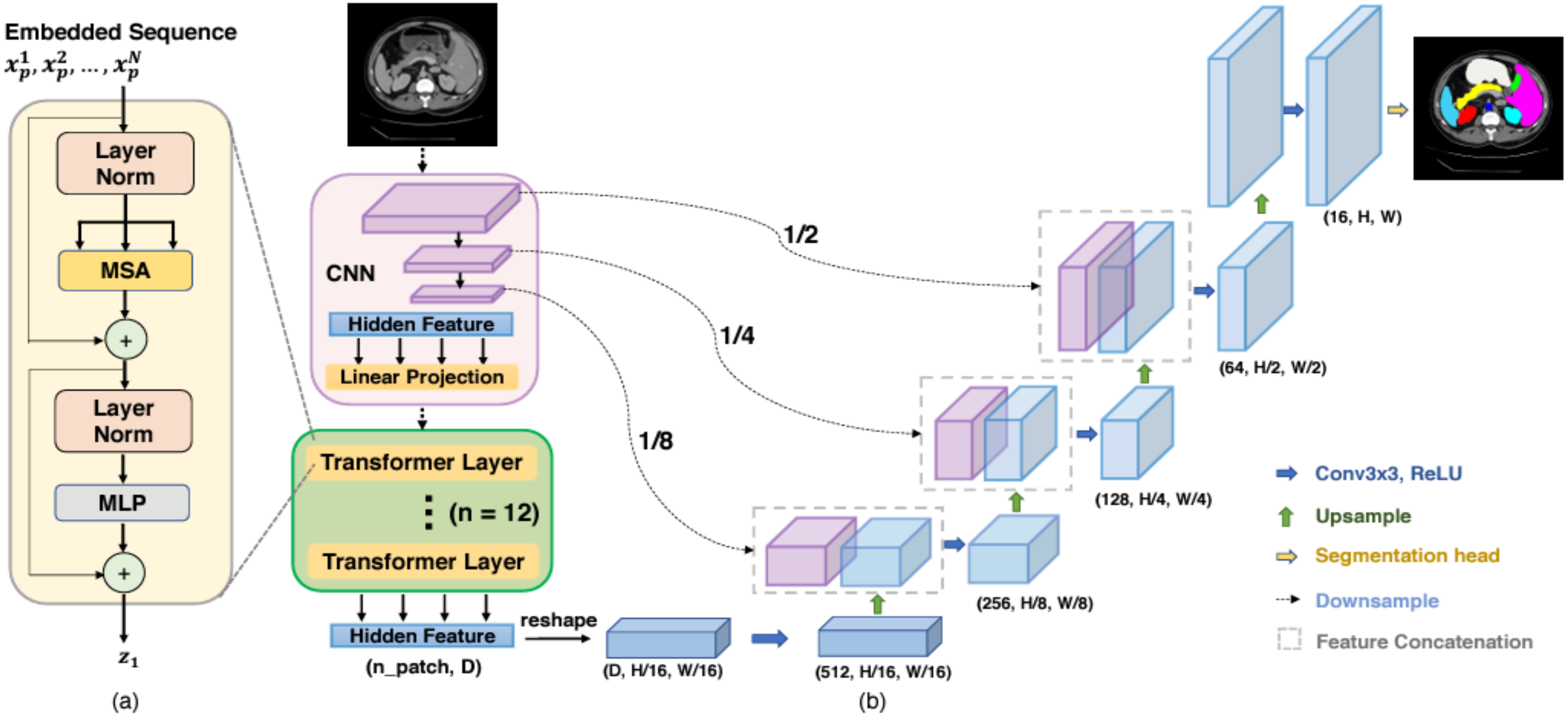
- ResUNet



Diakogiannis et al., (2019). “ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data”, ISPRS Journal of Photogrammetery and Remote Sensing

U-Net variants

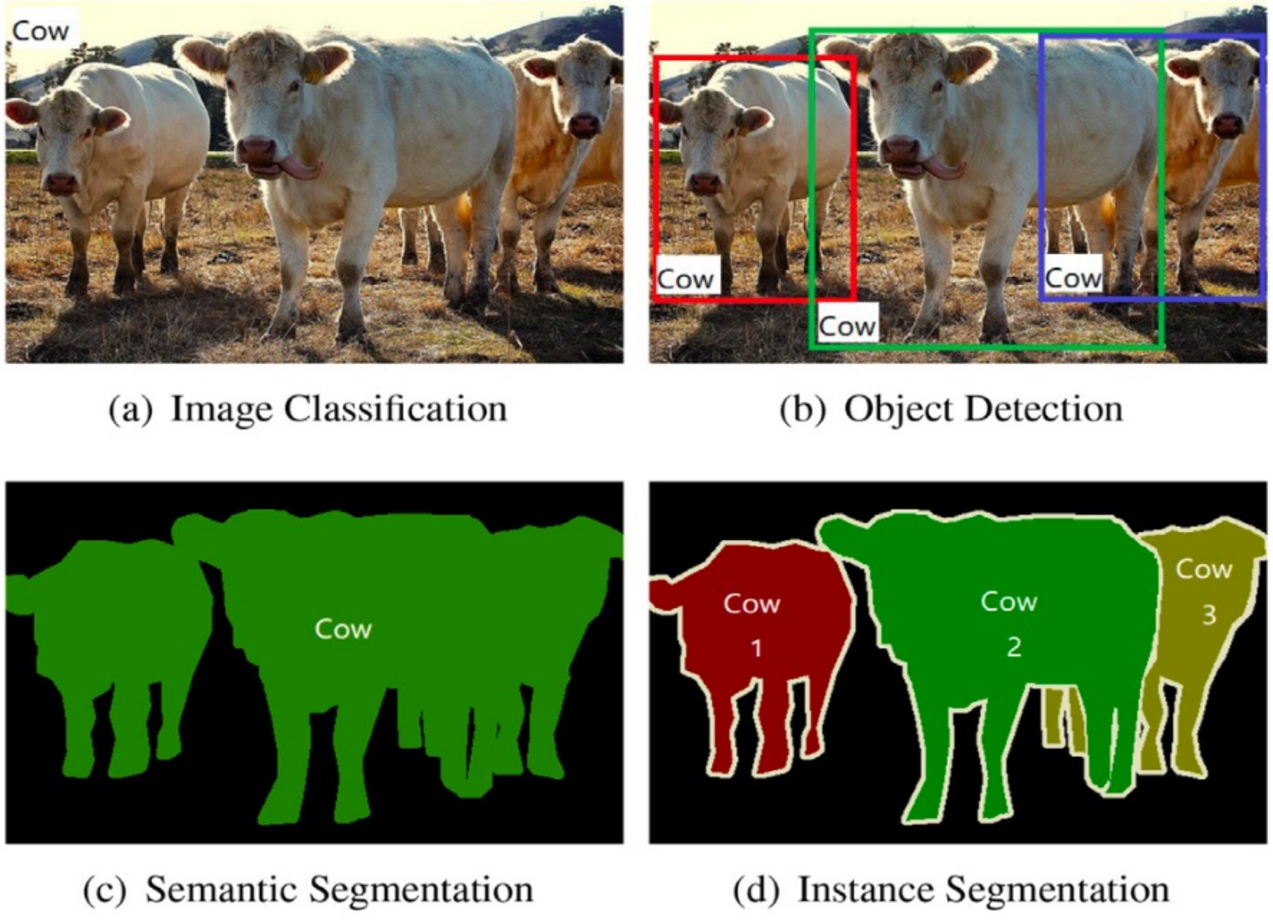
➢ TransUNet



Chen et al., (2021). “TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation”, ArXiv

Instance Segmentation

- Differentiate instances

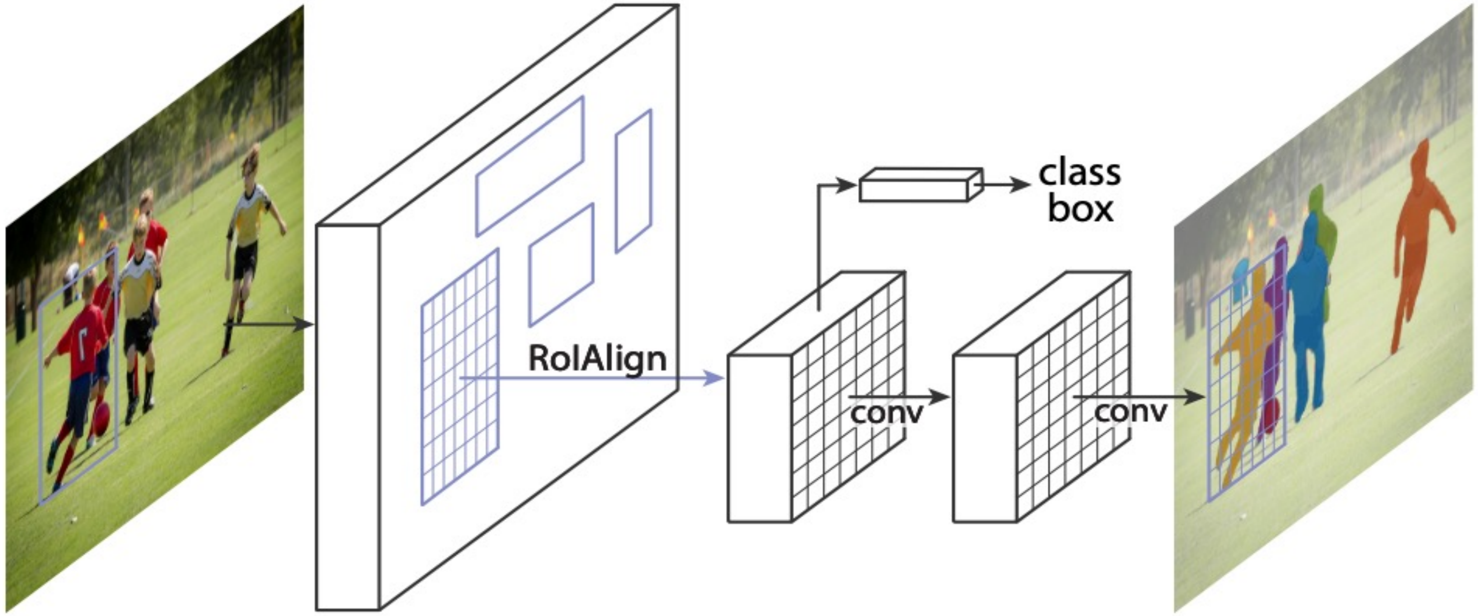


(a) Image Classification (b) Object Detection

(c) Semantic Segmentation (d) Instance Segmentation

Mask R-CNN

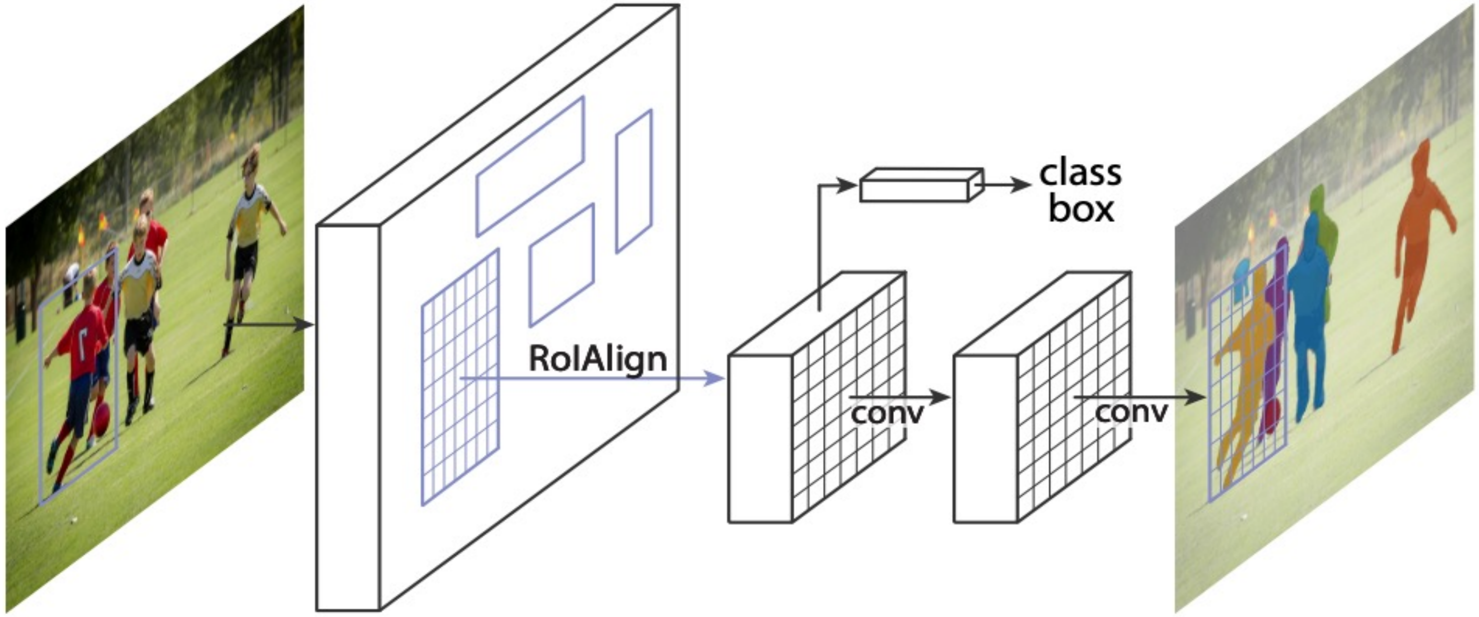
- It is an extension of the Faster R-CNN framework for solving instance segmentation problem.
- Detect and delineate each object in an image in a fine-grained pixel level.



He et al., “Mask R-CNN”. ICCV 2017.

Mask R-CNN

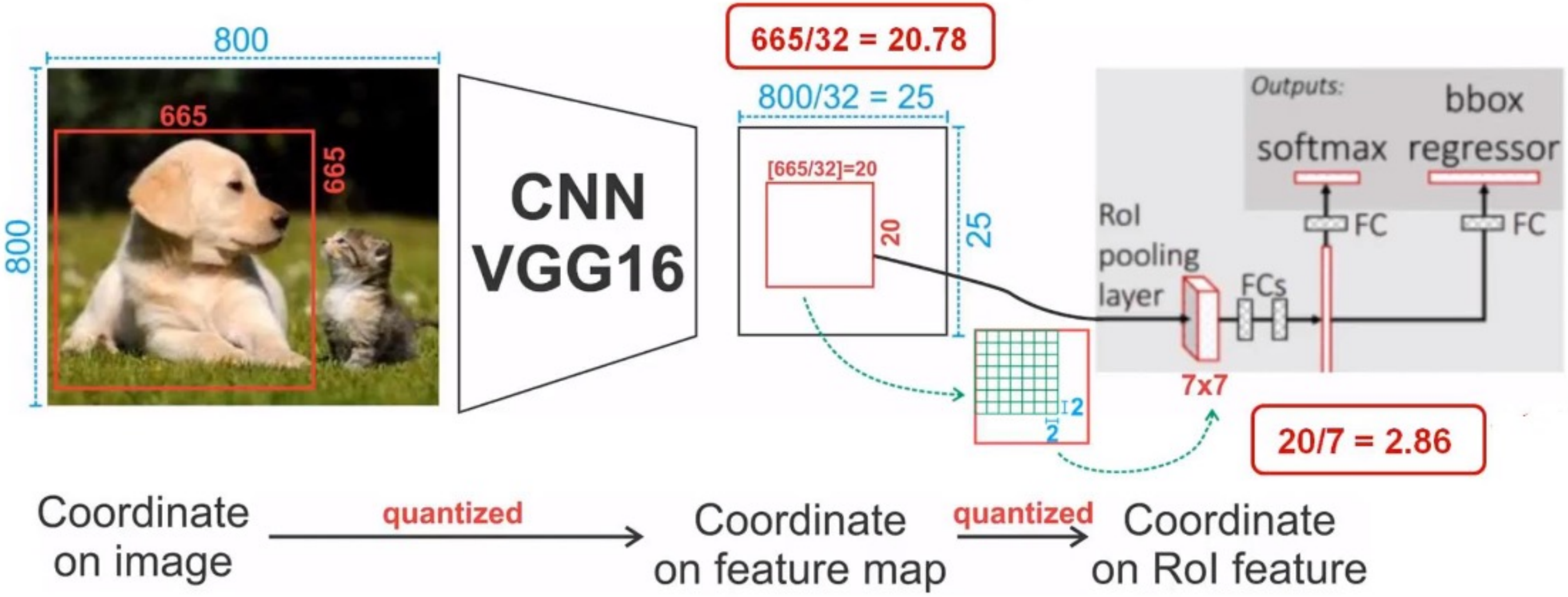
- It is an extension of the Faster R-CNN framework for solving instance segmentation problem.
- Detect and delineate each object in an image in a fine-grained pixel level.
- Mask R-CNN outputs a binary mask for each RoI on top of the Faster R-CNN



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

Need for RoI Align

One pixel in RoI means many pixels in the original image

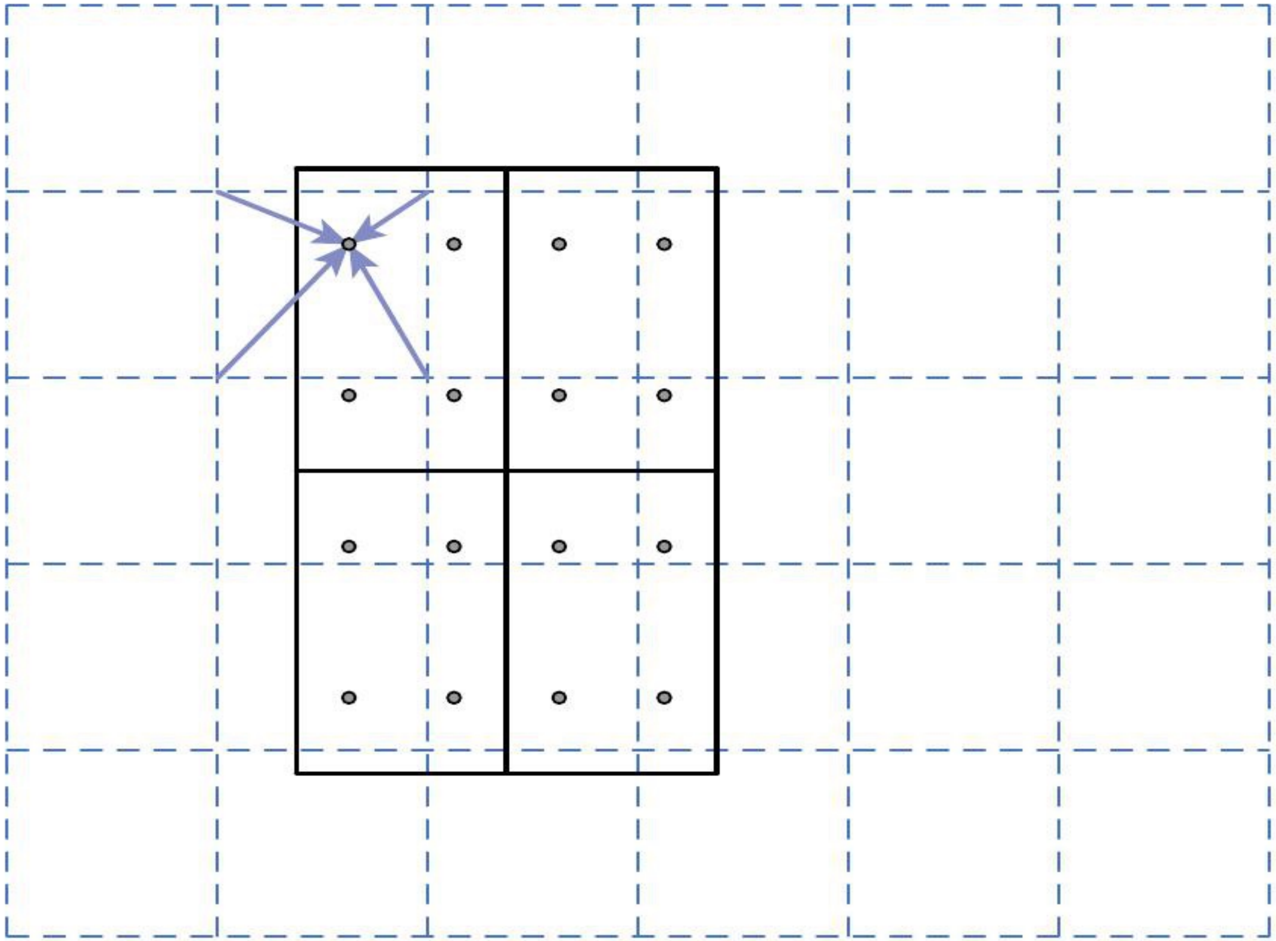


https://www.youtube.com/watch?v=UI25zSysk2A&list=PLkRkKTC6HZMxZrxnHUDYSLiPZxiUUFD2C&index=2

RoIAlign

- To extract the pixel-pixel mask, the RoI to be well aligned to preserve the explicit per-pixel spatial correspondence
- RoIPool: Quantize a floating number RoI to the discrete granularity of the feature map
- RoIAlign: bilinear interpolation to compute the exact values of the input features.
- Multi-task loss on each sampled RoI:

$$L = L_{cls} + L_{box} + L_{mask}$$



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

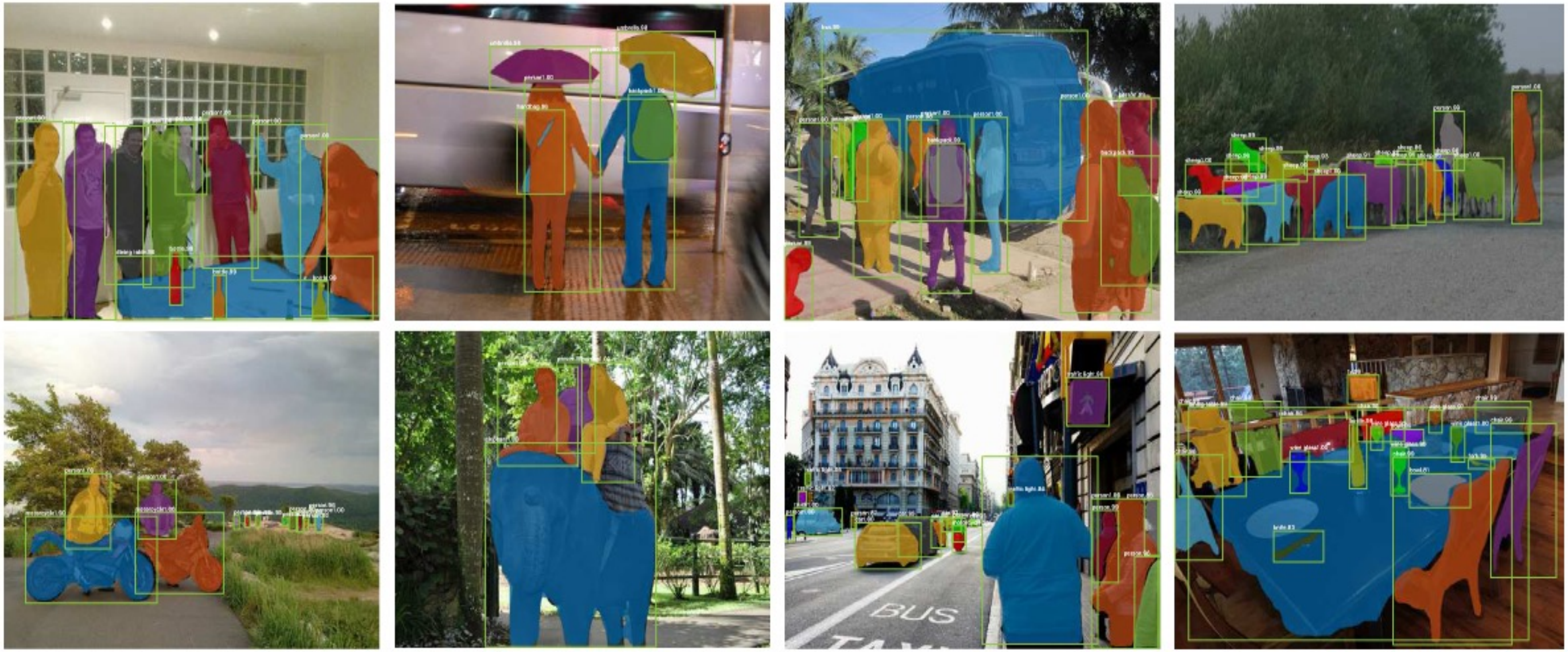
Mask R-CNN Architecture

- Architecture has two parts:
  - Backbone architecture: Used for feature extraction
  - Network Head: Comprises of object detection and segmentation
- Backbone architecture
  - ResNet
  - ResNeXt: Depth 50 and 101 layers
  - Feature Pyramid Network (FPN)
- Network Head: Use almost the same architecture as Faster R-CNN but add convolution mask prediction branch.

He et al., “Mask R-CNN”. ICCV 2017.

Mask R-CNN Results

➢ Results on MS COCO test set; based on ResNet-101.



He et al., “Mask R-CNN”. ICCV 2017.

Video Understanding

Video

- A sequence of images
- 4D tensor:
  - T x 3 x H x W ; or
  - 3 x T x H x W

Challenges in processing videos

- Capturing the information across frames
- Huge computational cost!
- Videos have approximately 30 frames per second (fps)

Video: T x 3 x H x W

Size of uncompressed video (3 bytes per pixel)

SD video (640 x 480): ~ 1.5 GB per minute

HD video (1920 x 1080): ~ 10 GB per minute

Source: Weizmann Dataset. Gorelick et al. “Actions as Space-Time Series”. TPAMI 2007

Challenges in processing videos

- ➢ Huge computational cost!
- ➢ Videos have approximately 30 frames per second (fps)

Video: T x 3 x H x W

Size of uncompressed video (3 bytes per pixel)

SD video (640 x 480): ~ 1.5 GB per minute

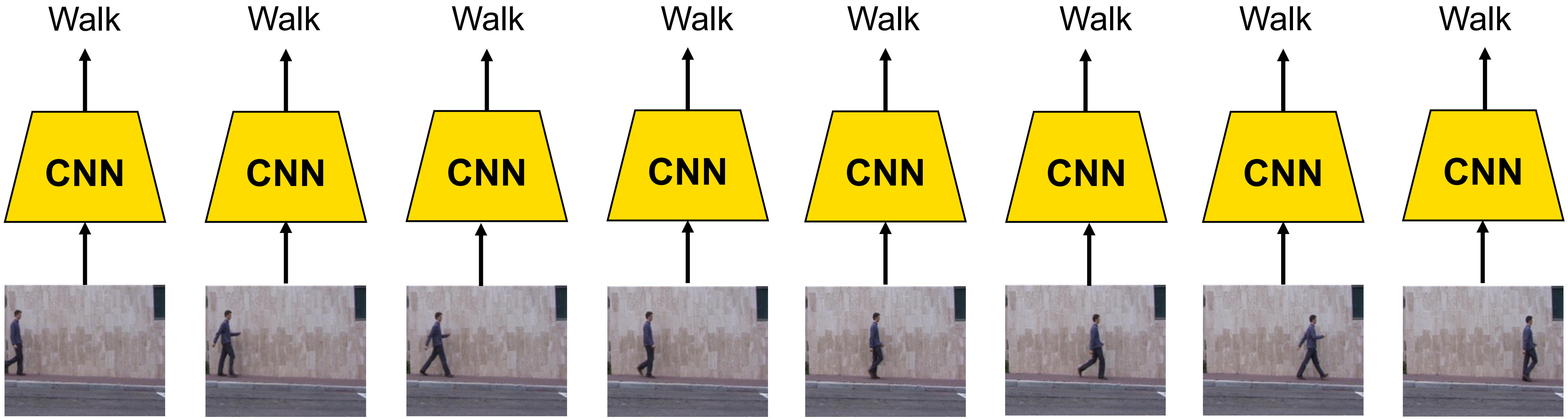
HD video (1920 x 1080): ~ 10 GB per minute

Solution: Train on short clips
(low fps and low spatial resolution)

Source: Weizmann Dataset. Gorelick et al. “Actions as Space-Time Series”. TPAMI 2007

Video Classification

➢ Simple approach: Apply 2D CNNs to classify frames

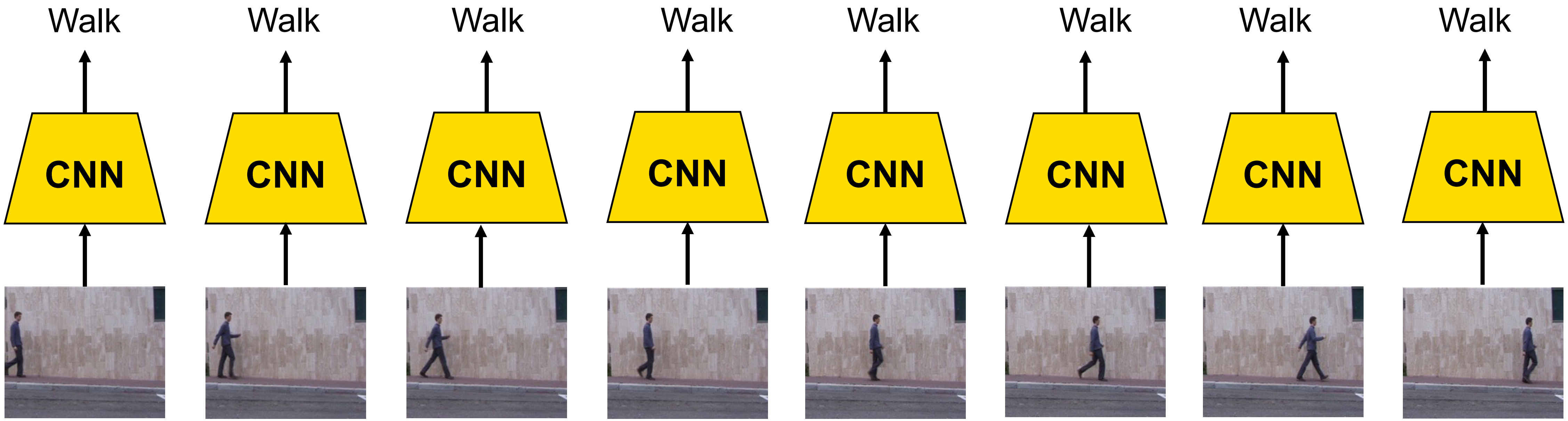


Very strong baseline for video classification!

Source: Weizmann Dataset. Gorelick et al. “Actions as Space-Time Series”. TPAMI 2007

Video Classification

- Simple approach: Apply 2D CNNs to classify frames



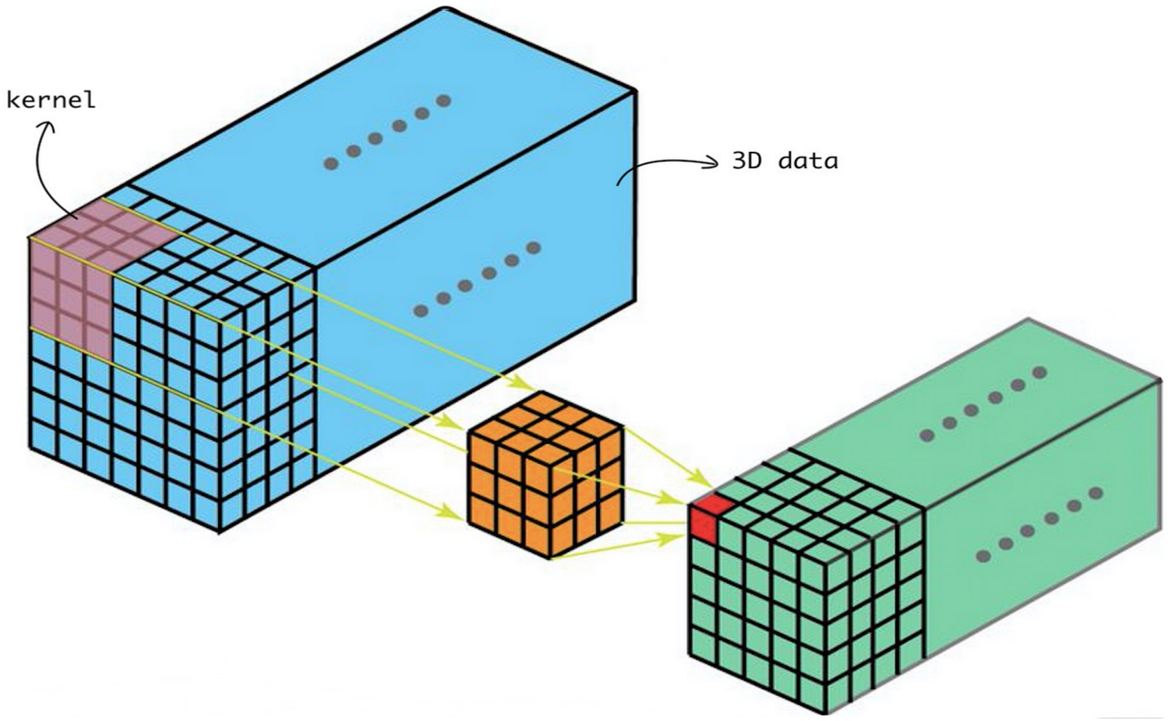
Fusion techniques:

Early Fusion
Mid-level Fusion
Late Fusion

Source: Weizmann Dataset. Gorelick et al. “Actions as Space-Time Series”. TPAMI 2007

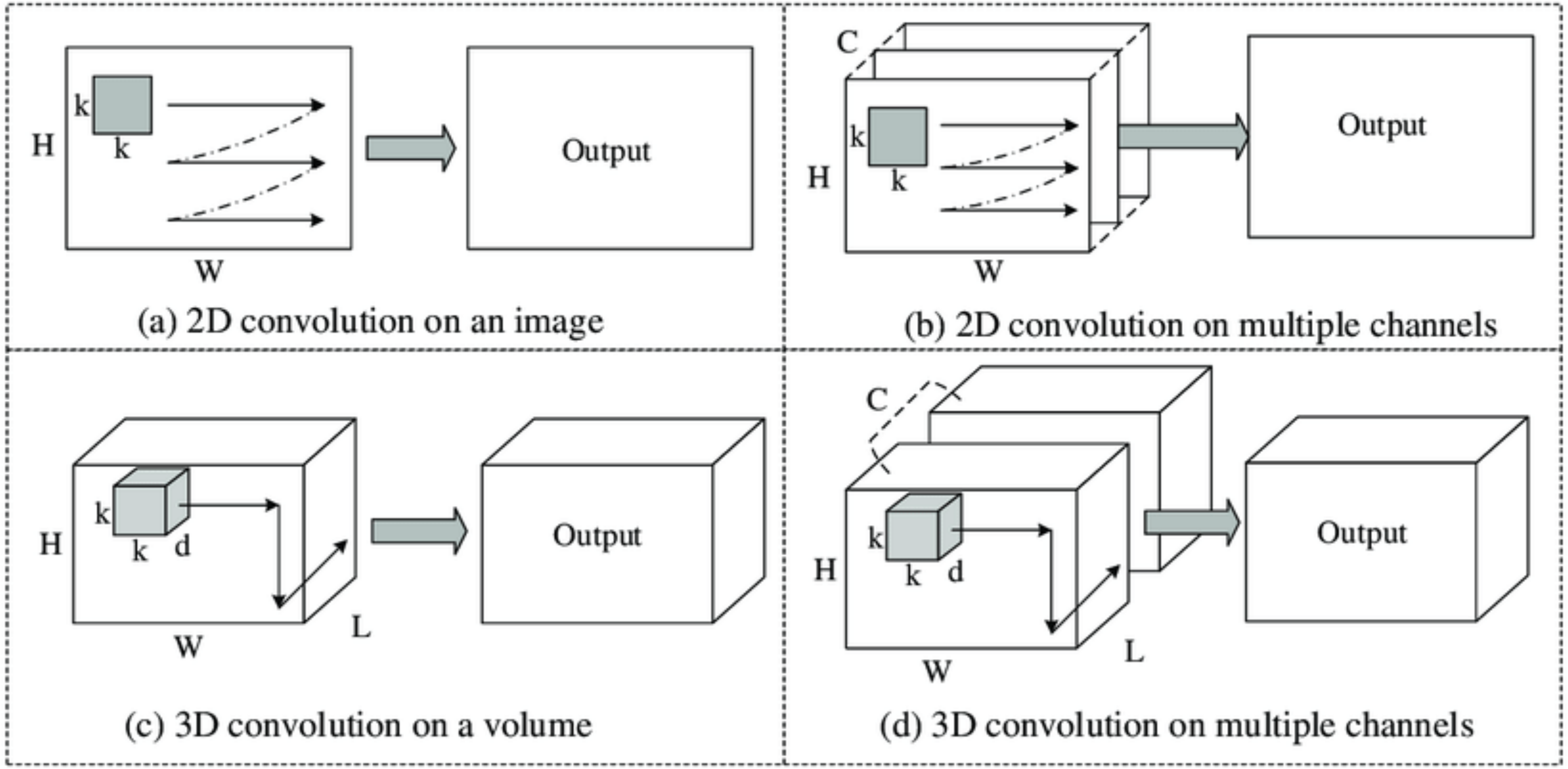
3D CNN

- How can be process entire clip?
- Idea: Use 3D versions of convolution and pooling to slowly fuse temporal information over the course of the network.



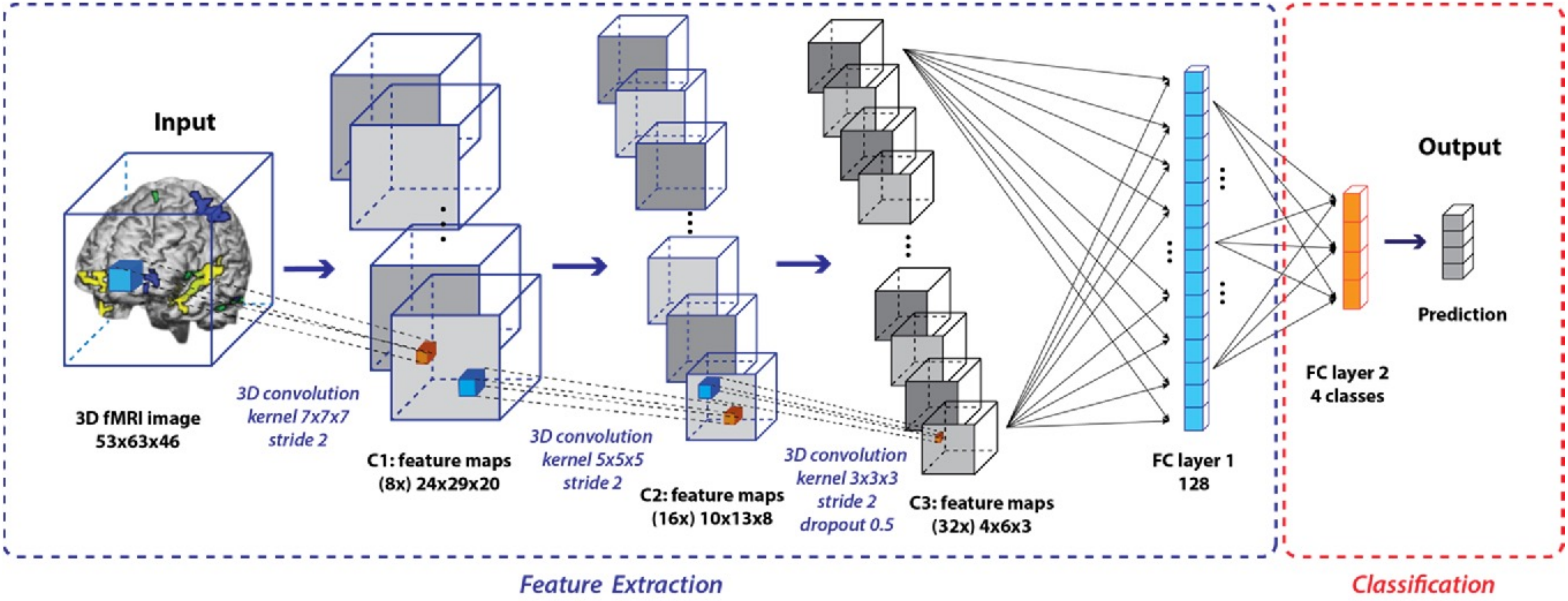
Source: https://towardsdatascience.com/understanding-1d-and-3d-convolution-neural-network-keras-9d8f76e29610

2D vs 3D Convolution



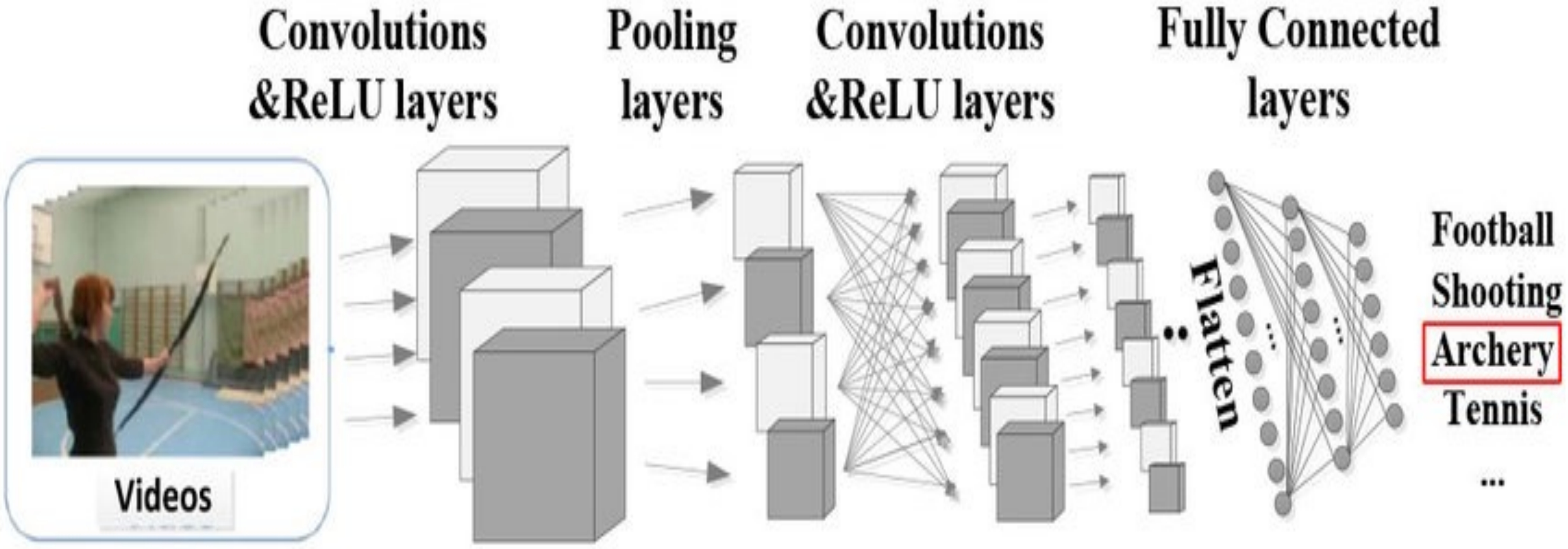
Source: Liu et al. A Uniform Architecture Design for Accelerating 2D and 3D CNNs on FPGAs

Classifying 3D data



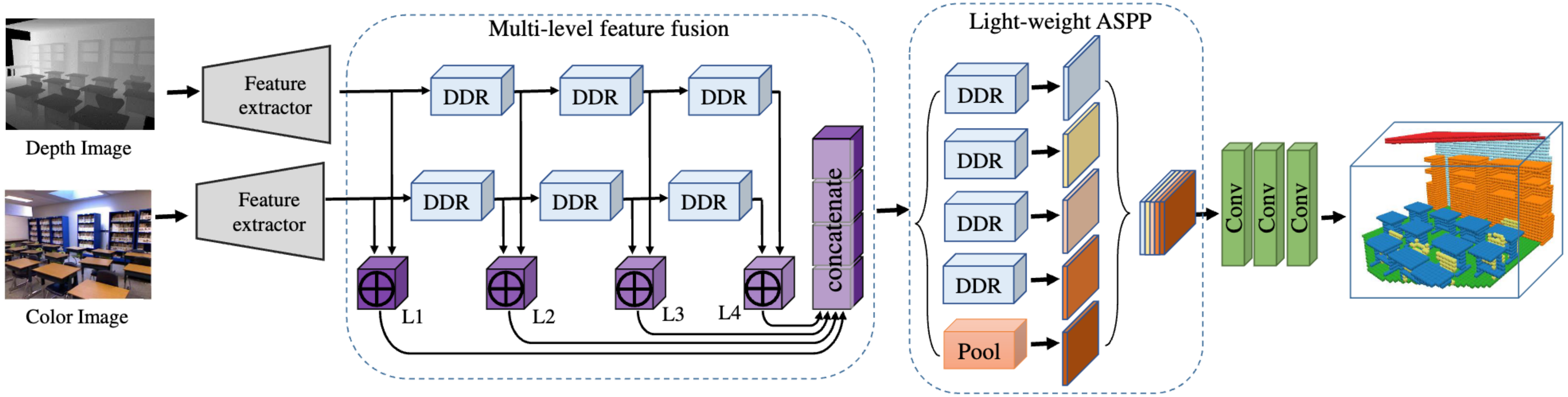
Source: Vu et al. "3D CNN for feature extraction and classification of fMRI volumes". PRNI 2018.

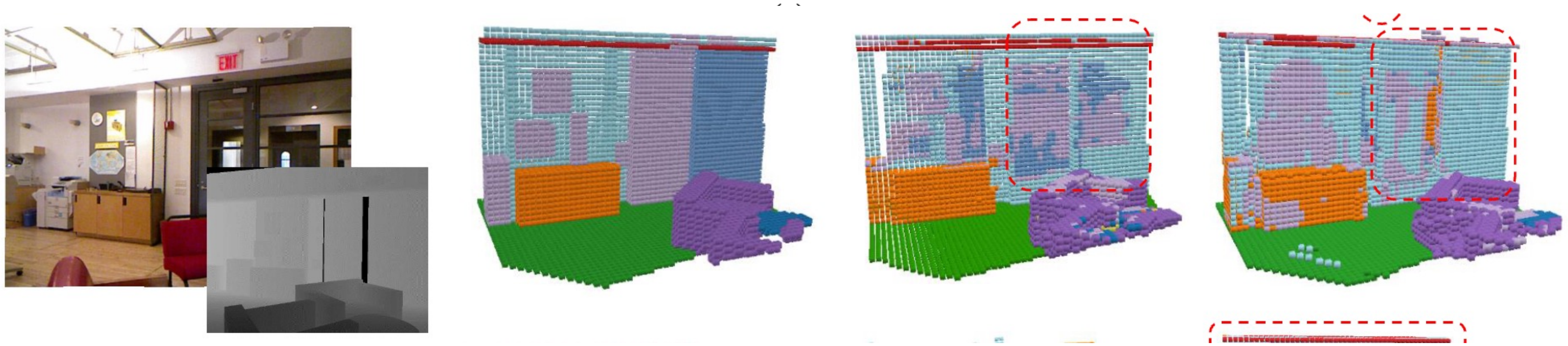
3D CNN for video classification



Source: Vu et al. “3D CNN for feature extraction and classification of fMRI volumes”. PRNI 2018.

3D CNN for 3D Scene Understanding





Li, Jie, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. "Rgbd based dimensional decomposition residual network for 3d semantic scene completion." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7693-7702. 2019.

Video Datasets

Sports-1M Dataset

Large-scale Video Classification with Convolutional Neural Networks

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, Li Fei-Fei

This sports action recognition dataset contains 1 million videos from 487 classes of sports, such as basketball, soccer, and ice hockey.

Source: https://cs.stanford.edu/people/karpathy/deepvideo/

Video Datasets

UCF101- Action Recognition

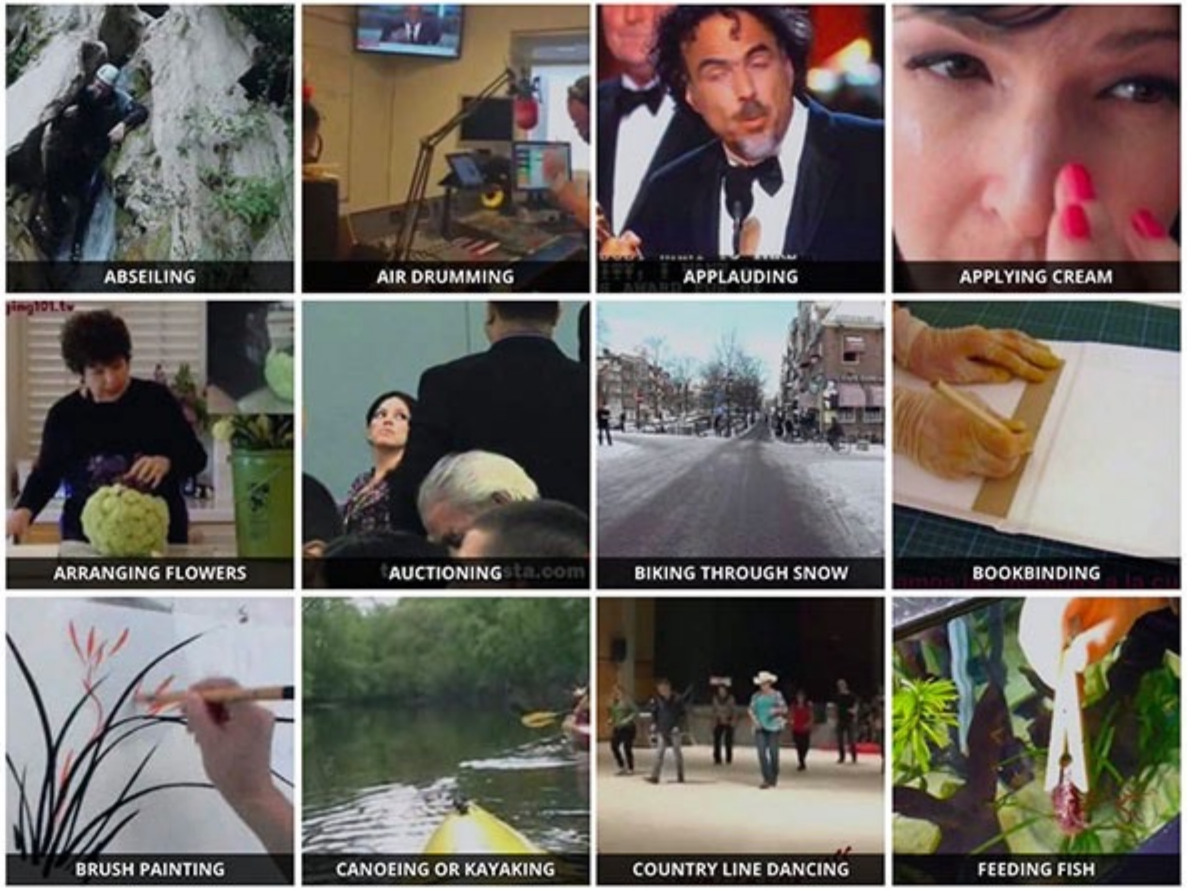
One of the most widely used video classification datasets is the UCF101 dataset, which consists of 13320 videos from 101 different action classes, such as walking, jogging, and playing soccer. The dataset is commonly used for evaluating the performance of video classification algorithms in a wide range of action recognition tasks.

Source: https://www.crcv.ucf.edu/data/UCF101.php

Video Datasets

Kinetics

A collection of large-scale, high-quality datasets of URL links of up to 650,000 video clips that cover 400/600/700 human action classes, depending on the dataset version. The videos include human-object interactions such as playing instruments, as well as human-human interactions such as shaking hands and hugging. Each action class has at least 400/600/700 video clips. Each clip is human annotated with a single action class and lasts around 10 seconds.

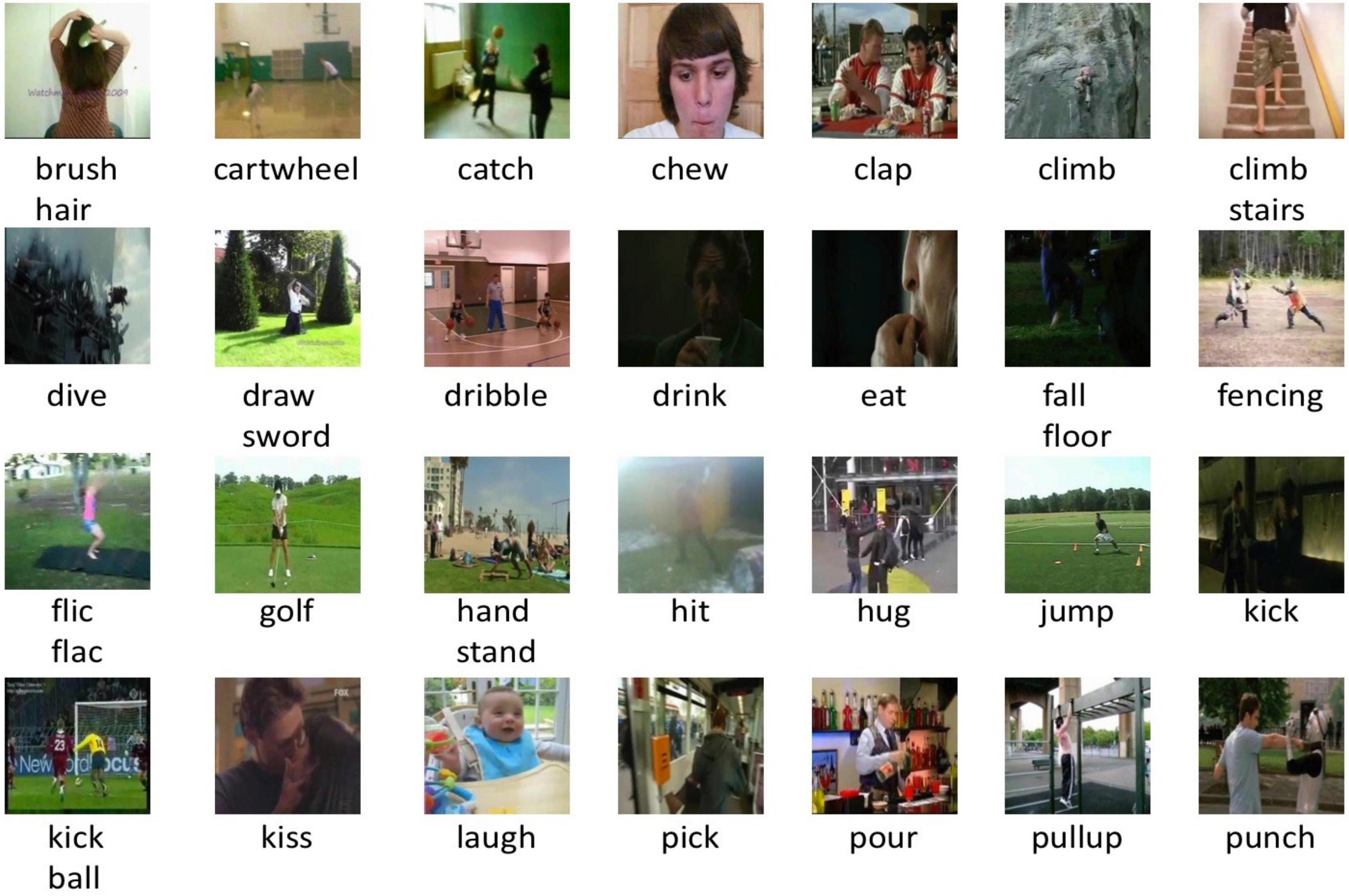


Source: https://www.deepmind.com/open-source/kinetics

Video Datasets

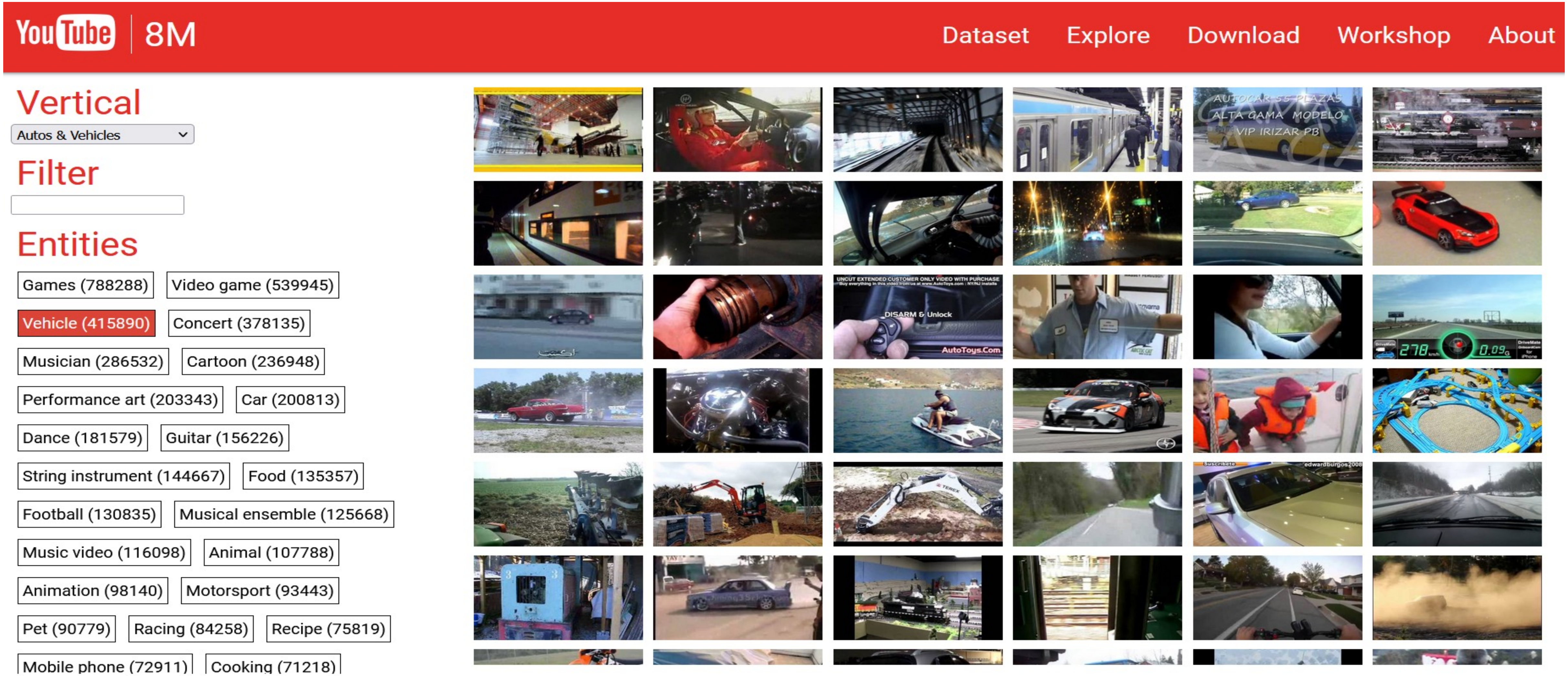
HMDB

This dataset contains 6849 videos from 51 different action classes. This dataset is similar to UCF101, but it has a smaller number of classes and videos.



Source: https://www.deepmind.com/open-source/kinetics

Video Datasets



Source: https://research.google.com/youtube8m/

Video Datasets

The Action Similarity Labelling (ASLAN) challenge

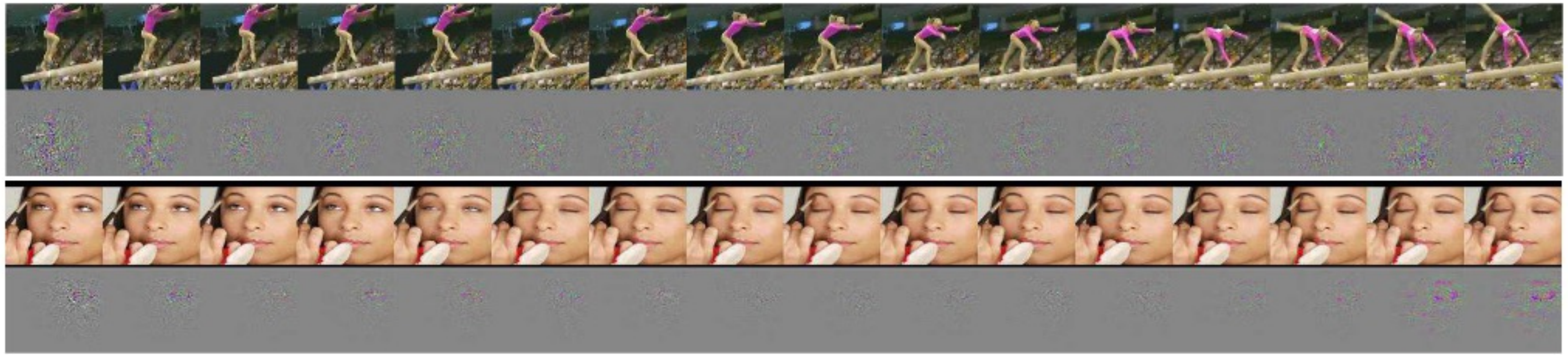
The ASLAN dataset consists of 3, 631 videos from 432 action classes. The task is to predict if a given pair of videos belong to the same or different action.

Source: https://talhassner.github.io/home/projects/ASLAN/ASLAN-main.html

C3D: Learning spatiotemporal features with 3D CNN

Learning Spatiotemporal Features with 3D Convolutional Networks

Du Tran[1,2], Lubomir Bourdev[1], Rob Fergus[1], Lorenzo Torresani[2], Manohar Paluri[1]
[1]Facebook AI Research, [2]Dartmouth College
{dutran,lorenzo}@cs.dartmouth.edu {lubomir,robfergus,mano}@fb.com



Visualization of C3D model:
The C3D model captures appearance for the first few frames but thereafter only attends to salient motion.

Source: Tran et al. “Learning Spatiotemporal Features with 3D Convolutional Networks”. ICCV 2015.

Layer	Size
Input	3 x 16 x 112 x 112
Conv1 (3 x 3 x 3)	64 x 16 x 112 x 112
Pool 1 (1 x 2 x 2)	64 x 16 x 56 x 56
Conv2 (3 x 3 x 3)	128 x 16 x 56 x 56
Pool 2 (2 x 2 x 2)	128 x 8 x 28 x 28
Conv3a (3 x 3 x 3)	256 x 8 x 28 x 28
Conv3b (3 x 3 x 3)	256 x 8 x 28 x 28
Pool 3 (2 x 2 x 2)	256 x 4 x 14 x 14
Conv4a (3 x 3 x 3)	512 x 4 x 14 x 14
Conv4b (3 x 3 x 3)	512 x 4 x 14 x 14
Pool 4 (2 x 2 x 2)	512 x 2 x 7 x 7
Conv5a (3 x 3 x 3)	512 x 2 x 7 x 7
Conv5b (3 x 3 x 3)	512 x 2 x 7 x 7
Pool 5	512 x 1 x 3 x 3
FC6	4096
FC7	4096
FC8	C

C3D Results

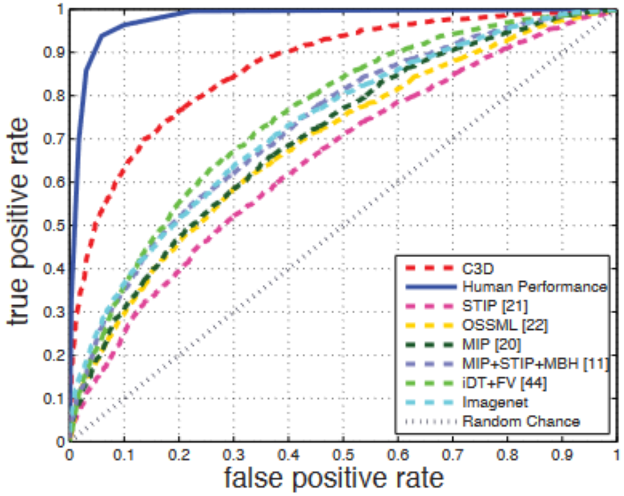
Action Recognition results on UCF-101dataset

Method	Accuracy (%)
Imagenet + linear SVM	68.8
iDT w/ BoW + linear SVM	76.2
Deep networks [18]	65.4
Spatial stream network [36]	72.6
LRCN [6]	71.1
LSTM composite model [39]	75.8
C3D (1 net) + linear SVM	82.3
C3D (3 nets) + linear SVM	**85.2**
iDT w/ Fisher vector [31]	87.9
Temporal stream network [36]	83.7
Two-stream networks [36]	88.0
LRCN [6]	82.9
LSTM composite model [39]	84.3
Conv. pooling on long clips [29]	88.2
LSTM on long clips [29]	88.6
Multi-skip feature stacking [25]	89.1
C3D (3 nets) + iDT + linear SVM	**90.4**

Action Similarity Labeling results on ASLAN

Method	Features	Model	Acc.	AUC
[21]	STIP	linear	60.9	65.3
[22]	STIP	metric	64.3	69.1
[20]	MIP	metric	65.5	71.9
[11]	MIP+STIP+MBH	metric	66.1	73.2
[45]	iDT+FV	metric	68.7	75.4
Baseline	Imagenet	linear	67.5	73.8
Ours	**C3D**	linear	**78.3**	**86.5**

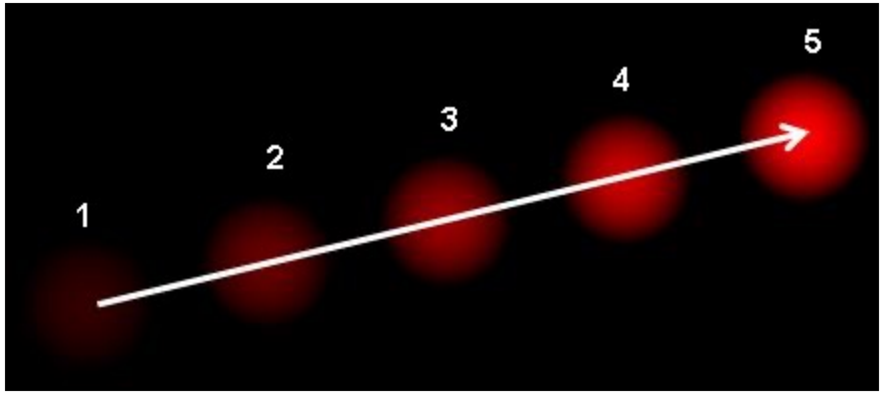
ROC curve of C3D evaluated on ASLAN

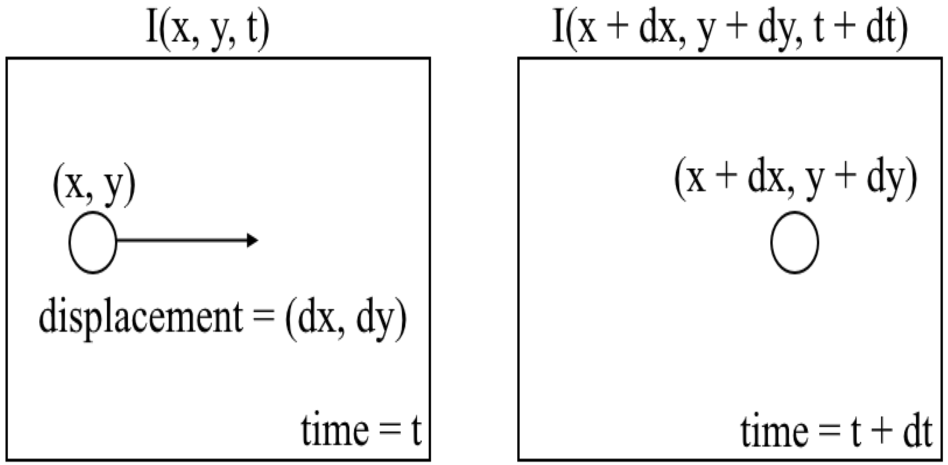


Source: Tran et al. “Learning Spatiotemporal Features with 3D Convolutional Networks”. ICCV 2015.

Recognizing Actions from Motion

- Actions can be recognized using only motion information
- **Optical Flow**:
  - It is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of object or camera. It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second.



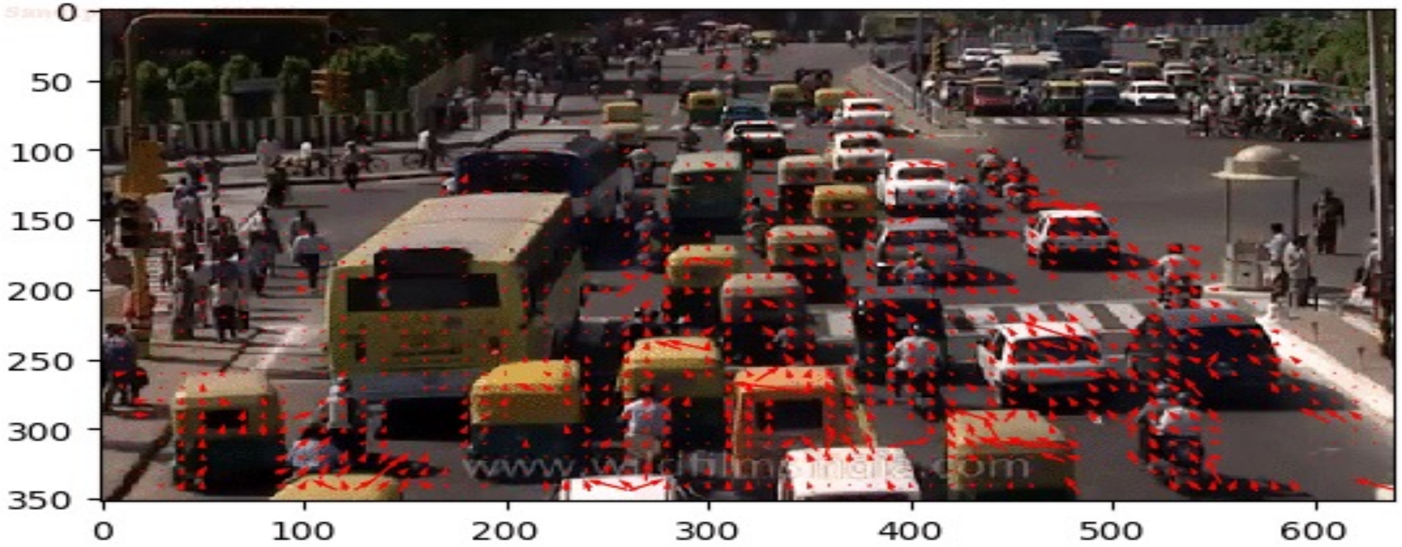


Source: OpenCV https://docs.opencv.org/3.4/d4/dee/tutorial_optical_flow.html

Optical Flow

- Useful in many applications:
  - Structure from Motion
  - Video Compression
  - Video Stabilization

Each arrow points in the direction of predicted flow of the corresponding pixel



Sparse vs Dense Optical Flow

Source: Introduction to Motion Estimation with Optical Flow https://nanonets.com/blog/optical-flow/

Optical Flow

➢ Useful in Action Recognition

Source: Introduction to Motion Estimation with Optical Flow https://nanonets.com/blog/optical-flow/

Optical Flow ConvNets

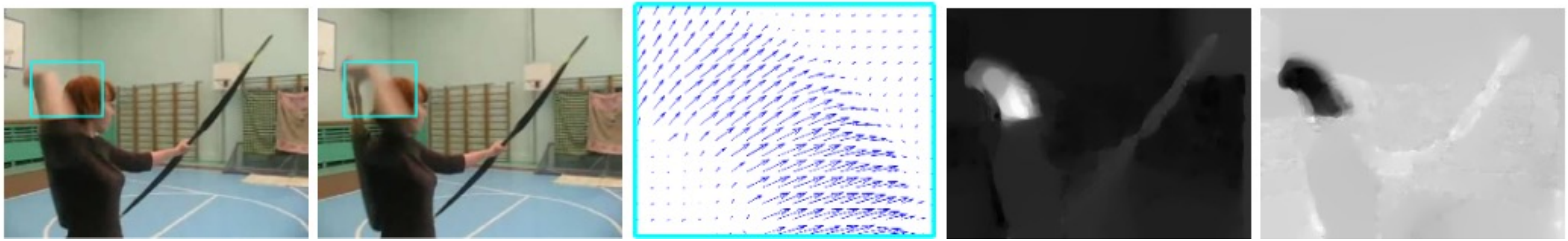
- Input to ConvNet is formed by stacking optical flow displacement fields between several consecutive frames.
- This explicitly describes the motion between video frames, making recognition easier.

(a), (b) : a pair of consecutive video frames

(c). a close-up of dense optical flow in the outlined area

(d). Horizontal component of the displacement vector field

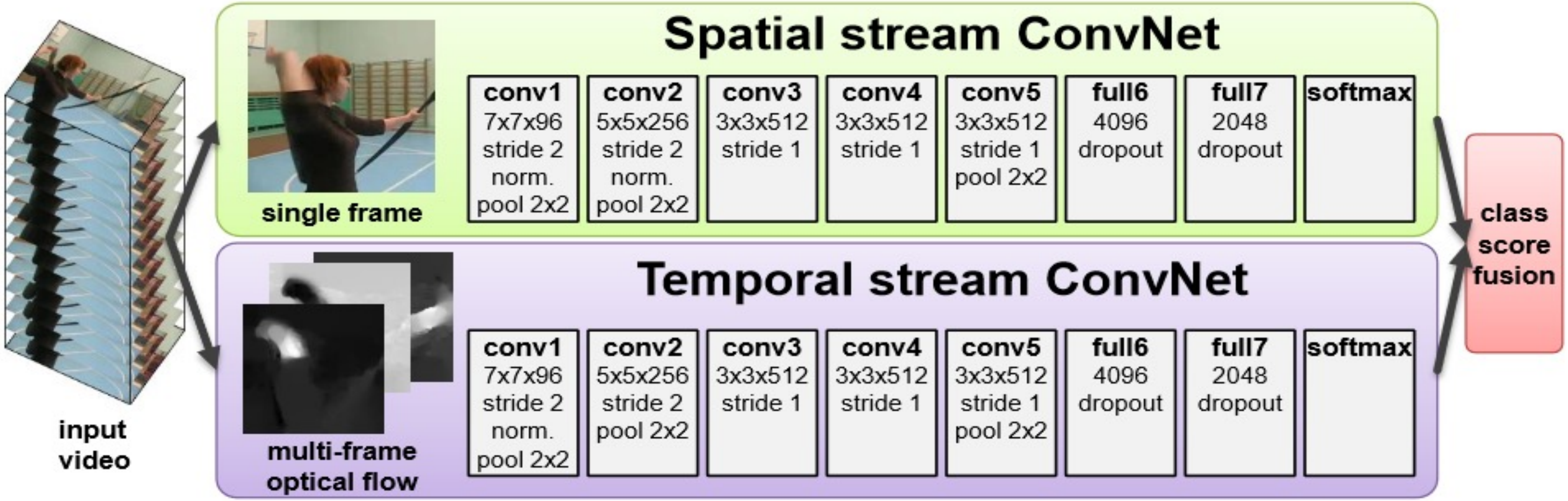
(d). Vertical component of the displacement vector field



Source: Simonyan and Zisserman. “Two-Stream Convolutional Networks for Action Recognition in Videos”, NeurIPS 2014.

Two-Stream Network for video classification

- Videos can naturally be decomposed into spatial and temporal components.
- The spatial component carries information about scenes and objects depicted in the video.
- The temporal component conveys the movement of the observer (the camera) and the objects.



Source: Simonyan and Zisserman. “Two-Stream Convolutional Networks for Action Recognition in Videos”, NeurIPS 2014.

Two-Stream Network Results

Mean accuracy on UCF101 and HMDB-51

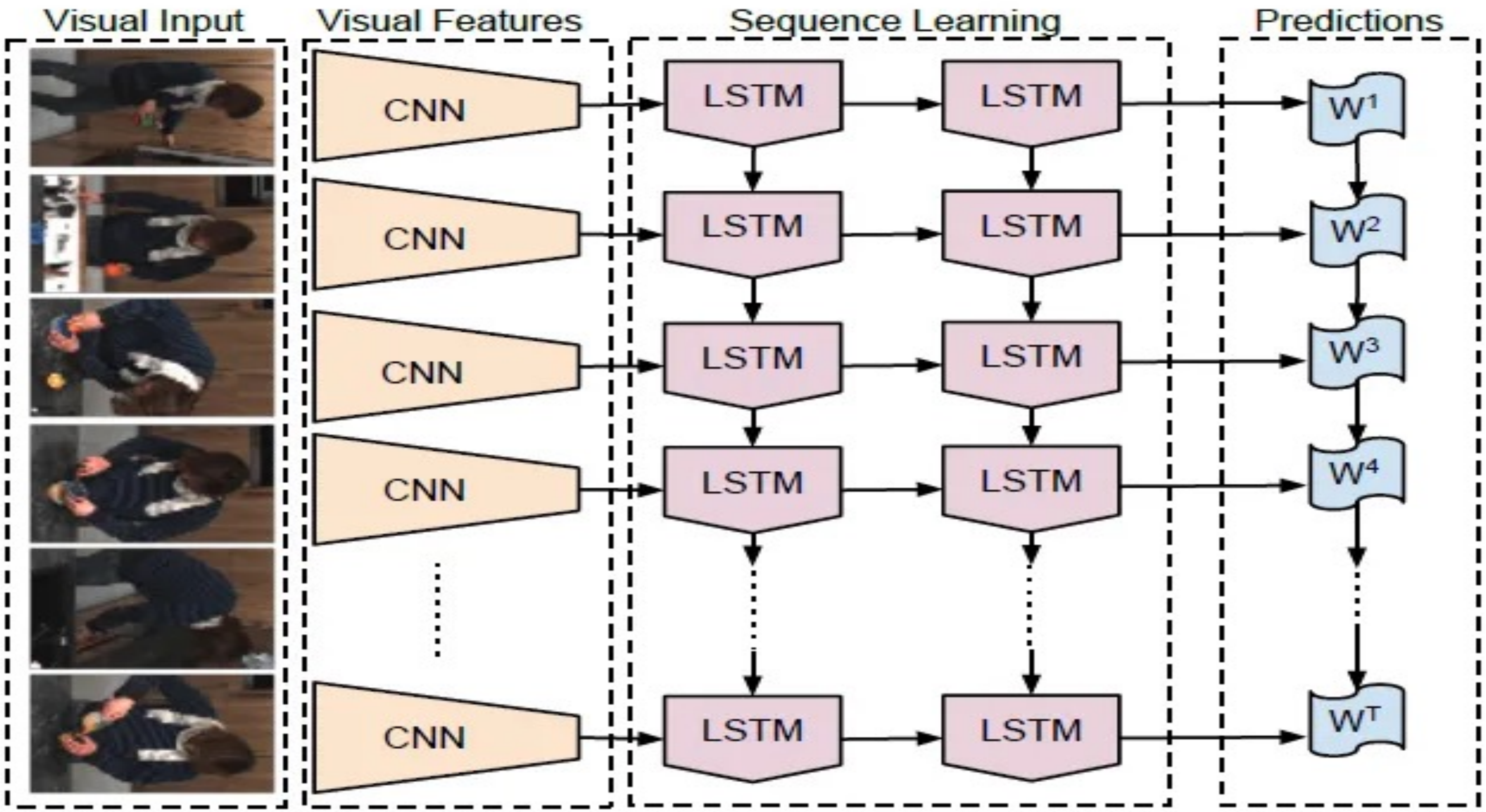
Method	UCF-101	HMDB-51
Improved dense trajectories (IDT) [26, 27]	85.9%	57.2%
IDT with higher-dimensional encodings [20]	**87.9%**	61.1%
IDT with stacked Fisher encoding [21] (based on Deep Fisher Net [23])	-	**66.8%**
Spatio-temporal HMAX network [11, 16]	-	22.8%
"Slow fusion" spatio-temporal ConvNet [14]	65.4%	-
Spatial stream ConvNet	73.0%	40.5%
Temporal stream ConvNet	83.7%	54.6%
Two-stream model (fusion by averaging)	86.9%	58.0%
Two-stream model (fusion by SVM)	**88.0%**	**59.4%**

Source: Simonyan and Zisserman. "Two-Stream Convolutional Networks for Action Recognition in Videos", NeurIPS 2014.

How to model Long-Term Temporal Dependency?

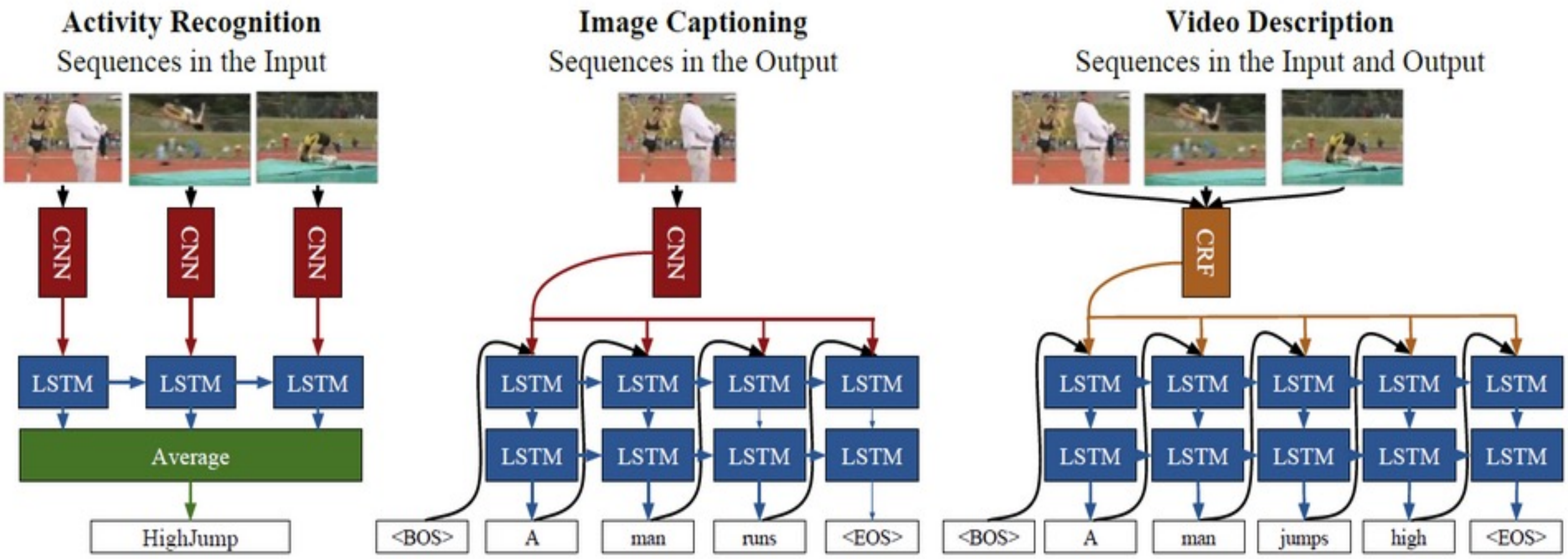
- Often salient information in videos are many frames apart.
- **Problem**: How can be model long-term temporal structure in videos?
- Recall:
  - Convolutional Neural Networks (CNNs) can capture local structure/local context
  - Recurrent Neural Networks (RNNs) can capture global structure/global context
- We can use a combination of CNNs + RNNs for modelling long-term temporal structure in videos.

Long-term Recurrent Convolutional Network (LRCN)



Source: Donahue et al.. "Long-term Recurrent Convolutional Networks for Visual Recognition and Description", CVPR 2015.

Long-term Recurrent Convolutional Network (LRCN)

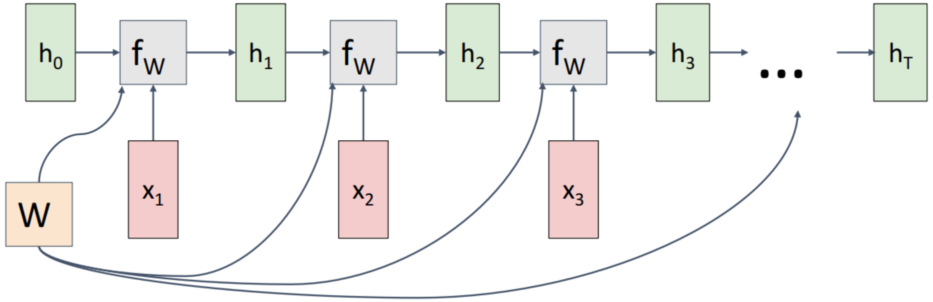


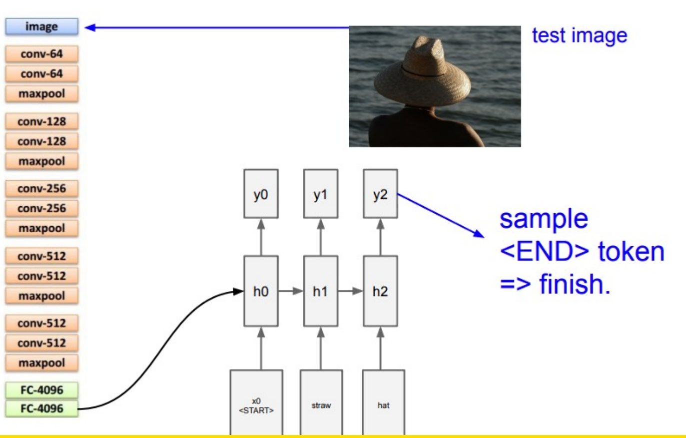
Source: Donahue et al.. "Long-term Recurrent Convolutional Networks for Visual Recognition and Description", CVPR 2015.

Recurrent Neural Networks (RNNs)

- Sequential modeling
- RNN, GRU, LSTM, …

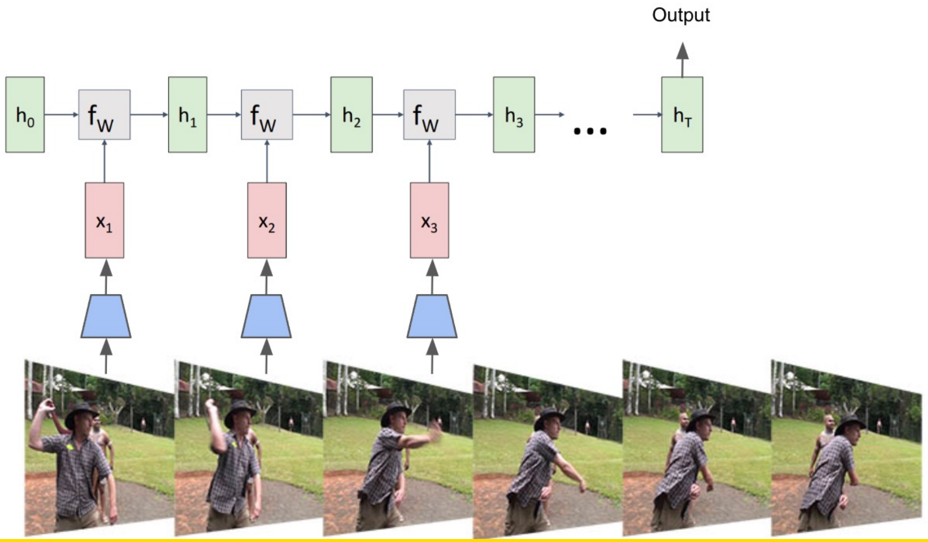
- Action recognition or video classification – can also be handled by 3D CNNs.
- Image captioning





A dog is running in the grass with a frisbee

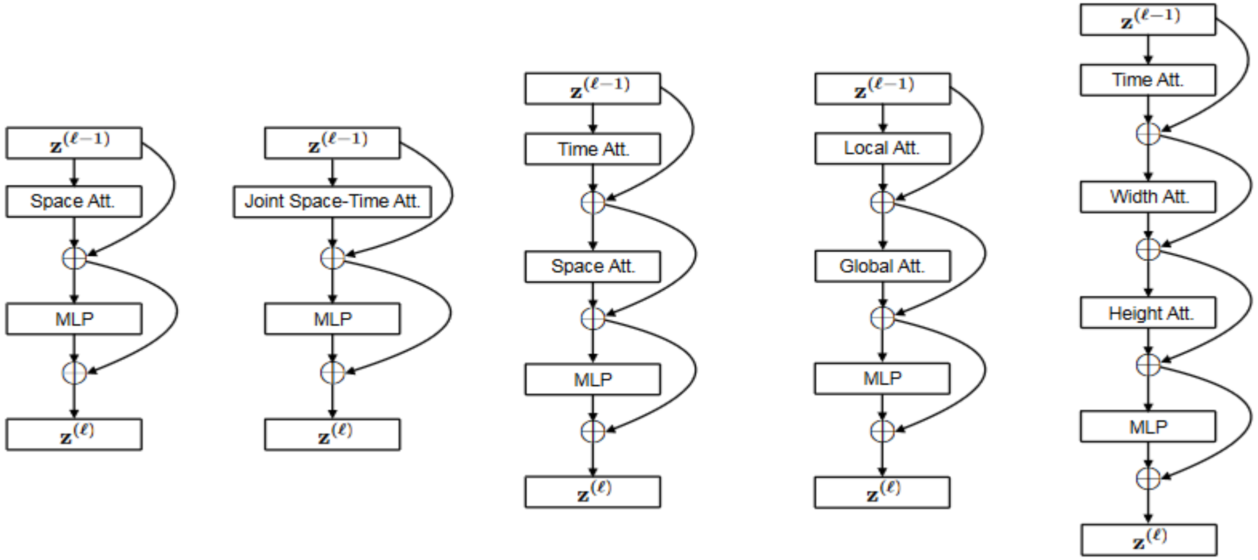
Two giraffes standing in a grassy field



Is Space-Time Attention All You Need for Video Understanding?

- **TimeSformer**: A convolution-free approach to video classification built exclusively on self-attention over space and time.
- It applies standard Transformer architecture to video by enabling spatio-temporal feature learning directly form sequence of frame-level patches.

Video self-attention blocks investigated in TimeSformer



Space Attention (S) | Joint Space-Time Attention (ST) | Divided Space-Time Attention (T+S) | Sparse Local Global Attention (L+G) | Axial Attention (T+W+H)

Source: Bertasius et al.. “Is Space-Time Attention All You Need for Video Understanding?”, ICML 2021.

TimeSformer Results

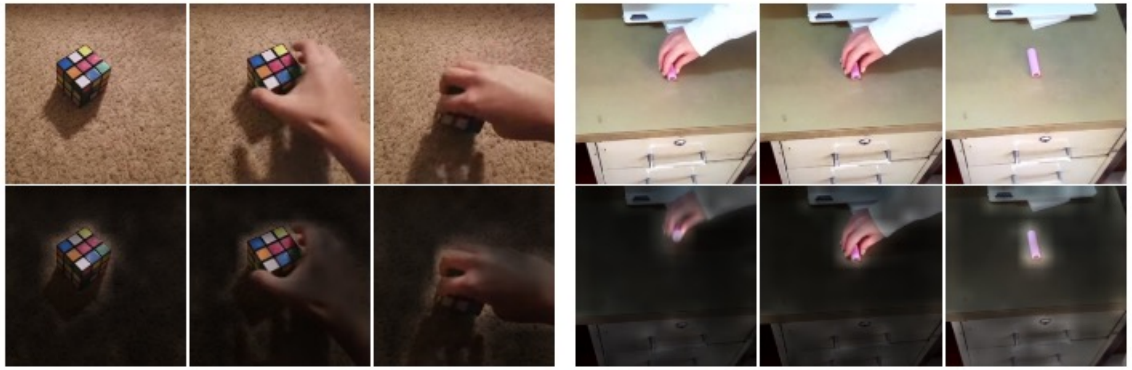
Video-level accuracy on Kinetics-400

Method	Top-1	Top-5	TFLOPs
R(2+1)D (Tran et al., 2018)	72.0	90.0	17.5
bLVNet (Fan et al., 2019)	73.5	91.2	0.84
TSM (Lin et al., 2019)	74.7	N/A	N/A
S3D-G (Xie et al., 2018)	74.7	93.4	N/A
Oct-I3D+NL (Chen et al., 2019)	75.7	N/A	0.84
D3D (Stroud et al., 2020)	75.9	N/A	N/A
I3D+NL (Wang et al., 2018b)	77.7	93.3	10.8
ip-CSN-152 (Tran et al., 2019)	77.8	92.8	3.2
CorrNet (Wang et al., 2020a)	79.2	N/A	6.7
LGD-3D-101 (Qiu et al., 2019)	79.4	94.4	N/A
SlowFast (Feichtenhofer et al., 2019b)	79.8	93.9	7.0
X3D-XXL (Feichtenhofer, 2020)	80.4	94.6	5.8
TimeSformer	78.0	93.7	**0.59**
TimeSformer-HR	79.7	94.4	5.11
TimeSformer-L	**80.7**	**94.7**	7.14

Table 5. Video-level accuracy on Kinetics-400.

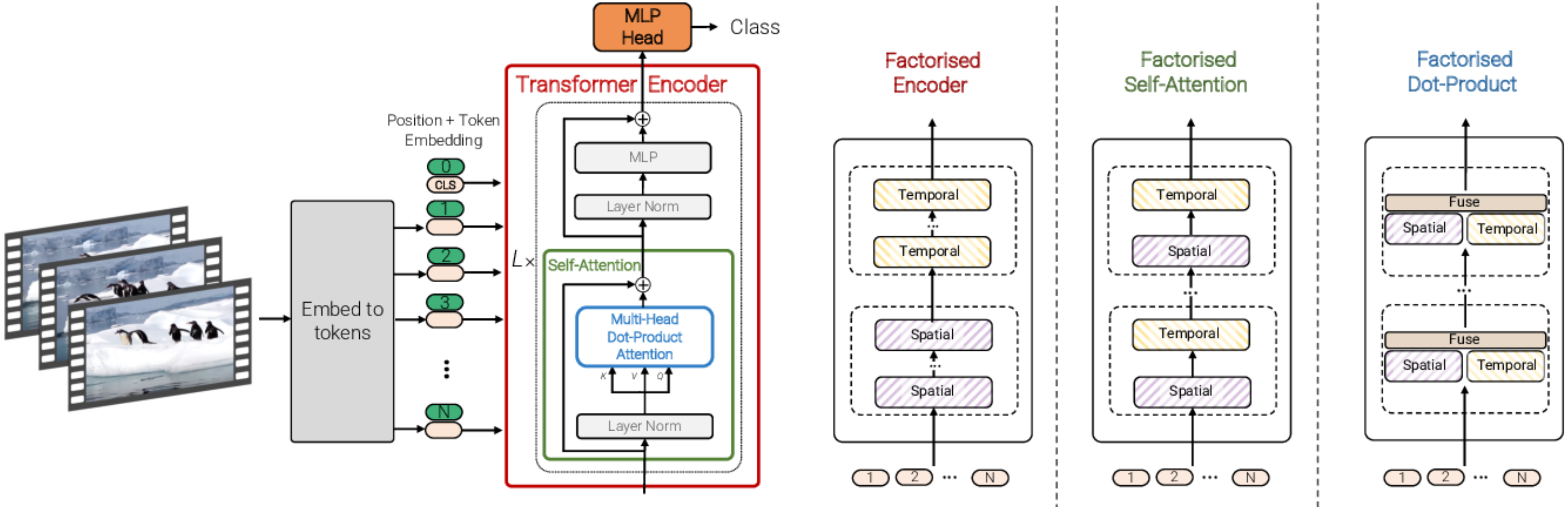
Method	Top-1	Top-5
I3D-R50+Cell (Wang et al., 2020c)	79.8	94.4
LGD-3D-101 (Qiu et al., 2019)	81.5	95.6
SlowFast (Feichtenhofer et al., 2019b)	81.8	95.1
X3D-XL (Feichtenhofer, 2020)	81.9	95.5
TimeSformer	79.1	94.4
TimeSformer-HR	81.8	**95.8**
TimeSformer-L	**82.2**	95.6

Visualization of space-time attention from the output taken to the input space.



Source: Bertasius et al.. “Is Space-Time Attention All You Need for Video Understanding?”, ICML 2021.

ViViT: A Video Vision Transformer



Source: Arnab et al.. "ViViT: A Video Vision Transformer", ICCV 2021.

Implementation

[1]. Implementing U-Net from scratch in PyTorch

https://nn.labml.ai/unet/index.html

https://towardsdatascience.com/cook-your-first-u-net-in-pytorch-b3297a844cf3

[2]. Semantic segmentation using U-Net in PyTorch

https://wandb.ai/ishandutta/semantic_segmentation_unet/reports/Semantic-Segmentation-with-UNets-in-PyTorch--VmlldzoyMzA3MTk1

https://github.com/PacktPublishing/Modern-Computer-Vision-with-PyTorch/blob/master/Chapter09/Semantic_Segmentation_with_U_Net.ipynb

[3]. U-Net model in PyTorch

https://pytorch.org/hub/mateuszbuda_brain-segmentation-pytorch_unet/

[4]. Northern Pike segmentation using U-Net

https://www.datainwater.com/post/pike_segmentation/

[5]. PyImageSearch's tutorial on U-Net implementation on TGS Salt Segmentation Challenge

https://pyimagesearch.com/2021/11/08/u-net-training-image-segmentation-models-in-pytorch/

Implementation

[6]. Semantic segmentation using TensorFlow Model Garden

https://www.tensorflow.org/tfmodels/vision/semantic_segmentation

[7]. Mask R-CNN in PyTorch

https://pytorch.org/vision/main/models/mask_rcnn.html

[8]. Mask R-CNN for pedestrian instance segmentation

https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html

[9]. Instance segmentation using Mask R-CNN

https://haochen23.github.io/2020/05/instance-segmentation-mask-rcnn.html

[10]. Fine-tuning Mask R-CNN on custom data using Detectron2

https://geekyrakshit.dev/geekyrakshit-blog/computervision/deeplearning/segmentation/objectdetction/neuralnetwork/instancesegmentation/convolution/detectron/maskrcnn/python/pytorch/2020/04/13/detectron-mask-rcnn.html

Further reading on discussed topics

- Chapter 7 of Deep Learning Book by Ian Goodfellow, Yoshua Bengio and Aaron Courville. https://www.deeplearningbook.org/
- Chapter 4: Object Detection and Image Segmentation from Practical Machine Learning for Computer Vision by Valliappa Lakshmanan, Martin Gorner, Ryan Gillard. https://www.oreilly.com/library/view/practical-machine-learning/9781098102357/ch04.html
- Chapter 7 of Deep Learning for Vision Systems by Mohamed Elgendy.

Acknowledgements

- Some material drawn from referenced and associated online sources
- Image sources credited where possible
- Some slides adapted from cs231n Lecture 9 “Object Detection and Image Segmentation”

References

[1]. Farabet et al. “Learning Hierarchical Features for Scene Labeling”, TPAMI 2013

https://ieeexplore.ieee.org/document/6338939

[2]. Long et al. “Fully Convolutional Networks for Semantic Segmentation”, CVPR 2015.

https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Long_Fully_Convolutional_Networks_2015_CVPR_paper.pdf

[3]. Ronneberger et al. (2015). U-net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015.

https://link.springer.com/chapter/10.1007/978-3-319-24574-4_28

[4]. Oktay et al., (2018). “Attention U-Net: Learning where to look for the Pancreas”, MIDL 2018.

https://openreview.net/forum?id=Skft7cijM

[5]. Diakogiannis et al., (2019). “ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data”, ISPRS Journal of Photogrammetery and Remote Sensing.

https://www.sciencedirect.com/science/article/abs/pii/S0924271620300149

[6]. Chen et al., (2021). “TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation”, ArXiv.

https://arxiv.org/abs/2102.04306

[7]. He et al., “Mask R-CNN”. ICCV 2017.

https://openaccess.thecvf.com/content_ICCV_2017/papers/He_Mask_R-CNN_ICCV_2017_paper.pdf

References

[8]. Vu et al. “3D CNN for feature extraction and classification of fMRI volumes”. PRNI 2018.

https://ieeexplore.ieee.org/document/8423964

[9]. Tran et al. “Learning Spatiotemporal Features with 3D Convolutional Networks”. ICCV 2015

https://www.cv-foundation.org/openaccess/content_iccv_2015/papers/Tran_Learning_Spatiotemporal_Features_ICCV_2015_paper.pdf

[10]. Simonyan and Zisserman. “Two-Stream Convolutional Networks for Action Recognition in Videos”, NeurIPS 2014.

https://papers.nips.cc/paper_files/paper/2014/hash/00ec53c4682d36f5c4359f4ae7bd7ba1-Abstract.html

[11]. Donahue et al.. “Long-term Recurrent Convolutional Networks for Visual Recognition and Description”, CVPR 2015.

https://openaccess.thecvf.com/content_cvpr_2015/papers/Donahue_Long-Term_Recurrent_Convolutional_2015_CVPR_paper.pdf

[12]. Bertasius et al.. “Is Space-Time Attention All You Need for Video Understanding?”, ICML 2021.

http://proceedings.mlr.press/v139/bertasius21a/bertasius21a.pdf

[13]. Arnab et al.. “ViViT: A Video Vision Transformer”, ICCV 2021.

https://openaccess.thecvf.com/content/ICCV2021/papers/Arnab_ViViT_A_Video_Vision_Transformer_ICCV_2021_paper.pdf

Example exam question

What kind of neural network is most suited for image segmentation tasks?

A. Multilayer perceptron (MLP)

B. Fully convolutional network (FCN)

C. Region proposal network (RPN)

D. Recurrent neural network (RNN)