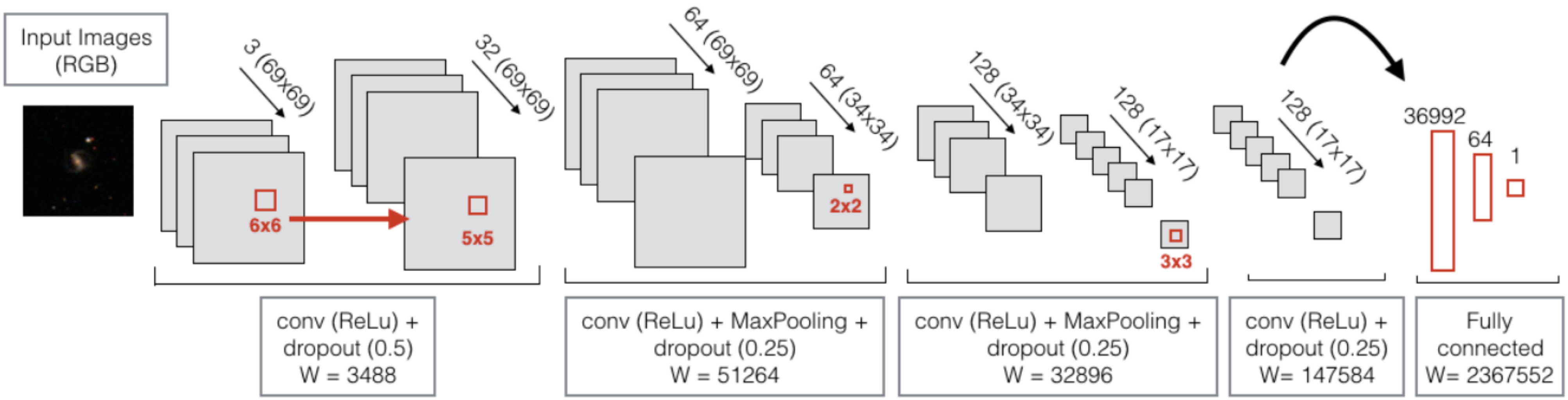
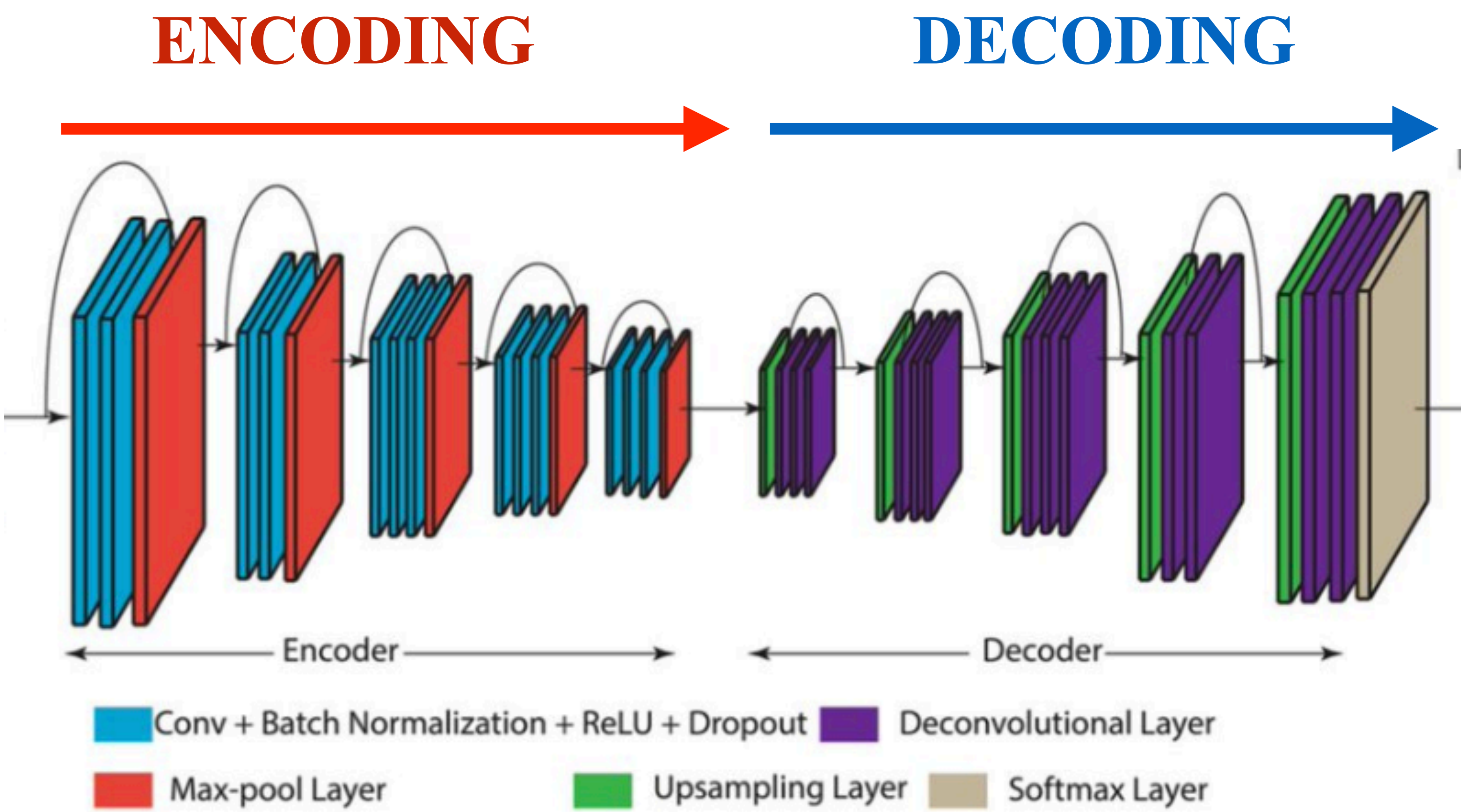
IMAGE2IMAGE NETWORKS

UP TO NOW CNNs MAP IMAGES (SIGNALS) INTO FLOATS

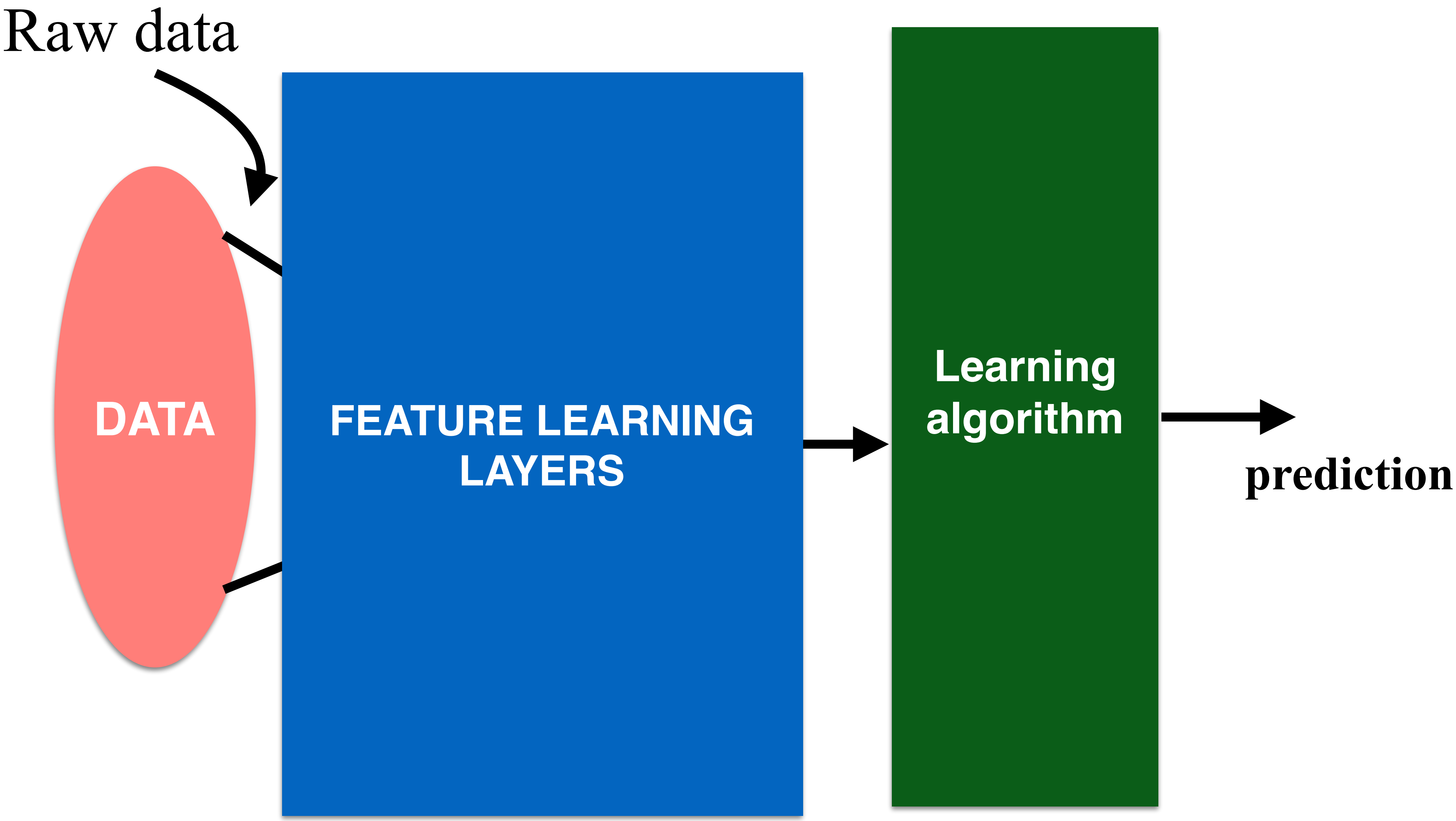


Dominguez-Sanchez+18

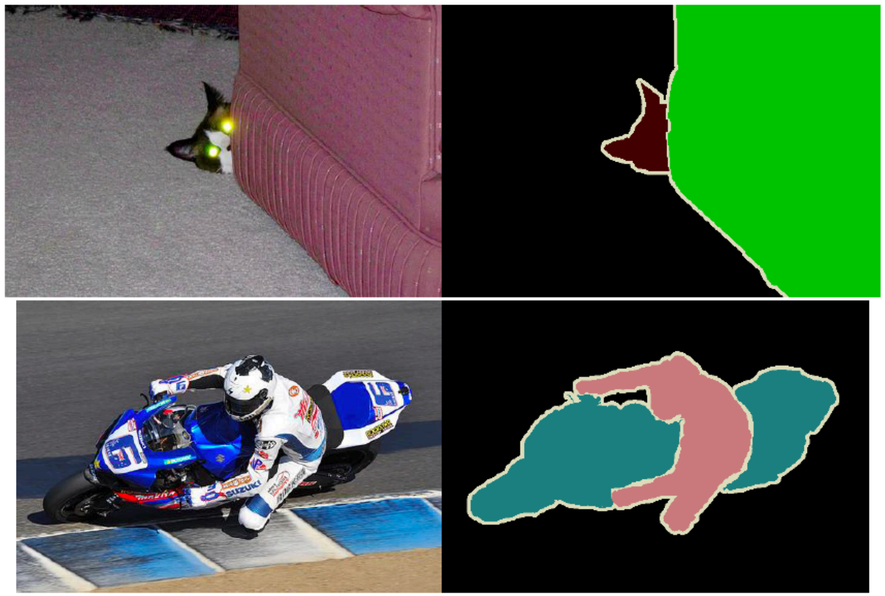
ENCODER-DECODERS GO FROM IMAGE 2 IMAGE



WE CALL THIS FULLY CONVOLUTIONAL NEURAL NETWORKS

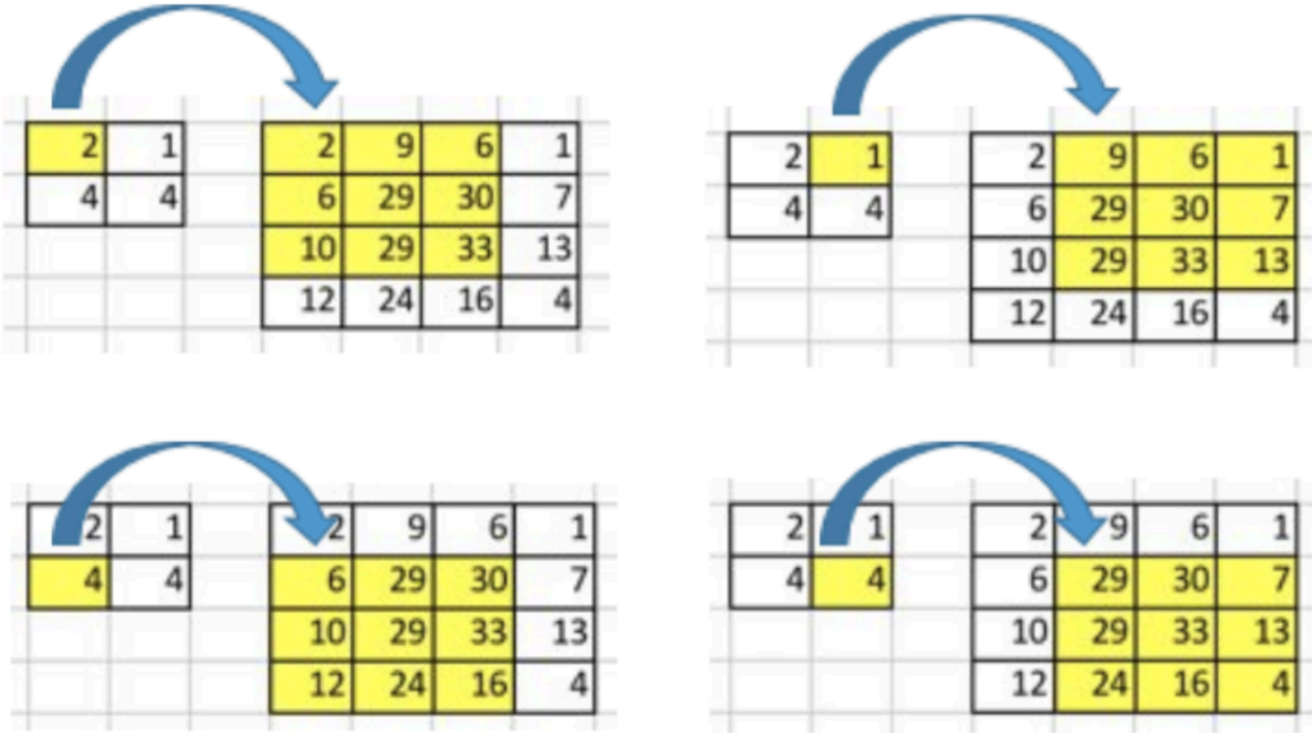

Raw data
DATA
FEATURE LEARNING LAYERS
Learning algorithm
prediction

FIRST: IMAGE SEGMENTATION WITH ENCODERS-DECODERS



TRANSPOSED CONVOLUTION

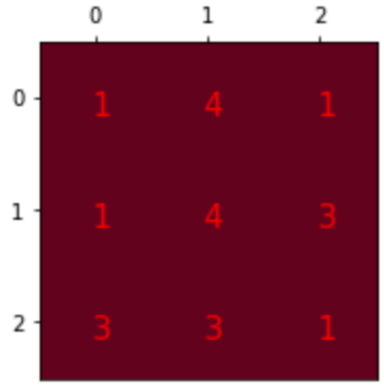
ALLOWS TO INCREASE THE SIZE



Going Backward of Convolution

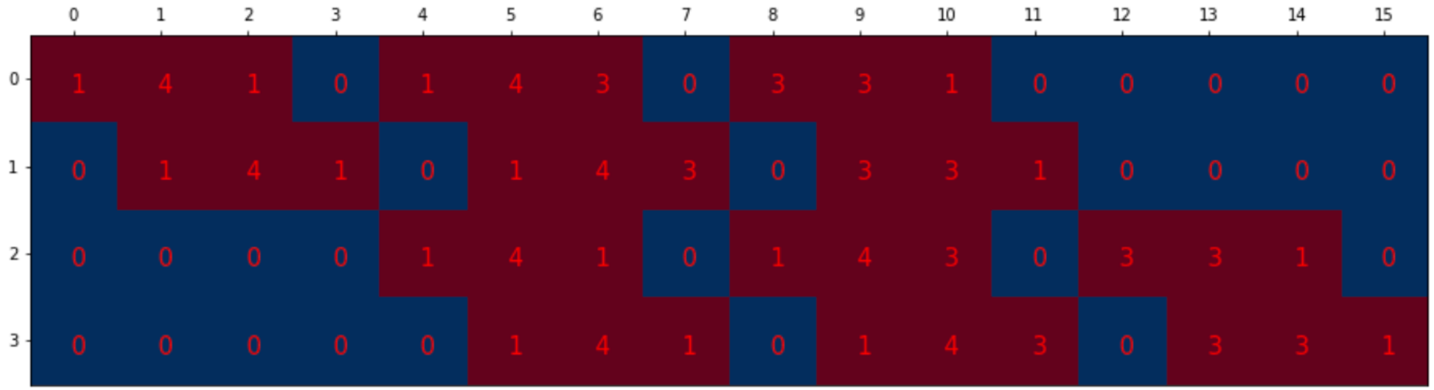
EXAMPLE TAKEN FROM HERE

CONVOLUTION MATRIX



Kernel (3, 3)

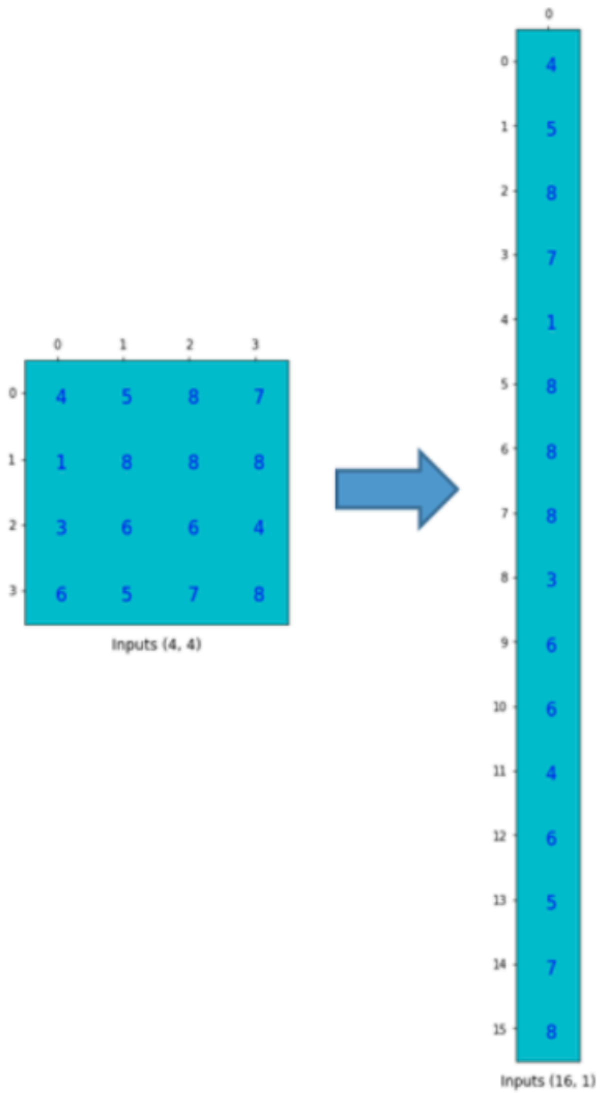
THE KERNEL CAN BE ARRANGED IN FORM OF A MATRIX:



Convolution Matrix (4, 16)

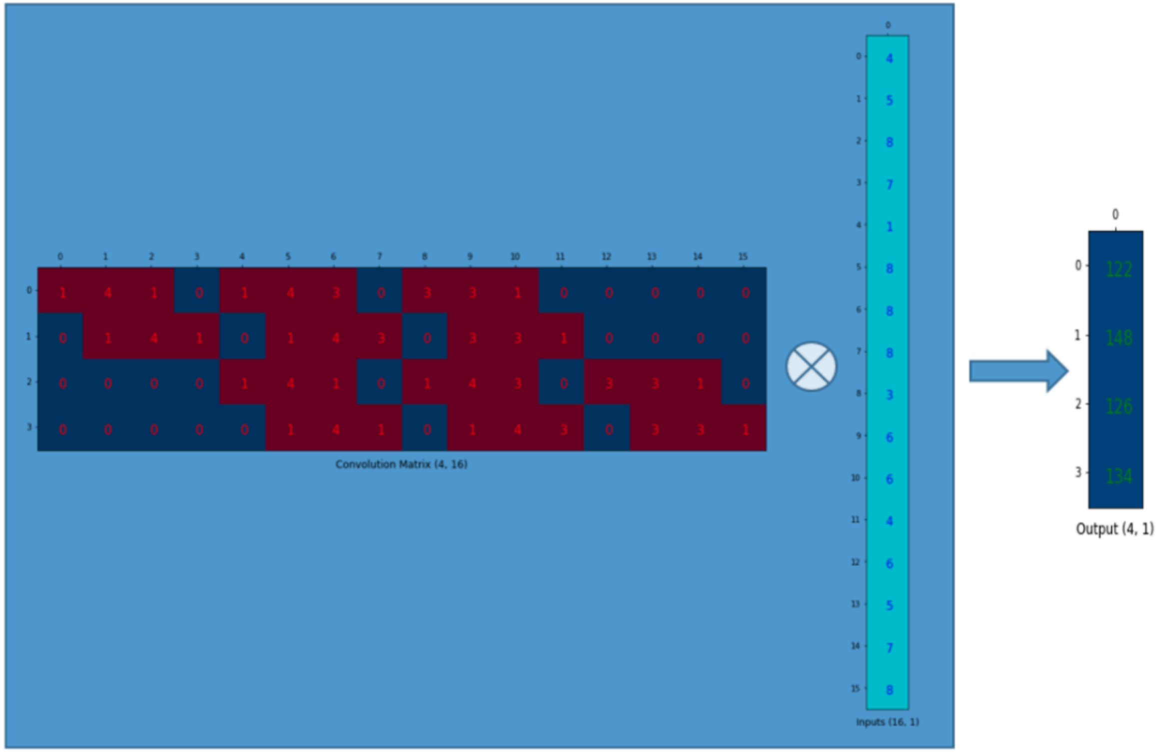
<u>**EXAMPLE TAKEN FROM HERE**</u>

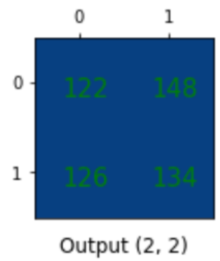
THE INPUT IS FLATTENED INTO A COLUMN VECTOR



EXAMPLE TAKEN FROM HERE

THE CONVOLUTION IS TRANSFORMED INTO A PRODUCT OF MATRICES

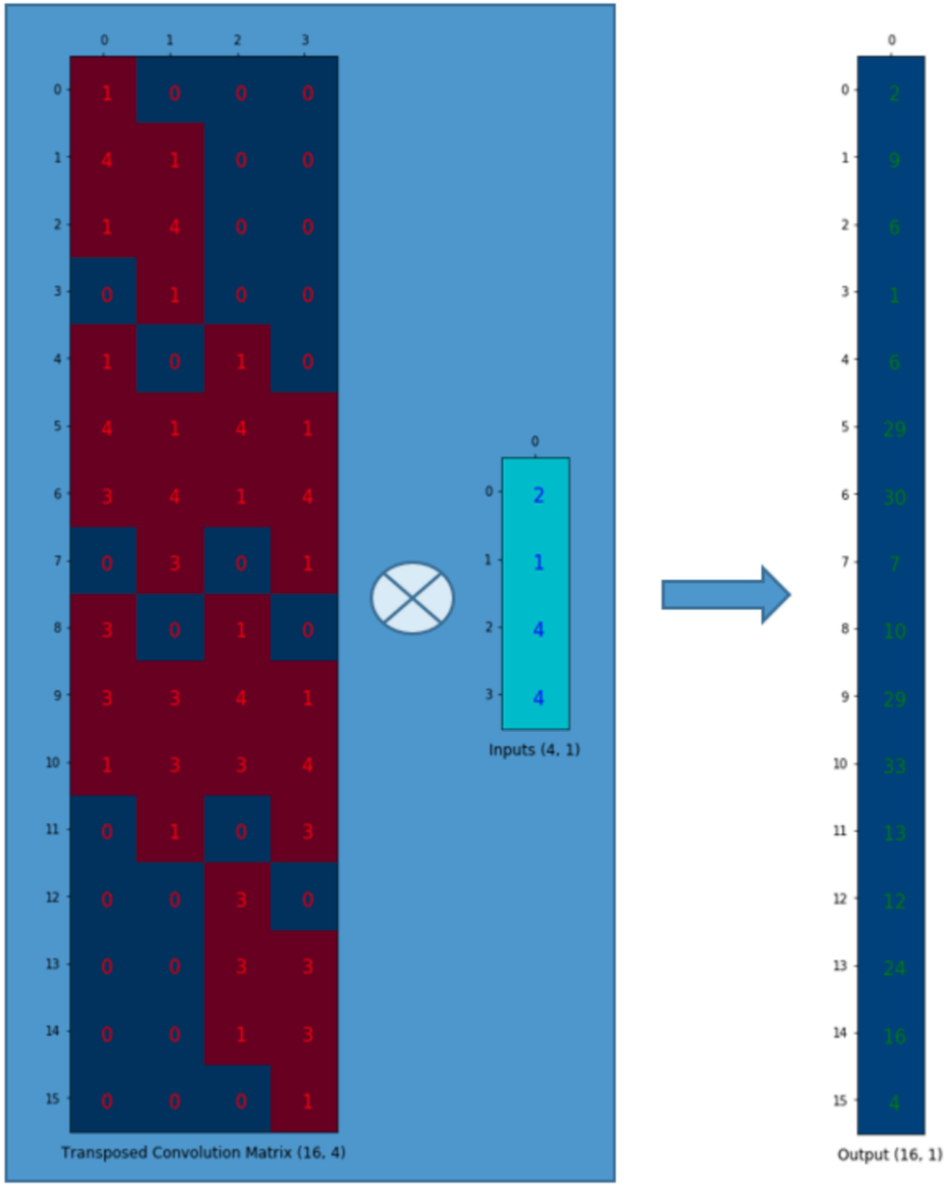


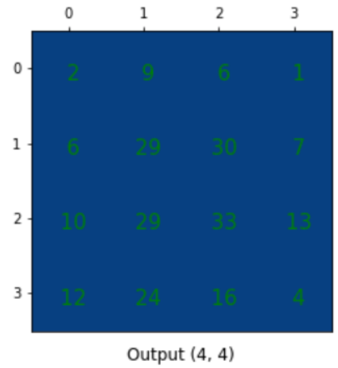


RESHAPED OUTPUT

EXAMPLE TAKEN FROM HERE

THE TRANSPOSED CONVOLUTION IS THE INVERSE OPERATION





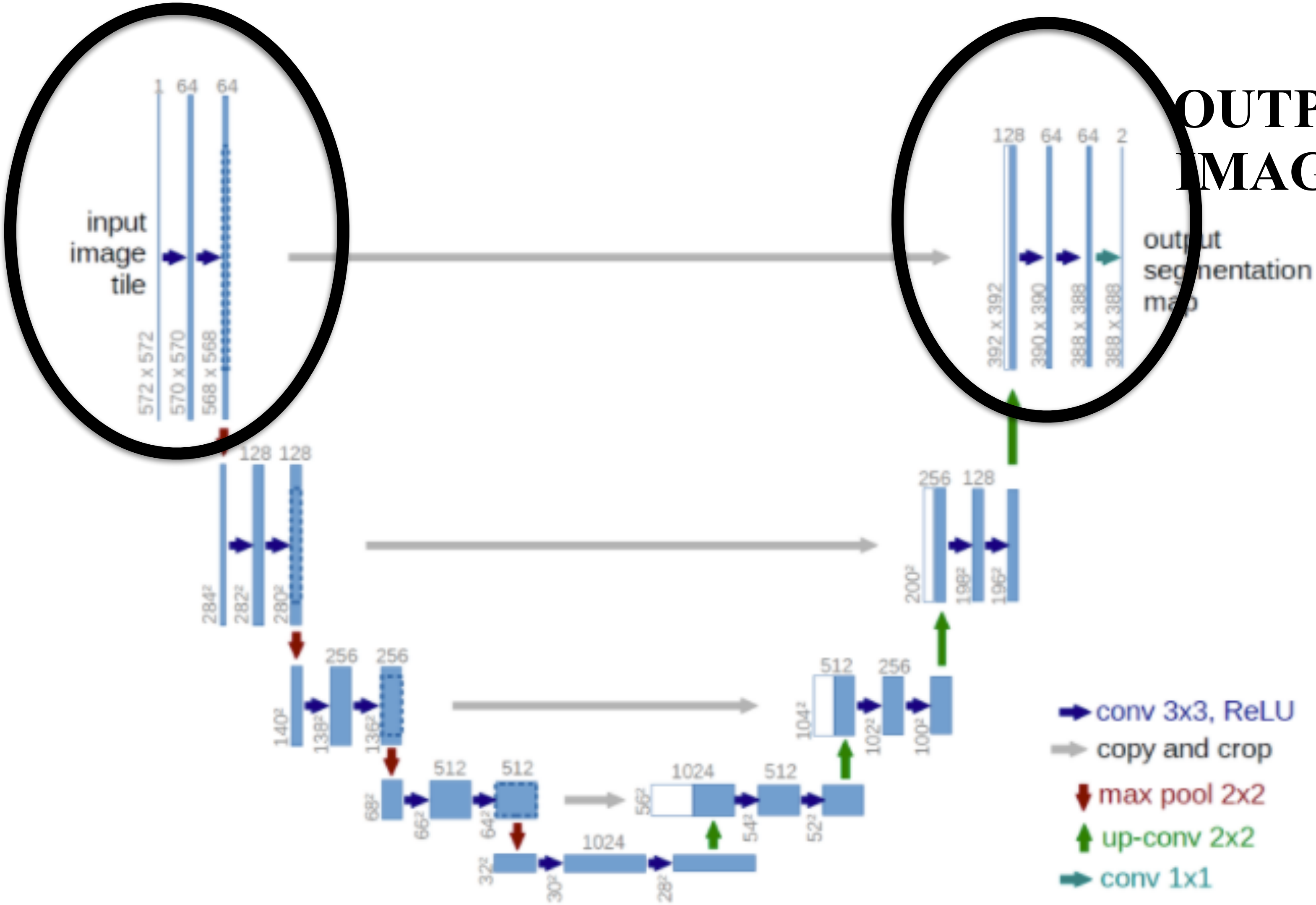
RESHAPED OUTPUT

EXAMPLE TAKEN FROM HERE

ENCODING-DECODING TO EXTRACT IMAGE FEATURES: U-NET

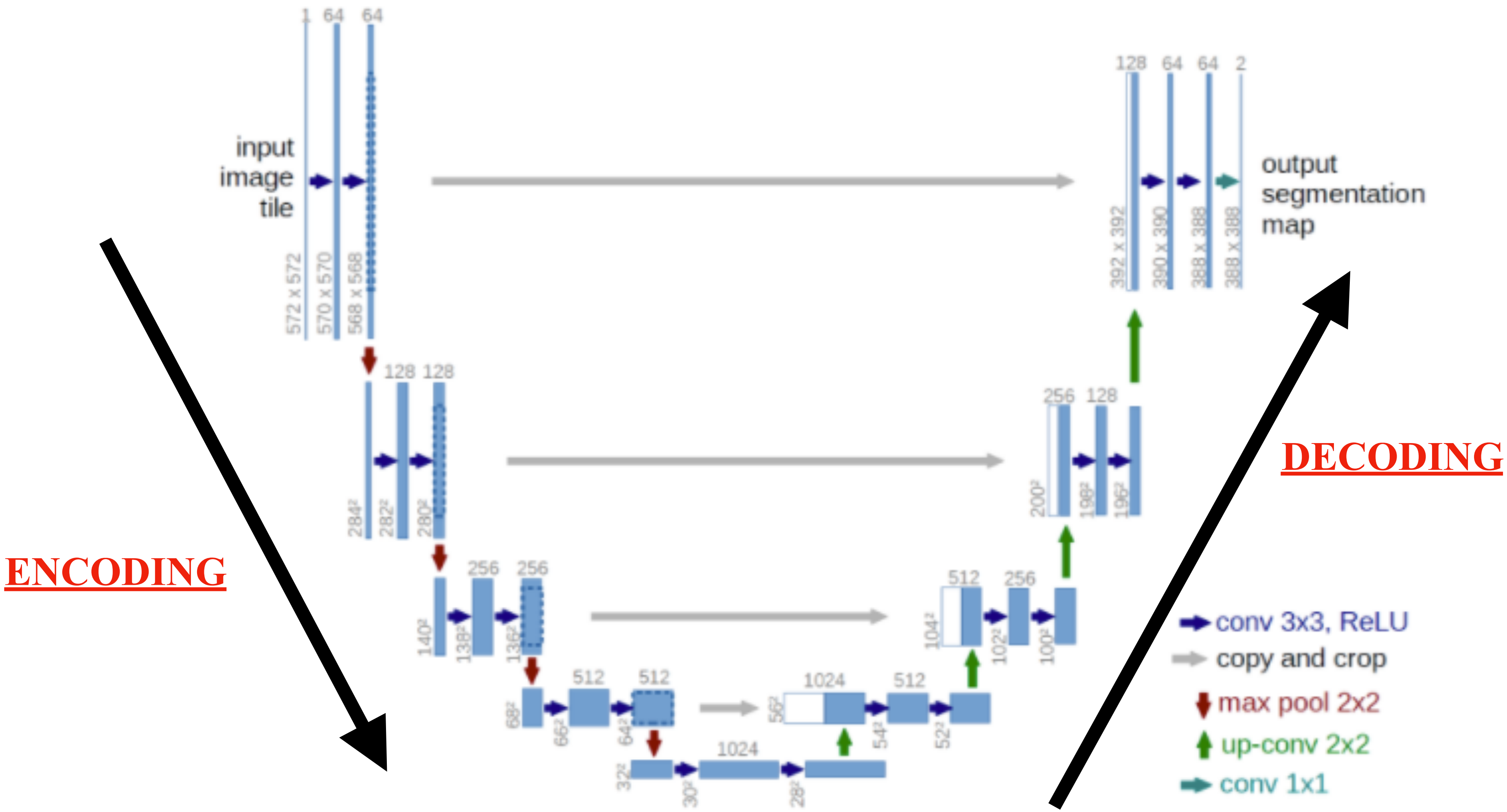
INPUT IMAGES

OUTPUT IMAGES



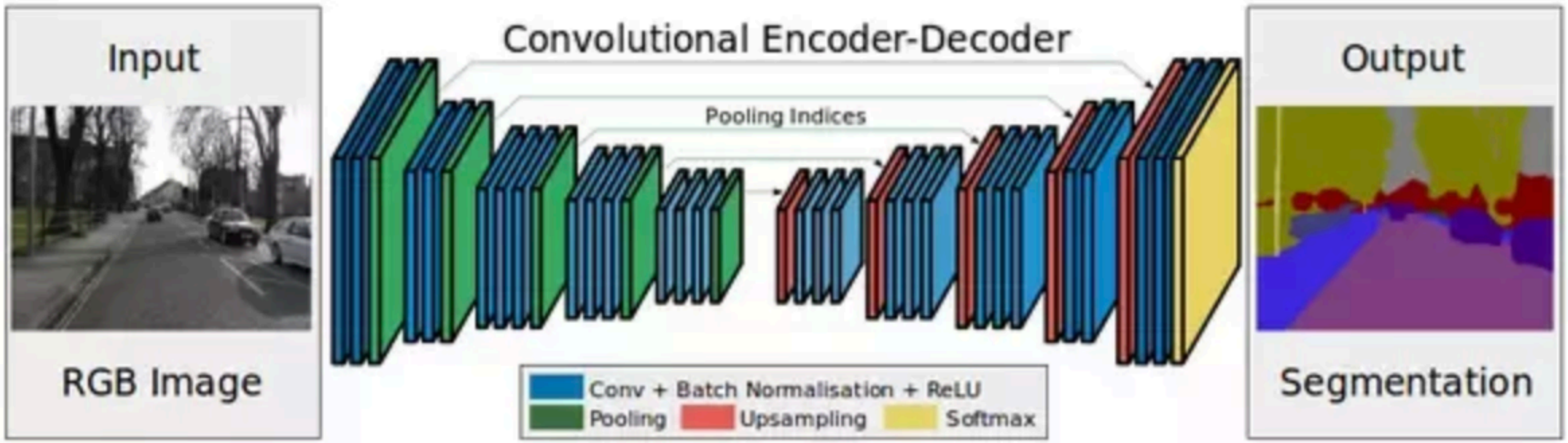
Ronnerberger+15

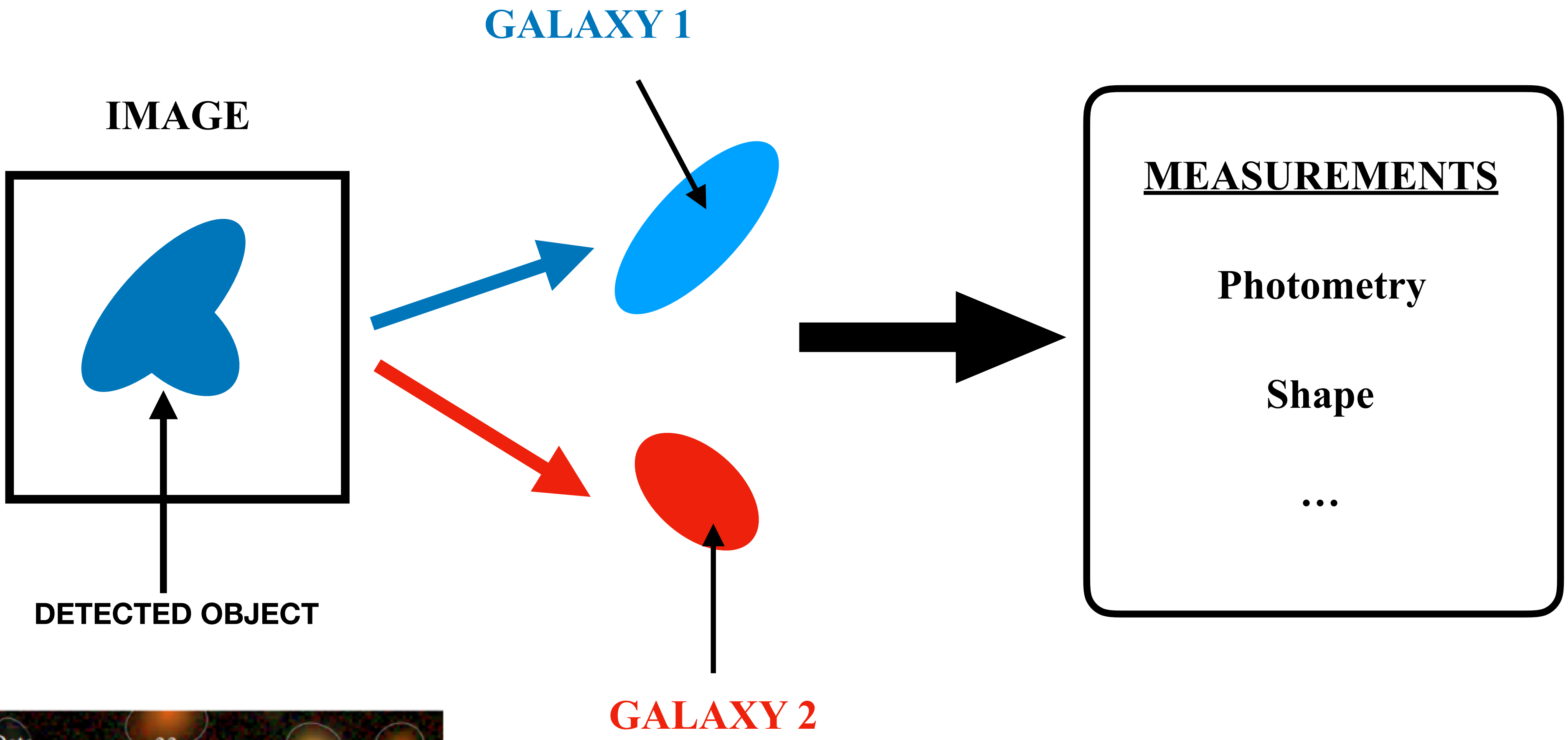
ENCODING-DECODING TO EXTRACT IMAGE FEATURES: U-NET

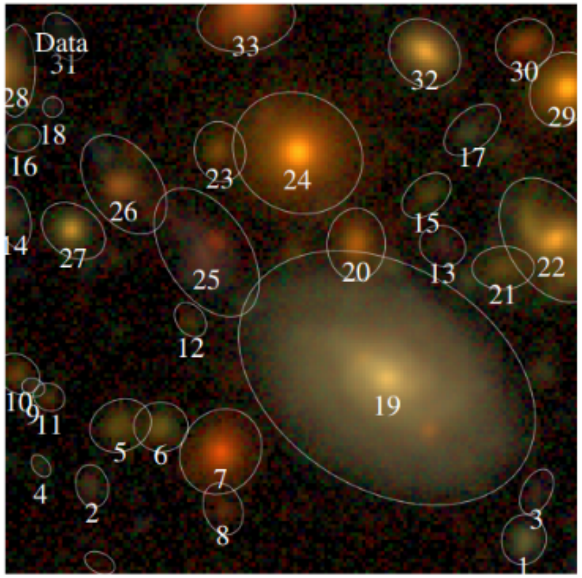


Ronnerberger+15

MAIN APPLICATION IS IMAGE SEGMENTATION







Bosch+17

>50% of objects will be affected by blending in future deep surveys such as LSST

Boucaud+19

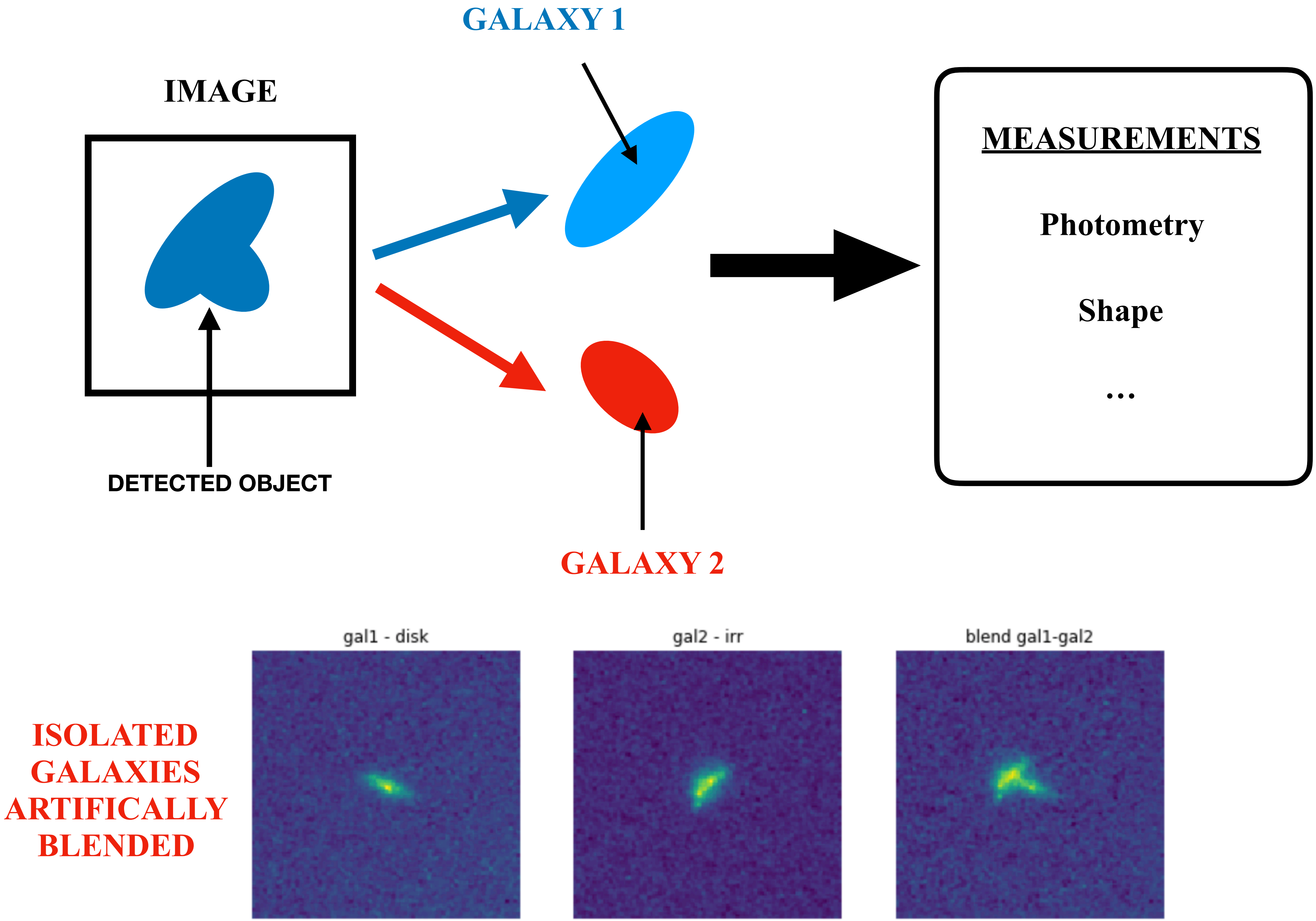


IMAGE
DETECTED OBJECT
GALAXY 1
GALAXY 2
MEASUREMENTS
Photometry
Shape
...
ISOLATED GALAXIES ARTIFICALLY BLENDED
gal1 - disk
gal2 - irr
blend gal1-gal2

U-NET (ENCODER DECODER)

VANILLA CNN

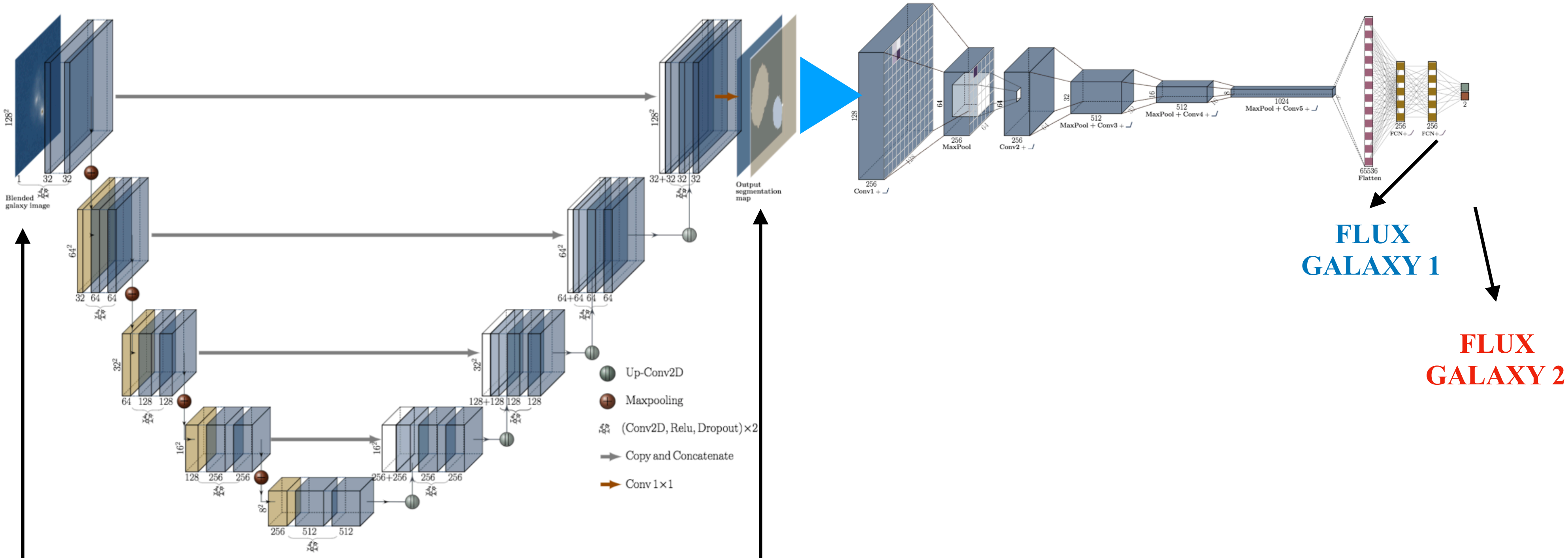
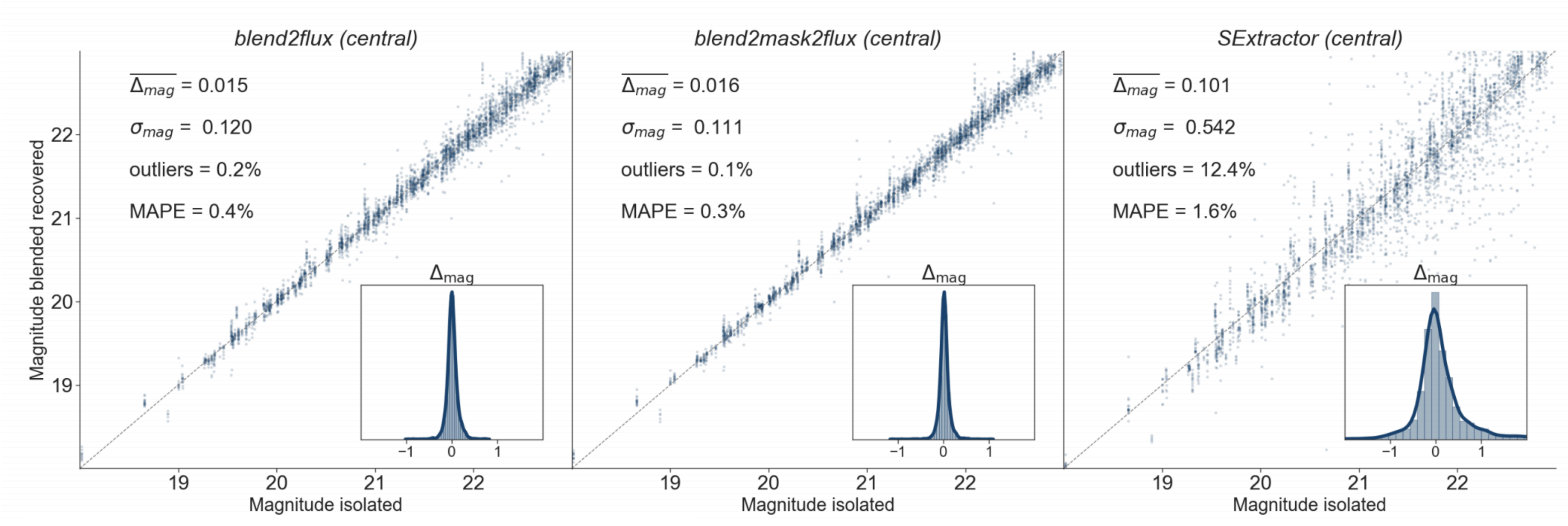
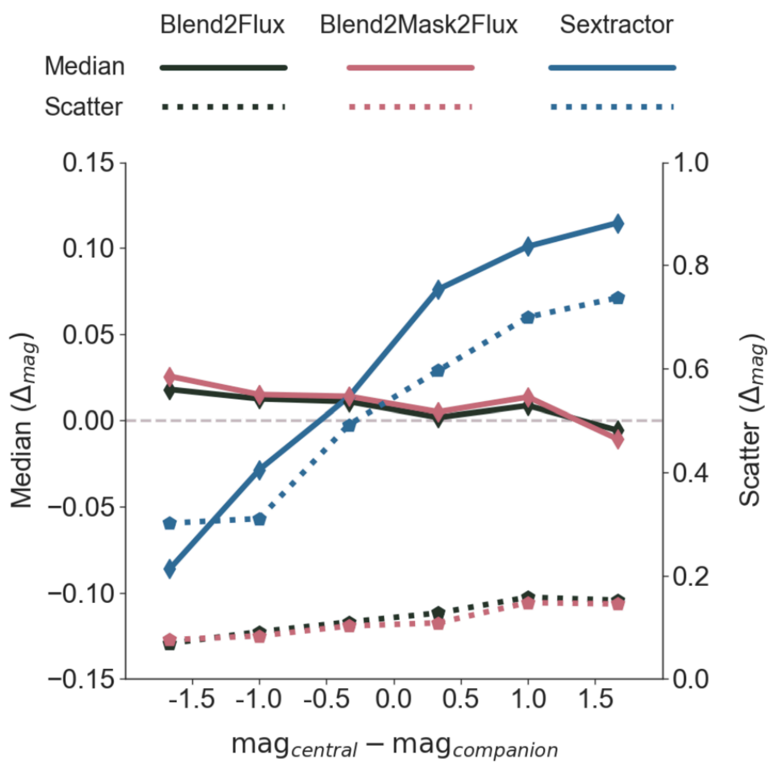


IMAGE OF BLENDED OBJECTS

OUTPUT SEGMENTATIO N MAP (BINARY 2 CHANNELS)

Boucaud, MHC+19

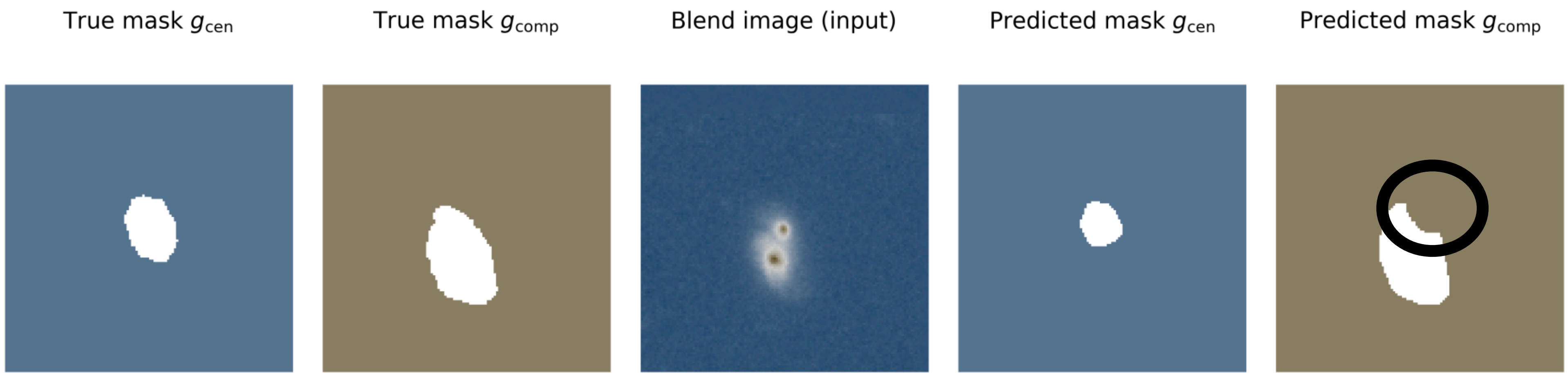




LESSONS LEARNED:

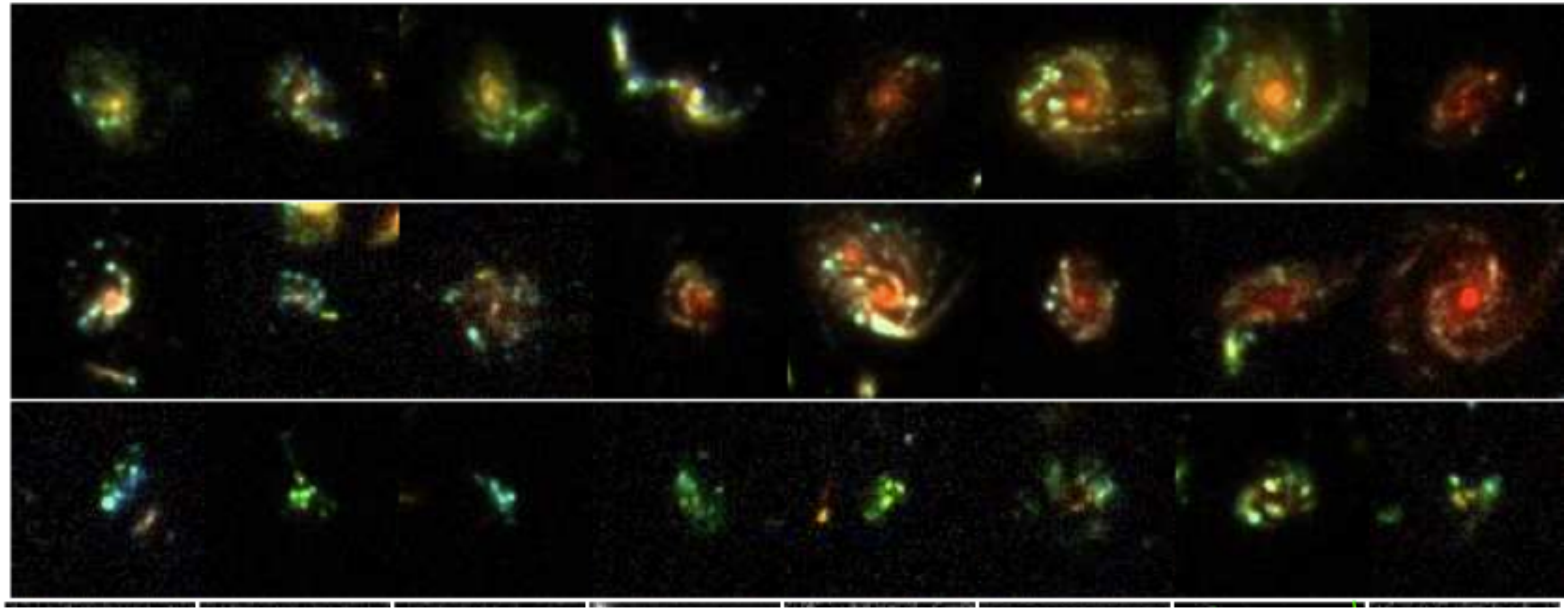
- It is easy to do better than Sextractor

- Masks do not provide additional information



THE U-NET LEARNS TO IGNORE THE BLENDED REGION…

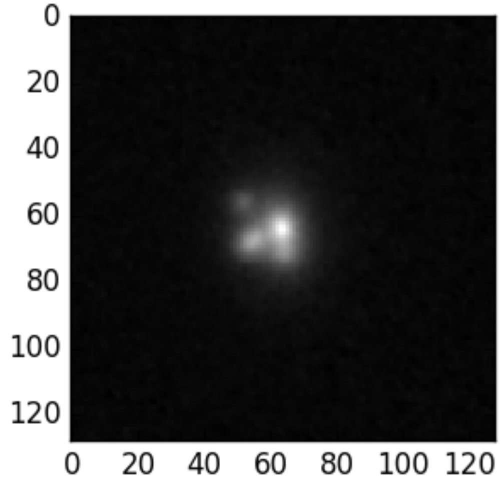
CLUMP DETECTION



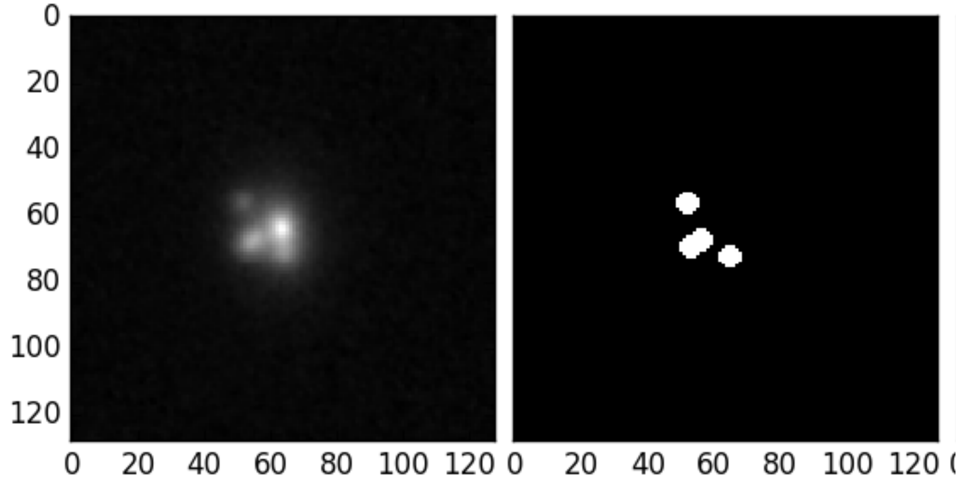
Guo+15,18

HIGH REDSHIFT GALAXIES PRESENT CLUMPS - THEIR ROLE IN BULGE FORMATION IS DEBATED

VERY SIMPLE SERSIC ANALYTIC SIMULATIONS + UNRESOLVED CLUMPS

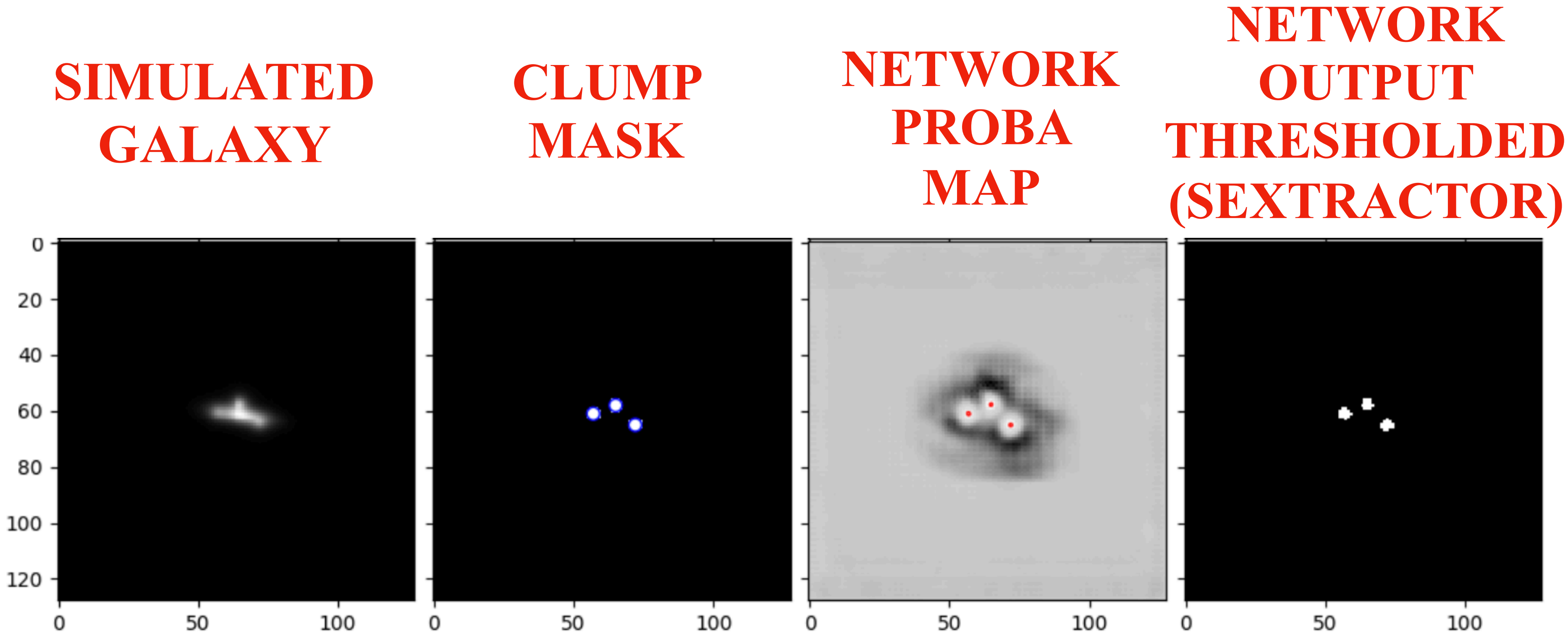


VERY SIMPLE SERSIC ANALYTIC SIMULATIONS + UNRESOLVED CLUMPS



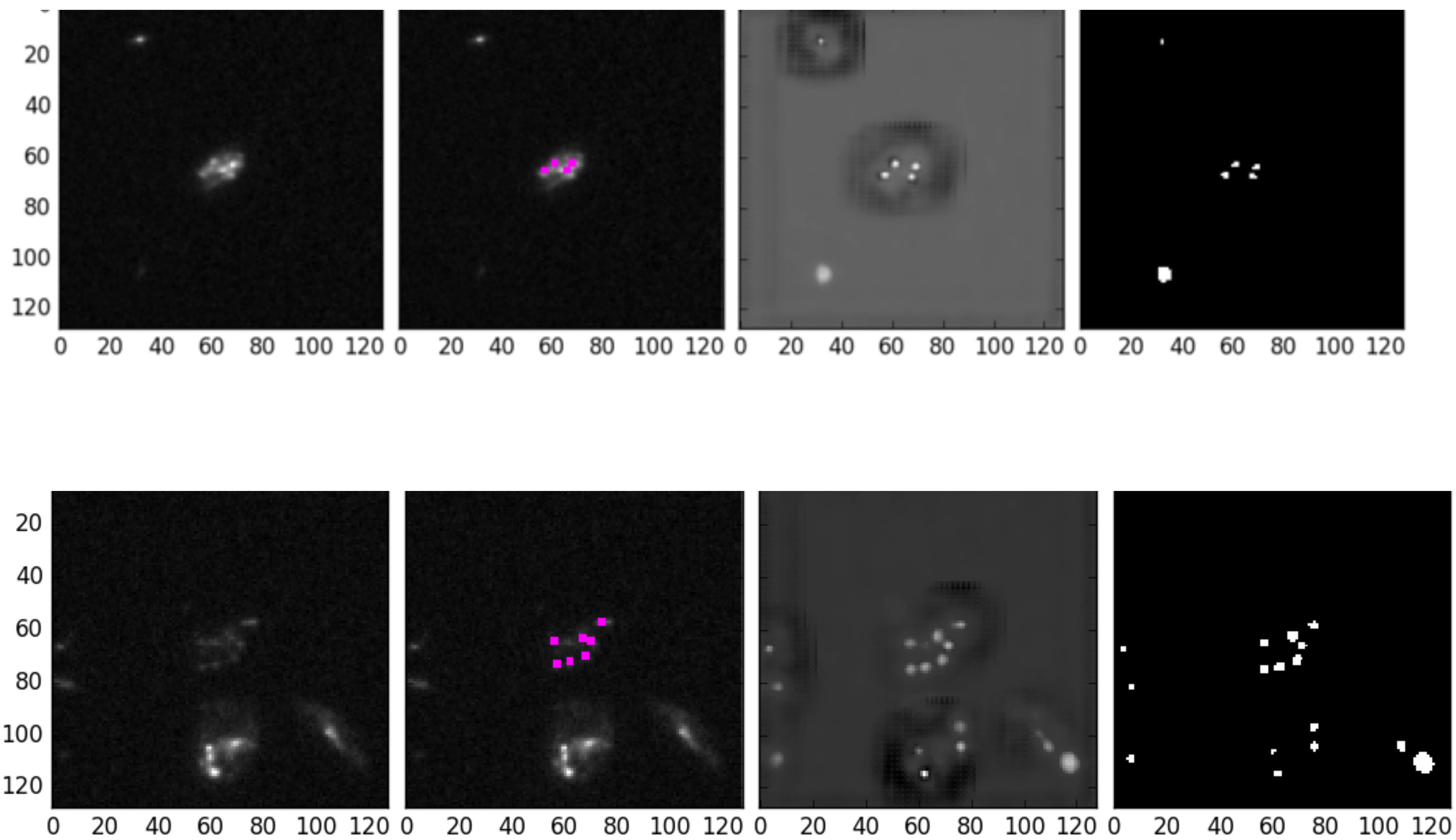
CLUMP POSITION IS KNOWN

LEE, MHC, PRIMACK, GUO+



LEE, MHC, PRIMACK+ IN PREP

SEEMS TO WORK REASONABLY WELL ON REAL OBSERVATIONS



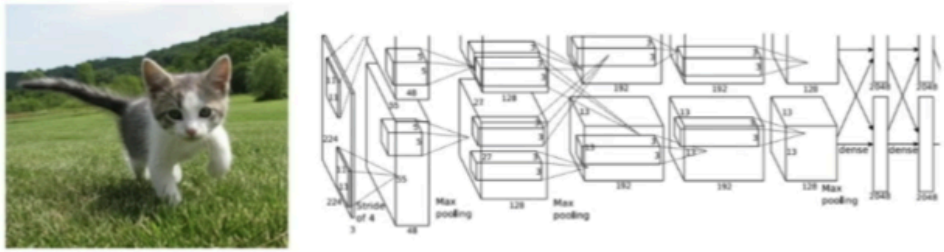
OBSERVED GALAXY

CLUMPS DETECTED BY GUO+

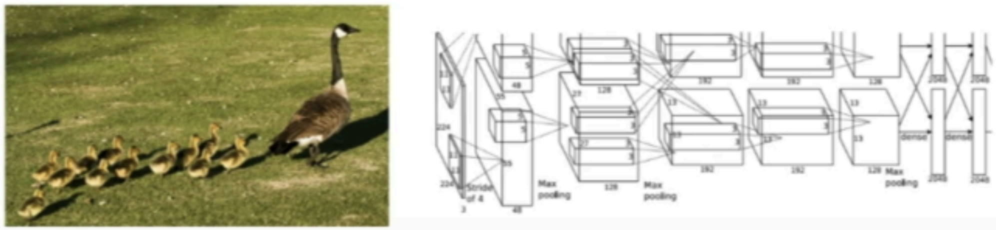
U-NET OUTPUTS

A WORD ON R-CNNs..

<u>JUST FOR YOUR RECORDS…</u>



CAT: (x, y, w, h)

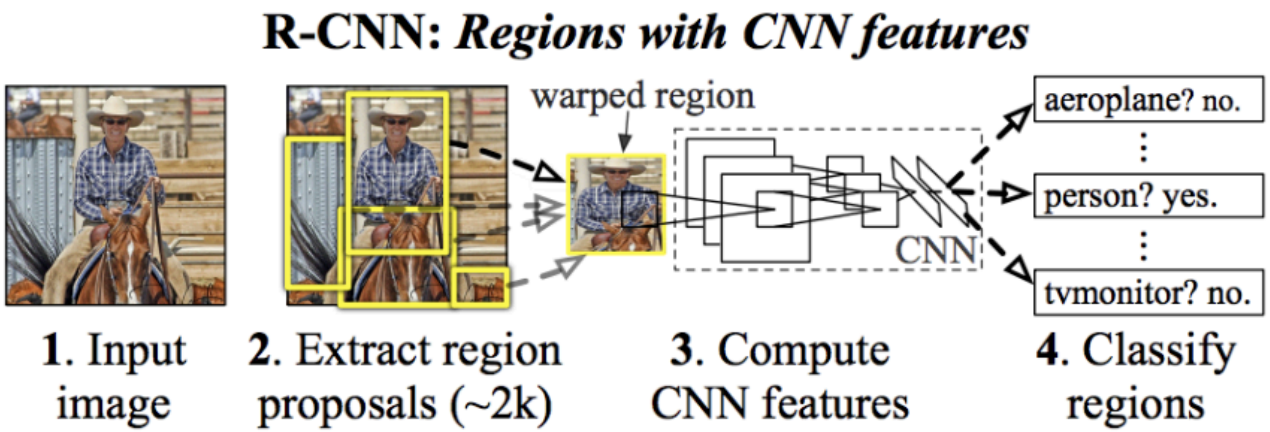


DUCK: (x, y, w, h)
DUCK: (x, y, w, h)
….

Girshick+<u>14</u>

A WORD ON R-CNNs..

JUST FOR YOUR RECORDS…

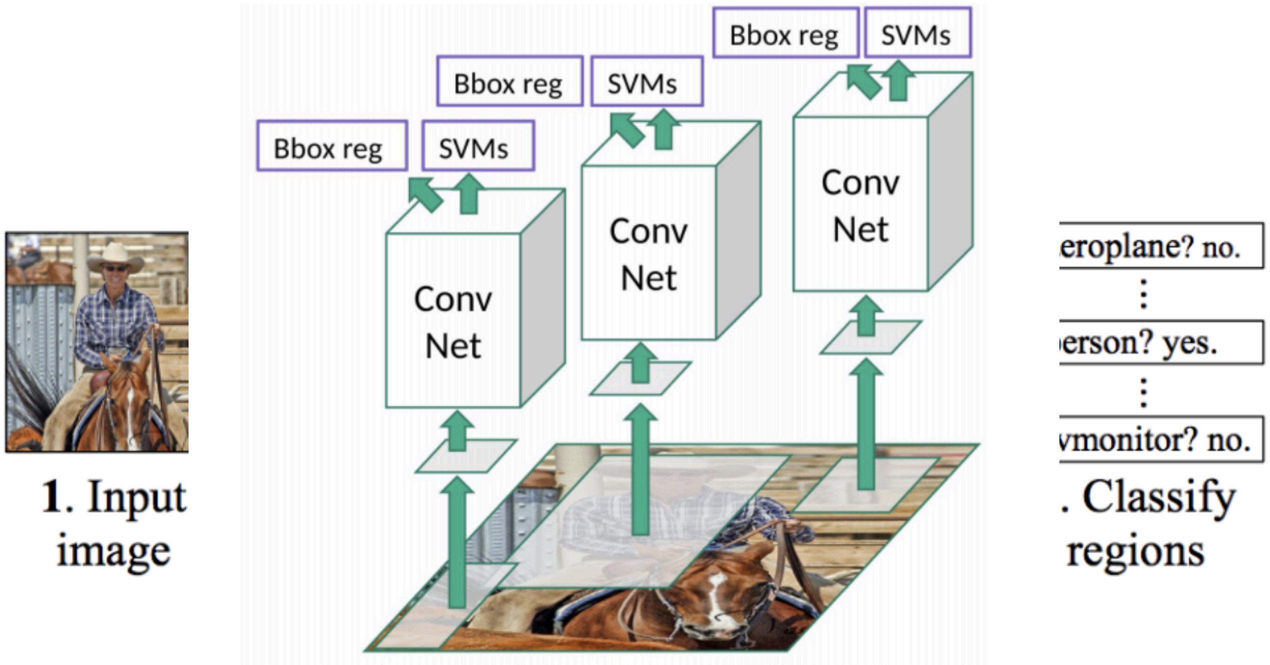


R-CNN

Girshick+14

A WORKD ON R-CNNs..

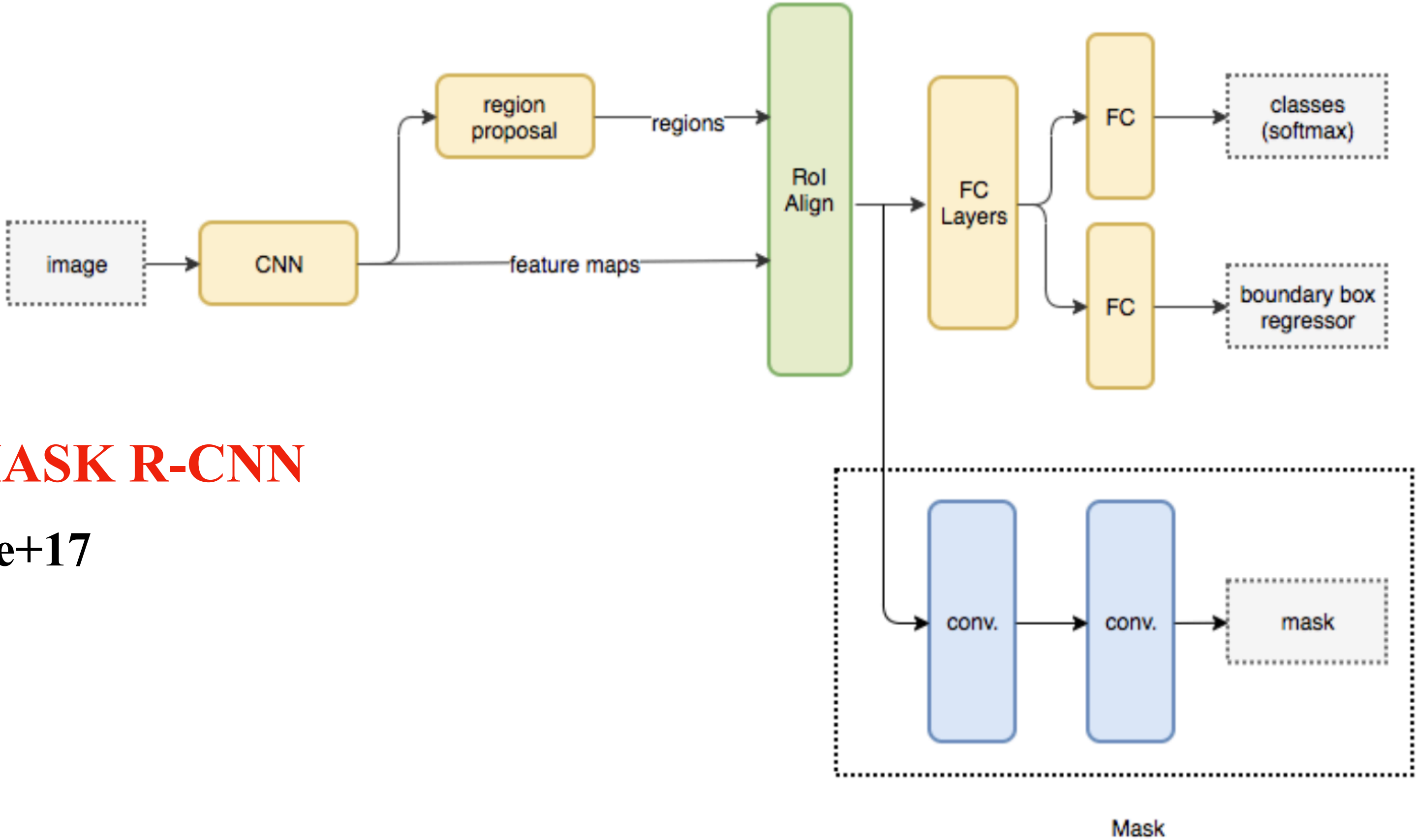
JUST FOR YOUR RECORDS…



R-CNN

Girshick+14

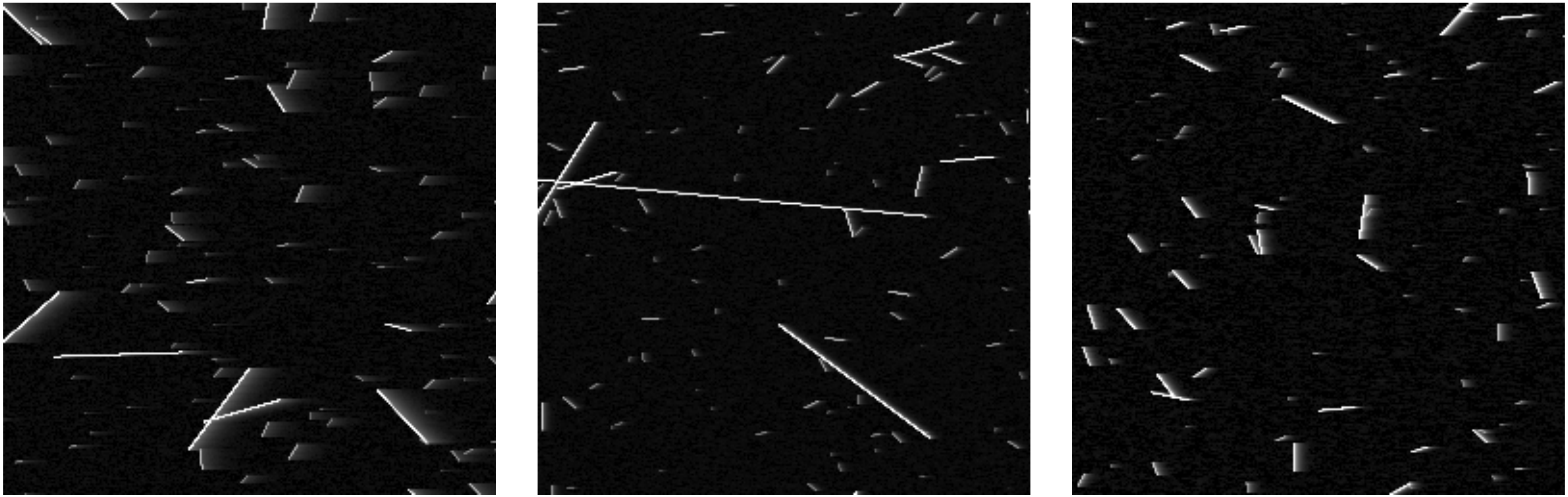
SIMULTANEOUS DETECTION + SEGMENTATION + CLASSIFICATION



MASK R-CNN

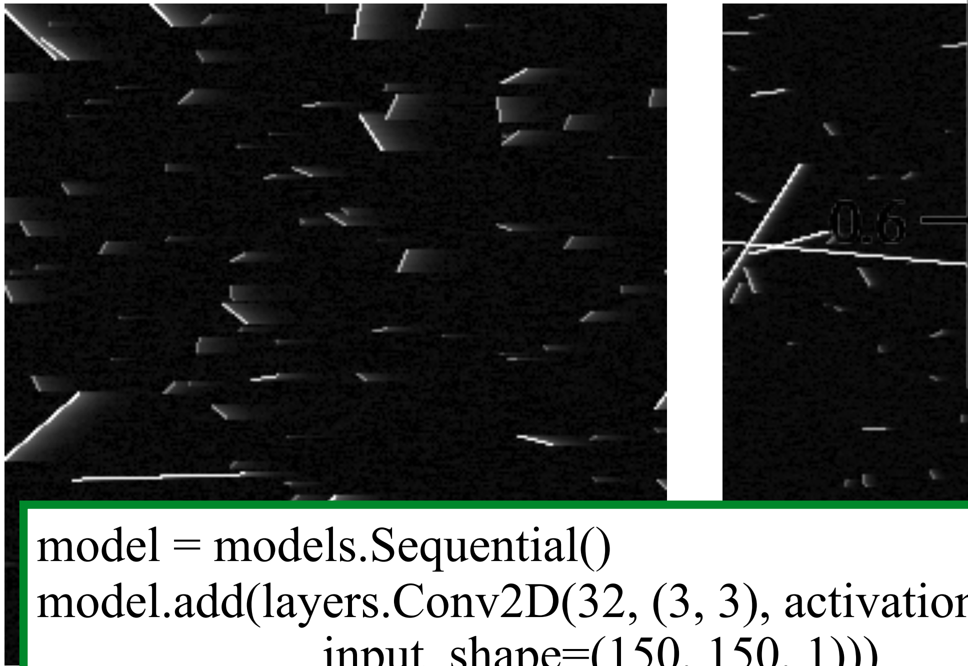
He+17

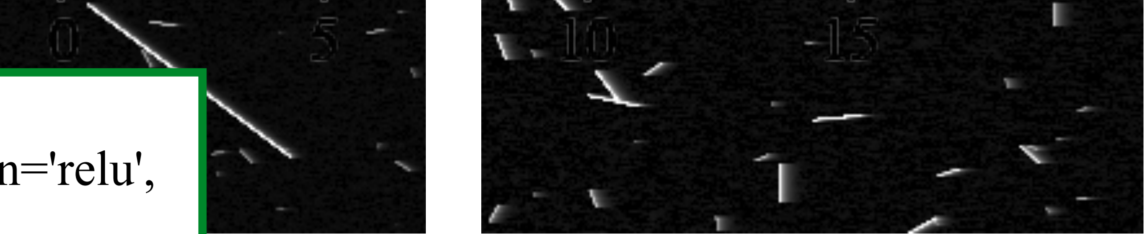
DATA QUALITY SELECTION FOR EUCLID

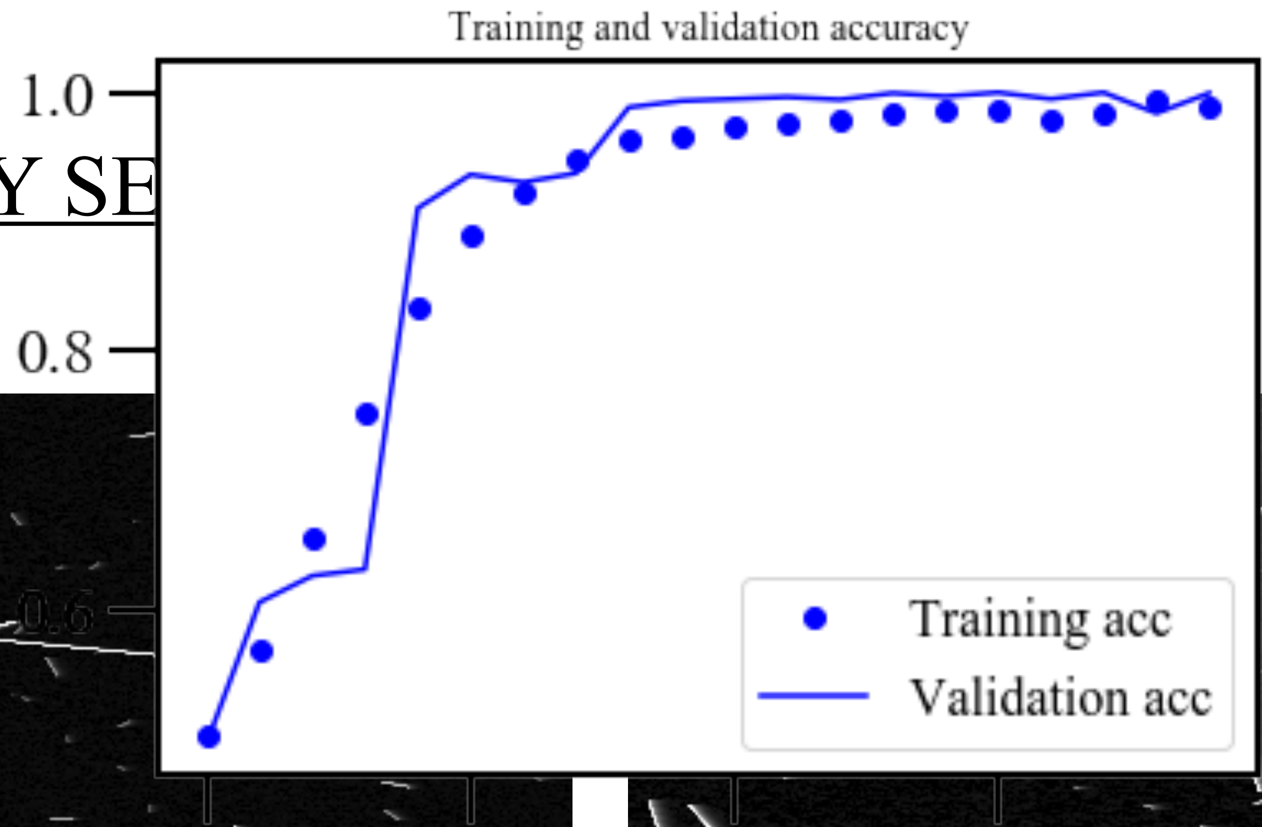


Thanks to H. McCracken

DATA QUALITY SE



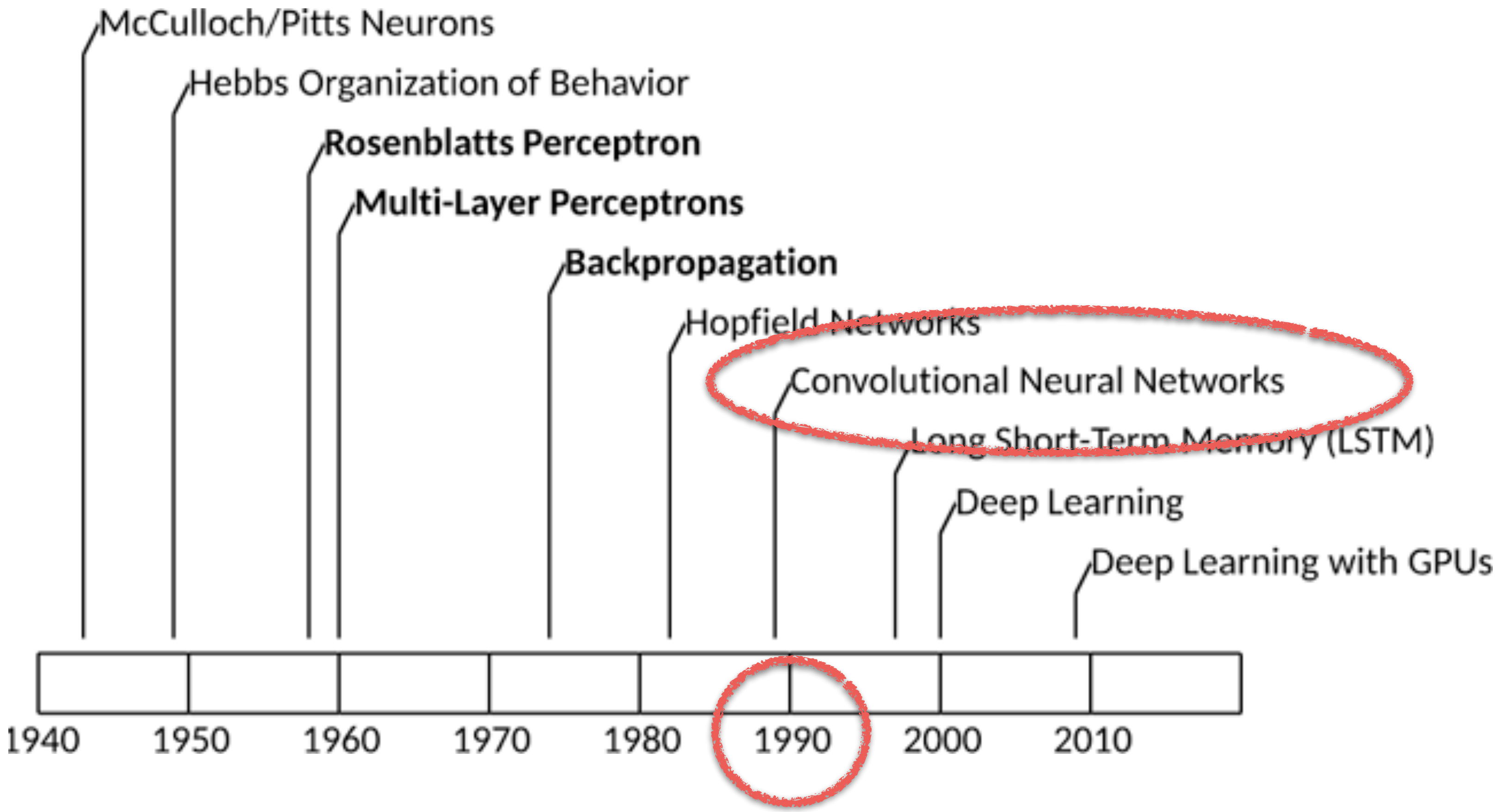





```
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3), activation='relu',
                        input_shape=(150, 150, 1)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (3, 3), activation='relu'))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(128, activation='relu'))
model.add(layers.Dense(1, activation='sigmoid'))
```

Thanks to H. McCracken

WELL, BUT THIS IS AN "OLD" IDEA - WHY NOW?



WELL, BUT THIS IS AN “OLD” IDEA - WHY NOW?

1 - MORE DATA TO TRAIN! DEEP NETWORKS HAVE A LARGE NUMBER OF PARAMETERS - THX TO SOCIAL MEDIA …

WELL, BUT THIS IS AN "OLD" IDEA - WHY NOW?

2 - GPUs - TRAINING OF THESE DEEP NETWORKS HAS REMAINED PROHIBITIVELY TIME CONSUMING WITH CPUs - THX TO VIDEO GAMES…

GPUs

NVIDA TITANX GPU

GPUs vs. CPUs

CPUs

FEWER CORES (~10x)

EACH CORE IS FASTER

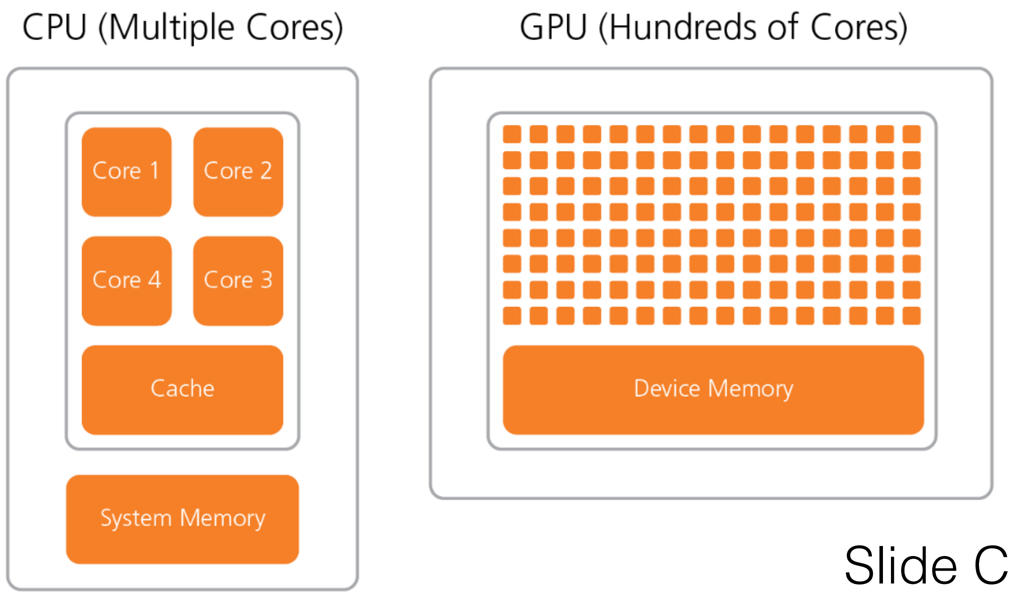
USEFUL FOR SEQUENTIAL TASKS

GPUs

MORE CORES (100x)

EACH CORE IS SLOWER

USEFUL FOR PARALLEL TASKS



Slide Credit:

GPUs vs. CPUs

More benchmarks available here.

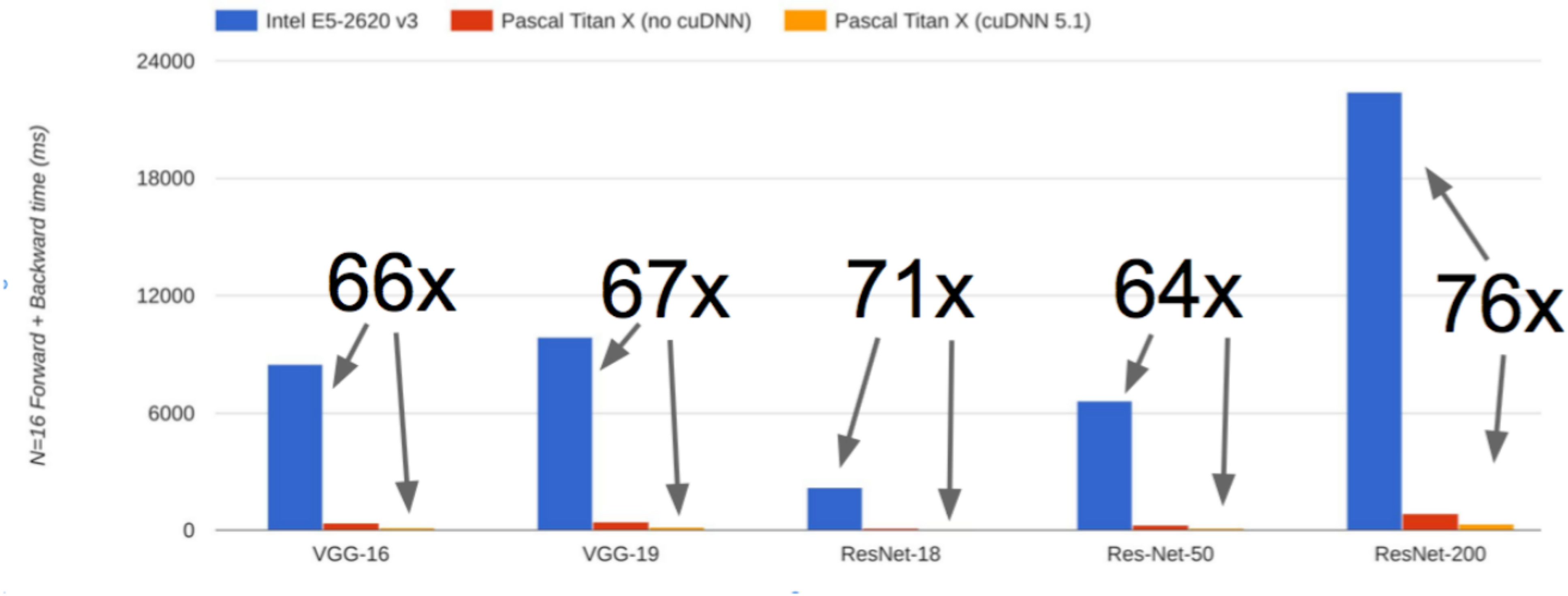


Figure credit: J. Johnson

GPUs for deep learning

NVIDIA GPUs ARE PROGRAMMED THROUGH CUDA [Compute Unified Device Architecture]

ANOTHER ALTERNATIVE IS OPENCL, SUPPORTED BY SEVERAL MANUFACTURES, LESS INVESTMENT [Way less used]

CuDNN IS A LIBRARY FOR SPECIFIC DEEP LEARNING COMPUTATIONS ON NVIDIA GPUs

THE PRICE TO PAY?

1. LARGE NUMBER OF PARAMETERS IMPLIES LARGE DATASETS TO TRAIN

2. LOOSE EVEN MORE DEGREE OF CONTROL OF WHAT THE ALGORITHM IS DOING SINCE THE FEATURE EXTRACTION PROCESS BECOMES UNSUPERVISED

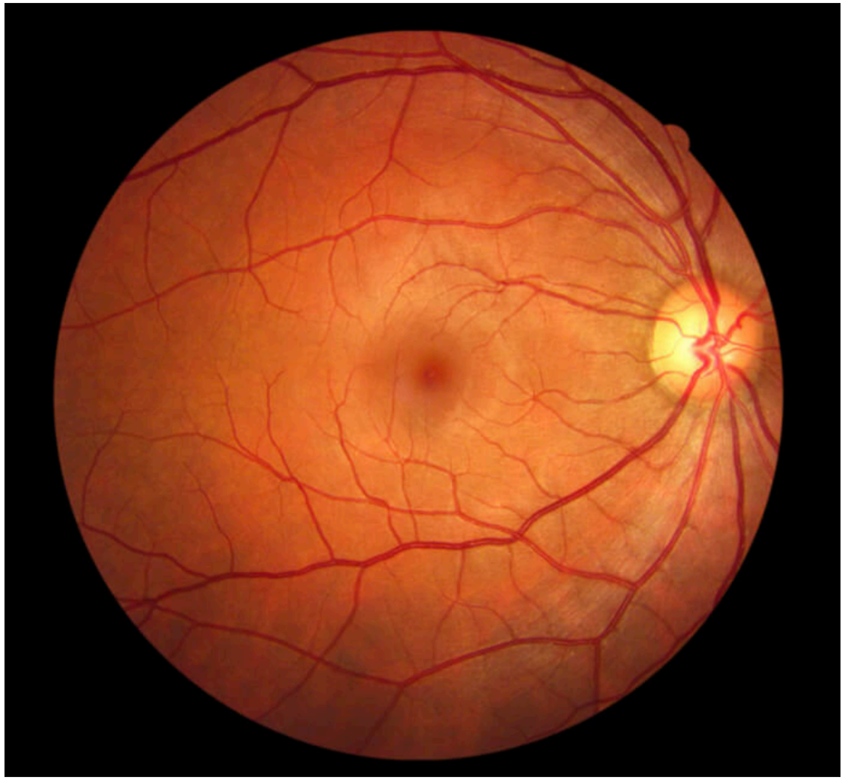


IMAGE OF THE BACK OF THE EYE

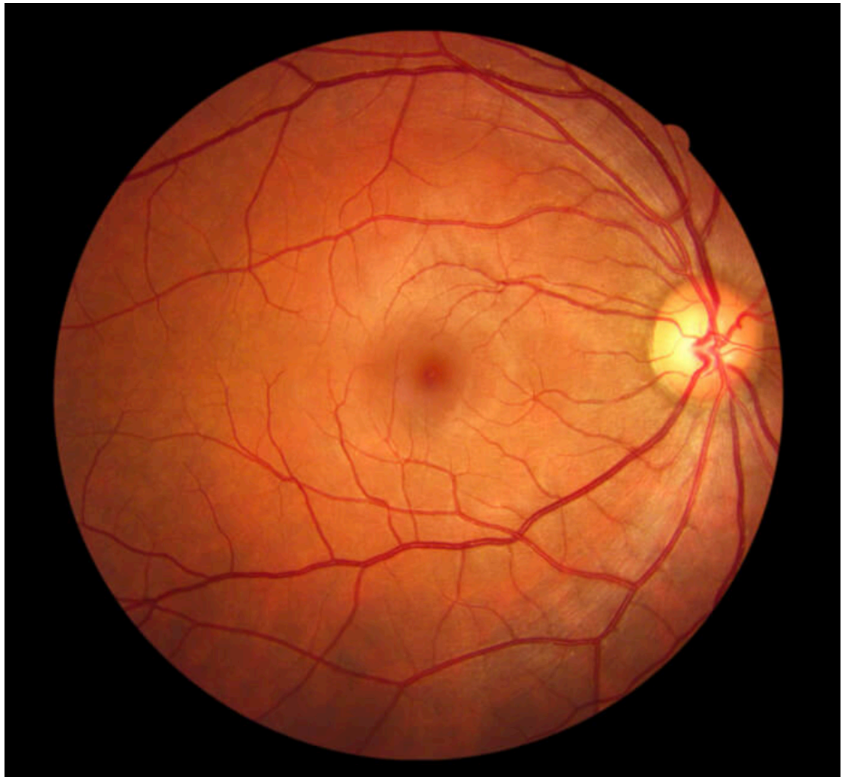


IMAGE OF THE BACK OF THE EYE

DEEP LEARNING CAN IDENTIFY THE PATIENT'S GENDER WITH 95% ACCURACY