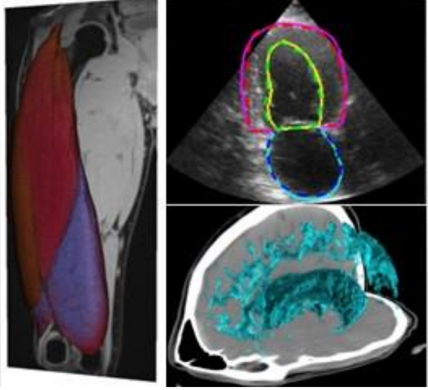
Deep learning for medical imaging school 2021

April 19—24 2021

LABEX PRIMES
UNIVERSITÉ DE LYON

Virtual edition



Fundamental concepts of deep learning

From the description of conventional architectures to medical imaging applications

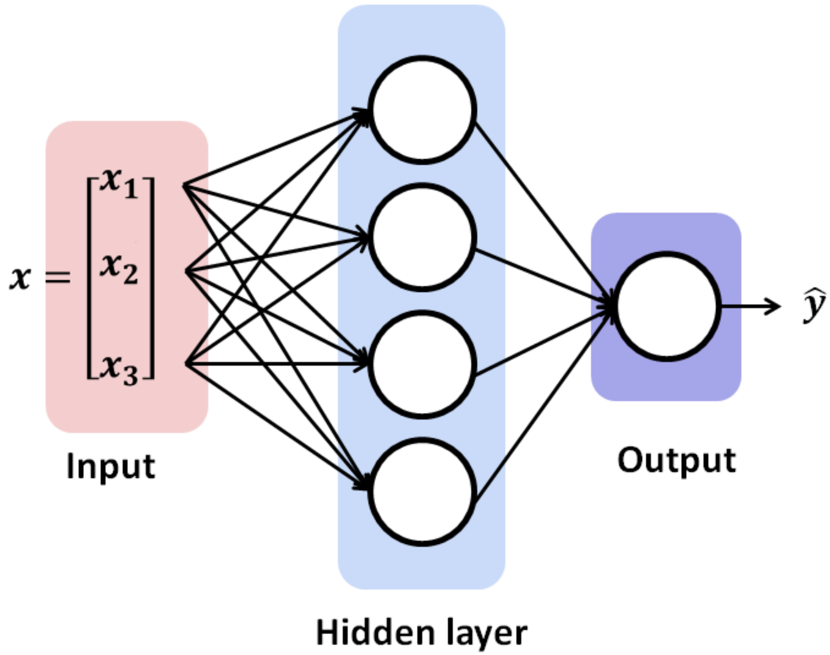
olivier.bernard@insa-lyon.fr
christian.desrosiers@etsmtl.ca

UNIVERSITÉ DE LYON | INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

UNIVERSITÉ DE SHERBROOKE

Image analysis through CNN

▶ Multi Layers Perceptron (MLP)



- Input image as a vector
- Image 256 x 256

  $$\boldsymbol{x}^{(i)} \in \mathbb{R}^{[65536\times 1]}$$

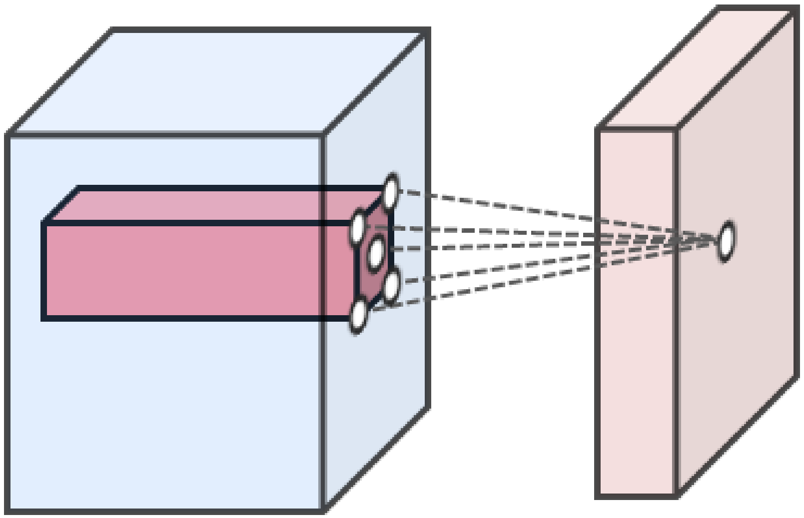
- One single hidden layer, $\boldsymbol{n}^{[1]} = 64$

  $$\#\ \boldsymbol{param} > 4\ \boldsymbol{M}$$

Too much parameters to learn for image analysis

Image analysis through CNN

- ▶ **How to adapt neural networks for image analysis ?**
  - **Introduction of *convolution layers***



- **Shared parameters**
- **Spatial consistency**

Much less parameters to learn !

Fundamental components

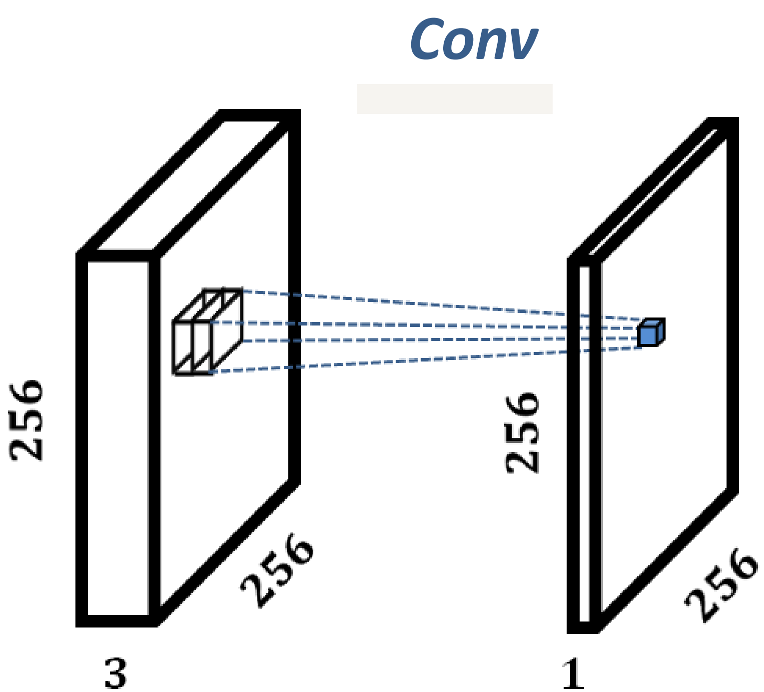
Convolution layers

Pooling leayers

Receptive field

Convolutional layer

- **Parameters to learn – *weights of convolutional filters***



$feature\ map$

- ***Ex.*** 3×3 **filter size**

$$\#\, param = 3 \times 3 \times 3 + 1$$
$$= 28$$

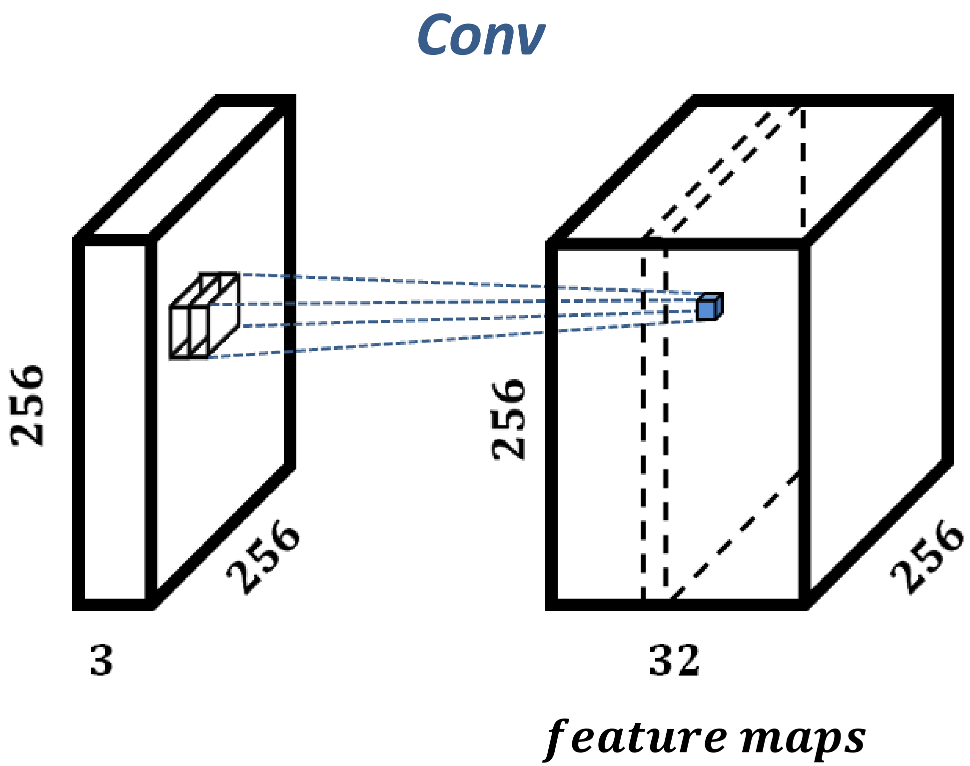
- ***Conv***

Input image filtering + activation function

$feature\ map$

Convolutional layer

▶ **Several *feature maps* per layer**

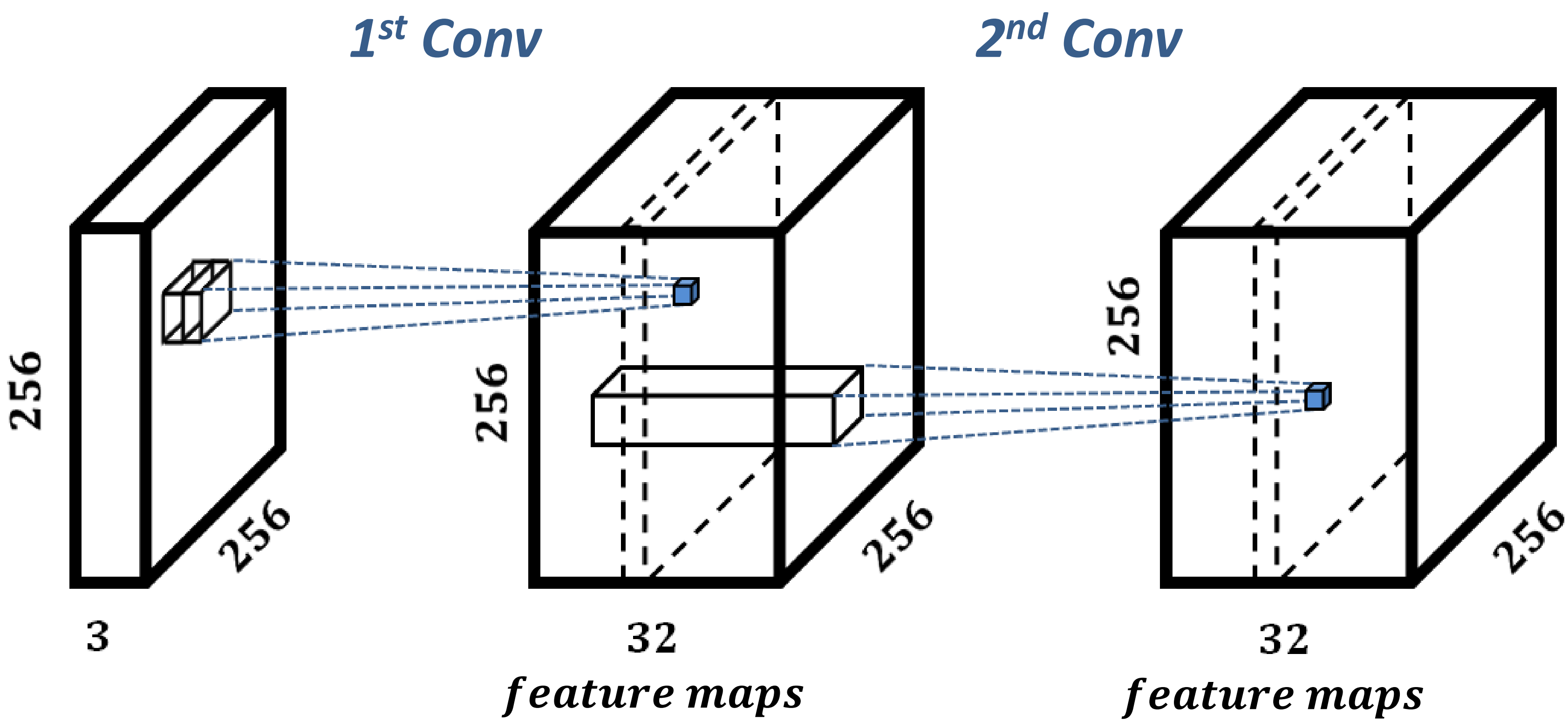


- ***Ex.*** 3×3 **filter size**

$$\begin{aligned} &\# \, param \\ &= 32 \times (3 \times 3 \times 3 + 1) \\ &= 896 \end{aligned}$$

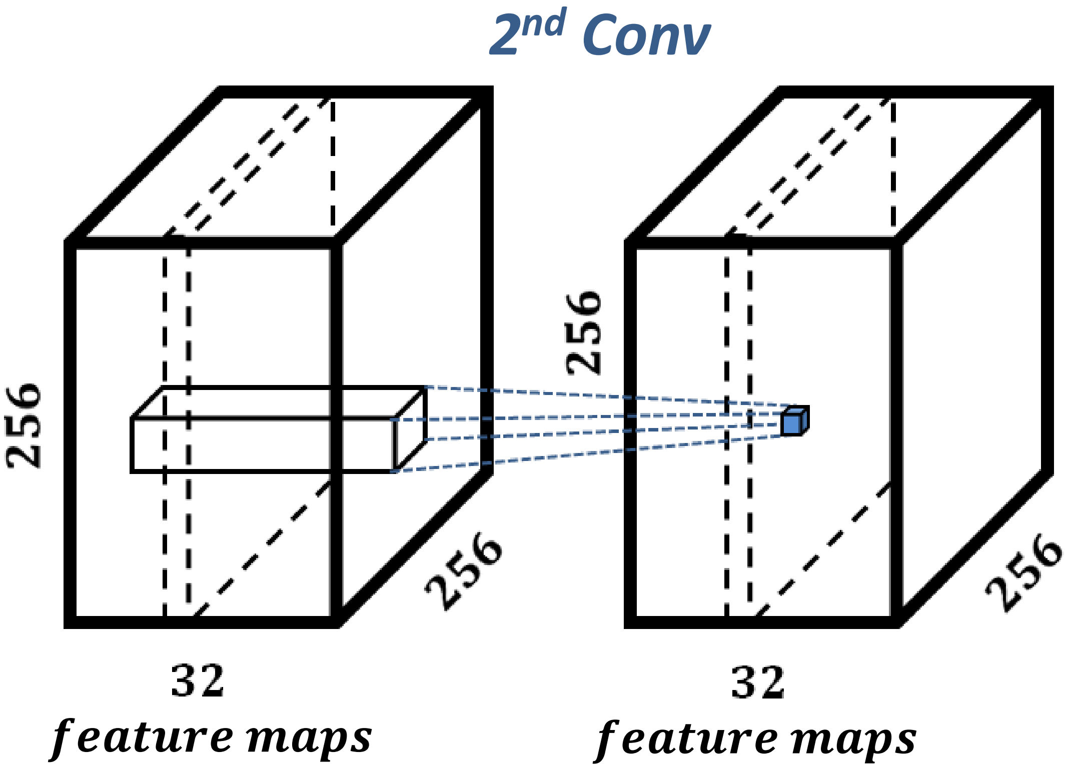
Convolutional layer

- **Multi-layers scheme**



Convolutional layer

▶ Multi-layers scheme



- *Ex.* 3×3 filter size

$$\begin{aligned}&\#\,param\\&= 32 \times (3 \times 3 \times 32 + 1)\\&= 9248\end{aligned}$$

Pooling

- **Applied separately to each feature map**
- **Reduction of the spatial resolution of the feature maps**
- **Reduction of the memory footprint / computational cost**
- **Introduction of invariance properties for small translation, rotation and scaling**

Max pooling

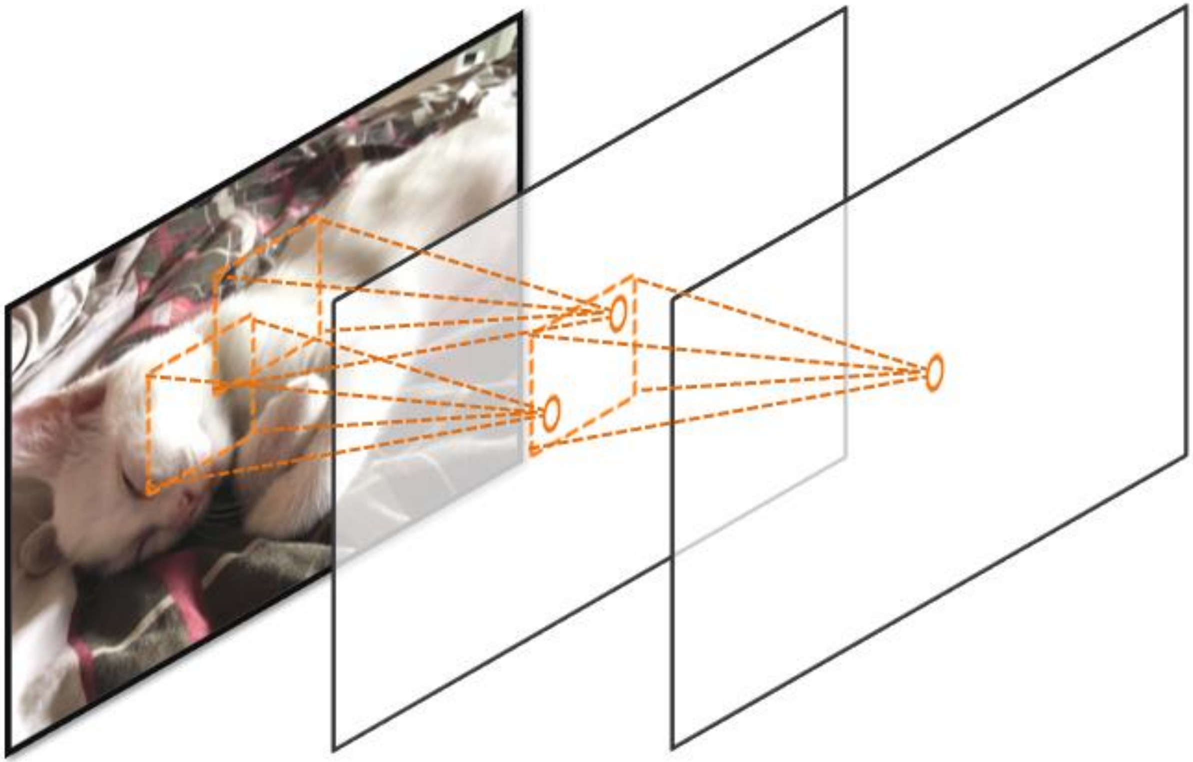
135	212	189	56
164	201	204	145
30	126	189	156
36	45	38	12

$\downarrow$ 2×2 Pool size

(Stride = 2)

212	204
126	189

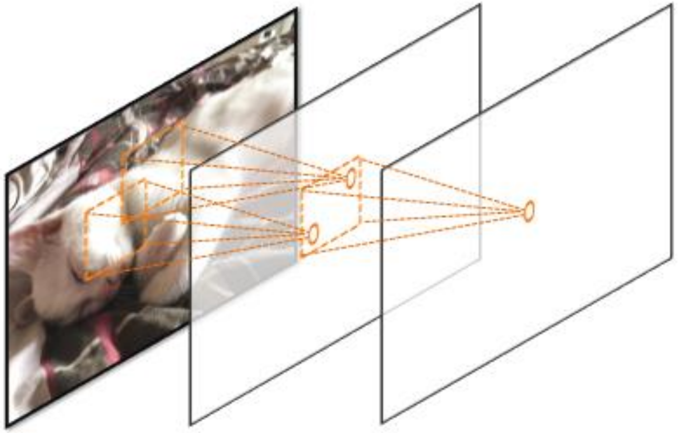
Receptive field



Part of the input image that impacts the value of a given point on a feature map

Receptive field

- Receptive field increases with the depth of a network
- A large receptive field is essential to capture spatial contextual information
- At a cost of higher number of parameters



How to have a large receptive field without too many parameters ?

Applications

Image classification

Image classification

Dog - 10%
Cat - 85%
Horse - 5%
Cow - 0%

Predict a single class (or a probability distribution for a set of classes) for a given image

Some difficulties

37	49	43	43	63	45	51	56	65	59	28
47	64	68	37	48	56	37	47	61	47	65
56	67	64	39	80	66	31	48	49	33	45
38	49	32	75	48	49	71	35	47	27	62
61	62	33	64	60	49	35	40	70	49	47
52	32	31	56	34	32	34	27	43	36	60
34	77	26	36	46	27	62	76	70	65	27
69	36	49	37	34	41	75	61	69	46	76
31	40	62	30	67	43	54	77	72	72	70
42	69	65	76	73	61	64	34	53	66	67
39	52	55	64	45	78	34	76	60	57	70
55	68	31	56	63	77	78	69	78	35	47
31	47	60	43	42	51	55	57	50	78	59
48	41	32	35	55	39	63	50	29	40	57
31	28	33	60	71	68	28	40	73	76	55
32	63	31	80	58	67	70	67	60	38	41
69	49	33	35	44	66	67	38	45	46	39
42	50	35	40	42	66	32	29	80	30	50
59	59	36	47	50	31	54	68	38	61	38
79	29	43	30	49	63	43	62	61	35	70

What a human sees **What a computer sees**

Others difficulties

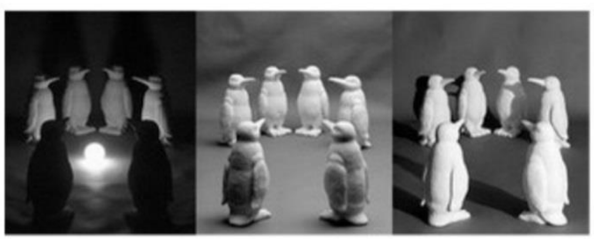
Observation point of view

Scale change

Deformation

Occlusion

Illumination conditions



Texture

Intra-class variation

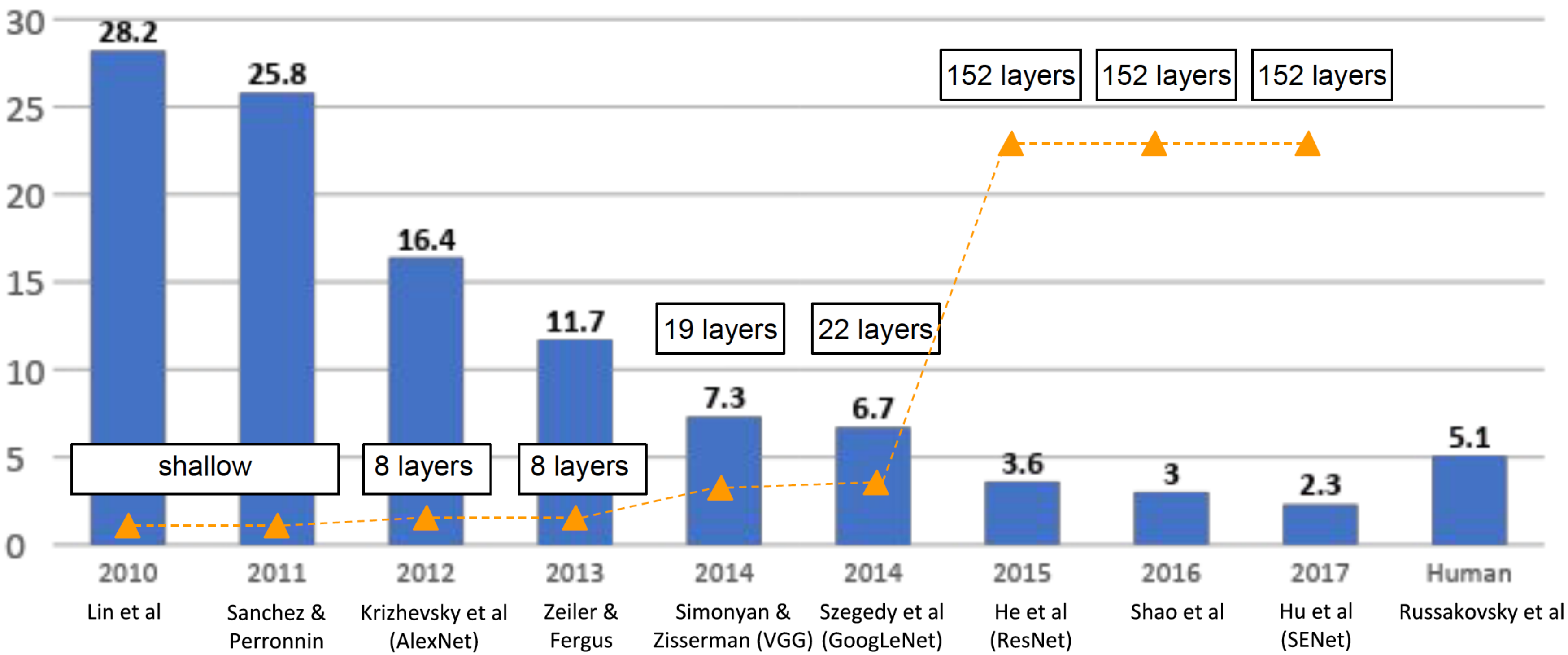
Simple for a human, what about for a computer ?

ImageNet

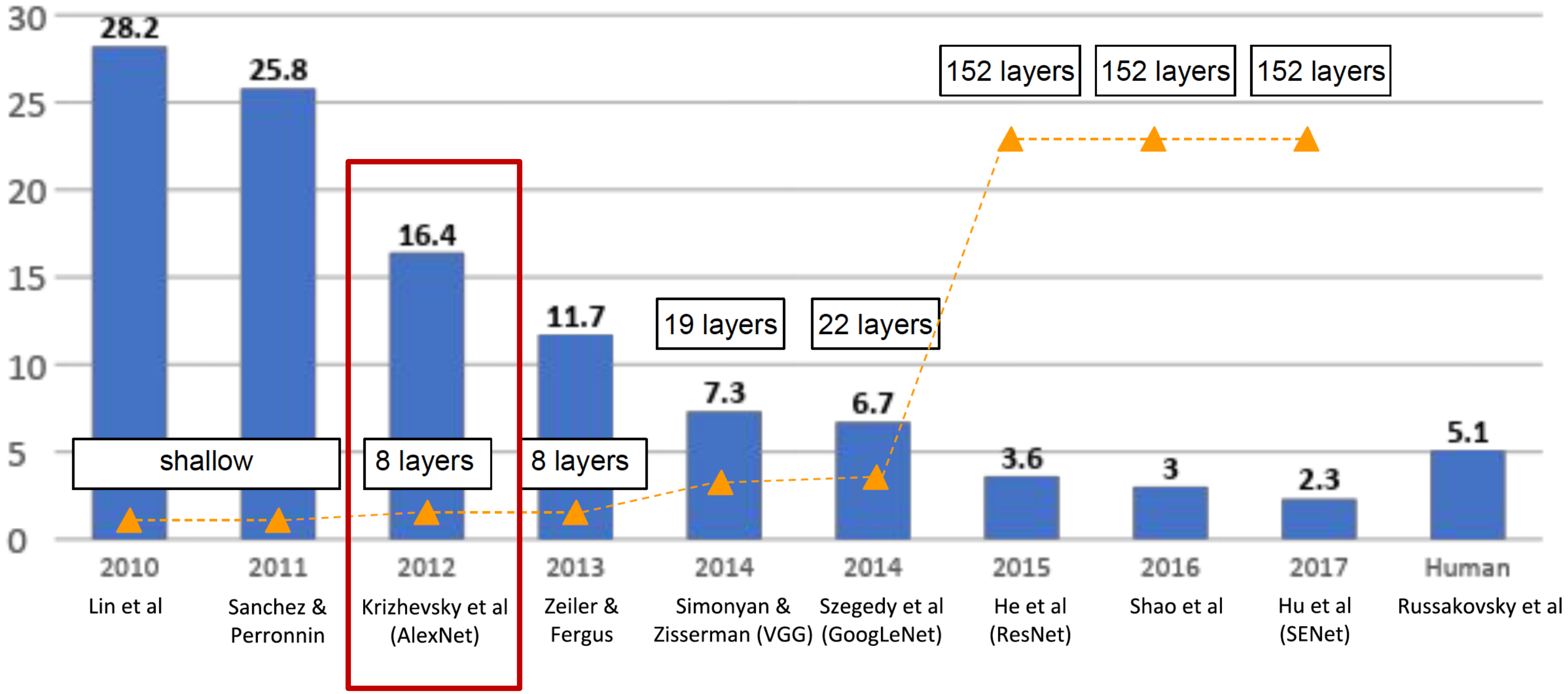
- Challenge for image classification (2010 $\rightarrow$ 2017)
- 1 000 object classes to recognize
- 1 431 167 images

ImageNet

▶ Annual ranking

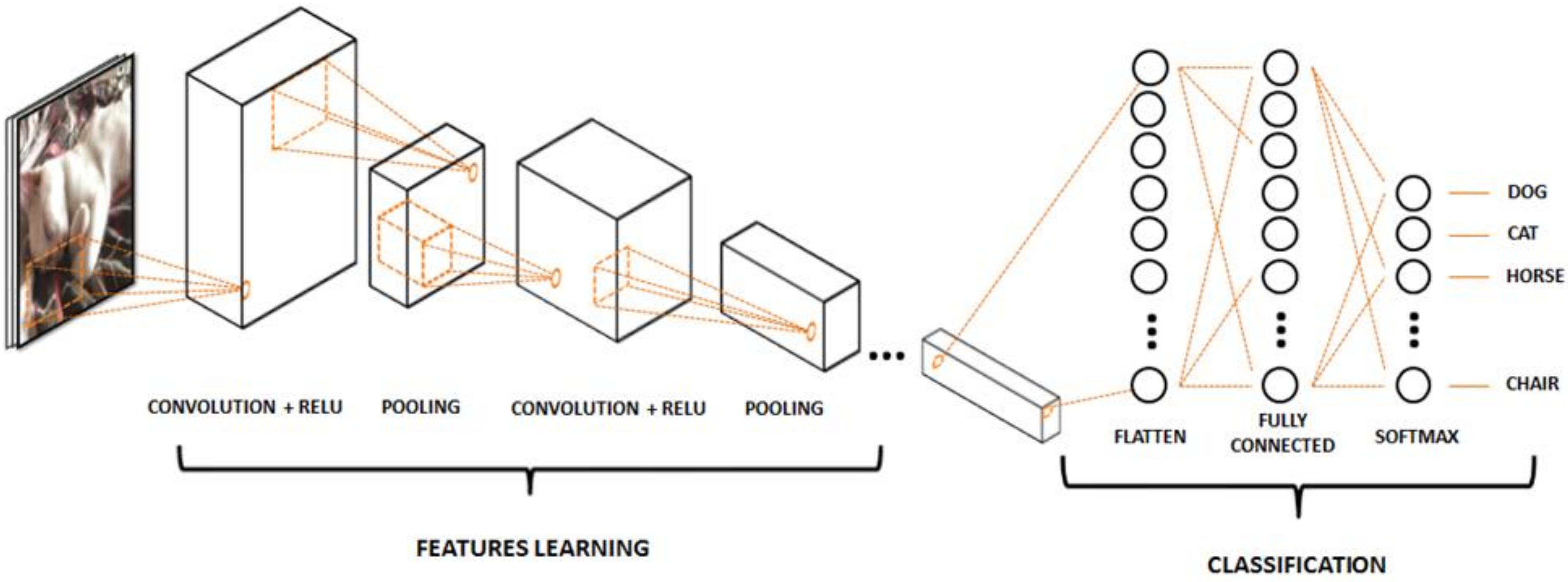


AlexNet



[Krizhevsky, NIPS, 2012]

AlexNet

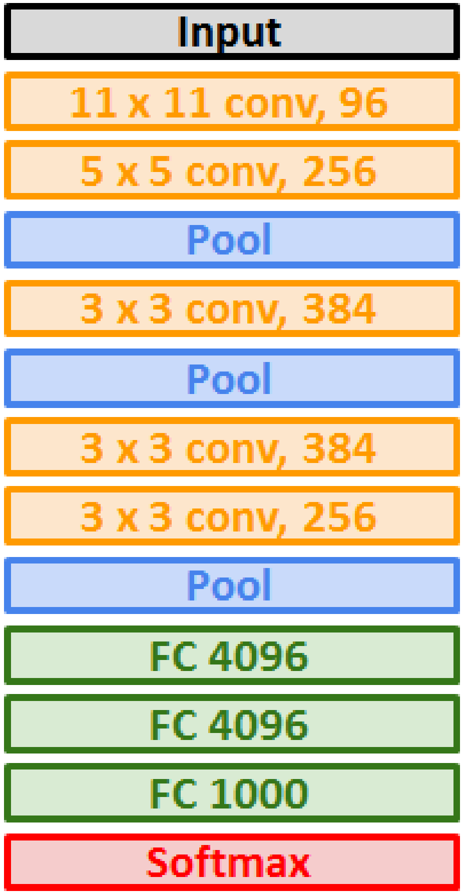


CONVOLUTION + RELU
POOLING
CONVOLUTION + RELU
POOLING
FEATURES LEARNING
FLATTEN
FULLY CONNECTED
SOFTMAX
DOG
CAT
HORSE
CHAIR
CLASSIFICATION

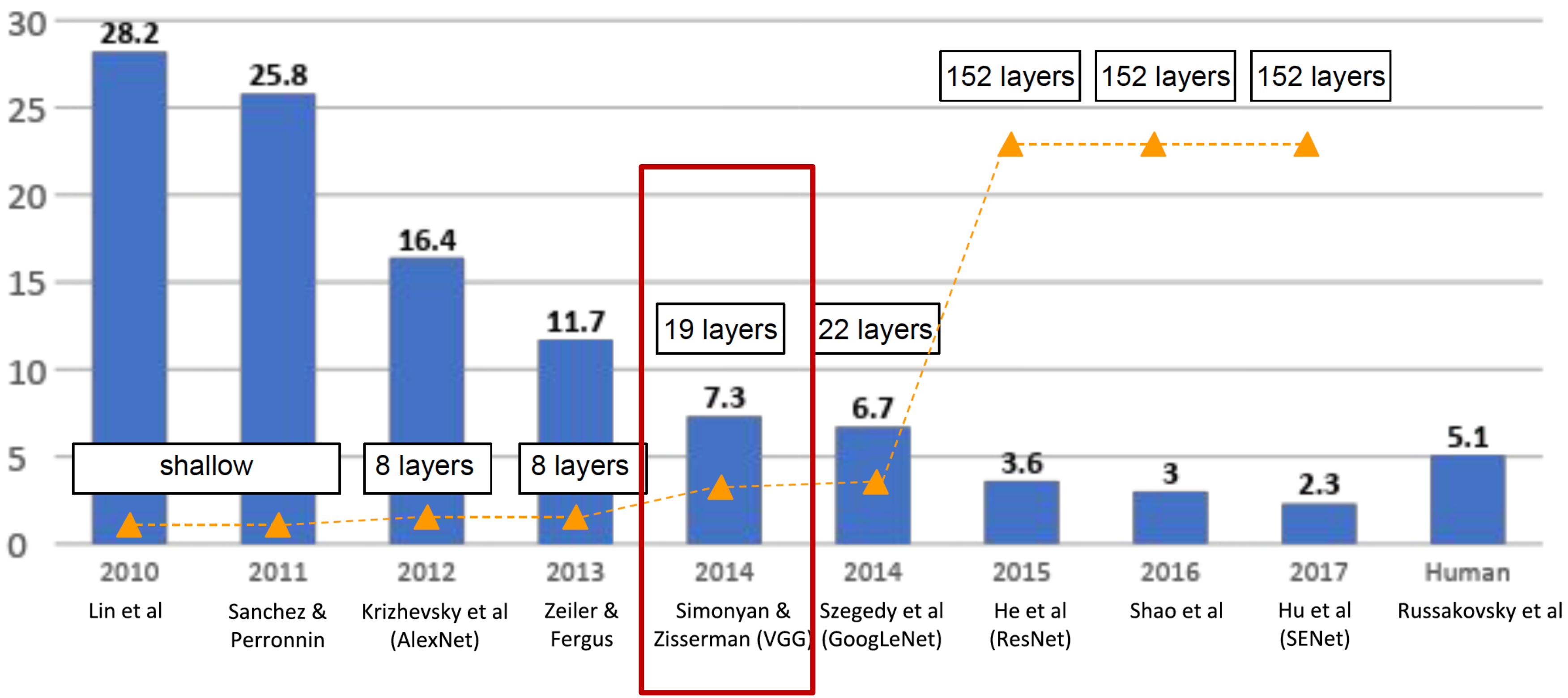
AlexNet

- First deep network to work nicely on ImageNet
- Exploit fundamental steps that are still using (ReLU, data augmentation, dropout)
- 8 convolutional layers
- # param ~ 62M
- Use GPU for training

Responsible for the deep learning revolution in computer vision



VGG



[Simonyan and Zisserman, arxiv, 2014]

VGG

- **Simpler architecture**

  3×3 convolutions, ReLU and 2×2 max pooling

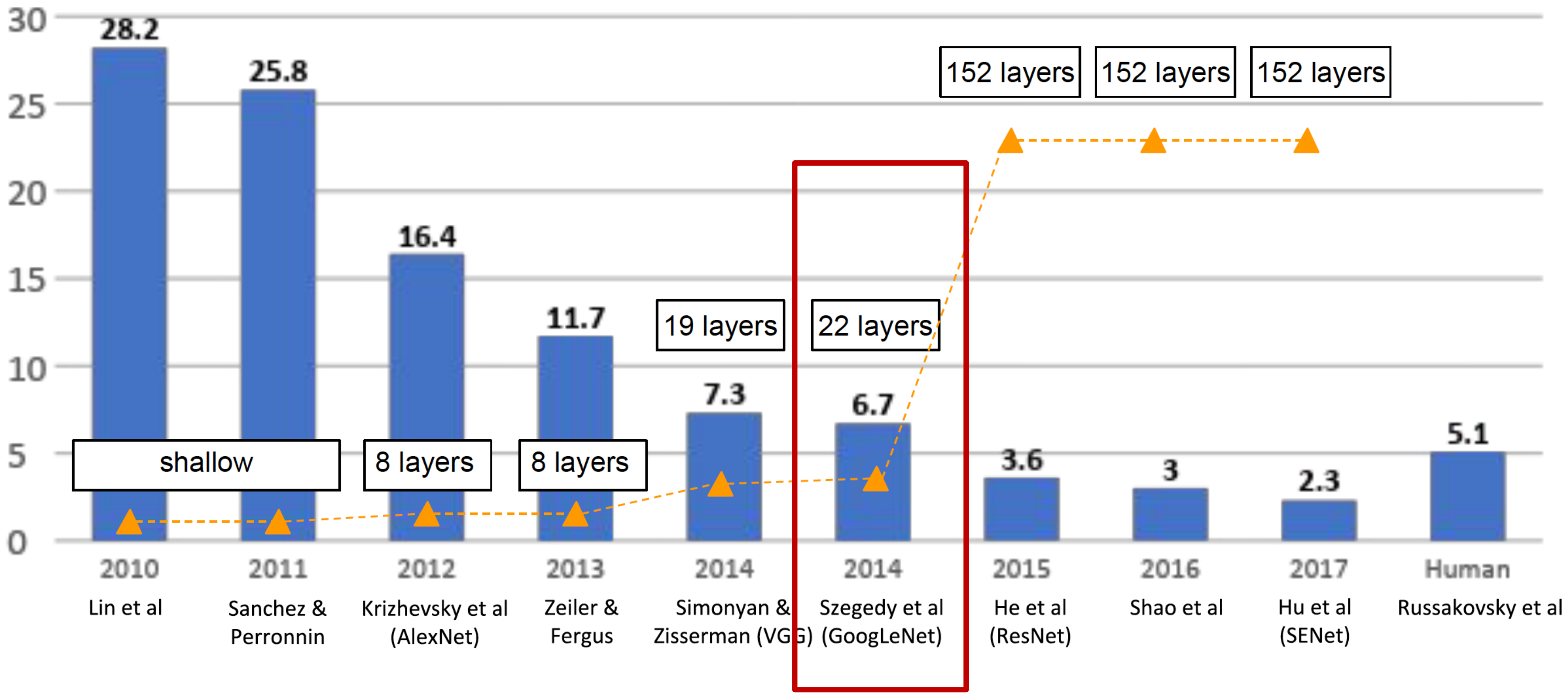
- **Deeper network**

  17 layers (vs 8 layer for AlexNet)

- **Key idea:**

Cascade 2 convolutions of size 3×3 produces the same receptive field than a single convolution of size 5×5 but with few parameters

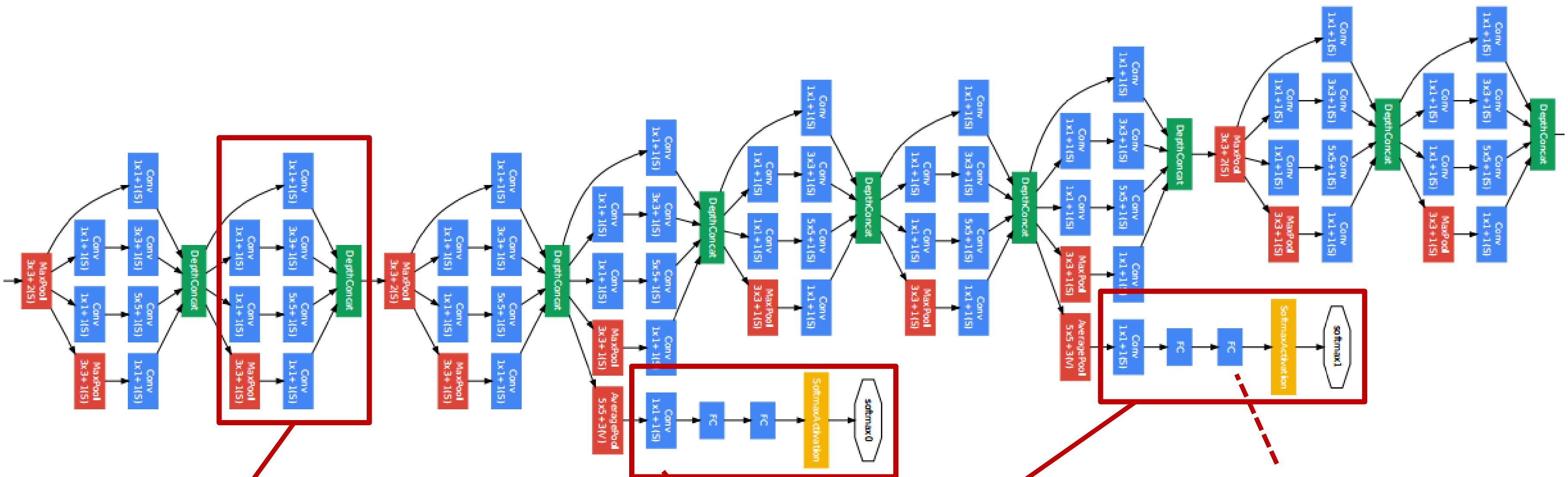
GoogLeNet



[Szegedy, CVPR, 2015]

GoogLeNet

- Network completely redesigned to be very deep



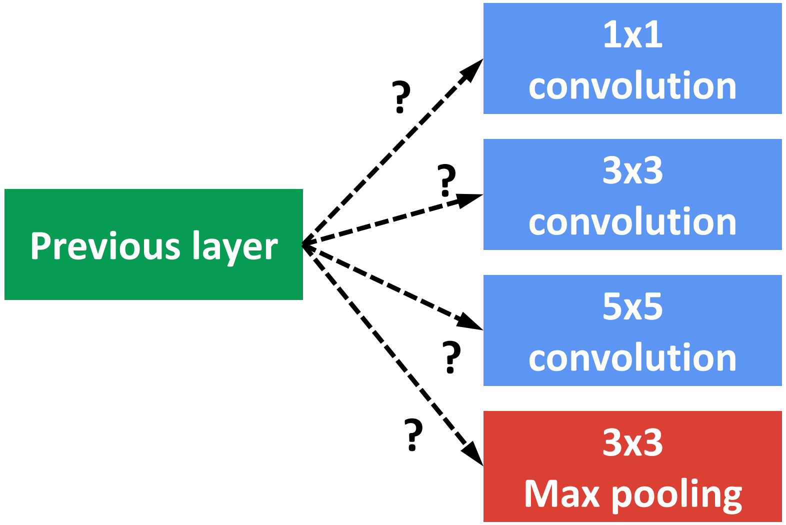
1) Repetitive blocks
Inception module

2) Intermediate loss function to inject gradients into the intermediate layers

3) Fully connect layers replaced by average pooling (less parameters)

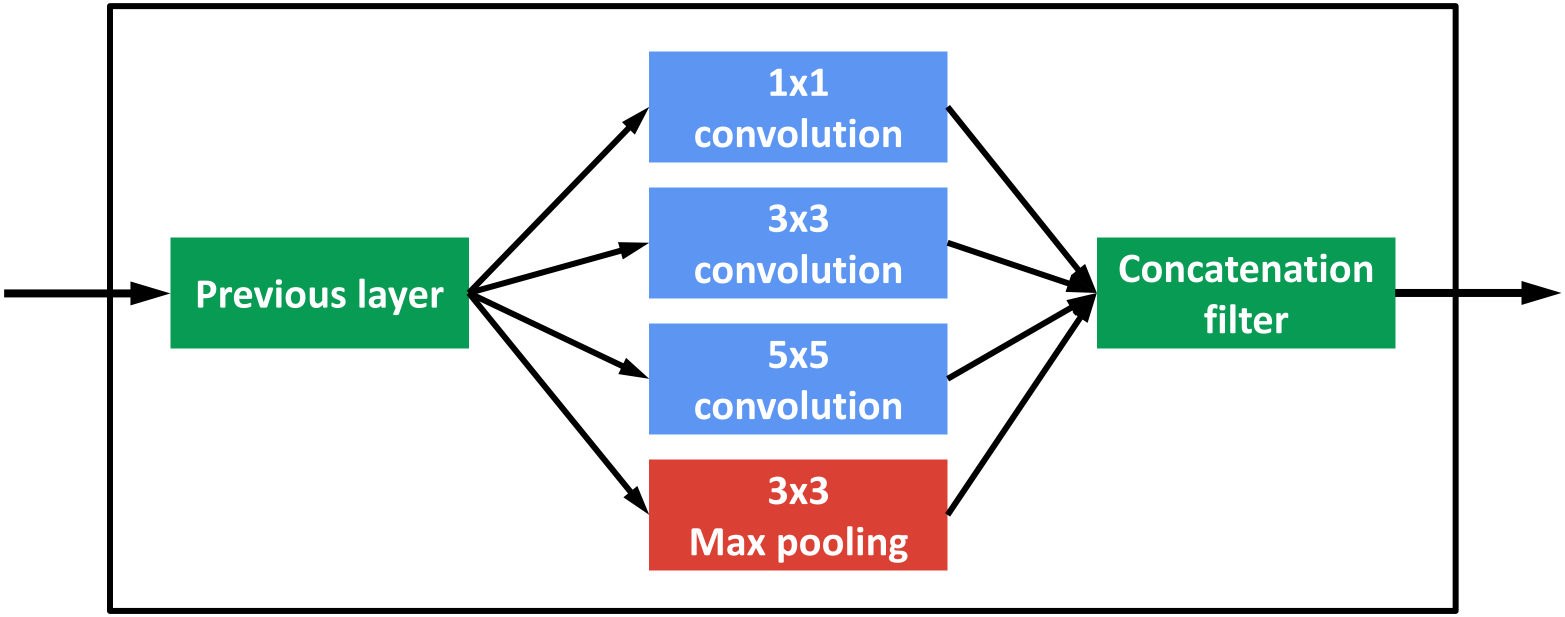
Inception module

- **Choice for each layer**
  - **Convolution or pooling ?**
  - **If convolution, what size of filter ?**



Inception module

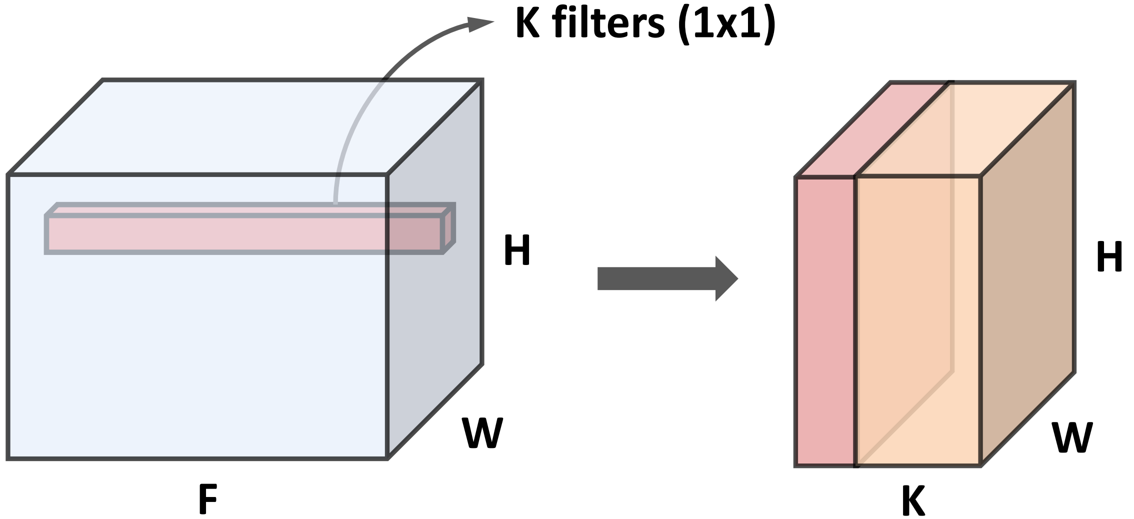
- **Key idea**
  - Compute each output in parallel
  - Concatenation of the results
  - Let the learning process choose !



Difficulty: too much outputs and parameters

Inception module

- **Key idea: 1×1 convolution**

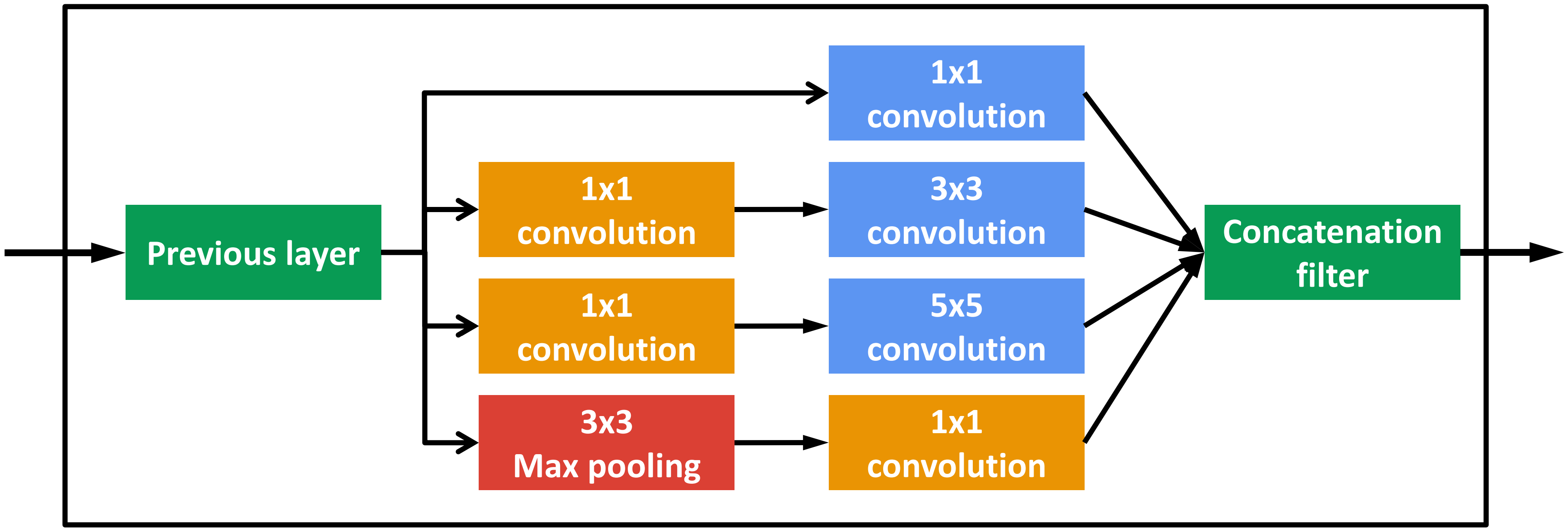


Dimension reduction for K<F

Acts as a feature pooling function that can be learned

Inception module

- **Key idea**



Dimension reduction through bottleneck layers composed by 1×1 convolutions

Efficiency

AlexNet

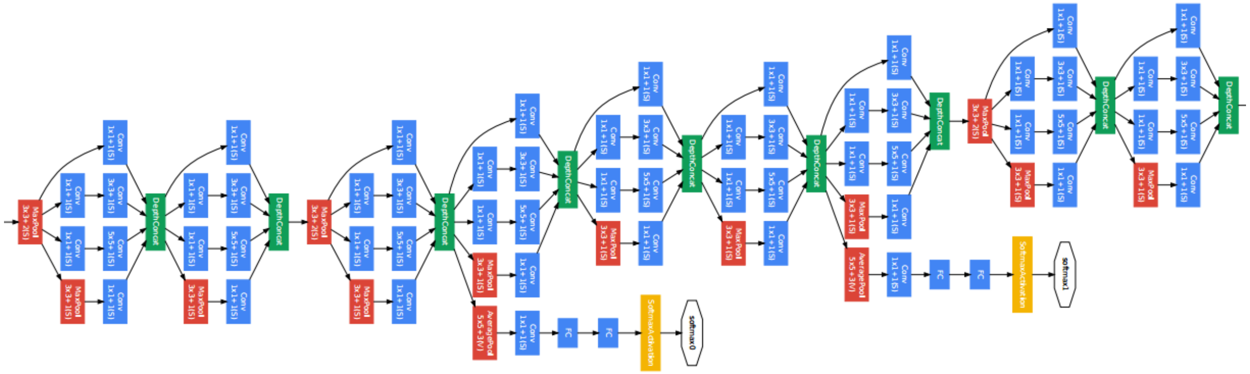
8 layers

 # param ~ 62M

GoogLeNet

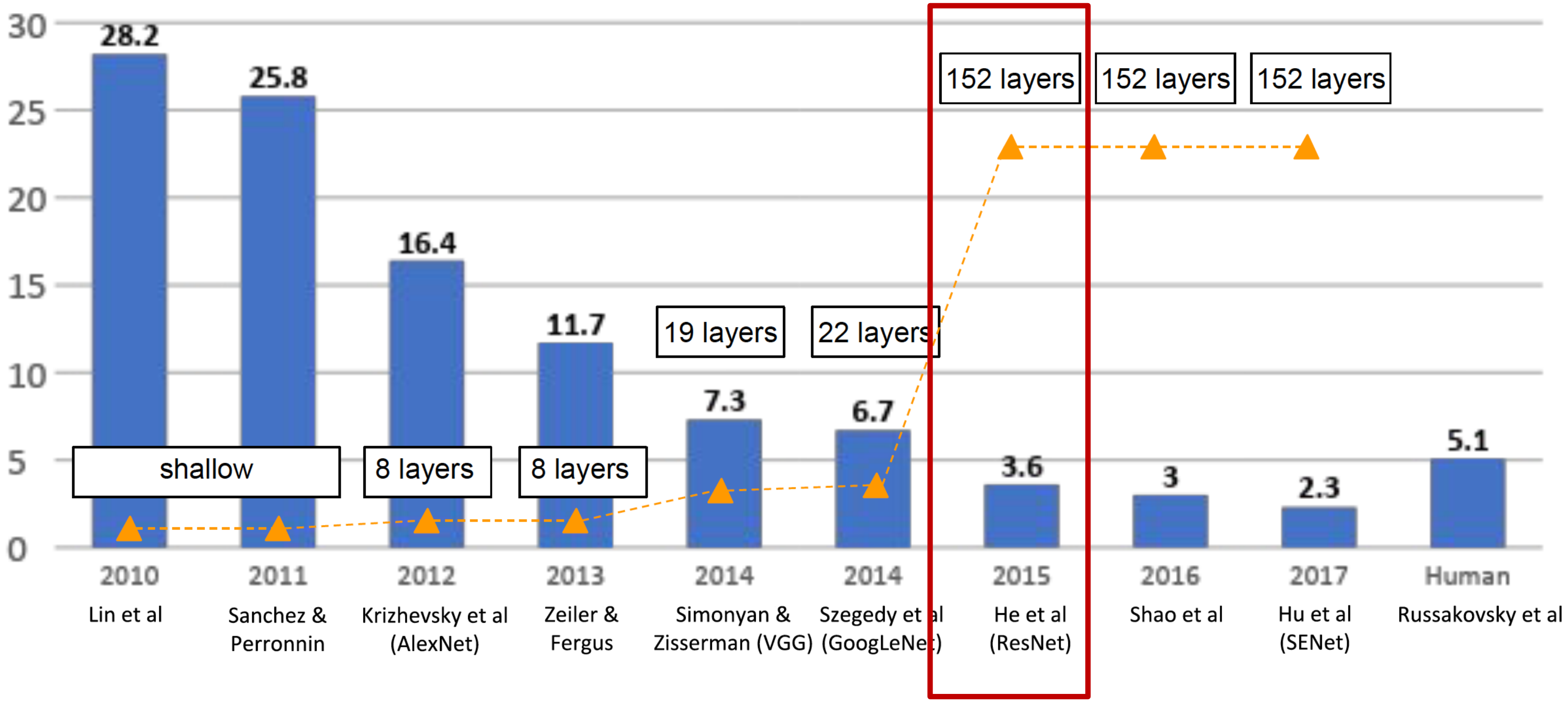
22 layers

 # param ~ 5M



GoogLeNet has 12x less parameters than AlexNet !

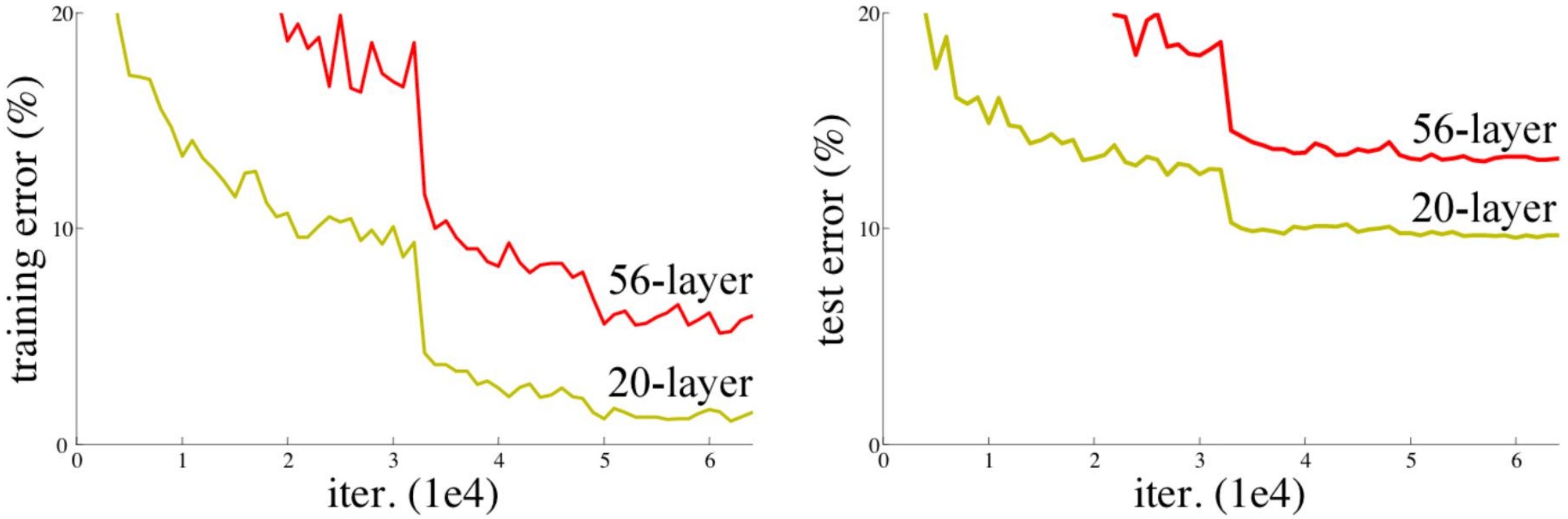
ResNet



[He, CVPR, 2016]

ResNet

► **What happens to an even deeper network?**



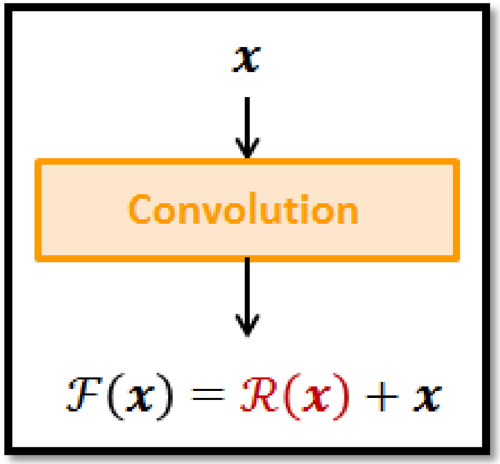
- **Higher training and testing errors!**

Optimization problem: *vanishing gradient*

ResNet

▶ Key idea

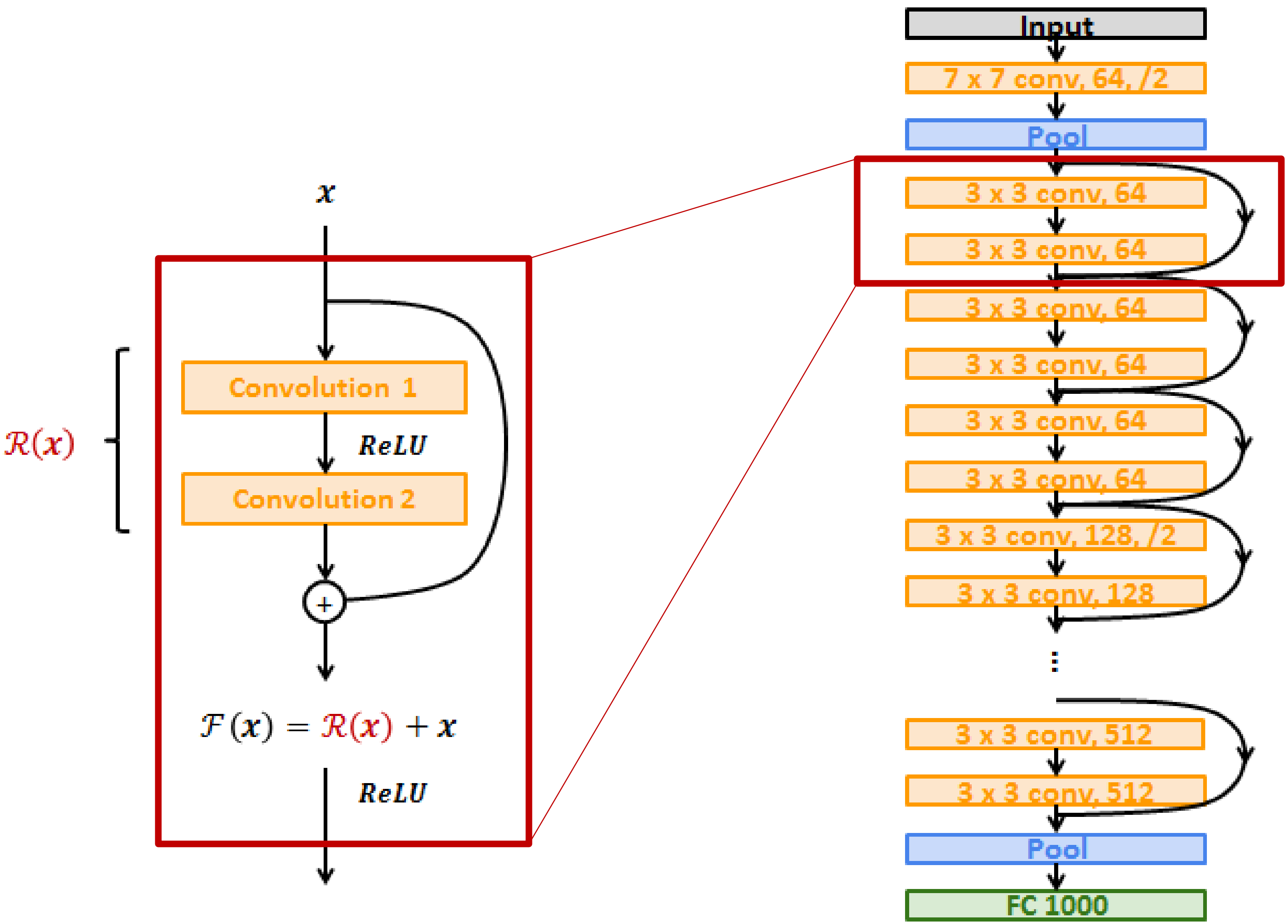
Estimate residual rather than the transformation itself



▶ Benefits

- **Modeling of less information, potentially easier to learn**
- **Residual connections preserve the gradient flow during back propagation**
- **Possible design of very deep architectures ($>$ 100 layers)**

ResNet

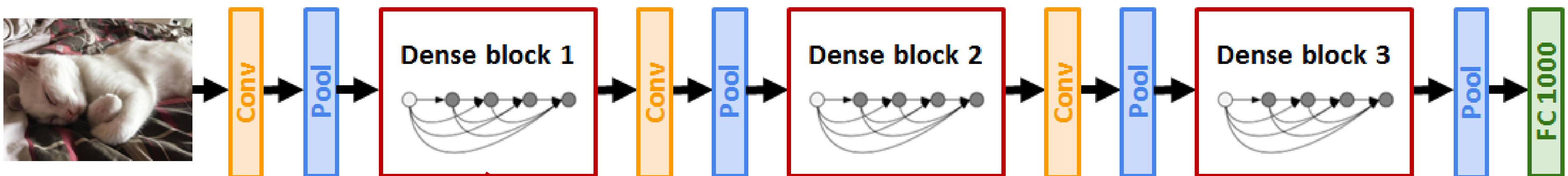


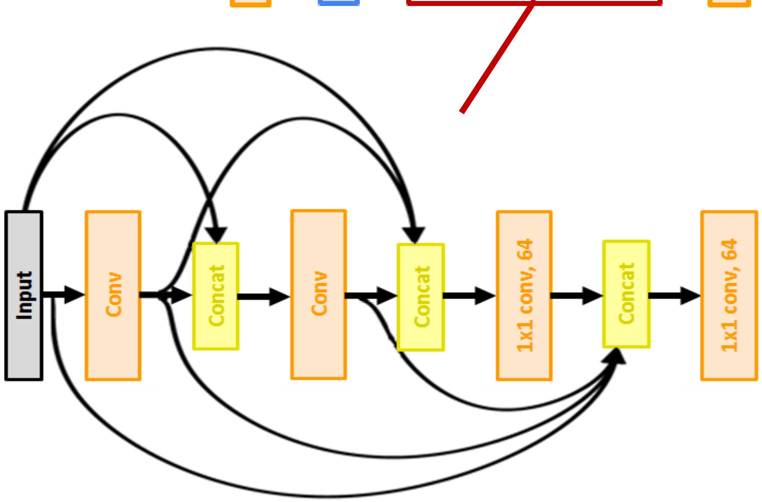
x
Convolution 1
ReLU
Convolution 2
+
ℛ(x)
ℱ(x) = ℛ(x) + x
ReLU
Input
7 x 7 conv, 64, /2
Pool
3 x 3 conv, 64
3 x 3 conv, 64
3 x 3 conv, 64
3 x 3 conv, 64
3 x 3 conv, 64
3 x 3 conv, 64
3 x 3 conv, 128, /2
3 x 3 conv, 128
⋮
3 x 3 conv, 512
3 x 3 conv, 512
Pool
FC 1000

DenseNet (Densely connected)

▶ **Key idea**

The features calculated in a layer are concatenated with the inputs of all other layers in a block





- **Efficient use of multi-scale features**
- **Gradient propagation through each layer during back propagation**

[Huang, CVPR, 2017]

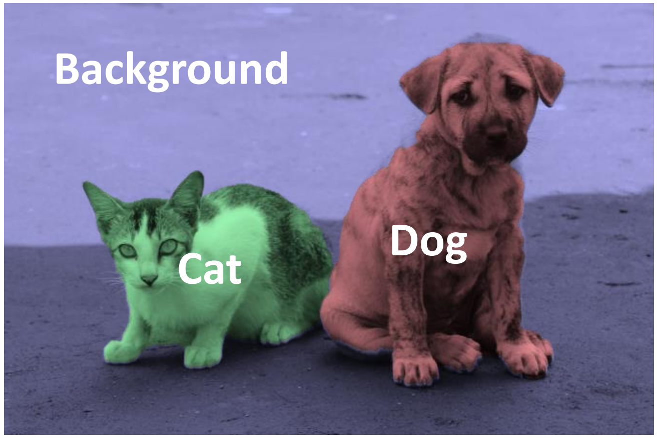
Applications

Semantic segmentation

Semantic segmentation

- Predict the right class for each pixel of an image

Input image



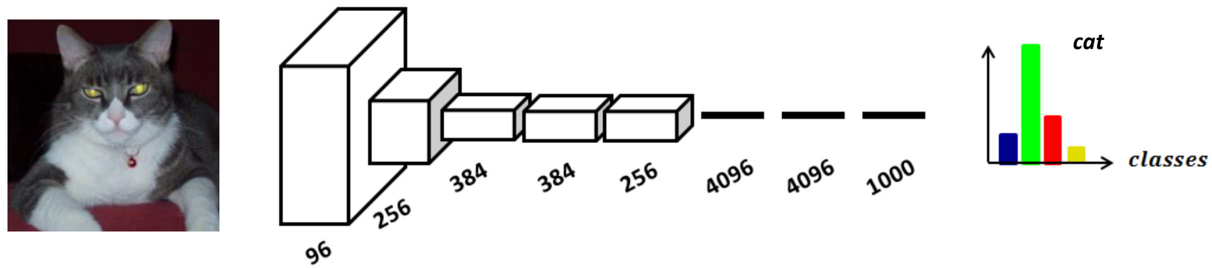
Segmentation

Can be seen as a dense and structured classification problem

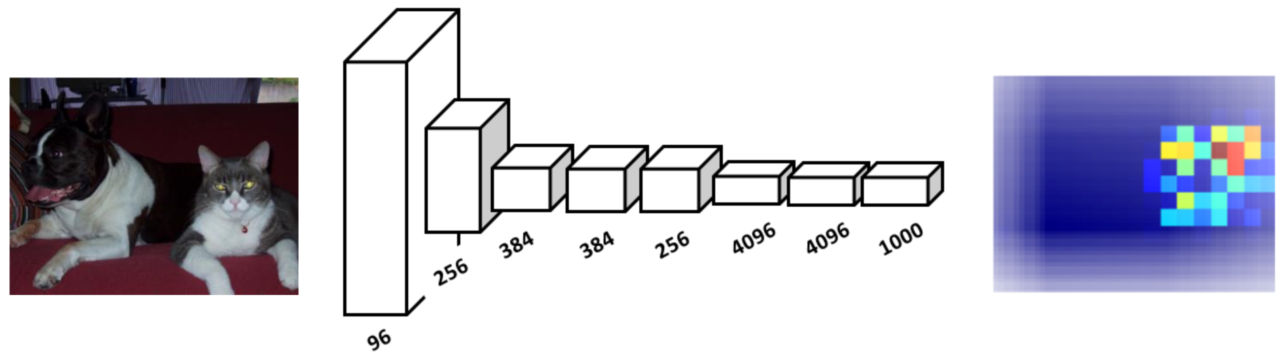
Fully-CNN: from classification to segmentation tasks

[Long, ICCV, 2015]

Standard CNN for classification



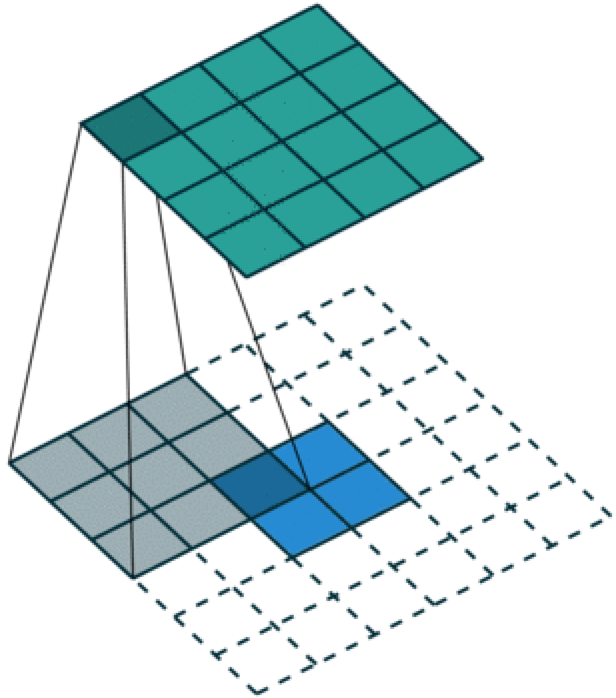
Fully-CNN



- **Generation of very coarse segmentation maps**

Adding oversampling operations at the end of the network

Oversampling layer

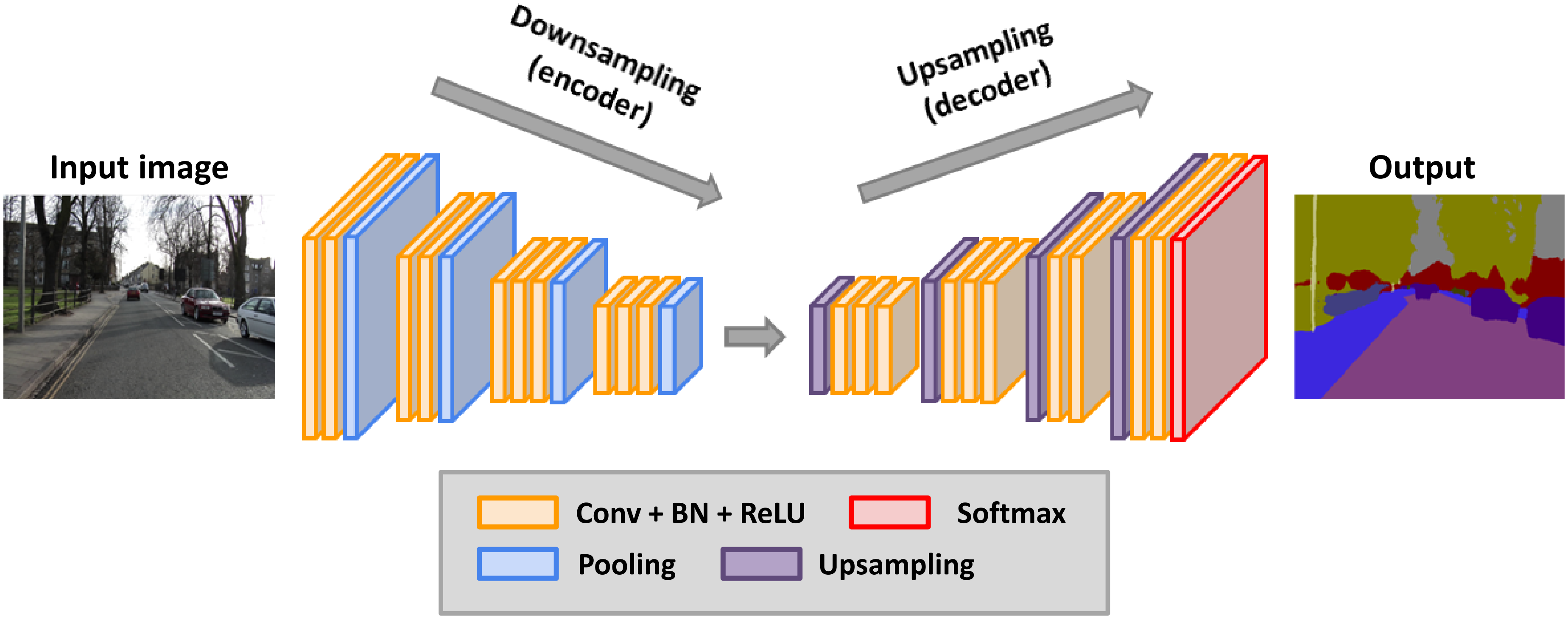


Padding 2, stride 1

Increased spatial dimension

Encoder / decoder based architectures

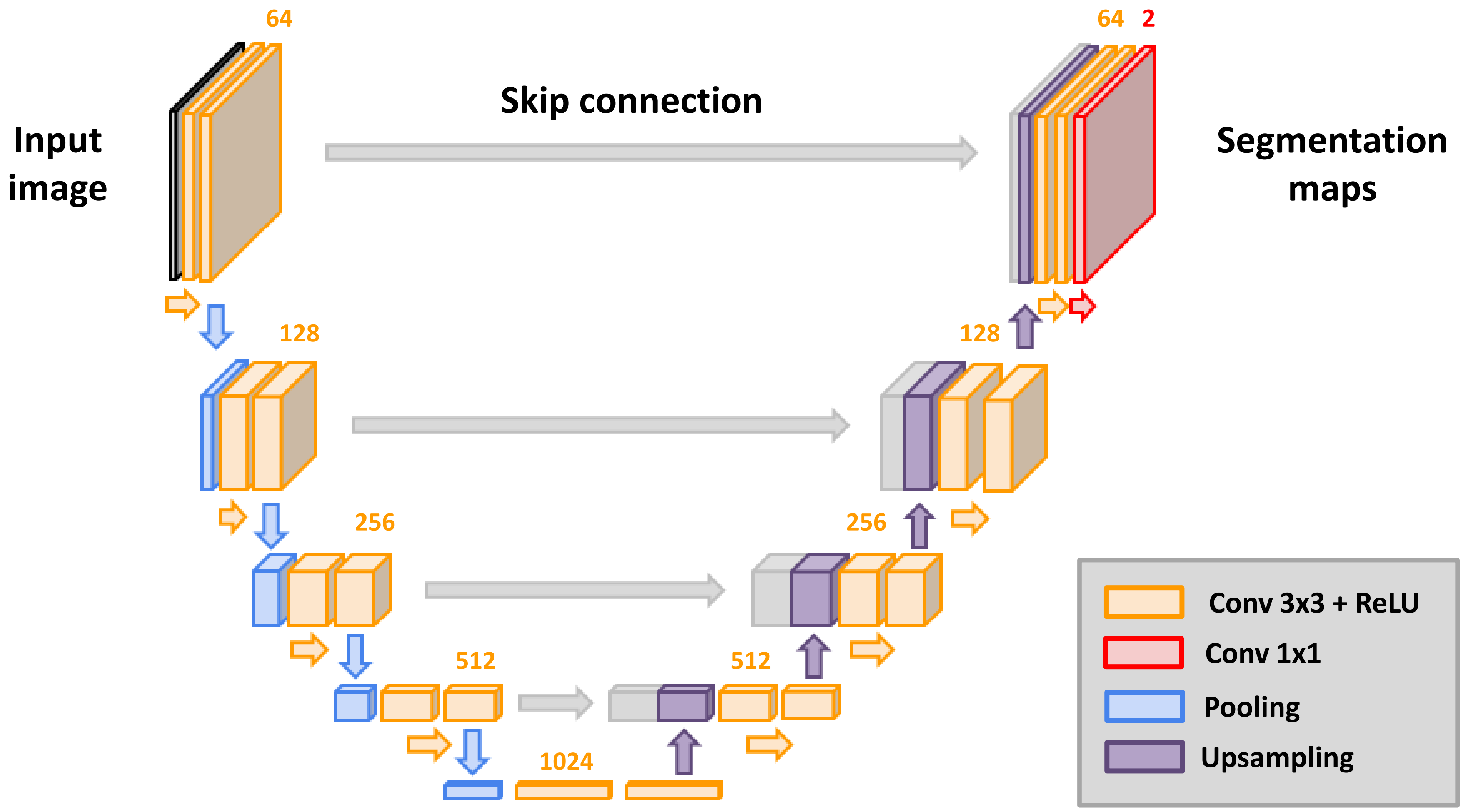
[Badrinarayanan, PAMI, 2017]



- **Spatial resolution lost during subsampling**

Adding of skip connections between the encoder and the decoder

U-Net



[Ronneberger, MICCAI, 2015]

ENet

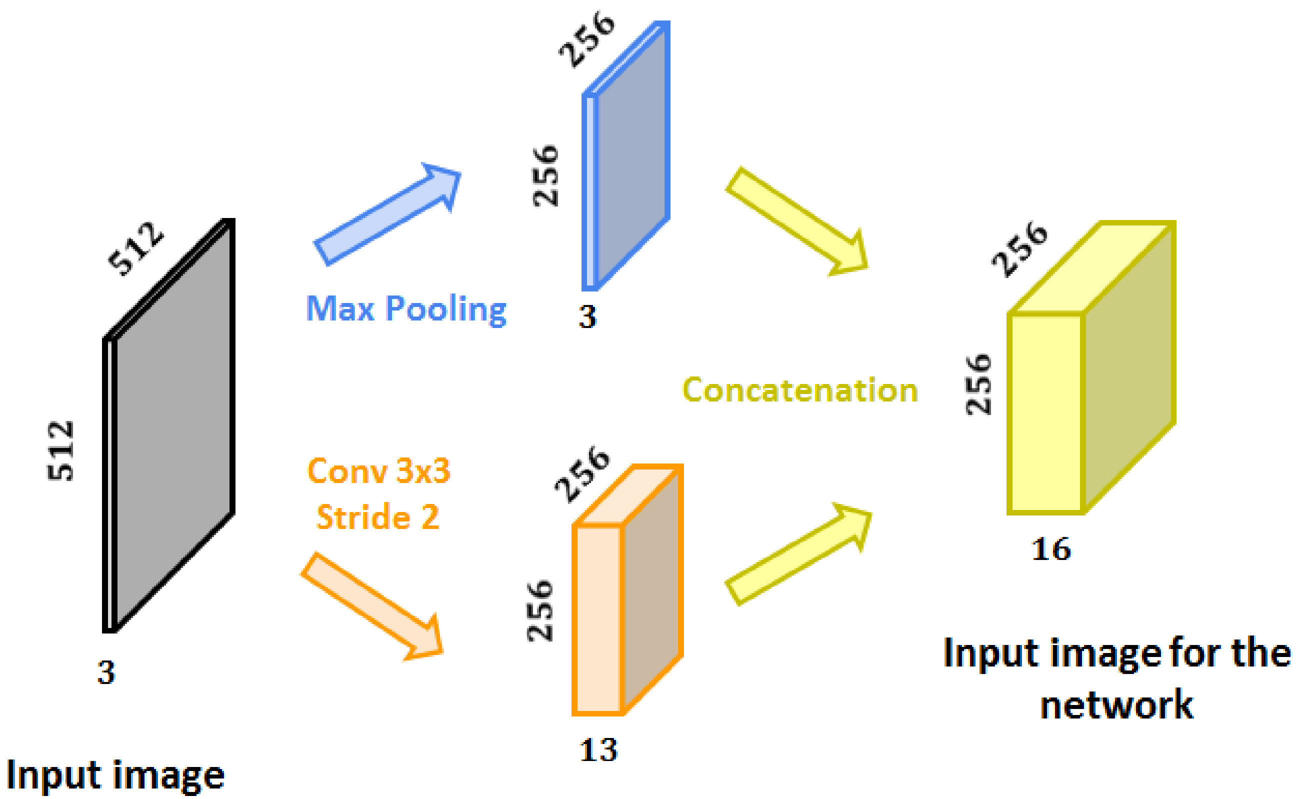
Exploit all the good ideas to create a light and efficient network

- **Key points**
  - **Expression of the input image in an optimal space with reduced dimensions**
  - **ResNet-based architecture to create a deep network**
  - **Use of features pooling (1x1 conv) to reduce the total number of parameters**

[Paszke, arXiv, 2016]

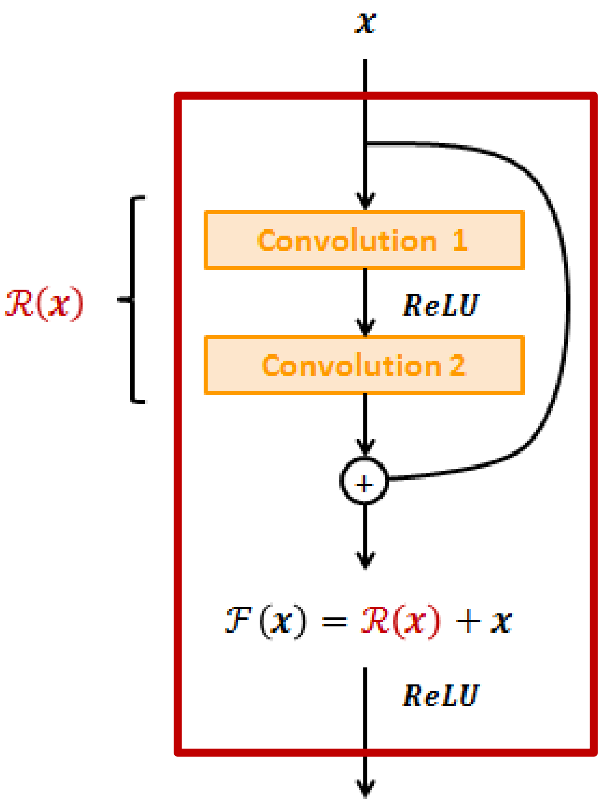
ENet

▶ **Modeling of the input image**

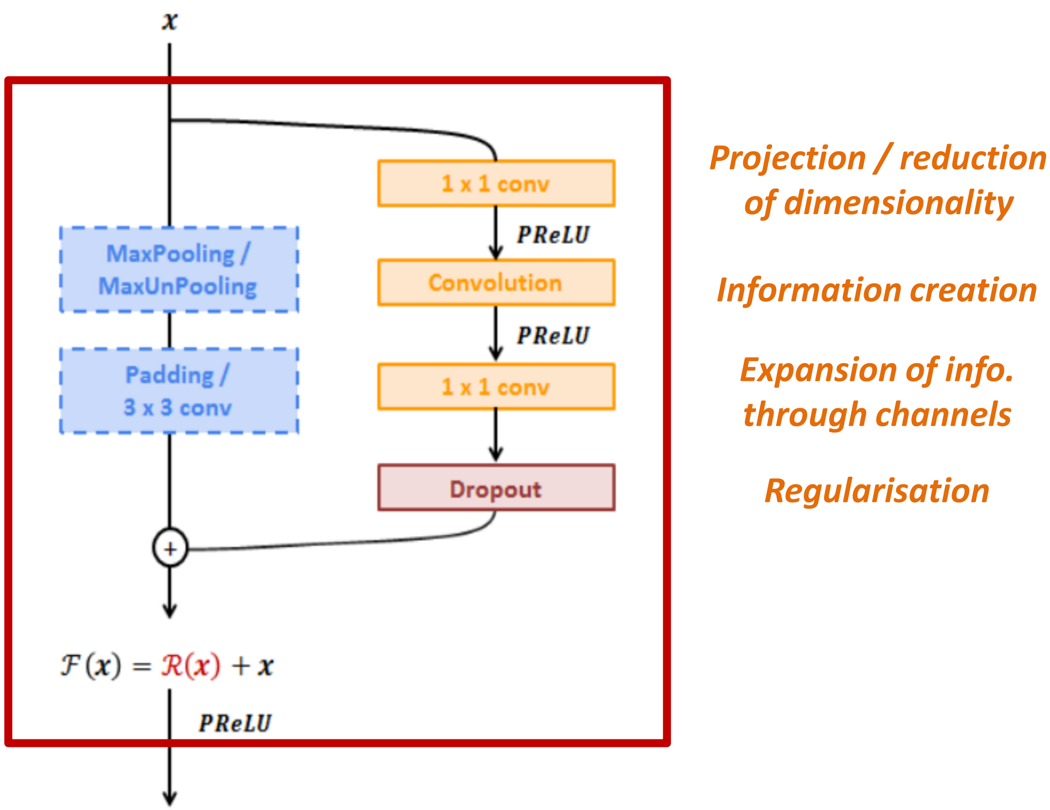


ENet

▶ ResNet-based architecture



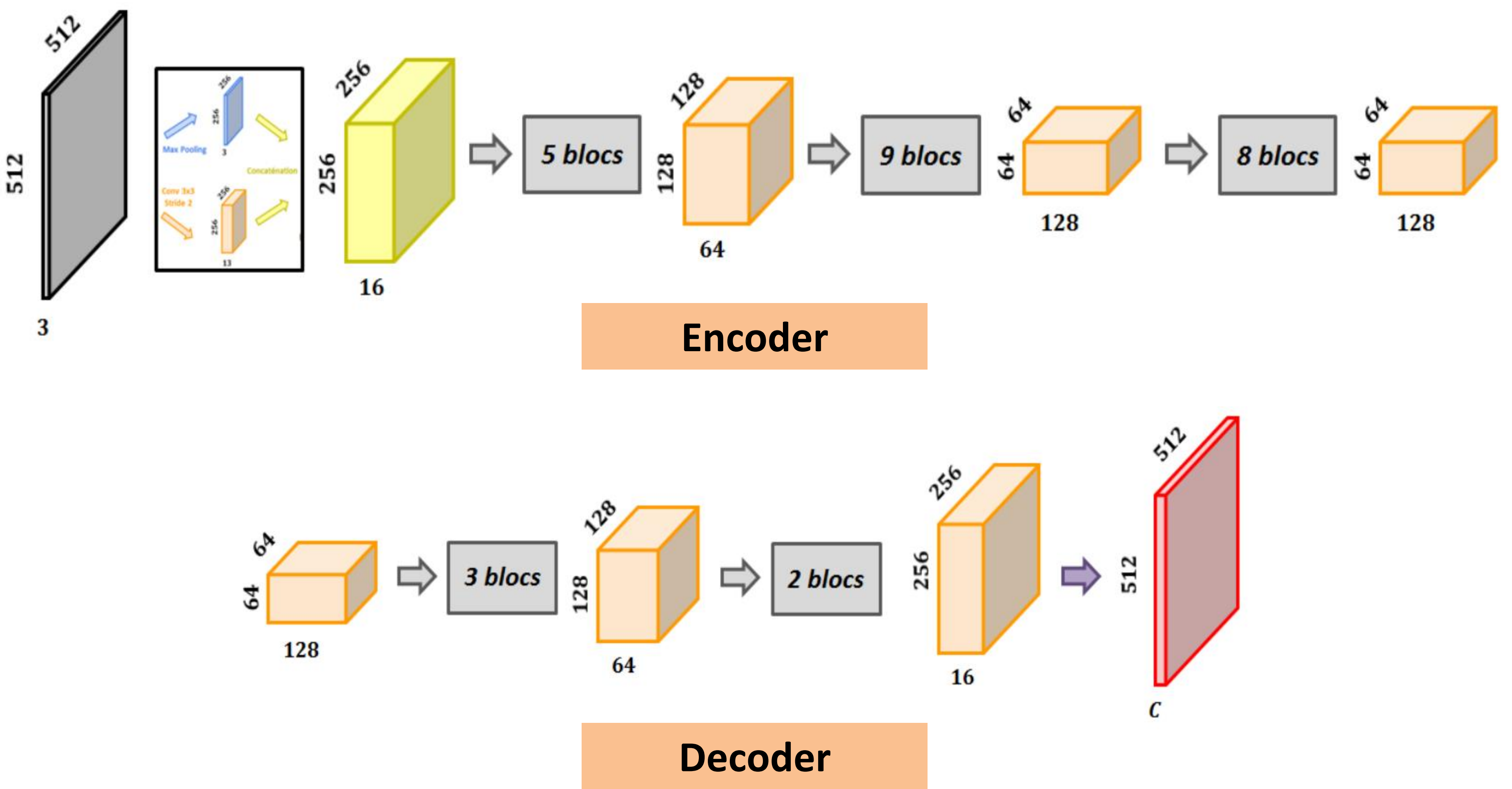
ResNet block



ENet block

ENet

- **Asymmetric architecture**



ENet

Performances

- Segmentation quality equivalent or better than the state-of-the art in deep learning
- # parameters: 0.37 M
- Network size < 6 MB
- Execution time (NVIDIA TitanX)

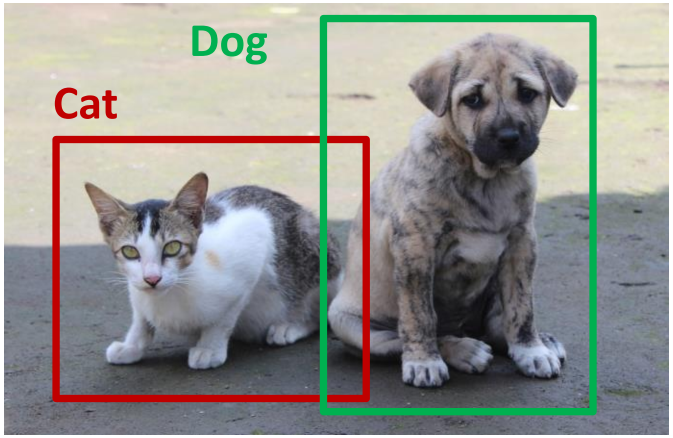
640x360 px	=>	7 ms
1280x720 px	=>	21 ms
1920x1080 px	=>	46 ms

Applications

Object detection

Object detection

Input image



Detected classes

Find the objects/classes present in an image and their location

R-CNN (Region-CNN)

- ▶ **Training stage**

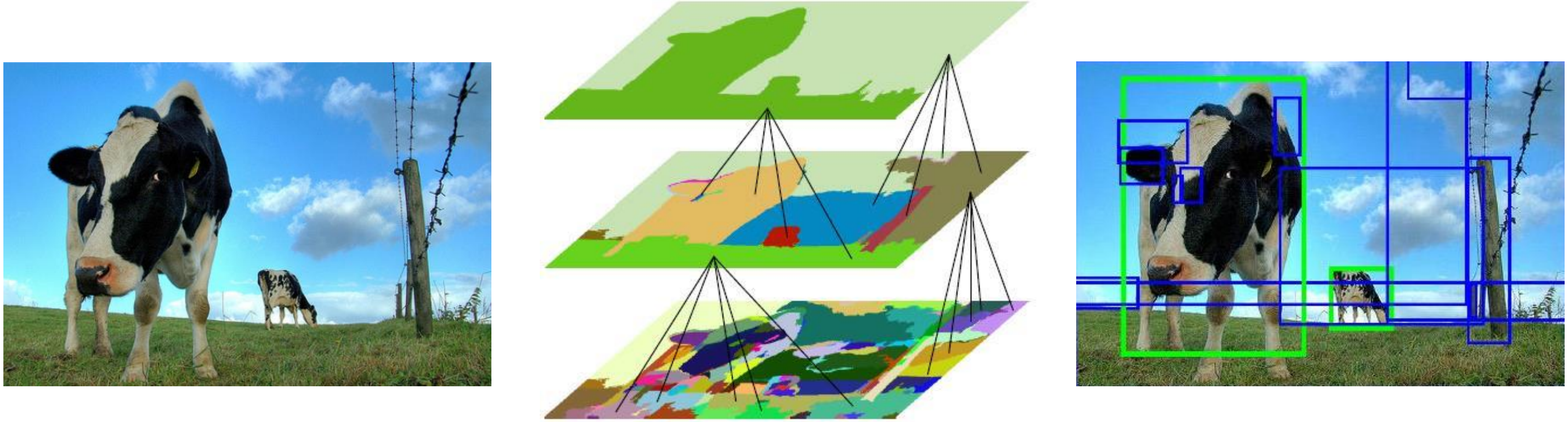
Input image | 2000 Region proposals | Region padding | Feature extraction (vector of 4096 dimensions) | Output data

Several training stage (CNN, SVM, regression for bounding box)

[Girshick, CVPR, 2014]

R-CNN (Region-CNN)

▶ Region extraction

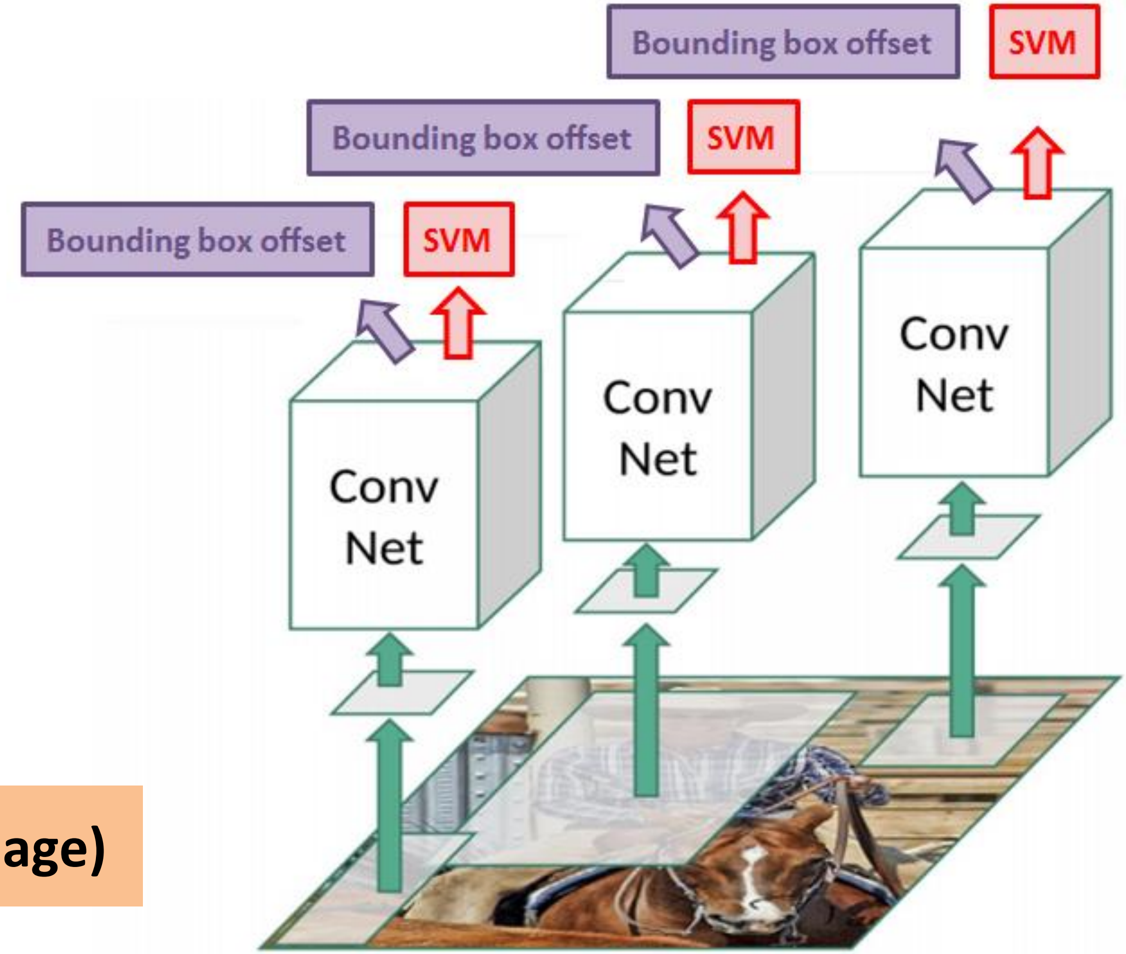


- Classical method using graph

Significant generation of bad candidates

R-CNN (Region-CNN)

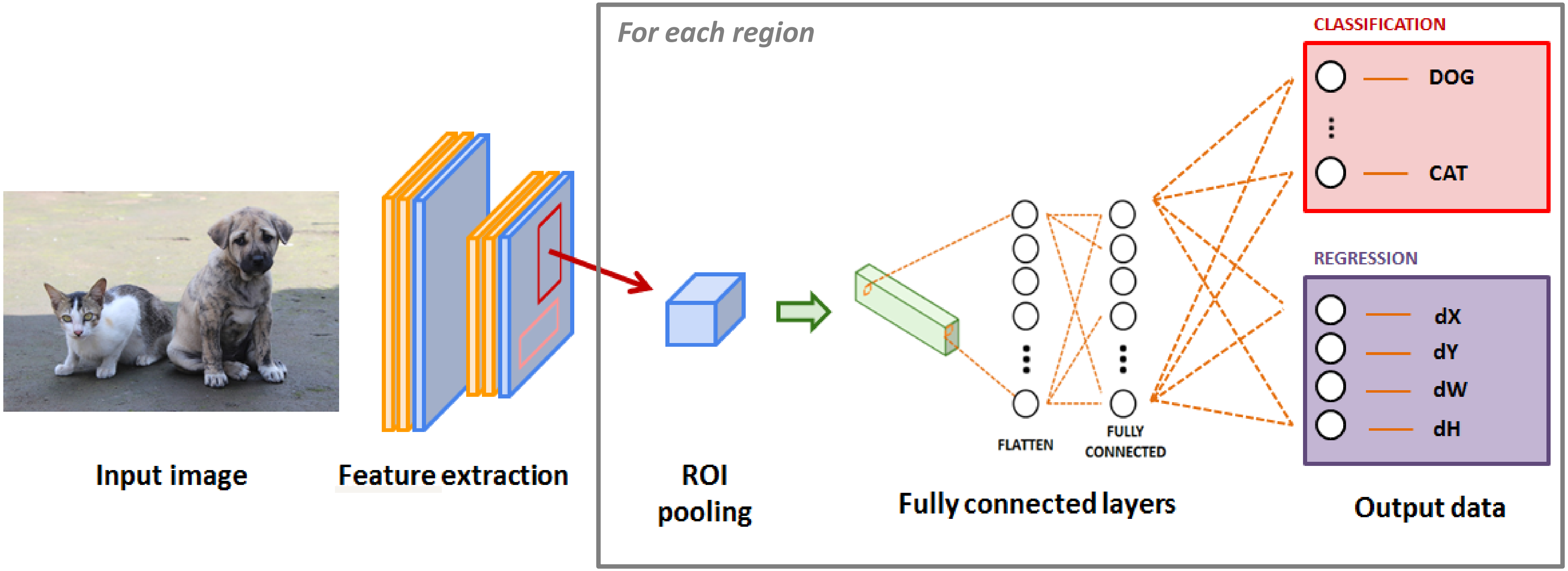
▶ **Inference stage**



- **Extraction of the region proposal**
- **Inferences over 2000 regions**

Very slow ($\sim 50\ s$ par image)

Fast R-CNN

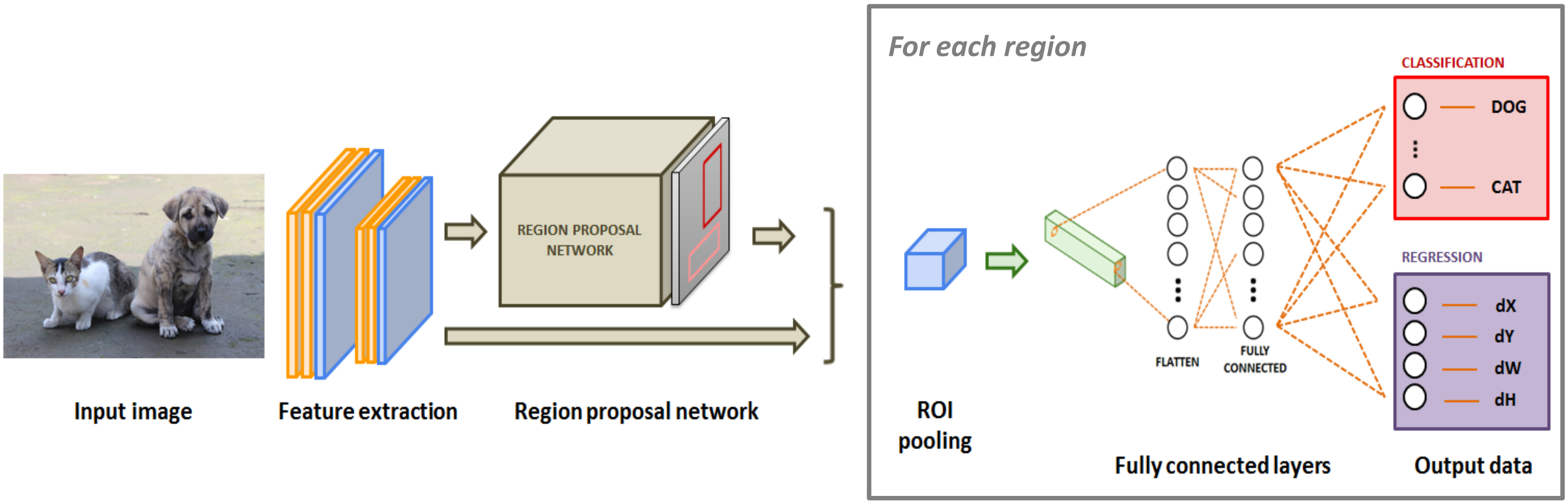


- **20x faster than R-CNN during inference !**

Extraction of the region proposals remains a weak point of the method

Faster R-CNN

[Ren, NIPS, 2017]

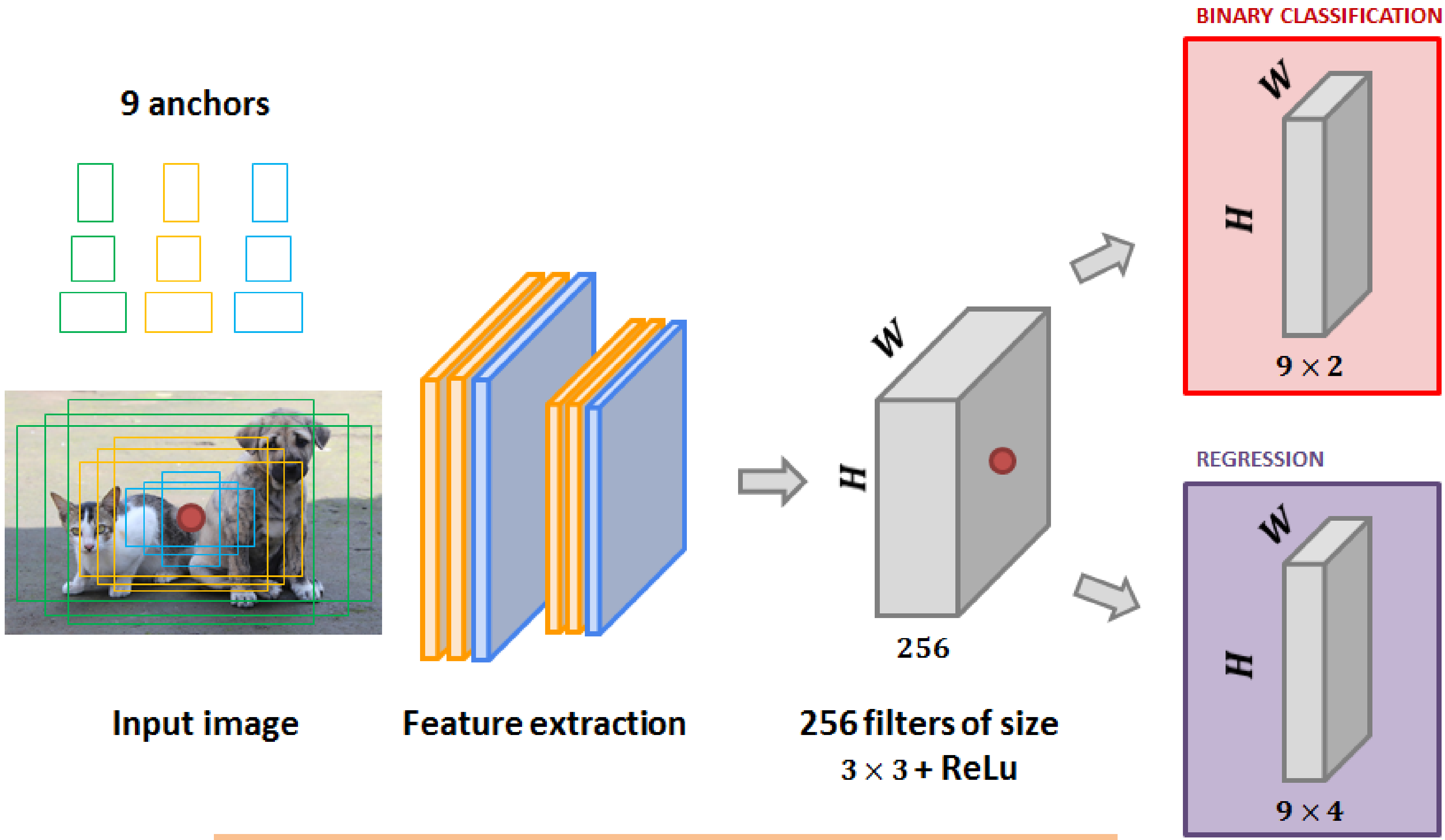


- **Integration of a region proposal network**

End-to-end trainable network !

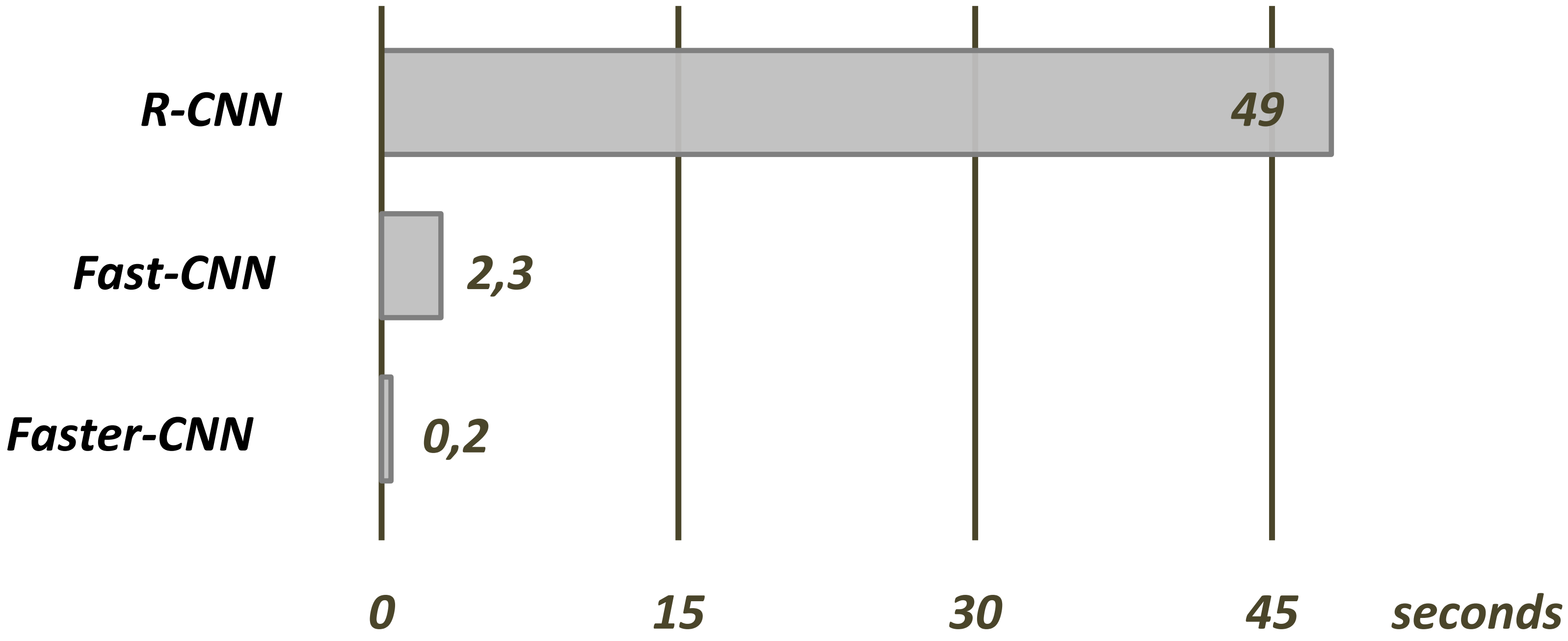
Region Proposal Network (RPN)

[Ren, NIPS, 2017]



$W \times H \times 9$ **detected regions**

Execution performance (inference stage)

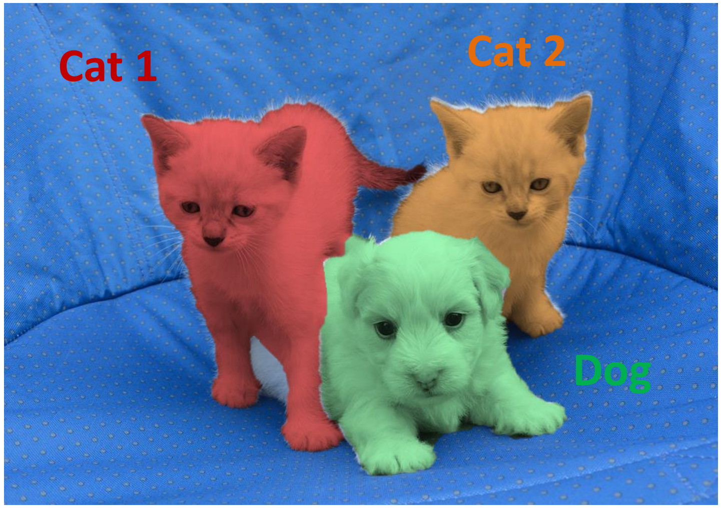


Applications

Instance segmentation

Instance segmentation

Input image

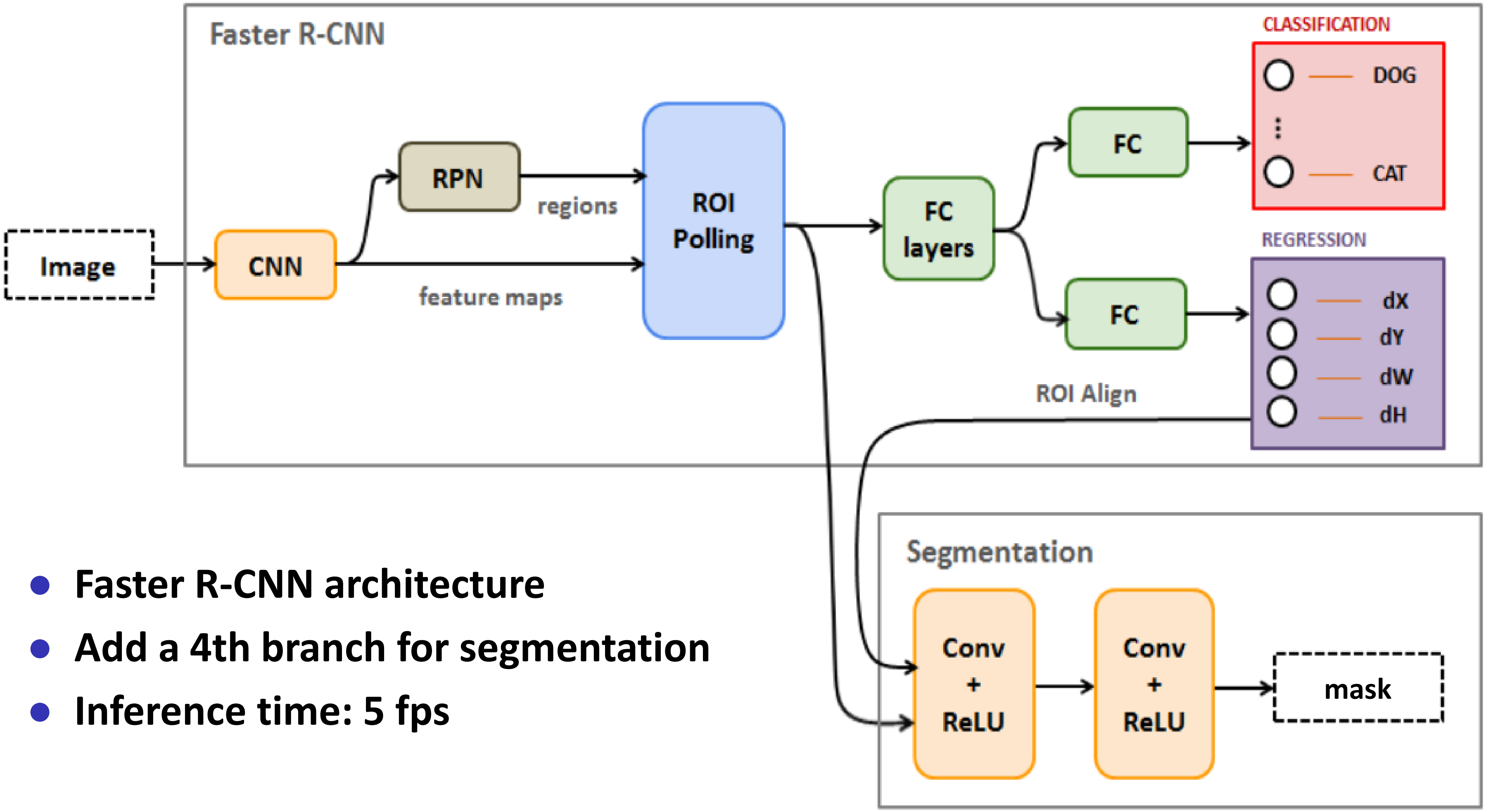


Segmented result

Detects and segments all instances of objects/classes present in an image

Mask R-CNN

[He, ICCV, 2017]



- **Faster R-CNN architecture**
- **Add a 4th branch for segmentation**
- **Inference time: 5 fps**

Mask R-CNN

- **Example of application – self-driving car**

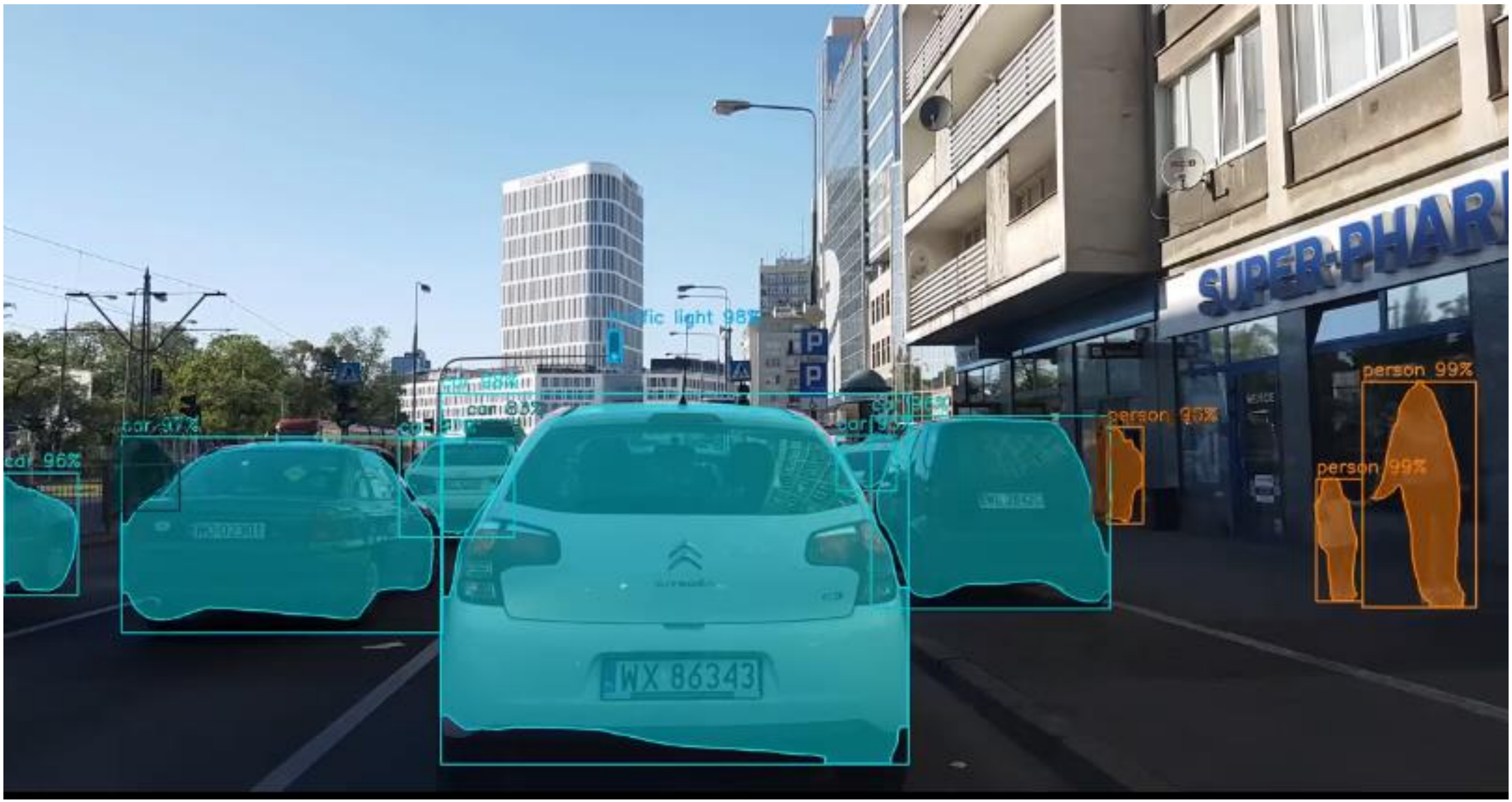
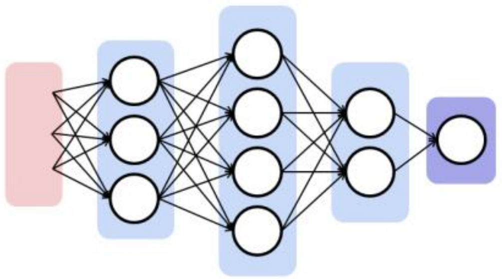


Illustration of the use of CNNs in medical application

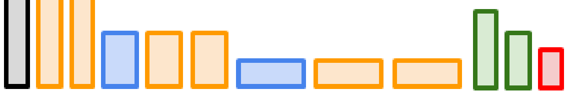
Deep learning in cardio-vascular imaging

Many methods have been successfully applied so far

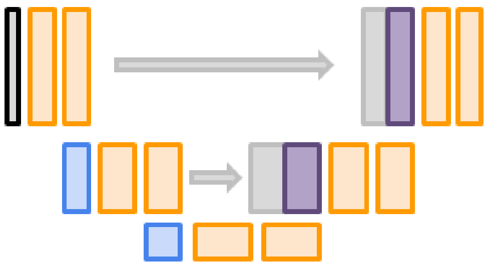
▶ Algorithms



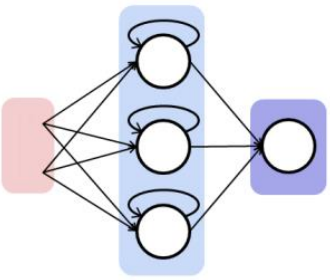
Fully-connected neural network



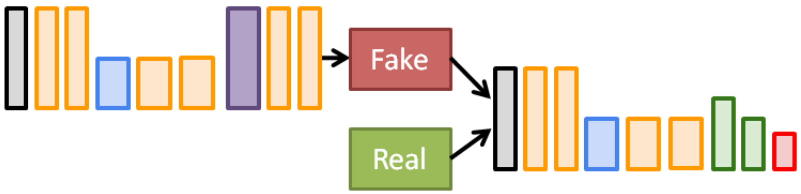
Convolutional neural networks



Fully convolutional neural networks



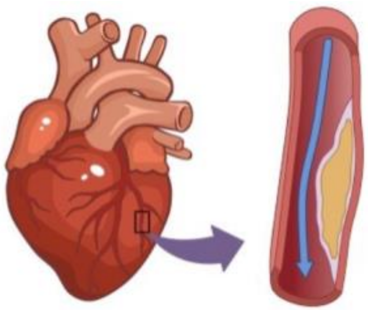
Recurrent neural network



Generative adversarial network

Many methods have been successfully applied so far

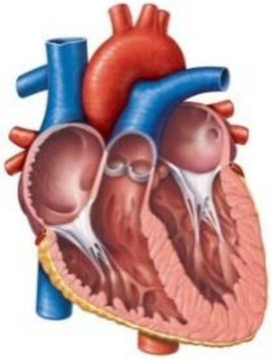
▶ Applications



Plaque risk assessment

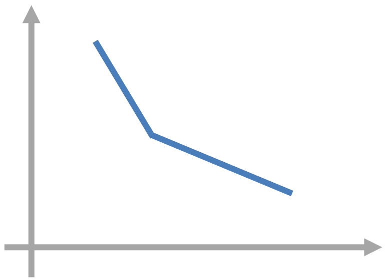
Calcium Score	Risk
0	Very low
1-99	Low
100-399	Moderate
> 400	High

Calcium scoring



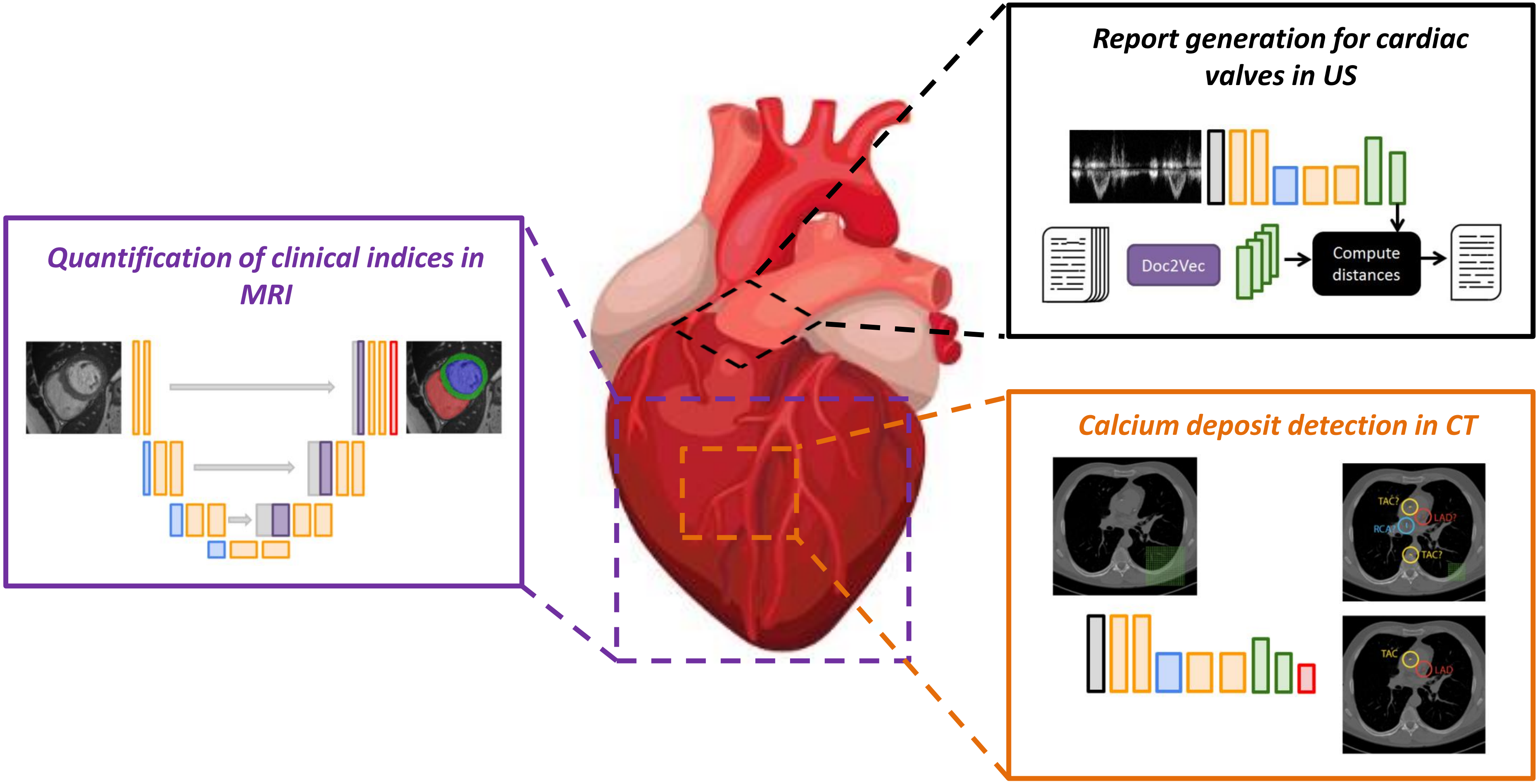
Ejection fraction estimation

Content-based image retrieval



CT dose reduction

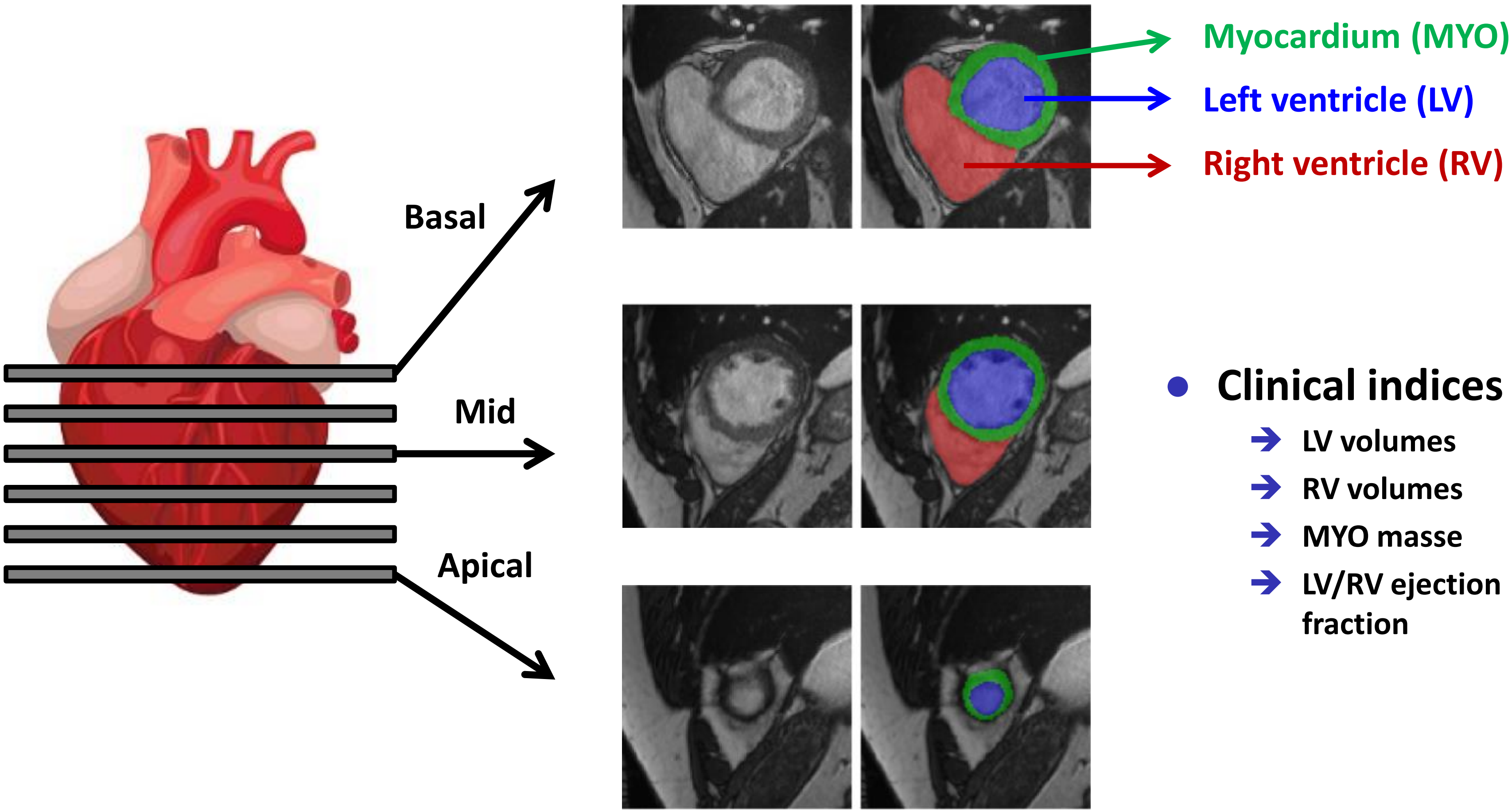
Many methods have been successfully applied so far



Automatic quantification of cardiac volumes and clinical indices

Segmentation of cardiac structures

▶ MR imaging



- **Clinical indices**
  - → LV volumes
  - → RV volumes
  - → MYO masse
  - → LV/RV ejection fraction

Segmentation of cardiac structures

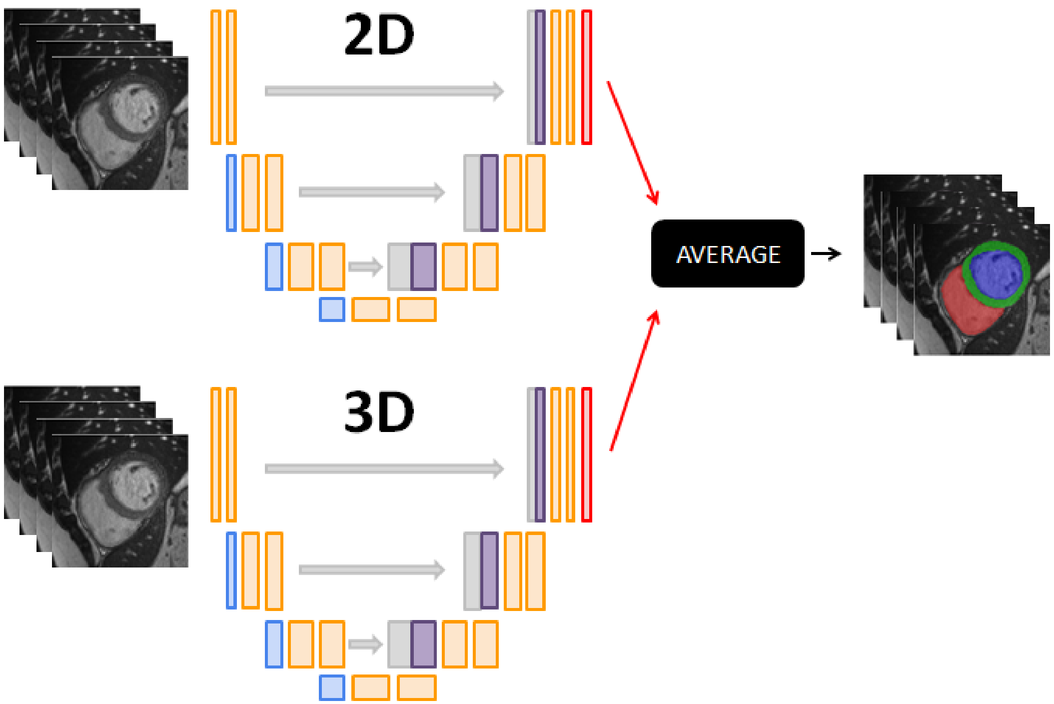
- **Important literature**
- **Several open access datasets with online evaluation platform**
- **Capacity to compare and still improve methods**
- **Information on the inter / intra observer variability**

CMRI datasets										
Name	**Year**	**Nb Subjects**		**Ground truth**				**Genericity**		**Online evaluation**
		train	test	LV	RV	Myo	Pathology	× Centre	× Vendor	
Sunnybrook	2009	45	—	✔	✘	✔	✔	✘	✘	✘
STACOM	2011	100	100	✔	✘	✔	✘	✘	✘	✘
MICCAI RV	2012	16	32	✘	✔	✘	✘	✘	✘	✘
Kaggle	2015	500	200	✘	✘	✘	✘	✘	✘	✘
ACDC	2017	100	50	✔	✔	✔	✔	✘	✘	✔
M&Ms	2020	150	200	✔	✔	✔	✔	✔	✔	✘

[Bernard, IEEE TMI, 2018]

Ensemble U-Net segmentation method

▶ **One of the current best performing methods on ACDC dataset**



- **Cross-entropy loss**
  - ➔ **3D U-Net**
- **Multiclass Dice loss**
  - ➔ **2D U-Net**

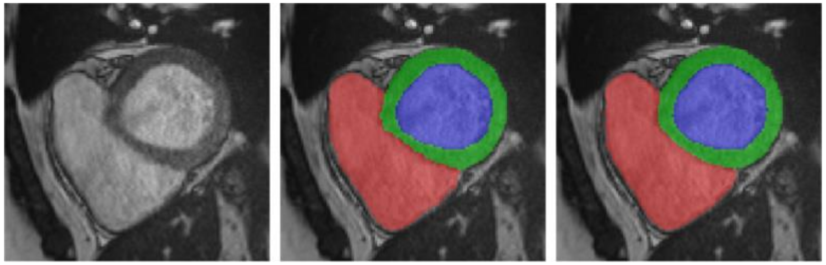
[Isensee, Miccai, 2017]

Ensemble U-Net segmentation method

Anatomical metrics

Methods	Left Ventricle Haus. dist. (mm)	Right Ventricle Haus. dist. (mm)	Myocardium Haus. dist. (mm)
Inter-observer	7,1	13,2	7,4
Intra-observer	4,7	8,4	5,6
Isensee et al.	6,2	9,9	7,2

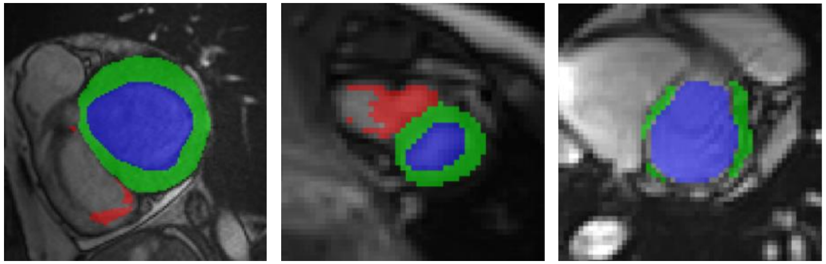
High segmentation quality



Clinical metrics

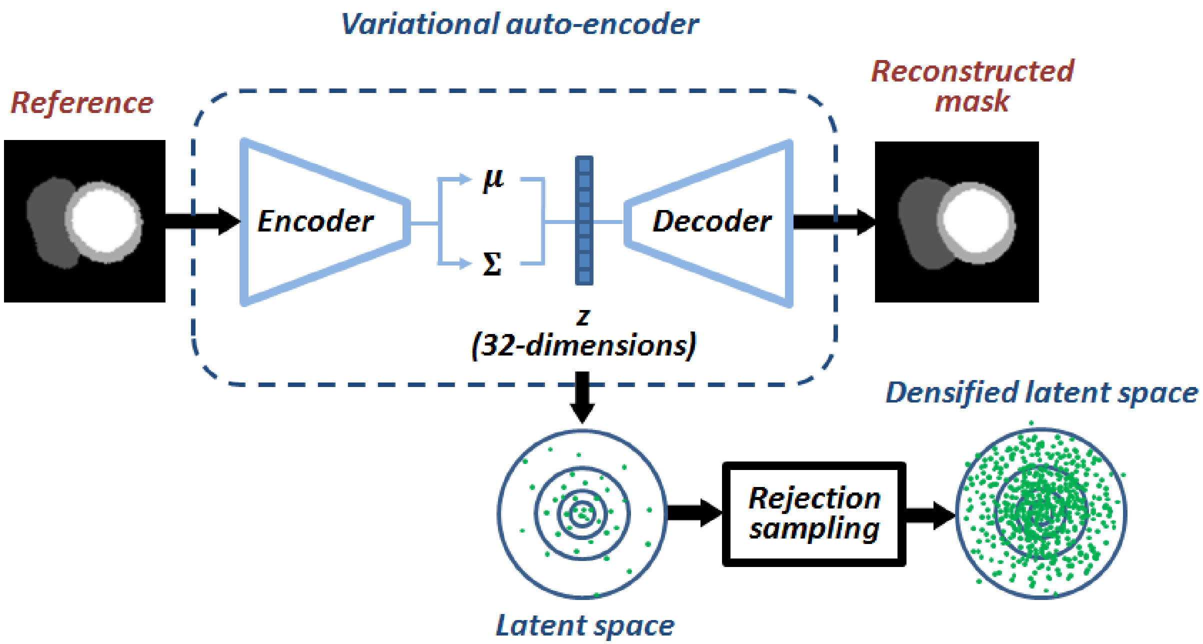
Methods	LV Eject. Fract. Correlation	RV Ejec. Frac. Correlation	Myo. Mass. Correlation
Isensee et al.	0,997	0,910	0,987

With few incoherence



How to guarantee anatomical coherence of the segmentation results ?

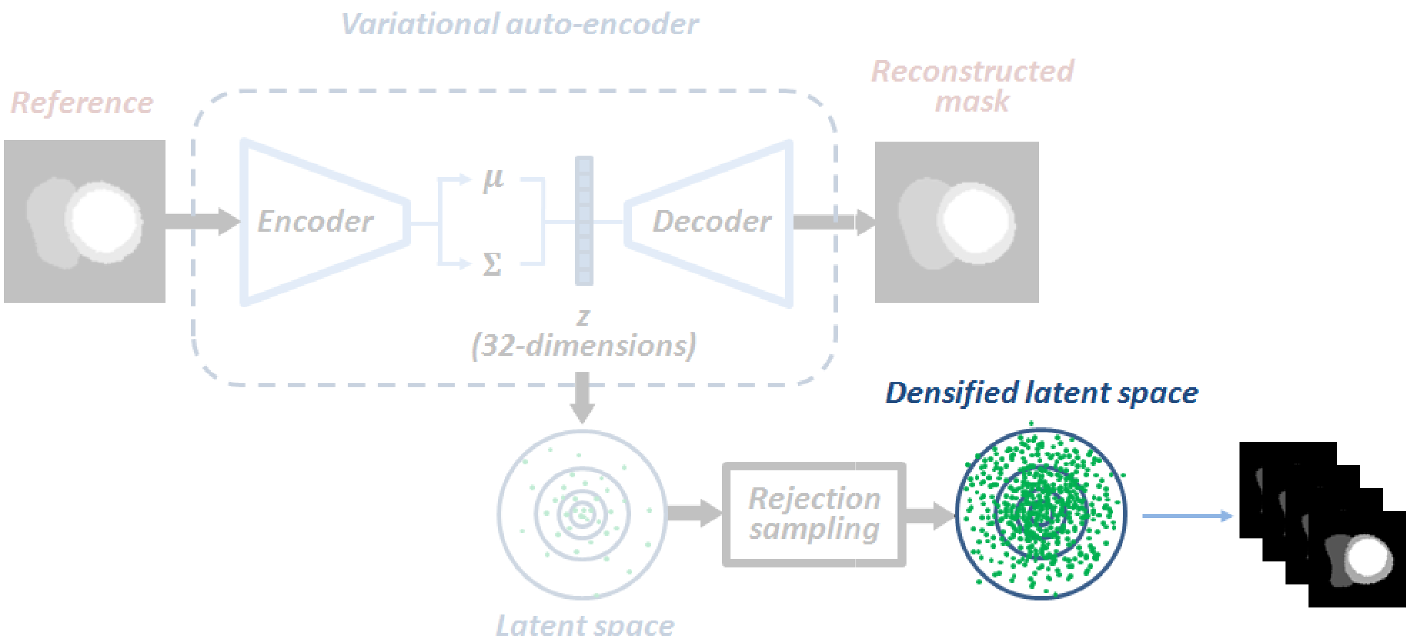
Cardiac segmentation with strong anatomical guarantees



Efficient encoding of anatomical shapes in a latent space

[Painchaud, IEEE TMI, 2020]

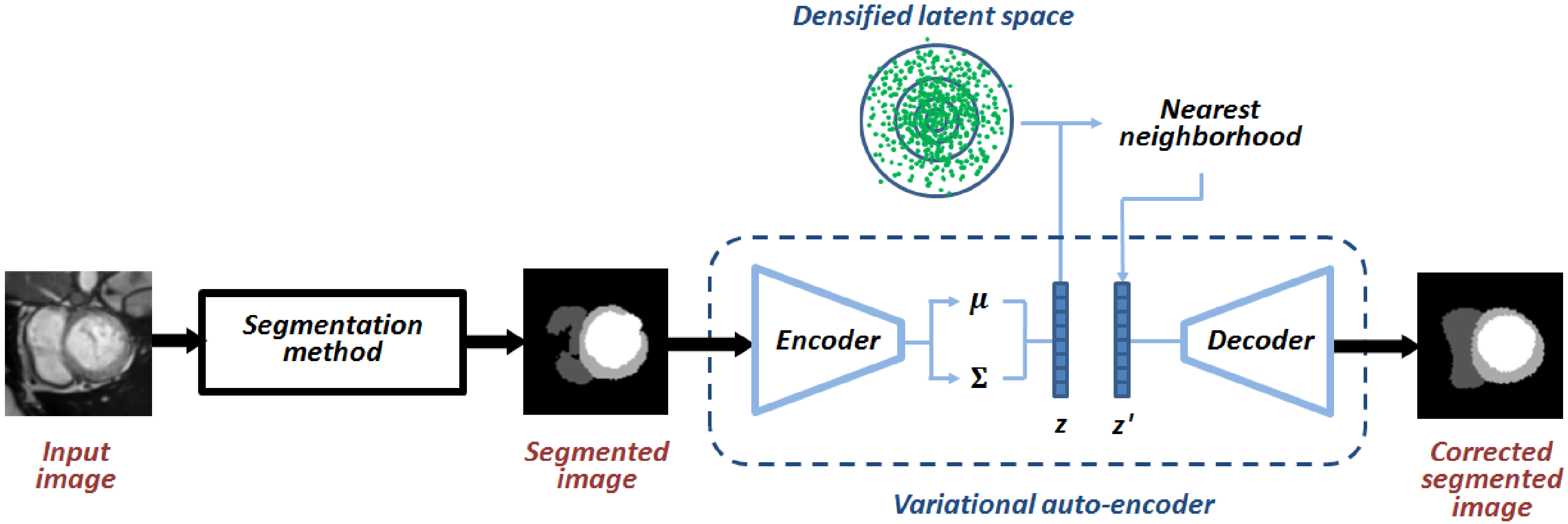
Cardiac segmentation with strong anatomical guarantees



16 anatomical metrics

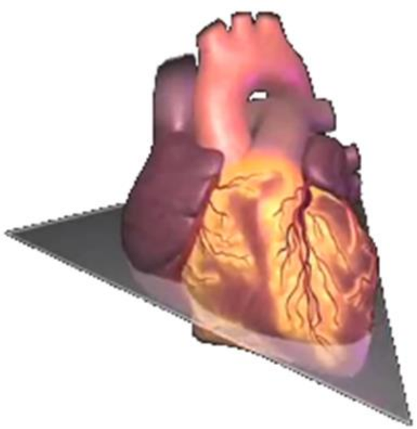
Cardiac segmentation with strong anatomical guarantees

Correction of segmentation to guarantee the plausibility of anatomical shapes → Almost same accuracy than the original methods but with correct anatomical shapes



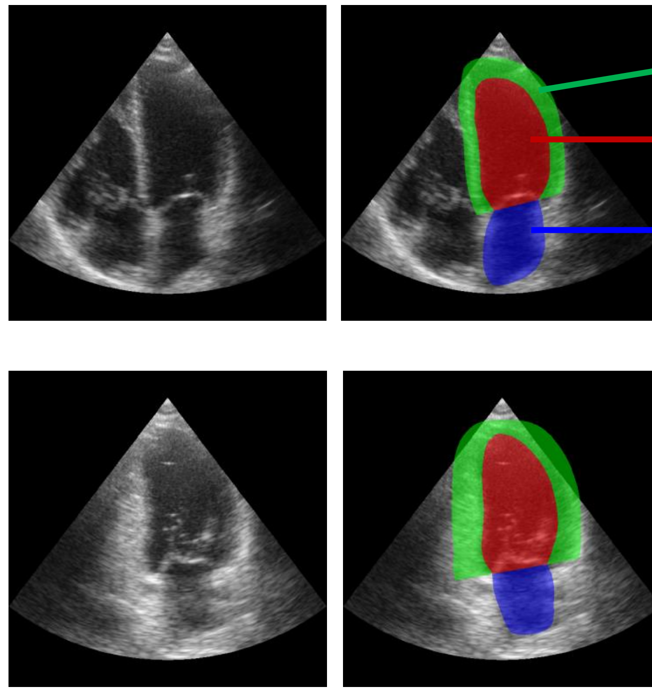
Segmentation of cardiac structures

▶ US imaging



Apical 4 chambers view

Apical 2 chambers view



- **Clinical indices**
  - ➔ LV volumes
  - ➔ LV ejection fraction

Segmentation of cardiac structures

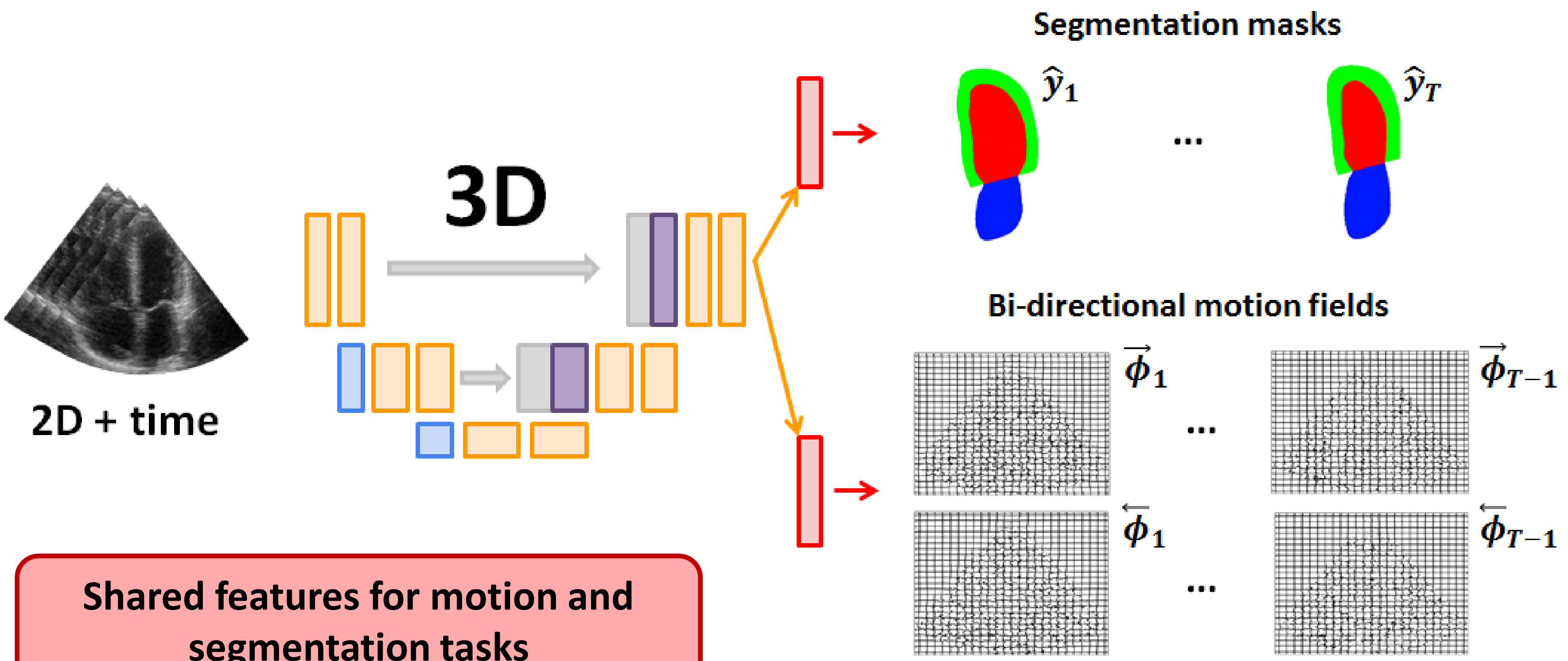
- Less literature compared to MRI
- Few open access datasets with online evaluation platform
- Capacity to compare and still improve methods
- Information on the inter / intra observer variability

Echocardiographic datasets												
Name	Year	Nb Subjects		Ground truth				View		Genericity		Online evaluation
		train	test	LV_{endo}	LV_{epi}	LA	Pathology	A4C	A2C	× Centre	× Vendor	
CETUS	2014	15	30	✔	✘	✘	✔	✔	✔	✔	✔	✘
CAMUS	2019	450	50	✔	✔	✔	✔	✔	✔	✘	✘	✔
EchoNet	2019	10036	—	✔	✘	✘	✔	✔	✘	✘	—	✘

[Leclerc, IEEE TMI, 2019]

Temporal-consistent segmentation method

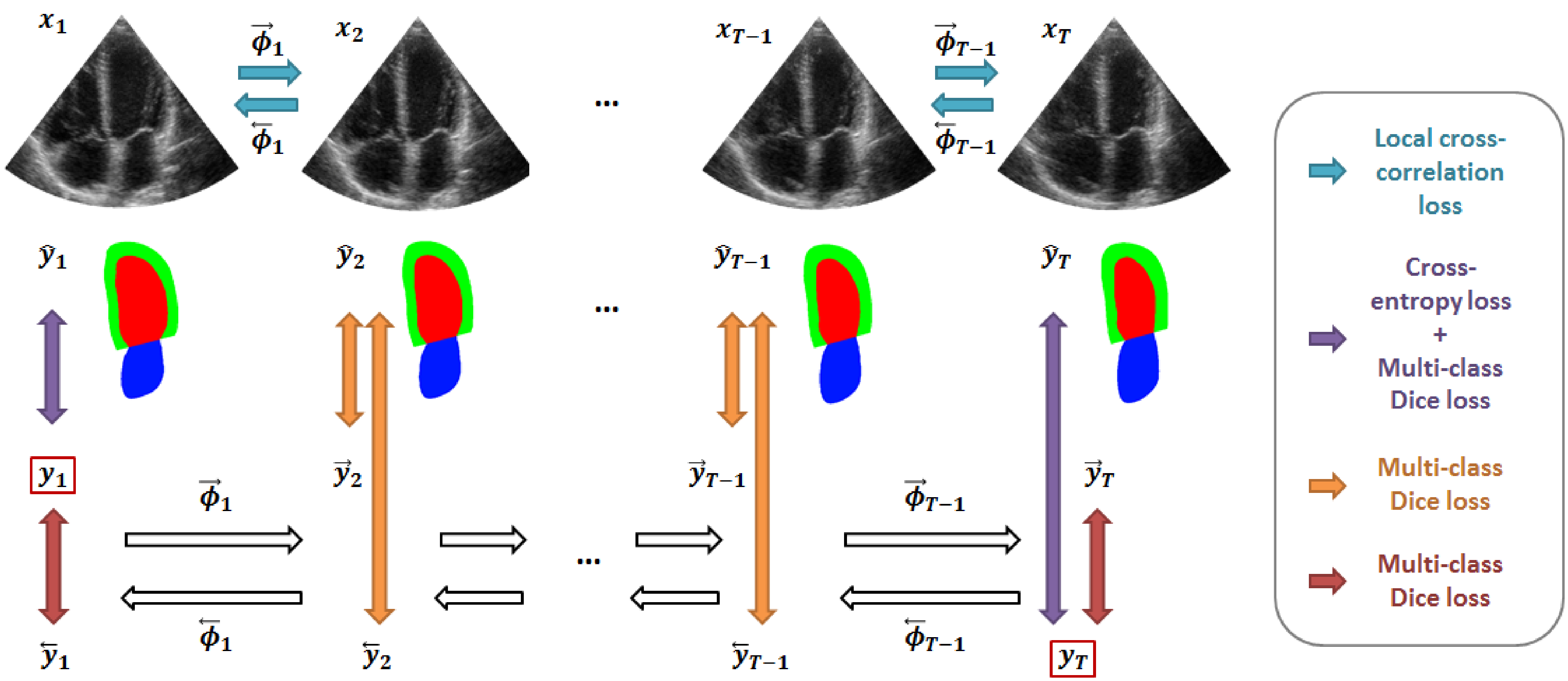
▶ **One of the current best performing methods on CAMUS dataset**



Shared features for motion and segmentation tasks

[Wei, Miccai, 2020]

Temporal-consistent segmentation method



Temporal-consistent segmentation method

Anatomical metrics

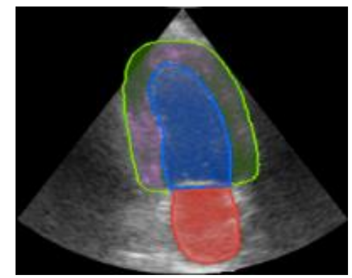
Methods	LV endocardium Haus. dist. (mm)	LV epicardium Haus. dist. (mm)	Left atrium Haus. dist. (mm)
Inter-observer	7,1	7,5	-
Intra-observer	4,6	5,0	-
Wei *et al.*	4,6	4,9	5,0

Clinical metrics

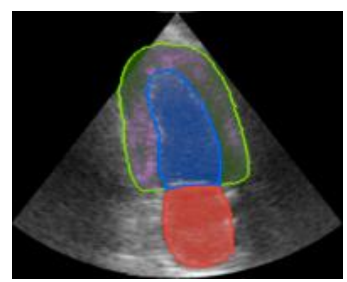
Methods	LV volume ED Correlation	LV volume ES Correlation	LV Eject. Fract. Correlation
Inter-observer	0,940	0,956	0,801
Intra-observer	0,978	0,981	0,896
Wei *et al.*	0,958	0,979	0,926

High segmentation quality with temporal consistency

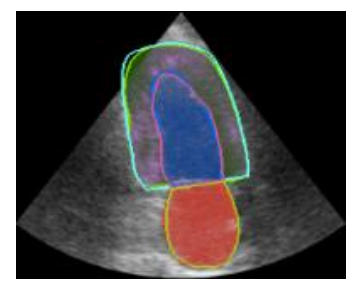
ED (t=1)



t=4



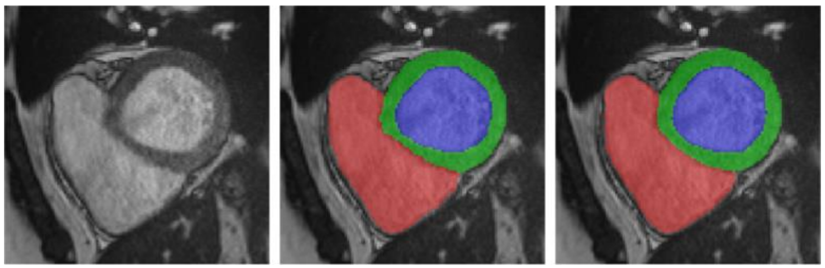
t=7

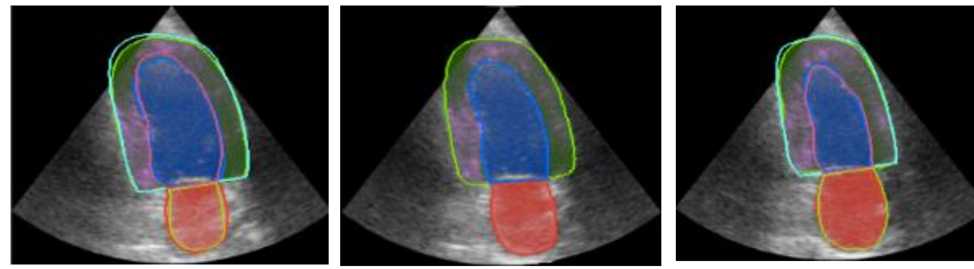


ES (t=11)

Automatic quantification of cardiac volumes

▶ Is the problem solved ?



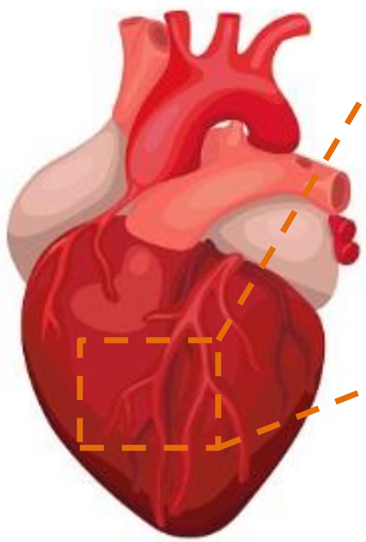


1. **Needs for the clinicians to reinforce the annotation recommendations**
2. **Validation on complete (>1000) datasets with multi-centers / multi-vendors**

Calcium deposit detection in low dose chest CT scans

Calcium deposit detection in low dose chest CT scans

▶ CT imaging



- Low dose chest CT scans acquired for lung cancer screening enable
  - → Quantification of atherosclerotic calcification
  - → Identification of subjects at increased cardiovascular risk

Possibility to complement lung cancer screening programs to help identify subjects at elevated cardiovascular risk without the need for further imaging !

[Lessmann, IEEE TMI, 2017]

Calcium deposit detection in low dose chest CT scans

Dataset

- 3D CT scans from 1744 patients
- Multi-centers
  - → 31 medical centers
- Multi-vendors
  - → 13 different scanner models
- Inter-observer assessment
  - → subset of 100 scans
  - → annotation from 3 experts

Annotations

- Manually labeled
  - → distributed among 5 experts
- Calcifications segmented in
  - → Coronary arteries
  - → Aorta
  - → Aortic and mitral valves
- Time spent
  - → 5-10 min. for easy cases
  - → 60-90 min. for difficult cases

Calcium deposit detection in low dose chest CT scans

- 2 steps CNN approaches with varying receptive field

  - First network

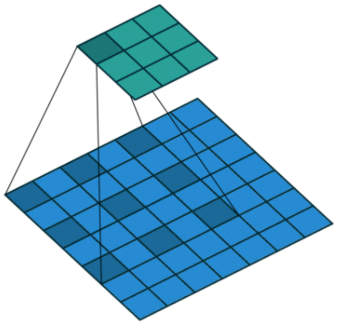
    - Large receptive field (RF)

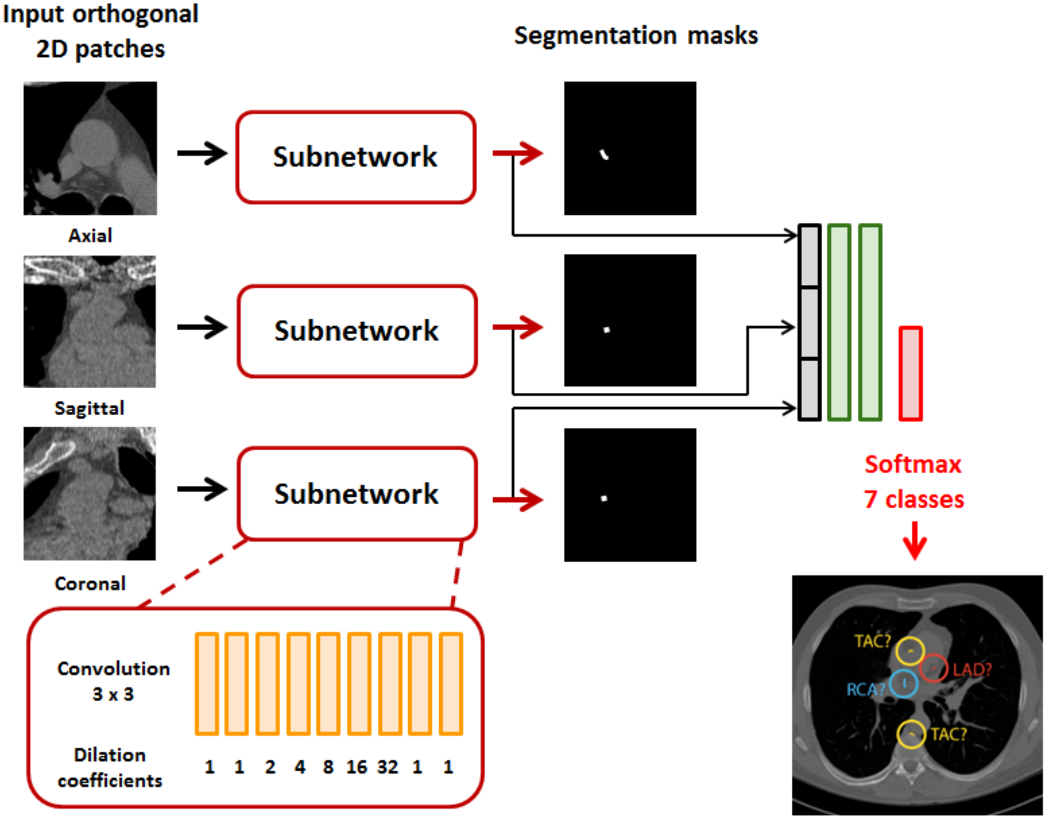
      Patches: 155 px²

      RF: 131 px²

    - Cross entropy losses

    - Increasing dilation coefficient





Calcium deposit detection in low dose chest CT scans

- ▶ 2 steps CNN approaches with varying receptive field

  - **Second network**

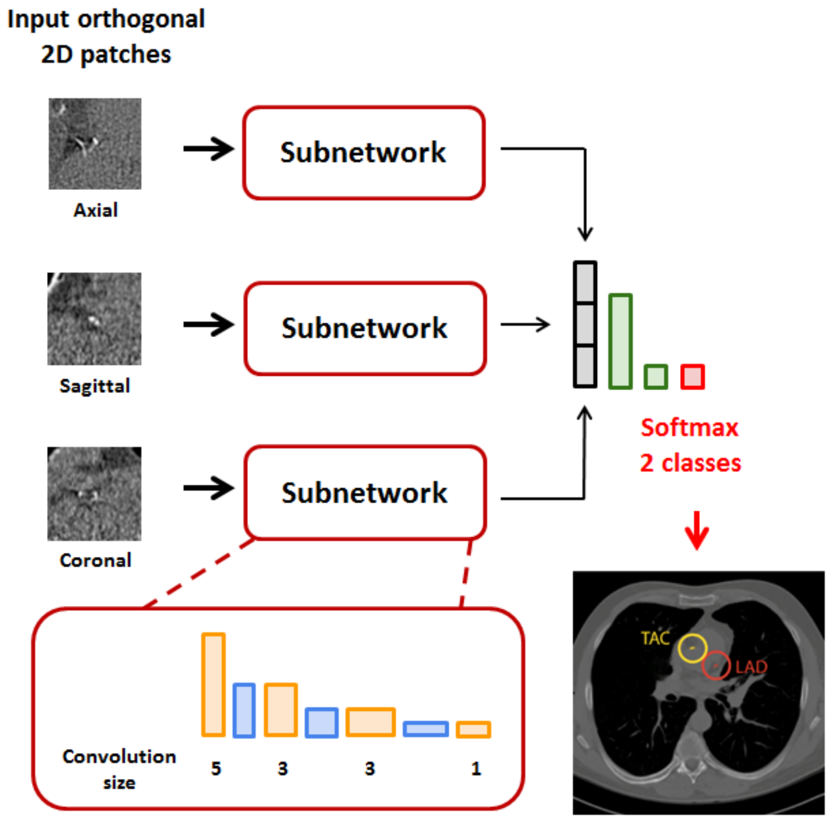
    - ➔ Classify true positive and false positive from 1st CNN results

    - ➔ Smaller receptive field (RF)

      Patches: 65 px²

      RF: 65 px²

    - ➔ Single cross entropy loss



Calcium deposit detection in low dose chest CT scans

▶ Overall performance

Classification results

	CAC	LAD	LCX	RCA
Inter-observer				
Sensitivity (%)	95	93	84	96
False positive vol. (mm^3)	21	10	9	12
F_1 *score calcium*	0.95	0.94	0.87	0.96
Automatic method				
Sensitivity (%)	91	92	72	91
False positive vol. (mm^3)	35	18	14	11
F_1 *score calcium*	0.90	0.90	0.72	0.90

Confusion matrix

Calcium score categories

I ➔ 0-10 II ➔ 11-100

III ➔ 101-1000 IV ➔ > 1000

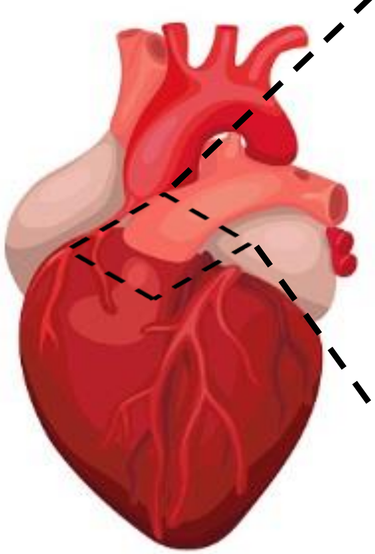
	Automatic method			
Reference	I	II	III	IV
I	**90**	17	1	0
II	3	**59**	4	0
III	**0**	2	**99**	2
IV	**0**	0	1	**32**

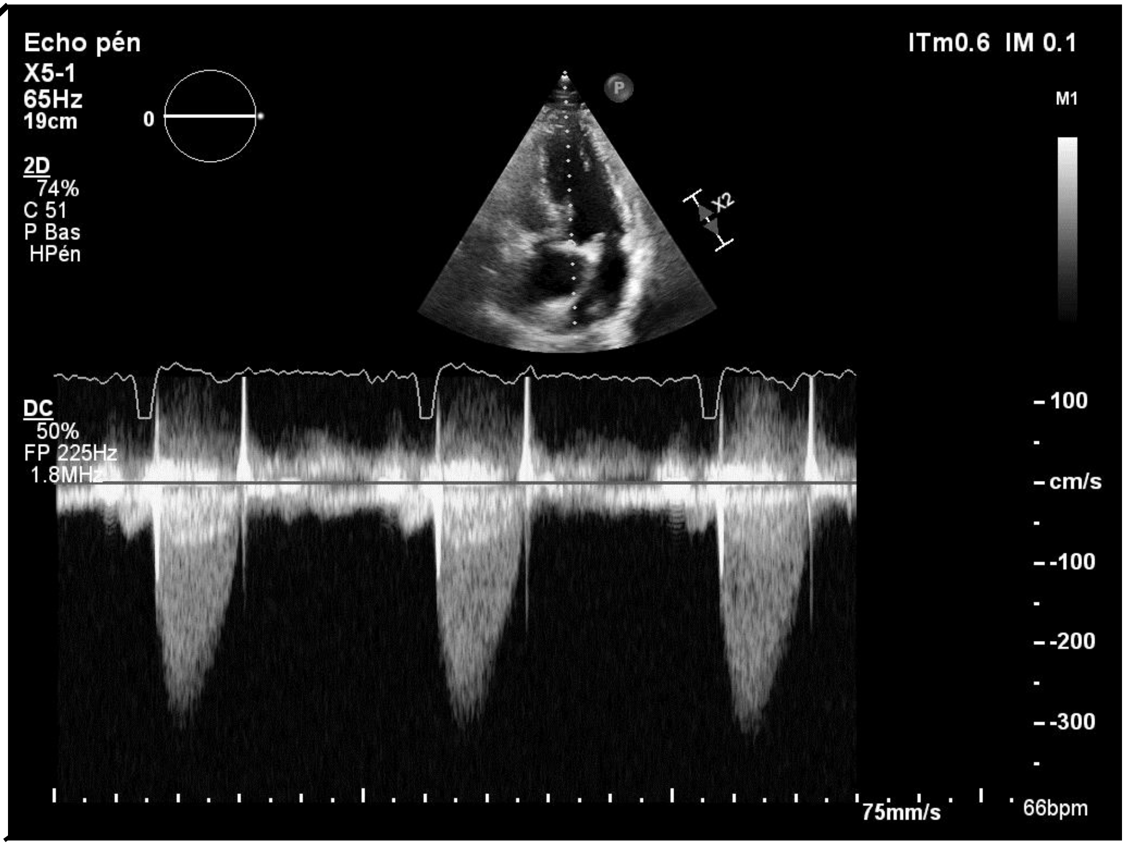
90% of agreements

Report generation of cardiac valves in US

Report generation for cardiac valves in US

▶ US imaging





Conventional Doppler exams

[Moradi, Miccai, 2016]

Report generation for cardiac valves in US

- Important source of information without annotation but with clinical reports

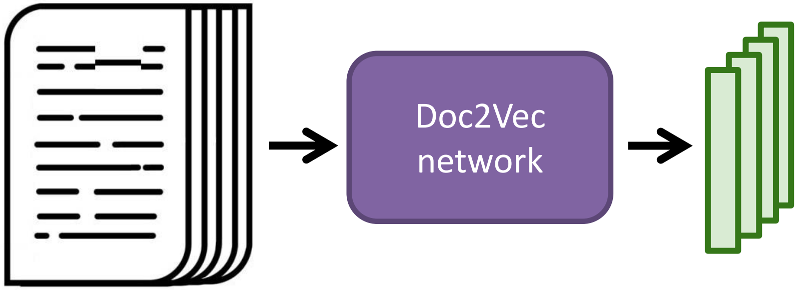
- Automatic labeling of semantic concepts
  - Imaged valve
  - Disease type
  - Severity

- How to automatically label this huge source of information in an a posteriori manner ?

Report generation for cardiac valves in US

Learning of a fixed length vector representation of text paragraph

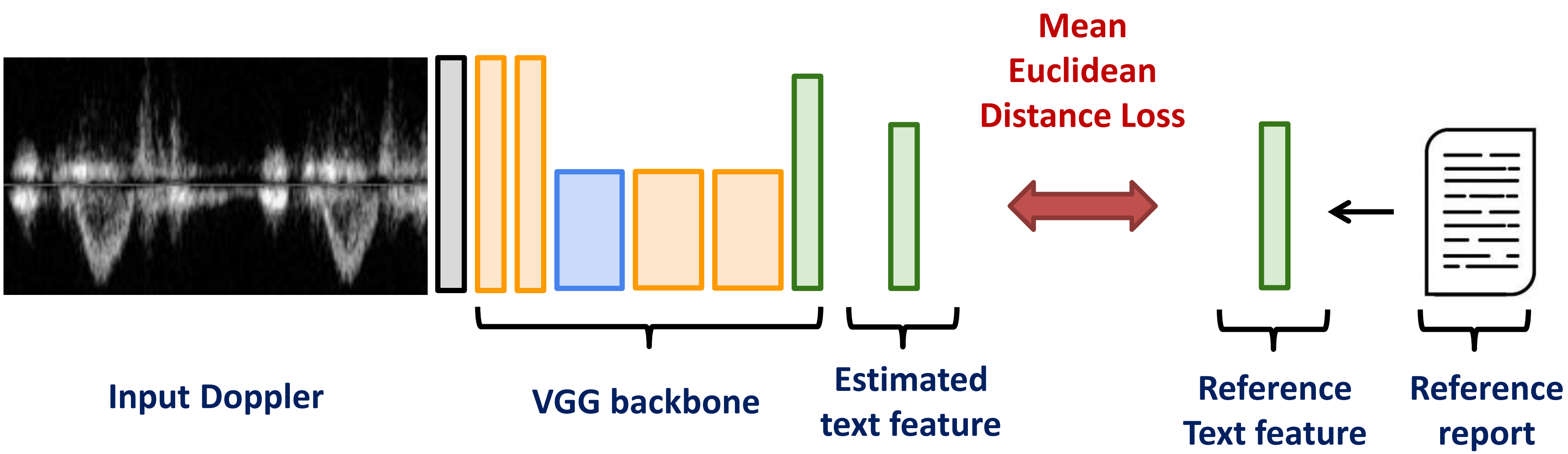
- **Input**
  - ➔ **10253 text paragraphs with valve labels from clinical reports**
- **Output**
  - ➔ **Text feature vector of size *m***



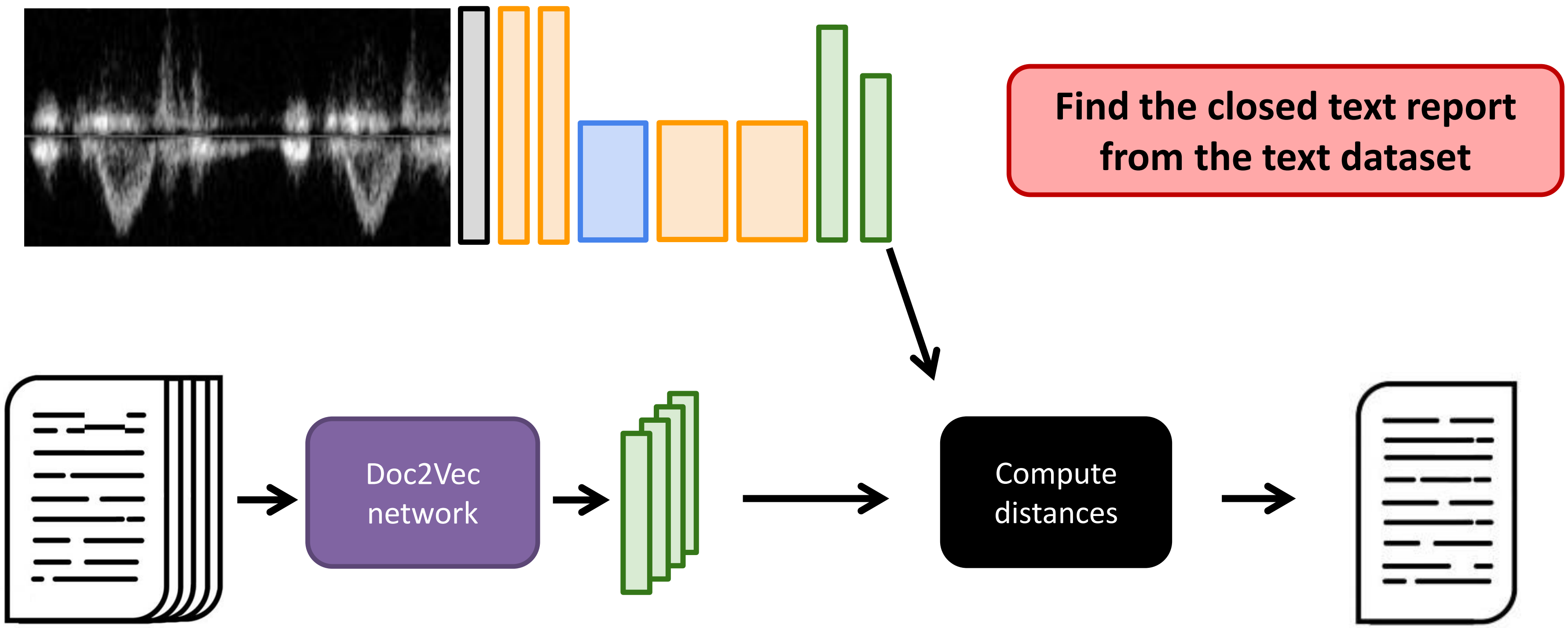
Report generation for cardiac valves in US

Transform network: from image to text feature

- **Learning phase**
  - ➔ 226 images and corresponding text reports

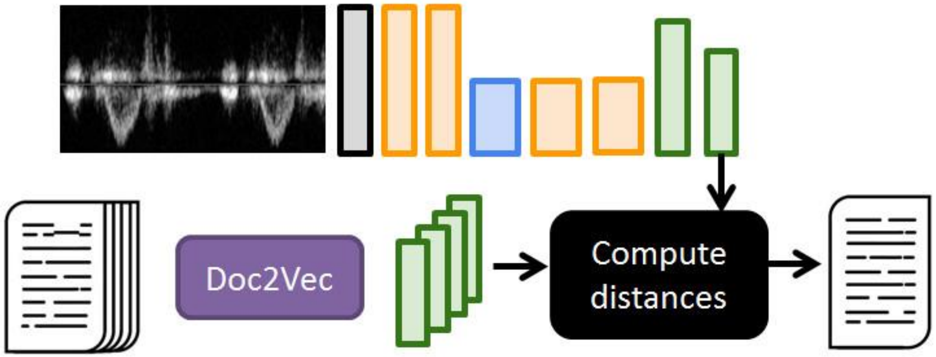


Report generation for cardiac valves in US



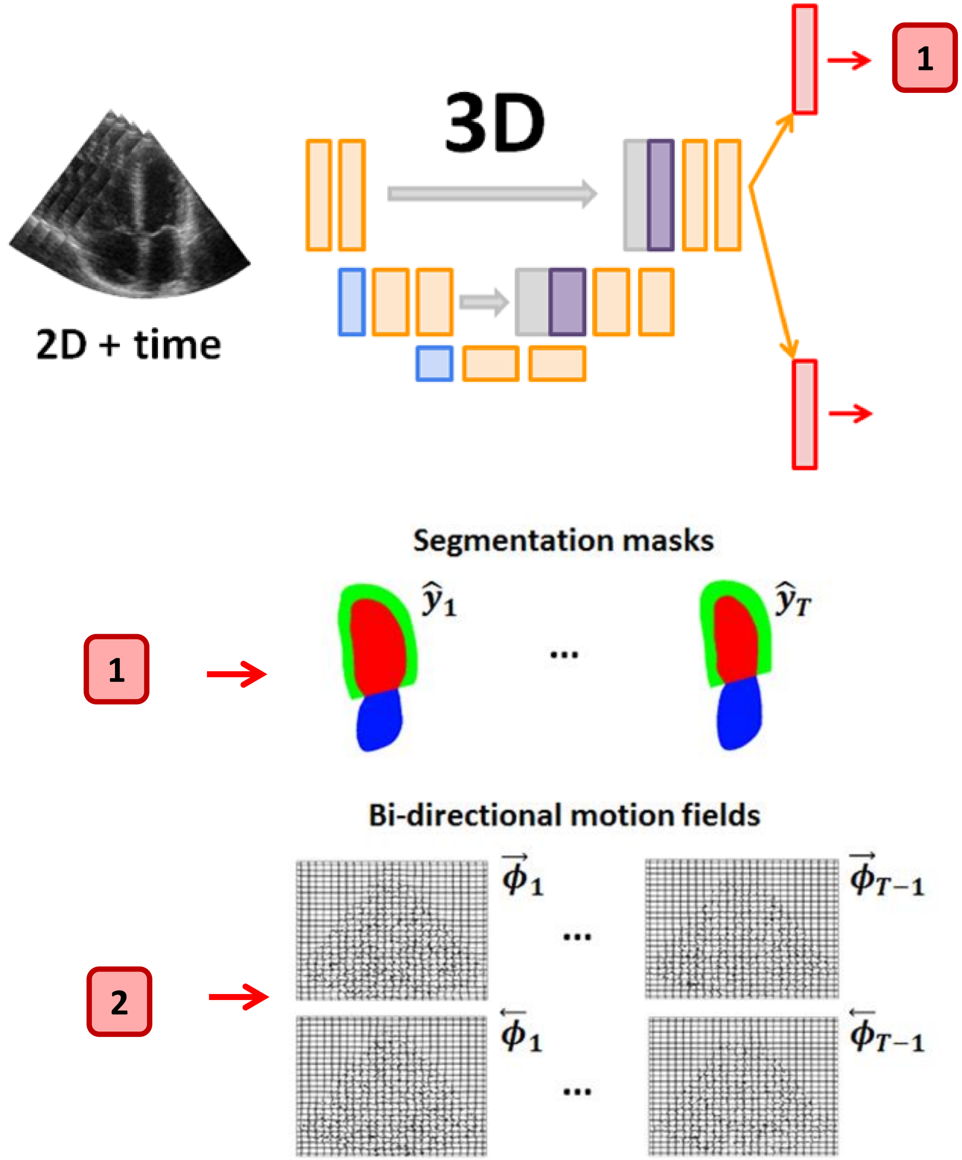
Report generation for cardiac valves in US

Extraction of semantic descriptors from the retrieved paragraphs

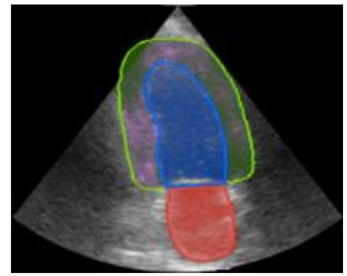


- **Automatic extraction of**
  - ➔ Valve type
  - ➔ Valve disease
  - ➔ Pathology severity
- **Performances**
  - ➔ Small validation on 48 tested samples
  - ➔ 91% of correct disease classification
  - ➔ 77% of correct disease severity classification

That’s all folks

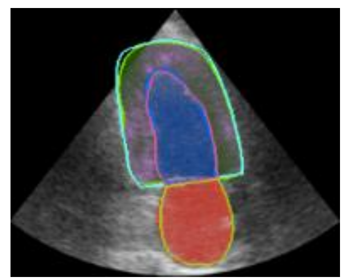

3D
2D + time
1
Segmentation masks
1
$\hat{y}_1$
...
$\hat{y}_T$
Bi-directional motion fields
2
$\vec{\phi}_1$
...
$\vec{\phi}_{T-1}$
$\overleftarrow{\phi}_1$
...
$\overleftarrow{\phi}_{T-1}$

ED (t=1)



t=4

t=7



ES (t=11)