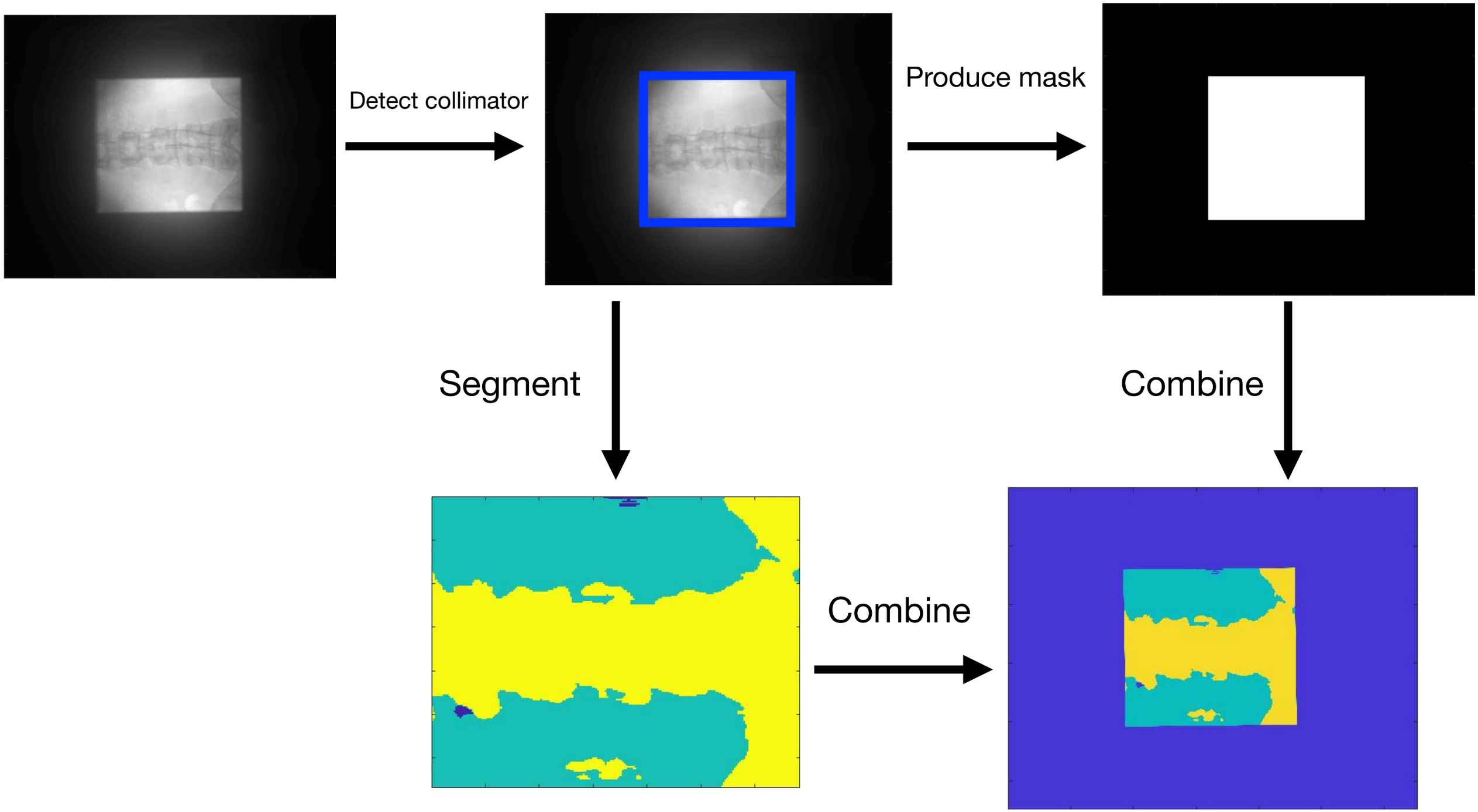
Automatic Prior Selection

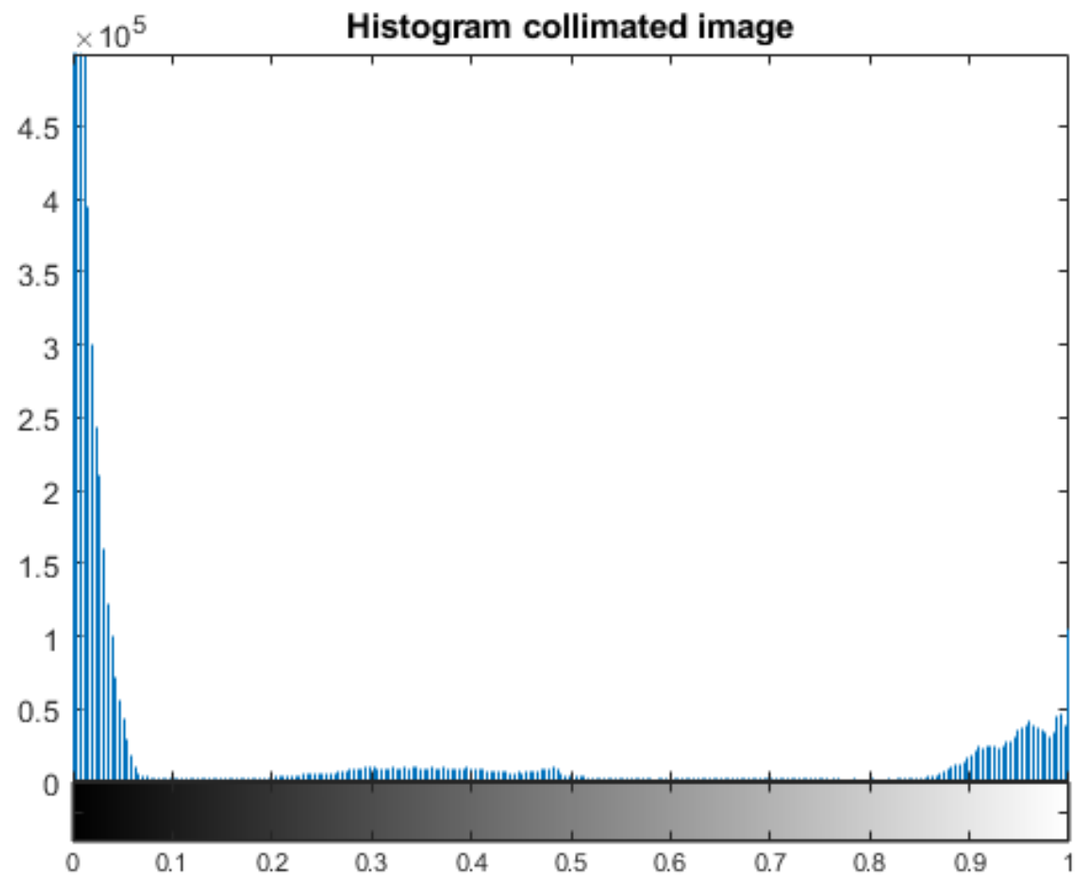
Joseph Bullock
Arnau Quera-Bofarull
Carolina Cuesta-Lázaro

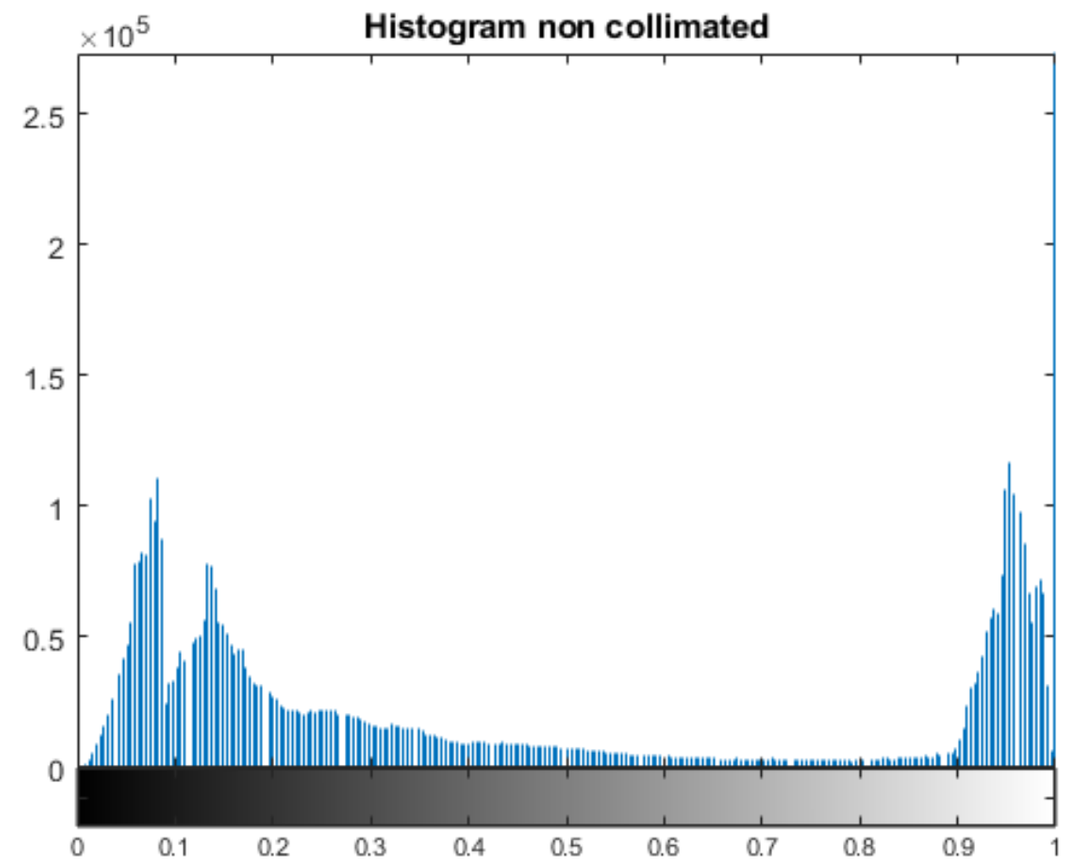
The Problem



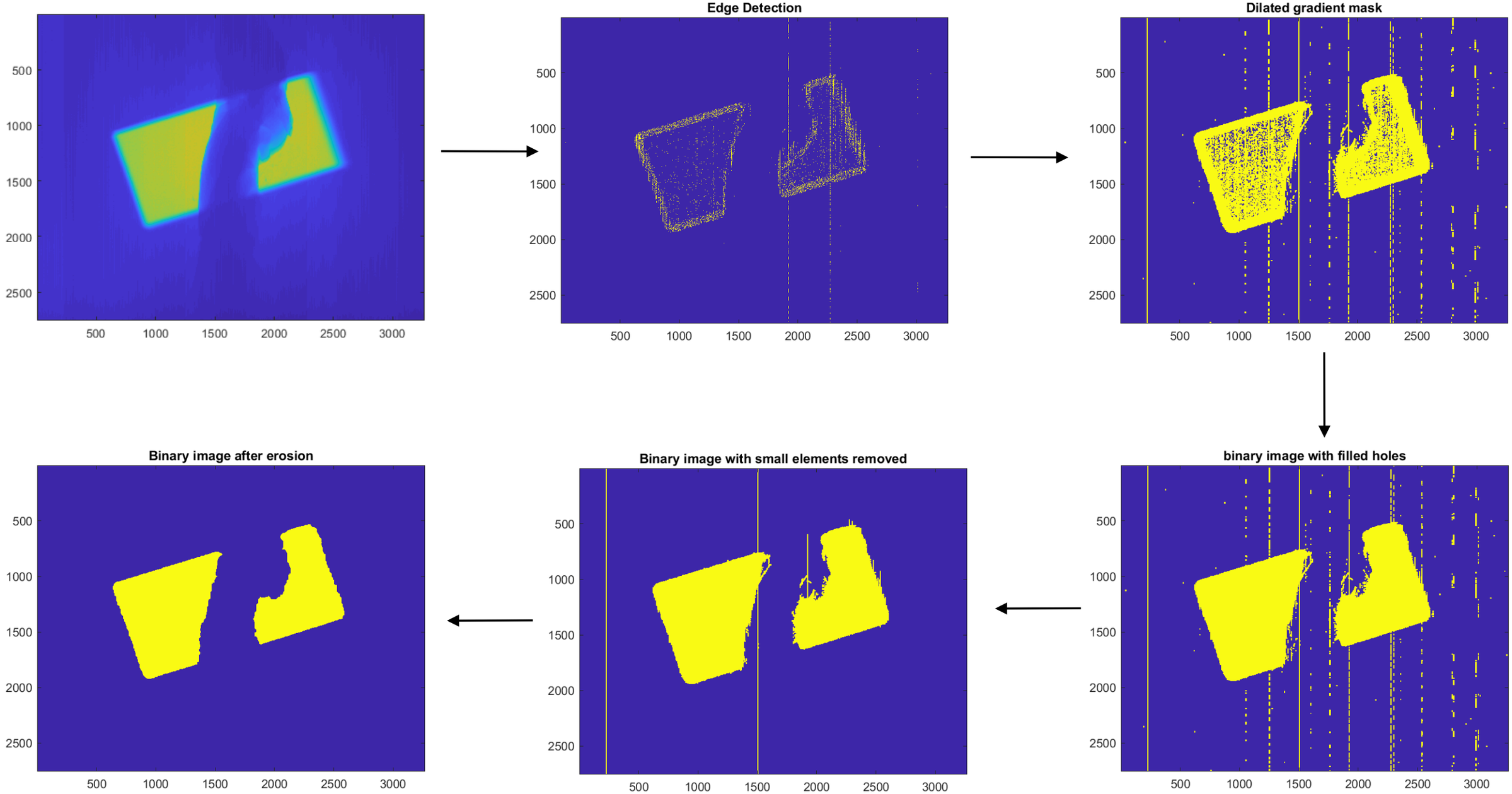
Collimator

Collimator Detection

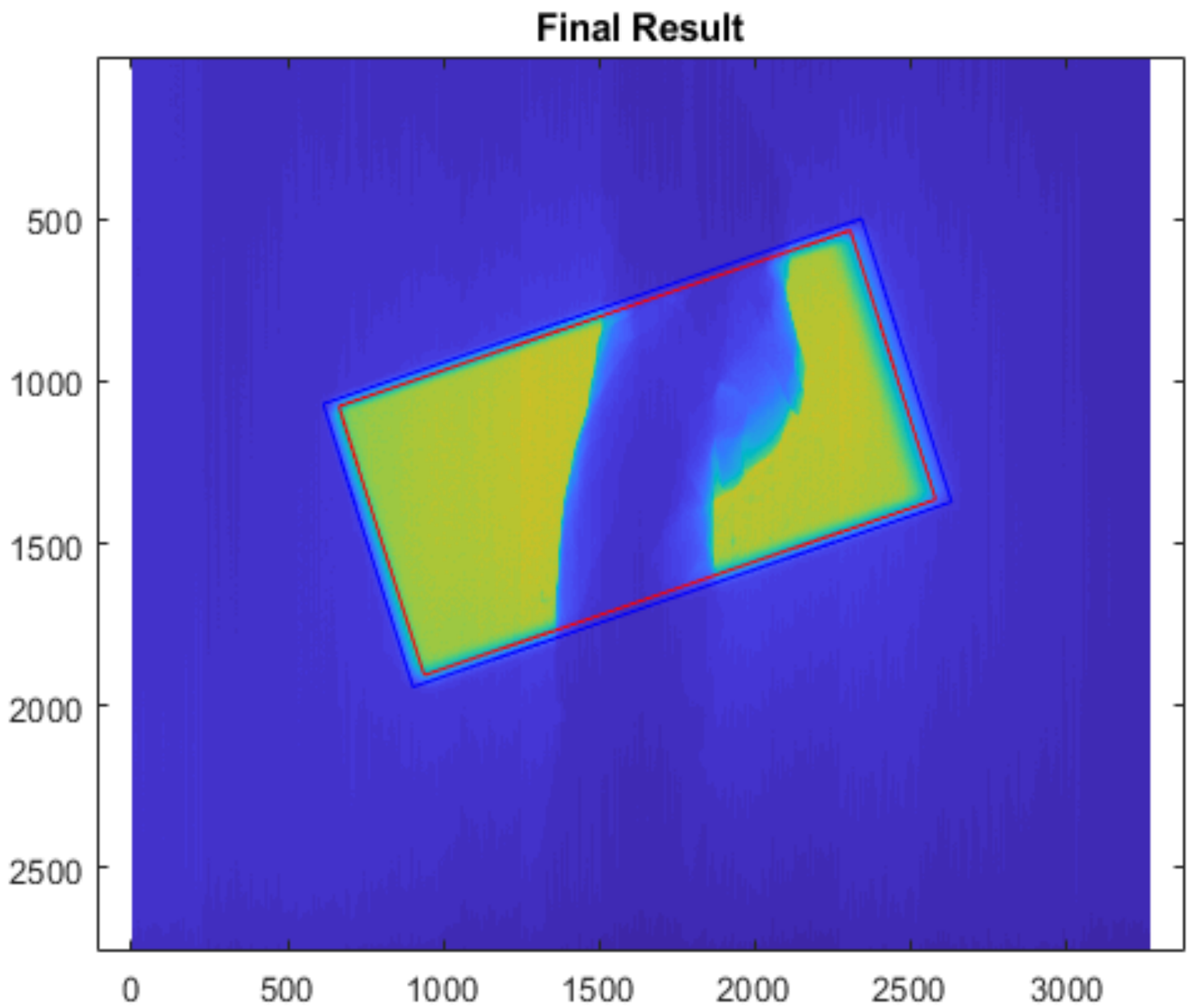




Filtering


Edge Detection
Dilated gradient mask
binary image with filled holes
Binary image with small elements removed
Binary image after erosion
500
1000
1500
2000
2500
3000

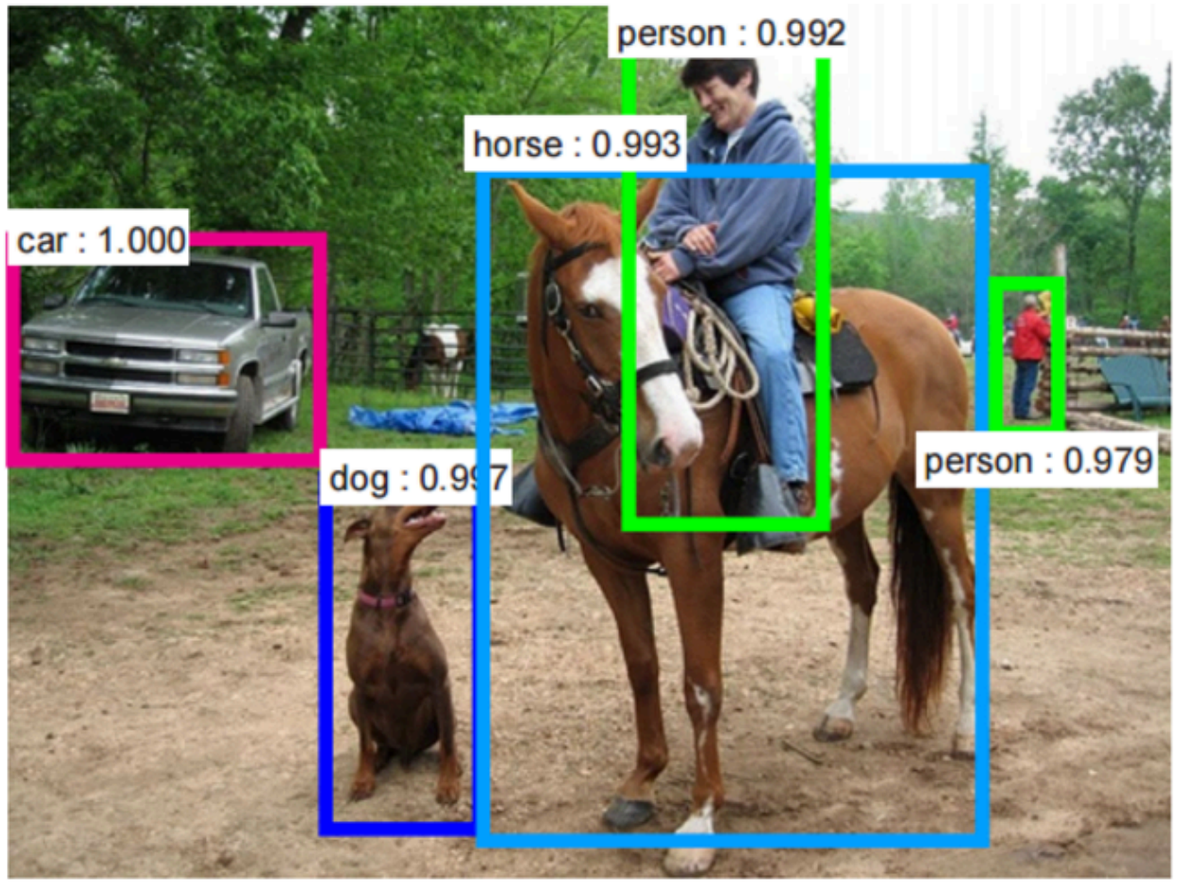
Final Result

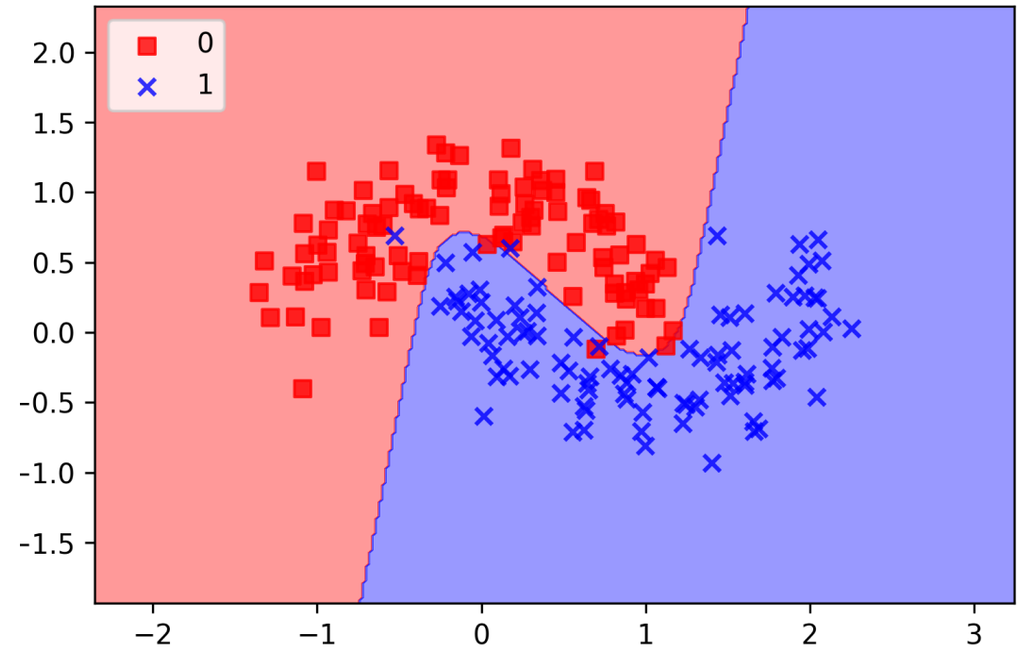


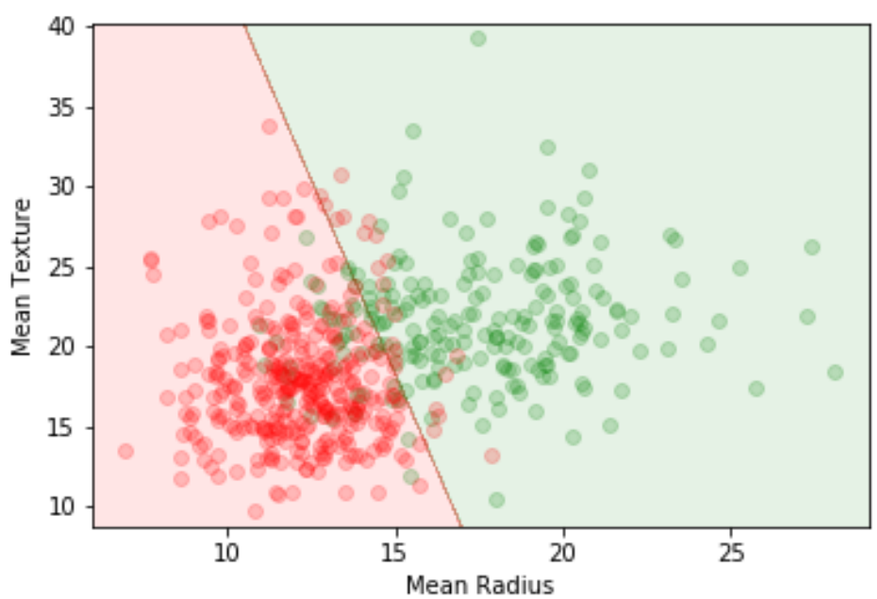
Neural Networks

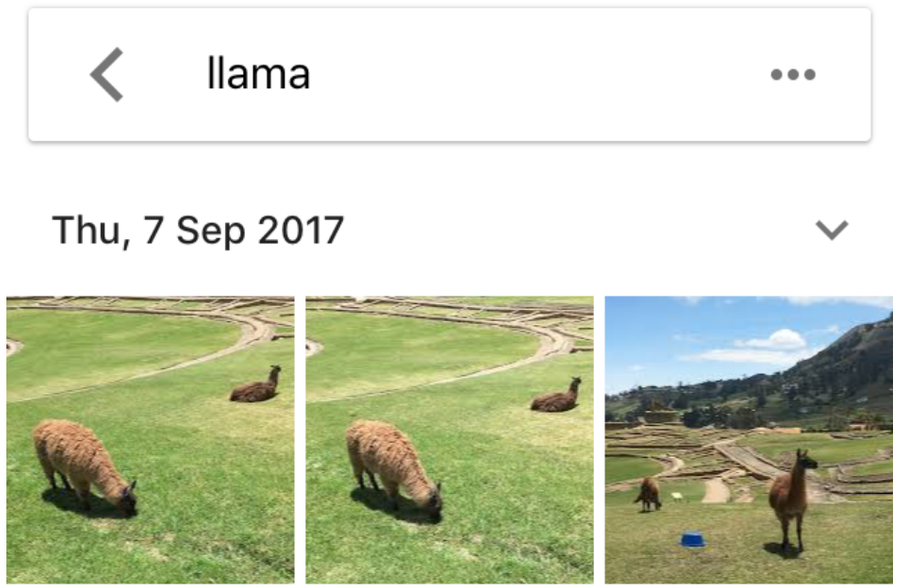
What is Machine Learning?

Classification problem



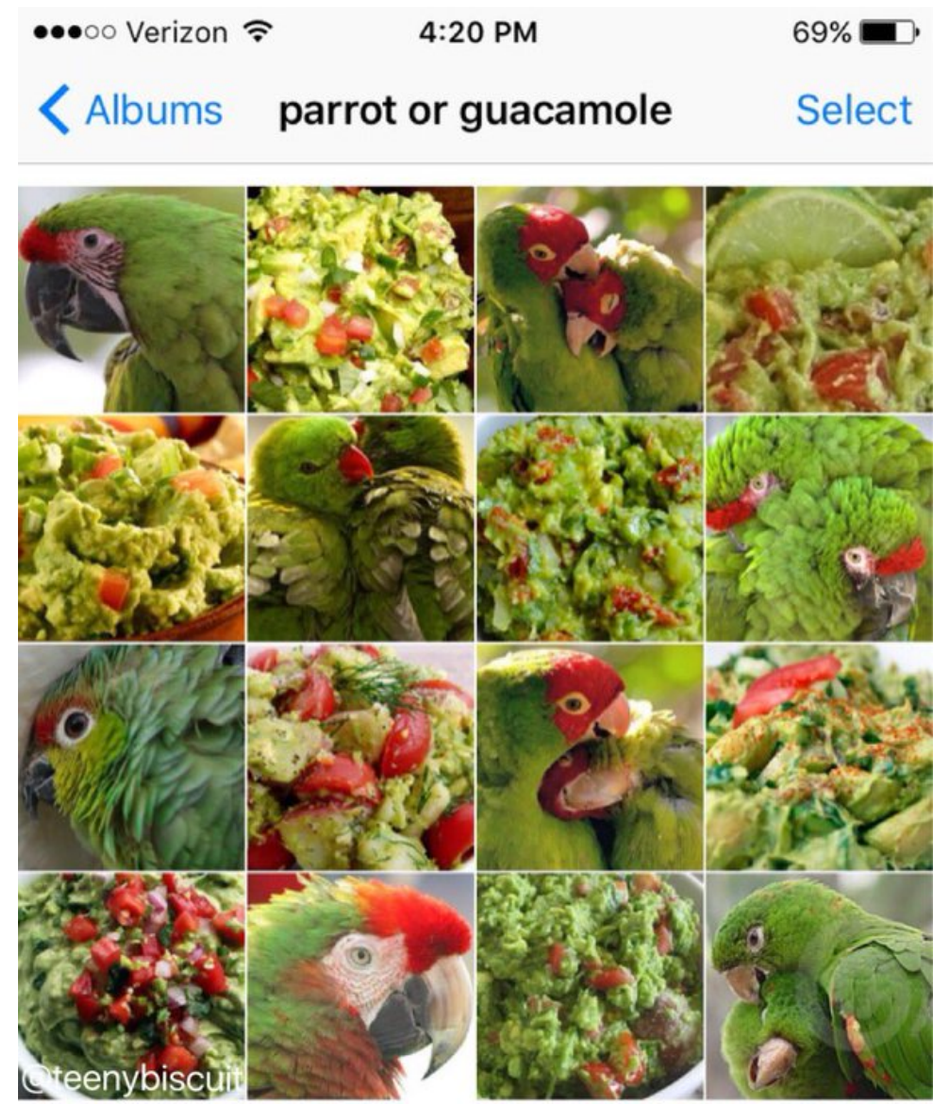






Left top: Data Science Blog; Left bottom: adapted from Maitre '18

Far from perfect

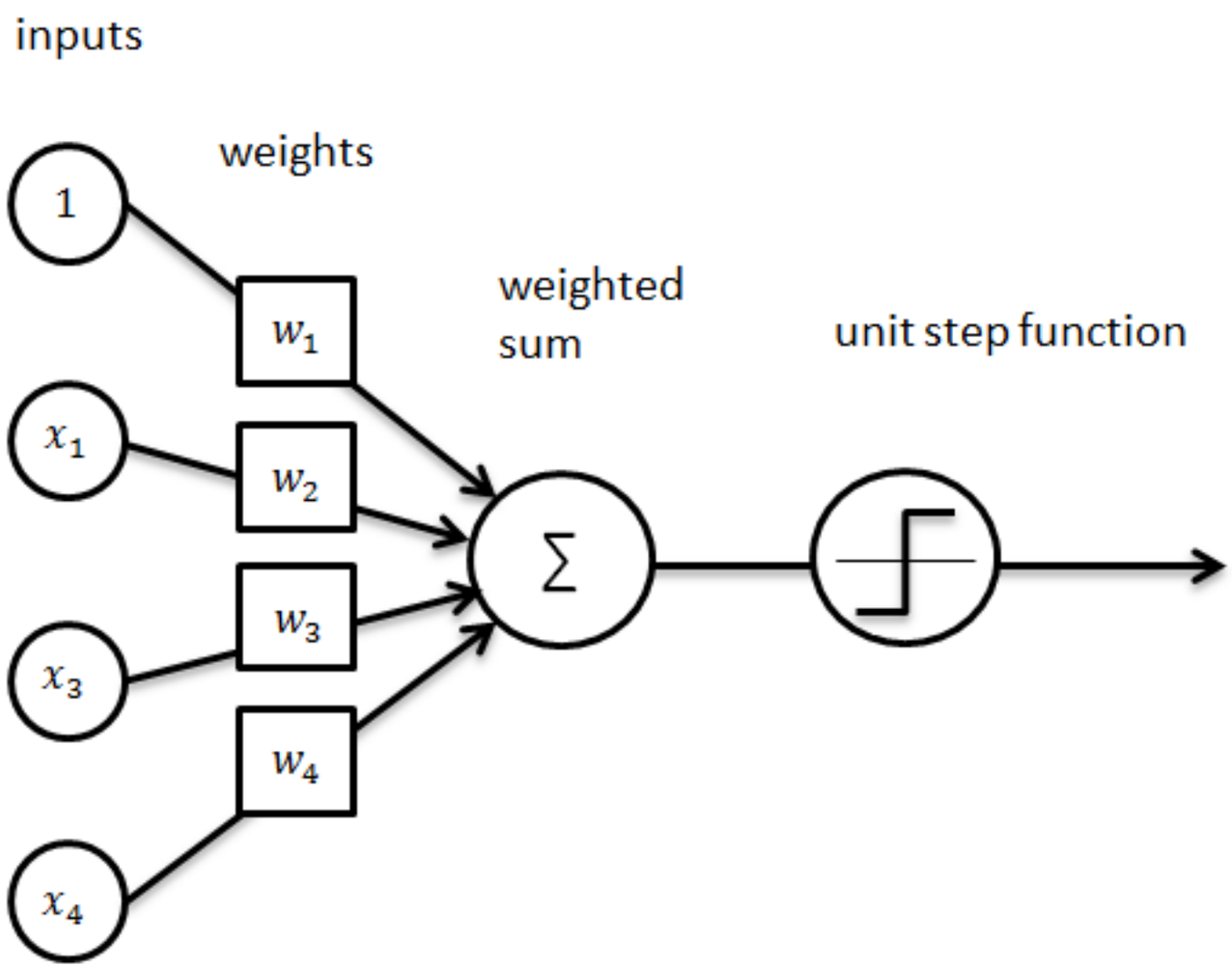


Why Machine Learning?

- Good at handling large amounts of data
- Can fit linear and non-linear functions efficiently
- Extracts features from a high dimensional feature space
- Good at automatic de-noising

Perceptron

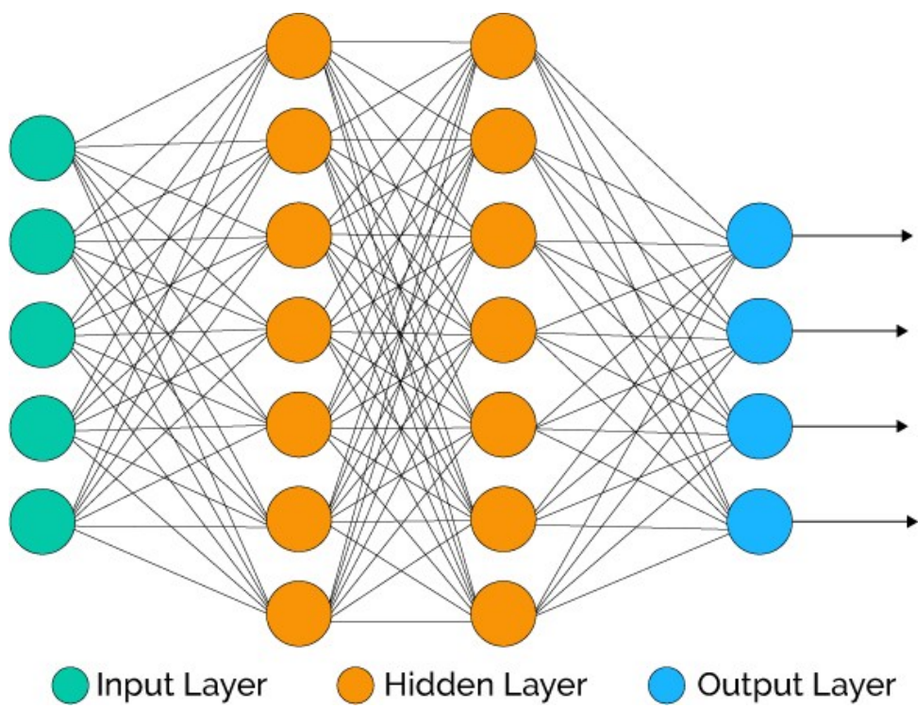
- Most basic form of neural network
- Neural network are a collection of nodes with weighted links between them

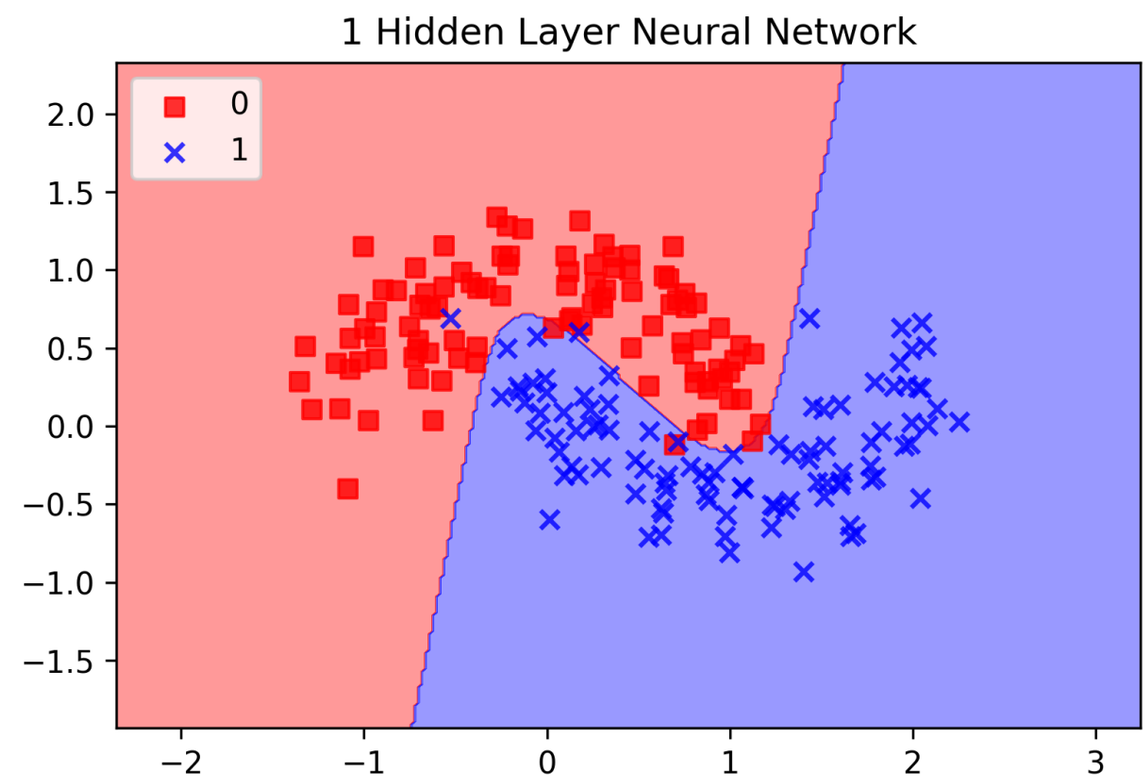


$$J(z^{(i)}) = \frac{1}{2}|f(z^{(i)}) - y^{(i)}|^2$$

[Image: Taspinar '16]

Deep Learning



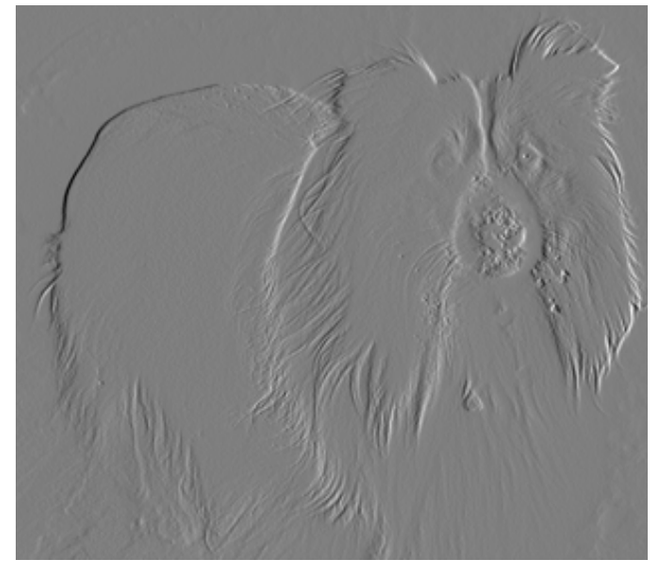


[Left image: Medium; Right image: Reppel '17]

[Edge detection: Goodfellow et al '16; Bottom right image: Reppel '17]

Convolutional Neural Network

Input



Output

1	-1

Kernel

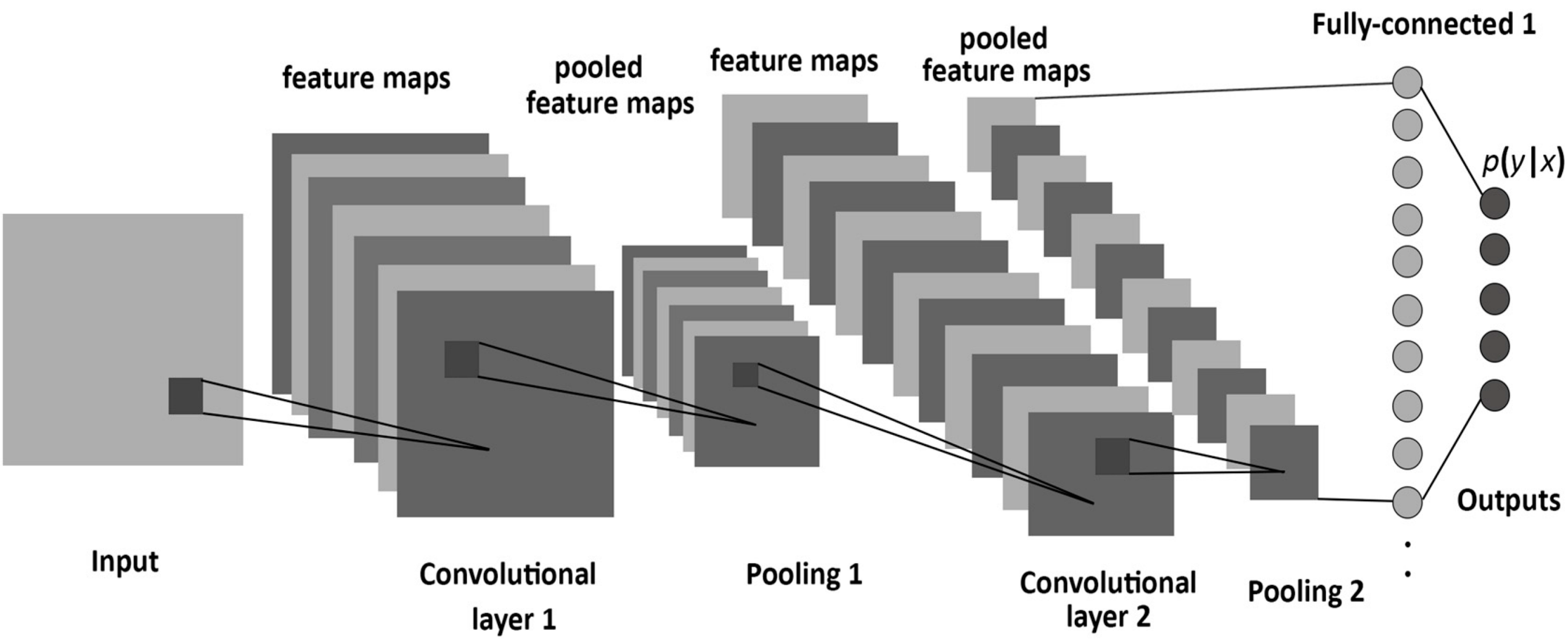
$1_{\times 1}$	$1_{\times 0}$	$1_{\times 1}$	0	0
$0_{\times 0}$	$1_{\times 1}$	$1_{\times 0}$	1	0
$0_{\times 1}$	$0_{\times 0}$	$1_{\times 1}$	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature

Convolutional Neural Network



[Image: Albelwi and Mahmood '17]

Data Labelling and Augmentation

Data collection and labelling

- Built up a database of 150 images across a range of bodypart classes
- Used GIMP to label the images by performing manual pixel-by-pixel segmentation
- Segmented into 3 regions: bone, soft tissue, open beam
- Augmented images to gain ~9000 images in total

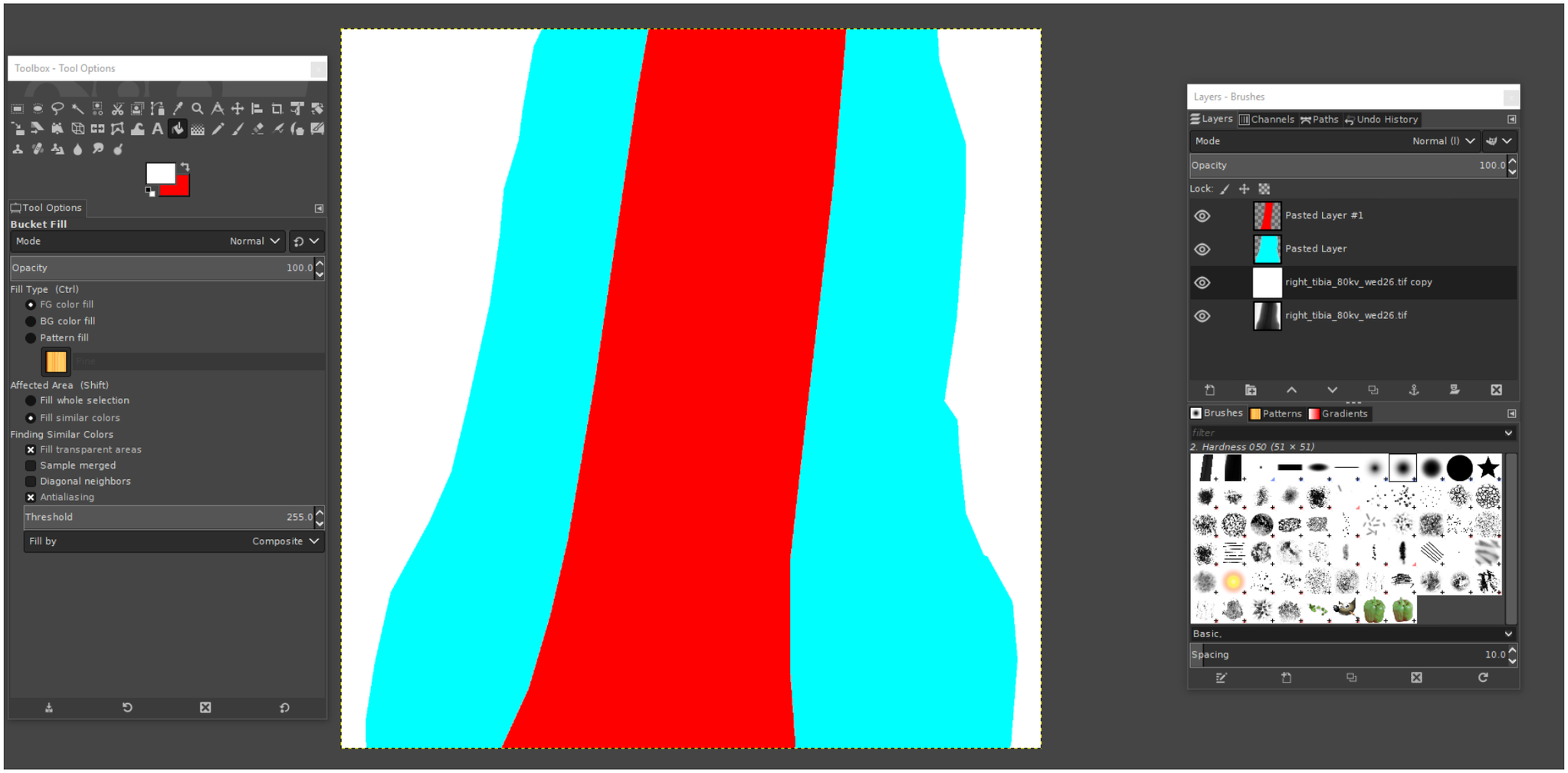
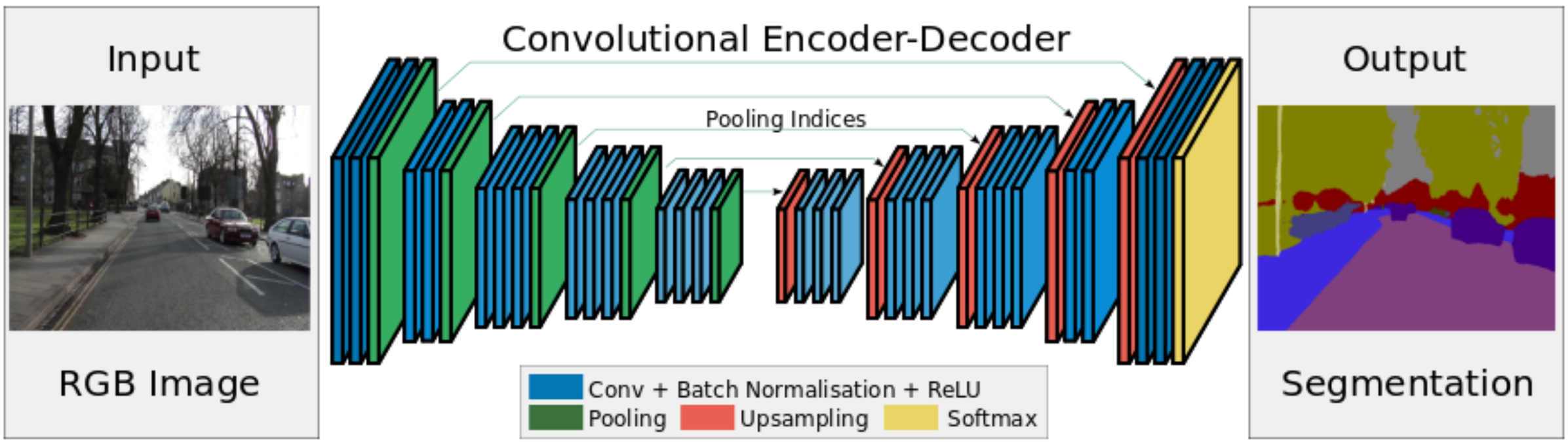


Image Segmentation

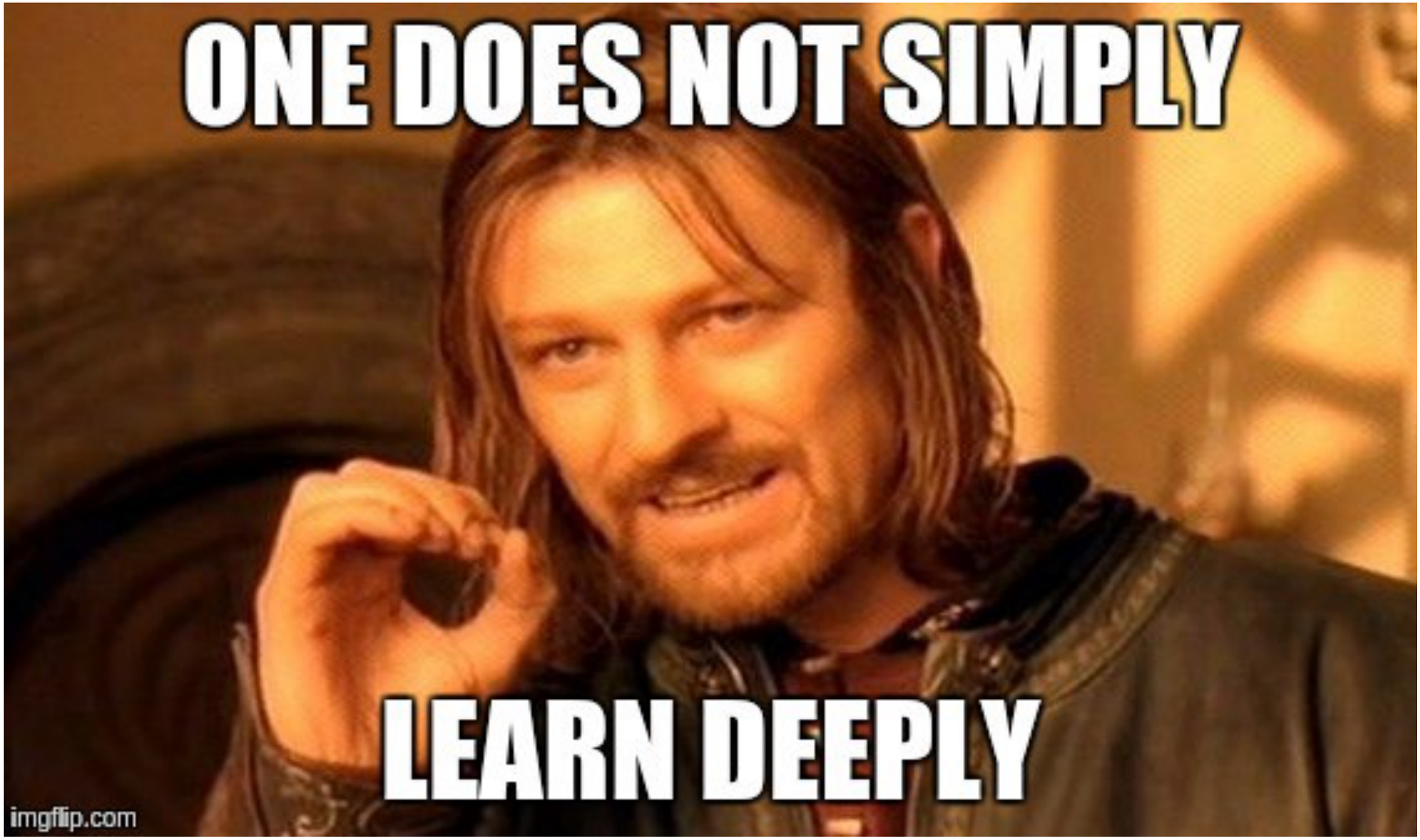
Image Segmentation

SegNet



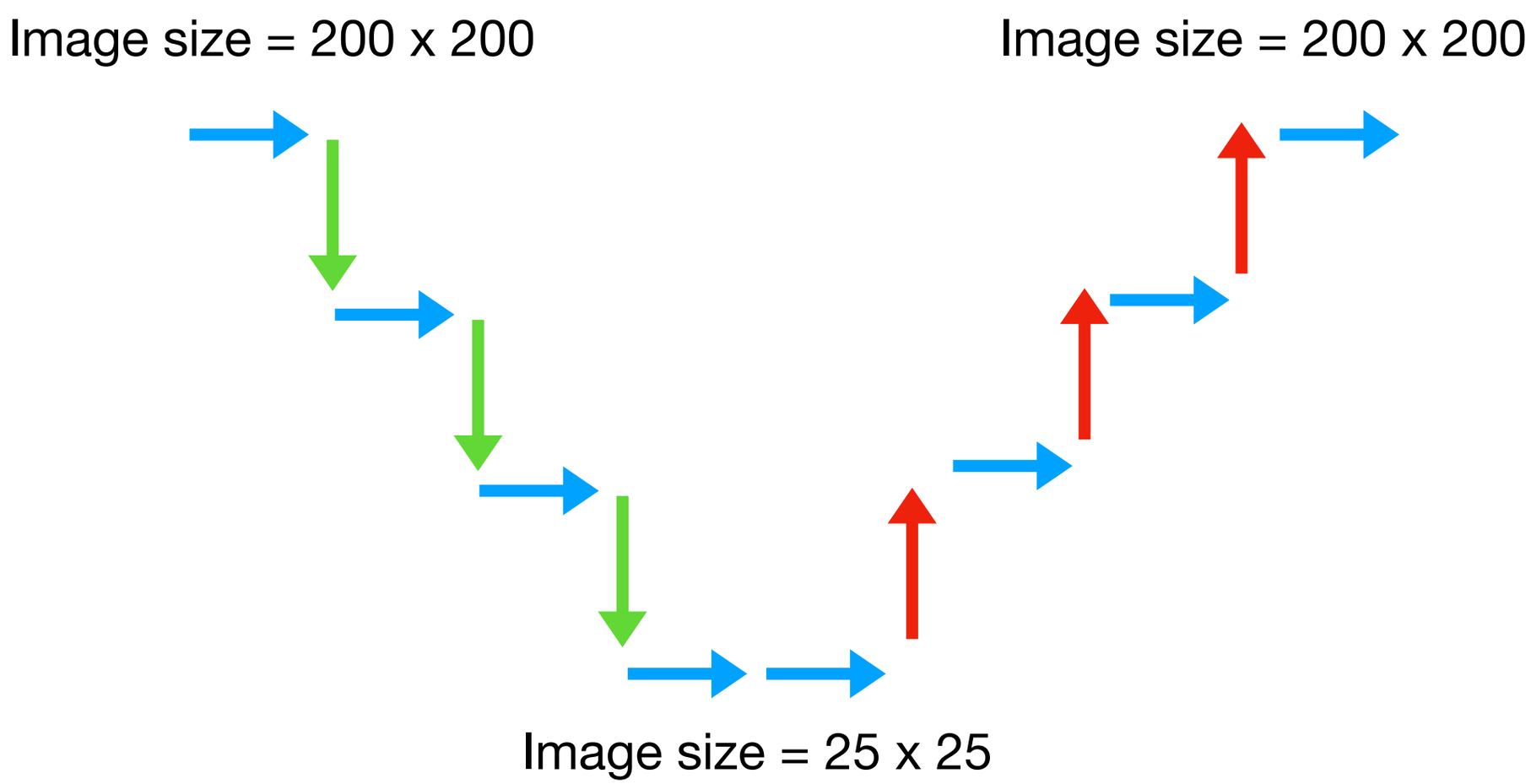
- Performs very well on complex image segmentation
- Requires a very large dataset to train well partly due to the large number of parameters

[Image: Kendall, Badrinarayanan and Cipolla '17]


ONE DOES NOT SIMPLY
LEARN DEEPLY
imgflip.com

Network Architecture

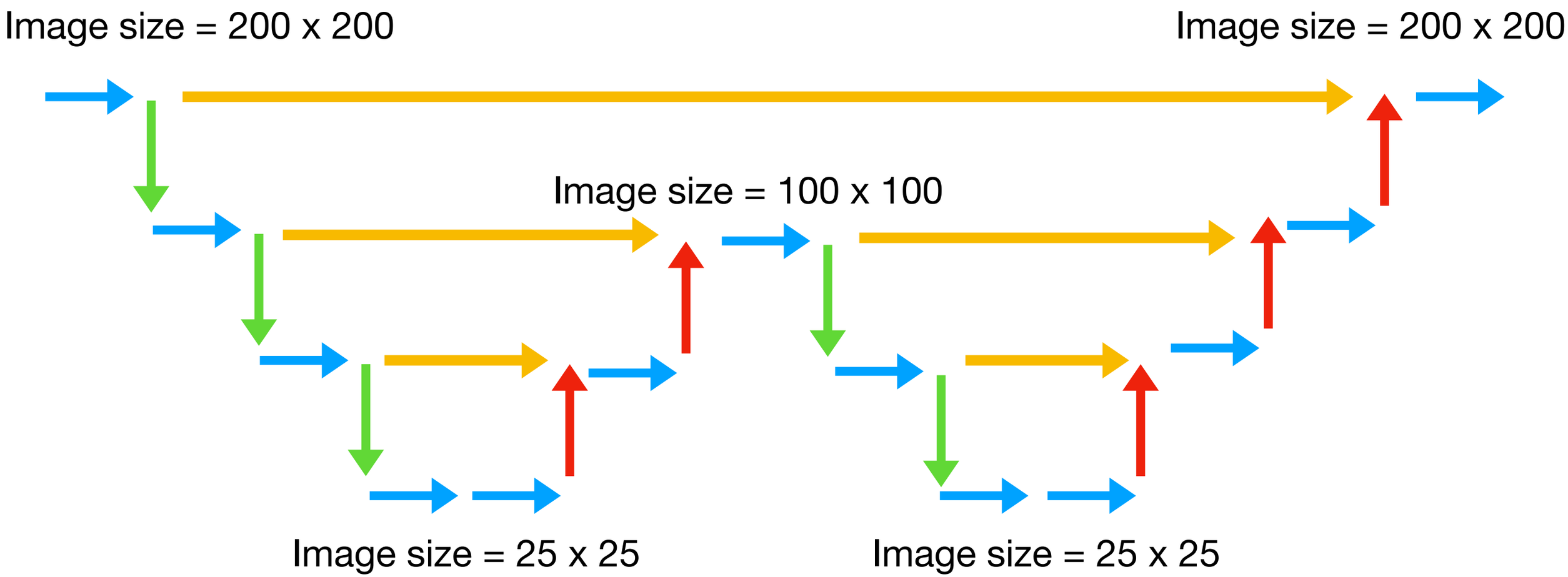
Initial attempt - simplified SegNet



- Performed relatively well on our smaller dataset
- Low true positive rate after post processing of the output

Network Architecture

Adapting the architecture



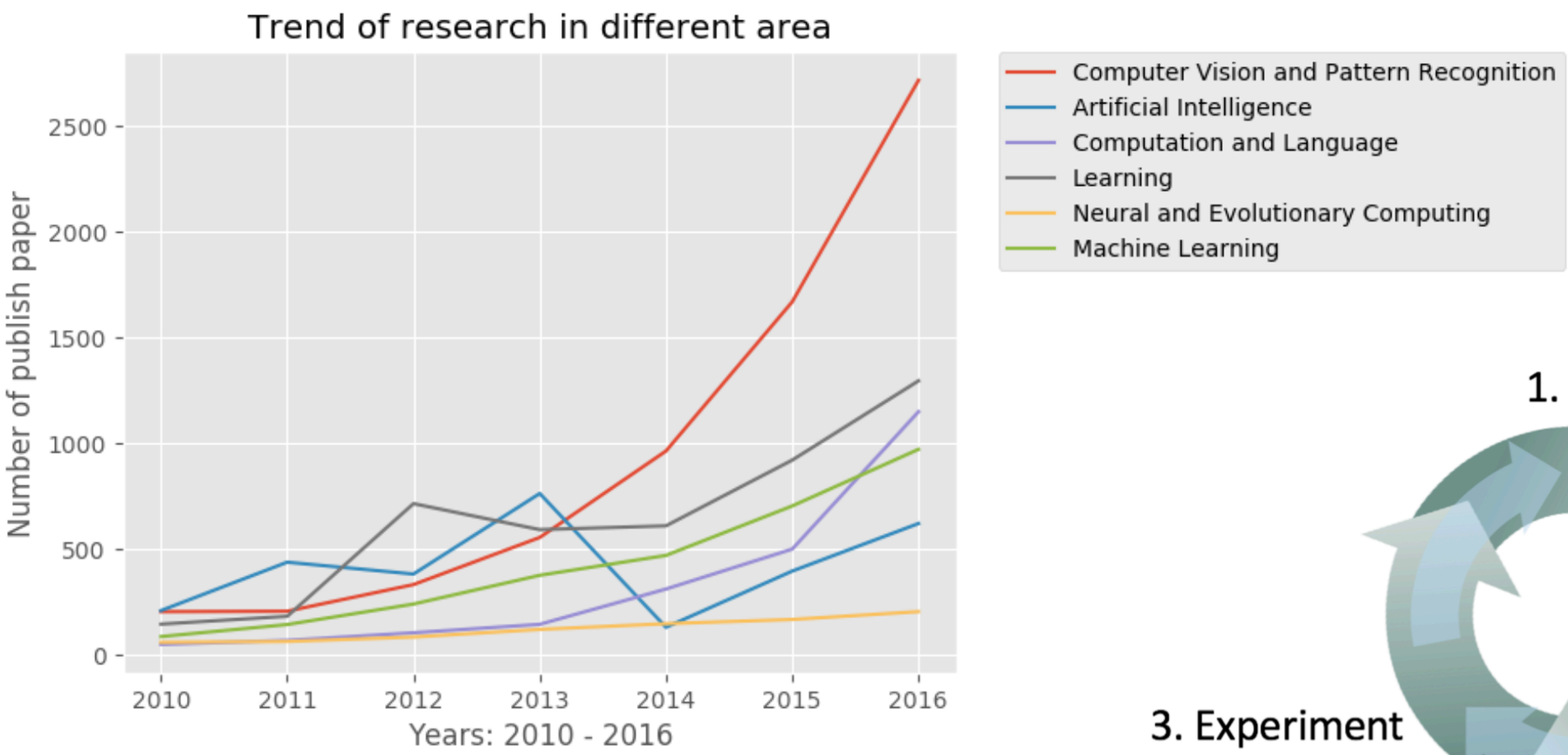
- Fits better as it has more parameters so can perform more complex feature extraction
- Weight sharing, layer normalisation and not increasing the ‘depth’ of the network helps prevent overfitting

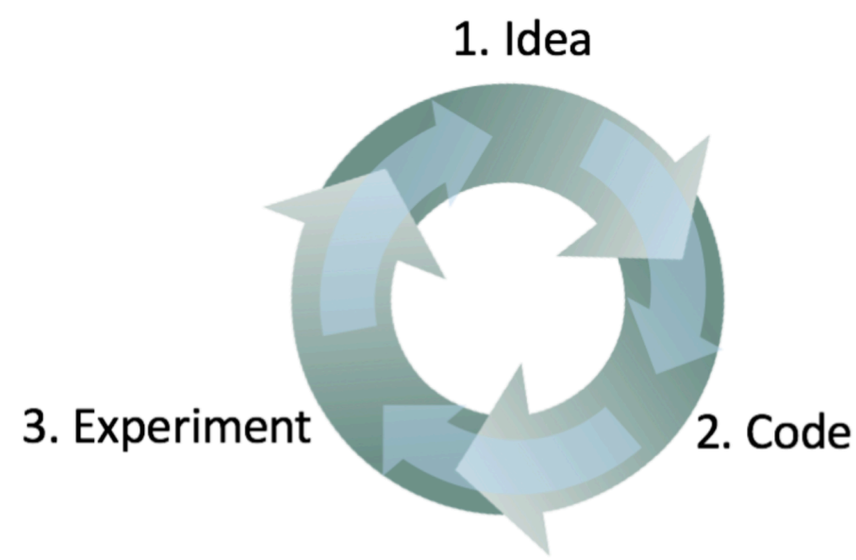
Learning Curves and Postprocessing

Development Process

Development Process

- The development process takes a long time due to hyperparameter tuning
- Architecture development can take many forms
- Research in this field fast paced and publications are constantly suggesting new architectural and hyperparameter changes for enhanced performance



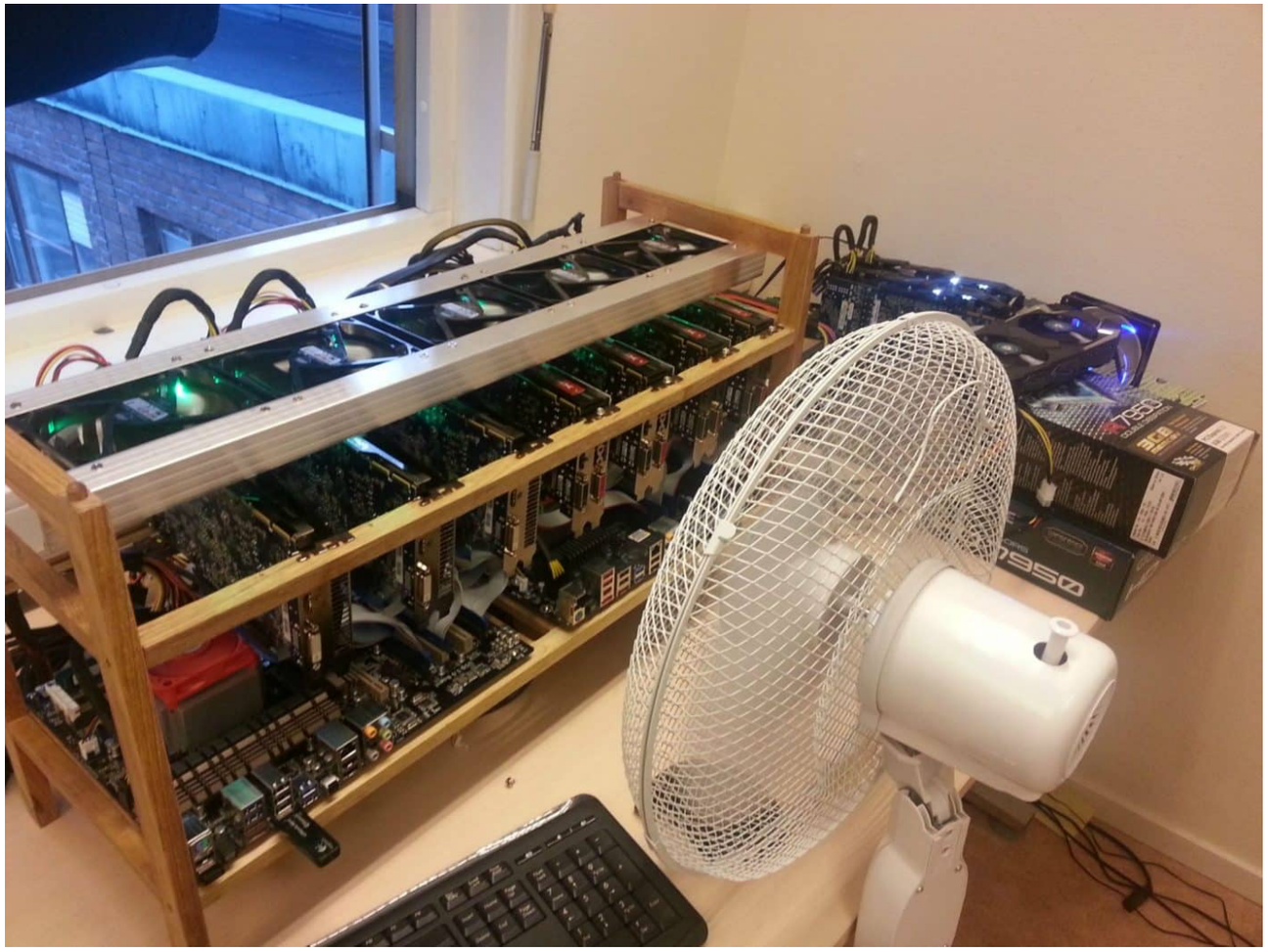


[Graph: Shah '17; Image: Ng '18]

Development Process

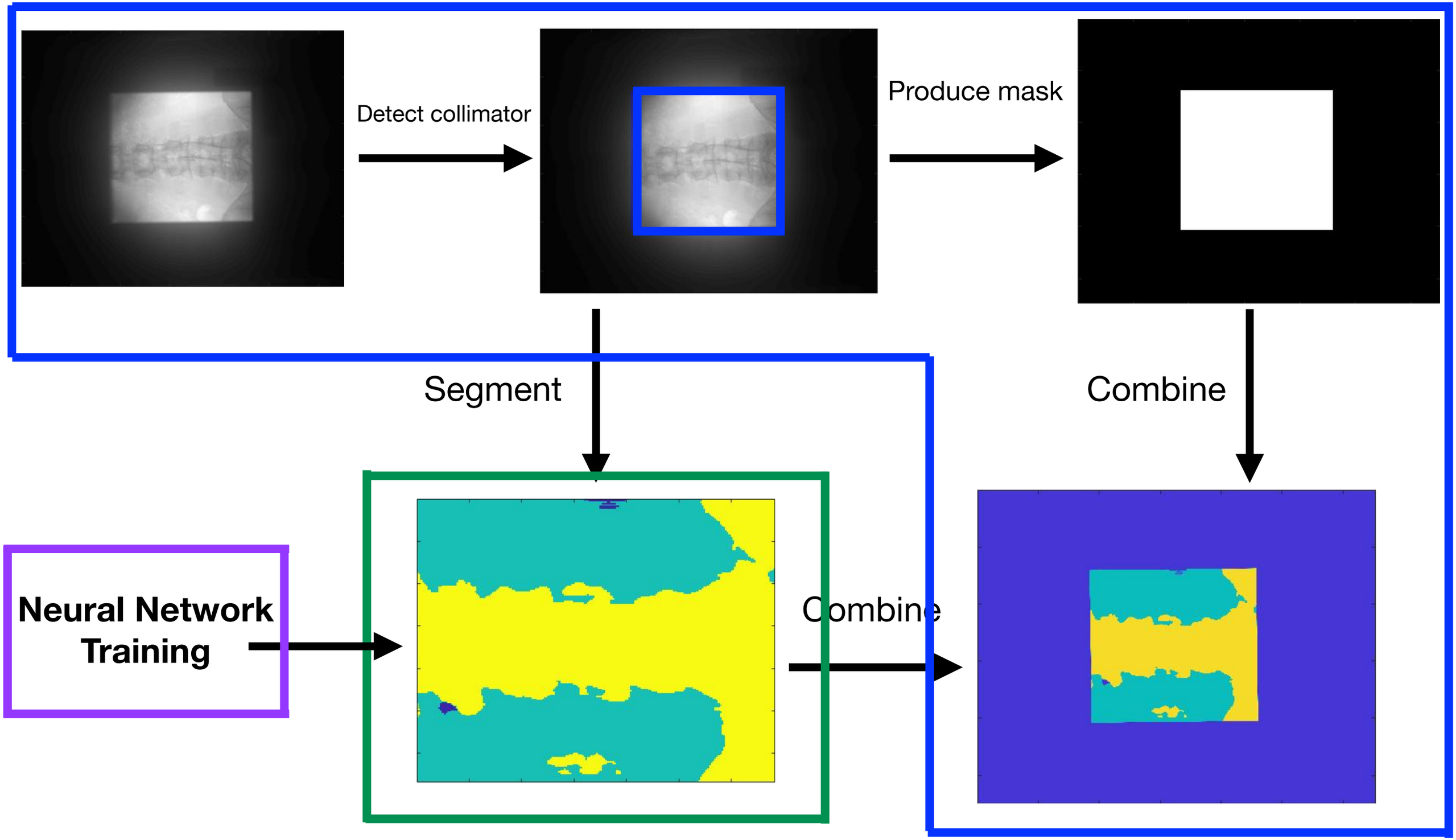
What is a long time?

- Architectures were being designed from week 2
- Hyperparameter tuning was being carried out from week 3/4
- Used ~1000 GPU hours across:
  - 3x4GB GPUs
  - 1x8GB GPU
  - AWS 12GB GPU



Conclusions

Final Product



Future Work

- Using random forest (i.e. filter selection) techniques for postprocessing
- Further hyperparameter tuning
- Data collection
- Application to anomaly detection
  - Machine learning well equipped for this process

Learning Outcomes

- Knowledge of building and executing a machine learning product in an industry setting - transferable to academia
- Project planning
- Image segmentation expertise
- Cloud computing
- MATLAB