



# Computer Vision και Deep Learning

Προχωρημένα Θέματα Επιστήμης  
και Αναλυτικής Δεδομένων

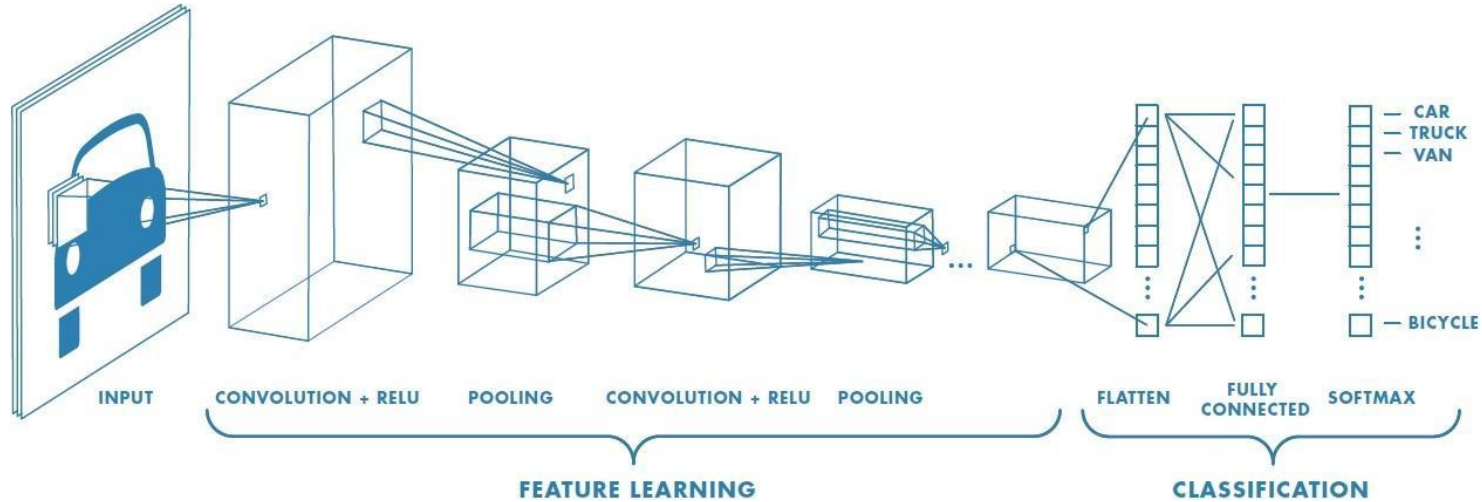
Παρασκευή Τζούβελη

# Computer Vision και deep learning

- [Convolutional Neural Network \(CNN\)](#)
- [Αρχιτεκτονικές CNN](#)
- [Transfer Learning](#)
- [Μετρικές απόδοσης](#)
- [Region-Based Convolutional Neural Network](#)

# Convolutional Neural Network

[Understanding of Convolutional Neural Network \(CNN\) – Deep Learning](#), Prabhu in Towards Data Science



# Convolutional Neural Network

## Convolution Layer

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Input

1	0	1
0	1	0
1	0	1

Filter / Kernel

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

4		

## Stride


Stride 1


Feature Map


Stride 2


Feature Map

## Padding

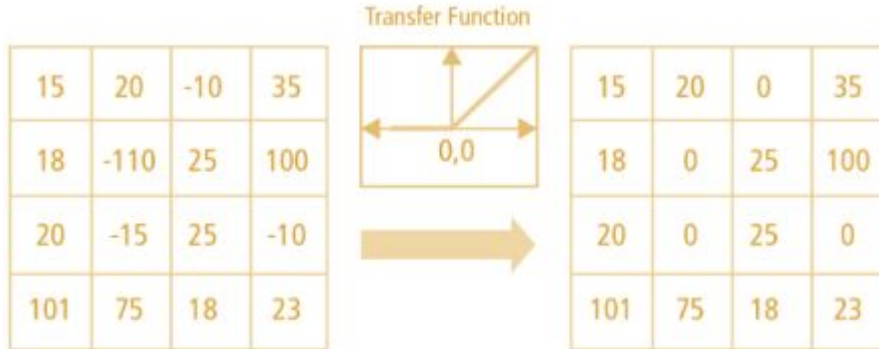

Stride 1 with Padding


Feature Map

# Convolutional Neural Network

ReLU: Rectified Linear Unit for a non-linear operation

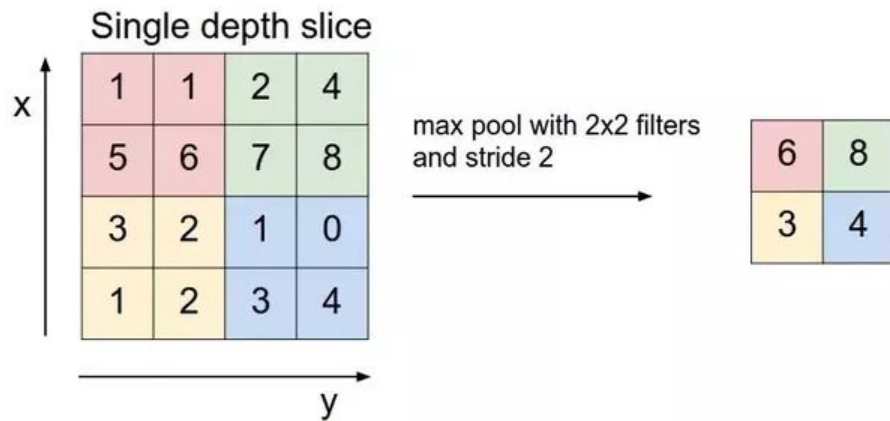
$$f(x) = \max(0, x).$$



# Convolutional Neural Network

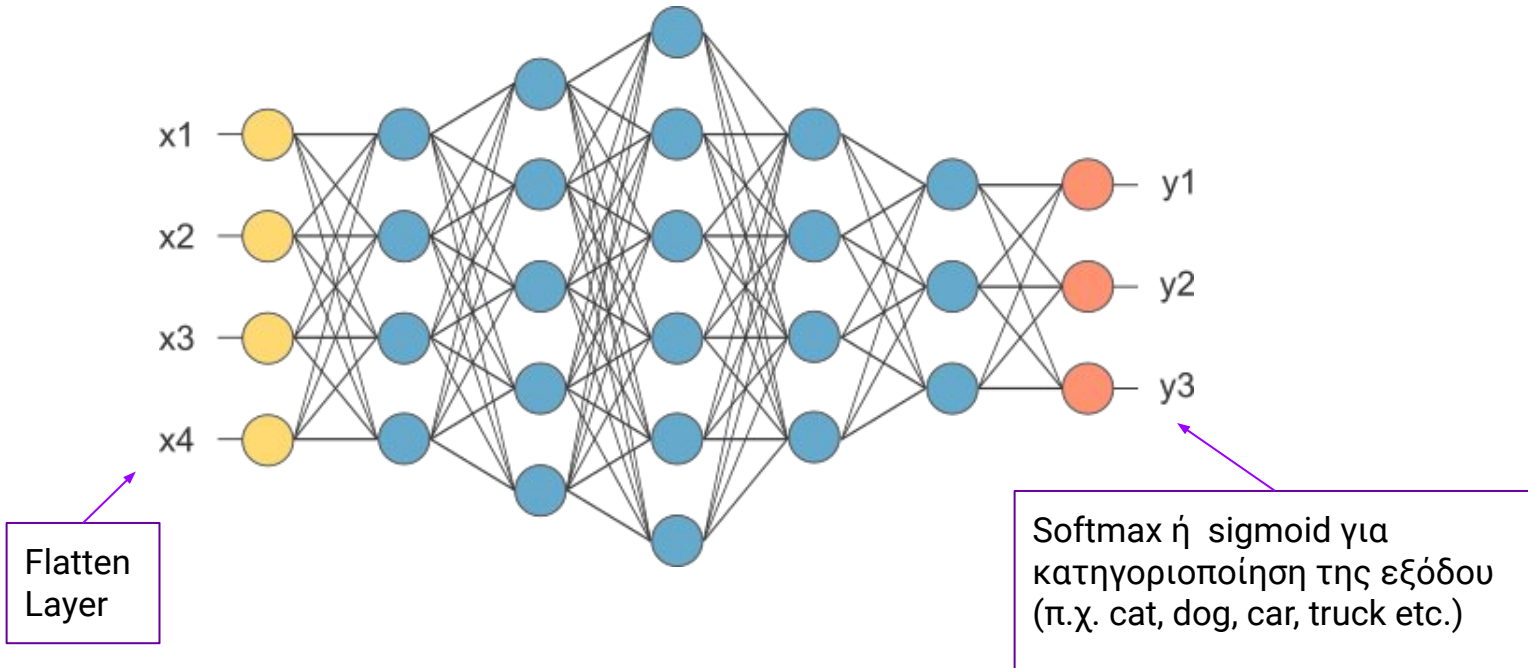
## Pooling Layer

- Max Pooling
- Average Pooling
- Sum Pooling



# Convolutional Neural Network

## Fully Connected Layer



# CNN Architectures: LeNet

[Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.](#)

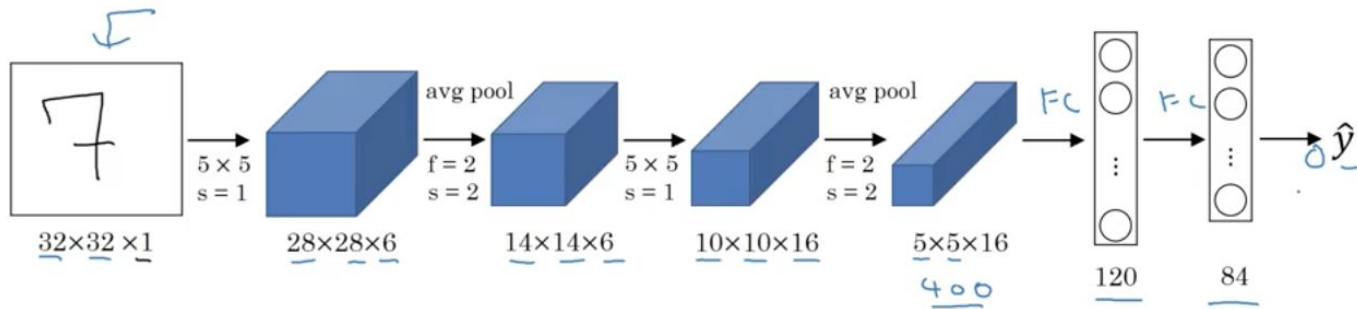
## LeNet - 5

W x H → 32 x 32 (Width x Height)

F(w x h) → 5 x 5 (Filter)

S → 1 (Stride)

P → 0 (Pooling)



$$\left( \frac{W - F_w + 2P}{S_w} \right) + 1 = \left( \frac{32 - 5 + 0}{1} \right) + 1 = 27 + 1 = 28$$

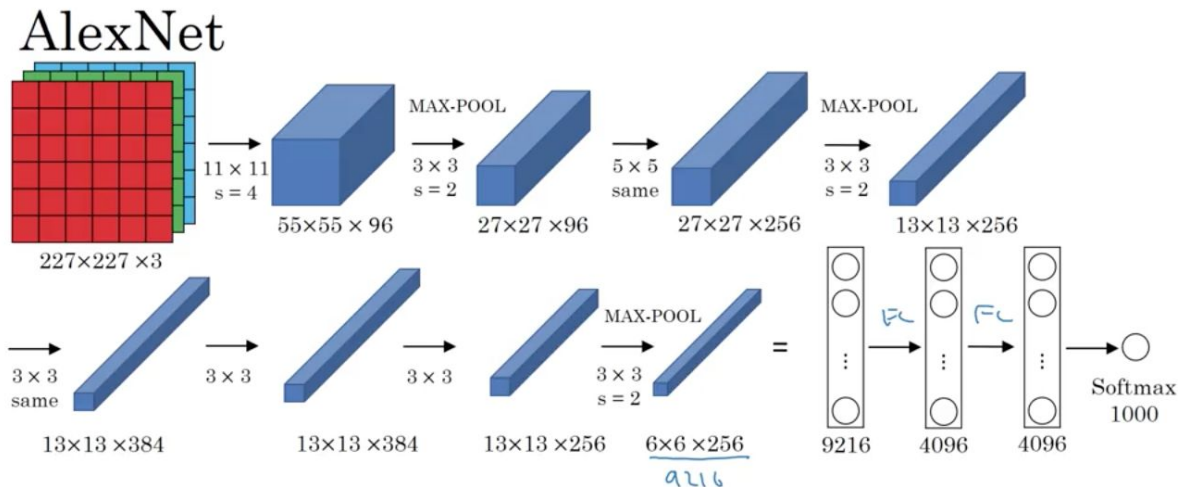
$$\left( \frac{H - F_h + 2P}{S_h} \right) + 1 = \left( \frac{32 - 5 + 0}{1} \right) + 1 = 27 + 1 = 28$$

$$\text{Output Volume} = 28 \times 28$$



# CNN Architectures: AlexNet

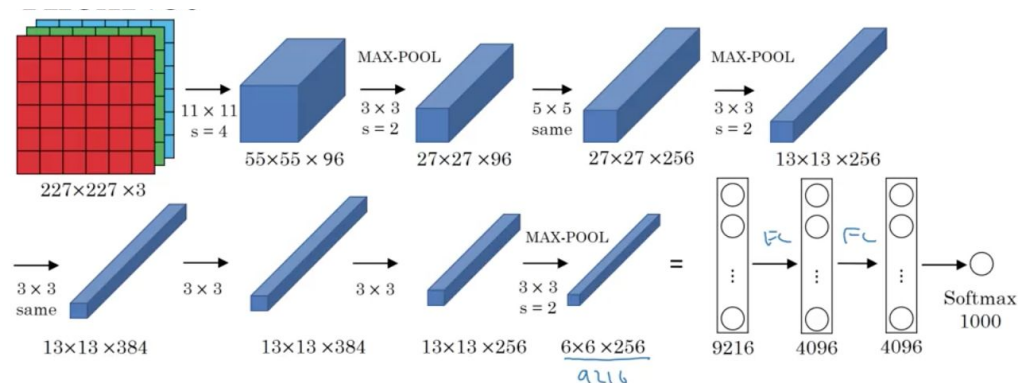
- Activation function
  - ReLU (όχι Sigmoid ή Tanh)
    - 5 x ταχύτητα,
    - ίδια ακρίβεια
- OverFitting
  - Dropout (όχι regularisation)
    - Διπλασιασμός χρόνου εκπαίδευσης
- Περισσότερα δεδομένα και μεγαλύτερο μοντέλο
  - 7 hidden layers, 650K units και 60M parameters.



[Krizhevsky et al., 2012. ImageNet classification with deep convolutional neural networks]

Andrew Ng

# CNN: Αριθμός παραμέτρων και μεγεθών τανυστών



## Τανυστής (Tensor)

εικόνα με αυθαίρετο αριθμό  
καναλιών

**Conv-1:** 96 πυρήνες μεγέθους  $11 \times 11$  που εφαρμόζονται με βήμα ( $s$ =stride) 4 και γέμισμα ( $p$ =padding) 0

**MaxPool-1:** μέγεθος συγκέντρωσης (pooling)  $3 \times 3$

**Conv-2:** 256 πυρήνες μεγέθους  $5 \times 5$ ,  $s=1$ ,  $p=2$

**MaxPool-2:** pooling size  $3 \times 3$ ,  $s=2$ .

**Conv-3:** 384 πυρήνες μεγέθους  $3 \times 3$ ,  $s=1$ ,  $p=1$

**Conv-4:** 384 πυρήνες μεγέθους  $3 \times 3$ ,  $s=1$ ,  $p=1$

**Conv-5:** 256 πυρήνες μεγέθους  $3 \times 3$ ,  $s=1$ ,  $p=1$

**MaxPool-3:** pooling size  $3 \times 3$  και  $s=2$

**FC-1:** 4096 νευρώνες

**FC-2:** 4096 νευρώνες

**FC-3:** 1000 νευρώνες

# Πώς να υπολογίσετε το μέγεθος του ταχυστή σε κάθε στάδιο;

Οπότε, αν στην είσοδο του Convolution Layer έχω τένσορες διάστασης  $W_1 \times H_1 \times D_1$  απαιτείται ο ορισμός 4 υπερπαραμέτρων:

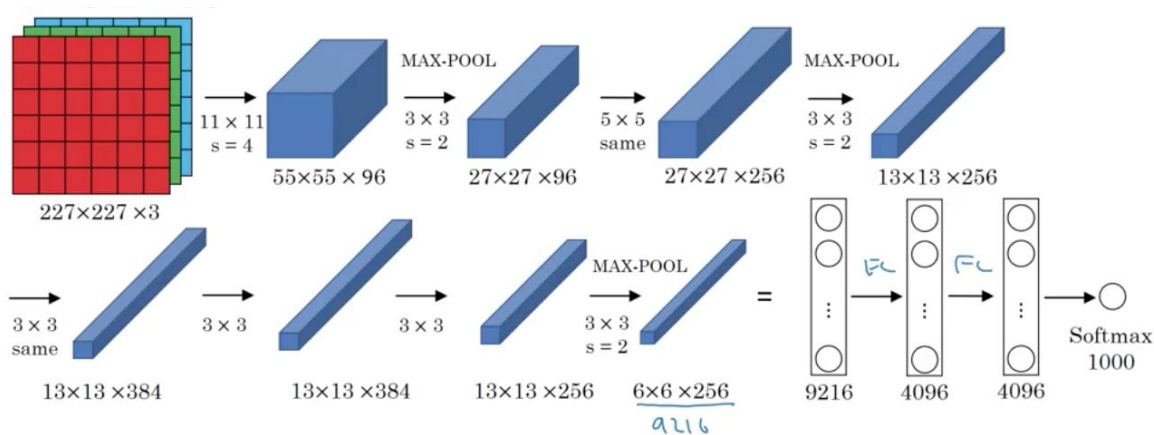
1. το πλήθος των φίλτρων ( $K$ )
2. το μέγεθος των φίλτρων ( $F$ )
3. το stride ( $S$ )
4. το padding ( $P$ )

Στην έξοδο του Convolution Layer θα πάρω τένσορες μεγέθους  $W_2 \times H_2 \times D_2$  όπου:

1.  $W_2 = \frac{(W_1 - F + 2 * P)}{S} + 1$
2.  $H_2 = \frac{(H_1 - F + 2 * P)}{S} + 1$
3.  $D_2 = K$

[Convolution demo](#)

Πώς να υπολογίσετε το μέγεθος του ταχυστή σε κάθε στάδιο;



$$1. W_2 = \frac{(W_1 - F + 2 * P)}{S} + 1$$

$$2. H_2 = \frac{(H_1 - F + 2 * P)}{S} + 1$$

$$3. D_2 = K$$

$$W_2, H_2 = \frac{227 - 11 + 2 \times 0}{4} + 1 = 55$$

$$K = 96$$

# Μέγεθος ταχυστή εξόδου ενός επιπέδου MaxPool

$O$  = Μέγεθος (πλάτος) της εικόνας εξόδου

$I$  = Μέγεθος (πλάτος) εικόνας εισόδου

$S$  = Βήμα

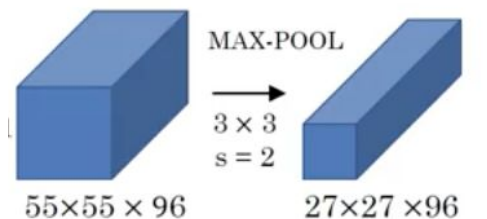
$P_s$  = Μέγεθος

$$O = \frac{I - P_s}{S} + 1$$

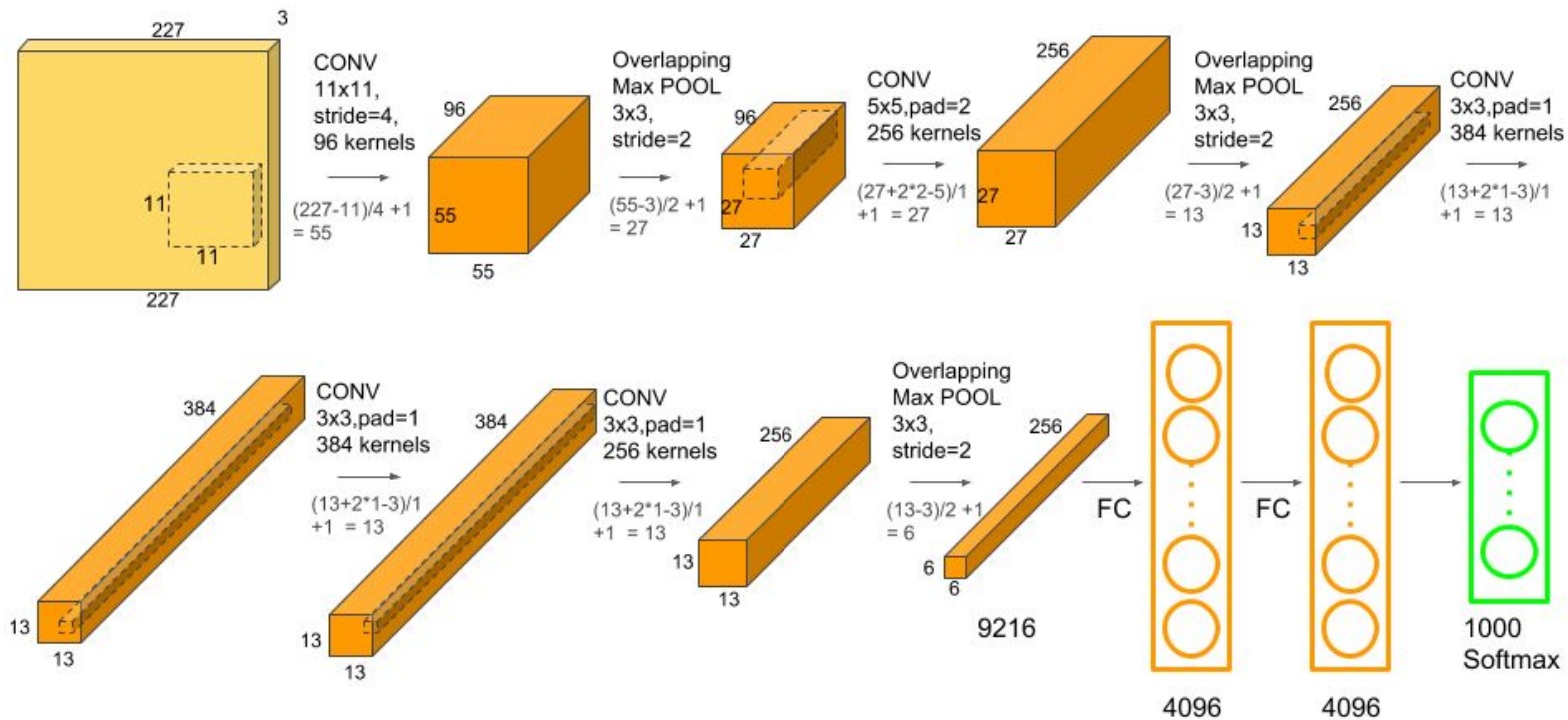
Στο AlexNet, το επίπεδο MaxPool μετά το σύνολο των φίλτρων συνέλιξης έχει μέγεθος  $3 \times 3$  και διασκελισμό 2 και η εικόνα εισόδου σε αυτό το στάδιο είναι μεγέθους  $55 \times 55 \times 96$ .

Άρα για την εικόνα εξόδου θα έχουμε:

$$O = \frac{55 - 3}{2} + 1 = 27$$



# AlexNet



# Αριθμός παραμέτρων ενός επιπέδου συνέλιξης

Σε ένα CNN, κάθε επίπεδο έχει δύο είδη παραμέτρων: βάρη και biases.

Ο συνολικός αριθμός των παραμέτρων είναι απλώς το άθροισμα όλων των βαρών και των biases

Ας ορίσουμε:  $W_c = K^2 \times C \times N$

$$B_c = N$$

$$P_c = W_c + B_c$$

όπου:  $W_c$  = Αριθμός βαρών του επιπέδου συνέλιξης

$B_c$  = Αριθμός biases του επιπέδου συνέλιξης

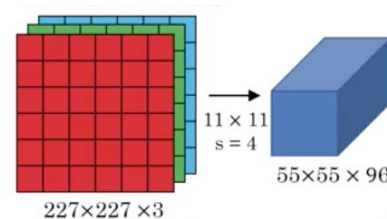
$P_c$  = Αριθμός παραμέτρων του επιπέδου συνέλιξης

$K$  = Μέγεθος (πλάτος) των πυρήνων που χρησιμοποιούνται στο επίπεδο συνέλιξης

$N$  = Αριθμός πυρήνων.

$C$  = Αριθμός καναλιών της εικόνας εισόδου.

## AlexNet Conv-1



$$W_c = 11^2 \times 3 \times 96 = 34,848$$

$$B_c = 96$$

$$P_c = 34,848 + 96 = 34,944$$

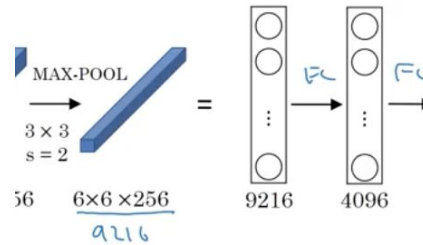
# Αριθμός παραμέτρων ενός επιπέδου MaxPool

Το μέγεθος, ο βηματισμός είναι υπερπαραμέτροι.

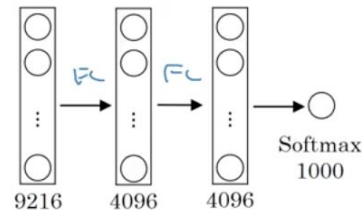
# Αριθμός παραμέτρων ενός πλήρως συνδεδεμένου επιπέδου (FC)

Υπάρχουν δύο είδη πλήρως συνδεδεμένων επιπέδων σε ένα CNN.

- το πρώτο επίπεδο FC συνδέεται με το τελευταίο επίπεδο συνέλιξης



- τα μεταγενέστερα επίπεδα FC συνδέονται με άλλα επίπεδα FC





# Αριθμός παραμέτρων ενός πλήρως συνδεδεμένου επιπέδου (FC)

- Το πρώτο επίπεδο FC συνδέεται με το τελευταίο επίπεδο συνέλιξης

$$W_{cf} = O^2 \times N \times F$$

Ας ορίσουμε,  $B_{cf} = F$

$$P_{cf} = W_{cf} + B_{cf}$$

$W_{cf}$  = Αριθμός βαρών ενός επιπέδου FC που είναι συνδεδεμένο με ένα επίπεδο συνέλιξης.

$B_{cf}$  = Αριθμός biases ενός επιπέδου FC που είναι συνδεδεμένο σε ένα επίπεδο συνέλιξης.

$P_{cf}$  = Αριθμός παραμέτρων του επιπέδου συνέλιξης

$O$  = Μέγεθος (πλάτος) της εικόνας εξόδου του προηγούμενου επιπέδου συνέλιξης.

$N$  = Αριθμός πυρήνων στο προηγούμενο επίπεδο συνέλιξης.

$F$  = Αριθμός νευρώνων στο στρώμα FC.

## Παράδειγμα

Το πρώτο πλήρως συνδεδεμένο επίπεδο του AlexNet συνδέεται με ένα επίπεδο συνέλιξης

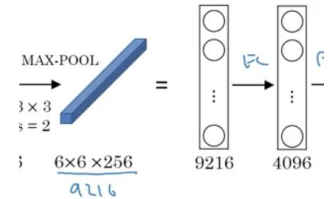
Για αυτό το επίπεδο,  $O=6$ ,  $N=256$  και  $F=4096$

$$W_{cf} = 6^2 \times 256 \times 4096 = 37,748,736$$

$$B_{cf} = 4096$$

Ως εκ τούτου

$$P_{cf} = W_{cf} + B_{cf} = 37,752,832$$



# Αριθμός παραμέτρων ενός πλήρως συνδεδεμένου επιπέδου (FC)

- τα μεταγενέστερα επίπεδα FC συνδέονται με άλλα επίπεδα FC

Ας ορίσουμε,

$$W_{ff} = F_{-1} \times F$$

$$B_{ff} = F$$

$$P_{ff} = W_{ff} + B_{ff}$$

$W_{ff}$  = Αριθμός βαρών ενός επιπέδου FC που είναι συνδεδεμένο με ένα επίπεδο συνέλιξης.

$B_{ff}$  = Αριθμός biases ενός επιπέδου FC που είναι συνδεδεμένο σε ένα επίπεδο συνέλιξης.

$P_{ff}$  = Αριθμός παραμέτρων του επιπέδου συνέλιξης

$F$  = Αριθμός νευρώνων στο στρώμα FC.

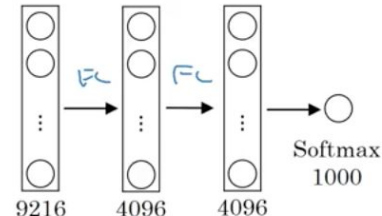
$F_{-1}$  = Αριθμός νευρώνων στο προηγούμενο στρώμα FC.

## Παράδειγμα

$$W_{ff} = 4096 \times 1000 = 4,096,000$$

$$B_{ff} = 1,000$$

$$P_{ff} = W_{ff} + B_{ff} = 4,097,000$$



# Πώς να υπολογίσετε τον συνολικό αριθμό παραμέτρων στο δίκτυο

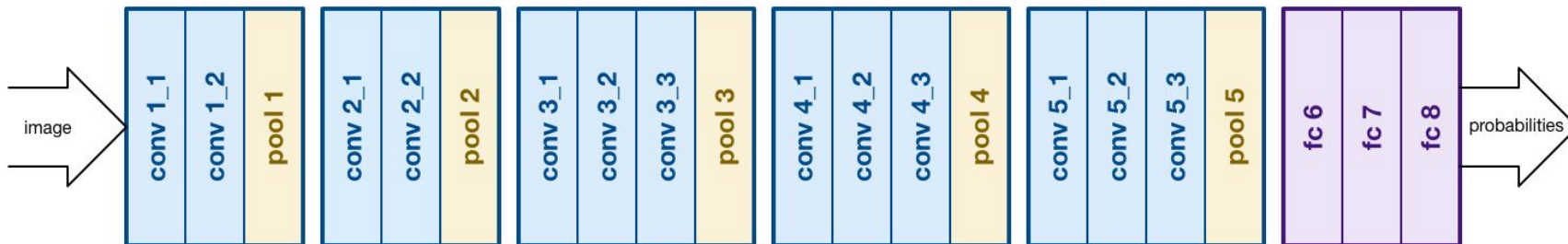
Layer Name	Tensor Size	Weights	Biases	Parameters
Input Image	227x227x3	0	0	0
Conv-1	55x55x96	34,848	96	34,944
MaxPool-1	27x27x96	0	0	0
Conv-2	27x27x256	614,400	256	614,656
MaxPool-2	13x13x256	0	0	0
Conv-3	13x13x384	884,736	384	885,120
Conv-4	13x13x384	1,327,104	384	1,327,488
Conv-5	13x13x256	884,736	256	884,992
MaxPool-3	6x6x256	0	0	0
FC-1	4096x1	37,748,736	4,096	37,752,832
FC-2	4096x1	16,777,216	4,096	16,781,312
FC-3	1000x1	4,096,000	1,000	4,097,000
Output	1000x1	0	0	0
<b>Total</b>				<b>62,378,344</b>

# CNN Architectures : VGG

[Simonyan, K. and Zisserman, A. \(2015\) Very Deep Convolutional Networks for Large-Scale Image Recognition. 3rd International Conference on Learning Representations \(ICLR2015\).](#)

13 συνελκτικά και 3 πλήρως συνδεδεμένα επίπεδα, ReLU, φίλτρα μικρότερου μεγέθους ( $2 \times 2$  και  $3 \times 3$ ) από το AlexNet , 138M παραμέτρους, 500MB

- Μείωση του αριθμού των παραμέτρων στα επίπεδα CONV
- Βελτίωση του χρόνου εκπαίδευσης
- Σχεδίασαν επίσης βαθύτερες παραλλαγές, VGG-16, VGG-19.



# CNN Architectures : VGG-16

Η ιδέα πίσω από την ύπαρξη πυρήνων σταθερού μεγέθους είναι ότι όλοι οι conv πυρήνες μεταβλητού μεγέθους που χρησιμοποιούνται στο Alexnet (11x11, 5x5, 3x3) μπορούν να αναπαραχθούν χρησιμοποιώντας πολλαπλούς πυρήνες 3x3 ως δομικά στοιχεία.

π.χ. Έστω επίπεδο εισόδου μεγέθους 5x5x1

**Περίπτωση 1:** 1ο conv επίπεδο: ένας πυρήνας 5x5 και βήμα 1 → Έξοδος: χάρτης χαρακτηριστικών 1x1

**Πλήθος μεταβλητών**  $5 \times 5 \times 1 + 1 = 26$  ( $(m \times n + 1) \times k$ , **k: πλήθος πυρήνων**)

**Περίπτωση 2:** 1ο conv επίπεδο: δύο πυρήνες 3x3 και βήμα 1 → Έξοδος: χάρτης χαρακτηριστικών 1x1.

**Πλήθος μεταβλητών**  $3 \times 3 \times 2 = 18$  → Μείωση 28%

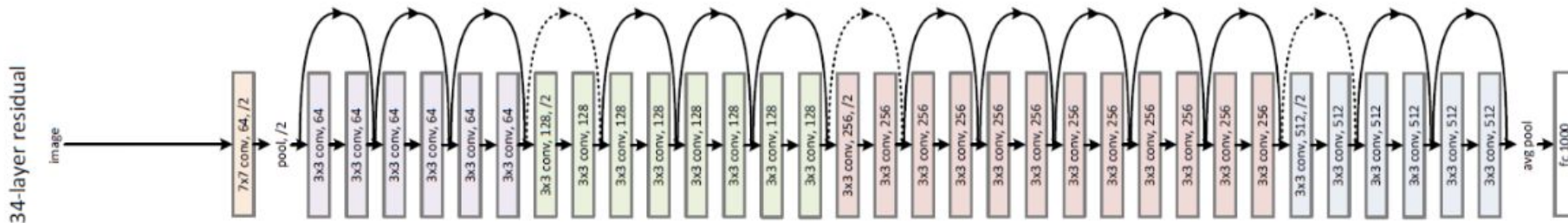
Αντίστοιχα αντί για χρήση πυρήνων 7x7 (11x11) εφαρμόσουμε 3 (5) 3x3 πυρήνες → μείωση αριθμού εκπαιδευόμενων μεταβλητών κατά 44,9% (62,8%)

★ Ταχύτερη εκμάθηση

★ Αποφυγή overfitting

# CNN Architectures: ResNet (MSRA)

[Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Deep Residual Learning for Image Recognition, CVPR 2015](#)



## Resnet18

- ◆ 152 επίπεδα, 11M παράμετροι, πυρήνες, 3x3 (όπως το VGGNet), 2 pooling επίπεδα

**Σύνδεση ταυτότητας (Identity connection)** ανά δύο επιπέδων CONV, διάσταση εισόδου ίδια με της εξόδου

**Σύνδεση προβολής (Projection connection)** όπου οι διαστάσεις εισόδου διαφέρουν με αυτές της εξόδου.

Υπάρχουν πολλές εκδόσεις αρχιτεκτονικών ResNetXX όπου το «XX» υποδηλώνει τον αριθμό των επιπέδων (ResNet50, ResNet101)

# CNN Architectures : ResNet (MSRA)

Αντί να μάθει την αναπαράσταση  $x \rightarrow F(x)$ , το δίκτυο μαθαίνει την  $x \rightarrow F(x) + G(x)$

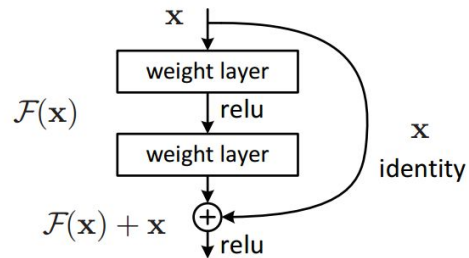
→ Όταν  $x=F(x)$ , η συνάρτηση  $G(x) = x$  είναι μια συνάρτηση ταυτότητας → **σύνδεση ταυτότητας**  
Η αναπαράσταση όταν  $x=F(x)$  μαθαίνεται με μηδενισμό των βαρών στο ενδιάμεσο στρώμα κατά τη διάρκεια της εκπαίδευσης

→ Όταν  $x \neq F(x)$  (stride > 1 στα μεταξύ τους επίπεδα CONV) → **σύνδεση προβολής**  
Η συνάρτηση  $G(x)$  αλλάζει τις διαστάσεις της εισόδου  $x$  σε εκείνη της εξόδου  $F(x)$

Είδη αντιστοίχισης

- Non-trainable Mapping (Padding)  
Η είσοδος  $x$  είναι απλώς γεμάτη με μηδενικά για να ταιριάζει η διάσταση με εκείνη του  $F(x)$ .
- Trainable Mapping (Conv Layer)  
1x1 Conv layer χρησιμοποιείται για την αντιστοίχιση του  $x$  με την  $G(x)$ .

**residual  
block**



# Διαχωρίσιμες Συνελίξεις

Υπάρχουν δύο κύριοι τύποι διαχωρίσιμων συνελίξεων:

- **Χωρικά διαχωρίσιμες συνελίξεις (Spatial Separable Convolutions)**
  - Διαπραγματεύεται κυρίως τις χωρικές διαστάσεις μιας εικόνας και του πυρήνα: το **πλάτος** και το **ύψος**
- **Κατά βάθος διαχωρίσιμες συνελίξεις (Depthwise separable convolutions)**
  - Διαπραγματεύεται όχι μόνο τις χωρικές διαστάσεις μιας εικόνας και του πυρήνα: το **πλάτος** και το **ύψος** αλλά και τη **διάσταση του βάθους** δηλαδή το πλήθος των καναλιών της εικόνας



# Χωρικά διαχωρίσιμη συνέλιξη (Spatial Separable Convolution)

- Μια χωρικά διαχωρίσιμη συνέλιξη απλώς χωρίζει έναν πυρήνα σε δύο, μικρότερους πυρήνες.

- Η πιο συνηθισμένη περίπτωση θα ήταν να χωρίσετε έναν πυρήνα 3x3 σε πυρήνα 3x1 και 1x3,

όπως:

$$\begin{bmatrix} 3 & 6 & 9 \\ 4 & 8 & 12 \\ 5 & 10 & 15 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix} \times [1 \ 2 \ 3]$$

- Έτσι, αντί να κάνουμε μια συνέλιξη με 9 πολλαπλασιασμούς, κάνουμε δύο συνέλιξεις με 3 πολλαπλασιασμούς η καθεμία (6 συνολικά) για να πετύχουμε το ίδιο αποτέλεσμα.

- **Με λιγότερους πολλαπλασιασμούς:**

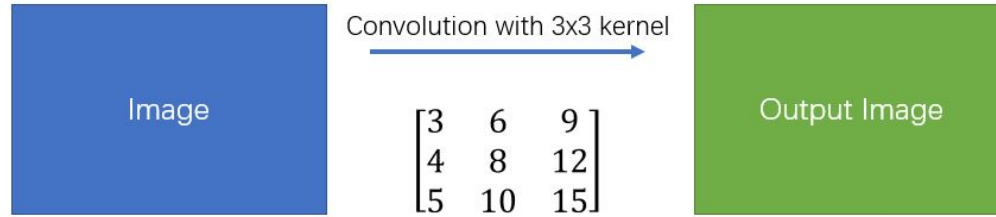
- η υπολογιστική πολυπλοκότητα μειώνεται
- το δίκτυο μπορεί να λειτουργεί πιο γρήγορα

π.χ. διαχωρισμός του πυρήνα Sobel

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \times [-1 \ 0 \ 1]$$

# Χωρικά διαχωρίσιμη συνέλιξη (Spatial Separable Convolution)

## Simple Convolution



## Spatial Separable Convolution



# Χωρικά διαχωρίσιμη συνέλιξη (Spatial Separable Convolution)

- Πόσο εφικτό είναι όλοι οι πυρήνες που υιοθετεί το δίκτυο να μπορούν να χωριστούν σε δύο μικρότερους πυρήνες;
  - Το δίκτυο, μπορεί να καταλήξει να χρησιμοποιεί μόνο μια μικρή ποσότητα πυρήνων, αυτούς που μπορεί να χωριστεί σε δύο μικρότερους πυρήνες.  
→ Μη αποδοτική τεχνική

# Κατά βάθος διαχωρίσιμη συνελίξη (Depthwise separable convolution)

Λειτουργεί με πυρήνες που δεν μπορούν να «παραγοντοποιηθούν» σε δύο μικρότερους πυρήνες.

- δεν ασχολείται μόνο με τις χωρικές διαστάσεις, αλλά και με τη διάσταση βάθους
- παρόμοια με τη χωρική συνέλιξη, μια κατά βάθος διαχωρίσιμη συνέλιξη χωρίζει έναν πυρήνα σε 2 ξεχωριστούς πυρήνες που κάνουν δύο συνέλιξη: τη συνέλιξη κατά βάθος και τη σημειακή συνέλιξη.

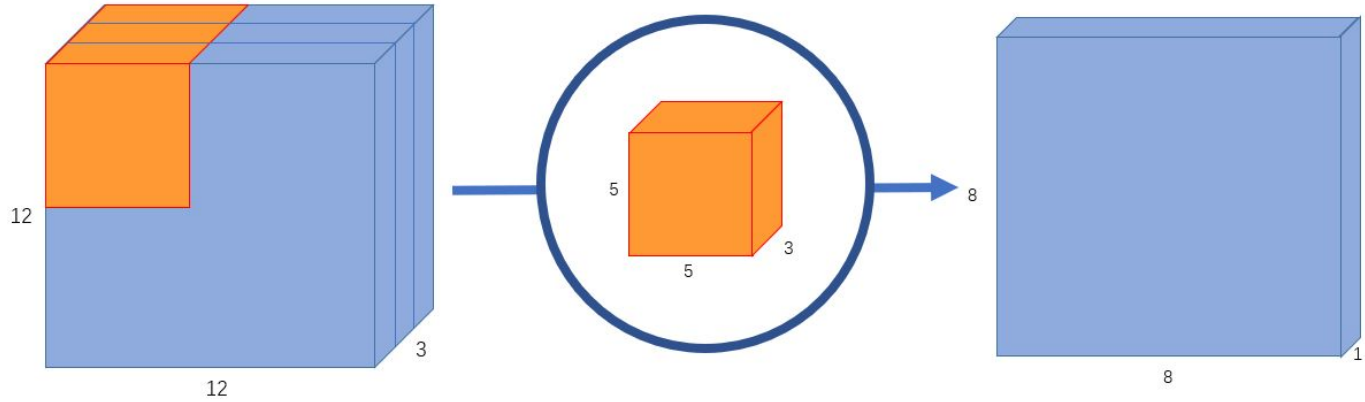
π.χ. Έστω ότι έχουμε στην είσοδο ενός δικτύου μία εικόνα RGB

- Μπορούμε να απεικονίσουμε κάθε κανάλι ως μια συγκεκριμένη ερμηνεία αυτής της εικόνας
  - κάθε κανάλι δείχνει τη ένταση του κάθε χρώματος της εικόνας
    - κόκκινο κανάλι - ένταση του κόκκινου χρώματος της εικόνας
- Μια εικόνα με 256 κανάλια
  - Κάθε κανάλι δείχνει διαφορετική ερμηνεία της εικόνας

- `keras.layers.SeparableConv2D`  
- `tf.layers.separable_conv2d`

# Κατά βάθος διαχωρίσιμη συνελίξη (Depthwise separable convolution)

Κανονική συνέλιξη:

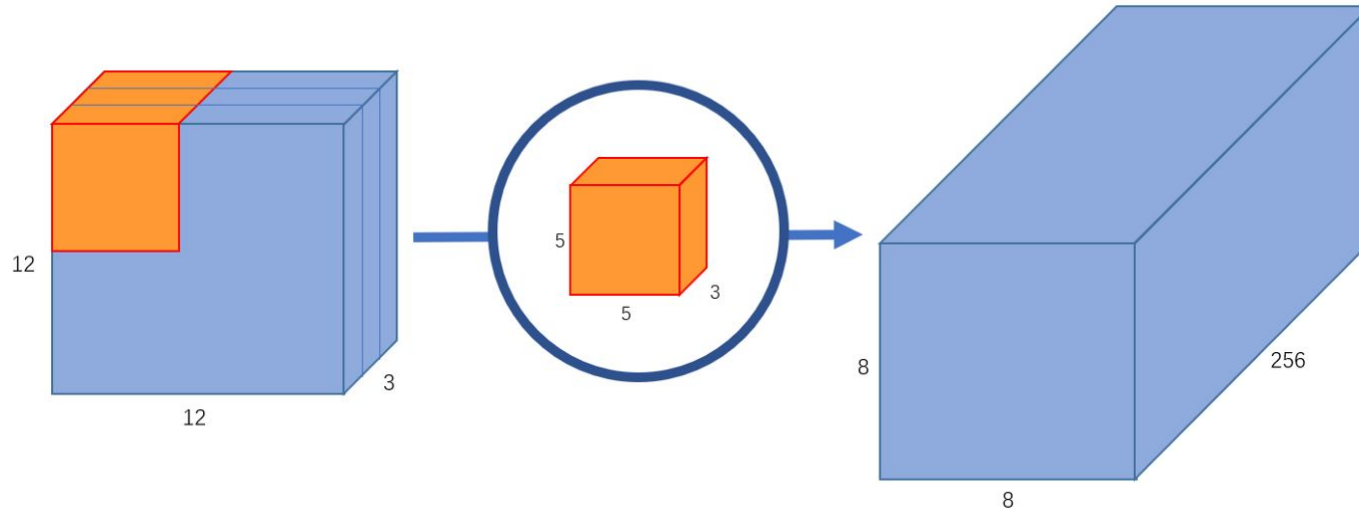


$$W_2 = (12 - 5 + 0) / 1 = 8, \quad H_2 = (12 - 5 + 0) / 1 = 8, \quad D_2 = 1$$

→ κάνουμε  $5 \times 5 \times 3 = 75$  πολλαπλασιασμούς κάθε φορά που κινείται ο πυρήνας

# Κατά βάθος διαχωρίσιμη συνελιξη (Depthwise separable convolution)

- Τι γίνεται αν θέλουμε να αυξήσουμε τον αριθμό των καναλιών στην εικόνα εξόδου μας;
  - Μπορούμε να δημιουργήσουμε 256 πυρήνες για να δημιουργήσουμε 256 εικόνες  $8 \times 8 \times 1$
- Τι γίνεται αν θέλουμε μια έξοδο μεγέθους  $8 \times 8 \times 256$ ;
  - Μπορούμε να στοιβάζουμε τις 256 εικόνες  $8 \times 8 \times 1$  μαζί για να δημιουργήσουμε μια έξοδο εικόνας  $8 \times 8 \times 256$ .



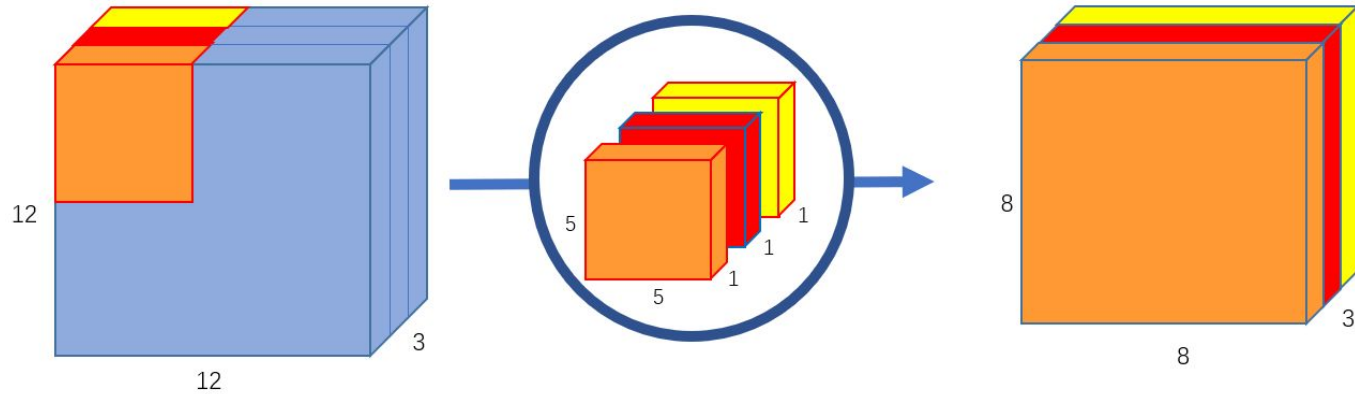
# Κατά βάθος διαχωρίσιμη συνέλιξη (Depthwise separable convolution)

Μια κατά βάθος συνέλιξη χωρίζει αυτή τη διαδικασία σε 2 μέρη:

- A. Μια συνέλιξη κατά βάθος (depthwise convolution)
- B. Μια συνέλιξη κατά σημείο (pointwise convolution)

# Κατά βάθος διαχωρίσιμη συνελιξη (Depthwise separable convolution)

## A. Συνέλιξη κατά βάθος

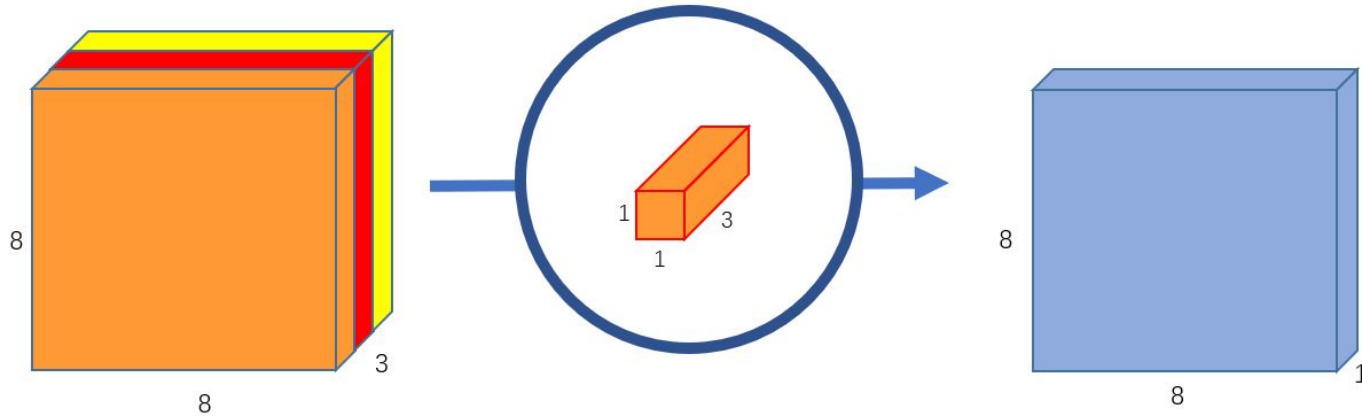


- Χρησιμοποιεί 3 πυρήνες  $5 \times 5 \times 1$
- Κάθε πυρήνας  $5 \times 5 \times 1$  εφαρμόζεται σε 1 κανάλι της εικόνας, λαμβάνοντας τα βαθμωτά γινόμενα κάθε ομάδας 25 pixels και δίνοντας μια εικόνα  $8 \times 8 \times 1$
- Η στοίβαξη αυτών των εικόνων μαζί δημιουργεί μια εικόνα  $8 \times 8 \times 3$



# Κατά βάθος διαχωρίσιμη συνελιξη (Depthwise separable convolution)

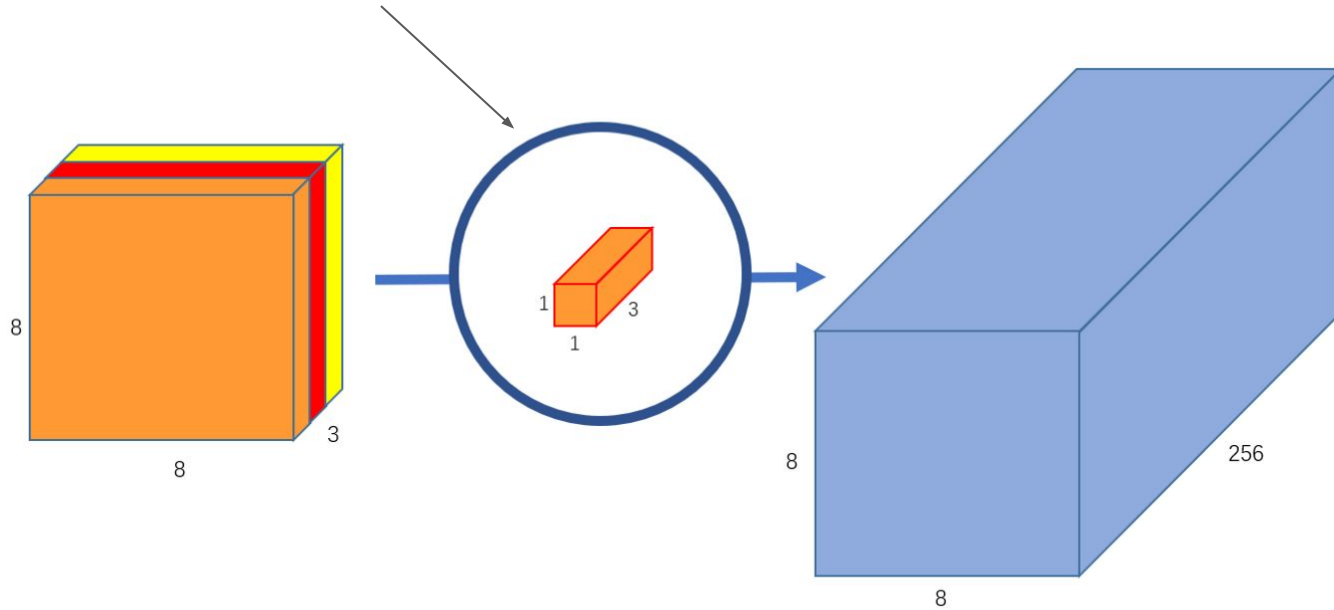
## B. Συνέλιξη κατά σημείο (pointwise convolution)



- Χρησιμοποιεί έναν πυρήνα 1x1 ή έναν πυρήνα όπου έχει βάθος όσο το πλήθος των καναλιών της εικόνας εισόδου (3)
  - Επομένως, επαναλαμβάνουμε έναν πυρήνα 1x1x3 μέσω της εικόνας 8x8x3, για να λάβουμε μια εικόνα 8x8x1

# Κατά βάθος διαχωρίσιμη συνελίξη (Depthwise separable convolution)

Συνέλιξη κατά σημείο με 256 πυρήνες  $\rightarrow$  εξαγωγή εικόνας με 256 κανάλια



Μπορούμε να δημιουργήσουμε 256 πυρήνες  $1 \times 1 \times 3$  που δίνουν μια εικόνα  $8 \times 8 \times 1$  για να πάρουμε μια τελική εικόνα σχήματος  $8 \times 8 \times 256$ .

- **2D Convolution**
  - Αριθμός απαιτούμενων πολλαπλασιασμών
    - $(8 \times 8) \times (5 \times 5 \times 3) \times (256) = 4.800 \times (256) = 1.228.800$
- **Depthwise Separable Convolution**
  - Αριθμός των απαιτούμενων πολλαπλασιασμών:
    - A. Συνέλιξη κατά βάθος
      - Χωρίζεται σε μεμονωμένα κανάλια, επομένως απαιτείται φίλτρο  $5 \times 5 \times 1$  στη θέση του  $5 \times 5 \times 3$ , και εφόσον υπάρχουν τρία κανάλια
      - Ο συνολικός αριθμός των απαιτούμενων φίλτρων  $5 \times 5 \times 1$  είναι 3, άρα,  
 $(8 \times 8) \times (5 \times 5 \times 1) \times (3) = 4.800$
    - B. Συνέλιξη κατά σημείο
      - Ο συνολικός αριθμός των απαιτούμενων καναλιών είναι 256, άρα,
      - $(8 \times 8) \times (1 \times 1 \times 3) \times (256) = 192 \times (256) = 49.152$

Συνολικός αριθμός πολλαπλασιασμών =  $4.800 + 49.152 = 53.952$

$1.228.800 / 53.952 = 23 \Rightarrow$  απαιτούνται 23 φορές λιγότεροι πολλαπλασιασμοί

→ η αποτελεσματικότητα των Depthwise Separable συνελίξεων είναι τόσο υψηλή

- Και στις δύο περιπτώσεις:
  - περνάμε την εικόνα μέσα από έναν πυρήνα  $5 \times 5$ ,
  - τη συρρικνώνουμε σε ένα κανάλι
  - και μετά την επεκτείνουμε σε 256 κανάλια.

Πώς γίνεται το ένα να είναι περισσότερο από δύο φορές πιο γρήγορο από το άλλο;

- Στην κανονική συνέλιξη:
  - μετασχηματίζουμε την εικόνα 256 φορές.
  - κάθε μετασχηματισμός χρησιμοποιεί  $5 \times 5 \times 3 \times 8 \times 8 = 4800$  πολλαπλασιασμούς.
- Στη κατά βάθος διαχωρίσιμη συνελιξη,
  - μετασχηματίζουμε πραγματικά την εικόνα μόνο μία φορά – στην κατά βάθος συνέλιξη.
  - έπειτα παίρνουμε τη μετασχηματισμένη εικόνα και απλώς την επιμηκύνουμε σε 256 κανάλια.

→ χωρίς να χρειάζεται να μετασχηματίσουμε την εικόνα ξανά και ξανά, μπορούμε να εξοικονομήσουμε υπολογιστική ισχύ.

# Κατά βάθος διαχωρίσιμη συνελιξη (Depthwise separable convolution)

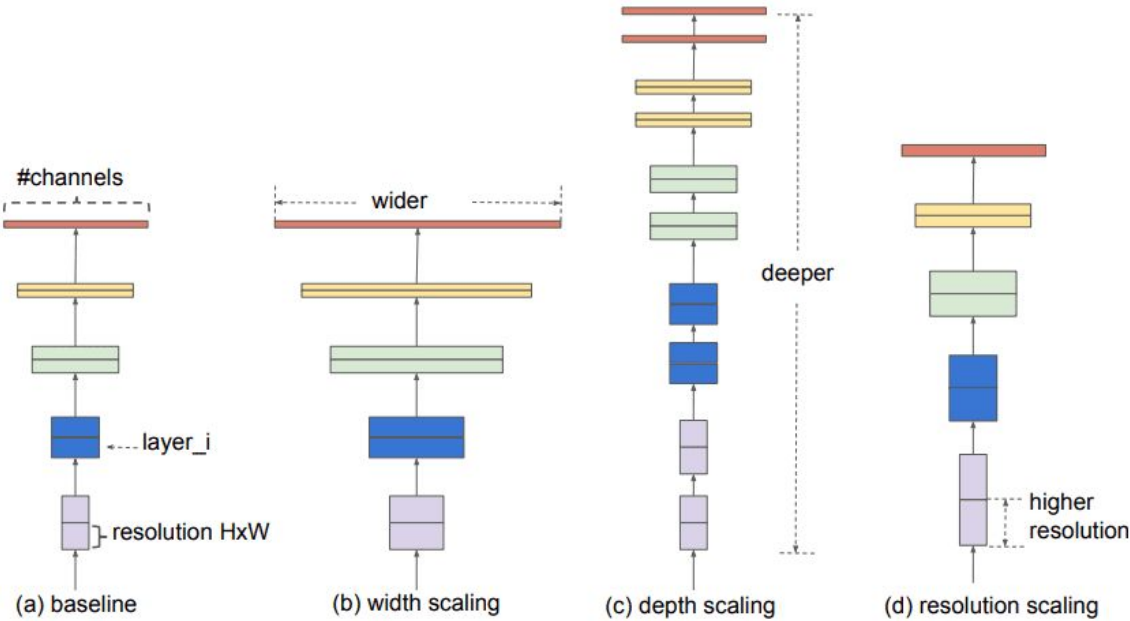
## Μειονεκτήματα

- Μειώνει τον αριθμό των παραμέτρων σε μια συνέλιξη
  - εάν το δίκτυο είναι ήδη μικρό, μπορεί να καταλήξει με πολύ λίγες παραμέτρους  
→ το δίκτυο μπορεί να μην μάθει σωστά κατά τη διάρκεια της εκπαίδευσης
- Εάν χρησιμοποιηθεί σωστά:
  - καταφέρνει να ενισχύσει την αποτελεσματικότητα  
→ γεγονός που το καθιστά μια αρκετά δημοφιλή επιλογή.

# Πυρήνας 1x1

- Κύριος σκοπός ενός πυρήνα 1x1 είναι να εφαρμόσει μη γραμμικότητα.
  - μετά από κάθε επίπεδο ενός νευρωνικού δικτύου, μπορούμε να εφαρμόσουμε ένα επίπεδο ενεργοποίησης (ReLU ή PReLU ή Softmax ή άλλο)
  - τα επίπεδα ενεργοποίησης είναι μη γραμμικά, σε αντίθεση με τα επίπεδα συνέλιξης
    - Ένας γραμμικός συνδυασμός γραμμών εξακολουθεί να είναι μια γραμμή → δεν μας αφήνει να ανακαλύψουμε....καμπύλες
  - Τα μη γραμμικά επίπεδα διευρύνουν τις δυνατότητες για το μοντέλο, όπως είναι αυτό που κάνει γενικά ένα «βαθύ» δίκτυο καλύτερο από ένα «ευρύ» δίκτυο.
    - Για να αυξήσουμε τον αριθμό των μη γραμμικών επιπέδων χωρίς να αυξήσουμε σημαντικά τον αριθμό των παραμέτρων και των υπολογισμών, μπορούμε να εφαρμόσουμε έναν πυρήνα 1x1 και να προσθέσουμε ένα επίπεδο ενεργοποίησης μετά από αυτόν.
    - Αυτό βοηθά να δώσει στο δίκτυο ένα πρόσθετο επίπεδο βάθους.

# Είδη δικτύων



- Ευρύτερο δίκτυο (wider network)
  - περισσότερους χάρτες χαρακτηριστικών (φίλτρα) στα συνελκτικά επίπεδα
- Βαθύτερο δίκτυο (deeper network)
  - περισσότερα συνελκτικά επίπεδα
- Δίκτυο με υψηλότερη ανάλυση (higher resolution)
  - επεξεργάζεται εικόνες εισόδου με μεγαλύτερο πλάτος και βάθος (χωρικές αναλύσεις).
  - οι χάρτες χαρακτηριστικών που παράγονται θα έχουν υψηλότερες χωρικές διαστάσεις.

# CNN Architectures: Inception v1 (GoogLeNet)

Τα εμφανή μέρη (Salient parts) στην εικόνα μπορεί να έχουν εξαιρετικά μεγάλες διακυμάνσεις στο μέγεθος.

π.χ. η περιοχή που καταλαμβάνει ο σκύλος είναι διαφορετική σε κάθε εικόνα.



From left: A dog occupying most of the image, a dog occupying a part of it, and a dog occupying very little space (Images obtained from [Unsplash](#)).

- Υπάρχει τεράστια ποικιλία στη θέση των πληροφοριών,
- Η επιλογή του σωστού μεγέθους πυρήνα για συνέλιξη γίνεται δύσκολη.
  - ένας μεγαλύτερος πυρήνας προτιμάται για πληροφορίες που διανέμονται πιο σφαιρικά,
  - ένας μικρότερος πυρήνας προτιμάται για πληροφορίες που διανέμονται πιο τοπικά.
- Τα πολύ βαθιά δίκτυα είναι επιρρεπή σε υπερπροσαρμογή.
- Είναι επίσης δύσκολο να περάσεις ενημερώσεις gradient σε ολόκληρο το δίκτυο.
- Η στοίβαξη μεγάλων πυρήνων συνέλιξης είναι υπολογιστικά ακριβή και αφελής.



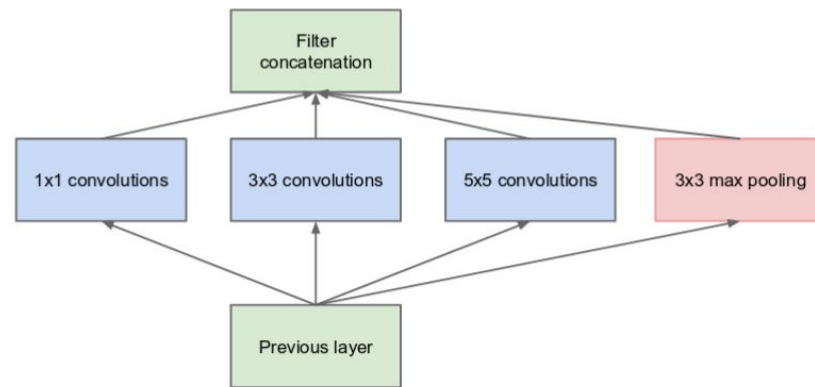
# CNN Architectures: Inception v1 (GoogLeNet)

Λύση: Γιατί να μην λειτουργούν **φίλτρα πολλαπλών μεγεθών στο ίδιο επίπεδο**;

→ Το δίκτυο ουσιαστικά θα γινόταν λίγο «**πλατύτερο**» παρά «**βαθύτερο**».

## "Αφελής" μονάδα inception:

- Εκτελεί συνέλιξη σε μια είσοδο, με 3 διαφορετικά μεγέθη φίλτρων (1x1, 3x3, 5x5).
- Επιπρόσθετα, πραγματοποιείται και max pooling.
- Οι έξοδοι συνδέονται και αποστέλλονται στην επόμενη μονάδα inception.



(a) Inception module, naïve version

# CNN Architectures: Inception v1 (GoogLeNet)

→ Τα βαθιά νευρωνικά δίκτυα είναι υπολογιστικά ακριβά.

Για **μείωση του υπολογιστικού κόστους** οι συγγραφείς πρότειναν το εξής:

- Περιορισμών του αριθμού των καναλιών εισόδου προσθέτοντας μια επιπλέον **συνέλιξη 1x1** πριν από τις συνελεύσεις 3x3 και 5x5.
  - Αν και η προσθήκη μιας επιπλέον λειτουργίας μπορεί να φαίνεται αντιφατική,
    - οι συνέλιξεις 1x1 είναι πολύ φθηνότερες από τις συνέλιξεις 5x5
    - ο μειωμένος αριθμός καναλιών εισόδου βοηθά επίσης.

Η συνέλιξη 1x1 εισάγεται μετά το μέγιστο επίπεδο συνένωσης και όχι πριν.

# CNN Architectures: Inception v1 (GoogLeNet)

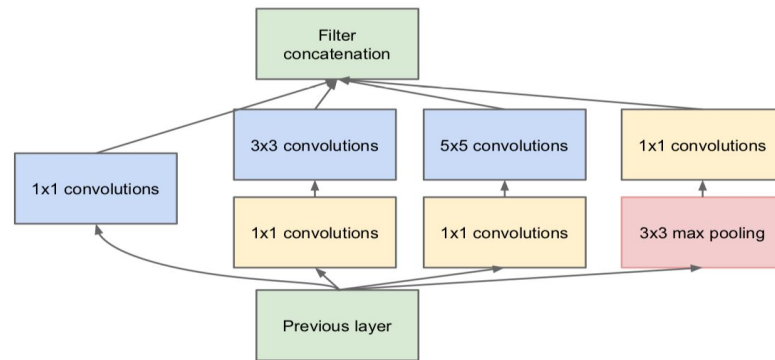
## Μονάδα Inception

Καταγράφει προεξέχοντα χαρακτηριστικά (salient features) σε διαφορετικά επίπεδα.

- 4 παράλληλες λειτουργίες
  - ◆ 1x1 convn layer, μείωση βάθους
  - ◆ 3x3 convn layer, Κατανεμημένα χαρακτηριστικά (distributed features)
  - ◆ 5x5 convn layer, Γενικά χαρακτηριστικά (global features)
  - ◆ max pooling, Χαμηλού επιπέδου χαρακτηριστικά (low level features)

→ Φίλτρο συνένωσης

π.χ. εάν οι εικόνες στο σύνολο δεδομένων έχουν πολλά καθολικά χαρακτηριστικά και ελάχιστα χαρακτηριστικά χαμηλού επιπέδου, τότε το εκπαιδευμένο δίκτυο Inception θα έχει πολύ μικρά βάρη που αντιστοιχούν στον πυρήνα 3x3 σε σύγκριση με τον πυρήνα 5x5.



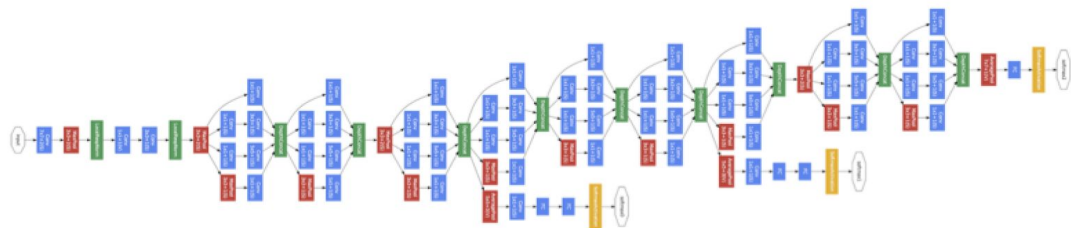
Inception Module (source: original paper)

# CNN Architectures: Inception v1 (GoogLeNet)

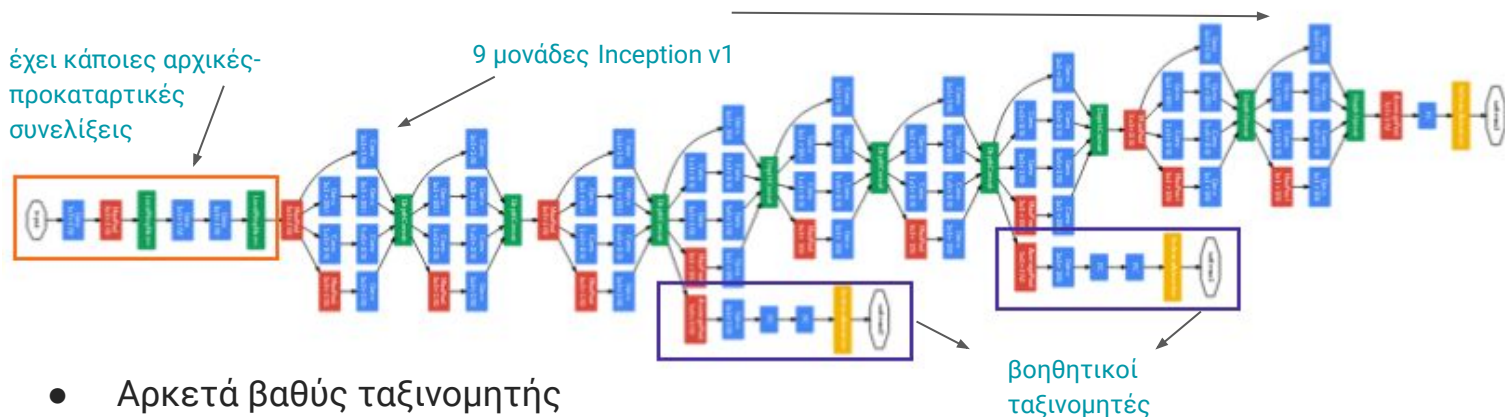
→ Διαδοχικά αλλά και παράλληλα CNN  
(error rate 6.7%)

→ Πολλαπλοί πυρήνες διαφορετικών  
μεγεθών εφαρμόζονται στο ίδιο  
επίπεδο με σκοπό τη ανίχνευση  
συγκεκριμένων χαρακτηριστικών  
περιοχής

- ◆ Μεγάλοι πυρήνες → καθολικά χαρακτηριστικά που κατανέμονται σε μεγάλη περιοχή της εικόνας,
- ◆ Μικροί πυρήνες → ανίχνευση συγκεκριμένων χαρακτηριστικών περιοχής που κατανέμονται σε ολόκληρο το πλαίσιο εικόνας.



# CNN Architectures: Inception v1 (GoogLeNet)



- Αρκετά βαθύς ταξινομητής
  - Όπως συμβαίνει με κάθε πολύ βαθύ δίκτυο, υπόκειται στο πρόβλημα της εξαφάνισης της κλίσης.
- Για να αποτρέψουν το μεσαίο τμήμα του δικτύου από το να «σβήσει», οι συγγραφείς εισήγαγαν δύο βοηθητικούς ταξινομητές.
  - Ουσιαστικά εφάρμοσαν softmax στις εξόδους σε δύο από τις inception μονάδες και υπολόγισαν μια βοηθητική απώλεια στις ίδιες ετικέτες.
  - Η συνάρτηση συνολικής απώλειας είναι ένα σταθμισμένο άθροισμα της επικουρικής απώλειας και της πραγματικής απώλειας:  $total\_loss = real\_loss + 0.3 * aux\_loss\_1 + 0.3 * aux\_loss\_2$

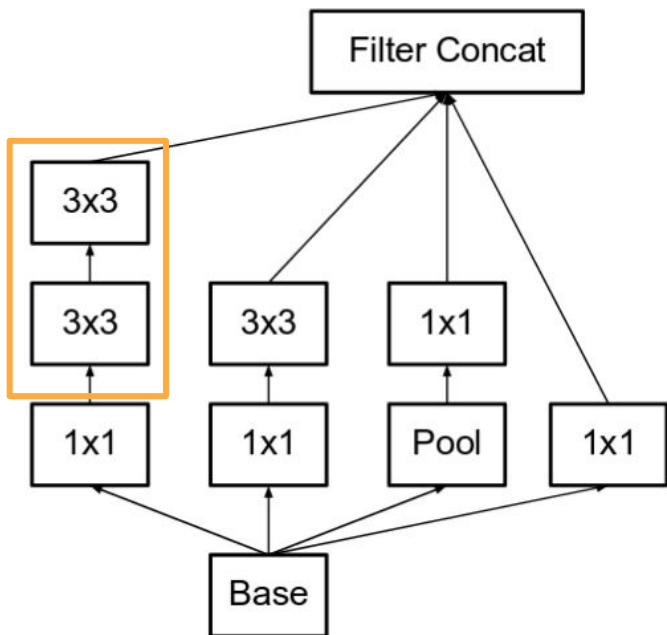
# CNN Architectures: Inception v2 και v3

Το Inception v2 και το Inception v3 παρουσιάστηκαν στο paper [C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," 2016 IEEE Conference on Computer Vision and Pattern Recognition \(CVPR\), 2016, pp. 2818-2826, doi: 10.1109/CVPR.2016.308.](#)

**Inception v2:** Μείωση του φαινομένου συμφόρησης( representational bottleneck)

- Οι συγγραφείς πρότειναν μια σειρά από αναβαθμίσεις με σκοπό την:
  - Αύξηση της ακρίβειας
  - Μείωση της υπολογιστικής πολυπλοκότητας
- Η διαίσθηση ήταν ότι τα νευρωνικά δίκτυα αποδίδουν καλύτερα όταν οι συνελίξεις δεν άλλαζαν δραστικά τις διαστάσεις της εισόδου.
- Η υπερβολική μείωση των διαστάσεων μπορεί να προκαλέσει απώλεια πληροφοριών, γνωστή ως «representational bottleneck»
- Χρησιμοποιώντας έξυπνες μεθόδους παραγοντοποίησης, οι συνελίξεις μπορούν να γίνουν πιο αποτελεσματικές όσον αφορά την υπολογιστική πολυπλοκότητα.

# CNN Architectures: Inception v2



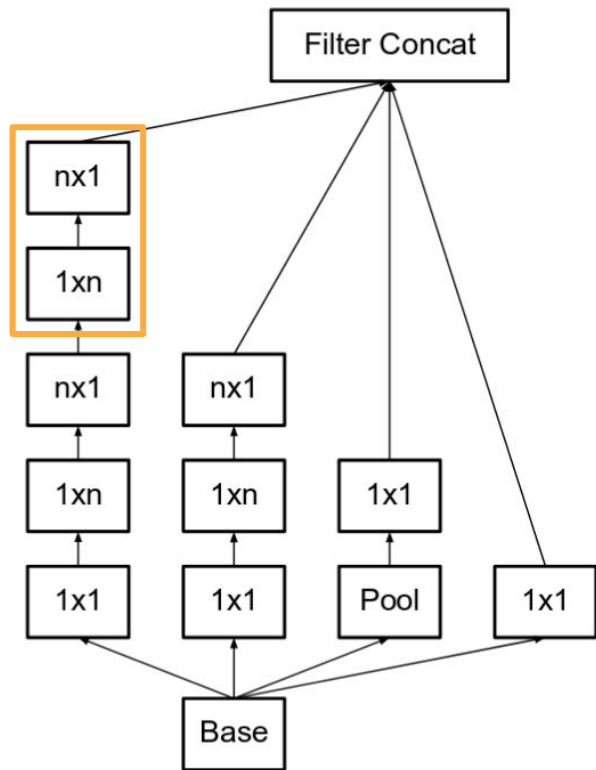
## Λύση

- Παραγοντοποίηση της συνέλιξη 5x5 σε δύο συνέλιξεις 3x3 για βελτίωση της υπολογιστικής ταχύτητας.
  - ◆ Αν και αυτό μπορεί να φαίνεται αντιφατικό, μια συνέλιξη 5x5 είναι 2,78 φορές πιο ακριβή από μια συνέλιξη 3x3.

Έτσι, η στοίβαξη δύο συνελιξεων 3x3 οδηγεί στην πραγματικότητα σε αύξηση της απόδοσης.

The left-most 5x5 convolution of the old inception module, is now represented as two 3x3 convolutions.

# CNN Architectures: Inception v2



→ Επιπλέον, παραγοντοποίηση των συνελίξεων μεγέθους φίλτρου  $n \times n$  σε συνδυασμό  $1 \times n$  και  $n \times 1$  συνελίξεων.

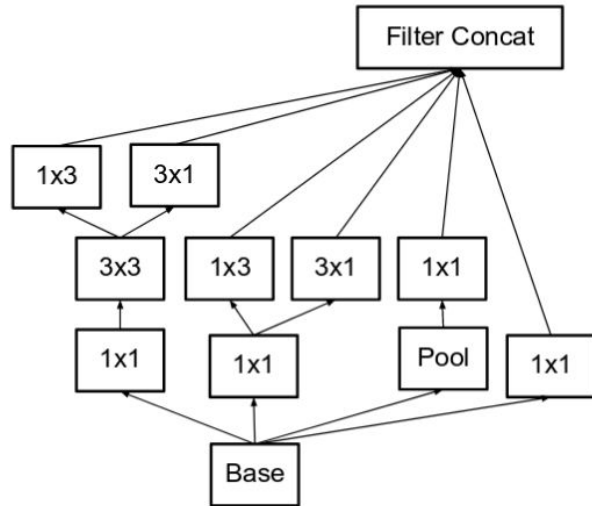
- ◆ Για παράδειγμα, μια συνέλιξη  $3 \times 3$  ισοδυναμεί με την εκτέλεση πρώτα μιας συνέλιξης  $1 \times 3$  και, στη συνέχεια, την εκτέλεση μιας συνέλιξης  $3 \times 1$  στην έξοδό της.

Η μέθοδος αυτή είναι 33% φθηνότερη από την απλή συνέλιξη  $3 \times 3$ .

Here, put  $n=3$  to obtain the equivalent of the previous image. The left-most  $5 \times 5$  convolution can be represented as two  $3 \times 3$  convolutions, which in turn are represented as  $1 \times 3$  and  $3 \times 1$  in series.



# CNN Architectures: Inception v2



- Τα φίλτρα στη μονάδα επεκτάθηκαν (έγιναν ευρύτερα αντί βαθύτερα) για να μειώσουν το φαινόμενο συμφόρησης( representational bottleneck)
- Αν η μονάδα γινόταν πιο βαθιά, θα υπήρχε υπερβολική μείωση των διαστάσεων και, ως εκ τούτου, απώλεια πληροφοριών.

Making the inception module wider. This type is equivalent to the module shown above.

(Source: [Inception v2](#))

# CNN Architectures: Inception v2

Οι παραπάνω τρεις αρχές χρησιμοποιήθηκαν για την κατασκευή τριών διαφορετικών τύπων μονάδων inception (figure 5 = slide pg 22, figure 6 = slide pg 23, figure 7 = slide pg 24)

type	patch size/stride or remarks	input size
conv	$3 \times 3 / 2$	$299 \times 299 \times 3$
conv	$3 \times 3 / 1$	$149 \times 149 \times 32$
conv padded	$3 \times 3 / 1$	$147 \times 147 \times 32$
pool	$3 \times 3 / 2$	$147 \times 147 \times 64$
conv	$3 \times 3 / 1$	$73 \times 73 \times 64$
conv	$3 \times 3 / 2$	$71 \times 71 \times 80$
conv	$3 \times 3 / 1$	$35 \times 35 \times 192$
3×Inception	As in figure 5	$35 \times 35 \times 288$
5×Inception	As in figure 6	$17 \times 17 \times 768$
2×Inception	As in figure 7	$8 \times 8 \times 1280$
pool	$8 \times 8$	$8 \times 8 \times 2048$
linear	logits	$1 \times 1 \times 2048$
softmax	classifier	$1 \times 1 \times 1000$

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$

# CNN Architectures: Inception v3

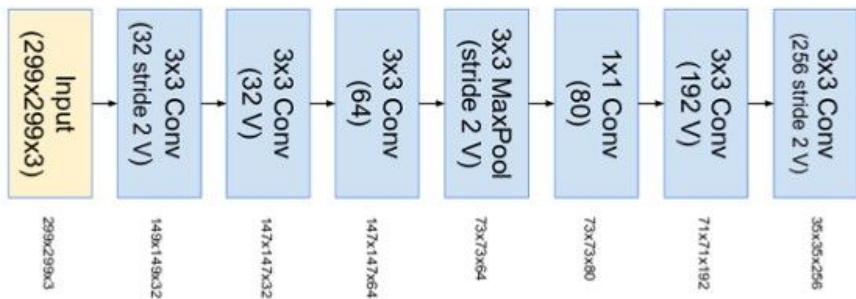
- Οι συγγραφείς παρατήρησαν ότι οι βοηθητικοί ταξινομητές δεν συνέβαλαν πολύ μέχρι το τέλος της εκπαιδευτικής διαδικασίας, όταν οι ακρίβειες πλησίαζαν στον κορεσμό.
  - ◆ Υποστήριξαν ότι λειτουργούν ως κανονικοποιητές, ειδικά εάν έχουν λειτουργίες BatchNorm ή Dropout.
- Διερεύνησαν τις δυνατότητες βελτίωσης στο Inception v2 χωρίς δραστική αλλαγή των μονάδων.

## Λύση

- RMSProp Optimizer([11.8. RMSProp – Dive into Deep Learning 0.17.5 documentation](#)).
- Παραγοντοποιημένες συνελίξεις 7x7.
- BatchNorm στους βοηθητικούς ταξινομητές.
- Εξομάλυνση ετικετών (Ένας τύπος κανονικοποίησης που προστίθεται στη loss function που εμποδίζει το δίκτυο να αποκτήσει υπερβολική αυτοπεποίθηση για μια κατηγορία. Αποτρέπει την υπερεκπαίδευση)

# CNN Architectures: Inception v4-Inception-ResNet

[Szegedy, C., Ioffe, S., Vanhoucke, V., et al. \(2017\) Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, 4-9 February 2017, 4278-4284](#)

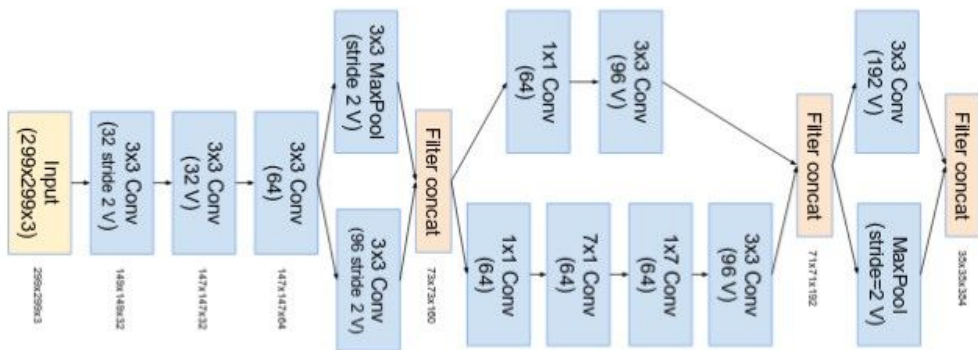


→ Οι συγγραφείς παρατήρησαν επίσης ότι ορισμένες από τις μονάδες ήταν πιο περίπλοκες από ό,τι ήταν απαραίτητο.

- ◆ Ενίσχυση της απόδοσης προσθέτοντας περισσότερες ομοιόμορφες μονάδες.

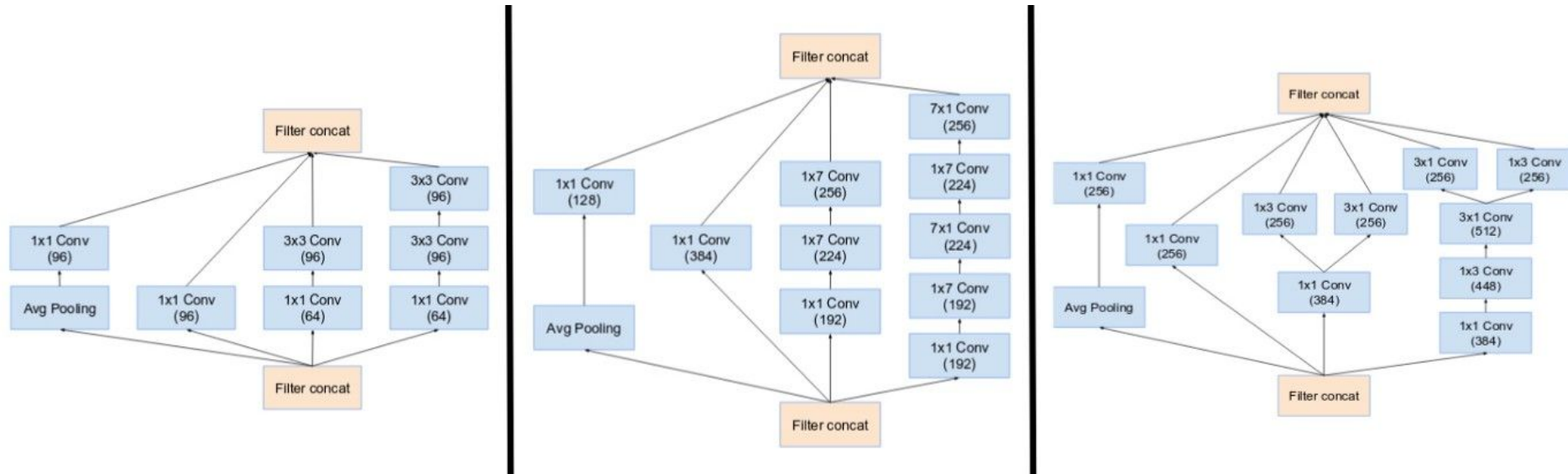
## Λύση

1. Το «στέλεχος-stem» (αρχικό σύνολο λειτουργιών που εκτελέστηκαν πριν από την εισαγωγή των Inception μπλοκ) του Inception v4 τροποποιήθηκε.



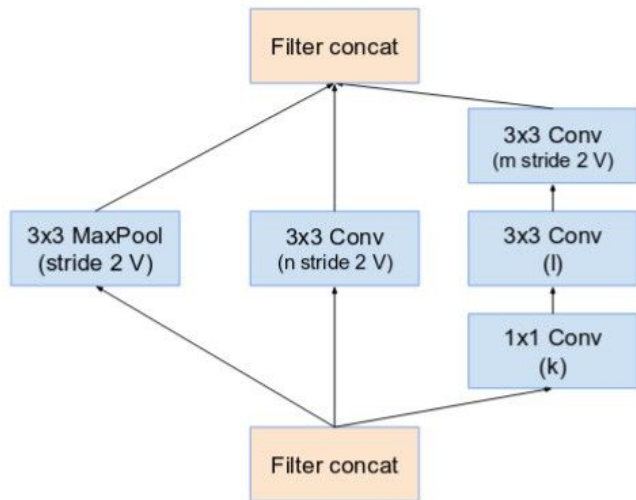
# CNN Architectures: Inception v4-Inception-ResNet

Τρεις κύριες ενότητες Inception, με το όνομα A, B και C που μοιάζουν πολύ με τα αντίστοιχα Inception v2

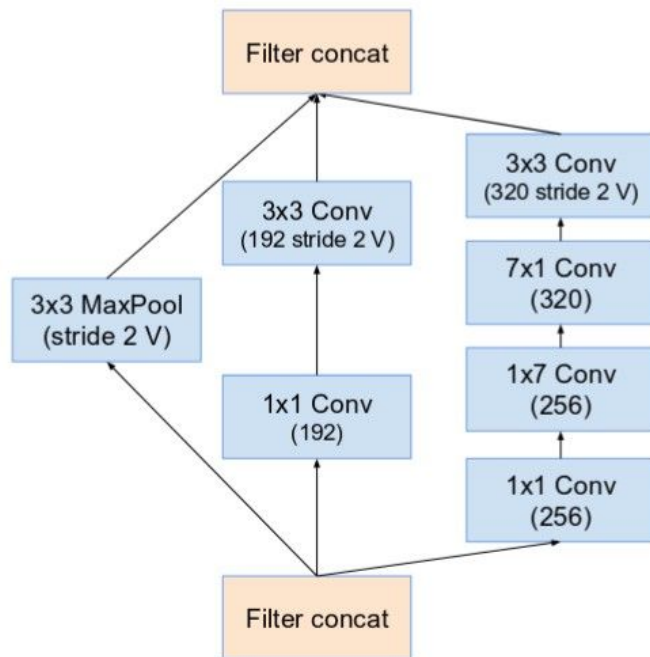


# CNN Architectures: Inception v4-Inception-ResNet

Το Inception v4 εισήγαγε εξειδικευμένα «Reduction Blocks» τα οποία χρησιμοποιούνται για την αλλαγή του πλάτους και του ύψους του πλέγματος.



Reduction Block A (μείωση μεγέθους 35x35 σε 17x17)



Reduction Block B (μείωση μεγέθους 17x17 σε 8x8)

# CNN Architectures: Inception-ResNet v1 & v2

Εμπνευσμένο από την απόδοση του ResNet, προτάθηκε μια υβριδική μονάδα inception.

- ❖ Υπάρχουν δύο υπο-εκδόσεις του Inception ResNet, δηλαδή v1 και v2.
  - Το Inception-ResNet v1 έχει υπολογιστικό κόστος παρόμοιο με αυτό του Inception v3.
  - Το Inception-ResNet v2 έχει υπολογιστικό κόστος παρόμοιο με αυτό του Inception v4.
- ❖ Έχουν διαφορετικά στελέχη-stems, όπως φαίνεται στην ενότητα Inception v4.
  - Και οι δύο υπο-εκδόσεις έχουν την ίδια δομή για τις ενότητες A, B, C και τα reduction blocks
  - Η μόνη διαφορά είναι οι ρυθμίσεις υπερ-παραμέτρων.

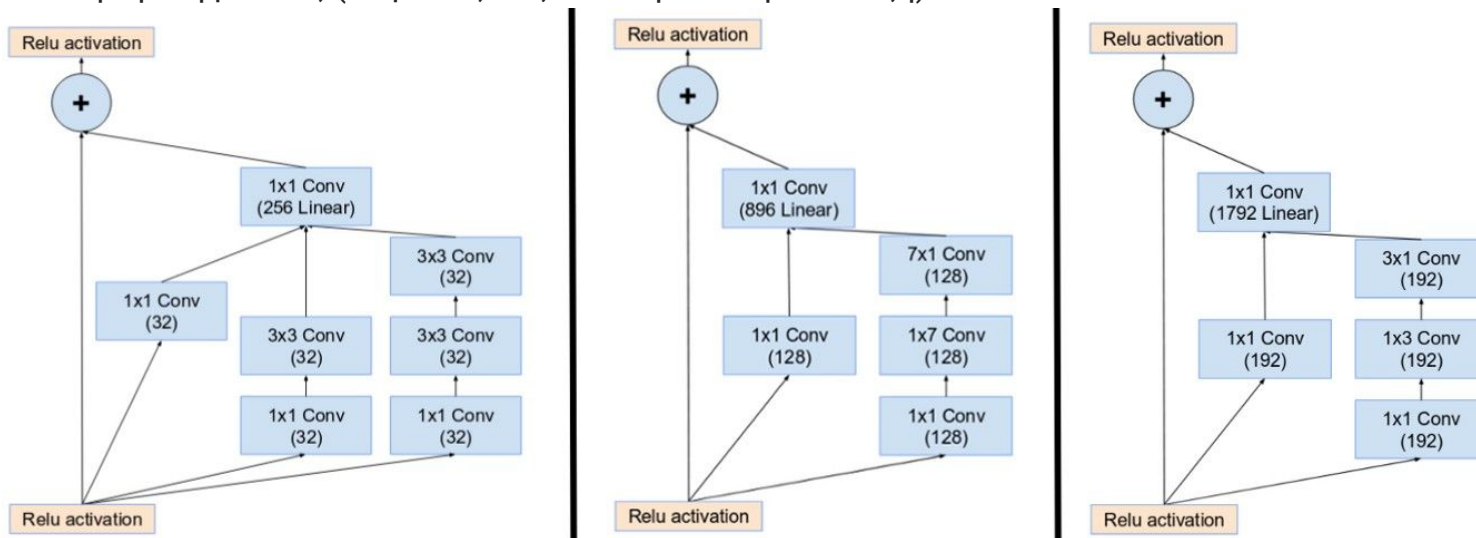
## Ιδέα

Εισαγωγή residual connections που προσθέτουν την έξοδο της λειτουργίας συνέλιξης της μονάδας inception στην είσοδο.

# CNN Architectures: Inception-ResNet v1 & v2

## Λύση

- Για το residual addition η είσοδος και η έξοδος μετά τη συνέλιξη πρέπει να έχουν τις ίδιες διαστάσεις.
  - Ως εκ τούτου, χρησιμοποιούμε συνελίξεις 1x1 μετά τις αρχικές συνελίξεις, για να ταιριάξουμε τα μεγέθη βάθους (Το βάθος αυξάνεται μετά τη συνέλιξη).



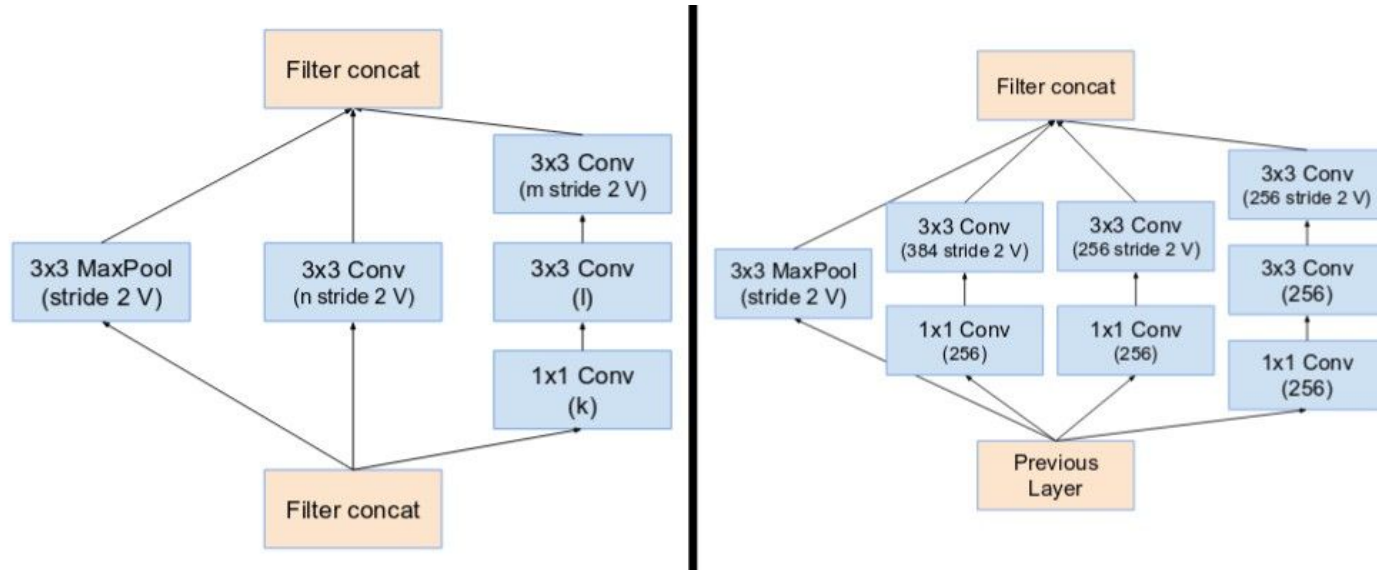
Inception modules A,B,C στο Inception ResNet

(Σημειώστε πώς το pooling layer αντικαταστάθηκε από την residual connection, καθώς και η πρόσθετη 1x1 συνέλιξη πριν από την πρόθεση (+))



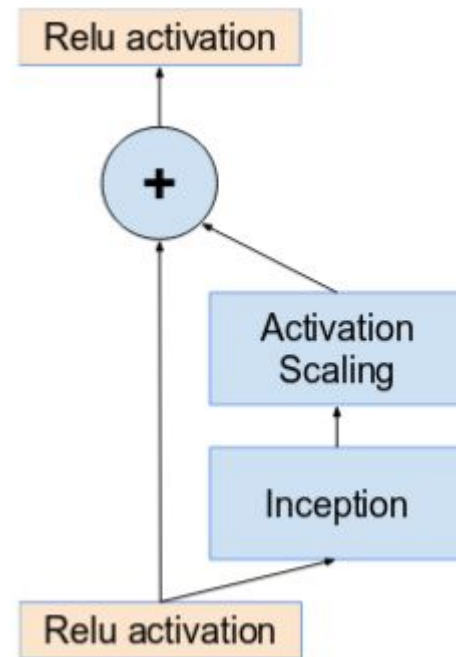
# CNN Architectures: Inception-ResNet v1 & v2

Η λειτουργία pooling εντός των κύριων inception μονάδων αντικαταστάθηκε υπέρ των residual connections.



# CNN Architectures: Inception-ResNet v1 & v2

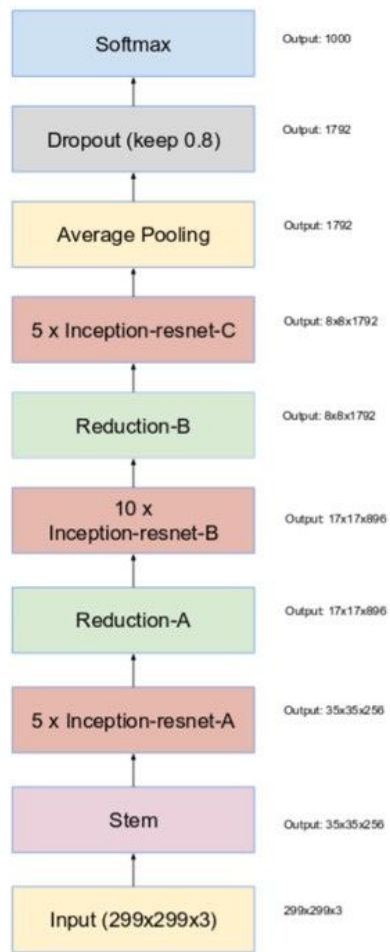
- Τα δίκτυα με residual units βαθύτερα στην αρχιτεκτονική προκάλεσαν το δίκτυο να "πεθάνει" εάν ο αριθμός των φίλτρων ξεπερνούσε τα 1000.
  - Ως εκ τούτου, για να αυξηθεί η σταθερότητα, οι συγγραφείς κλιμάκωσαν τις υπολειπόμενες ενεργοποιήσεις κατά μια τιμή περίπου από 0,1 έως 0,3.



# Inception v4



# Inception-ResNet

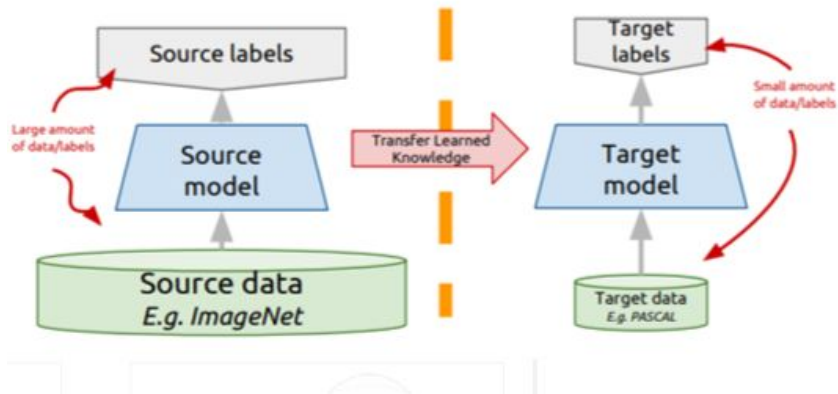


Τα μοντέλα Inception-ResNet επιτυγχάνουν υψηλότερη ακρίβεια σε λιγότερες εποχές.

# Transfer Learning για Deep Learning

## Ορισμός

Με δεδομένη μια εργασία Transfer Learning που ορίζεται από  $\langle D_s, T_s, D_t, T_t, f_T(\cdot) \rangle$ , η μεταφορά μάθησης (transfer learning) στοχεύει στη μάθηση της μη γραμμικής συνάρτησης  $f_T$  που αντικατοπτρίζει ένα βαθύ νευρωνικό δίκτυο.



# Στρατηγικές Deep Transfer Learning

→ Προεκπαιδευμένα μοντέλα για εξαγωγή χαρακτηριστικών

Off-the-shelf Pre-trained Models as fixed Feature Extractors

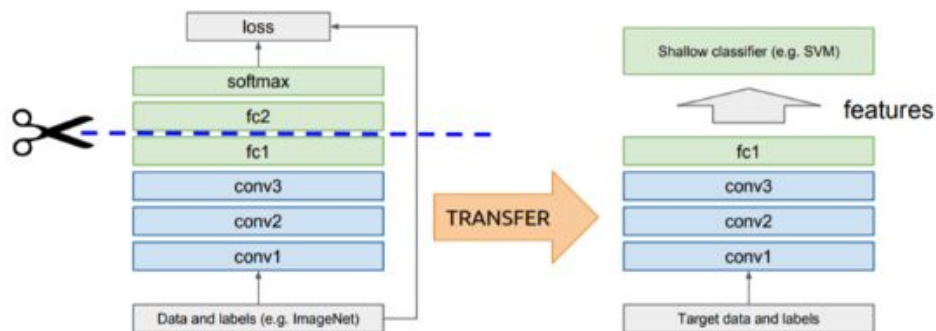
→ Ακριβής προσαρμογή προεκπαιδευμένων μοντέλων

Fine Tuning Off-the-shelf Pre-trained Models

# Προεκπαιδευμένα μοντέλα για εξαγωγή χαρακτηριστικών

- Η έξοδος μετά από κάποιο επίπεδο ενός δικτύου βαθιάς μάθησης, που εκπαιδεύτηκε σε διαφορετική εργασία ( $T_s \neq T_t$ ), χρησιμοποιείται ως γενικευμένος ανιχνευτής χαρακτηριστικών.
- Εκπαίδευση νέου μοντέλου (π.χ. SVM) με μεταφορά αυτών των χαρακτηριστικών.

Assumes that  $D_S = D_T$

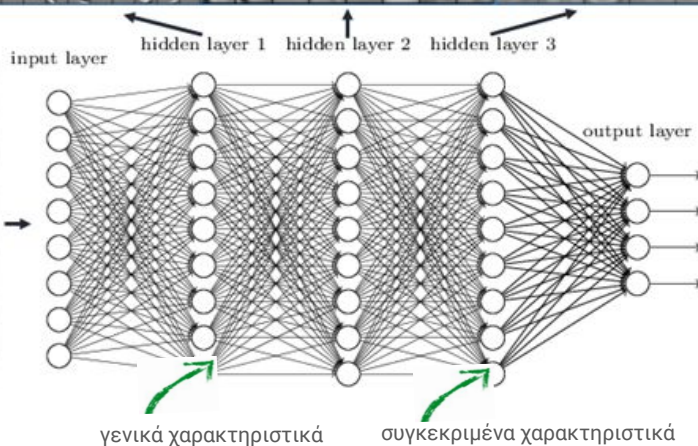
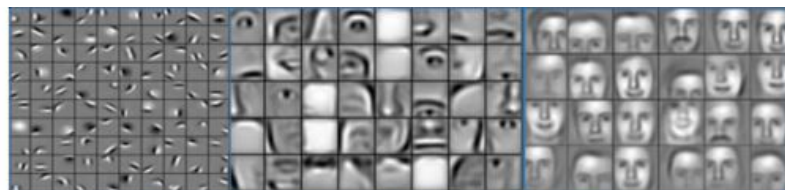


Transfer Learning with Pre-trained Deep Learning Models as Feature Extractors

# Ακριβής προσαρμογή προεκπαιδευμένων μοντέλων

Δεν αντικαθιστούμε απλώς το τελικό επίπεδο (για ταξινόμηση / παλινδρόμηση), αλλά επανεκπαιδεύουμε επιλεκτικά ορισμένα από τα προηγούμενα επίπεδα.

Deep neural networks learn hierarchical feature representations

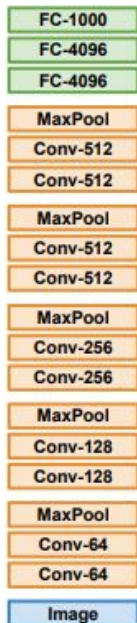


# Παράδειγμα: ConvNet για εξαγωγή χαρακτηριστικών

Donahue et al, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition", ICML 2014  
Razavian et al, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition", CVPR Workshops 2014

## Transfer Learning with CNNs

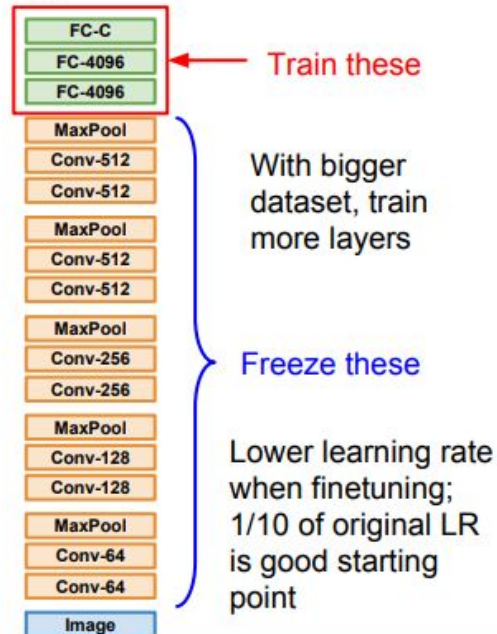
1. Train on Imagenet



2. Small Dataset (C classes)



3. Bigger dataset



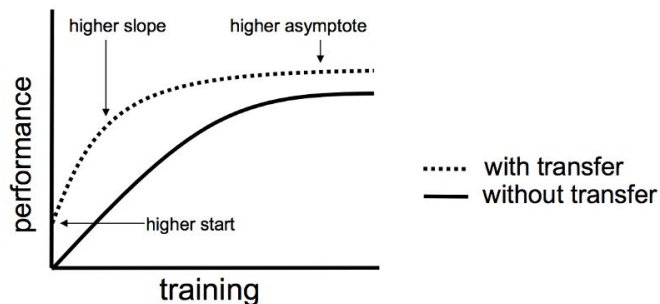


# Πρακτικές συμβουλές

→ Περιορισμοί από προκατασκευασμένα μοντέλα.

- ◆ Η χρήση ενός προκαθορισμένου δίκτυου, ενδέχεται να είναι δεσμευτική ως προς την αρχιτεκτονική που μπορείτε να χρησιμοποιήσετε για το νέο σύνολο δεδομένων μας.
  - π.χ. δεν μπορείτε να αφαιρέσετε αυθαίρετα Conv επίπεδα από το προκαθορισμένο δίκτυο.
- ◆ Συνήθως χρησιμοποιούμε μικρότερο learning rate για τα ρυθμισμένα βάρη ConvNet, σε σύγκριση με τα (τυχαία αρχικοποιημένα) βάρη που θα χρησιμοποιούσαμε για το νέο γραμμικό ταξινομητή που υπολογίζει τα βάρη ταξινόμησης του νέου συνόλου δεδομένων μας.
  - Αυτό συμβαίνει επειδή περιμένουμε ότι τα ρυθμισμένα βάρη ConvNet είναι σχετικά καλά, επομένως δεν θέλουμε να τα παραμορφώσουμε πολύ γρήγορα και πάρα πολύ.

# Κανόνες αξιολόγησης για χρήση TL



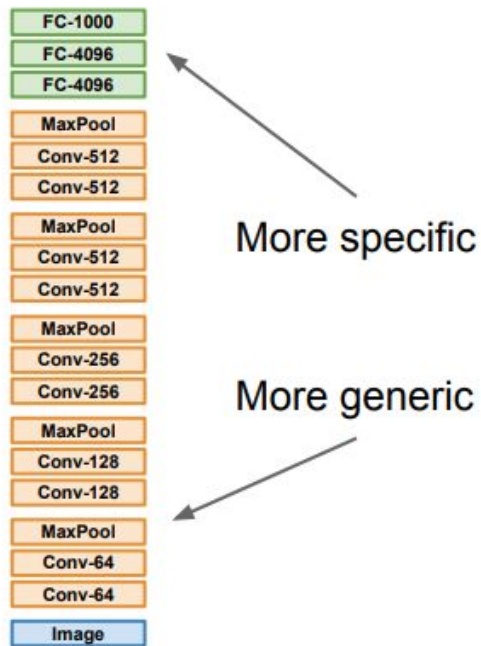
Για να αξιολογήσουμε αν ένα μοντέλο μπορεί να μεταφερθεί σε άλλη εργασία πρέπει να λάβουμε υπόψιν μας τα εξής:

**Higher start:** Η αρχική ικανότητα (πριν τελειοποιήσετε το μοντέλο) στο μοντέλο προέλευσης είναι υψηλότερη από ό, τι διαφορετικά θα ήταν.

**Higher slope:** Ο ρυθμός βελτίωσης της ικανότητας κατά τη διάρκεια της εκπαίδευσης του μοντέλου προέλευσης είναι πιο απότομος από ότι διαφορετικά θα ήταν.

**Higher asymptote :** Η συγκλίνουσα ικανότητα του εκπαιδευμένου μοντέλου είναι καλύτερη από ό, τι διαφορετικά θα ήταν.

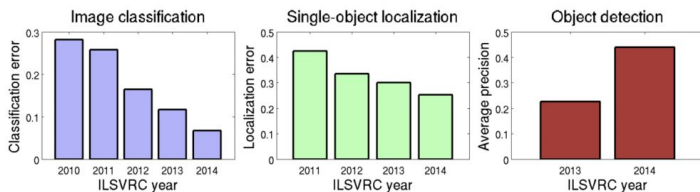
# Πρακτικές συμβουλές βελτίωσης



	<b>very similar dataset</b>	<b>very different dataset</b>
<b>very little data</b>	Use Linear Classifier on top layer	You're in trouble... Try linear classifier from different stages
<b>quite a lot of data</b>	Finetune a few layers	Finetune a larger number of layers

# ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

- ILSVRC: ετήσιος διαγωνισμός που χρησιμοποιεί υποσύνολα από το σύνολο δεδομένων ImageNet για ανάπτυξη και συγκριτική αξιολόγηση αλγορίθμων τελευταίας τεχνολογίας.
- ImageNet: πολύ μεγάλη συλλογή χαρακτηρισμένων (Amazon Mechanical Turk Worker) φωτογραφιών για την ανάπτυξη αλγορίθμων όρασης υπολογιστή.
- Οι εργασίες του ILSVRC οδήγησαν σε σημαντικές αρχιτεκτονικές μοντέλων και τεχνικές σύνδεσης της όρασης υπολογιστή και της βαθιάς μάθησης



# Κατηγορίες

## Image classification

Πρόβλεψη των κατηγοριών των αντικειμένων που υπάρχουν στην εικόνα

## Single-object localization

Image classification + σχεδιασμός bounding box

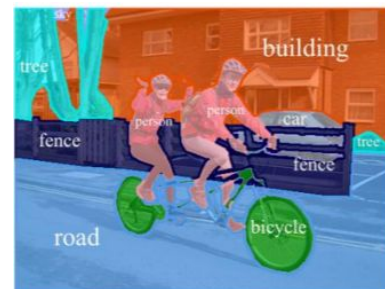
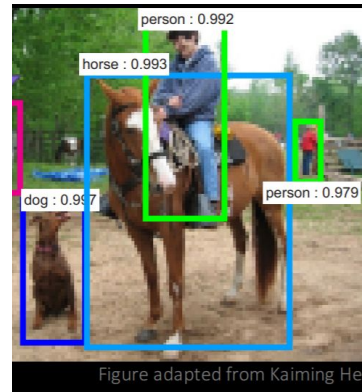


## Object detection

Image classification + σχεδιασμός bounding box γύρω από κάθε αντικείμενο.

## Object Segmentation

Ανίχνευση όλων των αντικειμένων της εικόνας σε pixel level



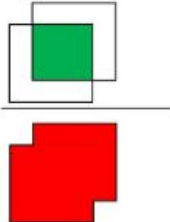
# Εισαγωγικές Έννοιες Ανίχνευσης Αντικειμένων

## 1. Μετρική της ομοιότητας μεταξύ δύο αντικειμένων

Σκοπός : σύγκριση και αξιολόγηση ομοιότητας αντικειμένων  
(π.χ. Ανιχνευμένο αντικείμενο με το αληθινό αντικείμενο (ground truth))

Ευρέως χρησιμοποιούμενο μέγεθος: Intersection over Union - IoU

Το IoU ορίζεται ως το εμβαδόν της τομής των δύο πλαισίων προς το εμβαδόν της ένωσής τους.

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} = \frac{\text{Area of Green Square}}{\text{Area of Red and Green Squares}}$$


## 2. Πλαίσιο Οριοθέτησης (Bounding Box)

- Ως πλαίσιο οριοθέτησης ενός αντικειμένου σε μία εικόνα ορίζεται το μικρότερο δυνατό ορθογώνιο τμήμα της εικόνας στο εσωτερικό του οποίου βρίσκεται ολόκληρο το αντικείμενο.
- Για την περιγραφή ενός πλαισίου οριοθέτησης είναι απαραίτητες 4 τιμές. π.χ.
  - οι συντεταγμένες της κάτω αριστερής και της πάνω δεξιάς γωνίας του
  - οι συντεταγμένες της πάνω αριστερής γωνίας, το πλάτος  $w$  και το ύψος  $h$  του πλαισίου
  - οι συντεταγμένες του κέντρου του πλαισίου, το πλάτος του  $w$  και το ύψος του  $h$



### 3. Περιοχή Ενδιαφέροντος (Region of Interest - ROI)

Ορίζεται μία ορθογώνια περιοχή της εικόνας εισόδου η οποία θεωρητικά είναι πιθανό να περιέχει ένα αντικείμενο.

Οι περιοχές αυτές μπορούν να υπολογιστούν:

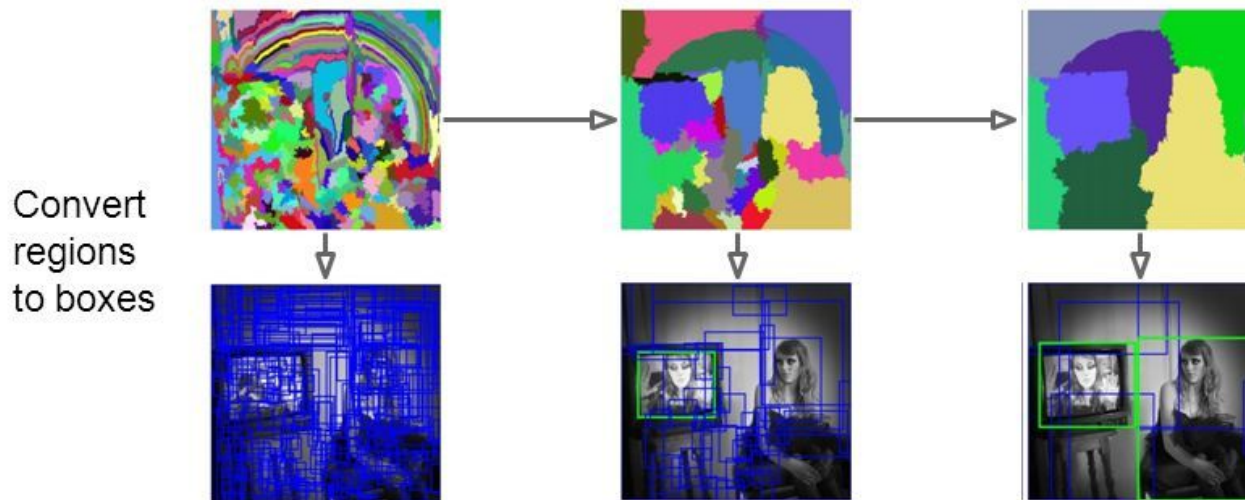
- με χρήση κάποιου εξωτερικού αλγορίθμου όπως το Selective Search ή το Edge Box detection,
- με χρήση ενός Δικτύου Προτάσεων Περιοχών (Region Proposal Network - RPN).



# Region-Based CNN: περιοχή ενδιαφέροντος (RoI)

## Region Proposals: Selective Search

Bottom-up segmentation, merging regions at multiple scales



## 4. Καταστολή μη μεγίστων (Non-Maximum Suppression)

Πρόβλημα: ύπαρξη πολλών προβλέψεων με μικρές διαφορές οι οποίες αντιστοιχούν στο ίδιο αντικείμενο.

Λύση: Καταστολή μη μεγίστων (Non-Maximum Suppression - NMS)

- Άπληστος (greedy) αλγόριθμος που συγχωνεύει αυτά τα αλληλοεπικαλυπτόμενα πλαίσια οριοθέτησης:
  - Ταξινομεί όλα τα πλαίσια οριοθέτησης σε αύξουσα σειρά ως προς την πιθανότητά τους να αντιστοιχούν σε κάποιο αντικείμενο.
  - Επιλέγει το πλαίσιο οριοθέτησης με τη μεγαλύτερη πιθανότητα και, συγκρίνοντάς το με κάθε ένα από τα Bounding Box με μικρότερη πιθανότητα, απορρίπτει όσα έχουν επικάλυψη IoU μικρότερη από μία προκαθορισμένη τιμή.

(Η τιμή αυτή αποτελεί μία από τις υπερπαραμέτρους του συστήματος) και επαναλαμβάνει τα βήματα όσες φορές είναι απαραίτητο.



# Object Detection: απλή προσέγγιση με CNN

1. Χωρίζουμε την εικόνα σε περιοχές και τροφοδοτούμε την κάθε περιοχή ως ξεχωριστή εικόνα στο CNN το οποίο τις ταξινομεί σε διάφορες τάξεις.
  2. Αφού χωρίσουμε κάθε περιοχή στην αντίστοιχη κλάση, μπορούμε να συνδυάσουμε όλες αυτές τις περιοχές για να πάρουμε την αρχική εικόνα με τα αντικείμενα που εντοπίστηκαν.
- (-) : Τα αντικείμενα μπορεί να έχουν διαφορετικά aspect ratios, χωρικές θέσεις και να έχουν υποστεί διάφορους μετασχηματισμούς
- (-) : Χρειάζεται πολύ μεγάλος αριθμός περιοχών, μεγάλη υπολογιστική ισχύ

**Λύση:** Region-based CNN

# Πρόβλημα: Ανίχνευσης Αντικειμένων

Σύνθεση δύο διαφορετικών προβλημάτων:

- ένα πρόβλημα ταξινόμησης και
- ένα πρόβλημα παλινδρόμησης, γνωστό και ως bounding box regression.

Με δεδομένη μία εικόνα εισόδου, πρέπει:

- να προβλεφθεί η τοποθεσία και η έκταση των αντικειμένων της εικόνας που ανήκουν σε ένα σύνολο προκαθορισμένων κλάσεων,
- να αποδοθεί η σωστή κλάση στο κάθε αντικείμενο.

Η τοποθεσία των αντικειμένων συνήθως εκφράζεται ως το ελάχιστο πλαίσιο οριοθέτησης που περικλείει εξ ολοκλήρου το αντικείμενο.

# Κατηγορίες μοντέλων ανίχνευση αντικειμένων

Χωρίζονται σε δύο κατηγορίες ως προς τη δομή τους:

- Τα μοντέλα ενός σταδίου (one-step models) χρησιμοποιούν :
  - ένα feed forward CNN για να προσδιορίσουν την τοποθεσία των αντικειμένων ενδιαφέροντος.
    - απλούστερα και ταχύτερα, αφού δεν παρέχουν region proposals
    - η απόδοσή τους είναι μειωμένη, κυρίως όταν απαιτείται και κατάτμηση της εικόνας

π.χ. YOLO, Multibox, AttentionNet, G-CNN

# Κατηγορίες μοντέλων ανίχνευση αντικειμένων

Τα μοντέλα δύο σταδίων (two-step models ή region-based models), χρησιμοποιούν

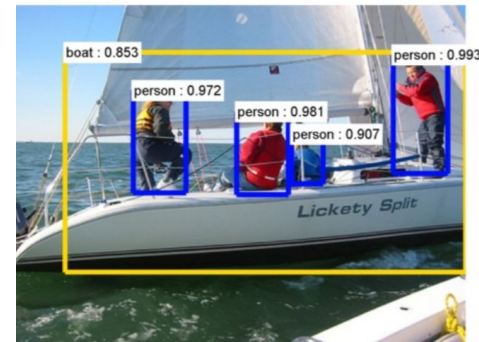
1. Έναν αλγόριθμο (π.χ. Selective Search) ή ένα μοντέλο (π.χ. region-based CNN) που δέχεται ως είσοδο την εικόνα και προτείνει διαφορετικές πιθανές περιοχές ενδιαφέροντος
2. Έναν feature extractor π.χ. CNN ώστε να υπολογιστεί ο χάρτης χαρακτηριστικών κάθε περιοχής ενδιαφέροντος ο οποίος δίνεται σε ένα πλήρως συνδεδεμένο υπεύθυνο για την ταξινόμηση.

π.χ. R-CNN, Fast R-CNN, FPN, Faster R-CNN

- έχουν αρκετές διαφορές αλλά περίπου κοινή δομή

# YOLO- You Only Look Once

You only look once (YOLO) at an image to predict what objects are present and where they are present.

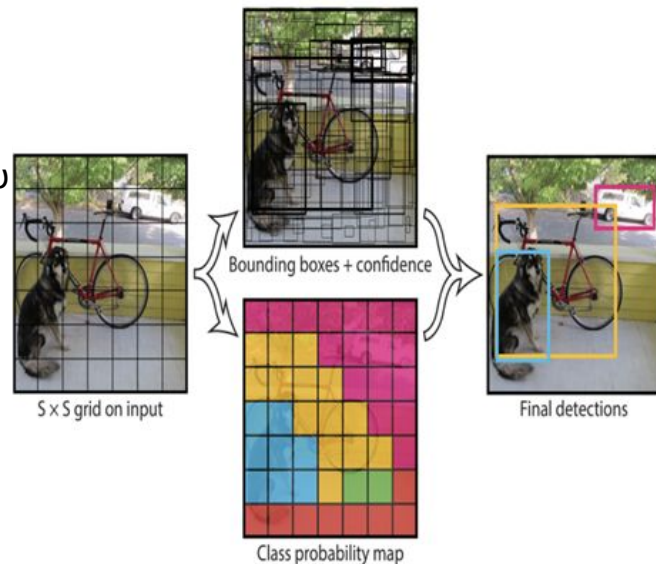


- Χρησιμοποιεί ένα απλό CNN και βλέπει ολόκληρη την εικόνα κατά τη διάρκεια του training και του validation, οπότε κωδικοποιεί σιωπηρά πληροφορίες για τις classes καθώς και τις εμφανίσεις τους, σε αντίθεση με τις τεχνικές sliding window ή region-based (κάνοντας έτσι λιγότερο από το ήμισυ του αριθμού των σφαλμάτων σε σύγκριση με το Fast R-CNN).
- Το YOLO χρησιμοποιεί features από ολόκληρη την εικόνα για να προβλέψει κάθε bounding box
- Προβλέπει επίσης όλα τα bounding box σε όλες τις classes για μια εικόνα ταυτόχρονα με τις αντίστοιχες πιθανότητες
- Αντιμετωπίζει την ανίχνευση ως πρόβλημα παλινδρόμησης
- Εξαιρετικά γρήγορος και ακριβής αλγόριθμος

[You Only Look Once: Unified, Real-Time Object Detection](#) Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi

# Λειτουργία YOLO- You Only Look Once

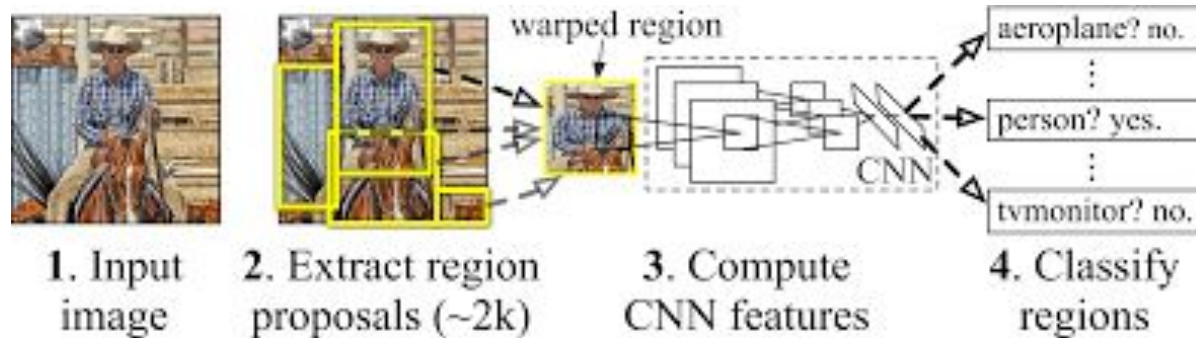
- Παίρνει μια εικόνα και τη χωρίζει σε πλέγμα  $S \times S$ .
- Κάθε κελί πλέγματος προβλέπει μόνο ένα αντικείμενο.
- Η ταξινόμηση εικόνας και ο εντοπισμός εφαρμόζονται σε κάθε κελί του πλέγματος.
- Εάν το κέντρο ενός αντικειμένου πέσει σε ένα κελί πλέγματος, αυτό το κελί πλέγματος είναι υπεύθυνο για την ανίχνευση αυτού του αντικειμένου.
- Κάθε ένα από τα κελιά πλέγματος προβλέπει bounding boxes  $B$  με βαθμολογίες εμπιστοσύνης για αυτά τα bounding boxes
  - Οι βαθμολογίες εμπιστοσύνης αντικατοπτρίζουν το πόσο σίγουρο είναι το μοντέλο ότι το πλέγμα περιέχει ένα αντικείμενο και πόσο ακριβές πιστεύει ότι το πλαίσιο είναι αυτό που προβλέπει.
  - Εάν δεν υπάρχουν αντικείμενα, τότε οι βαθμολογίες εμπιστοσύνης θα είναι 0
- Bounding box όταν ένα αντικείμενο υπάρχει στο κελί πλέγματος
- Πιθανότητα για class  $C$



y =	pc	0
	bx	?
	by	?
	bh	?
	bw	?
	c1	?
	c2	?
c3	?	



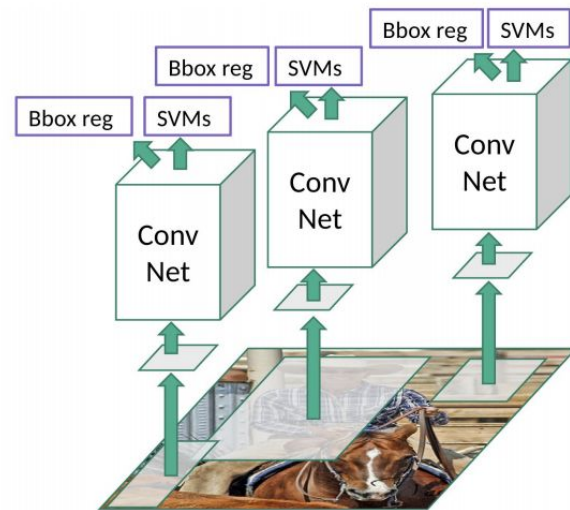
# Region-Based Convolutional Neural Network: R-CNN



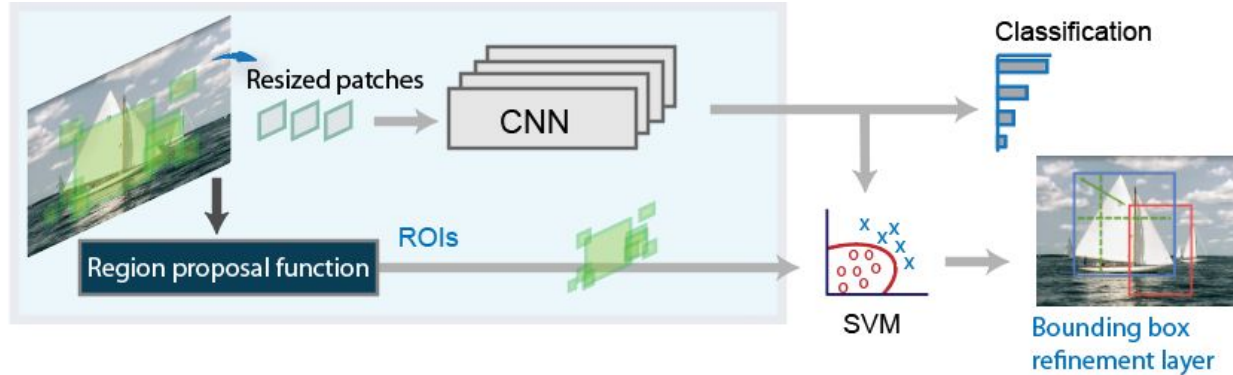
1. Χρησιμοποιεί τον αλγόριθμο Selective Search και παράγει 2000 προτάσεις περιοχών ανά εικόνα.
2. Η **κάθε περιοχή**, μετά από προσαρμογή του μεγέθους της, δίνεται ως είσοδος σε ένα προεκπαιδευμένο CNN
3. Η έξοδος από το ConvNet είναι ένα διάνυσμα 4096 χαρακτηριστικών
4. Εκπαιδευόμε το τελευταίο επίπεδο του δικτύου **της κάθε περιοχής** ένα ταξινομητή με βάση τον αριθμό των κατηγοριών που πρέπει να εντοπιστούν

# Region-Based Convolutional Neural Network: R-CNN

5. Αφού αποκτήσουμε τις περιοχές, εκπαιδεύουμε ένα δυαδικό SVM ανά περιοχή για να ταξινομήσουμε αντικείμενα και φόντο.
6. Τέλος, εκπαιδεύουμε ένα μοντέλο γραμμικής παλινδρόμησης για τη δημιουργία αυστηρότερων bounding boxes για κάθε αναγνωρισμένο αντικείμενο στην εικόνα.



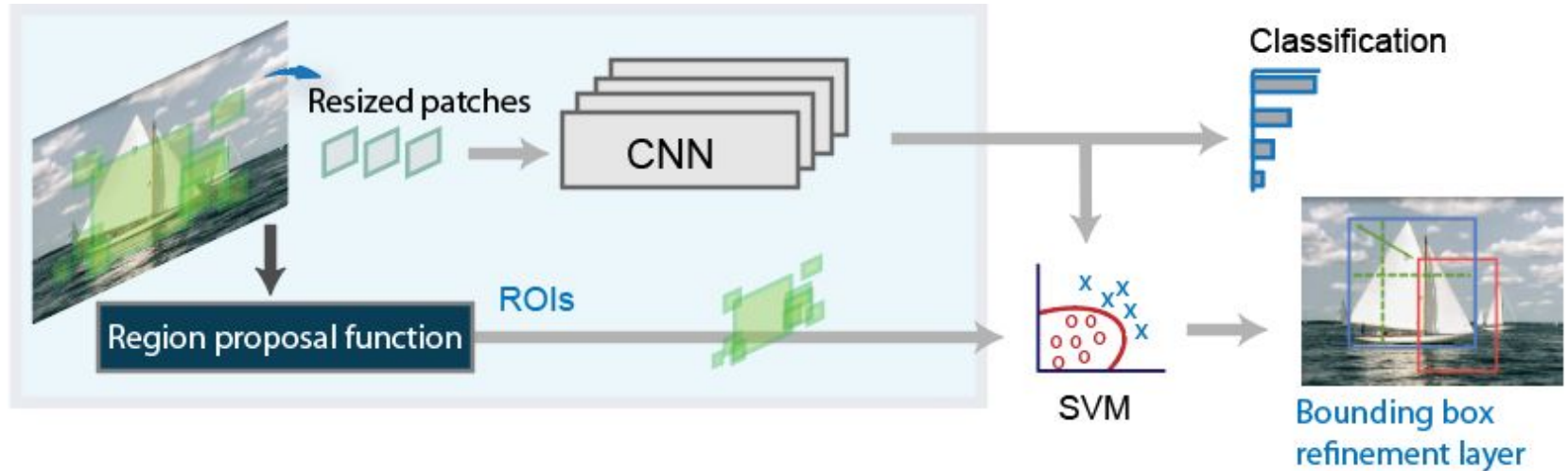
# Region-Based Convolutional Neural Network: R-CNN



## Μειονεκτήματα

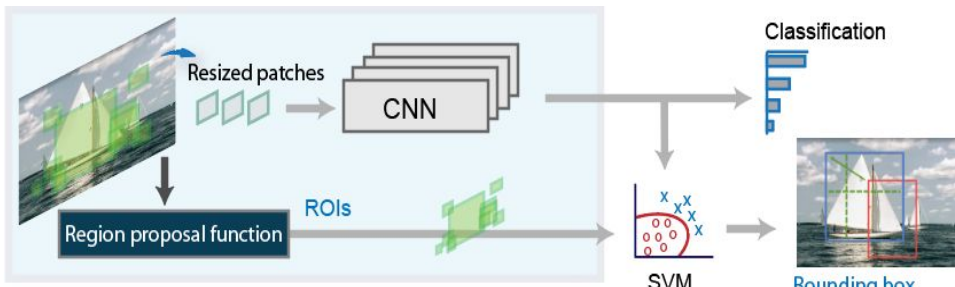
- Οι 2000 προτάσεις περιοχών ανά εικόνα → πολύ μεγάλο χρόνο εκπαίδευσης
- Ο χρόνος του testing είναι απαγορευτικά μεγάλος → μη χρήση του μοντέλου για εφαρμογές πραγματικού χρόνου.
- Προκαθορισμένη συμπεριφορά του αλγορίθμου Selective Search → η αναγνώριση δε βελτιώνεται μέσω εκπαίδευσης.

# Region-Based Convolutional Neural Network: R-CNN



1. Χρήση προεκπαιδευμένου CNN,
2. Εκπαιδύουμε το τελευταίο επίπεδο του δικτύου με βάση τον αριθμό των κατηγοριών που πρέπει να εντοπιστούν,
3. Λαμβάνουμε την περιοχή ενδιαφέροντος (RoI) για κάθε εικόνα,
4. Μετατρέπουμε όλες αυτές τις περιοχές έτσι ώστε να ταιριάζουν με το μέγεθος εισόδου CNN,
5. Αφού αποκτήσουμε τις περιοχές, εκπαιδύουμε το SVM για να ταξινομήσουμε αντικείμενα και φόντο. Για κάθε class, εκπαιδύουμε ένα δυαδικό SVM.
6. Τέλος, εκπαιδύουμε ένα μοντέλο γραμμικής παλινδρόμησης για τη δημιουργία αυστηρότερων bounding boxes για κάθε αναγνωρισμένο αντικείμενο στην εικόνα.

# Region-Based Convolutional Neural Network: R-CNN



Bounding Box (x,y,w,h)

**Learn a target transformation**

P: Predicted

G: Target

$$\begin{aligned} P^i &= (P_x^i, P_y^i, P_w^i, P_h^i) \\ G &= (G_x, G_y, G_w, G_h) \end{aligned} \quad (1)$$

**Learn ground-truth transformations**

- Scale-invariant translation of the center of P (x,y)
- Specify log space transformations of the width w and height h.

$$\begin{aligned} t_x &= (G_x - P_x) / P_w \\ t_y &= (G_y - P_y) / P_h \\ t_w &= \log(G_w / P_w) \\ t_h &= \log(G_h / P_h). \end{aligned}$$

(2)

**Learn predicted transformation  $d\Box(P)$**

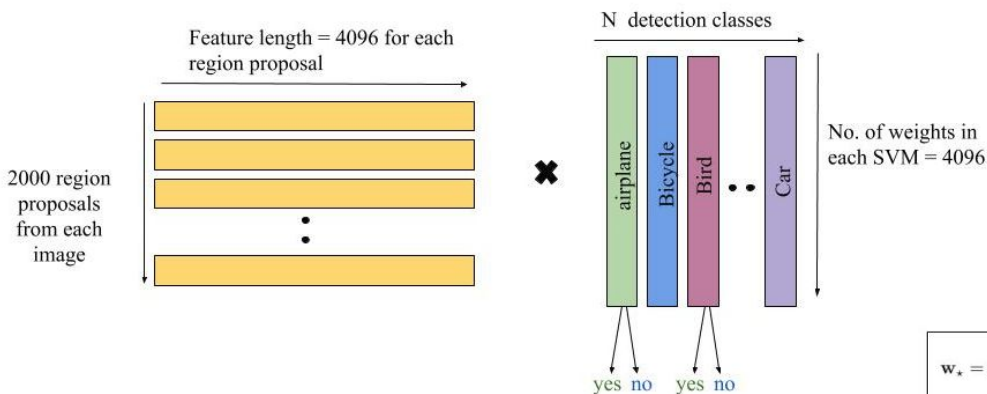
- $\hat{G}$ : corrected predicted box calculated
- $d\Box(P) = w\Box^T \Phi_5(P)$   
 $w\Box$ : learnable model parameters.  
 is learnt by optimizing the regularized least-squares objective function  
 $\Phi_5$ : is dependent on the actual image features.

$$\begin{aligned} \hat{G}_x &= P_w d_x(P) + P_x \\ \hat{G}_y &= P_h d_y(P) + P_y \\ \hat{G}_w &= P_w \exp(d_w(P)) \\ \hat{G}_h &= P_h \exp(d_h(P)). \end{aligned} \quad (3)$$

$$d_*(P) = \mathbf{w}_*^T \Phi_5(P)$$

$$\mathbf{w}_* = \underset{\tilde{\mathbf{w}}_*}{\operatorname{argmin}} \sum_i (t_i^* - \tilde{\mathbf{w}}_*^T \Phi_5(P^i))^2 + \lambda \|\tilde{\mathbf{w}}_*\|^2$$

(4)



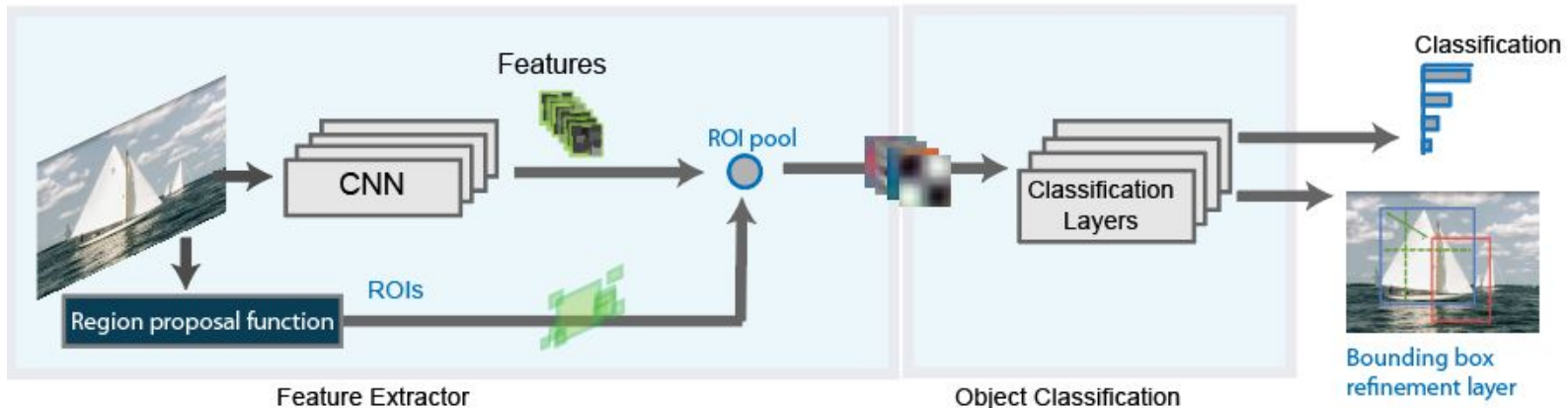
Ο αλγόριθμος μαθαίνει από ένα προβλεπόμενο πλαίσιο P μόνο αν βρίσκεται κοντά σε τουλάχιστον ένα πλαίσιο groundtruth.

Κάθε προβλεπόμενο πλαίσιο P αντιστοιχίζεται στο groundtruth του επιλέγοντας το πλαίσιο groundtruth με το οποίο έχει μέγιστη επικάλυψη (υπό την προϋπόθεση ότι έχει επικάλυψη IoU > 0,5).

# Region-Based CNN: Fast R-CNN

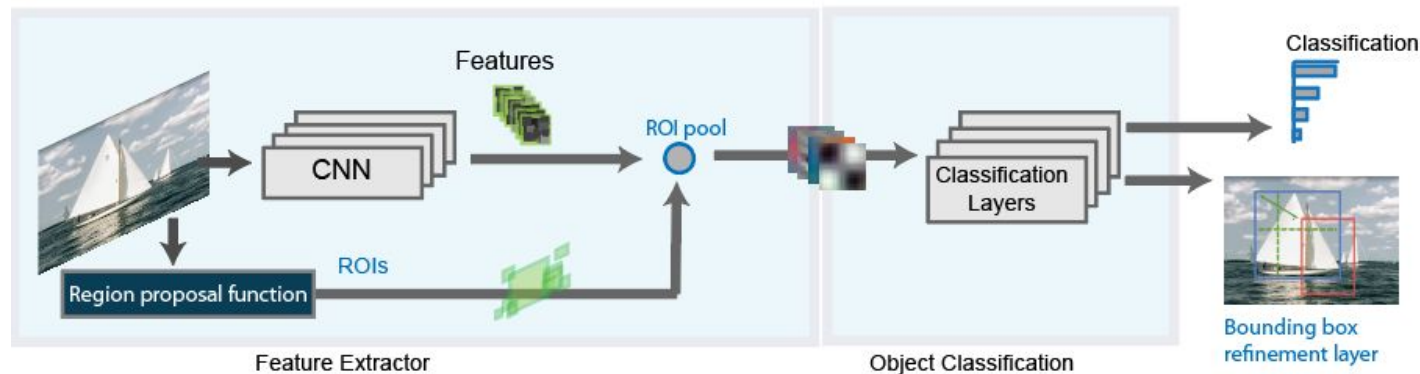
- Βελτιώνει σημαντικά την ταχύτητα του μοντέλου αλλάζοντας απλώς τη σειρά των επιπέδων του.
- Αντί να δίνεται ως είσοδος στο CNN κάθε μία από τις περιοχές ενδιαφέροντος, το CNN εξάγει τα χαρακτηριστικά ολόκληρης της εικόνας σε μορφή χάρτη χαρακτηριστικών, και στη συνέχεια για κάθε περιοχή απομονώνεται το αντίστοιχο τμήμα.

Με αυτό τον τρόπο, η εξαγωγή των χαρακτηριστικών γίνεται μόνο μία φορά αντί για 2000, γεγονός που είναι προφανές ότι βελτιώνει κατά πολύ την ταχύτητα της εκπαίδευσης και της πρόβλεψης.



# Region-Based CNN: Fast R-CNN

1. Επεξεργάζεται ολόκληρη την εικόνα,
2. Ενώ ο R-CNN detector κατηγοριοποιεί κάθε περιοχή, ο Fast R-CNN συγκεντρώνει τα features maps από το CNN που αντιστοιχούν σε κάθε προτεινόμενη περιοχή (region proposal),
3. Κάθε περιοχή περνά από ένα fully connected network και ένα softmax layer δίνει τις κατηγορίες εξόδου.
4. Μαζί με το στρώμα softmax, χρησιμοποιείται επίσης ένα linear regression layer, παράλληλα για την παραγωγή των συντεταγμένων του bounding box για προβλεπόμενες κατηγορίες.



# Region-Based CNN: Faster R-CNN

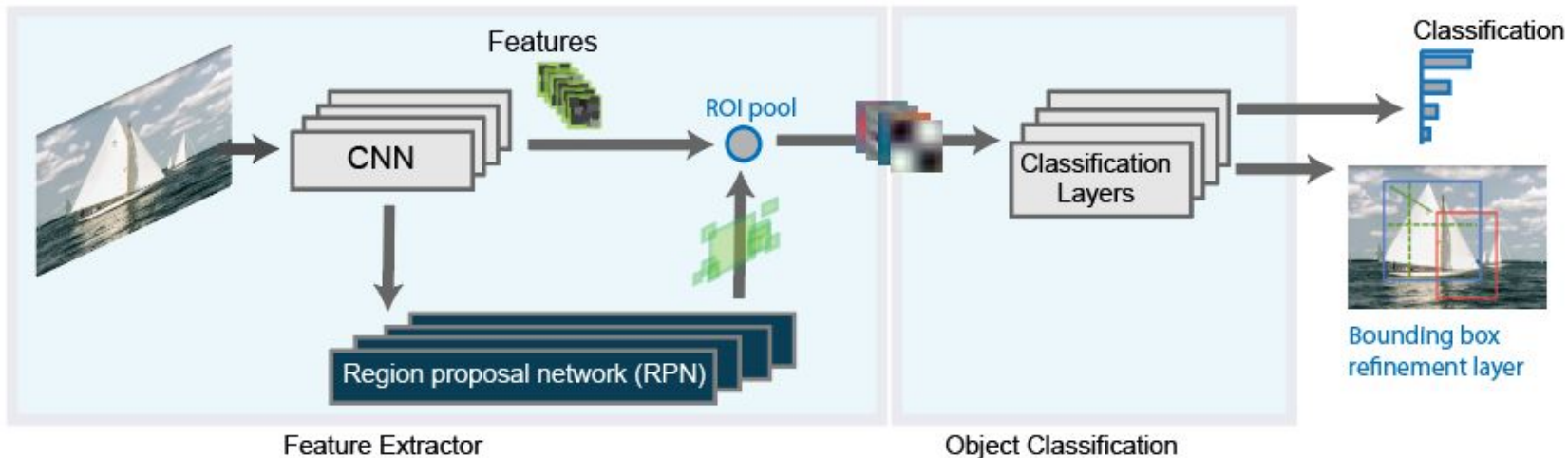
[Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun](#)

- Πρότειναν την αντικατάστασή του Selective Search, από τον δικό τους αλγόριθμο:  
**Δίκτυο Πρότασης Περιοχών**
  - Συγκεκριμένα, αντιλήφθηκαν ότι ο χάρτης χαρακτηριστικών που παράγεται από το συνελικτικό τμήμα του Fast R-CNN μπορεί να χρησιμοποιηθεί αποτελεσματικά και για το πρόβλημα της πρότασης περιοχών, αντικαθιστώντας τις αργές μεθόδους όπως η Selective Search με ένα εκπαιδεύσιμο νευρωνικό δίκτυο.

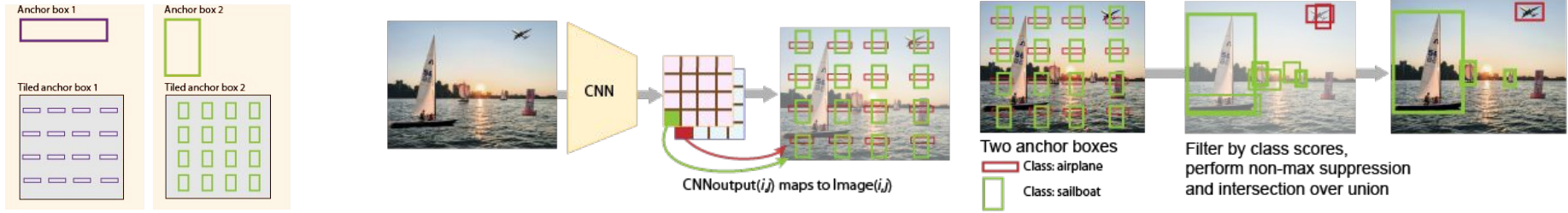


# Region-Based CNN: Faster R-CNN

- Ο Faster R-CNN προσθέτει ένα region proposal network (RPN) για να δημιουργήσει region proposals απευθείας μέσω δικτύου
- Το RPN χρησιμοποιεί Anchor Boxes για το Object Detection



# Region-Based CNN: Faster R-CNN → Anchor Boxes



Τα anchor boxes είναι ένα σύνολο από προκαθορισμένα bounding boxes με συγκεκριμένο πλάτος και ύψος:

- ορίζονται για να καταγράψουν την κλίμακα και το λόγο διαστάσεων συγκεκριμένων κατηγοριών αντικειμένων που θέλουμε να εντοπίσουμε, (μπορούμε να έχουμε anchor boxes διαφορετικών μεγεθών)
- επιλέγονται συνήθως με βάση τα μεγέθη αντικειμένων στα training datasets,

Κατά τη διάρκεια της ανίχνευσης:

- τα predefined anchor boxes διατρέχουν την εικόνα,
- το δίκτυο προβλέπει την πιθανότητα και άλλα χαρακτηριστικά (background, IoU, offsets) για κάθε anchor box καθώς διατρέχει την εικόνα,
- επιστρέφεται ένα μοναδικό σύνολο προβλέψεων για κάθε καθορισμένο bounding box

Ο τελικός feature map αντιπροσωπεύει ανιχνεύσεις αντικειμένων για κάθε κατηγορία

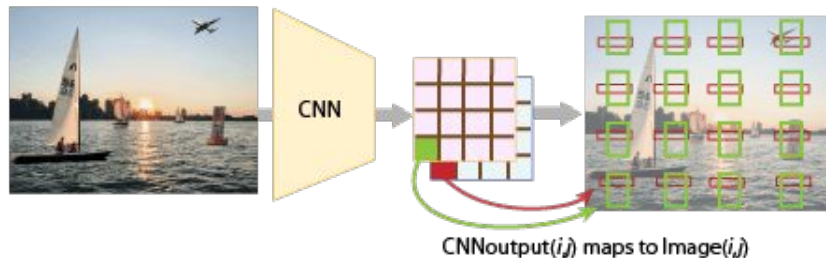
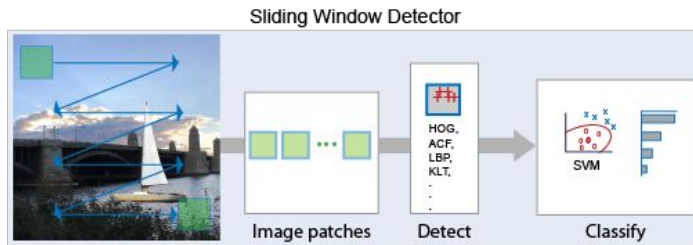
Η χρήση anchor boxes επιτρέπει σε ένα δίκτυο να ανιχνεύει πολλά αντικείμενα, αντικείμενα διαφορετικών κλιμάκων και αλληλεπικαλυπτόμενα αντικείμενα.

# Anchor Boxes

Με τα anchor boxes μπορούν να αξιολογηθούν όλες οι προβλέψεις αντικειμένων ταυτόχρονα.

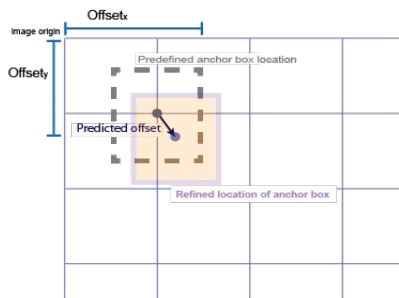
- Τα anchor boxes εξαλείφουν την ανάγκη σάρωσης μιας εικόνας με συρόμενο παράθυρο που υπολογίζει μια ξεχωριστή πρόβλεψη σε κάθε πιθανή θέση.
- Ένας ανιχνευτής αντικειμένων που χρησιμοποιεί anchor boxes μπορεί να επεξεργαστεί μια ολόκληρη εικόνα ταυτόχρονα (ιδανικό για CNN), καθιστώντας δυνατή την ανίχνευση αντικειμένων σε πραγματικό χρόνο σε αντίθεση με ανιχνευτές που χρησιμοποιούν συρόμενο παράθυρο (π.χ. HOG descriptor).

Χρησιμοποιώντας anchor boxes, σχεδιάζουμε αποτελεσματικούς ανιχνευτές αντικειμένων βαθιάς μάθησης, συμπεριλαμβάνοντας και τα τρία στάδια (ανίχνευση, κωδικοποίηση χαρακτηριστικών και ταξινόμηση) ενός ανιχνευτή αντικειμένων με βάση συρόμενα παράθυρα.



# Σφάλματα εντοπισμού και βελτίωση

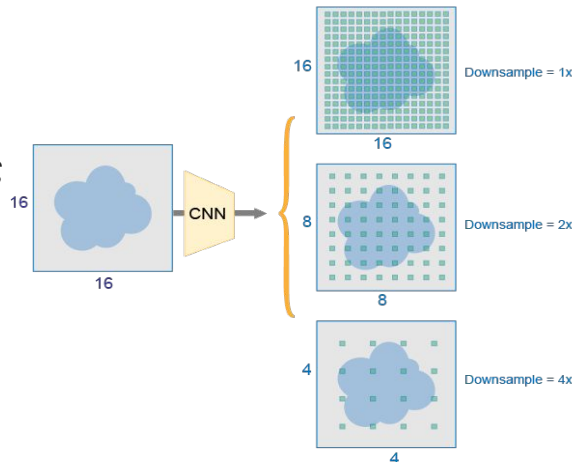
## Σφάλματα εντοπισμού



- Οι ανιχνευτές αντικειμένων βαθιάς μάθησης μαθαίνουν αντισταθμίσεις για να εφαρμόζονται σε κάθε Tiled Anchor Box που βελτιώνει τη θέση και το μέγεθος του Anchor Box.

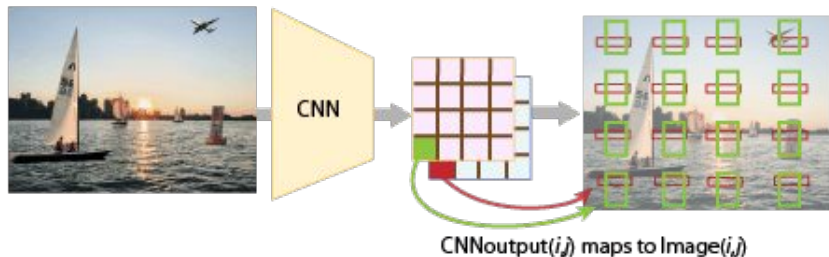
## Βελτίωση

- Η απόσταση μεταξύ των Anchor Boxes είναι συνάρτηση του ποσού της δειγματοληψίας που υπάρχει στο CNN (max Pooling 2d Layer και του stride του Conv Layer)
- Τα feature maps που παράγουν τα αρχικά επίπεδα του CNN έχουν υψηλότερη χωρική ανάλυση, αλλά μπορεί να εξαγάγουν λιγότερες σημασιολογικές πληροφορίες σε σύγκριση με τα επίπεδα που βρίσκονται πιο κάτω στο δίκτυο



# Πώς λειτουργούν τα Anchor Boxes

- Η θέση ενός Anchor Box καθορίζεται με αντιστοίχιση της θέσης της εξόδου του δικτύου “πάνω” στην εικόνα εισόδου.
- Η διαδικασία επαναλαμβάνεται για κάθε έξοδο δικτύου.
- Το αποτέλεσμα παράγει ένα σετ Anchor Boxes σε ολόκληρη την εικόνα.
- Κάθε Anchor Box αντιπροσωπεύει μια συγκεκριμένη πρόβλεψη μιας κλάσης.
  - Για παράδειγμα, έστω ότι υπάρχουν δύο Anchor Boxes για να κάνετε δύο προβλέψεις ανά τοποθεσία στην παρακάτω εικόνα.
    - Κάθε Anchor Box έχει Tiled Anchor boxes στην εικόνα.
    - Ο αριθμός των εξόδων δικτύου ισούται με τον αριθμό των Tiled Anchor Box
    - Το δίκτυο παράγει προβλέψεις για όλες τις εξόδους.



# Δημιουργία ανιχνευτών αντικειμένων

- Αφαιρούνται τα Tiled Anchor Boxes που ανήκουν στην κατηγορία φόντου και τα υπόλοιπα φιλτράρονται από τη βαθμολογία εμπιστοσύνης τους.
- Τα Anchor Boxes με τη μεγαλύτερη βαθμολογία εμπιστοσύνης επιλέγονται χρησιμοποιώντας non-max suppression (NMS).

---

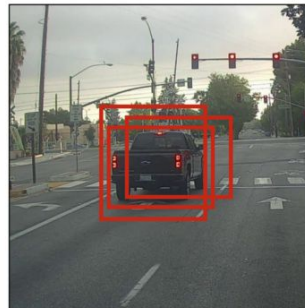
## Algorithm 1 Non-Max Suppression

---

```
1: procedure NMS( $B, c$ )
2:    $B_{nms} \leftarrow \emptyset$  Initialize empty set
3:   for  $b_i \in B$  do Iterate over all the boxes
4:      $discard \leftarrow \text{False}$  Take boolean variable and set it as false. This variable indicates whether b(i) should be kept or discarded
5:     for  $b_j \in B$  do Start another loop to compare with b(i)
6:       if  $\text{same}(b_i, b_j) > \lambda_{nms}$  then If both boxes having same IOU
7:         if  $\text{score}(c, b_j) > \text{score}(c, b_i)$  then
8:            $discard \leftarrow \text{True}$  Compare the scores. If score of b(i) is less than that of b(j), b(i) should be discarded, so set the flag to True.
9:         if not  $discard$  then Once b(i) is compared with all other boxes and still the discarded flag is False, then b(i) should be considered. So add it to the final list.
10:           $B_{nms} \leftarrow B_{nms} \cup b_i$ 
11:   return  $B_{nms}$  Do the same procedure for remaining boxes and return the final list
```

---

Before non-max suppression



Non-Max  
Suppression



After non-max suppression



# Δημιουργία ανιχνευτών αντικειμένων

## Μέγεθος Anchor Boxes

- Η επεξεργασία πολλαπλών κλιμάκων επιτρέπει στο δίκτυο να ανιχνεύει αντικείμενα διαφορετικού μεγέθους.
- Για να επιτευχθεί ανίχνευση πολλαπλών κλιμάκων, πρέπει να καθοριστούν Anchor Boxes διαφορετικού μεγέθους, όπως 64x64, 128x128 και 256x256.
- Καθορίζουμε μεγέθη που αντιπροσωπεύουν την κλίμακα και τον λόγο διαστάσεων των αντικειμένων στα δεδομένα εκπαίδευσης.
  - Χρήση της μέτρησης IoU
    - είναι αναλλοίωτη στο μέγεθος των Anchor Boxes.
    - οδηγεί σε ομαδοποίηση πλαισίων με παρόμοιες αναλογίες που οδηγεί σε εκτιμήσεις Anchor Boxes



Blue is predicted bounding box and red is ground truth bounding box



Intersection



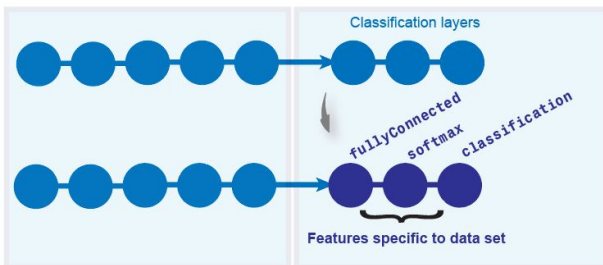
Union

# Region-Based CNN

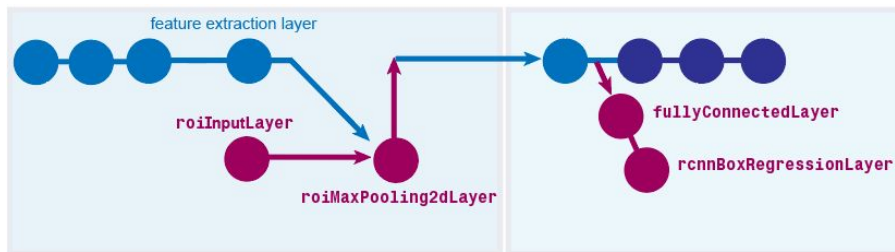
## Transfer Learning model

'alexnet', 'vgg16', 'vgg19', 'resnet50', 'resnet101', 'inceptionv3', 'googlenet', 'inceptionresnetv2', 'squeezeNet'

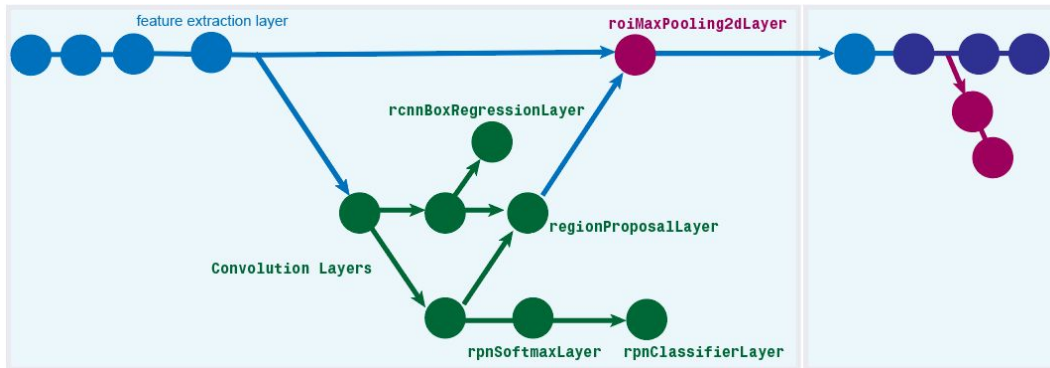
### R-CNN



### Fast R-CNN

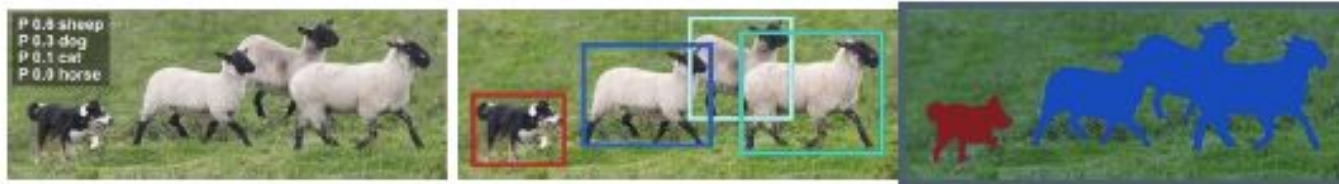


### Faster R-CNN





# Πρόβλημα Σημασιολογικής Κατάτμησης



(a) Ταξινόμηση Εικόνας

(b) Ανίχνευση Αντικειμένου

(c) Σημασιολογική Κατάτμηση

→ Ταξινόμηση σε επίπεδο εικονοστοιχείου

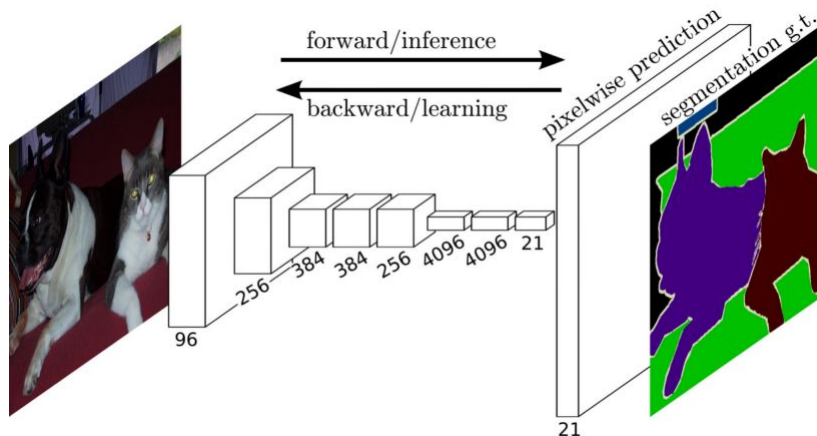
- ◆ Ανάθεση κλάσης σε κάθε εικονοστοιχείο της εικόνας
- ◆ Μέσω χωρικής ανάλυσης ενός εικονοστοιχείου.

→ Αφελής πρώτη προσέγγιση: Υλοποίηση ενός μοντέλου με διαδοχικά συνελκτικά επίπεδα, του οποίου η έξοδος θα είχε την ίδια διάσταση με την είσοδο.

- ◆ το κάθε εικονοστοιχείο της εξόδου θα αποτελούσε την πρόβλεψη για την κλάση του αντίστοιχου εικονοστοιχείου της αρχικής εικόνας.
- ◆ απαγορευτική υπολογιστική πολυπλοκότητα.

# Πρόβλημα Σημασιολογικής Κατάτμησης

Συνηθισμένες προσεγγίσεις: encoder-decoder.



1. Η διάσταση της εικόνας μειώνεται αρχικά (encoder), παράγοντας χαμηλότερης ανάλυσης χάρτες χαρακτηριστικών οι οποίοι έχουν πολύ καλά αποτελέσματα για την ταξινόμηση μεταξύ των κλάσεων,
2. Στη συνέχεια αυξάνεται και πάλι (decoder), μέχρι να προκύψει ο τελικός χάρτης κατάτμησης.

# Πρόβλημα Σημασιολογικής Κατάτμησης

## Πλήρως Συνελικτικό Δίκτυο (Fully Convolutional Network - FCN)

- CNN : Fully Connected Layers → Fully Convolutional Network
  - Κωδικοποιητής: Εξαγάγει τα χαρακτηριστικά και είναι εκπαιδευμένος με βάση το πρόβλημα της ταξινόμησης.
  - Αποκωδικοποιητής: Προβάλλει το χάρτη χαρακτηριστικών χαμηλής ανάλυσης που προέκυψε από τον κωδικοποιητή στην αρχική εικόνα.
    - Αποτελείται από μία σειρά συνελίξεις (backwards convolutions ή deconvolutions)
      - πραγματοποιούν αύξηση της χωρικής ανάλυσης με χρήση διγραμμικής παρεμβολής (bilinear interpolation).
    - Κάνει χρήση παρακαμπτήριων συνδέσεων (skip connections), που εκμεταλλεύονται τις παρόμοιες διαστάσεις των εκατέρωθεν επιπέδων του FCN και συνδέουν σειριακά τους χάρτες ενεργοποίησης του κωδικοποιητή με την αντίστοιχη δομή που προκύπτει μετά από κάθε αποσυνέλιξη.

# Πρόβλημα Κατάτμησης Στιγμιοτύπων (Instance Segmentation)

**Κατάτμηση Στιγμιοτύπων = Ανίχνευσης Αντικειμένων + Σημασιολογικής Κατάτμηση**

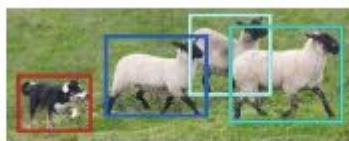
Στοχεύει στον εντοπισμό των διαφορετικών αντικειμένων σε μία εικόνα όχι με χρήση πλαισίων οριοθέτησης αλλά με ακρίβεια εικονοστοιχείου.

→ Κάθε εικονοστοιχείο ταξινομείται σε μία κλάση, όπως στη Σημασιολογική Κατάτμηση, αλλά τα διαφορετικά αντικείμενα θα έχουν άλλη μάσκα, ακόμα κι αν ανήκουν στην ίδια κλάση.

[B. Hariharan, P. Arbelaez, R. B. Girshick, and J. Malik. Simultaneous Detection and Segmentation. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, Computer Vision – ECCV 2014, volume 8695 of Lecture Notes in Computer Science. Springer, Cham, 2014.](#)



(a) Ταξινόμηση Εικόνας



(b) Ανίχνευση Αντικειμένου



(c) Σημασιολογική Κατάτμηση

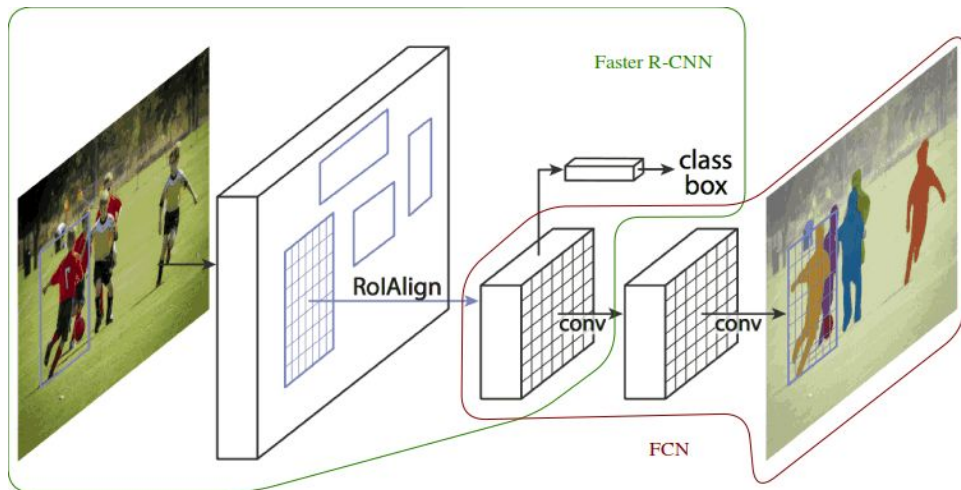


(d) Κατάτμηση Στιγμιοτύπων

# Πρόβλημα Κατάτμησης Στιγμιότυπων

## Mask R-CNN

- Faster R-CNN με δίκτυο RPN, για την πρόταση των υποψηφίων περιοχών,
- Τμήμα για τον υπολογισμό των масκών, αντίστοιχο με ένα Πλήρως Συνελικτικό Δίκτυο (FCN).



[Region-Based CNN](#)

[Object Detection and Tracking in 2020 | by Borijan Georgievski](#)

[TensorFlow Hub Object Detection Colab](#)

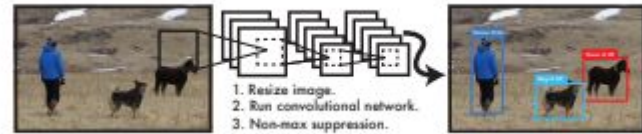
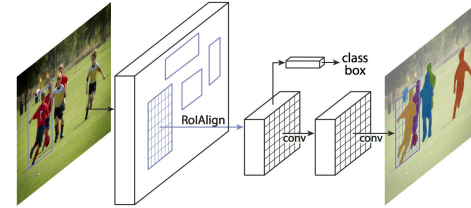
[Mask R-CNN](#)

[YOLO original paper](#)

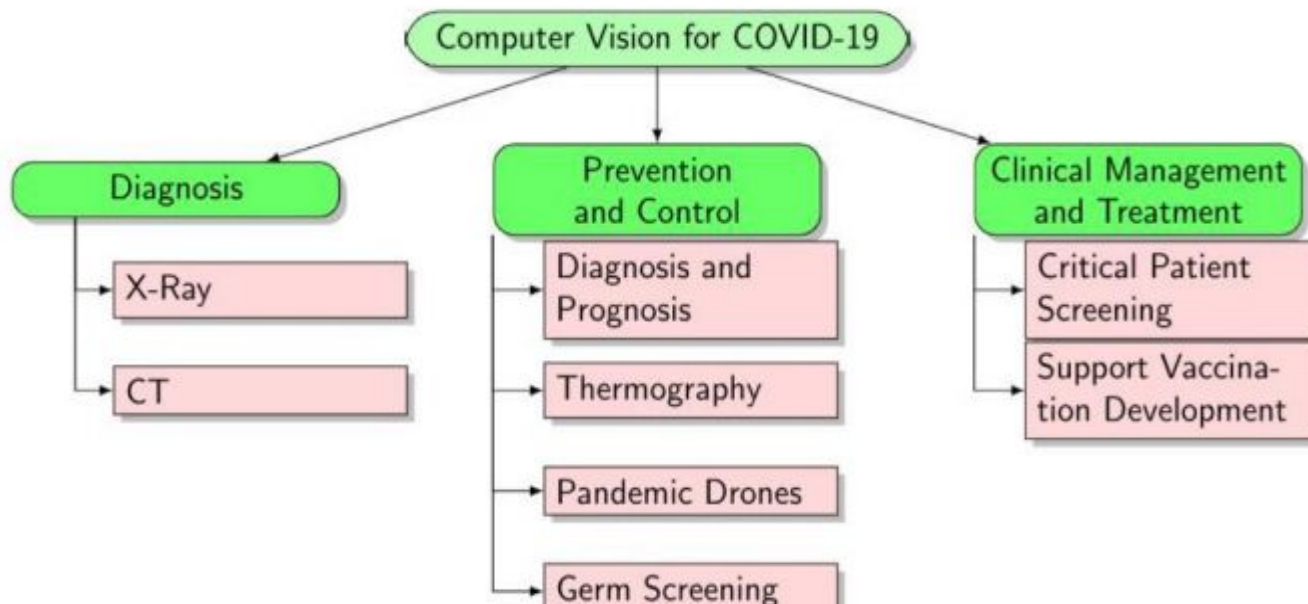
[YOLOv2 YOLO9000](#)

[DarkNet implementation](#)

[Introduction to YOLO Algorithm and YOLO Object Detection](#)



# Computer Vision, Deep Learning και COVID19



# Βασική διαγνωστική μέθοδος COVID-19: RT-qPCR tests

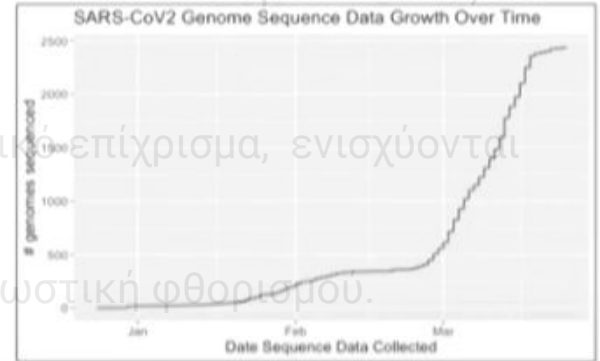
Τα Reverse Transcriptase quantitative Polymerase Chain Reaction tests θεωρούνται ως ο χρυσός κανόνας για τη διάγνωση του COVID-19

1. μικρές ποσότητες ιικού RNA που εξάγονται από ένα ρινικό επίχρισμα, ενισχύονται και ποσοτικοποιούνται,
2. η ανίχνευση ιών πραγματοποιείται χρησιμοποιώντας χημική φωσφορική φωτιστική φθορισμού.

(-) χρονοβόρα και χειροκίνητη, γεγονός που περιορίζει τη διαθεσιμότητά του σε μεγάλες κλίμακες,

(-) ορισμένες μελέτες έχουν δείξει επίσης ψευδώς θετική δοκιμή PCR

<https://arxiv.org/pdf/2004.09420.pdf>



As COVID-19 spreads around the world, there is a rapid expansion of available sequence data.



# Άλλες διαγνωστικές μέθοδοι COVID-19

- ανάλυση κλινικών συμπτωμάτων (βρογχοπνευμονία, πυρετός, βήχας, δύσπνοια, οξεία αναπνευστική δυσχέρεια -ARDS),
- επιδημιολογικό ιστορικό,
- **θετικές ακτινογραφικές εικόνες (CT / Chest radiograph (CXR)),**
- θετική παθογόνος παρακολούθηση.

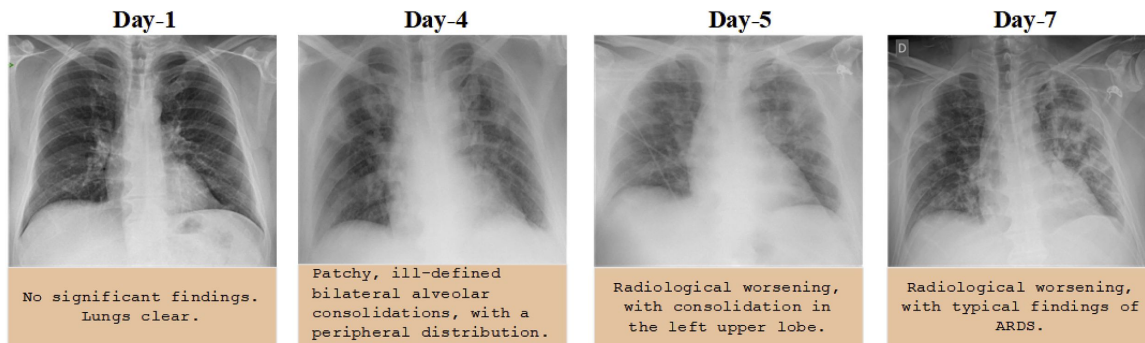
## Ακτινογραφικές εικόνες (CT / Chest radiograph (CXR)),

- Οι περισσότερες περιπτώσεις COVID-19 έχουν κοινά χαρακτηριστικά στις ακτινογραφικές εικόνες (early-stage bilateral, multi-focal, ground-glass opacities (GGO) with a peripheral or posterior distribution, primarily in the lower lobes, late-stage pulmonary consolidation)
- Οι εικόνες διαφόρων ιογενών τύπων πνευμονίας είναι παρόμοιες και αλληλεπικαλύπτονται με άλλες μολυσματικές και φλεγμονώδεις ασθένειες των πνευμόνων.
- Οι ακτινολόγοι δυσκολεύονται να διακρίνουν το COVID-19 από άλλες ιογενείς πνευμονίες

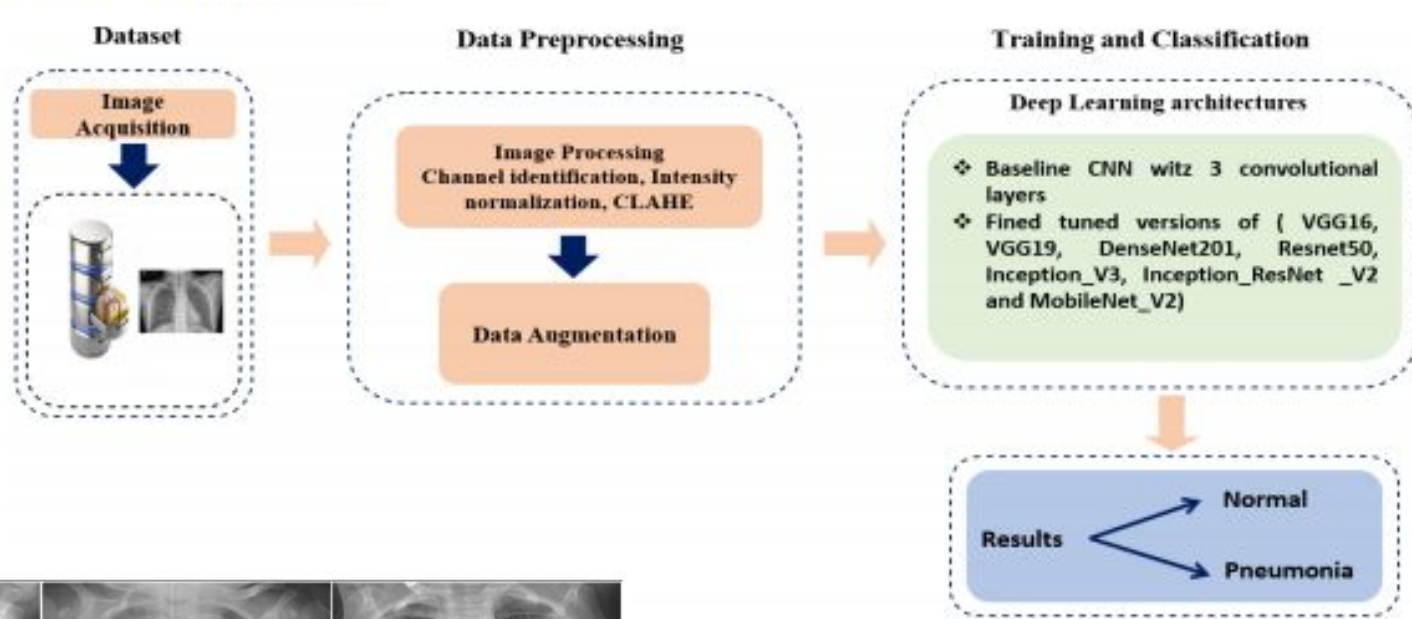
# COVID19 Diagnosis → X-ray και CT

## Χρήση Deep learning

1. Χρήση open source dataset από chest X-ray (CXR) και CT (Computed Tomography) εικόνες ασθενών, θετικών και μη, σε COVID- 19
2. Εκπαίδευση δικτύου βαθιάς μάθησης
3. Αξιολόγηση αποτελεσμάτων



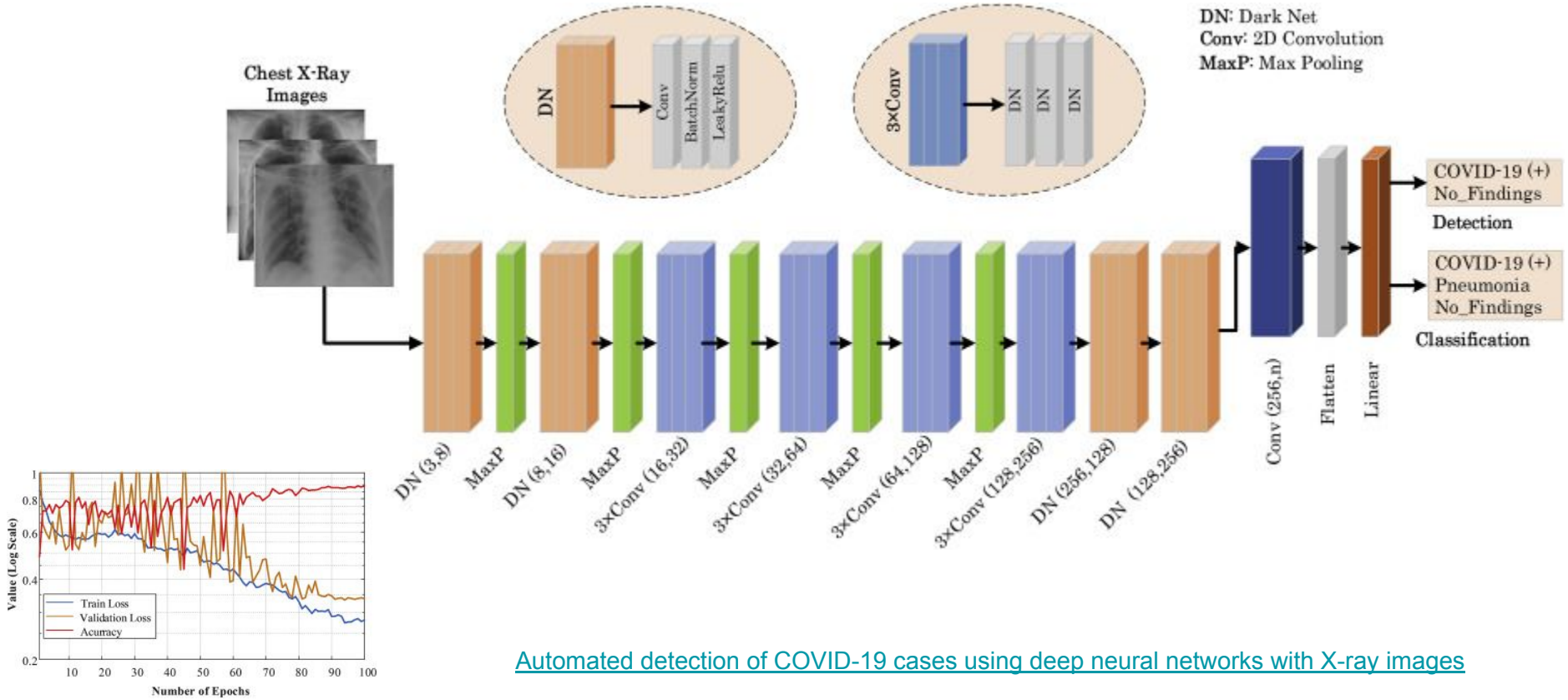
# Automated Methods for Detection and Classification Pneumonia based on X-Ray Images Using Deep Learning



(a). Normal (b). Bacterial Pneumonia (c). Viral Pneumonia

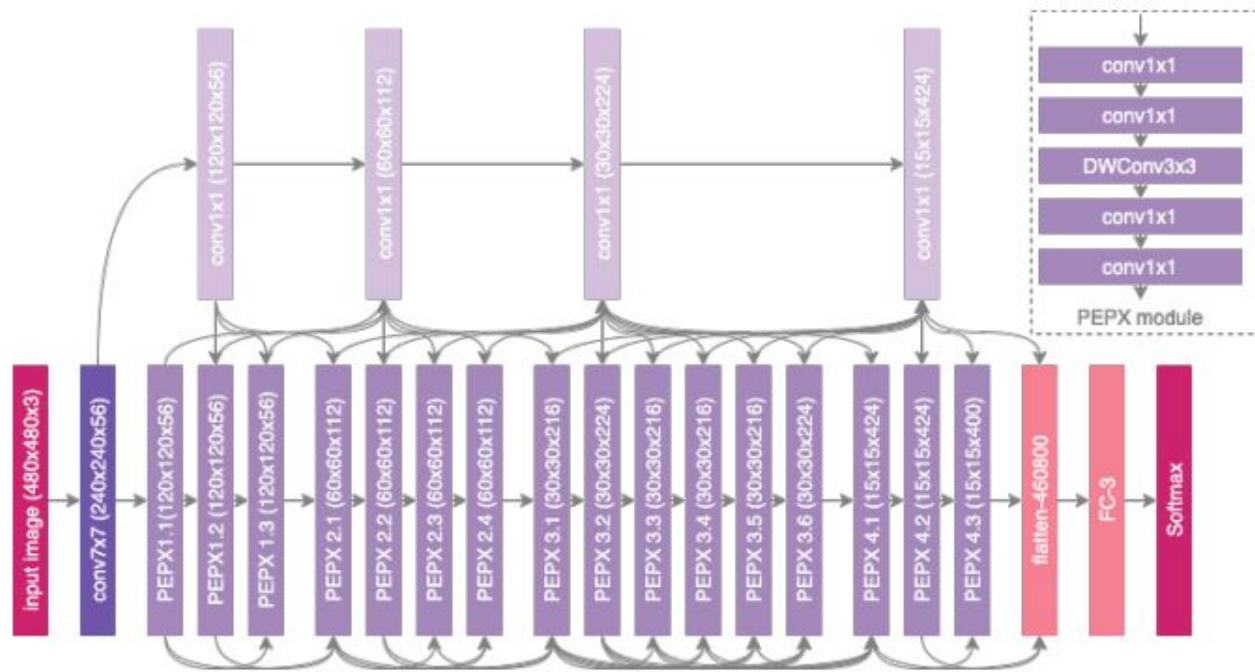
Fig. 4: Examples of Chest X-Rays in patients with pneumonia

# Automated detection of COVID-19 cases using deep neural networks with X-ray images



[Automated detection of COVID-19 cases using deep neural networks with X-ray images](#)

# COVID-Net: A Tailored Deep CNN Design for Detection of COVID-19 Cases from Chest X-Ray Images



**Figure 5. COVID-Net Architecture.** High architectural diversity and selective long-range connectivity can be observed as it is tailored for COVID-19 case detection from CXR images. The heavy use of a projection-expansion-projection design pattern in the COVID-Net architecture can also be observed, which provides enhanced representational capacity while maintaining computational efficiency.

Architecture	Params (M)	MACs (G)	Acc. (%)
VGG-19	20.37	89.63	83.0
ResNet-50	24.97	17.75	90.6
COVID-Net	<b>11.75</b>	<b>7.50</b>	<b>93.3</b>

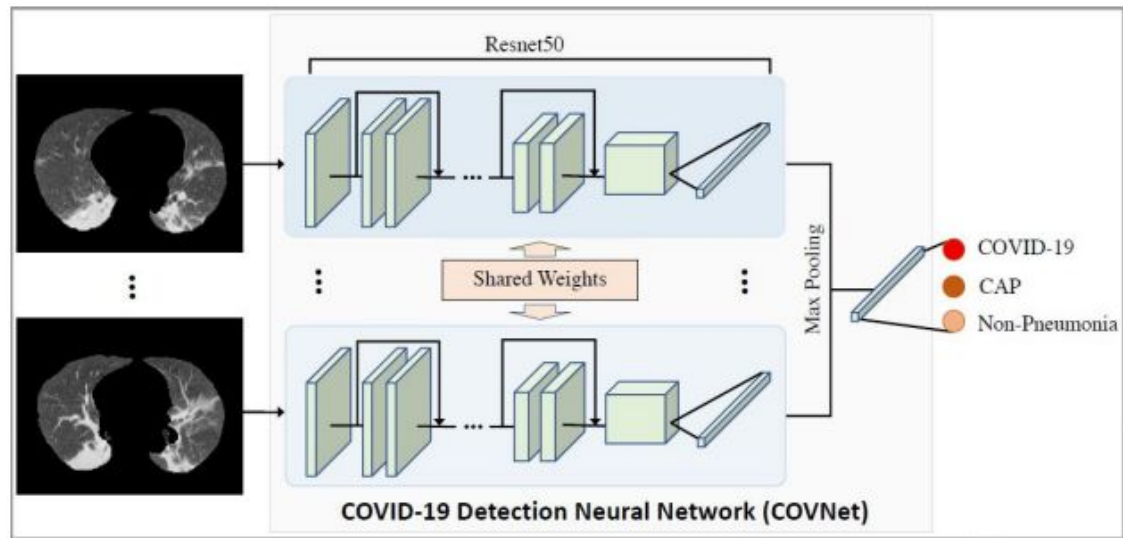
# COVID 19 Diagnosis → CT (Computed Tomography)

- Ανάλυση ακτινολογικών χαρακτηριστικών με βαθιά μάθηση για ακριβή διάγνωση COVID-19
  - ◆ ταχύτερα από τις τρέχουσες εξετάσεις αίματος,
  - ◆ εξοικονόμηση κρίσιμου χρόνου για τον έλεγχο των ασθενειών
- Το COVID-19 παρουσιάζει παρόμοια χαρακτηριστικά απεικόνισης με άλλους τύπους πνευμονίας, καθιστώντας δύσκολη τη διαφοροποίηση,
- Τα Royal College in Australia (Patrick Brennan, Professor, Diagnostic Imaging, University of Sydney), Royal College of Radiology in London, American College of Radiology υποστηρίζουν ότι οι εικόνες CT δεν μπορούν να χρησιμοποιηθούν ως πρωτεύον διαγνωστικό εργαλείο,
- Προτεινόμενη ρουτίνα: **CT-plus-RT-PCR testing για COVID-19.**

# A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19),

[Bo Xu et al, Tianjin Medical University Cancer Institute and Hospital](#), April 24, 2020

Dataset από εικόνες CT πνευμόνων: 740 τυπ  
325 επιβεβαιωμένα κρούσματα COVID-19  
Model: CNN, Accuracy model: 89.5% , Accur  
Πρόσβ

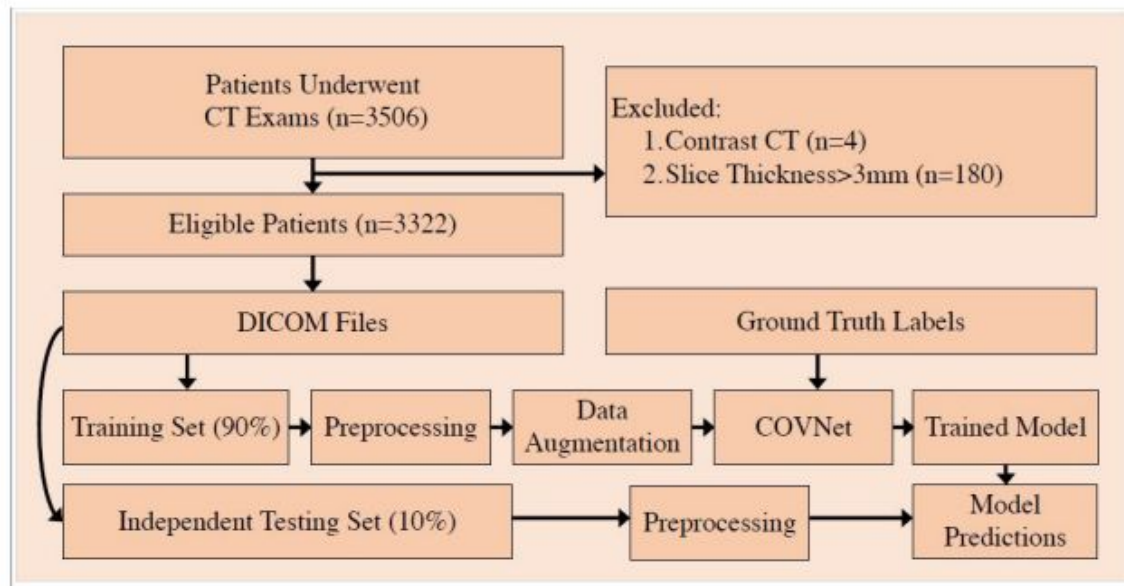


**Fig 2. COVID-19 detection neural network (COVNet) architecture.** The COVNet is a convolutional neural network (CNN) using ResNet50 as the backbone. It takes as input a series of CT slices and generates a classification prediction of the CT image. The CNN features from each slice of the CT series are combined by a max-pooling operation and the resulting feature map is fed to a fully connected layer to generate a probability score for each class.

# Artificial Intelligence Distinguishes COVID-19 from Community Acquired Pneumonia on Chest CT

[Jun Xia et al.](#)

[Wuhan Huangpi People's Hospital](#)

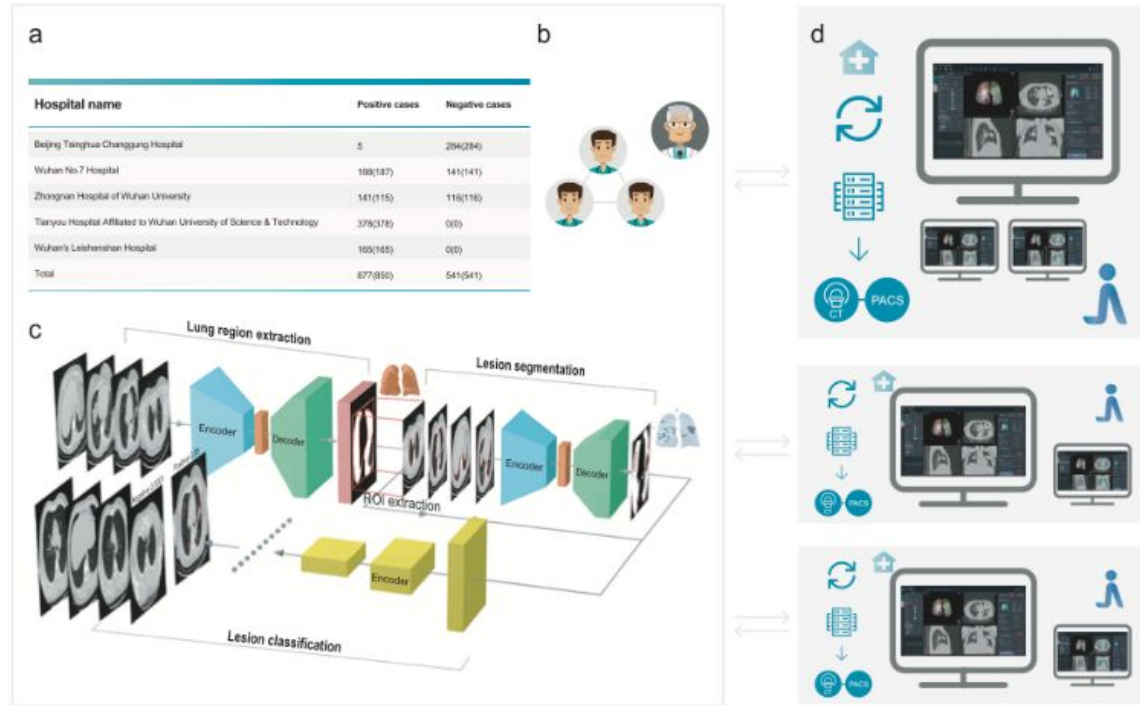


**Fig 1. Flow diagram.** We collected a dataset of 3506 patients with chest CT exams. After exclusion, 3,322 eligible patients were included for the model development and evaluation in this study. CT exams were extracted from DICOM files. The dataset was split into a training set (to training the model), and the independent testing set at the patient level. A supervised deep learning framework (COVNet) was developed to detect COVID-19 and community acquired pneumonia. The predictive performance of the model was evaluated by using an independent testing set. COVNet = COVID-19 detection neural network.



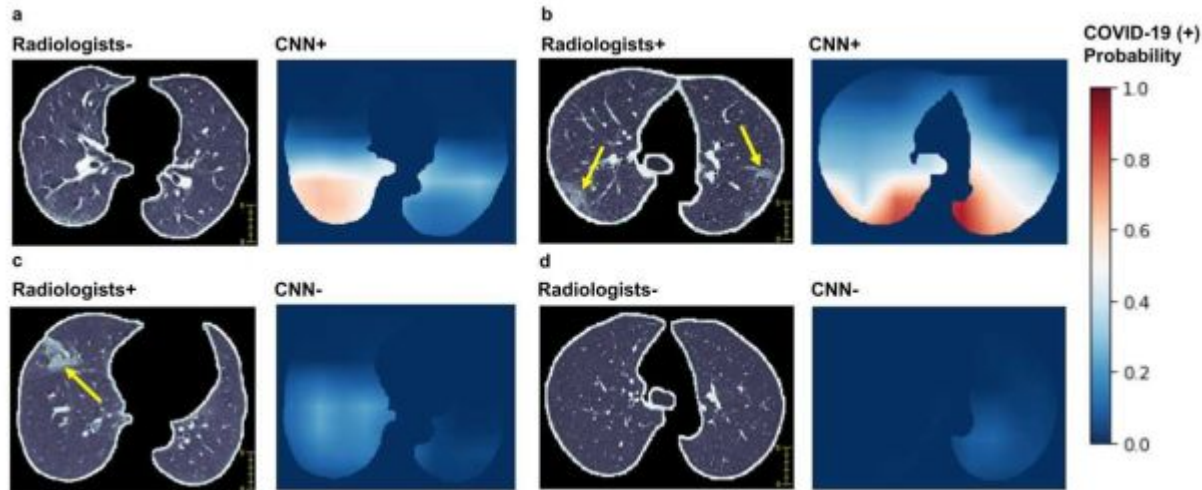
# Coronavirus: business and technology in a pandemic

[AI runs smack up against a big data problem in COVID-19 diagnosis](#)



*Wei Xu of Tsinghua University and colleagues at other Chinese institutions built an AI system that not only analyzes CT scans but also integrates with a radiologist's workflow. Trying to bring the system to Europe has been slowed by various factors including lack of access to data, says Xu.*

# Artificial intelligence-enabled rapid diagnosis of patients with COVID-19



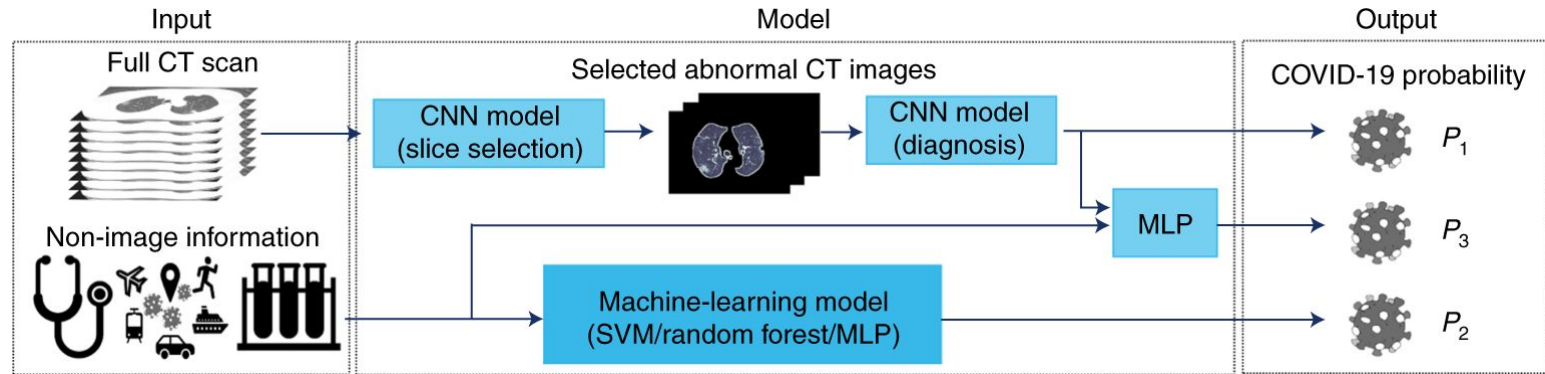
[Mei, X., Lee, H., Diao, K. et al. Artificial intelligence-enabled rapid diagnosis of patients with COVID-19. \*Nat Med\* \(2020\)](#)

Για κάθε ζεύγος εικόνων:

- η αριστερή εικόνα είναι μια εικόνα CT που χρησιμοποιείται ως είσοδος για το CNN
- η δεξιά εικόνα δείχνει τον χάρτη θερμότητας των pixel που ταξινόμησε το CNN ως SARS-CoV-2 (το κόκκινο δείχνει υψηλότερη πιθανότητα).

# Artificial intelligence-enabled rapid diagnosis of patients with COVID-19

Mei, X., Lee, H., Diao, K. et al. Artificial intelligence-enabled rapid diagnosis of patients with COVID-19. *Nat Med* (2020). <https://doi.org/10.1038/s41591-020-0931-3>



Τρία μοντέλα AI χρησιμοποιούνται για τη δημιουργία της πιθανότητας ενός ασθενούς να είναι COVID-19 (+):

1. Βασίζεται σε αξονική τομογραφία θώρακος (TL μοντέλο πνευμονικής φυματίωσης (PTB) , 99.4%) Οι δέκα κορυφαίες μη φυσιολογικές εικόνες CT ανά ασθενή τοποθετήθηκαν στο δεύτερο CNN (CNN διάγνωση) για να προβλέψουν την πιθανότητα θετικότητας COVID-19 ( $P_1$ ).
2. Βασίζεται σε δημογραφικά και κλινικά δεδομένα (ηλικία και φύλο του ασθενούς, ιστορικό έκθεσης, συμπτώματα και εργαστηριακές εξετάσεις) τέθηκαν σε ένα μοντέλο μηχανικής μάθησης για την ταξινόμηση της θετικότητας COVID-19 ( $P_2$ ).
3. Βασίζεται σε συνδυασμό της αξονικής τομογραφίας θώρακος και κλινικών πληροφοριών ενσωματώθηκε σε ένα δίκτυο MLP για τη δημιουργία της τελικής παραγωγής του κοινού μοντέλου ( $P_3$ ).

**Table I.** Summary of deep learning methods for COVID-19 diagnosis using radiology images

Papers	Data	AI Methods	Results
[2]	4,356 chest CT exams from 3,322 patients from 6 medical centers: 1,296 exams for COVID-19, 1,735 for CAP and 1,325 for non-pneumonia	A 3D convolutional ResNet-50 [3], namely COVNet	AUC for detecting COVID-19 is of 0.96
[4]	618 CT samples: 219 from 110 COVID-19 patients, 224 CT samples from 224 patients with influenza-A viral pneumonia, and 175 CT samples from healthy people	Location-attention network and ResNet-18 [3]	Accuracy of 86.7%
[7]	5,941 Posterior-anterior chest radiography images across 4 classes (normal: 1,583, bacterial pneumonia: 2,786, non-COVID-19 viral pneumonia: 1,504, and COVID-19: 68)	Drop-weights based Bayesian CNNs	Accuracy of 89.92%
[8]	1,065 CT images (325 COVID-19 and 740 viral pneumonia)	Modified inception transfer-learning model	Accuracy of 79.3% with specificity of 0.83 and sensitivity of 0.67
[9]	Clinical data and a series of chest CT data collected at different times on 133 patients of which 54 patients progressed to severe/critical periods whilst the rest did not	Multilayer perceptron and LSTM [40]	AUC of 0.954
[10]	970 CT volumes of 496 patients with confirmed COVID-19 and 1,385 negative cases	2D deep CNN	Accuracy of 94.98% and AUC of 97.91%
[11]	CT images of 1,136 training cases (723 positives for COVID-19) from 5 hospitals	A combination of 3D UNet++ [12] and ResNet-50 [3]	Sensitivity of 0.974 and specificity of 0.922
[13]	Chest X-ray images of 50 normal and 50 COVID-19 patients	Pre-trained ResNet-50	Accuracy of 98%
[14]	16,756 chest radiography images across 13,645 patient cases from two open access data repositories	A deep CNN, namely COVID-Net	Accuracy of 92.4%
[15]	CT images obtained from 157 international patients (China and U.S.)	ResNet-50	AUC of 0.996
[16]	1,341 normal, 1,345 viral pneumonia, and 190 COVID-19 chest X-ray images	AlexNet [18], ResNet-18 [3], DenseNet-201 [19], SqueezeNet [20]	Accuracy of 98.3%
[17]	170 X-ray images and 361 CT images of COVID-19 from 5 different sources	A new CNN and pre-trained AlexNet [18] with transfer learning	Accuracy of 98% on X-ray images and 94.1% on CT images

[Artificial Intelligence in the Battle against Coronavirus \(COVID-19\): A Survey and Future Research Directions](#)

# COVID 19 Diagnosis

[COVID-19 CT scans](#)

<https://arxiv.org/ftp/arxiv/papers/2004/2004.03431.pdf>

[lindawangg/COVID-Net: COVID-Net Open Source Initiative](#)

[Machine Learning Analysis of Chest CT Scan Images as a Complementary Digital Test of Coronavirus \(COVID-19\) Patients](#)

<https://arxiv.org/ftp/arxiv/papers/2003/2003.05037.pdf>

<https://arxiv.org/pdf/2003.13865.pdf>

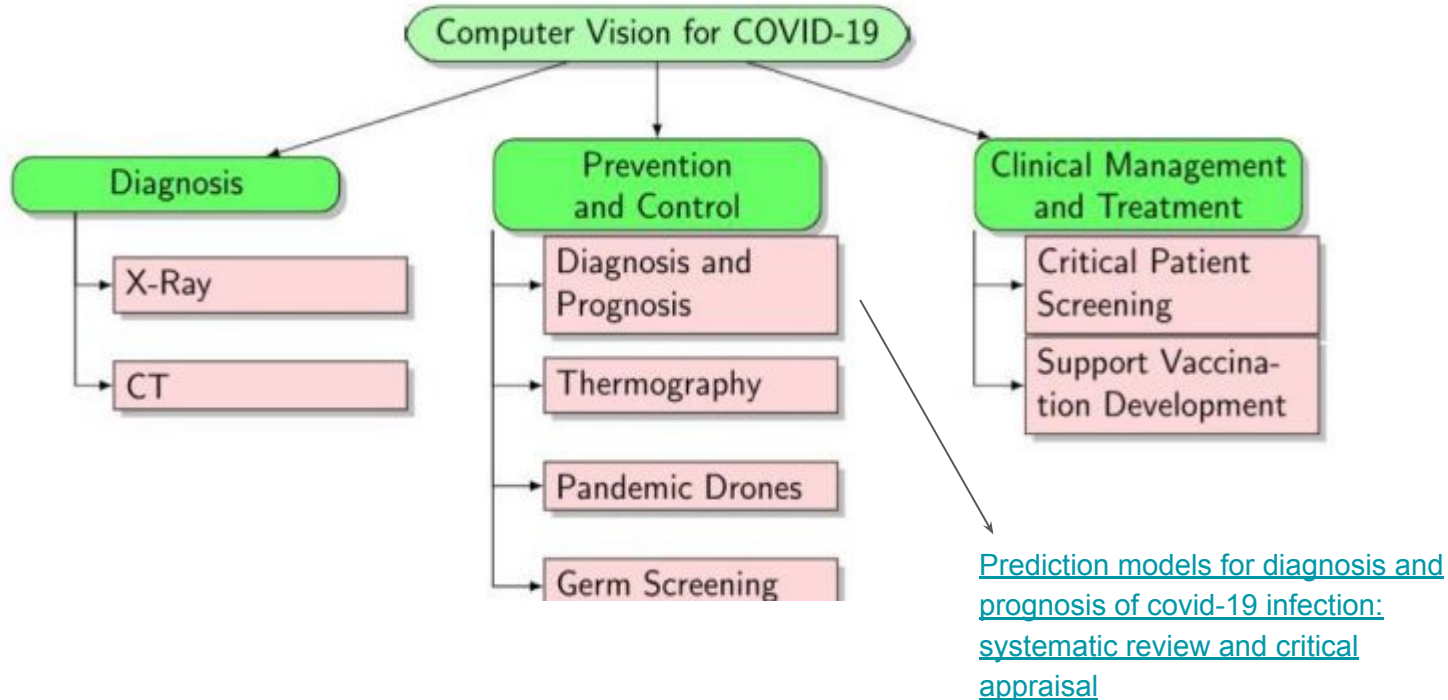
[Deep Learning for Medical Imaging: COVID-19 Detection » Deep Learning - MATLAB & Simulink](#)

[Detecting COVID-19 in X-ray images with Keras, TensorFlow, and Deep Learning](#)

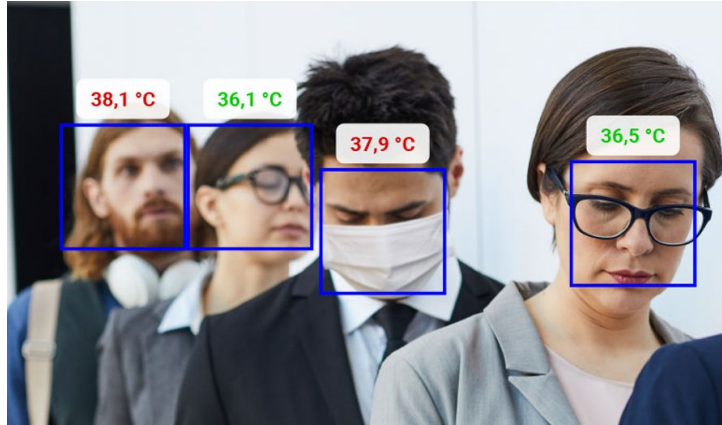
[COVID-CT-Dataset: A CT Image Dataset about COVID-19](#)

<https://arxiv.org/abs/2002.09334>

# CV for COVID19: Prevention & Control



# CV for COVID19: Prevention & Control → Thermograph



# AI4COVID-19: AI Enabled Preliminary Diagnosis for COVID-19 from Cough Samples via an App

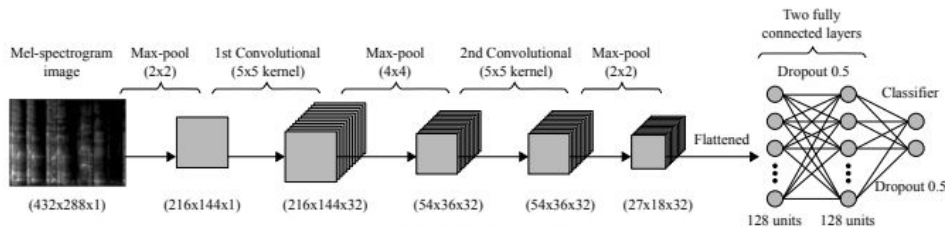


Figure 3: Cough detection classifier.

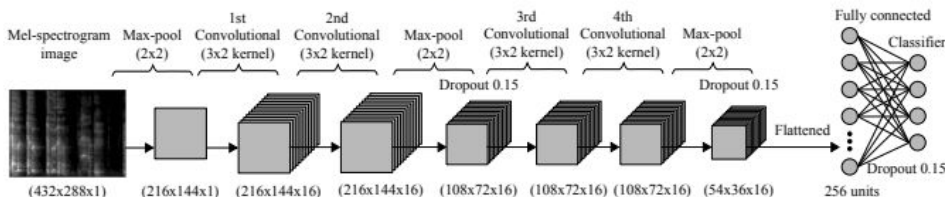


Figure 4: Deep Learning-based Multi-Class classifier (DL-MC).

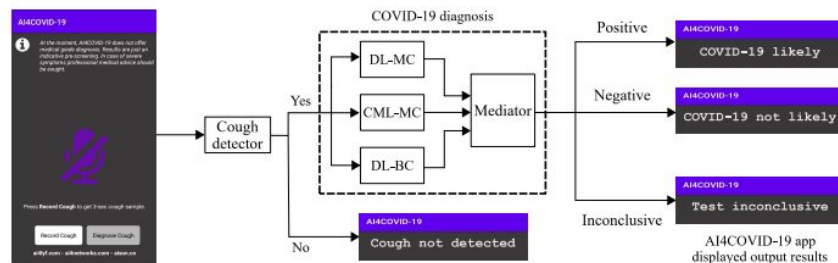


Figure 2: Proposed system architecture of AI4COVID-19, showing snapshot of Smartphone App at user front-end and back-end cloud AI-engine blocks consisting of Cough Detector block (further elaborated in Figure 3) and COVID-19 diagnosis block containing Deep Learning-based Multi-Class classifier (DL-MC), Classical Machine Learning-based Multi-Class classifier (CML-MC), and Deep Learning-based Binary-Class classifier (DL-BC), further elaborated in Figure 4 and Figure 5.

Διακρίνει βήχα ασθενούς COVID19 και διάφορων τύπων βήχα ασθενών χωρίς COVID-19 με ακρίβεια άνω του 90%.



# Real-time tracking of self-reported symptoms to predict potential COVID-19

12:16

COVID-19 status

Have you had a test for COVID-19?

No

Next Question

ΣΙ

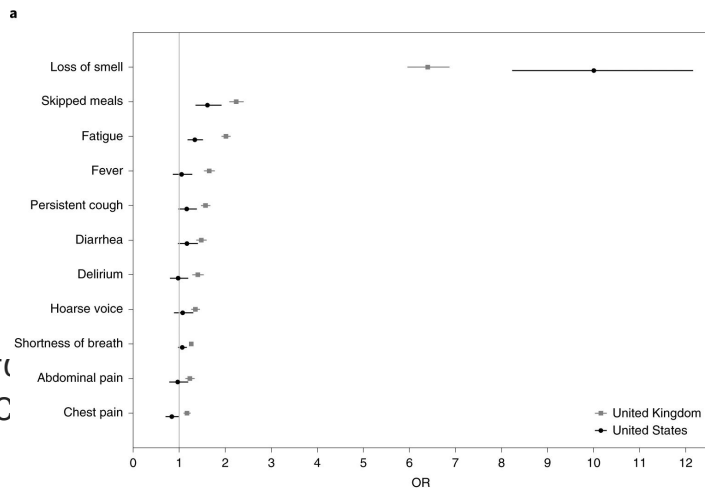
Π'Α

12:16

How do you feel right now?

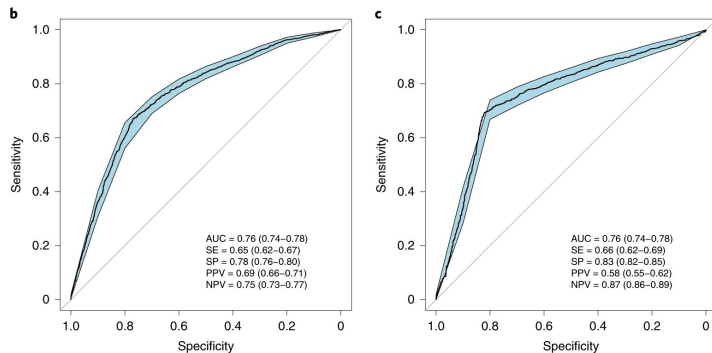
I feel as healthy as normal

I'm not feeling quite right



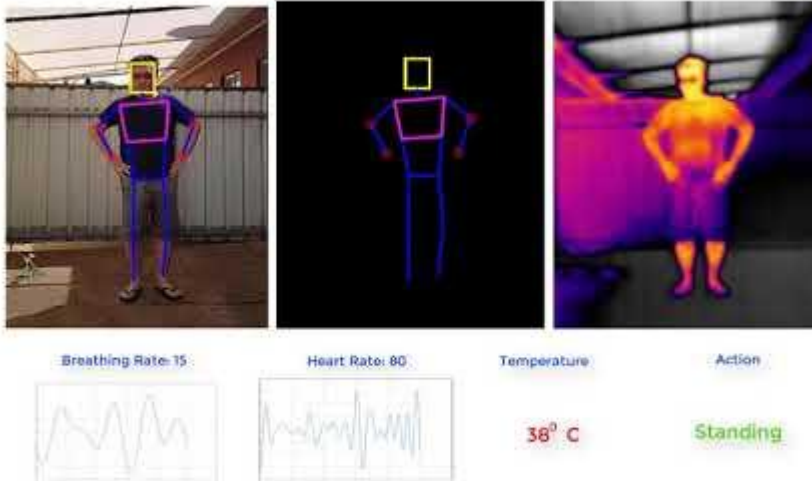
$$\text{Prediction model} = -1.32 - (0.01 \times \text{age}) + (0.44 \times \text{sex}) + (1.75 \times \text{loss of smell and taste}) + (0.31 \times \text{severe or significant persistent cough}) + (0.49 \times \text{severe fatigue}) + (0.39 \times \text{skipped meals})$$

σε 15.638 Βρετανούς και 2.763 Αμερικάνους



[Real-time tracking of self-reported symptoms to predict potential COVID-19](#)

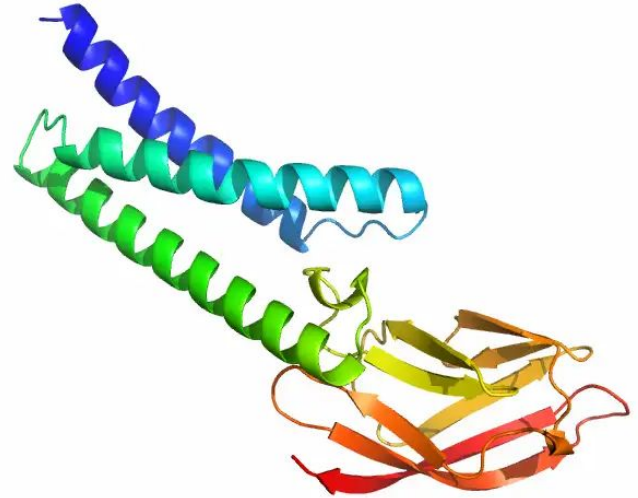
# COVID19: Prevention & Control → Pandemic Drones



η,  
οής από απόσταση,  
ως βήχας ή φτέρνισμα σε ένα πλαίσιο

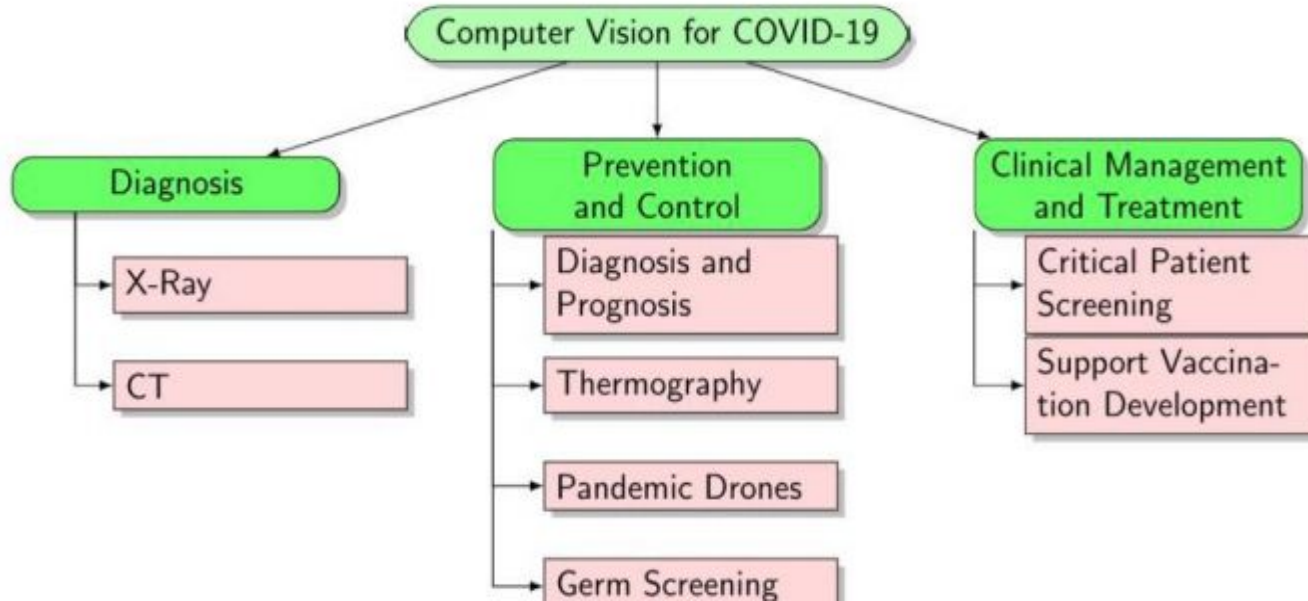
# COVID19: Prevention & Control → Germ Screening

- Το Google DeepMind παρουσίασε το AlphaFold, ένα πρωτ τρισδιάστατη δομή μιας πρωτεΐνης με βάση τη γενετική
- Στις αρχές Μαρτίου, το σύστημα δοκιμάστηκε στο Covid-
- Η DeepMind κυκλοφόρησε προβλέψεις δομής πρωτεϊνών σχετίζονται με το SARS-CoV-2, τον ιό που προκαλεί το C κοινότητα να κατανοήσει καλύτερα τον ιό.



[Mila and its partners rally the scientific community to develop novel data-driven solutions to assist with COVID-19 outbreak](#)

# CV for COVID19: Clinical Management and Treatment



# CV for COVID19: Clinical Management and Treatment → Critical Patient Screening

[AI-Assisted Elderly Care for Acute Infection and Chronic Disease, Fei Fei Li et al](#)

## How to take care of seniors while keeping them safe?

-  Early Symptom Detection of COVID-19
-  Monitor Patients with Mild Symptoms
-  Manage Chronic Conditions

## Endowing in-home care with AI-Powered Smart Sensor Technology



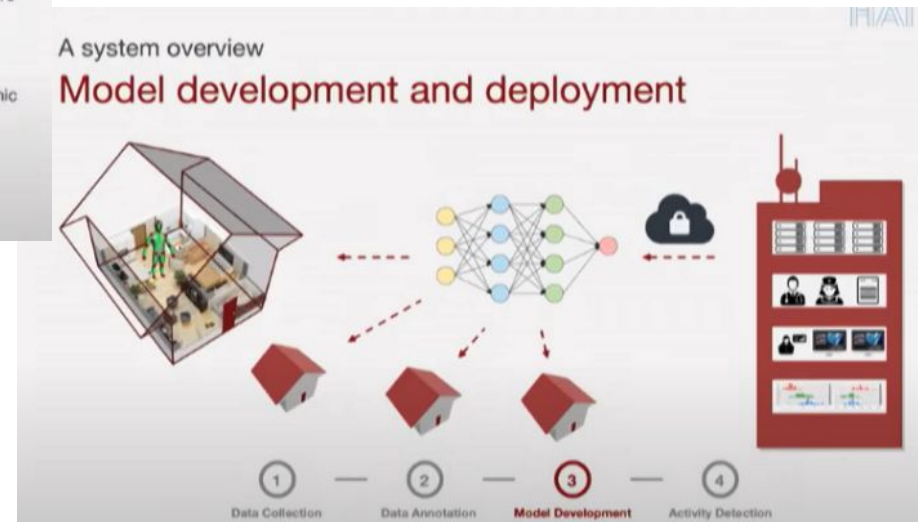
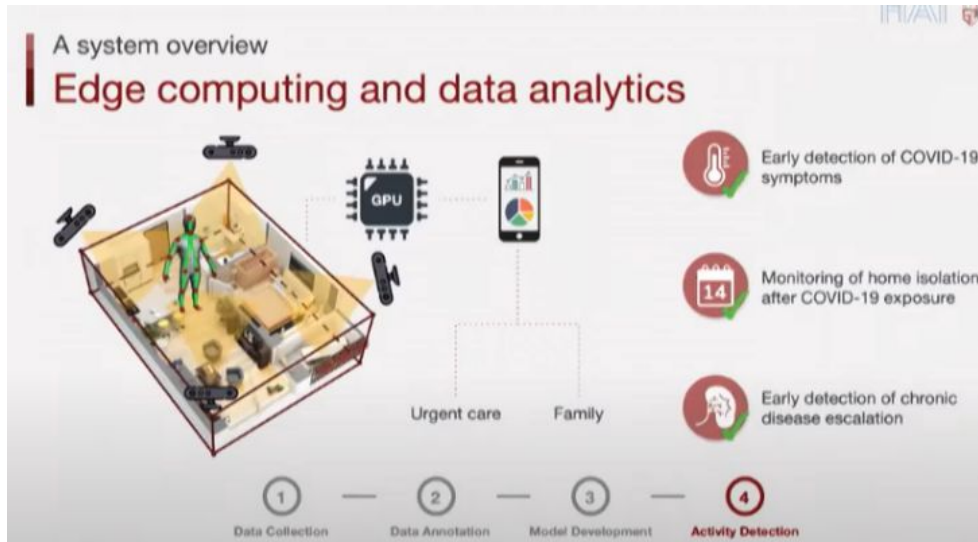
**Versatile**

- Mobility
- Infection
- Sleep
- Diet

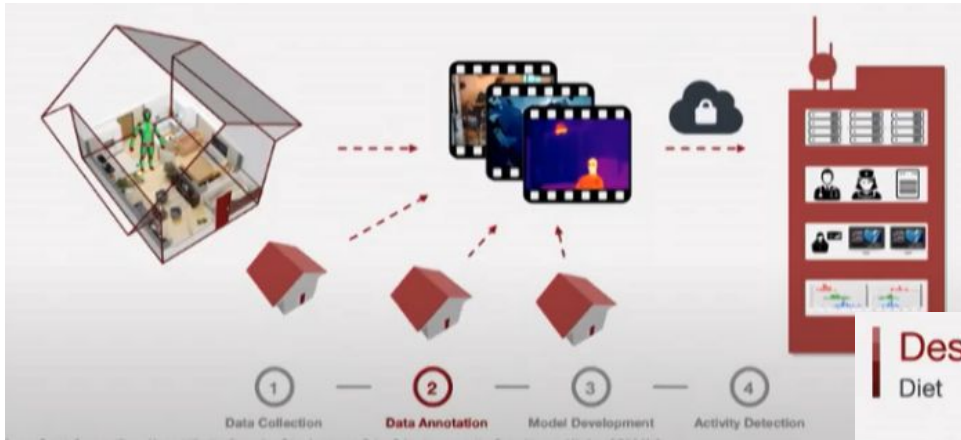
**Scalable**

- Low-cost
- Burden-free

# AI-Assisted Elderly Care for Acute Infection and Chronic Disease



# AI-Assisted Elderly Care for Acute Infection and Chronic Disease



## Descriptive analytics of clinical status

### Diet

Eating  
Fluid intake  
Alcohol consumption  
Pill consumption

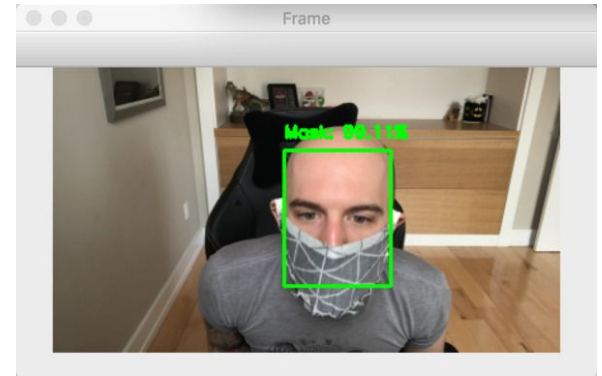
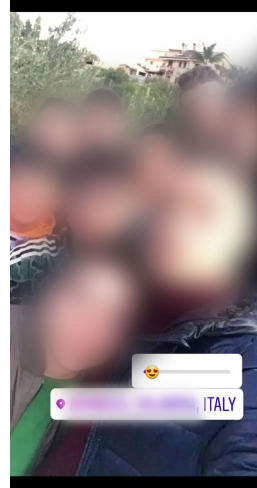
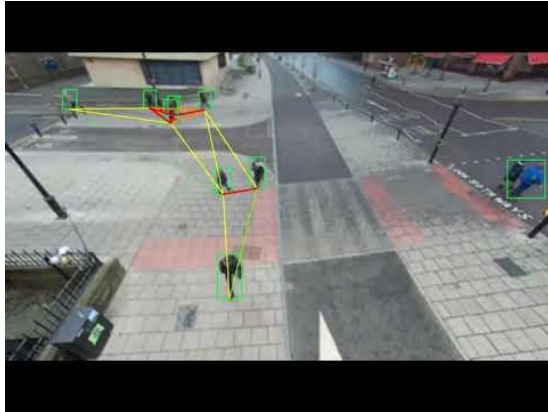
Activity: eating  
Subjects: caregiver, senior  
Objects: silverware, toast  
Scene graph:

Spatial-temporal scene graph

# Surveillance

[Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques](#)

[Using Public Drinking Water AI to Identify Lockdown Violations](#)





# Επιπλέον υλικό

[Stanford researchers help develop privacy-focused coronavirus alert app | Stanford News](#)

[How Computer Vision Came In Handy For Social Distancing](#)

[It's a Record-Breaking Crowd! A Must-Read Tutorial to Build your First Crowd Counting Model using Deep Learning](#)

[Μπορεί το Apple Watch να ανιχνεύσει τον COVID-19: Το πανεπιστήμιο του Stanford θα προσπαθήσει να μάθει](#)

[Within the Lack of Chest COVID-19 X-ray Dataset: A Novel Detection Model Based on GAN and Deep Transfer Learning](#)

[AI exposed Brits ignoring advice to stay home and socially distance](#)

[Computer vision system detects fever remotely to rule out COVID-19](#)

[YouTube as a source of medical information on the novel coronavirus 2019 disease \(COVID-19\) pandemic](#)

[Imaging COVID-19 AI – An initiative for automated diagnosis and quantitative analysis of COVID-19 on imaging](#)

[How can AI help with the COVID-19 vaccine search?](#)

[AI exposed Brits ignoring advice to stay home and socially distance](#)

# Dataset

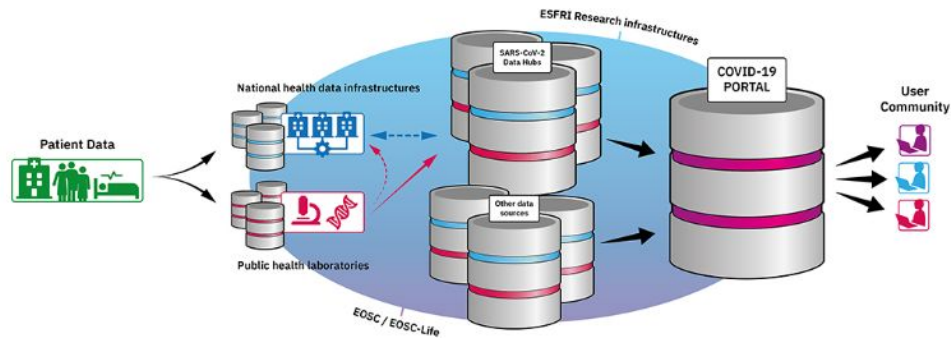
<https://ec.europa.eu/jrc/en>

<https://www.entrepreneur.com/article/348368>

<https://www.imveurope.com/tags/coronavirus>

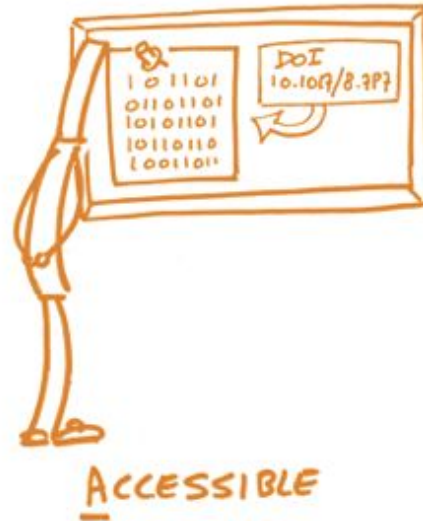
<https://c3.ai/products/c3-ai-covid-19-data-lake/>

<https://www.ert.gr/eidiseis/epistimi/covid-19-i-eyropaiki-epitropi-egkainiazei-platforma-antallagis-dedomenon-gia-ereynites/>

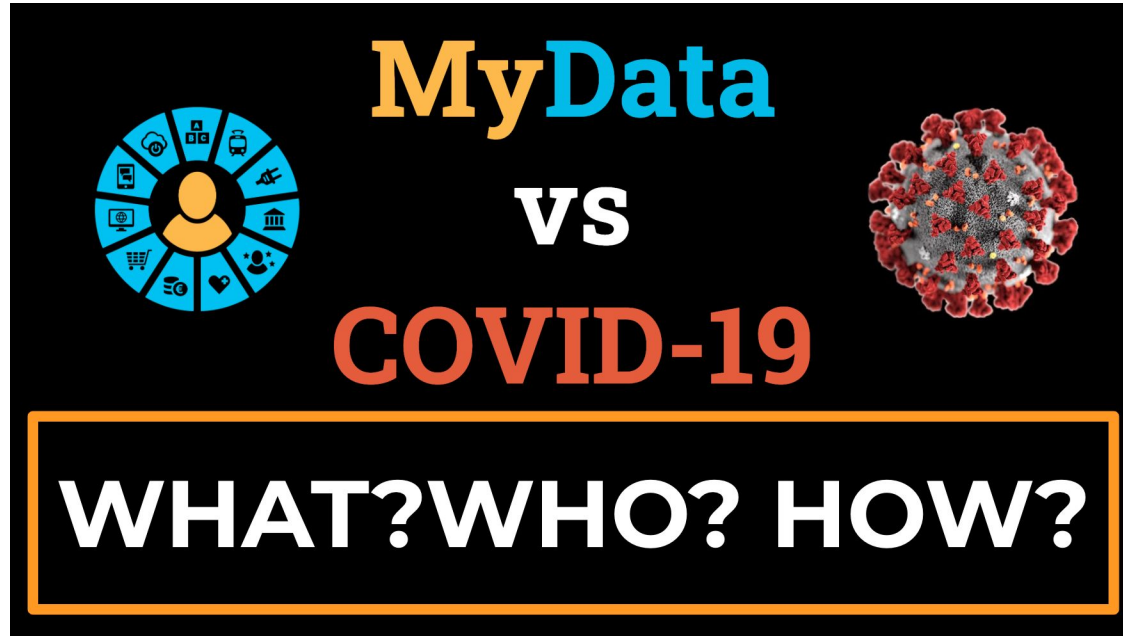


# FAIR Data

## FAIR DATA PRINCIPLES



# Ethical issues



**MyData**  
**VS**  
**COVID-19**

**WHAT?WHO? HOW?**