MultiDrone

Deep Learning for Target Detection

Presenter: **Prof. Anastasios Tefas**
Aristotle University of Thessaloniki
tefas@csd.auth.gr
www.multidrone.eu

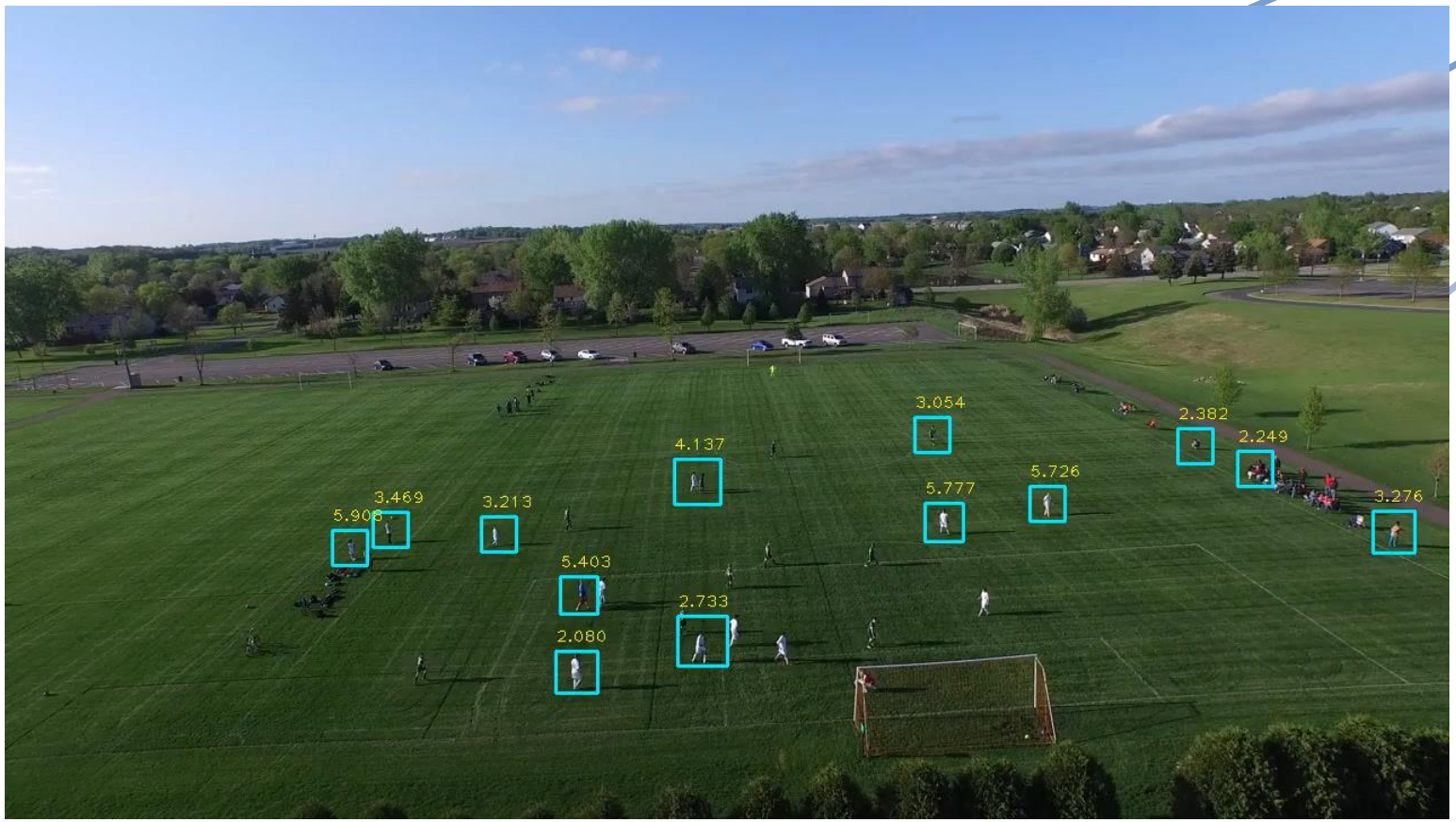
Contributors: V. Nousi, D. Triantafyllidou, M. Tzelepi, A. Tefas, N. Nikolaidis, I. Pitas (Aristotle University of Thessaloniki)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Target detection

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Target detection



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Target detection

- **Target/object examples: athletes, boats, biclycles.**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Target detection

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

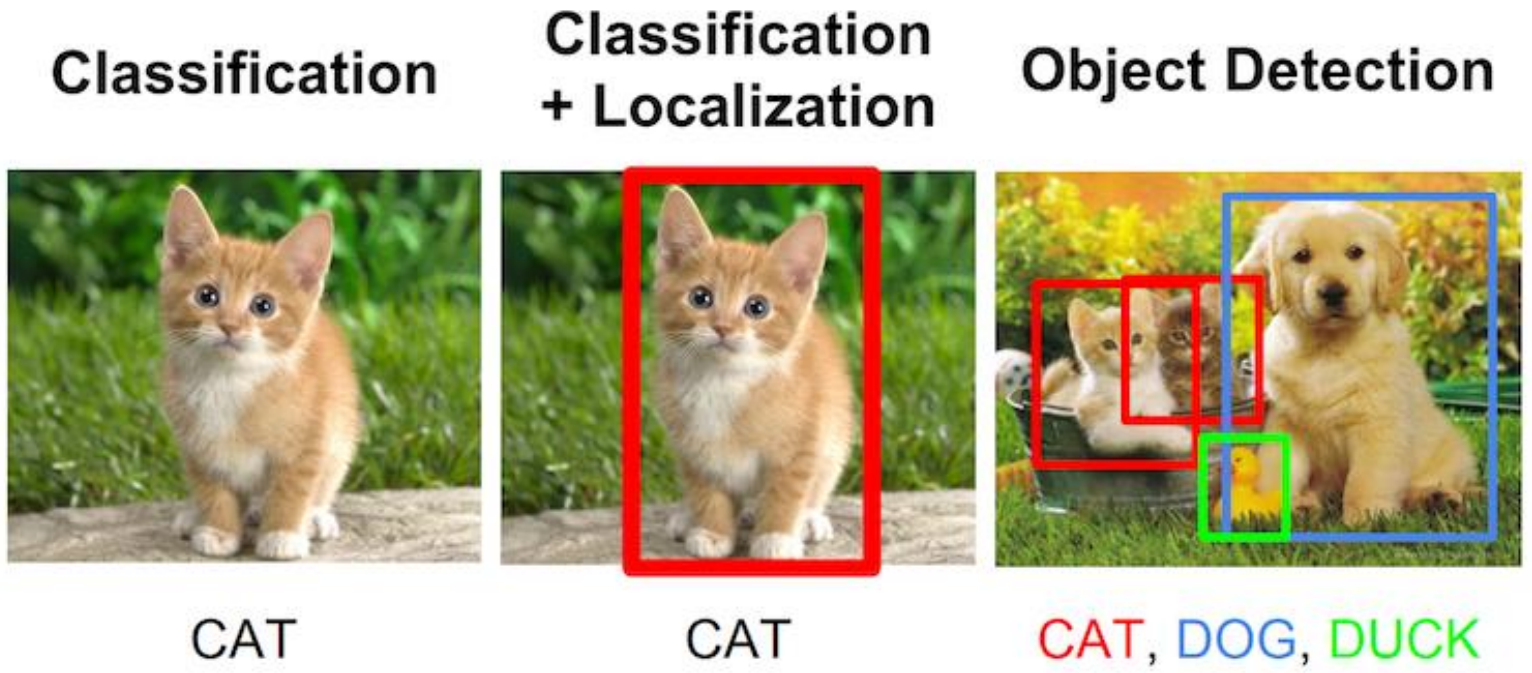
Object detection

- **Single view object detection**
  - **Deep learning (CNN) object detection.**
  - Light weight CNNs for object detection.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- Object detection = classification + localization:
- Find **what** is in a picture as well as **where** it is.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **Input**: an image.
- **Output**: **bounding boxes** containing depicted objects.
  - Each image contains a **different number of objects (outputs).**
- Typical approach: train a **specialized classifier** and deploy in **sliding-window style** to detect all object of that class.
  - Very inefficient, quite ineffective.
- **Goal**: combine classification and localization into a **single architecture for multiple, multiclass object detection.**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Classification and Regression

- **Classification:** If we have a class label set $\boldsymbol{C} = \{c_1, \ldots, c_L\}$, train a NN model to assign a class label vector $\hat{\boldsymbol{y}} \in [0,1]^L$ to an object $\boldsymbol{x}$: $\hat{\boldsymbol{y}} = f_{NN}(\boldsymbol{x}, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ are the CNN trainable parameter vector.
  - Essentially, we assign (predict) probabilities $P(\hat{\boldsymbol{y}} \mid \boldsymbol{x})$ that an object $\boldsymbol{x}$ belongs to each of the L classes.
  - **Training:** Given $N_{training}$ ground truth pairs $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}$, $i = 1, \ldots, N_{training}$, estimate $\boldsymbol{\theta}$ by minimizing an error function $\min_{\boldsymbol{\theta}} J(\boldsymbol{y}\text{-}\hat{\boldsymbol{y}})$.
  - **Testing:** Given N_{test} ground truth validation pairs $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}, i = 1, \ldots, N_{test}$ calculate (*predict*) $\hat{\boldsymbol{y}}_i, i = 1, \ldots, N_{test}$ and calculate *a performance metric*.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Classification and Regression

- **Classification:**
  - Two class (*L=2*) and multiple class (*L>2*) classification.
  - **Example:** *Face detection* (two classes), *face recognition* (many classes).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Classification and Regression

- **Regression:** If we have a function $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$, train a NN model to predict **real-valued quantities** (vector $\boldsymbol{y}$ entries), $\hat{\boldsymbol{y}} = f_{NN}(\boldsymbol{x}, \boldsymbol{\theta})$, so that an error function $\min_{\boldsymbol{\theta}} J(\boldsymbol{y} - \hat{\boldsymbol{y}})$ is minimized.
  - **Training:** Given $N_{training}$ ground truth pairs $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}, i = 1, \dots, N_{training}$, estimate $\boldsymbol{\theta}$ by minimizing an error function $\min_{\boldsymbol{\theta}} J(\boldsymbol{y} - \hat{\boldsymbol{y}})$.
  - **Testing:** Given N_{test} ground truth validation pairs $\{\boldsymbol{x}_i, \boldsymbol{y}_i\}, i = 1, \dots, N_{test}$ calculate (*predict*) $\hat{\boldsymbol{y}}_i, i = 1, \dots, N_{test}$ and calculate an error function $J(\boldsymbol{y} - \hat{\boldsymbol{y}})$, e.g. MSE.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Classification and Regression

- **Regression:**
  - **Example:** In object detection, regress object ROI parameters (width *W*, height *H*, offsets *X*, *Y*).
  - **Function approximation:** it is essentially regression, when the function $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$ is known.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **Classification:** train a model to assign a category to an object, i.e., predict a probability of the object belonging to a certain class, i.e. *Pr(class | object).*

- **Regression:** train a model to predict **real-valued quantities**, e.g., ROI width W, height H, x and y offsets.

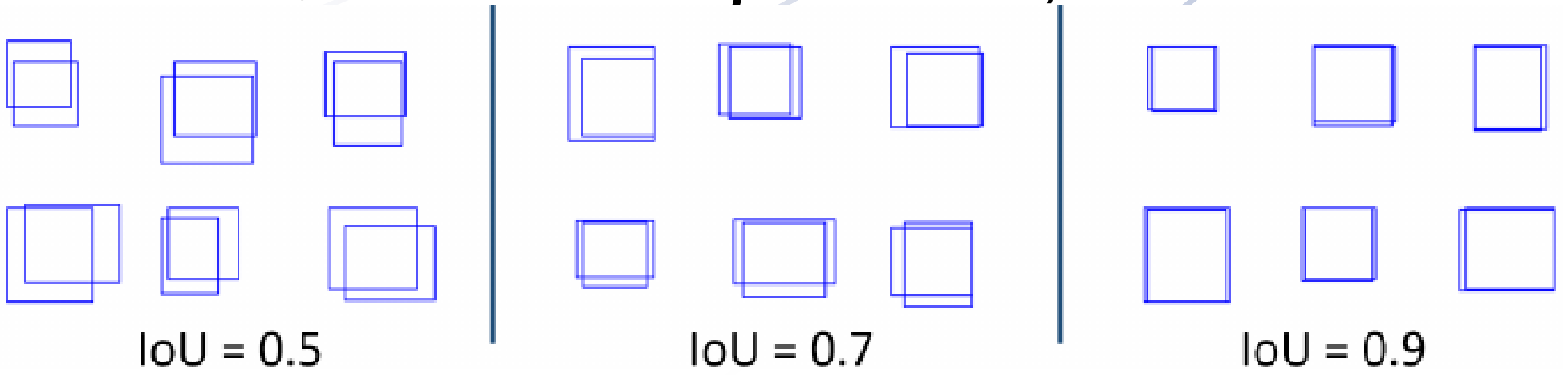
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection performance metrics

- **IoU**: Intersection over Union of predicted ROI (bounding box) *A* with ground truth ROI *B:*

  $J(A, B) = |A \cap B| / |A \cup B|$

- Also called **Jaccard Similarity Coefficient**).



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection performance metrics

- Object detection output: bounding boxes A_{ij} with corresponding confidence scores s_{ij}.
- If A_{ij} is matched to a groundtruth box B_{ik}, according to $J(A_{ij}, B_{ik}) > T(B_{ik})$, then $z_{ij} = 1$.
- The threshold $T(B_{ik})$ depends on the box size:

$$T(B_{ik}) = min(0.5, H*W / (H+1)*(W+1)).$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection performance metrics

- **Recall, Precision** definitions.
- For a confidence threshold t (real number):

$$recall(t) = \Sigma_{ij}\ 1[s_{ij}>=t]z_{ij}/C, \qquad (C \text{ is the number of classes})$$

$$precision(t) = \Sigma_{ij}\ 1[s_{ij}>=t]z_{ij} / \Sigma_{ij}\ 1[s_{ij}>=t].$$

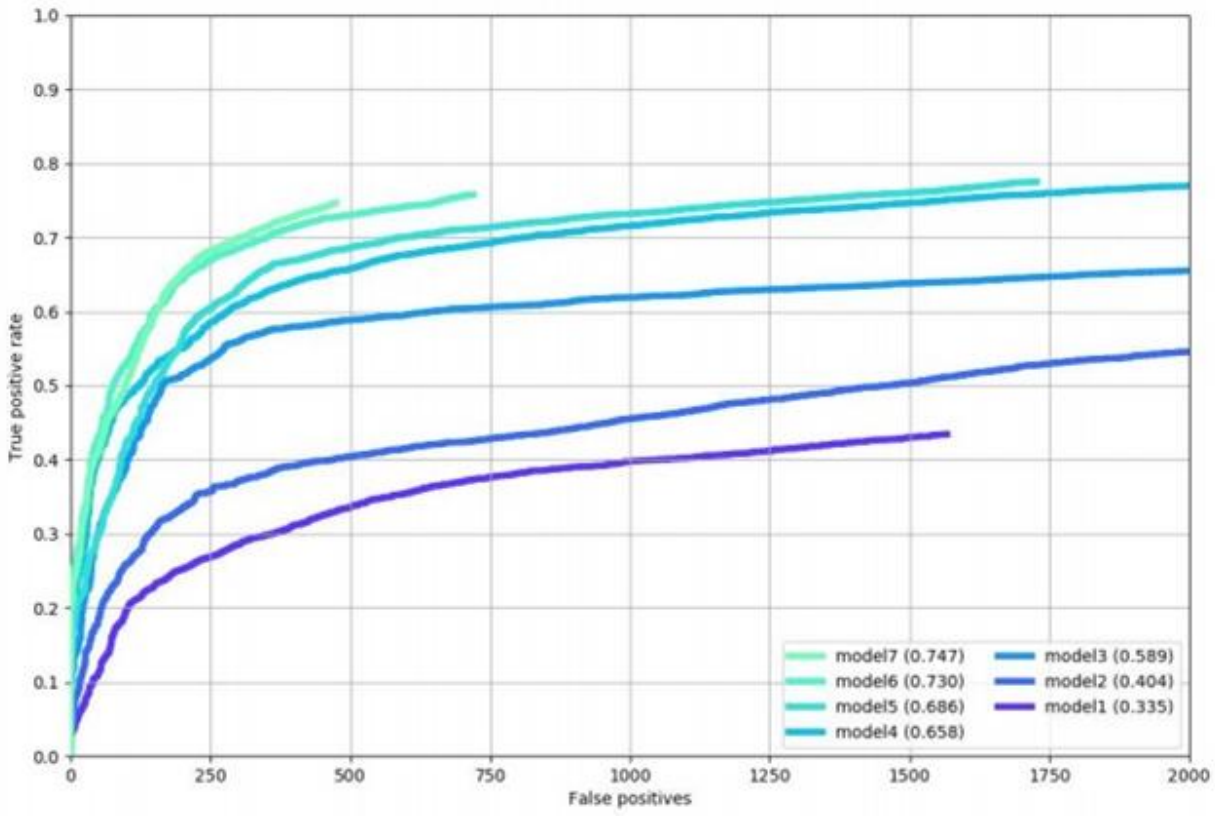
- **Mean Average Precision** is calculated over $n=1,\ldots, N$ levels of recall, by varying the confidence threshold:

$$mAP = 1/N \Sigma_n\ precision(t_n).$$

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **False Positive** (FP) vs **True positive** (TP) plots, as a function of detection threshold e.g., for various training stages.
- The closer the curve is to the upper left corner, the better.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Training

- All approaches treat **localization as a regression problem** to find [*H,W, X,Y*] using a CNN.
- All these CNNs utilize a mixed classification + localization loss of the form:

$$\mathcal{L}(a,\mathcal{I};\theta) = \beta_1 * \mathbf{1}[a \text{ is positive}] * \ell_{loc}(\phi(b_a;a), f_{loc}(\mathcal{I};a,\theta)) + \\ + \beta_2 * \ell_{cls}(y_a, f_{cls}(\mathcal{I};a,\theta))$$

β_1 and β_2 balance the localization and classification losses.

- α is the best matching ground truth ROI (anchor box) for the detected ROI (box) b_α.
- $\mathbf{1}[\alpha$ is positive]: indicator vector (vector of ones, if α matches b_α with good IoU).
- f_{loc} is the localization CNN function, f_{cls} is the classification CNN function.
- l_{loc}, l_{cls} are loss functions, e.g., MSE, cross-entropy.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

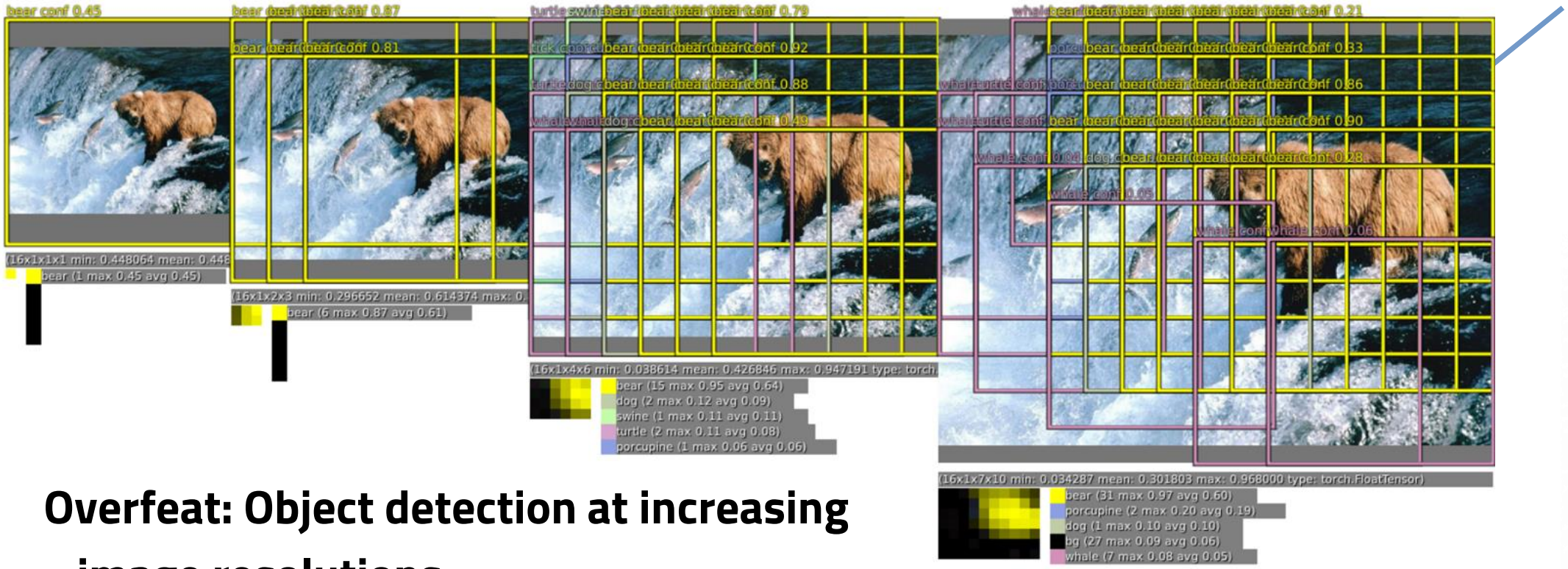
Object detection with CNNs

- **Deep Learning** (DL) approach: train a classifier on, say, 1000 classes of ILSVRC.

- **OverFeat** (2013) was one of the first DL approaches to object detection. Its convolutional method made multi-scale sliding window efficient.
- Based on AlexNet architecture.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection with CNNs



Overfeat: Object detection at increasing image resolutions

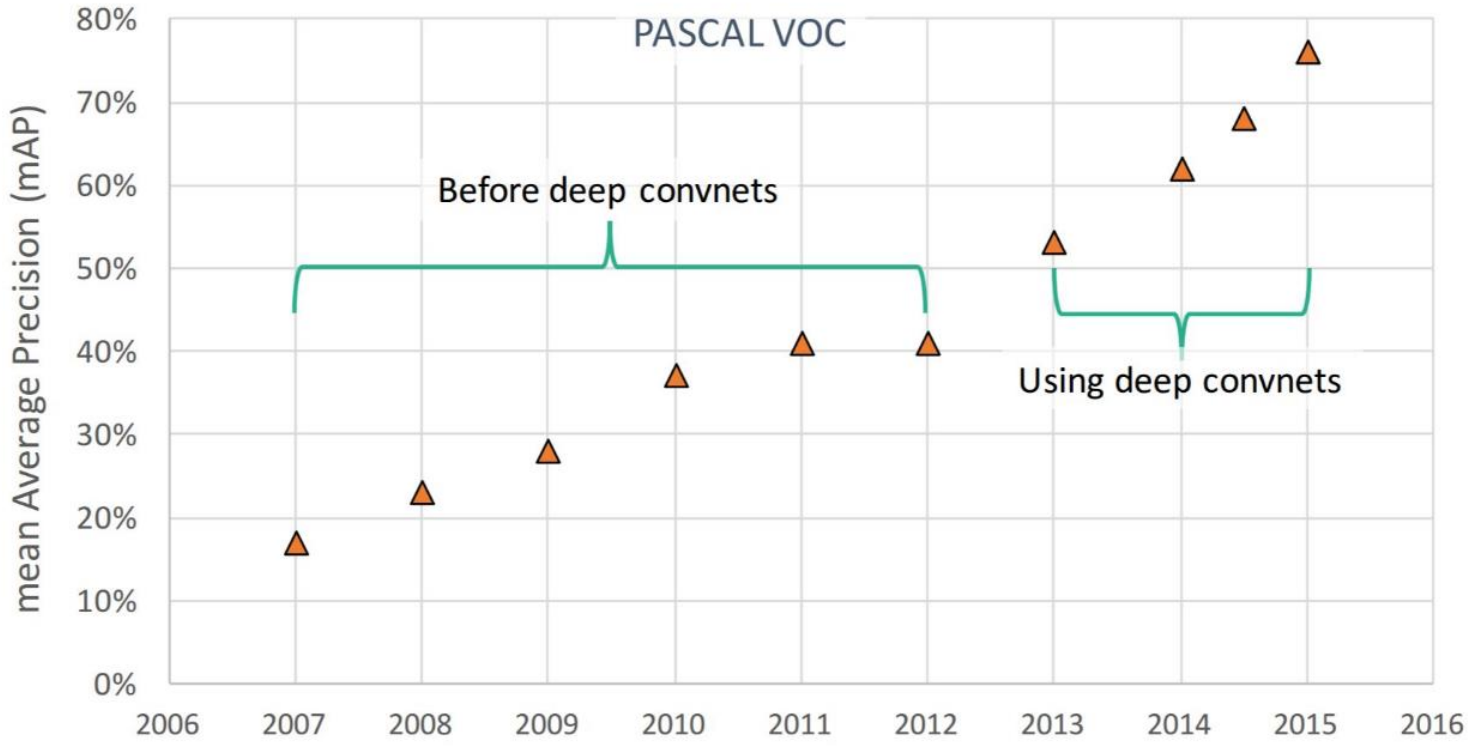
Sermanet, Pierre, et al. "Overfeat: Integrated recognition, localization and detection using convolutional networks." *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*. 2014.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection with CNNs

- Impact of Deep Learning.
- Pascal VOC

(object detection)



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

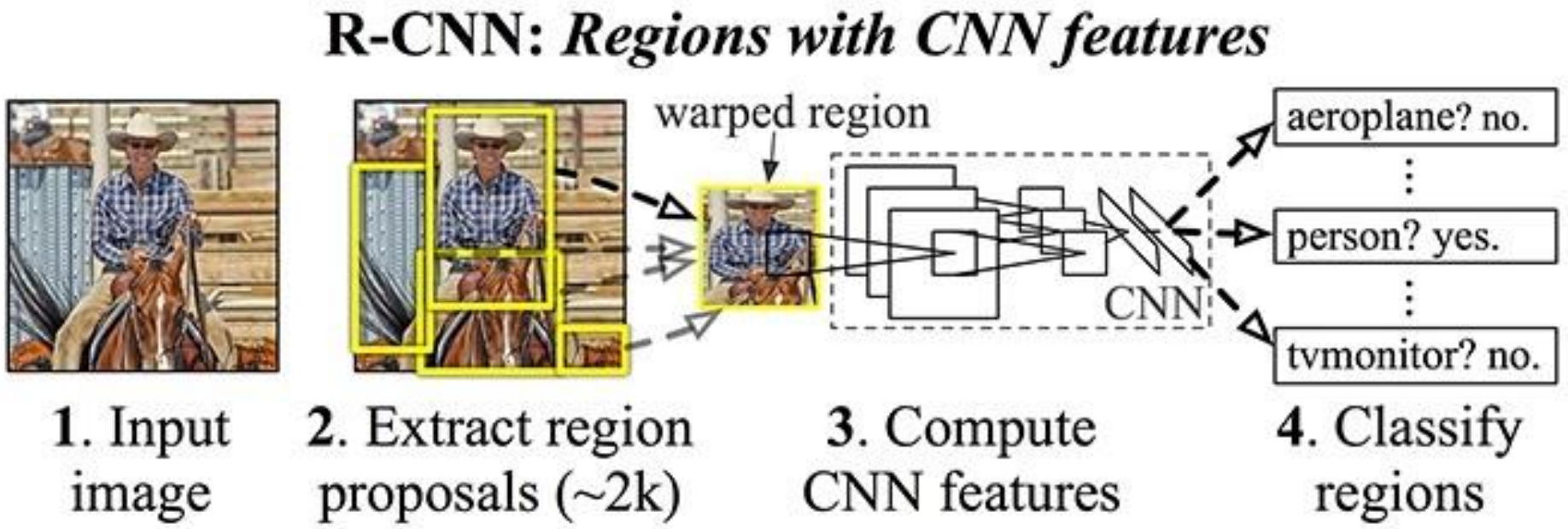
R-CNN

- R-CNN: Regions with CNN features.

- Three step approach:
  - Extract **region proposals** using an external proposal method (i.e., **Selective Search**). Cropped and resized proposed input image regions form **crops**, always having the same size.
  - Extract **CNN features for each crop.**
  - Classify features with an **SVM.**
  - **Regress** Region Of Interest (ROI) height (H) and width (W) based on the proposed and validated crops.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

R-CNN



Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

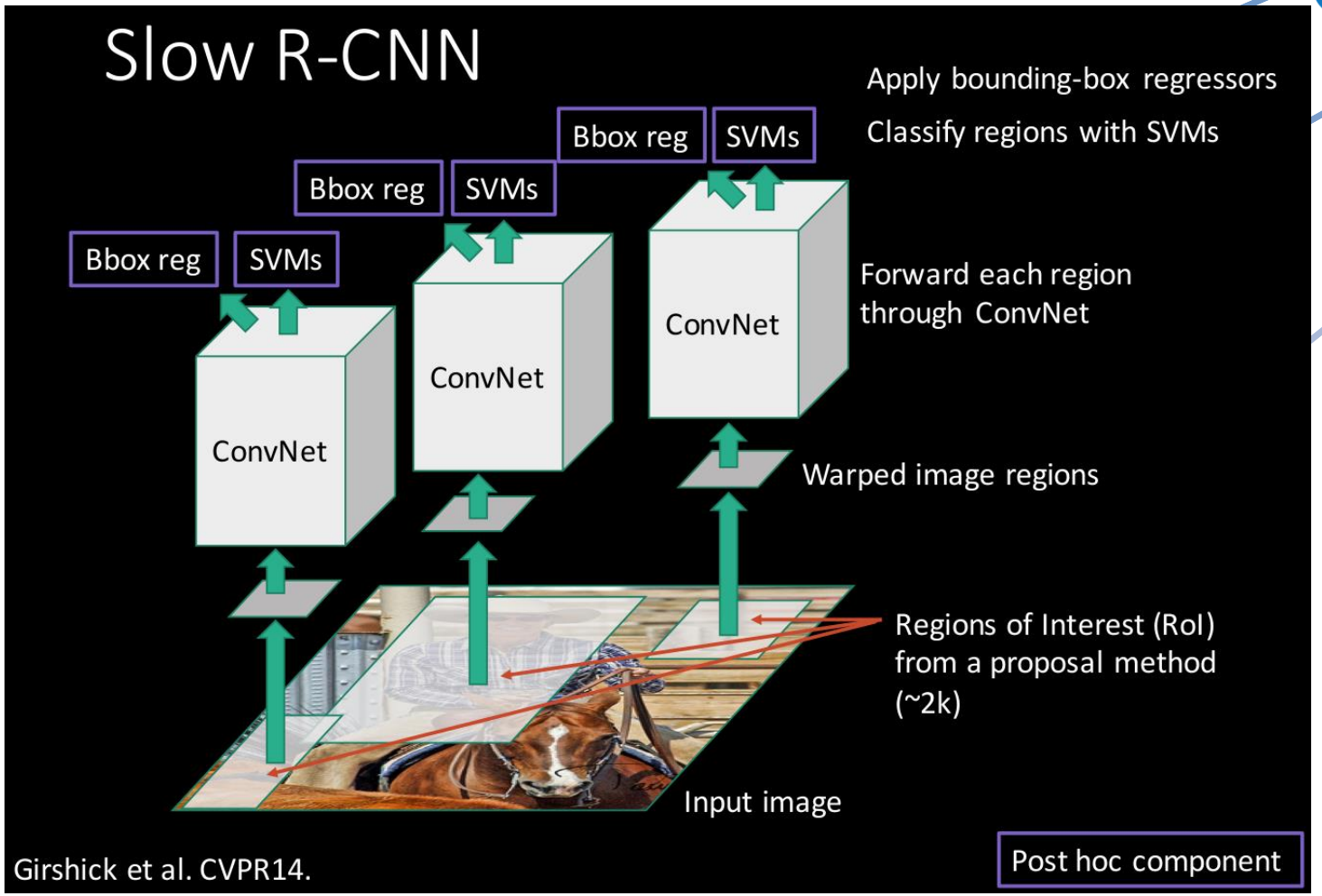
R-CNN

- **Region selection based on Selective Search**:

- Alternative to exhaustive, sliding-window search.
- Based on **region segmentation** techniques.
- Starting from **over-segmentation**, merge similar regions and produce region proposals.
- Merged regions are proposed.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

R-CNN



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Fast R-CNN

- R-CNN **weaknesses**:
  - Use of multiple **overlapping proposed regions** (crops) of the input image.
    - Too many **duplicate computations.**
  - Three stage architecture.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

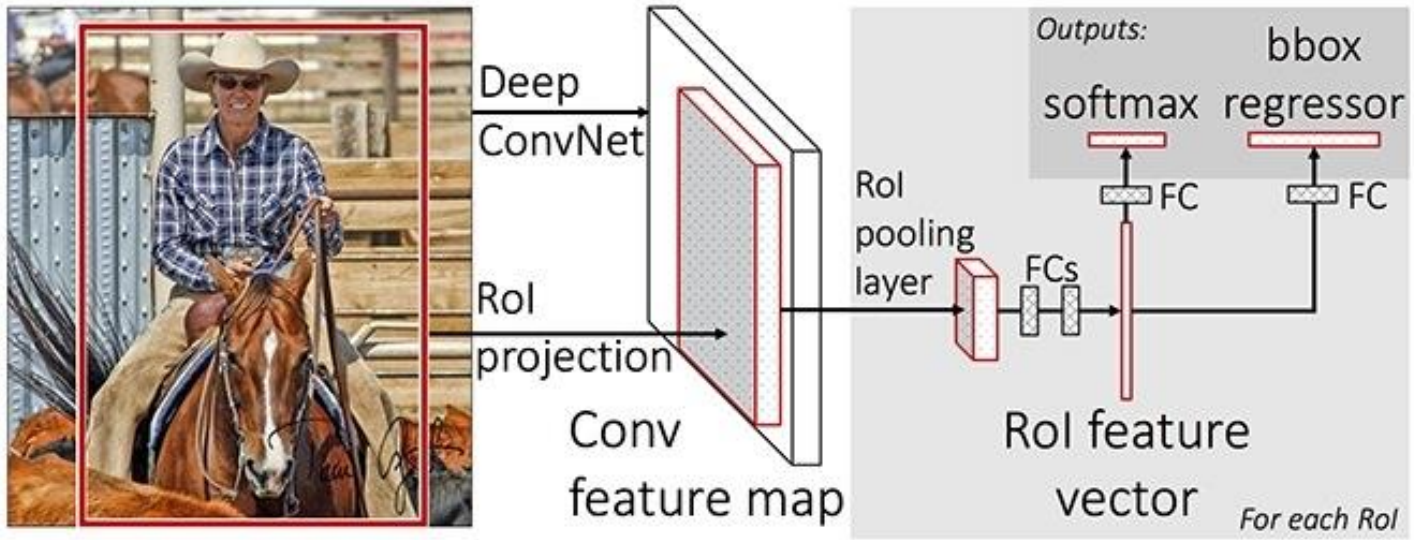
Fast R-CNN

- **Fast R-CNN** dealt with these weaknesses:
  - Input image is passed once from a CNN (ConvNet) to generate a **CNN feature map** (big speedup).
  - Selective search is used to generate region proposals, but crops were taken from the CNN feature map, instead of the input image (**ROI pooling**). Crops have always the same size.
  - The pooled features are then fed to the remainder of the network, consisting of **fully connected layers** for classification and ROI H, W refinement through regression.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Fast R-CNN

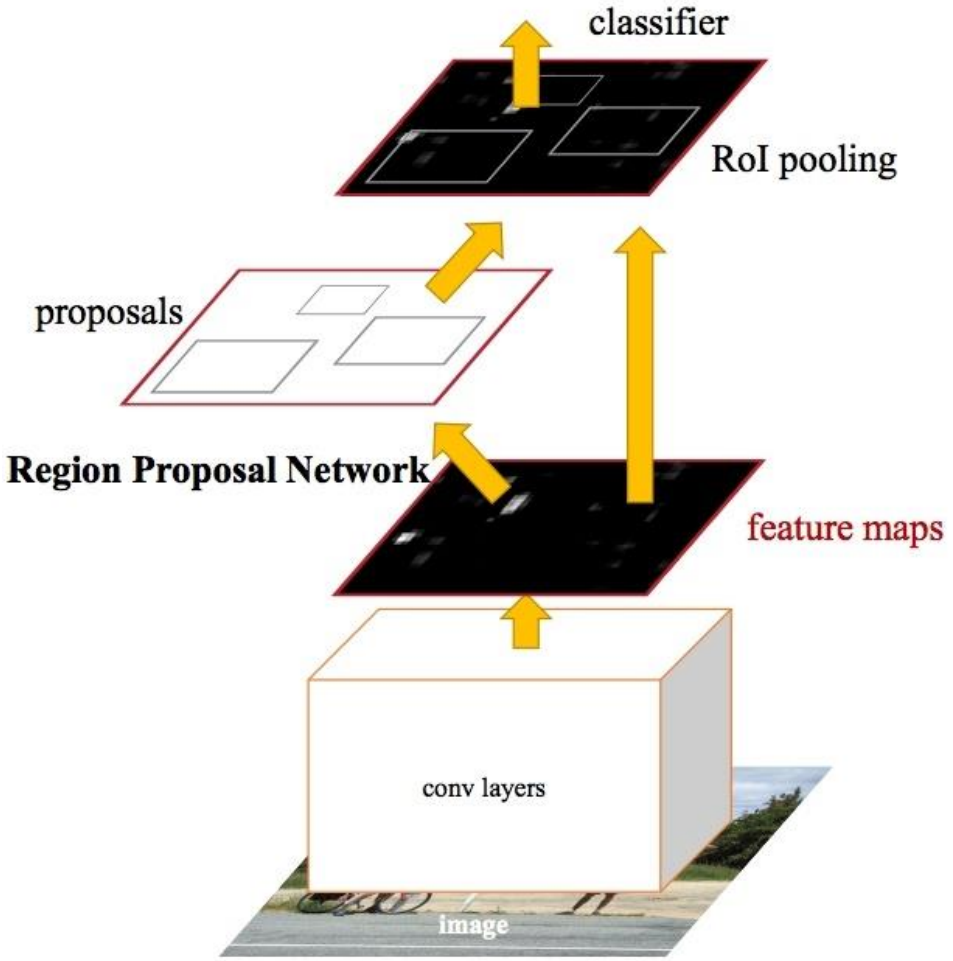
- Fast R-CNN **weaknesses**:
  - Multiple **overlapping RoIs**
    - duplicate computations.
  - **Externally** computed region proposals (selective search).



Girshick, Ross. "Fast R-CNN." *Computer Vision (ICCV), 2015 IEEE International Conference on*. IEEE, 2015.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Faster R-CNN



- Faster R-CNN: The **Region Proposal Network** shares layers with the feature extracting network and **internally produces region proposals** (no selective search).

Yuan, Peng, Yangxin Zhong, and Yang Yuan. "Faster R-CNN with Region Proposal Refinement."

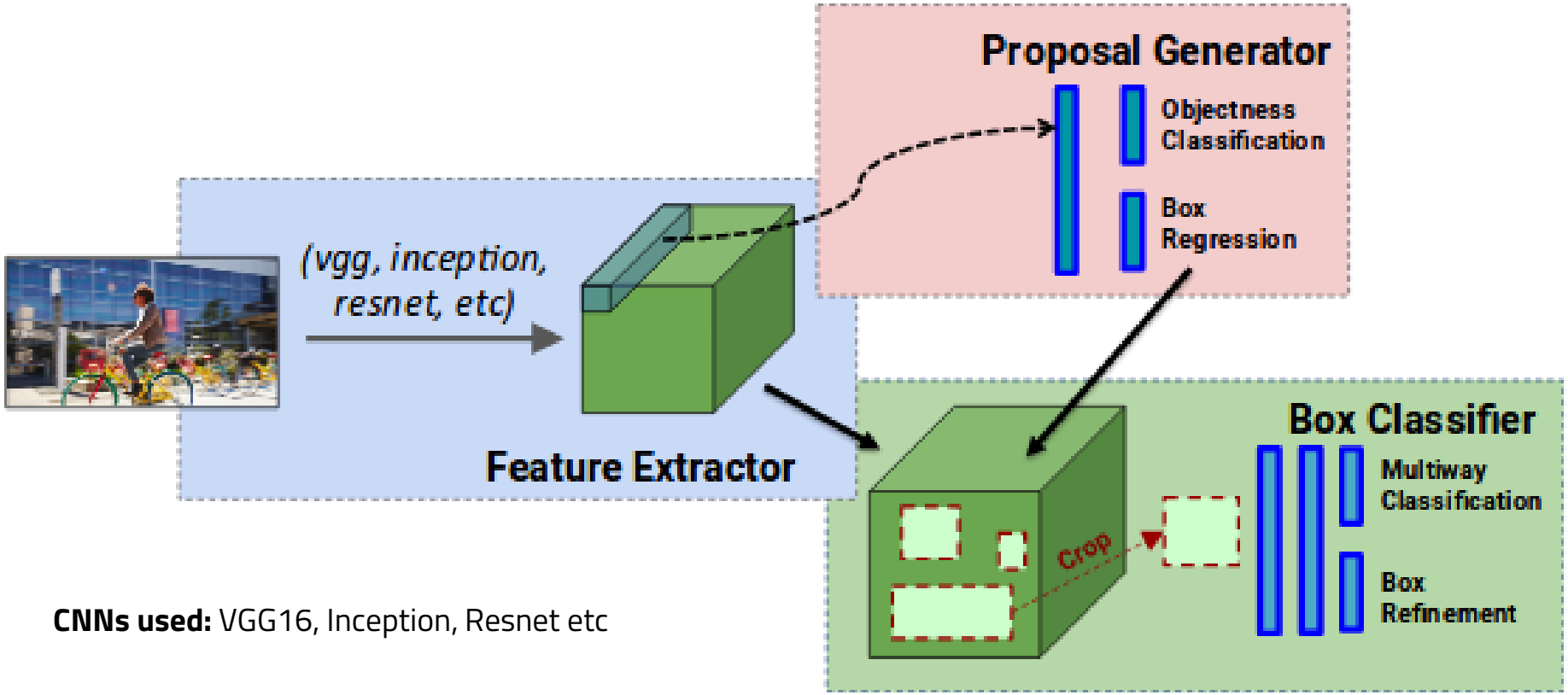
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Faster R-CNN

- The Region Proposal Network (RPN) produces proposals based on an *objectness* score, computed based on the **feature map activations.**
- The feature map is trained using ground-truth objects to produce high objectness score in their ROIs.
- The proposals are used by the RoI pooling and fed into the remainder of the (fully connected) layers for classification.
- Still, **part of the computation depends on the number of proposals**, making it quite **slow.**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Faster R-CNN



CNNs used: VGG16, Inception, Resnet etc

Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

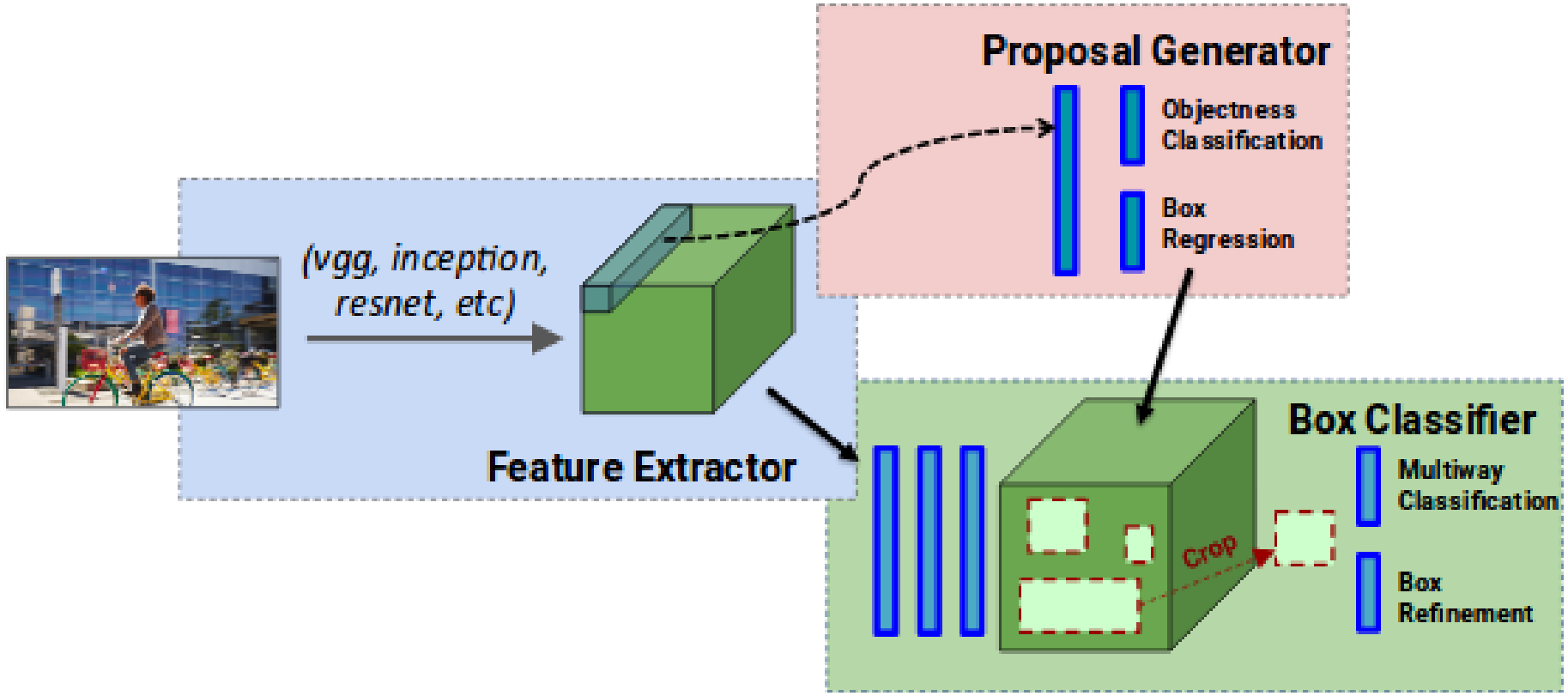
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

R-FCN

- **R-FCN**: Region-based Fully Convolutional Networks.
- Only convolutional layers, thus **fully convolutional.**
- Like **Faster R-CNN**, but crops features from the **last layer prior to prediction (region classification and refinement).**
- This **immensely decreases the per-region computation** that must be done.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

R-FCN



Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

SSD

- SSD: Single-Shot Detector.
- **Region-based object detection** (R-CNN, Fast R-CNN, Faster R-CNN, R-FCN): accurate, but **too slow for real-time** applications.
- SSD approach: **Combine a classification network and bounding box regression** into single architecture, without any external steps or duplicated computations.

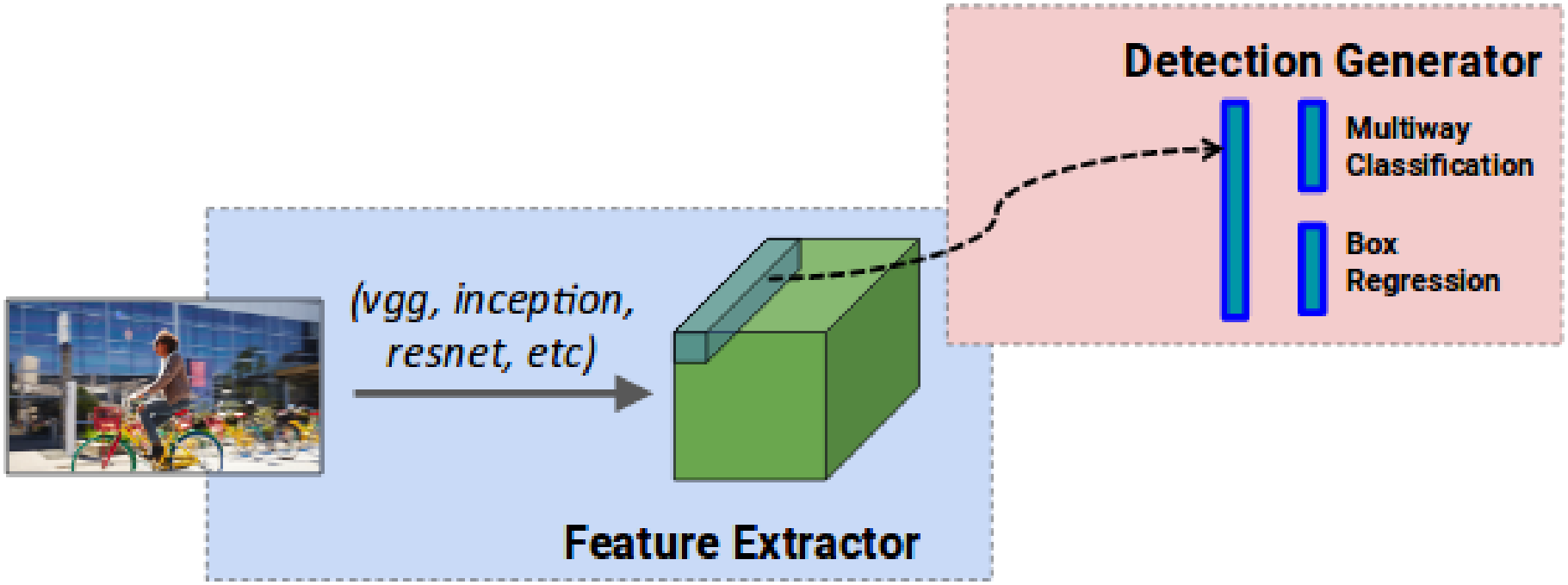
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

SSD

- It uses *anchors* (ROIs of precomputed size and aspect ratio). No region proposals are used.
- Anchors are overlapped at various spatial locations, aspect ratios and scales of the feature maps on various CNN layers.
- During training, anchor location and size are **refined via regression** to better fit objects.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

SSD

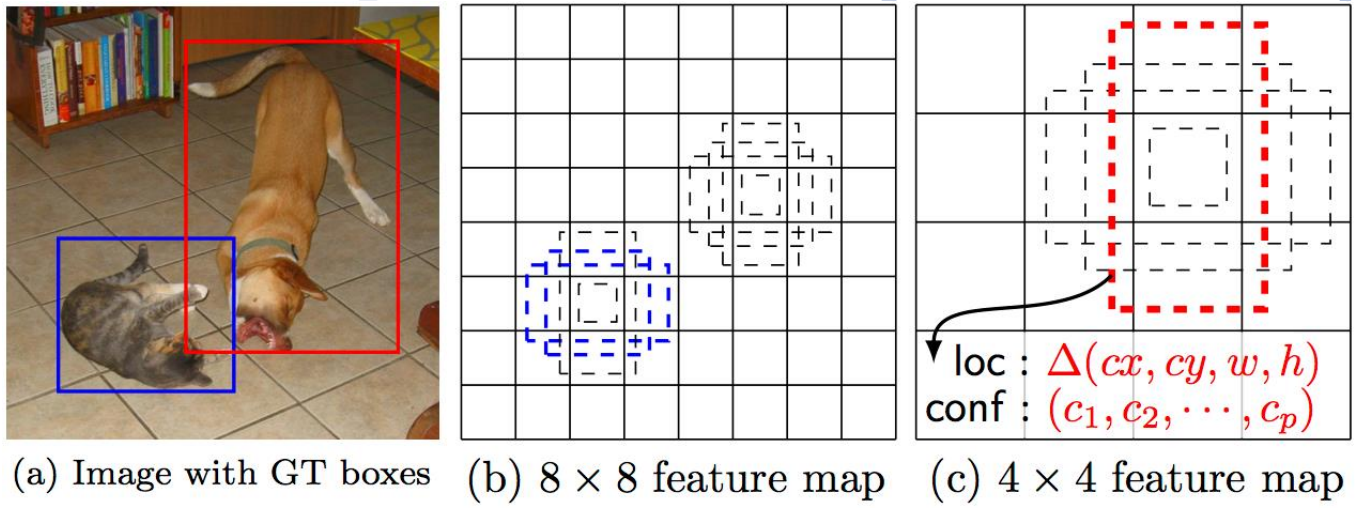


Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

SSD

- Example: The cat has 2 anchors (ROIs) that match on the 8x8 feature map, but none match the dog. We choose the one having biggest IoU and refine it.
- On the 4x4 feature map there is one anchor that matches the dog and is refined.



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Liu, Wei, et al. "Ssd: Single shot multibox detector." *European conference on computer vision*. Springer, Cham, 2016.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

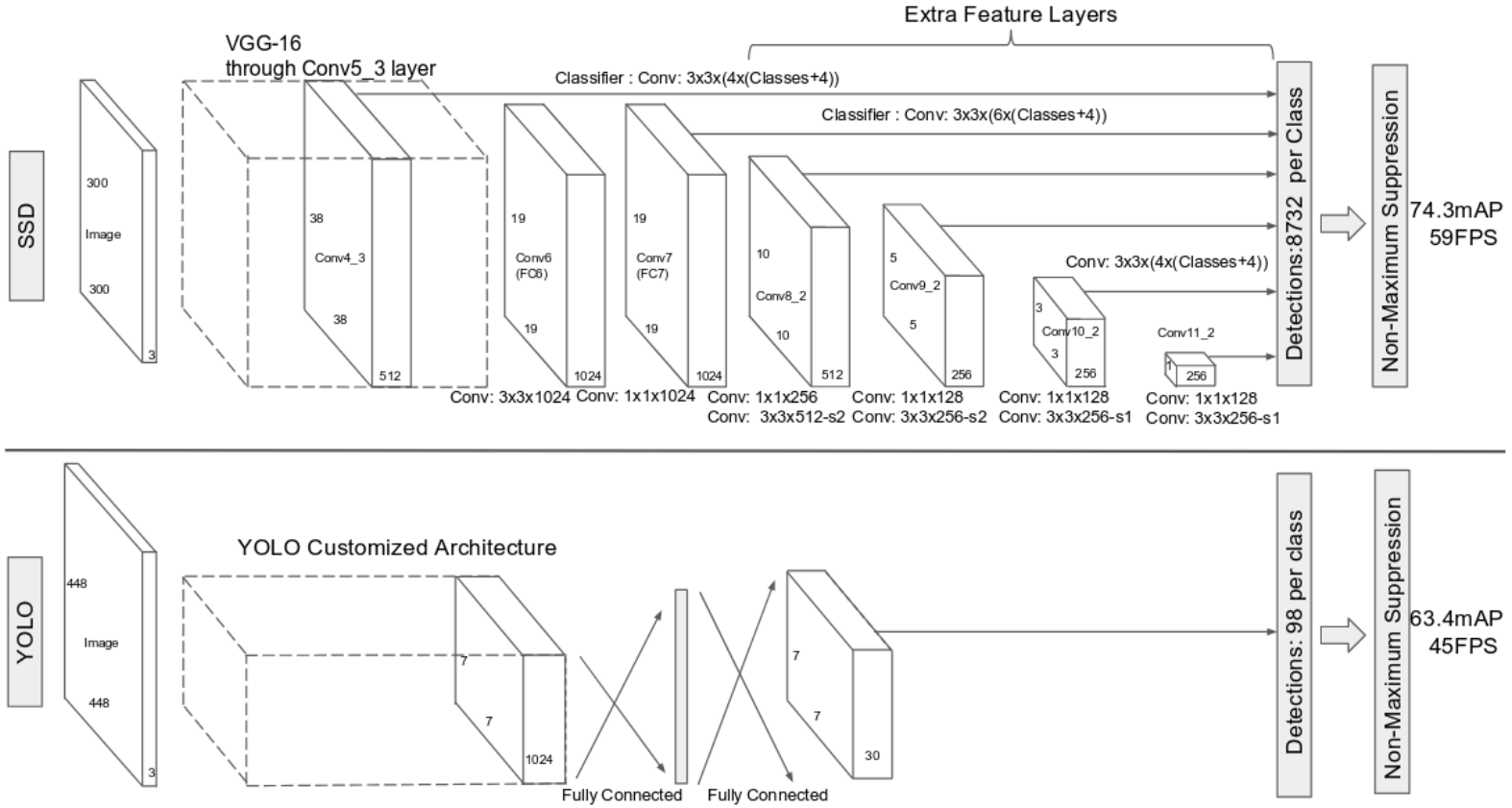
YOLO

- YOLO: You Only Look Once.
- The first version of YOLO was published before SSD, and **lacked in precision of localization.**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO

- **Simpler YOLO architecture:** Darkenet19 convolutional network plus FC layer.
- Prediction only at the final convolutional feature map.



Liu, Wei, et al. "Ssd: Single shot multibox detector." *European conference on computer vision*. Springer, Cham, 2016.

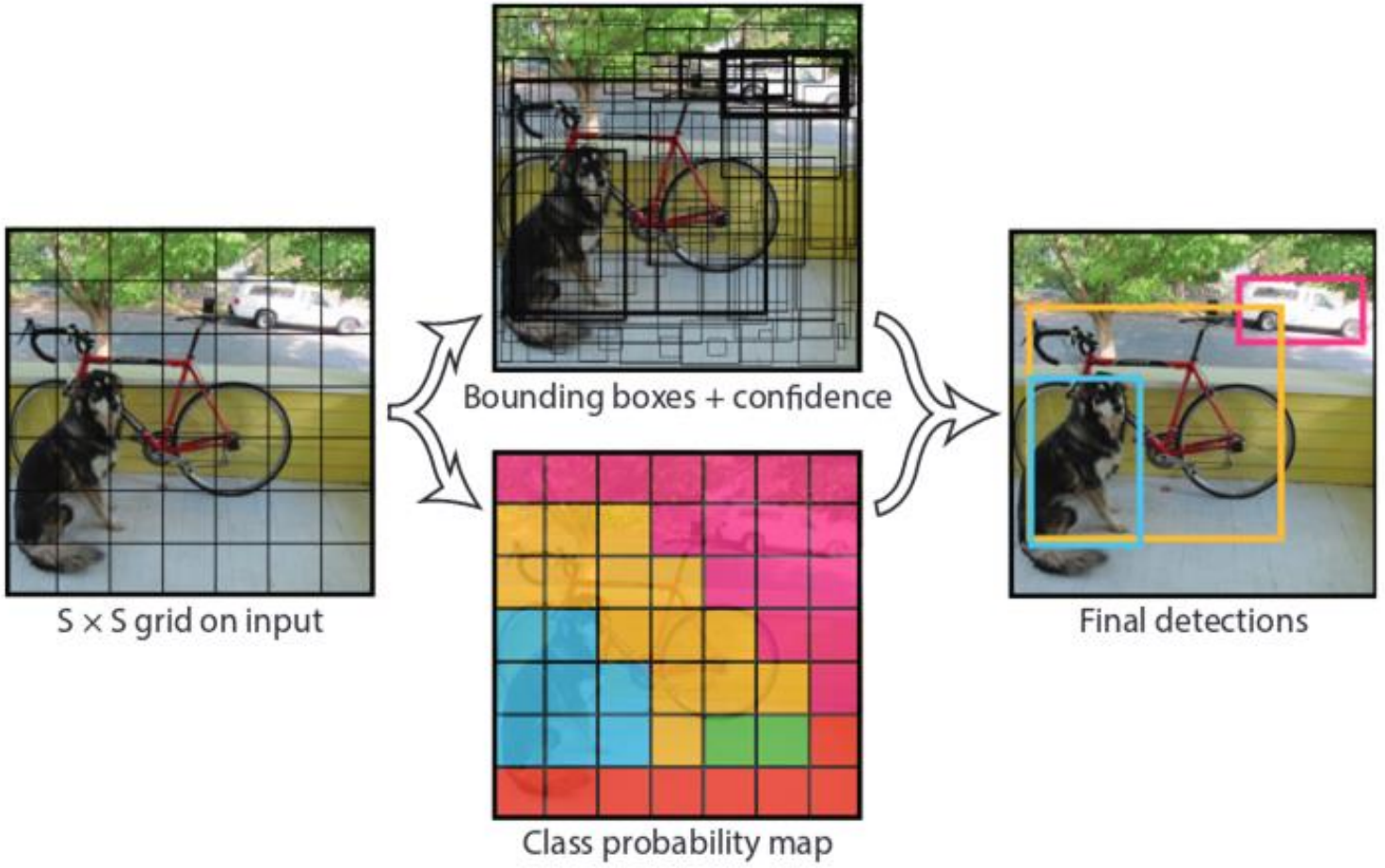
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO

- YOLO **divides** the input image into an *SxS* grid.
- If the **center** of an object falls within a cell of the grid, that cell is responsible for detecting that object.
- *N* is the maximal number of bounding boxes that each grid cell can detect.
- Each cell predicts *N* **bounding boxes** and confidence and classification scores for those boxes.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO



Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO

- **Confidence** is measured as *Pr(Object) * IoU(truth, pred)*, corresponding to a) how confident the model is that the box contains an object and b) how accurate it thinks the predicted box is.
- Each ROI is assigned **five object predicted values**: *H, W, X,Y* and *confidence.*

- The maximal number of detected objects is *NxSxS,* where *N* is the maximal number of detections per grid cell.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

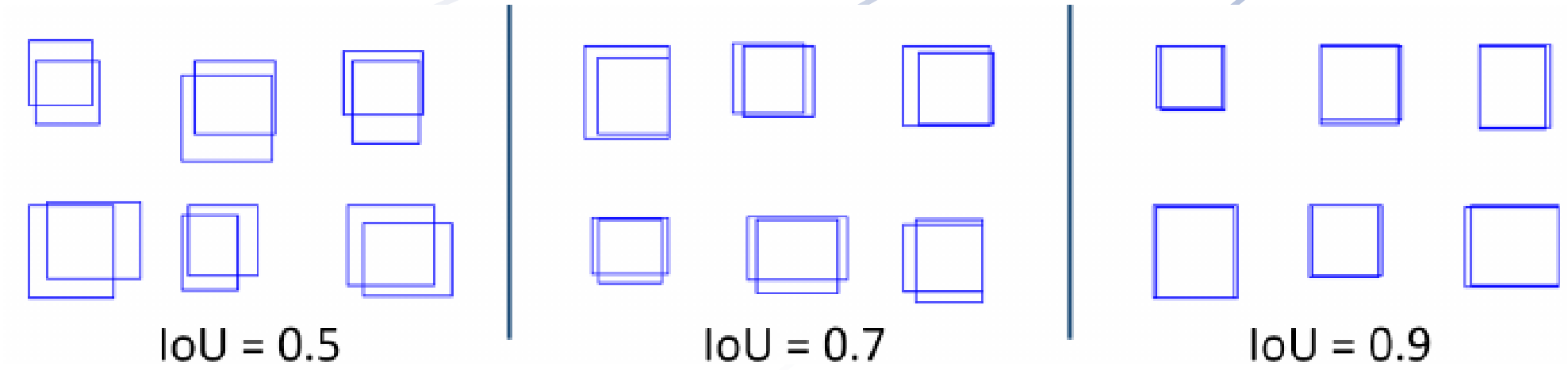
YOLO v2

- **Fully convolutional**, no densely-connected layers:
  - It may be run at varying input sizes.
- It can utilize **multi-scale capabilities** during training as well.
- **Very fast** architecture and implementation.
- Uses **precomputed anchors**.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO v2

- Anchors: precompute a predefined number of anchors (typically 5) by running **k-means** on all object ROIs of training set, using **IOU** as the error to minimize.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO v2

- **Input**: Image of arbitrary size HxW - best to be a multiple of 32, as the network downsamples by 32.
- **Output**: (H/32) x (W/32) x ((C+5)*N)
  - C is the number of classes to predict
  - N is the number of precomputed anchors to fit bounding boxes.
  - Depth d=(C+5)*N.
- For one class and five anchors (e.g., bicycles): (H/32) x (W/32) x 30.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

YOLO v2/ TinyYOLO

- **Input RGB image:** arbitrary size H, W.

Preferable sizes: odd multiples of 32. (416x416 =13x32x32x13 pixels).

- **Output feature map:** H/32xW/32xd=13x13x125 (d=125). Depth d (depends on the number of object classes and number of anchors used).

- **TinyYOLO** has same input/output but half the number of the convolutional layers.

Therefore, it is much faster.

layer		filters	size	input		output
0	conv	32	3 x 3 / 1	416 x 416 x 3	->	416 x 416 x 32
1	max		2 x 2 / 2	416 x 416 x 32	->	208 x 208 x 32
2	conv	64	3 x 3 / 1	208 x 208 x 32	->	208 x 208 x 64
3	max		2 x 2 / 2	208 x 208 x 64	->	104 x 104 x 64
4	conv	128	3 x 3 / 1	104 x 104 x 64	->	104 x 104 x 128
5	conv	64	1 x 1 / 1	104 x 104 x 128	->	104 x 104 x 64
6	conv	128	3 x 3 / 1	104 x 104 x 64	->	104 x 104 x 128
7	max		2 x 2 / 2	104 x 104 x 128	->	52 x 52 x 128
8	conv	256	3 x 3 / 1	52 x 52 x 128	->	52 x 52 x 256
9	conv	128	1 x 1 / 1	52 x 52 x 256	->	52 x 52 x 128
10	conv	256	3 x 3 / 1	52 x 52 x 128	->	52 x 52 x 256
11	max		2 x 2 / 2	52 x 52 x 256	->	26 x 26 x 256
12	conv	512	3 x 3 / 1	26 x 26 x 256	->	26 x 26 x 512
13	conv	256	1 x 1 / 1	26 x 26 x 512	->	26 x 26 x 256
14	conv	512	3 x 3 / 1	26 x 26 x 256	->	26 x 26 x 512
15	conv	256	1 x 1 / 1	26 x 26 x 512	->	26 x 26 x 256
16	conv	512	3 x 3 / 1	26 x 26 x 256	->	26 x 26 x 512
17	max		2 x 2 / 2	26 x 26 x 512	->	13 x 13 x 512
18	conv	1024	3 x 3 / 1	13 x 13 x 512	->	13 x 13 x1024
19	conv	512	1 x 1 / 1	13 x 13 x1024	->	13 x 13 x 512
20	conv	1024	3 x 3 / 1	13 x 13 x 512	->	13 x 13 x1024
21	conv	512	1 x 1 / 1	13 x 13 x1024	->	13 x 13 x 512
22	conv	1024	3 x 3 / 1	13 x 13 x 512	->	13 x 13 x1024
23	conv	1024	3 x 3 / 1	13 x 13 x1024	->	13 x 13 x1024
24	conv	1024	3 x 3 / 1	13 x 13 x1024	->	13 x 13 x1024
25	route	16				
26	conv	64	1 x 1 / 1	26 x 26 x 512	->	26 x 26 x 64
27	reorg		/ 2	26 x 26 x 64	->	13 x 13 x 256
28	route	27 24				
29	conv	1024	3 x 3 / 1	13 x 13 x1280	->	13 x 13 x1024
30	conv	125	1 x 1 / 1	13 x 13 x1024	->	13 x 13 x 125
31	detection					

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

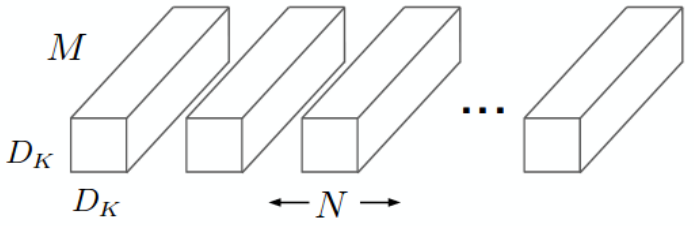
Network parameters

- **Any feature extractor** can be used for the feature extraction step of detection:
  - VGG16
  - ResNet, ResNet-101
  - Inception, Inception V2, V3
  - MobileNets.
- **MobileNets** were recently introduced by Tensorflow as a lightweight alternative:
  - The standard convolution is replaced by depth-wise separable convolutions, reducing the number of parameters and FLOPs.

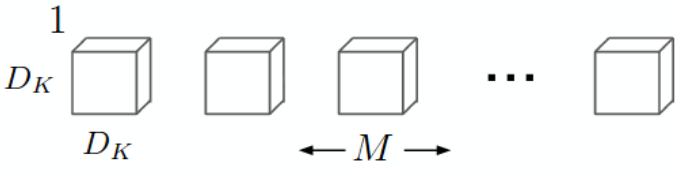
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Mobilenets

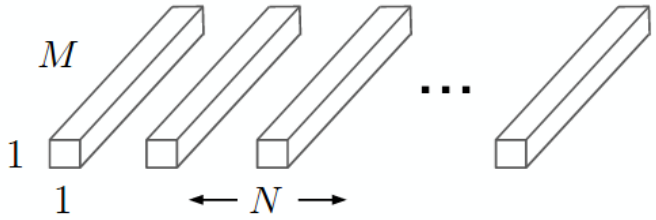
- **Standard convolutional filters:**
- N D_kxD_k filters, depth M (M=3 for RGB images).
- **Depth-wise separable convolutional filters (Mobilenet):**
- M D_kxD_k filters of depth 1, one per input channel (M=3 for RGB images) **and**
- N 1x1 filters (essentially weighted averaging of the M channels produced by the previous step).



(a) Standard Convolution Filters



(b) Depthwise Convolutional Filters

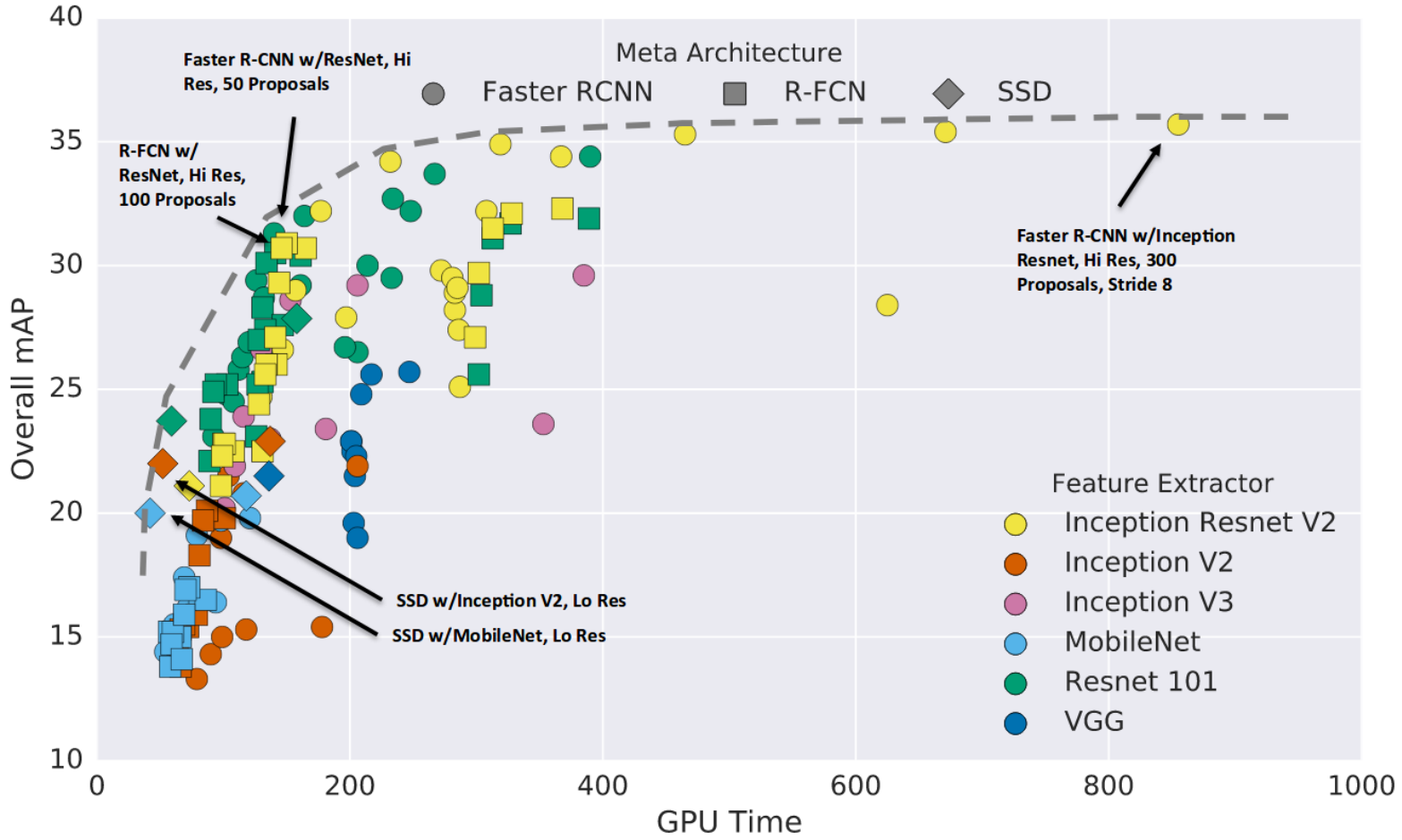


(c) 1×1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

Howard, Andrew G., et al. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications."

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

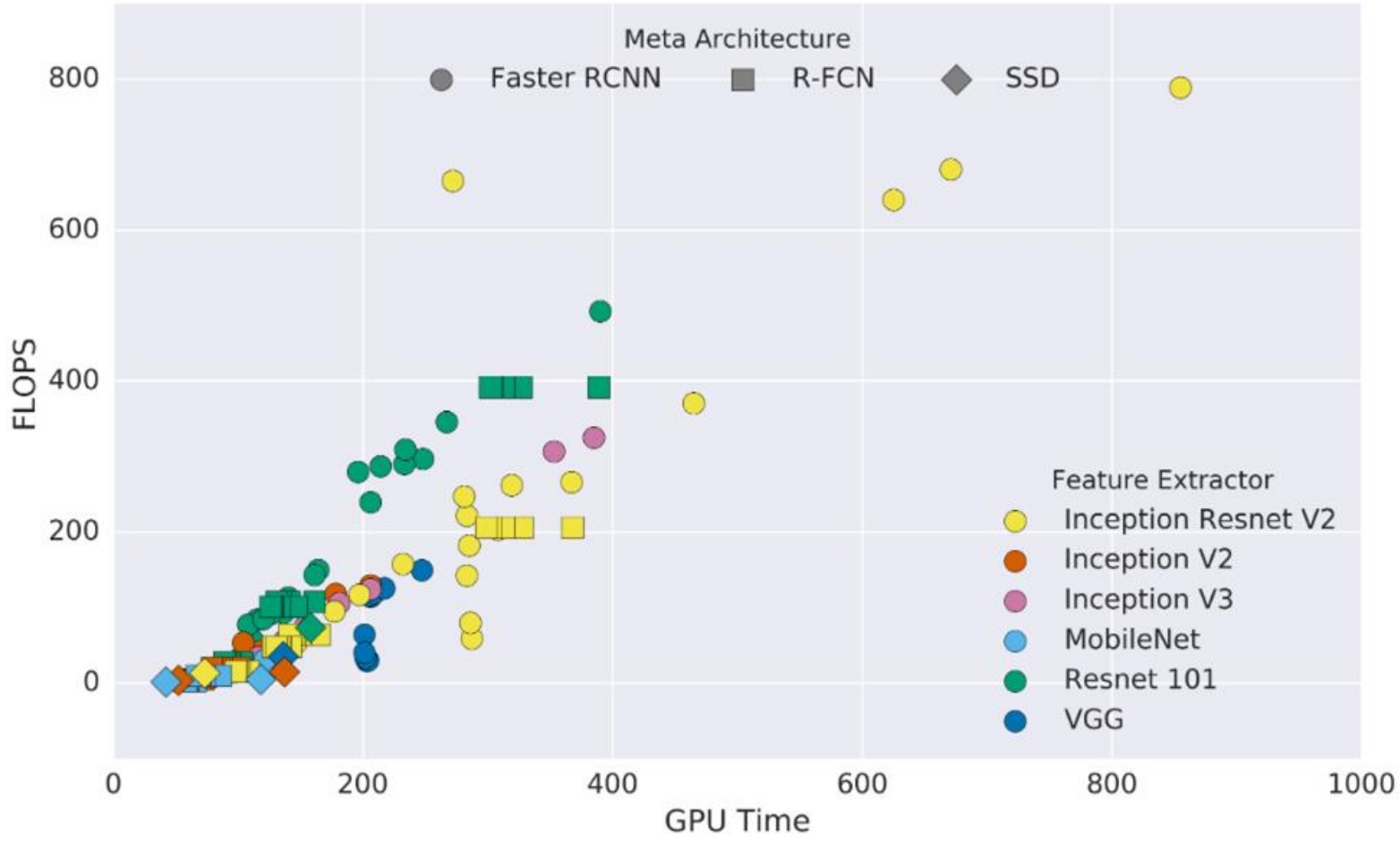
CNN comparison



Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

CNN comparison



Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

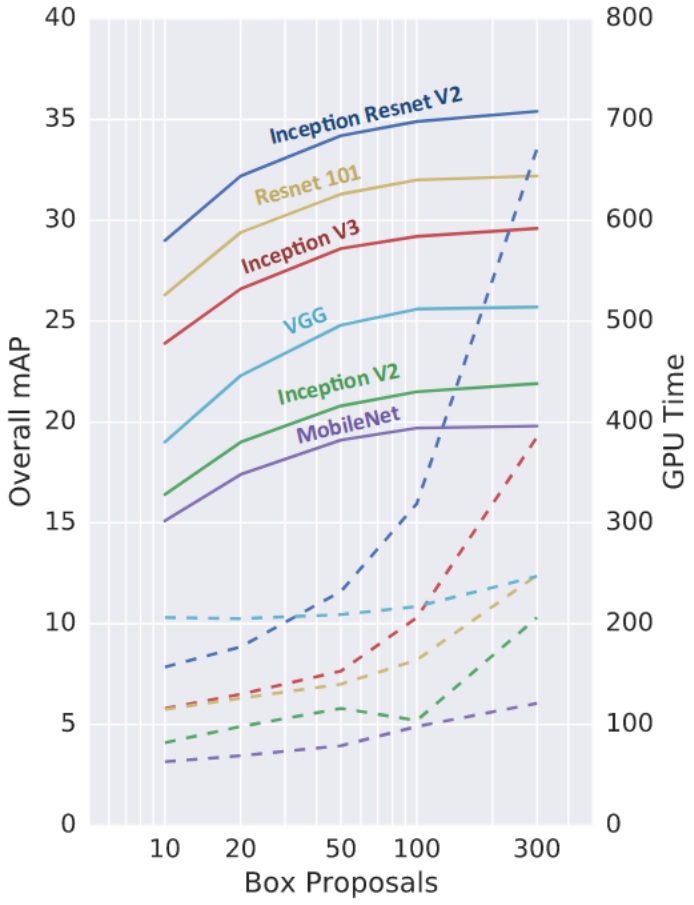
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Input image resolution NxN

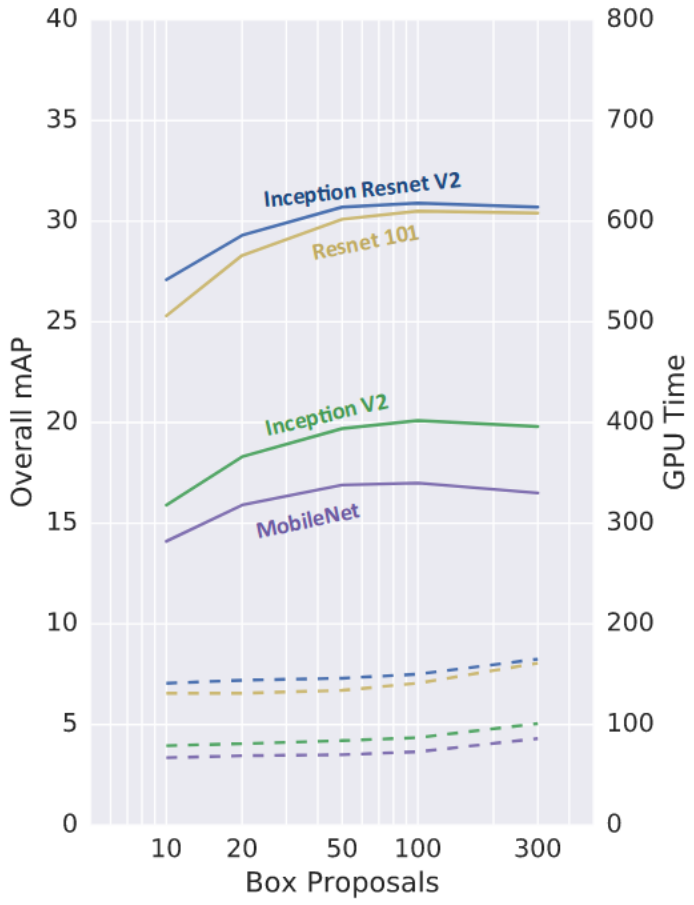
Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Number of Region Proposals



(a) FRCNN



(b) RFCN

Huang, Jonathan, et al. "Speed/accuracy trade-offs for modern convolutional object detectors." *IEEE CVPR*. 2017.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

CNN comparison

- **Faster R-CNN is more accurate but slower.**
- **YOLO, SSD are much faster** but not as accurate.
- YOLO, SSD make **more mistakes when objects are small** and have trouble correctly predicting the exact location of such objects.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection acceleration

- Examples of acceleration techniques:
  - Input size reduction.
  - Specific object detection instead of multi-object detection.
  - Parameter reduction.
  - Post-training optimizations with TensorRT (NVIDIA), including FP16 (floating point 16 bit) computations.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection on NVIDIA jetson TX2

- YOLO: good precision in general, but too heavyweight:
  - small objects are more challenging.
- Evaluation on VOC:

Input Image Size	FPS	mAP	Forward time (ms) No TensorRT	Forward time (ms) TensorRT	Forward time (ms) FP16
608x608	2.9	71.26	241.5	128.8	69.3
544x544	3.2	73.64	214.4	121.2	64.3
480x480	5.4	74.50	155.4	62.3	35.7
416x416	6.4	73.38	155.3	56.5	32.5
352x352	7.8	71.33	111.0	45.0	24.3
320x320	8.5	70.02	103.0	40.4	22.8

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection on NVIDIA jetson TX2

MultiDrone

- Tiny YOLO: low precision, but very lightweight.
- Evaluation on VOC:

Input Image Size	FPS	mAP	Forward time (ms) No TensorRT	Forward time (ms) TensorRT	Forward time (ms) FP16
608x608	6.5	51.28	76.5	37.5	22.1
544x544	8.2	52.93	68.4	34.8	20.5
480x480	13.4	55.00	50.1	17.2	11.7
416x416	16.5	56.28	49.9	15.7	10.3
352x352	20	55.05	37.1	13.0	7.9
320x320	23	53.81	34.0	11.7	7.2

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection on NVIDIA jetson TX2

- SSD: generally good precision, not as fast, still prone to mistakes when objects are small.
- MobileNets and Inception V2 can be used as feature extractors to provide speed ups.
- Tensorflow Implementation (subject to Tensorflow's memory mishandling).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection on NVIDIA jetson TX2

MultiDrone

MobileNets

Input Image Size	FPS	Recall
300x300	11.6	84.1
224x224	13.4	81.1
192x192	18.2	80.0
160x160	21.0	77.4
128x128	23.8	71.4

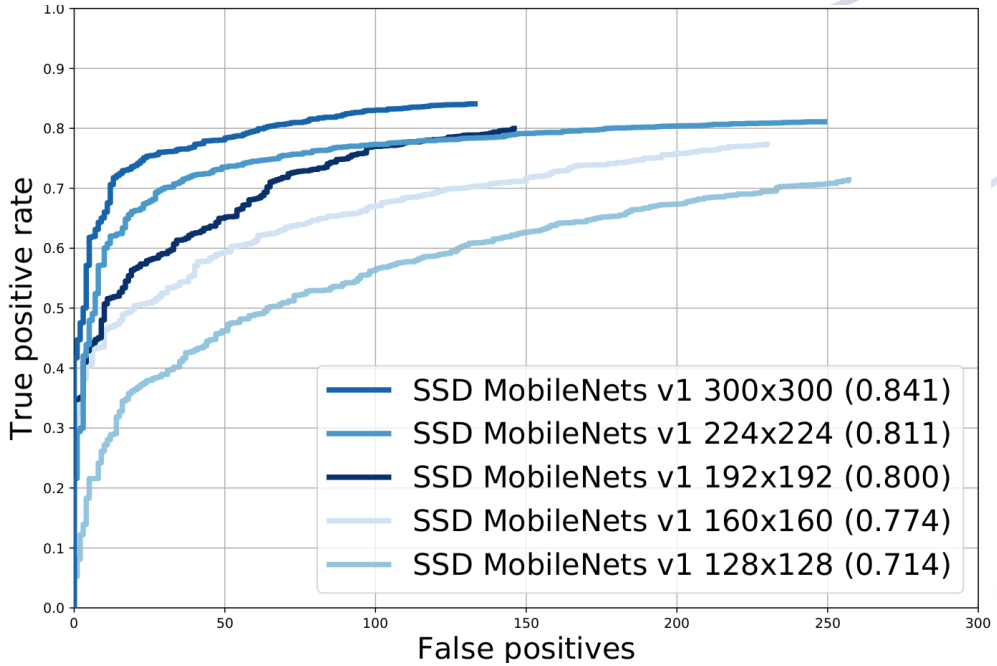
Inception V2

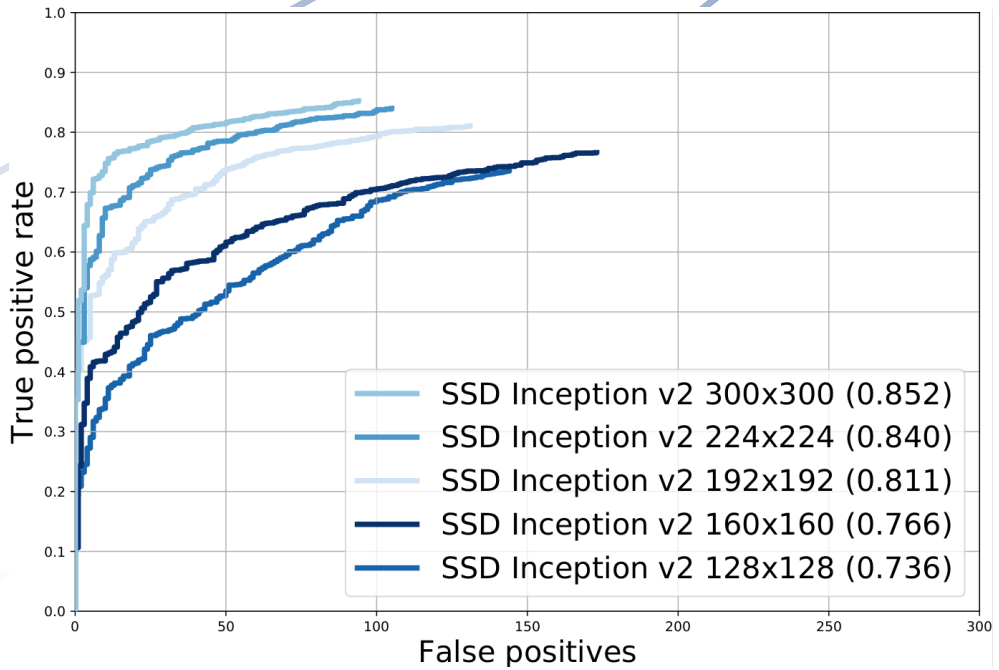
Input Image Size	FPS	Recall
300x300	8.5	85.2
224x224	12.3	84.0
192x192	13.8	81.1
160x160	15.6	76.6
128x128	17.1	73.6

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection comparisons

- SSD w/ MobileNets and Inception V2 for various input image sizes in Face Detection in FDDB facial data base. Curves are created by changing the confidence threshold.
- (Face recall in parentheses).





This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- Single view object detection
  - Deep learning (CNN) object detection.
  - **Light weight CNNs for object detection.**
- Multiple view object detection.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **State-of-the-art** object detectors (YOLO, SSD, etc) are based on **very Deep** and **multiple-channel CNNs.**
- **Light weight** architectures can provide equally satisfactory results.
- Such architectures are trained with incremental positive and negative example mining methods.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **Light weight** deep CNN architecture (~76K parameters).
- The network is **fully convolutional.**
- Input 32x32 pixel RGB image. Output d=2 (2 classes).

layer	kernel	filters	input	output	parameters
conv1	3×3	24	$32 \times 32 \times 3$	$30 \times 30 \times 24$	648
prelu1			$30 \times 30 \times 24$	$30 \times 30 \times 24$	24
conv2	4×4	24	$30 \times 30 \times 24$	$14 \times 14 \times 24$	9216
prelu2			$14 \times 14 \times 24$	$14 \times 14 \times 24$	24
conv3	4×4	32	$14 \times 14 \times 24$	$11 \times 11 \times 32$	12288
prelu3			$11 \times 11 \times 32$	$11 \times 11 \times 32$	32
conv4	4×4	48	$11 \times 11 \times 32$	$8 \times 8 \times 48$	24576
prelu4			$8 \times 8 \times 48$	$8 \times 8 \times 48$	48
conv5	4×4	32	$8 \times 8 \times 48$	$5 \times 5 \times 32$	24576
prelu5			$5 \times 5 \times 32$	$5 \times 5 \times 32$	32
conv6	3×3	16	$5 \times 5 \times 32$	$3 \times 3 \times 16$	4608
prelu6			$3 \times 3 \times 16$	$3 \times 3 \times 16$	16
conv7	3×3	2	$3 \times 3 \times 16$	$1 \times 1 \times 2$	288

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- Detection with **Light weight** deep CNNs.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- Detection with **Light weight** deep CNNs.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

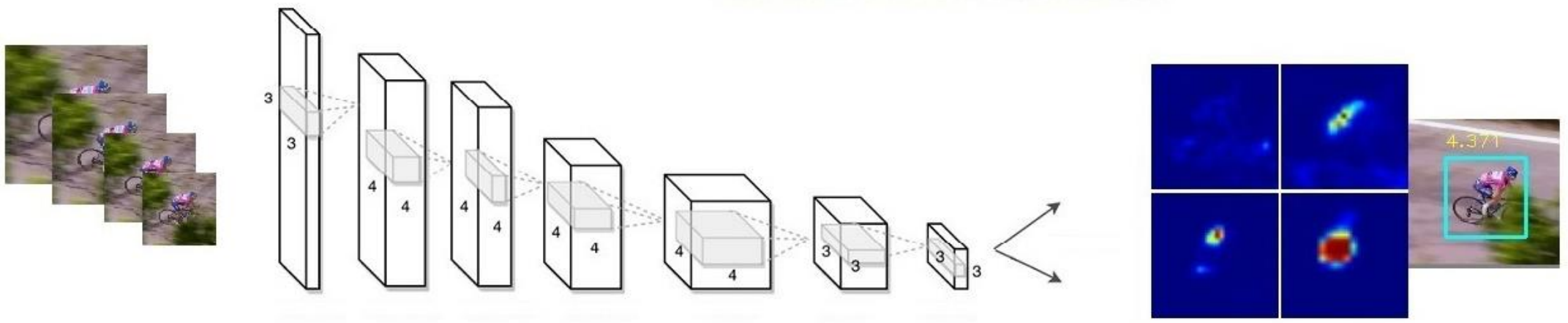
Object detection

- Detection with **Light weight fully** CNNs.
- The network is trained with **32x32** pixel training samples.
- It is **fully convolutional** and accepts images of **arbitrary** size.
- The network outputs a classification heatmap containing probability scores for each **32x32** pixel region of the input.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

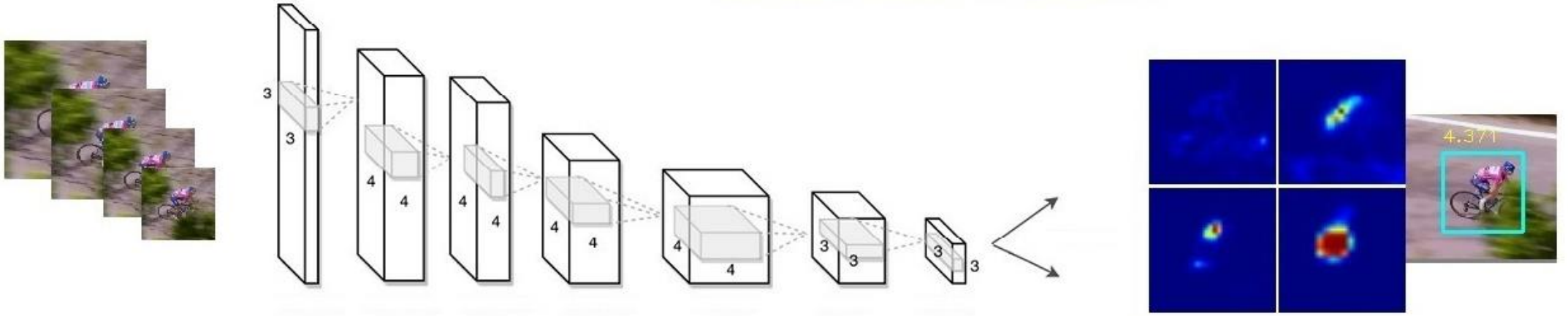
- **Detection with Light weight deep CNNs.**
- The method develops an image pyramid representation of varying resolutions and performs detection at multiple scales.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

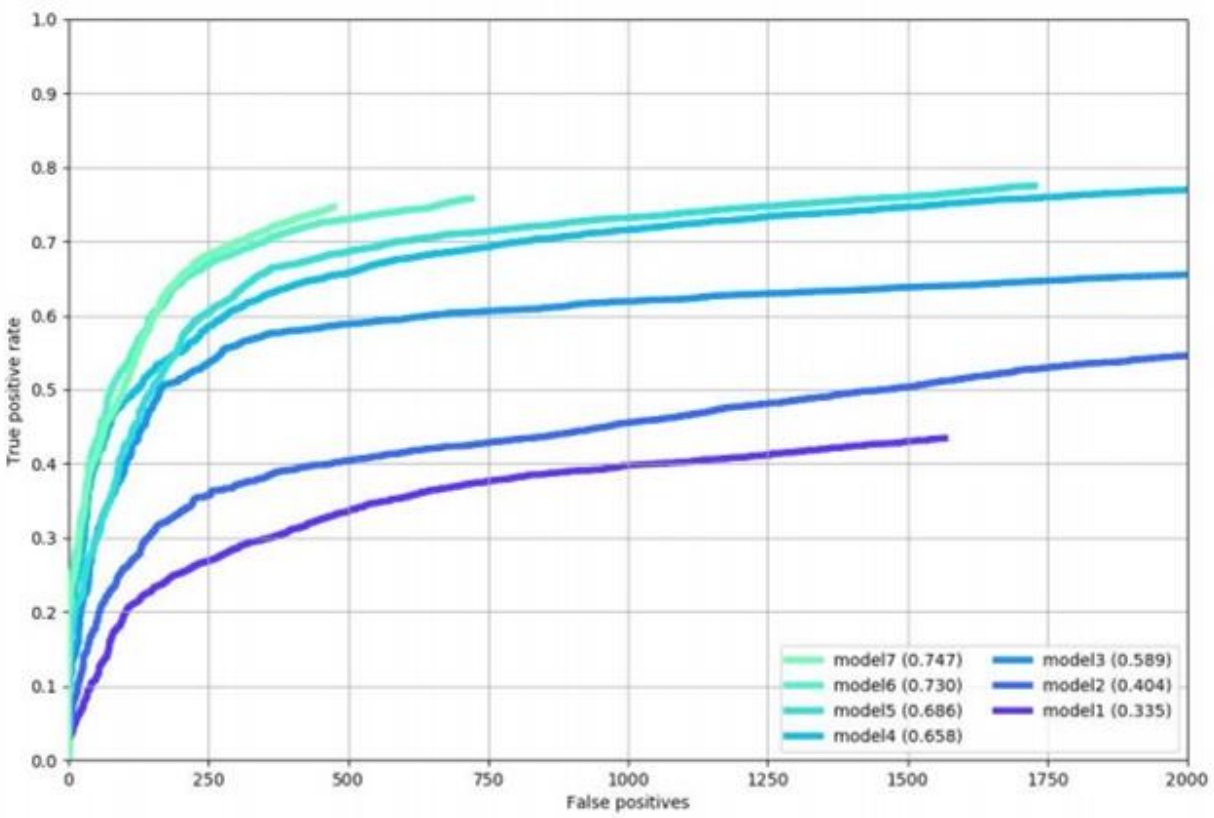
- Output heat maps are thresholded at the various pyramid layers.
- The image ROIs of size 32x32xl are created (l: pyramid level).
- The ROI parameters [*X,Y,H,W*] are clustered to create the final object ROIs.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- Evaluation of the model on a bicycle benchmark of RAI videos.
- True positive rate and False positive rate are evaluated for:
- Different heatmap thresholds,
- **Hard negative/positive mining**: increased training examples.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Object detection

- **Test execution time** for a 32x32 pixel object ROI of the proposed CNN architecture using NVIDIA's tensorRT library (in msec):

NVIDIA TX2		geForce GTX 1080
tensorRT	no tensorRT	
0.491933	2.84615	0.652406

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking

- **Tracker node**: Given the initialized position of a target, the tracker T is responsible for estimating the bounding box of the target in the subsequent frames.
- **Detector:** Given a bounding box defining the target in specific frame produced by the tracker, the detector D is responsible for verifying this result, and then provide the appropriate feedback to the system.
- **Master Visual Analysis:** If the verification from D fails, Master Visual Analysis is responsible for the re-initialization of the tracker T with the corrected bounding box.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking

Algorithm 1: Joint Detection & Tracking Framework

```
1 Start Detection Node D
2 Start Tracking Node T
3 Start Master Visual Analysis MVA
4 Run D, T and MVA in parallel
```

Algorithm 2: Detection Node *D*

```
1 Initialize detection
2 if time for verification then
3     Verify the tracking result;
4     if verification failed then
5         Calculate ROI;
6         Perform detection to ROI;
7         Publish corrected message d;
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking

Algorithm 3: Tracking Node *T*

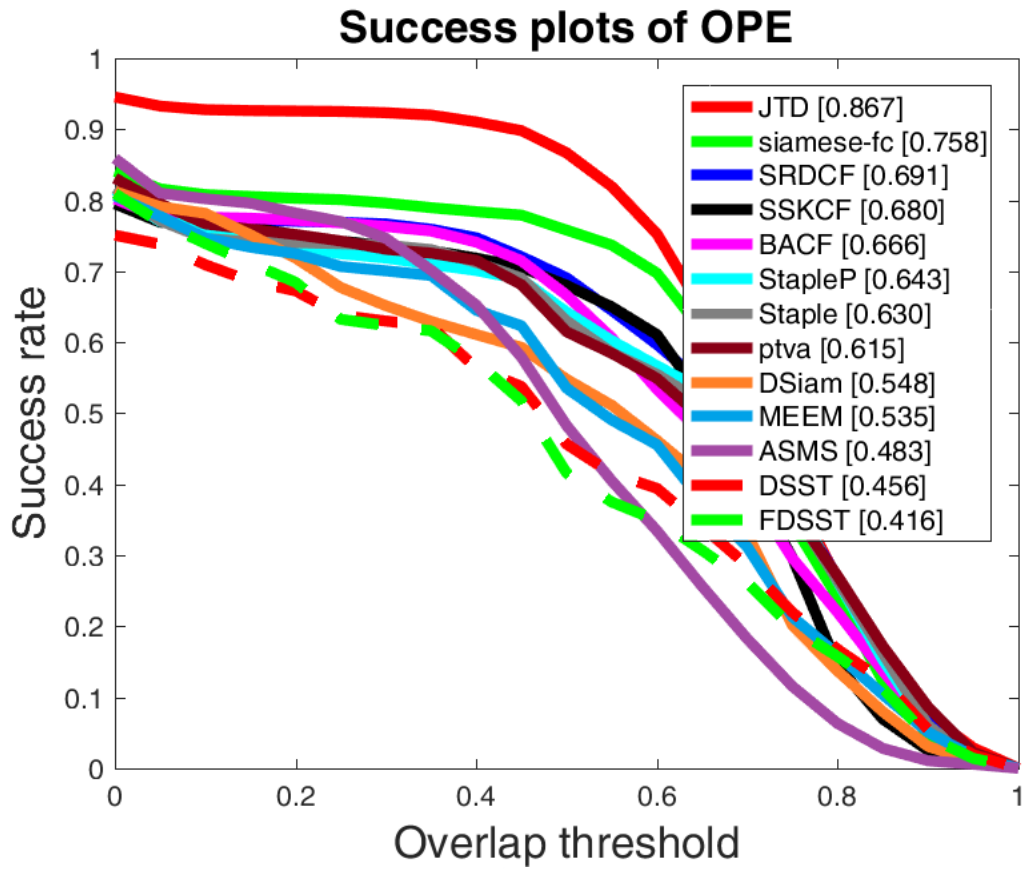
```
1 if received a request from MVA then
2     Initialize tracking
3 if T is already initialized then
4     Update tracking result;
5     Publish tracking message t;
```

Algorithm 4: Master Visual Analysis *MVA*

```
1 if T is not initialized then
2     Initialize tracker T;
3 if received a message d from D then
4     Reinitialize tracker T;
```

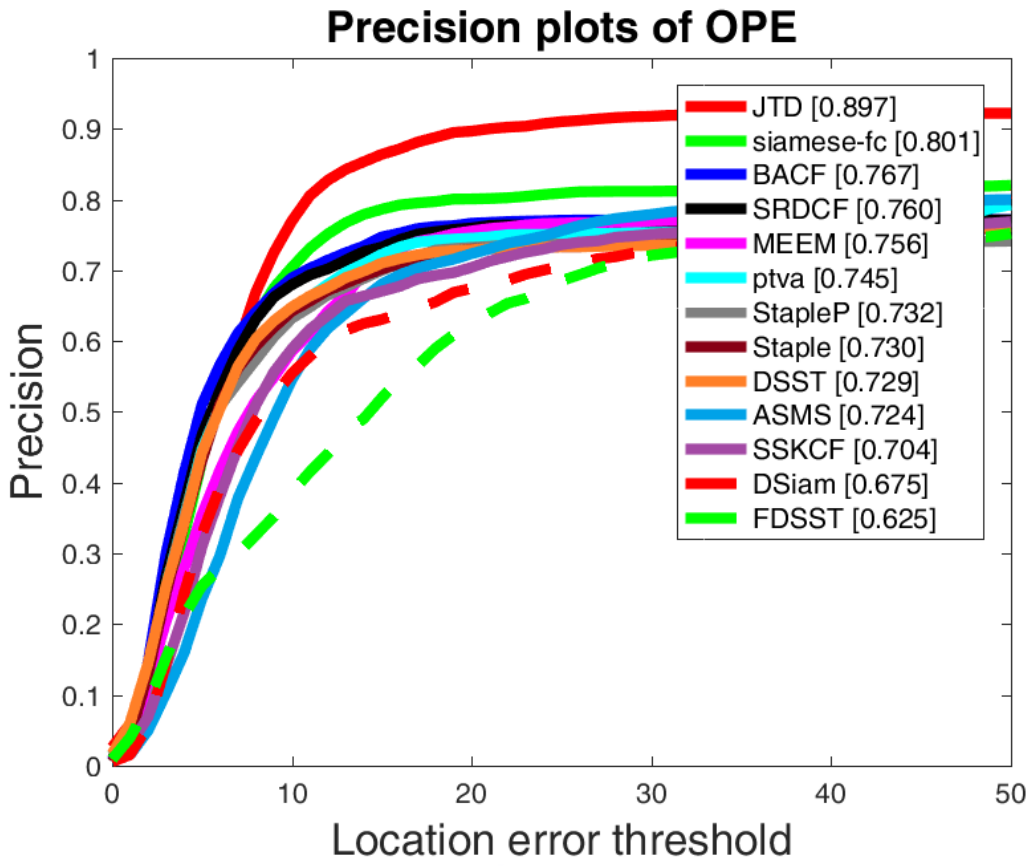
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking (JTD)



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

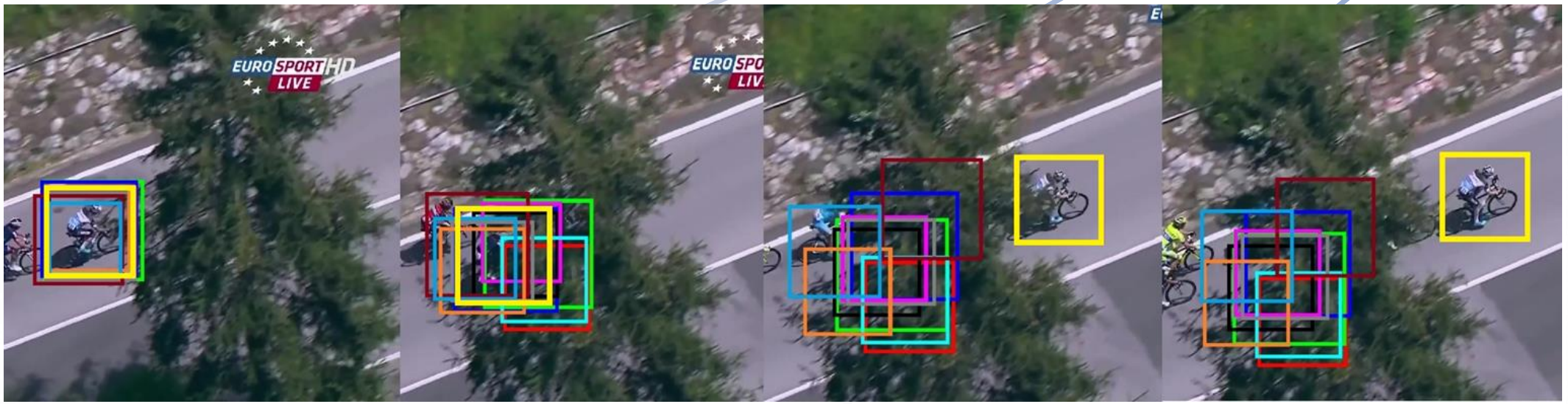
Joint Detection & Tracking (JTD)



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking

- Target re-initialization by the detector in hard tracking cases when tracking algorithms fail.



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Joint Detection & Tracking

- Target reinitialization by the detector in hard tracking cases when tracking algorithms fail.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)

Q & A

Thank you very much for your attention!

www.multidrone.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731667 (MULTIDRONE)