

CPH 100A: Advanced Imaging: *Localization and Segmentation*

Instructor: Adam Yala, PhD (yala@berkeley.edu)

Agenda

Recap

Motivation for Localization

Localization as Attention

Bounding box prediction

Segmentation

What's wrong with FNNs?



$$x^0 = [0, 1, 1, \dots, 0]$$



$$x^1 = [0, 0, 1, \dots, 0]$$

Small padding, very different feature vectors!

Can it learn?

Yes

Will it be easy?

No.

→ Learn invariance is hard.

→ Memory prohibitive

Conceptual role of Hypothesis class: Choose your Mountain range

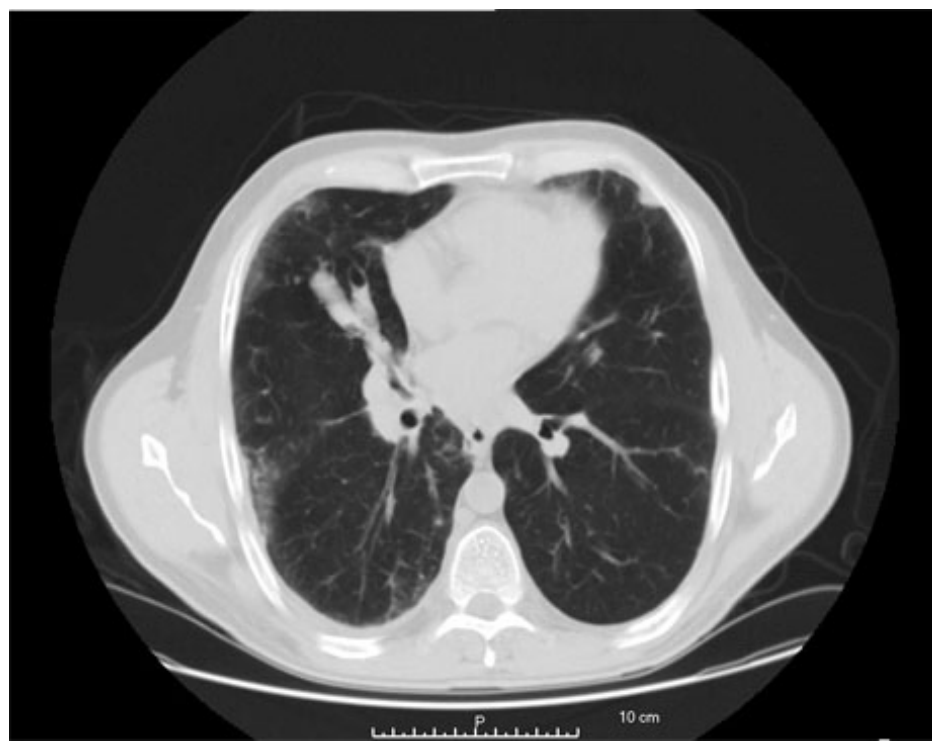


Desired properties for a good Hypothesis Class

Capture spatial dependencies: Pixel positions and locality matter!

Handle Translations / Nuissance variations: Objects of interest can appear anywhere

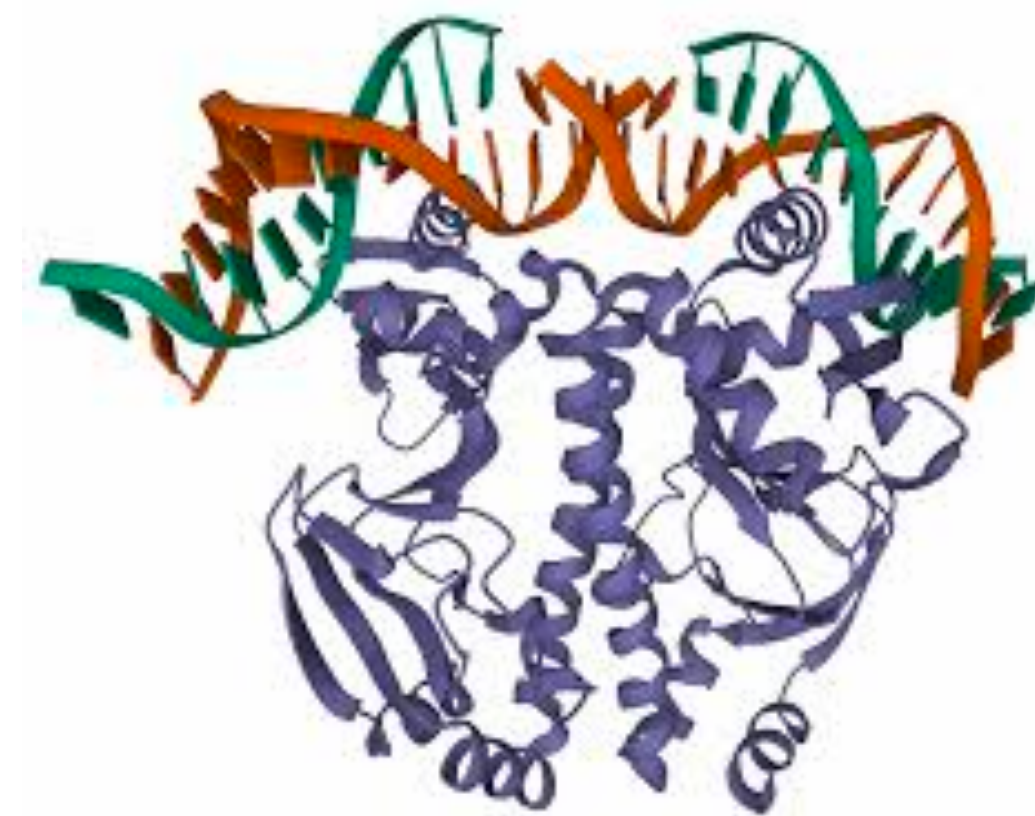
Scale: Allow efficient computation for large inputs



Images/ Volumes

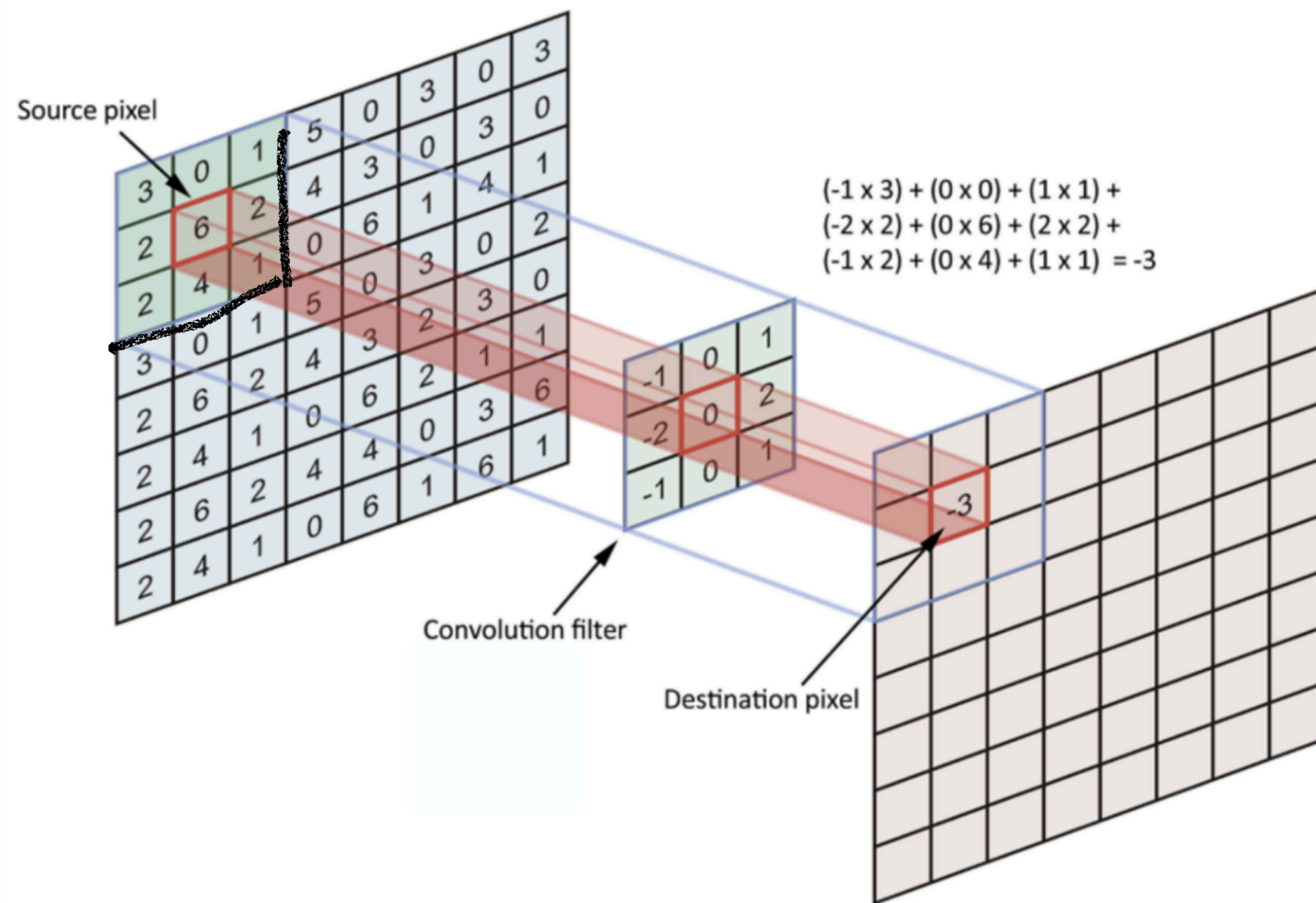


Text



Graphs

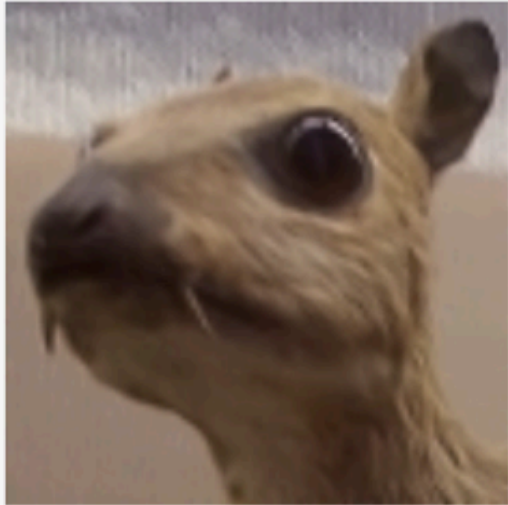

2D Convolutions



The convolution operation.

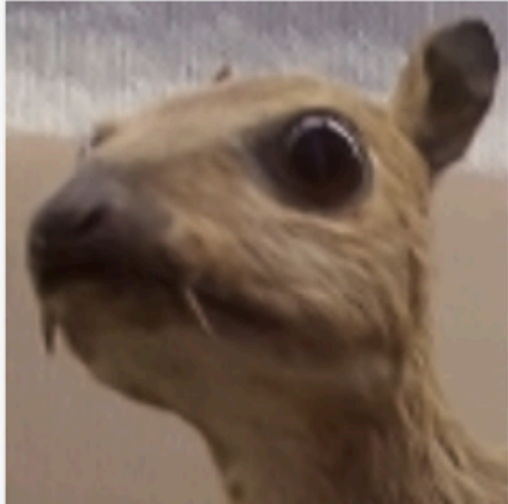
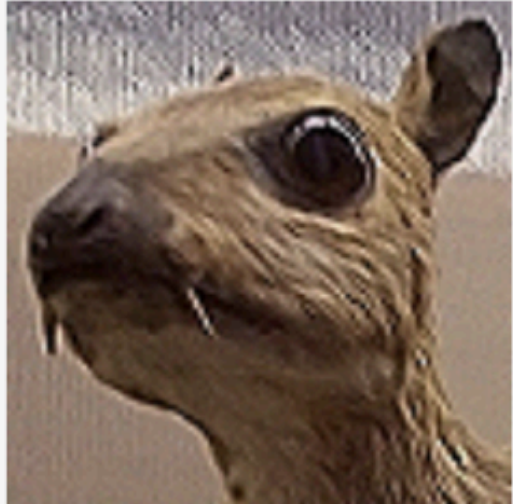
Examples of Convolutions

Edge detection

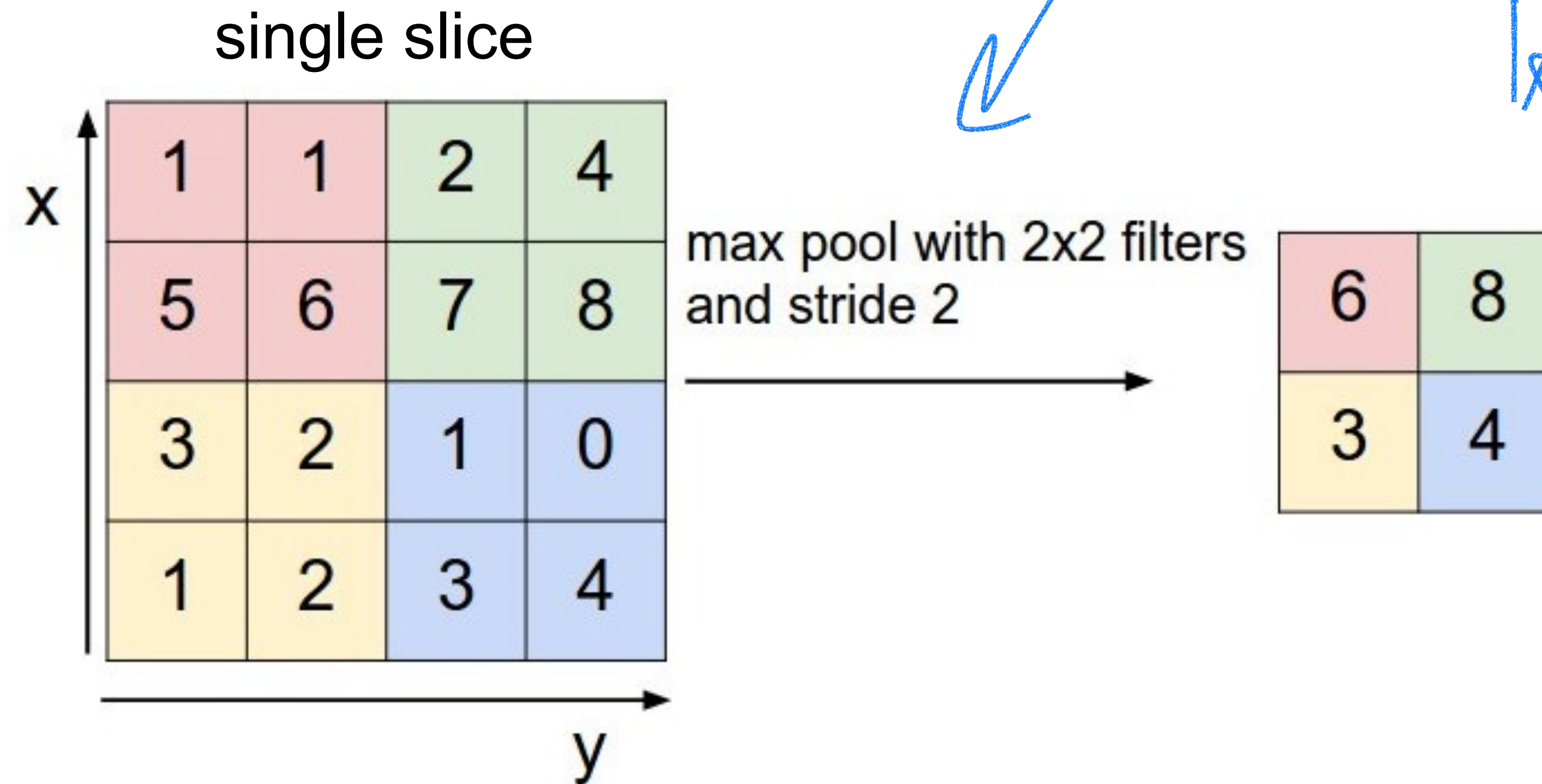
 * $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ = 

Kernel

Sharpen

 * $\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ = 

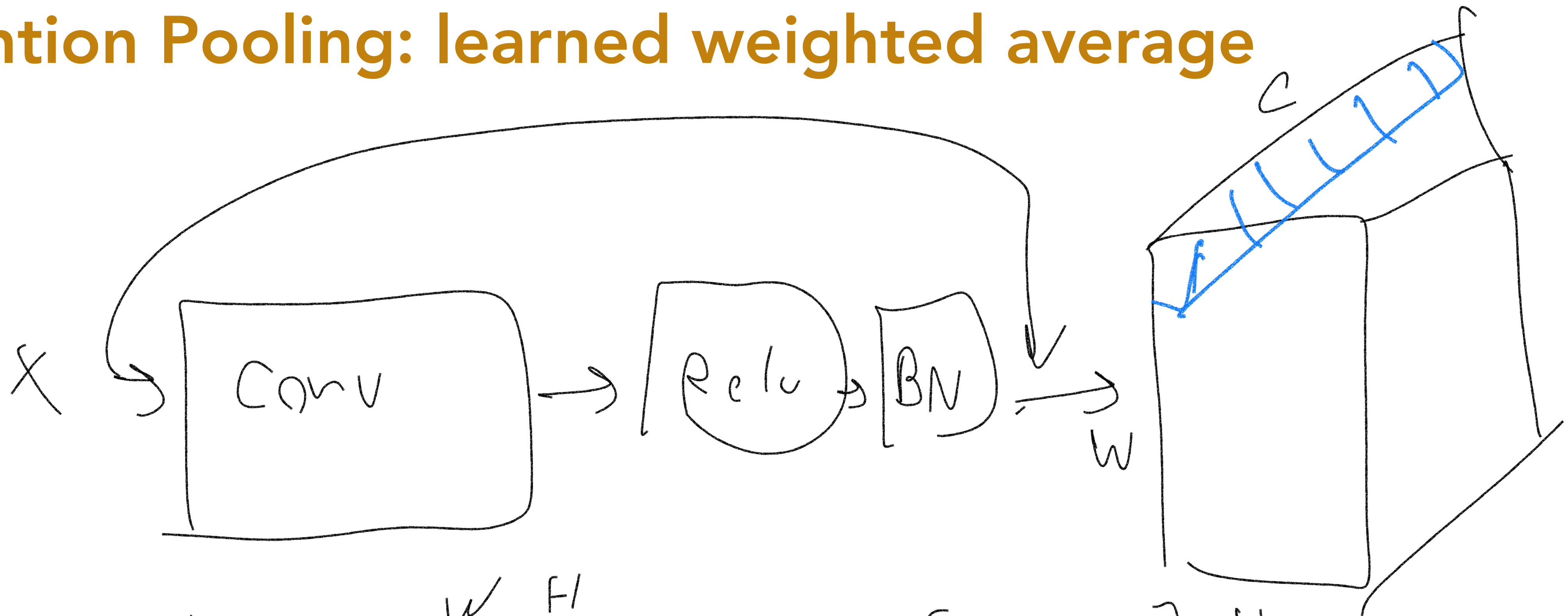
Max Pooling



Pooling as conv
kernel!

- Similar to filtering, but output the maximum entry instead of a weighted sum

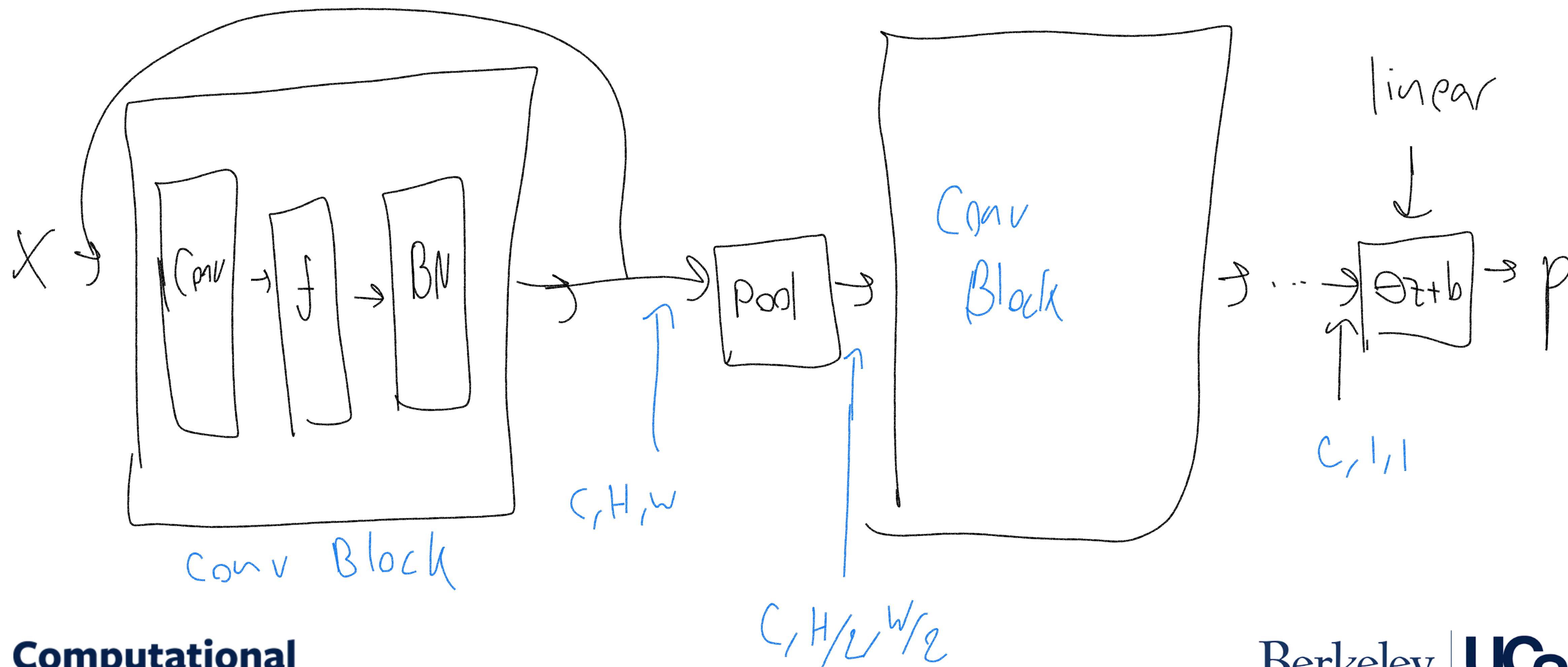
Attention Pooling: learned weighted average



$$\text{Attn Pool}(z) = \sum_i^W \sum_j^H \alpha_{ij} z[i, j]$$

$$\alpha = \text{Softmax}(f(z)) \quad |f(z)| \rightarrow (W, H, 1)$$

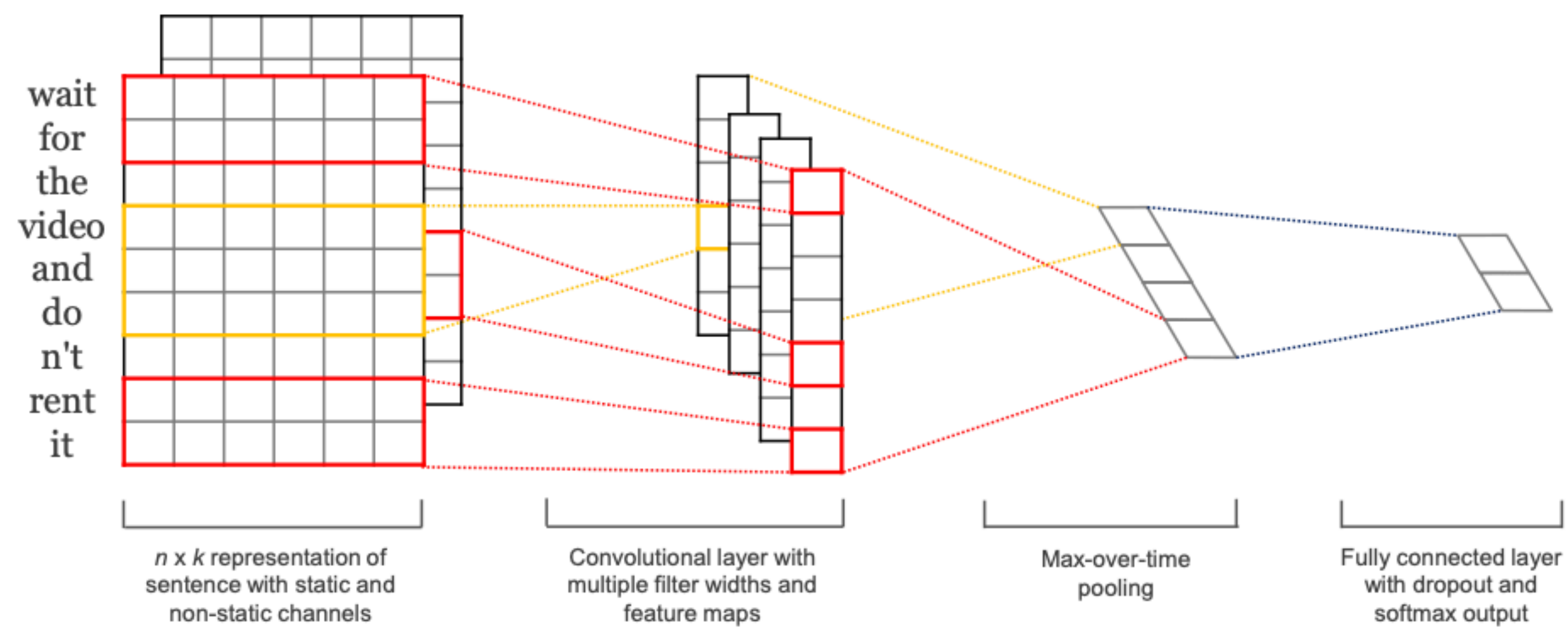
Putting it together: CNNs



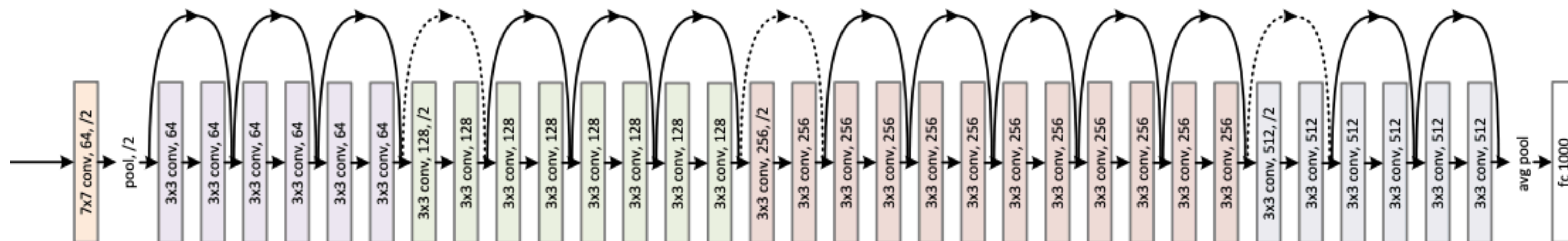
Popular 1D CNNs: Text

Convolutional Neural Networks for Sentence Classification

Yoon Kim
New York University
yhk255@nyu.edu

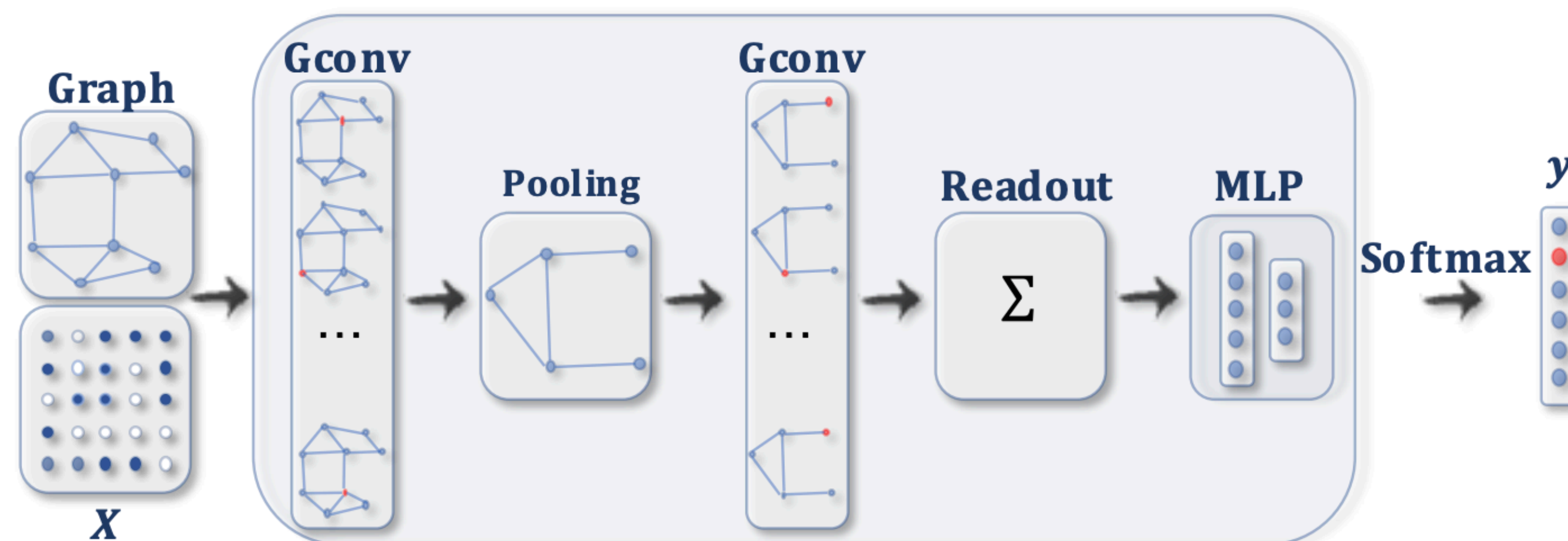


Popular 2D CNNs: ResNets



He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

Popular GNNs: Convolutions on graphs



(b) A ConvGNN with pooling and readout layers for graph classification [21]. A graph convolutional layer is followed by a pooling layer to coarsen a graph into sub-graphs so that node representations on coarsened graphs represent higher graph-level representations. A readout layer summarizes the final graph representation by taking the sum/mean of hidden representations of sub-graphs.

Wu, Zonghan, et al. "A comprehensive survey on graph neural networks." *IEEE transactions on neural networks and learning systems* 32.1 (2020): 4-24.

Agenda

Recap

Motivation for Localization

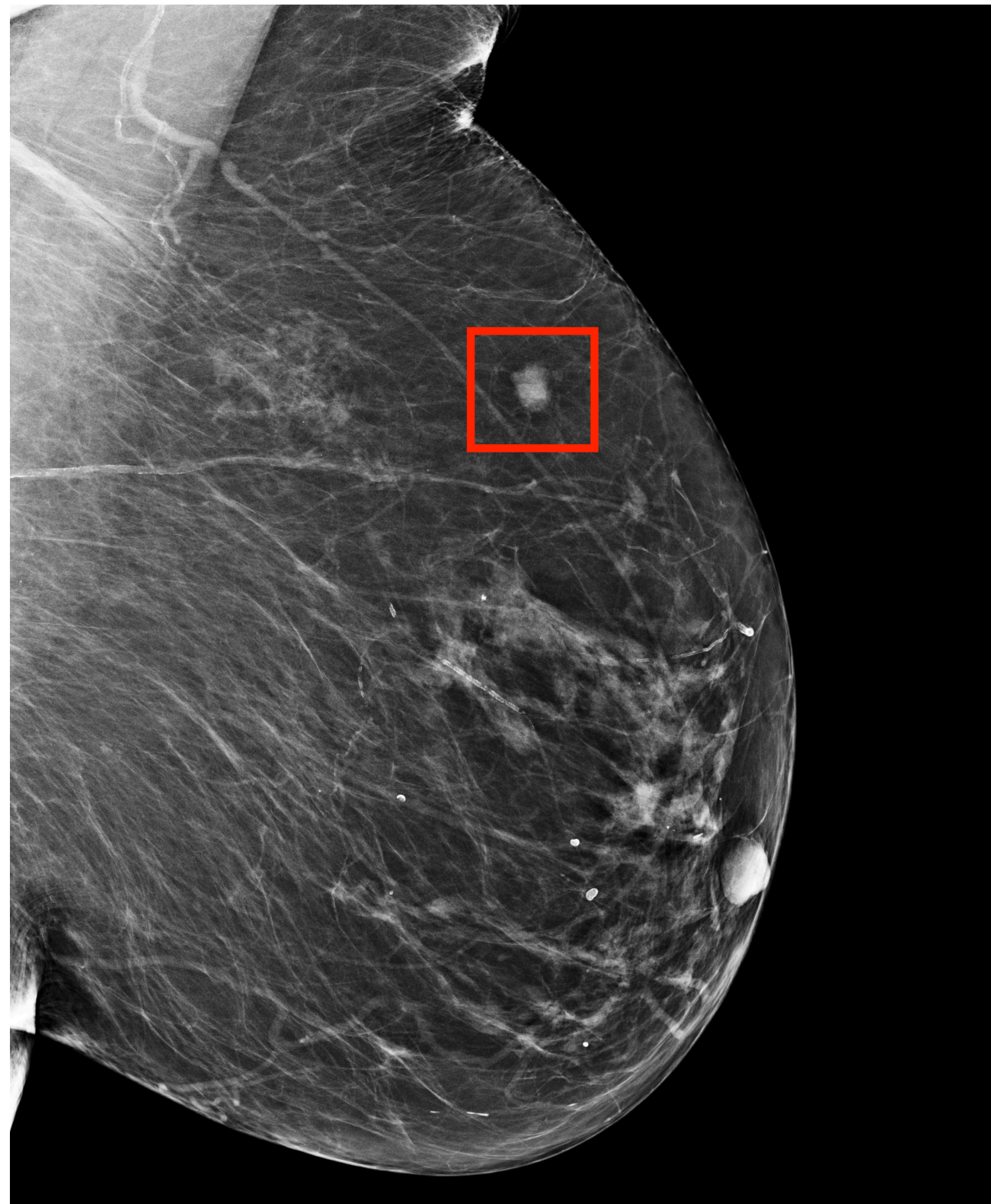
Localization as Attention

Bounding box prediction

Segmentation

Motivating Example:

Cancer detection on Mammography



Screening Statistics

1,000 Screens

100 Suspicious

20 Biopsies

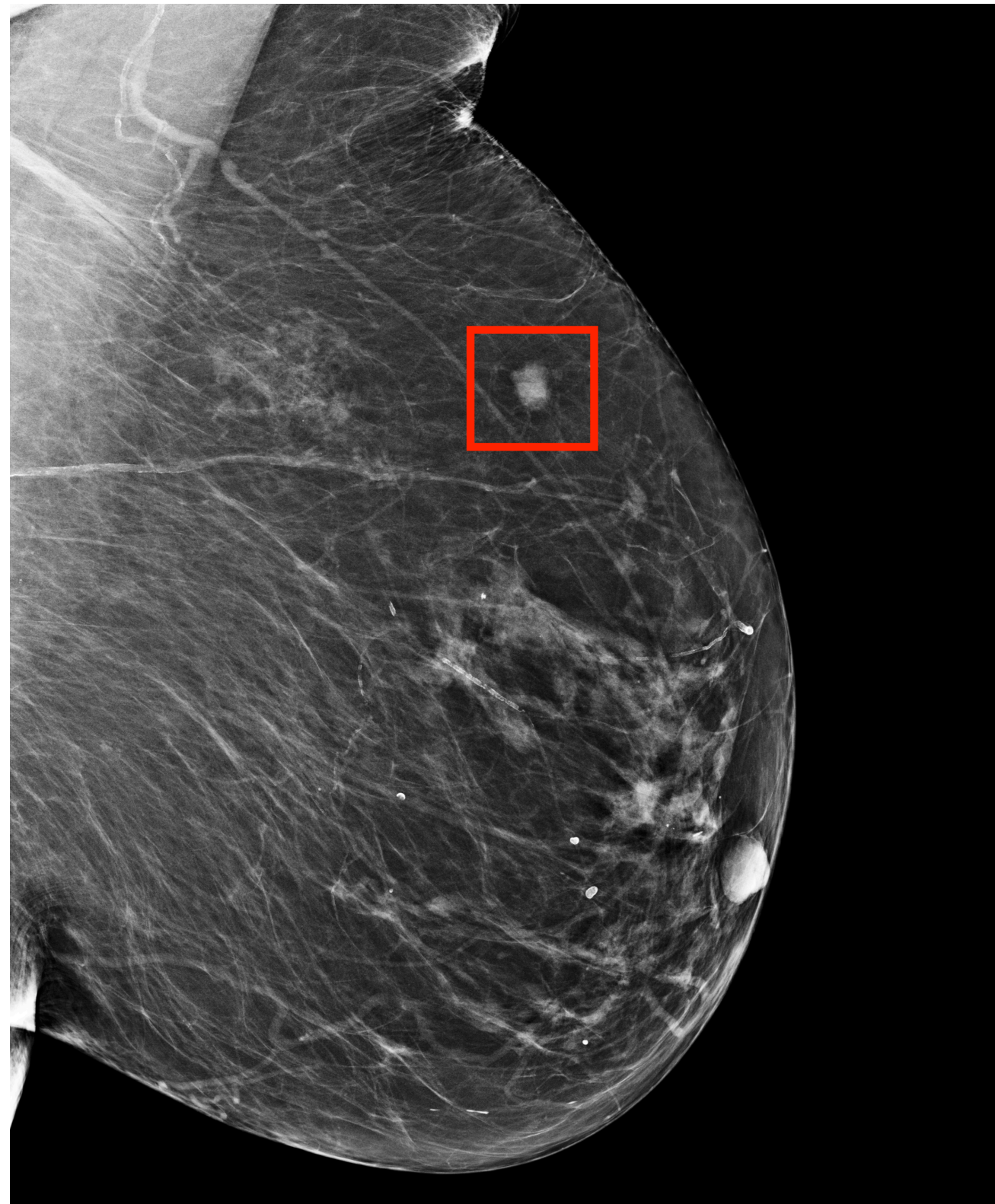
5 Cancers found, 1 missed

What are useful CPH interventions?

Do they need localization?

Motivating Example:

Localization as a useful modeling endpoint



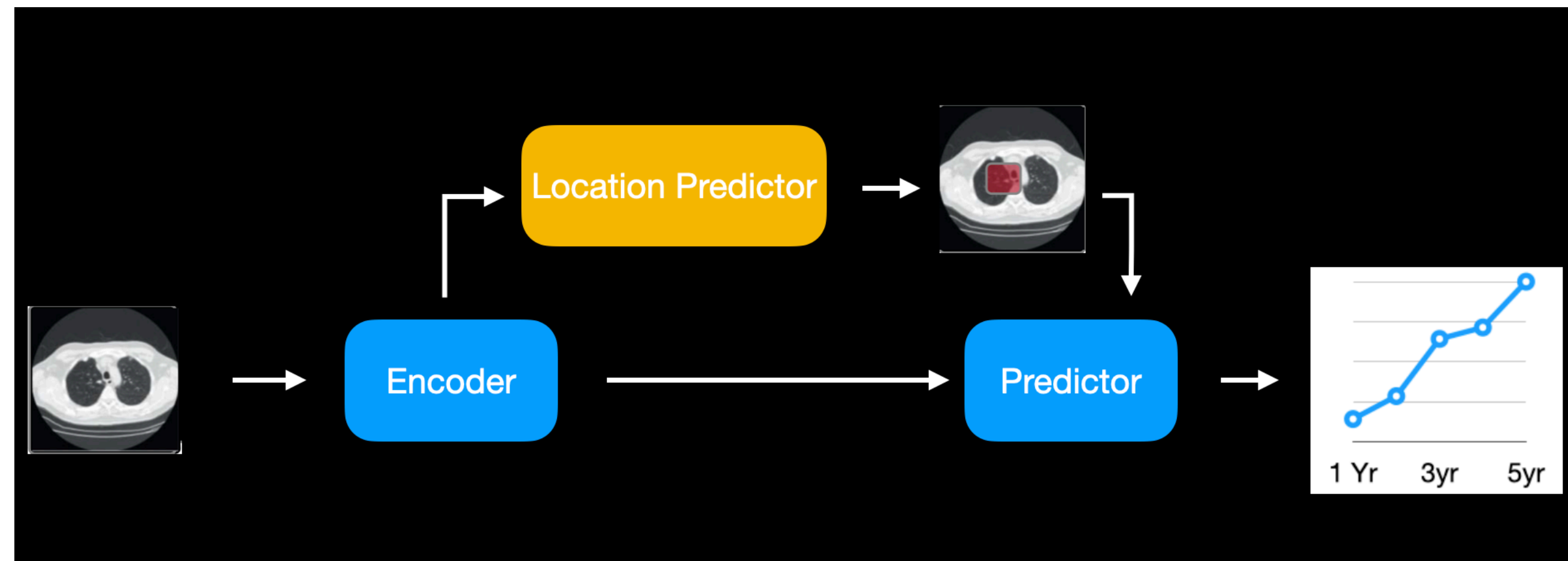
Can localization help improve radiologist **sensitivity**?

- Interpretability
- Guiding biopsy location

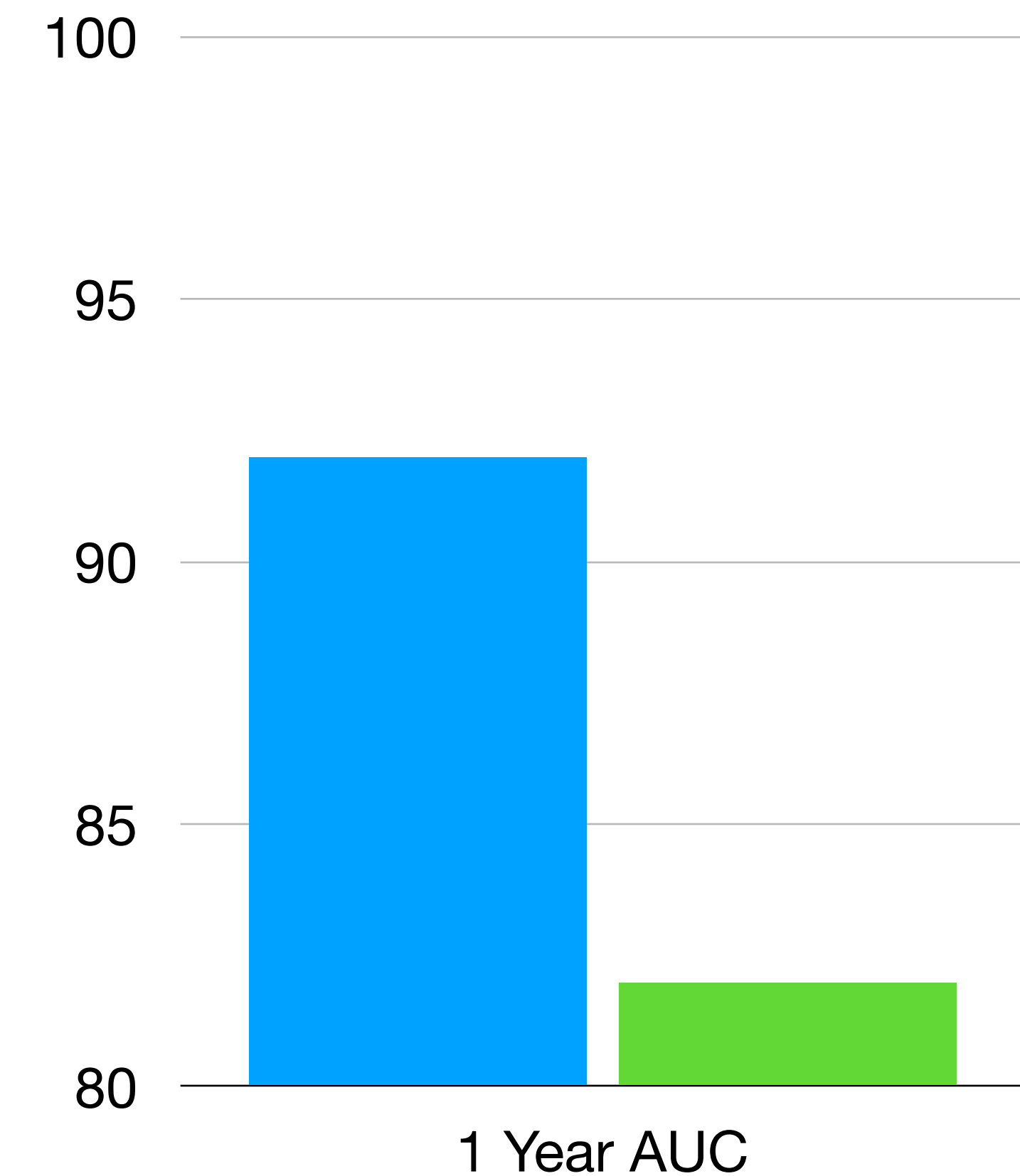
Can localization help improve radiologist **specificity or efficiency**?

Motivating Example:

Localization as a regularizer



Why does this help?



Agenda

Recap

Motivation for Localization

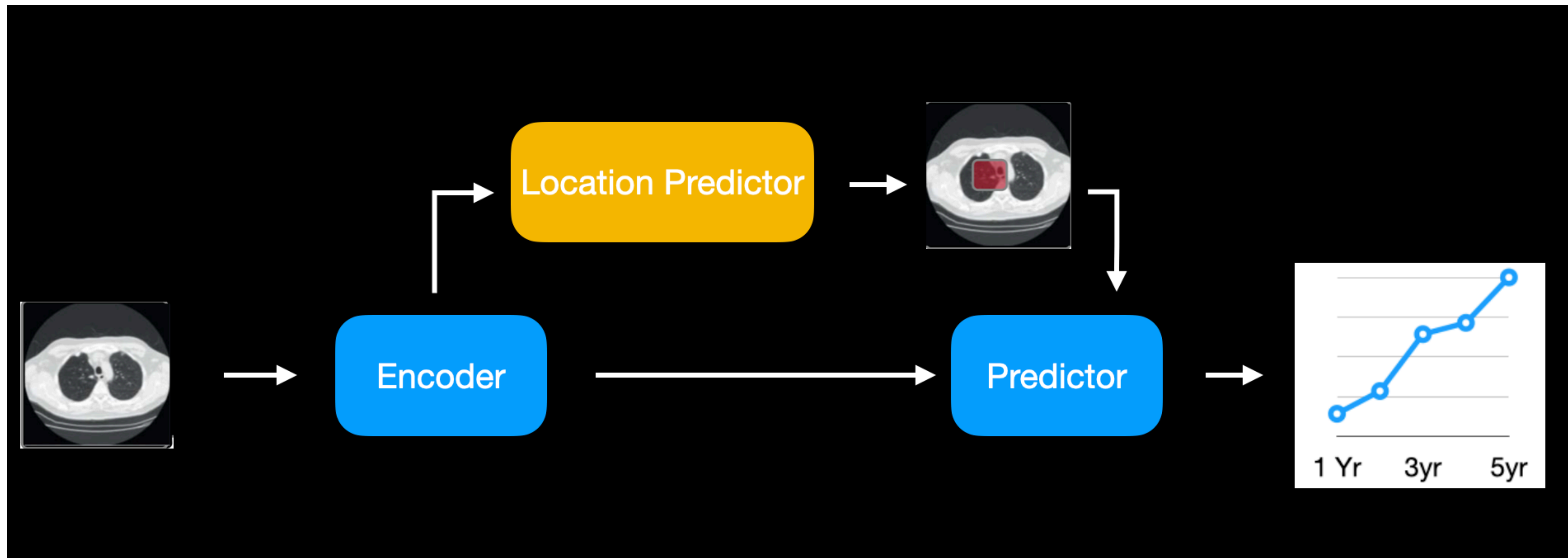
Localization as Attention

Bounding box prediction

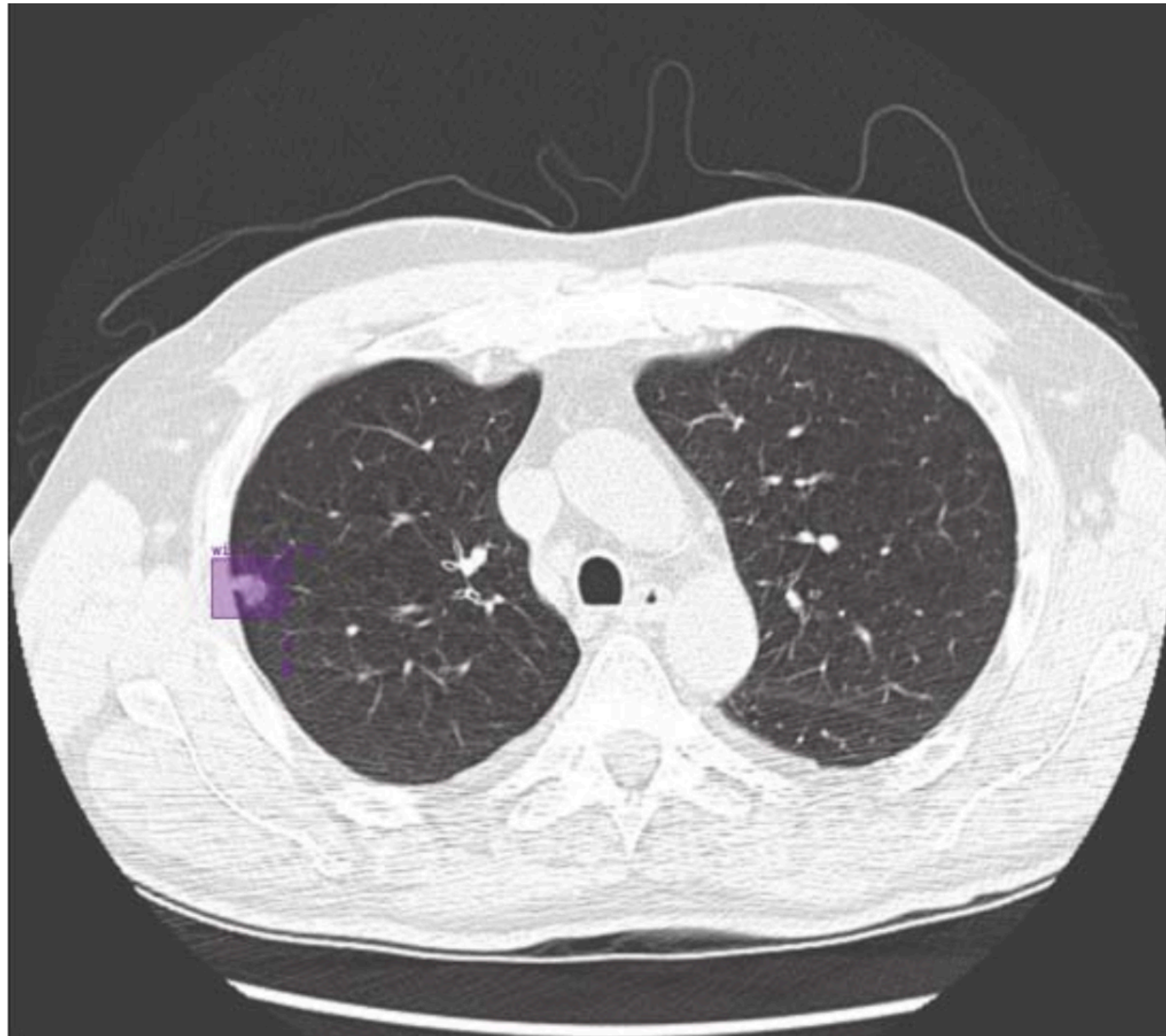
Segmentation

Localization as attention map supervision:

Localization as a regularizer



Problem Setting



$$X \in \mathbb{R}^{\begin{matrix} w & H & D \\ \downarrow & \downarrow & \downarrow \\ 512, 512, 200 \end{matrix}}$$

Radiologist drew bounding boxes
for each cancer

$$b_i =$$

Problem Setting



$$\begin{array}{ccc} w & H & D \\ \downarrow & \downarrow & \downarrow \\ 512, 512, 200 \end{array}$$
$$X \in \mathbb{R}$$

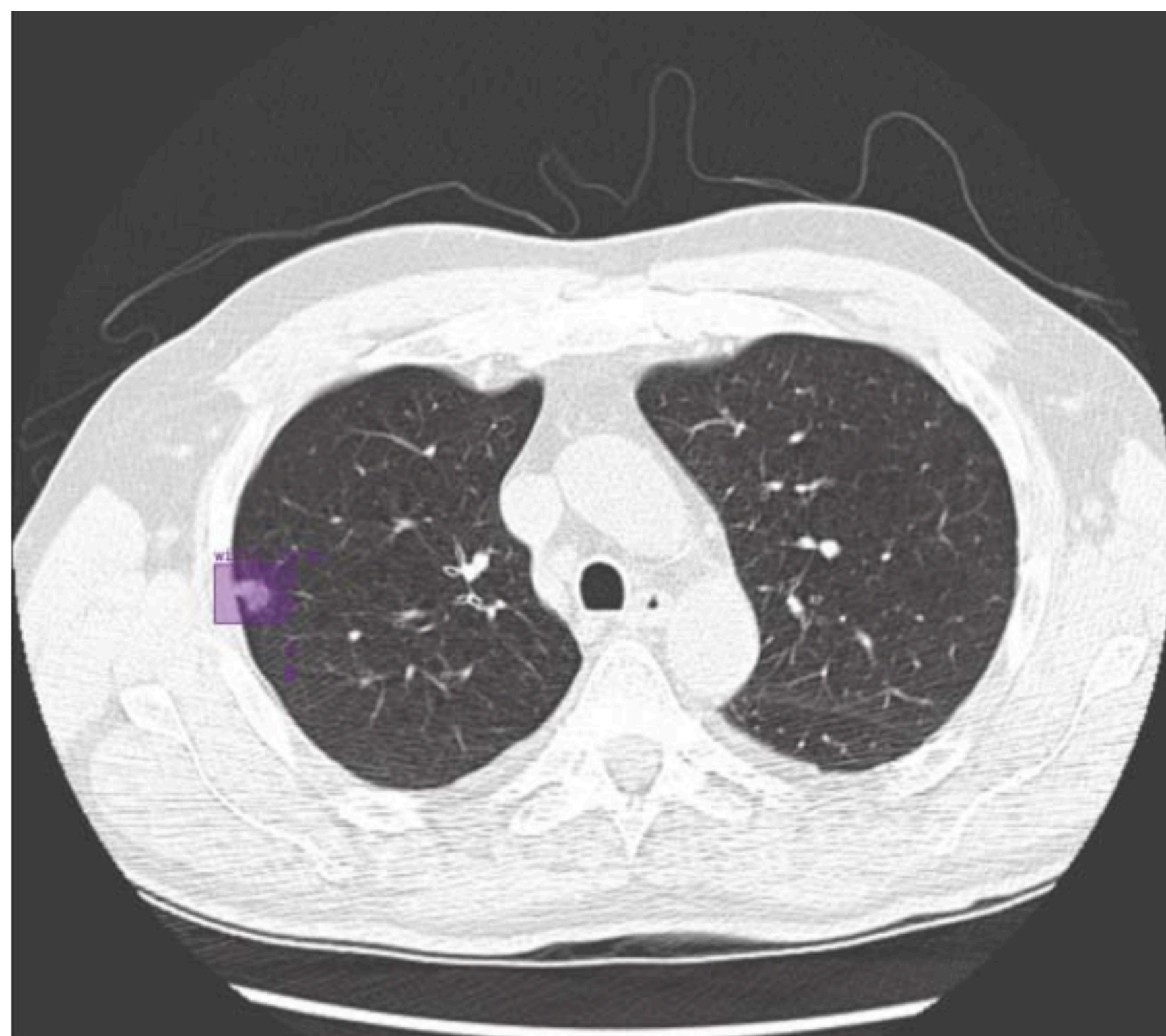
Radiologist drew bounding boxes
for each cancer

$$b_i = (c_x, c_y, c_z, w, h)$$

Rewrite \sim

$$A \in \mathbb{R}^{512, 512, 200} \leftarrow \text{Binary Mask}$$

Problem Setting: Feature Encoding



$$X \rightarrow \mathbb{R}^{H, W, D}$$

$$X \rightarrow \boxed{\text{CNN}} \rightarrow Z^{L-1}$$

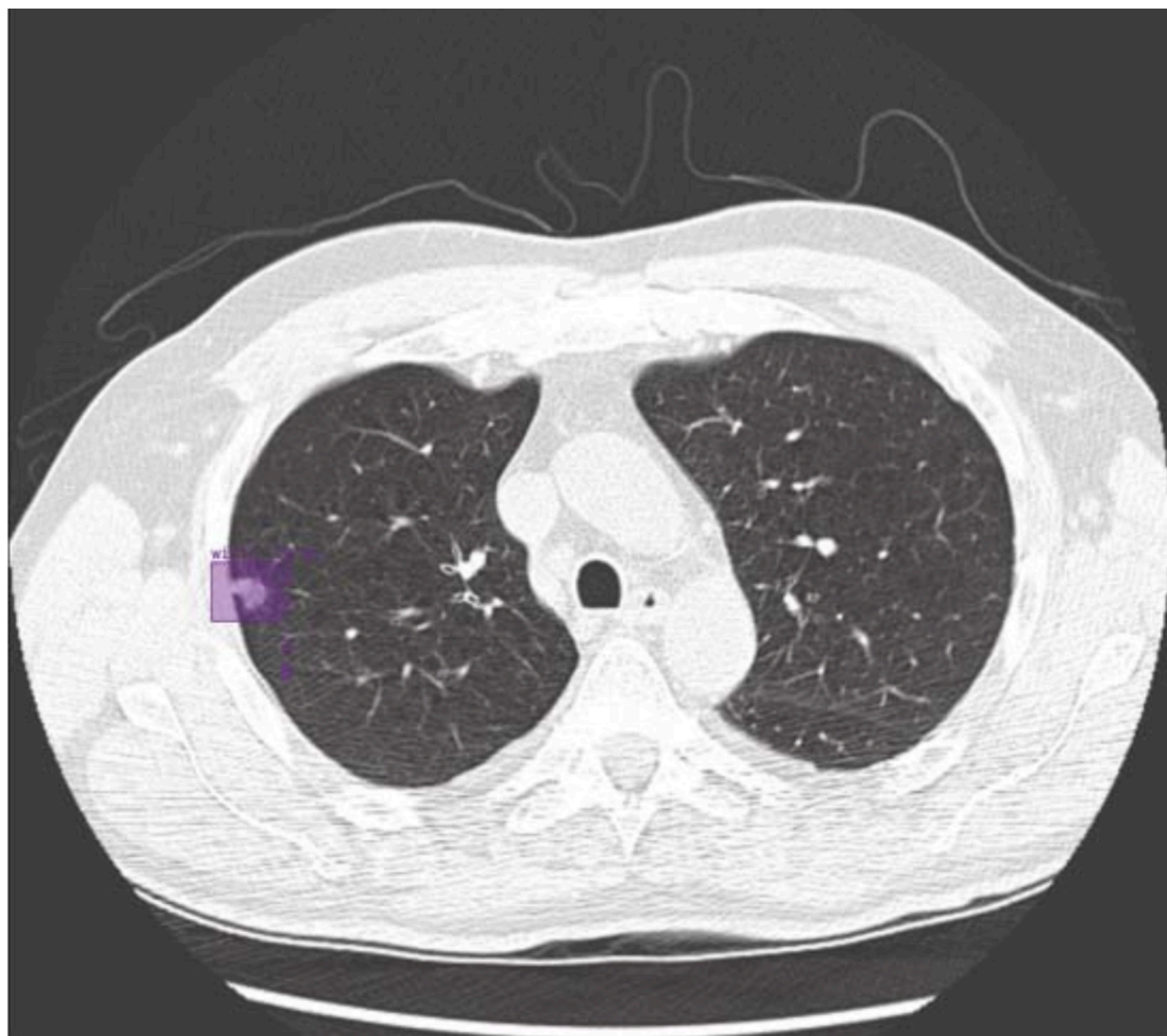
128, 128, 50

$$\mathbb{R}^{n, w, d}$$

$$Z \rightarrow [\text{AttnPool}(Z^{L-1}), \text{AttnPool}(Z^{L-1}), \dots]$$

$$p \rightarrow \text{FFN}(z) \rightarrow \sigma(\theta z + b)$$

Problem Setting: Feature Encoding



$$X \rightarrow \mathbb{R}^{H, W, D}$$

$$X \rightarrow \boxed{\text{CNN}} \rightarrow Z^{L-1}$$

$$Z \rightarrow [\text{AttnPool}(Z^{L-1}), \text{AttnPool}(Z^{L-1}), \dots]$$

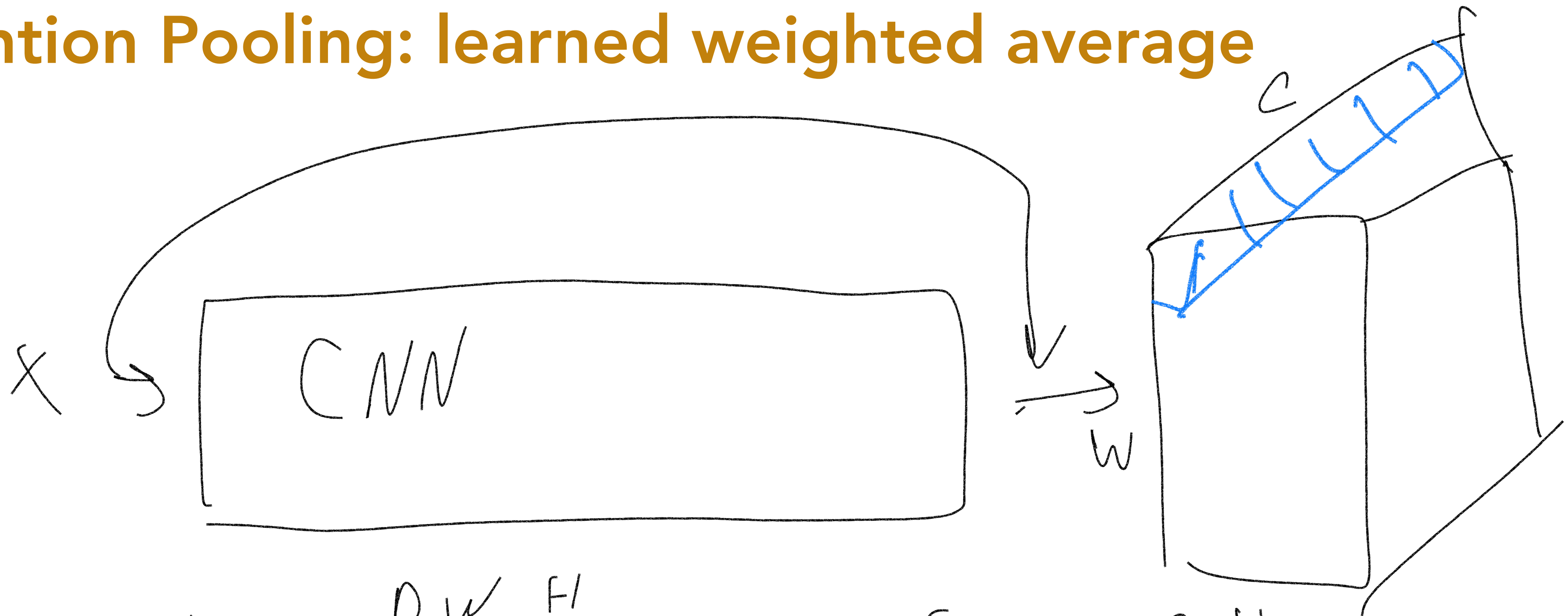
$$P \rightarrow \text{FFN}(Z) \rightarrow \sigma(\Theta Z + b)$$

$$A \rightarrow \text{Downsample} \rightarrow A^* \in \mathbb{R}^{h, w, d}$$

128, 128, 50
↓
 $\mathbb{R}^{h, w, d}$
Pooling
coarsens
spatial
resolution

Binary
Mask

Attention Pooling: learned weighted average



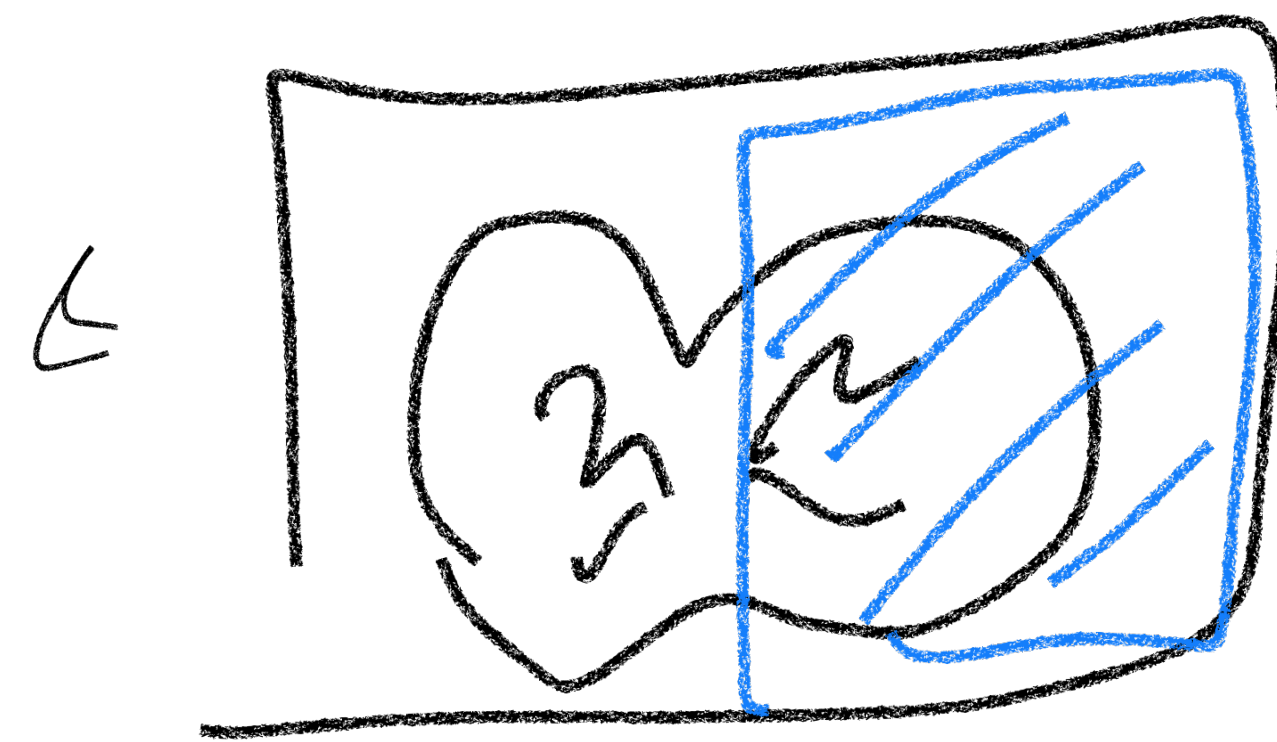
$$\text{Attn Pool}(z) = \sum_k \sum_i^D \sum_j^F \alpha_{i,j,k} z[i, j, k]^H$$

$$\alpha = \text{softmax}(f(z)) \quad |f(z)| \ni (w, H, D, 1)$$

Supervising Attention maps: 2D example

$$X = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

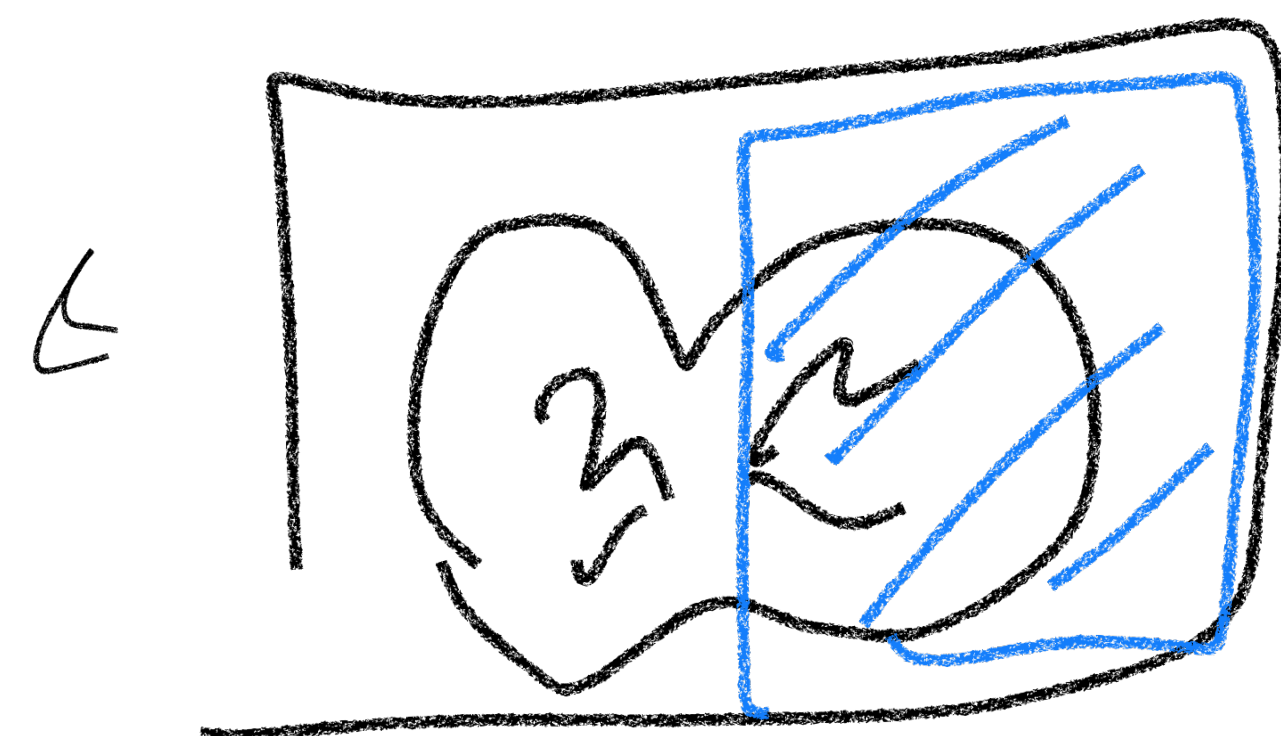
$$A = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$



Supervising Attention maps: 2D example

$$\alpha = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$



Agreement $\rightarrow \alpha \cdot A = 0.6$

$$L_{\text{Attn}}(\alpha, A) = -\log(\alpha \cdot A)$$

Will this use negative unsh?

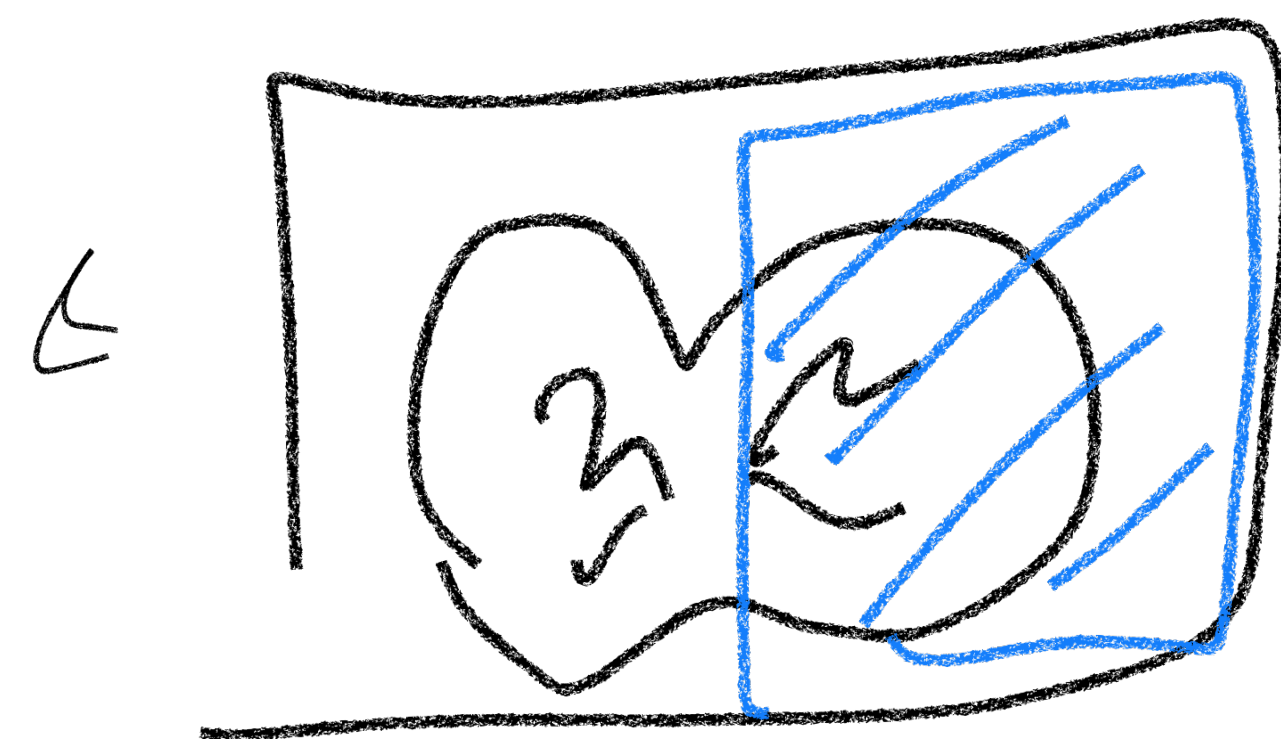
Yes! Soft max!

$$\text{softmax}(z) = \frac{e^z}{\sum e^{z_i}}$$

Supervising Attention maps: 2D example

$$\alpha = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$



Agreement $\rightarrow \alpha \cdot A = 0.6$

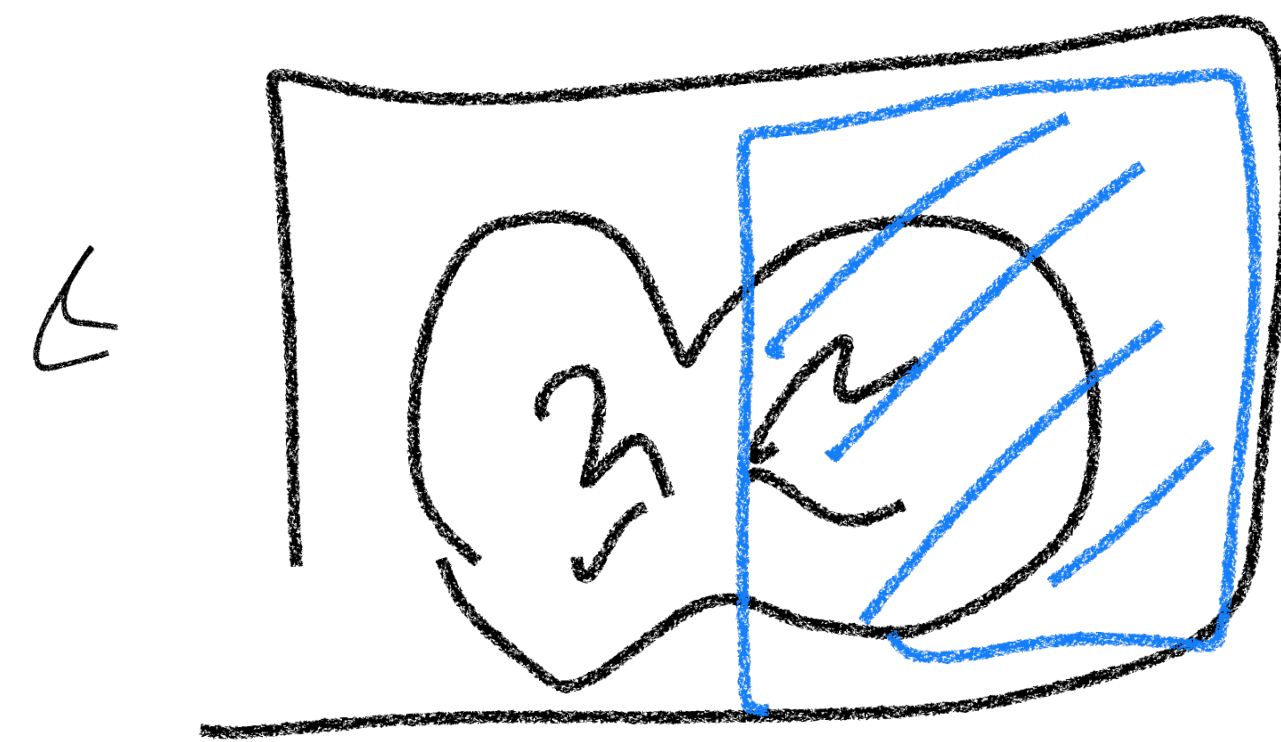
$$L_{\text{Attn}}(\alpha, A) = -\log(\alpha \cdot A)$$

why not cross ent per pixel?

Supervising Attention maps: 2D example

$$\alpha = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$



Agreement $\rightarrow \alpha \cdot A = 0.6$

$$L_{\text{Attn}}(\alpha, A) = -\log(\alpha \cdot A)$$

Why not cross ent per pixel?

$$L = -\sum A_i \log \alpha_i + (1 - A_i) \log(1 - \alpha_i)$$

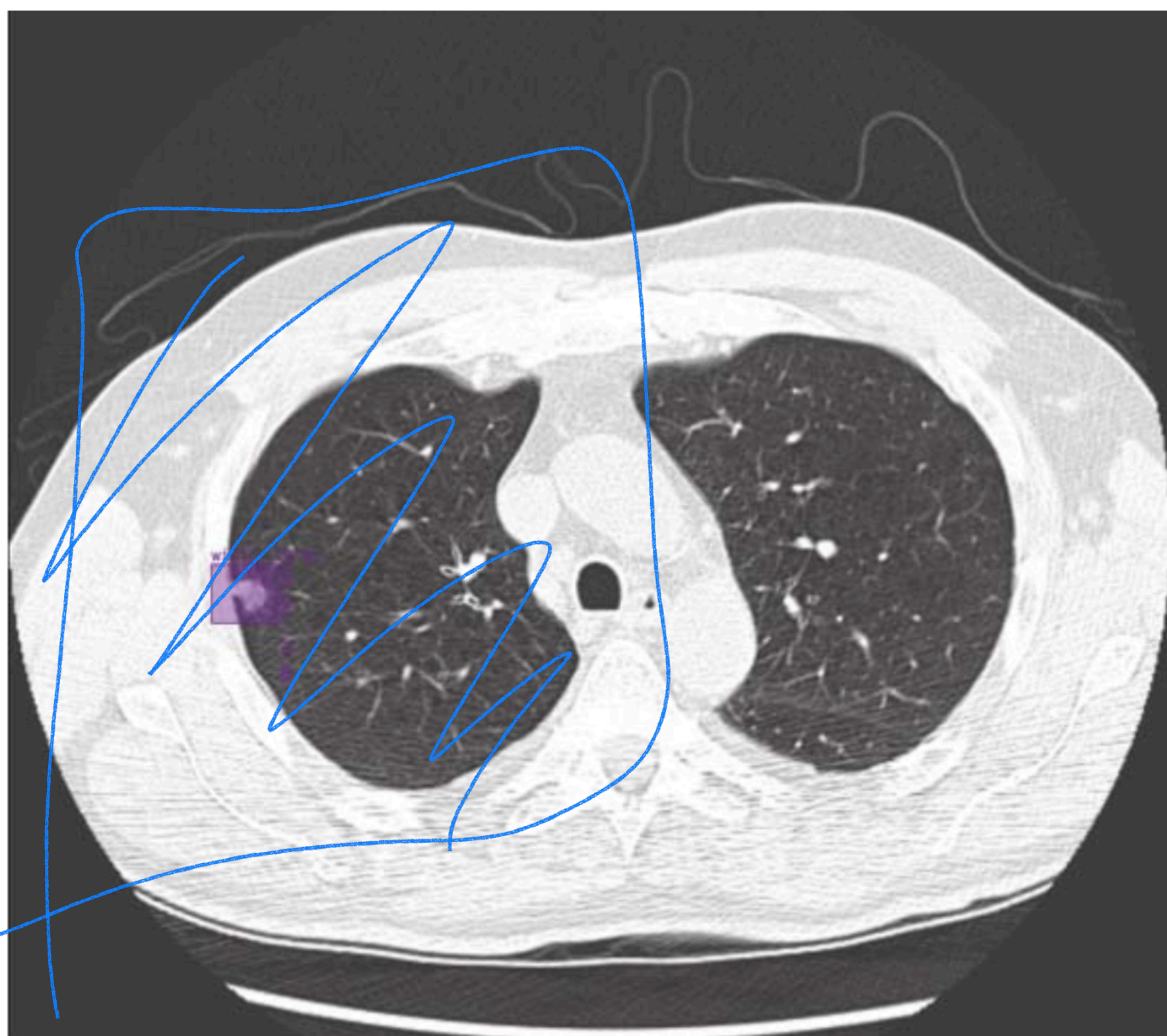
No good solution due
to $\sum A \rightarrow 1$, and softmax

Coarse supervision for attention maps

What if we don't know *exact*
location?

Ex: Cancer Risk Prediction

Cancer in left lung in
5 years



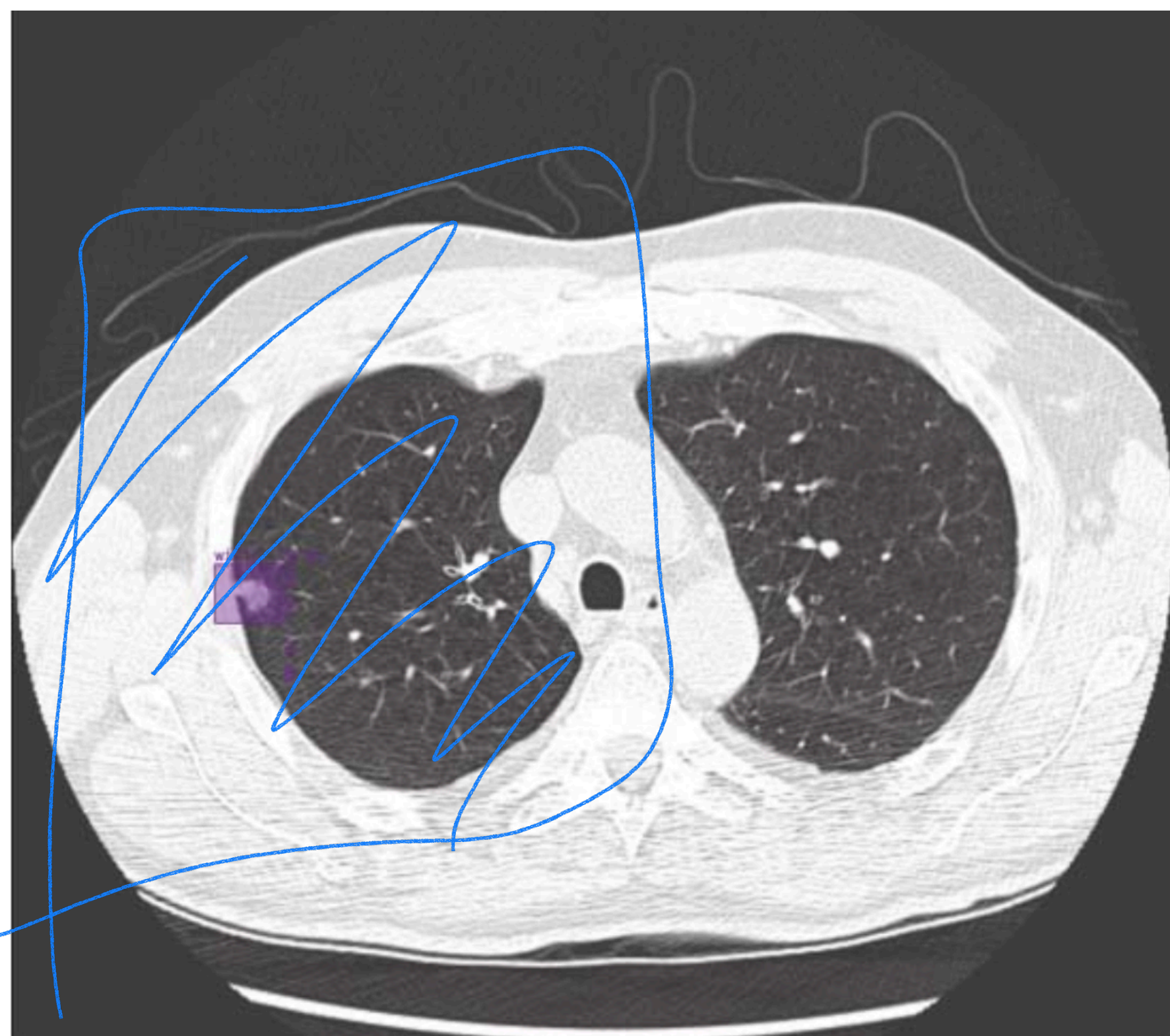
Coarse supervision for attention maps

What if we don't know exact location?

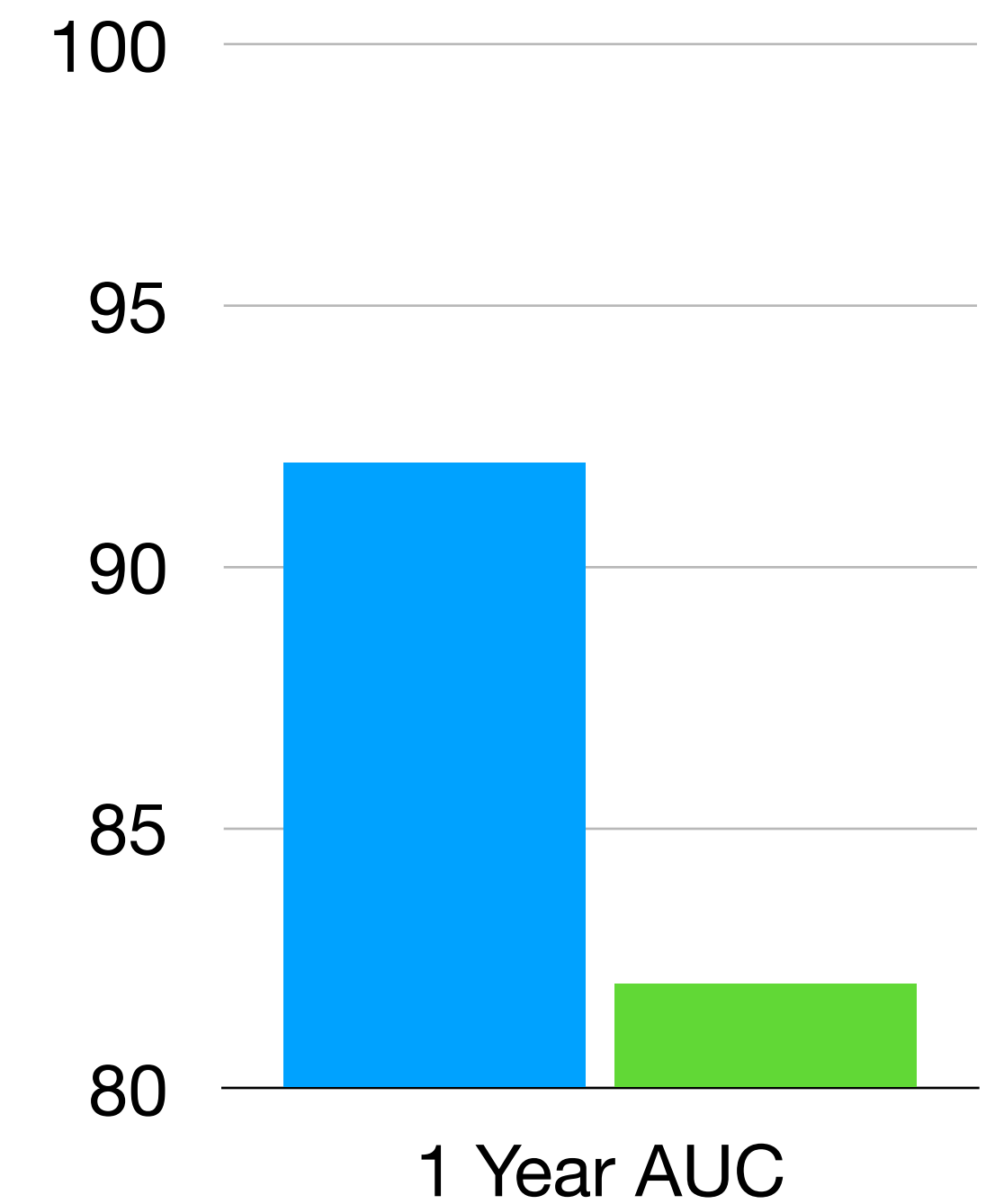
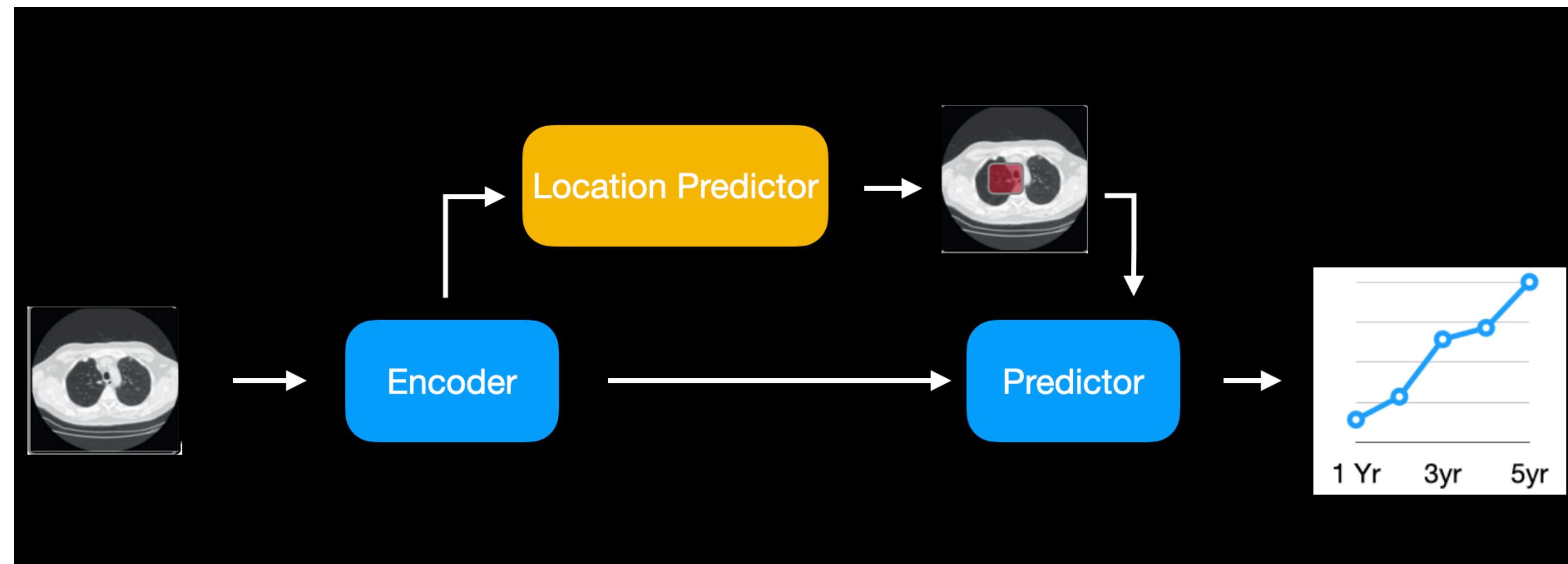
Ex: Cancer Risk Prediction

Cancer in left lung in
5 years

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}, \quad L_{\text{Attn}}(\theta, A) = -\log(\theta \cdot A)!$$

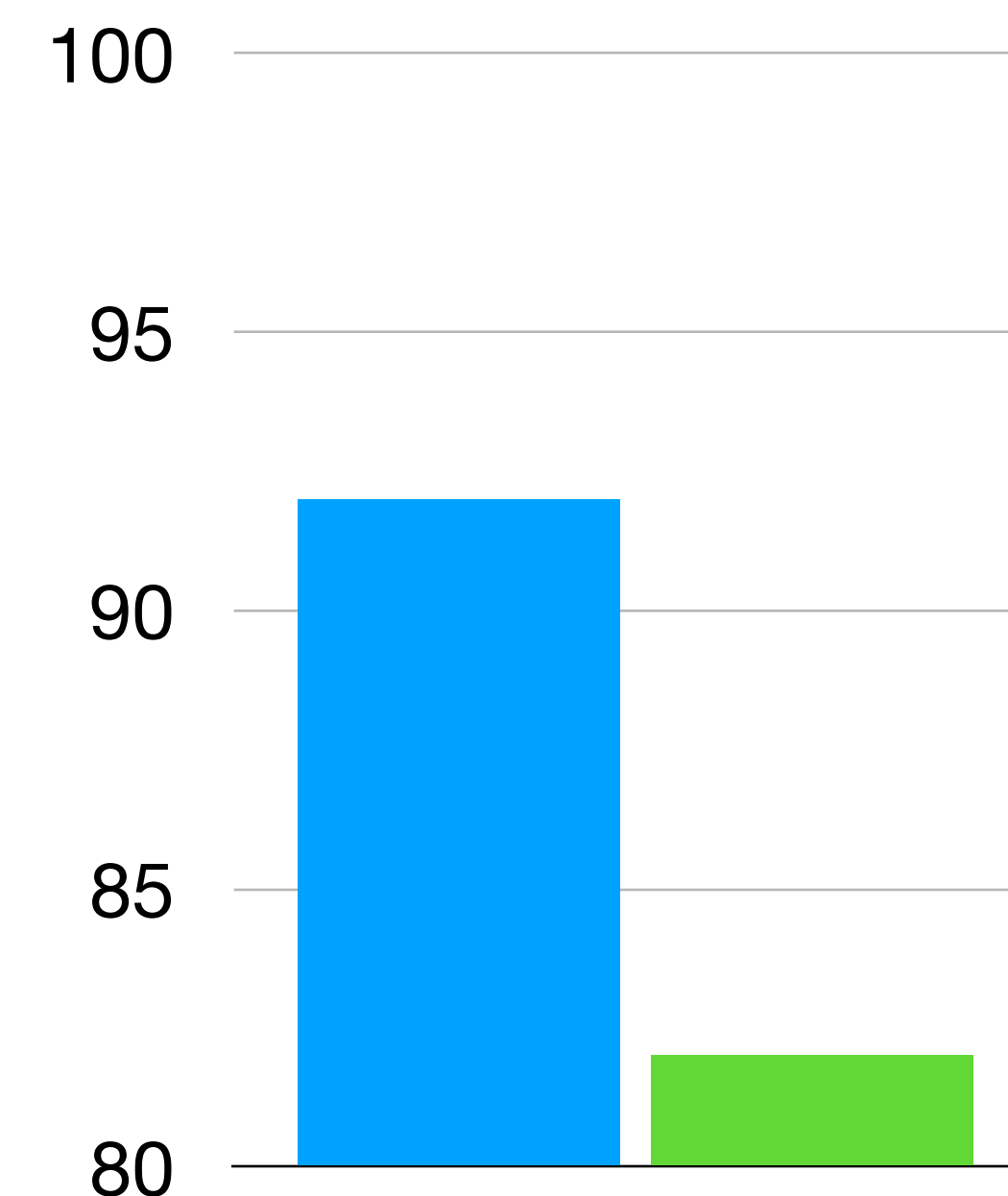
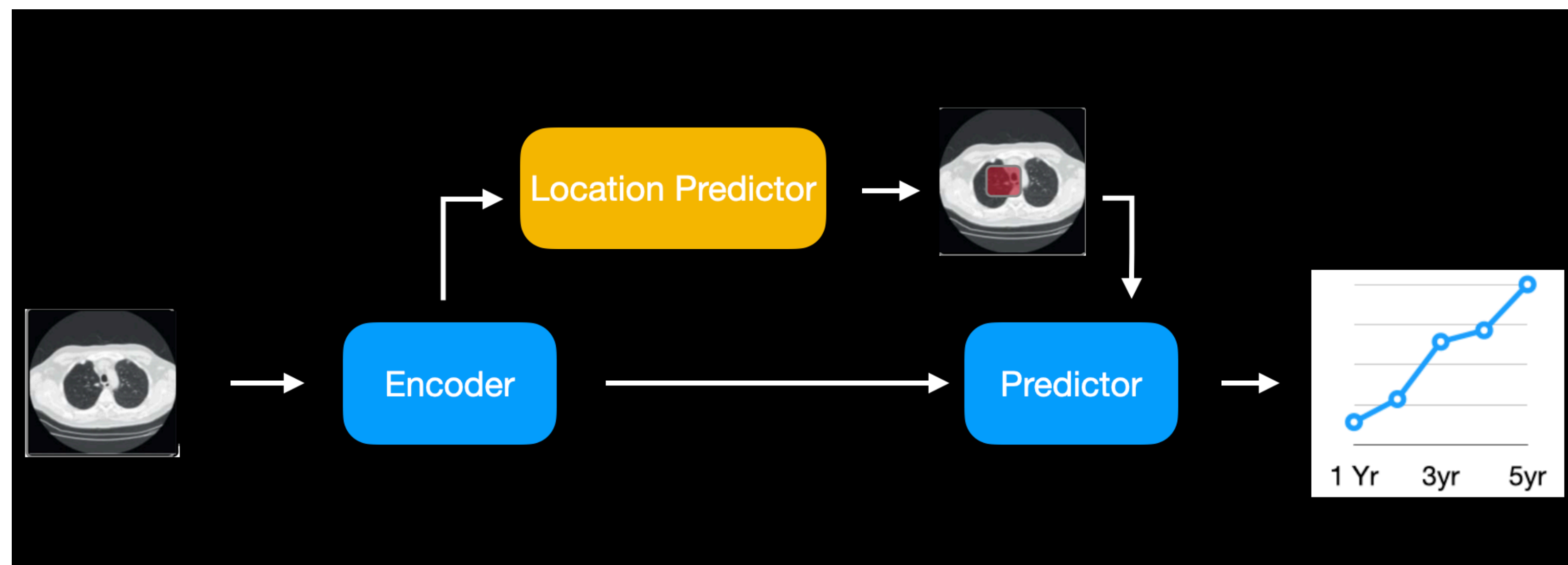


Putting it all together



$$L = L_{\text{Risk}} + \lambda L_{\text{Attn}} + \lambda L_{\text{Attn}}$$

Putting it all together

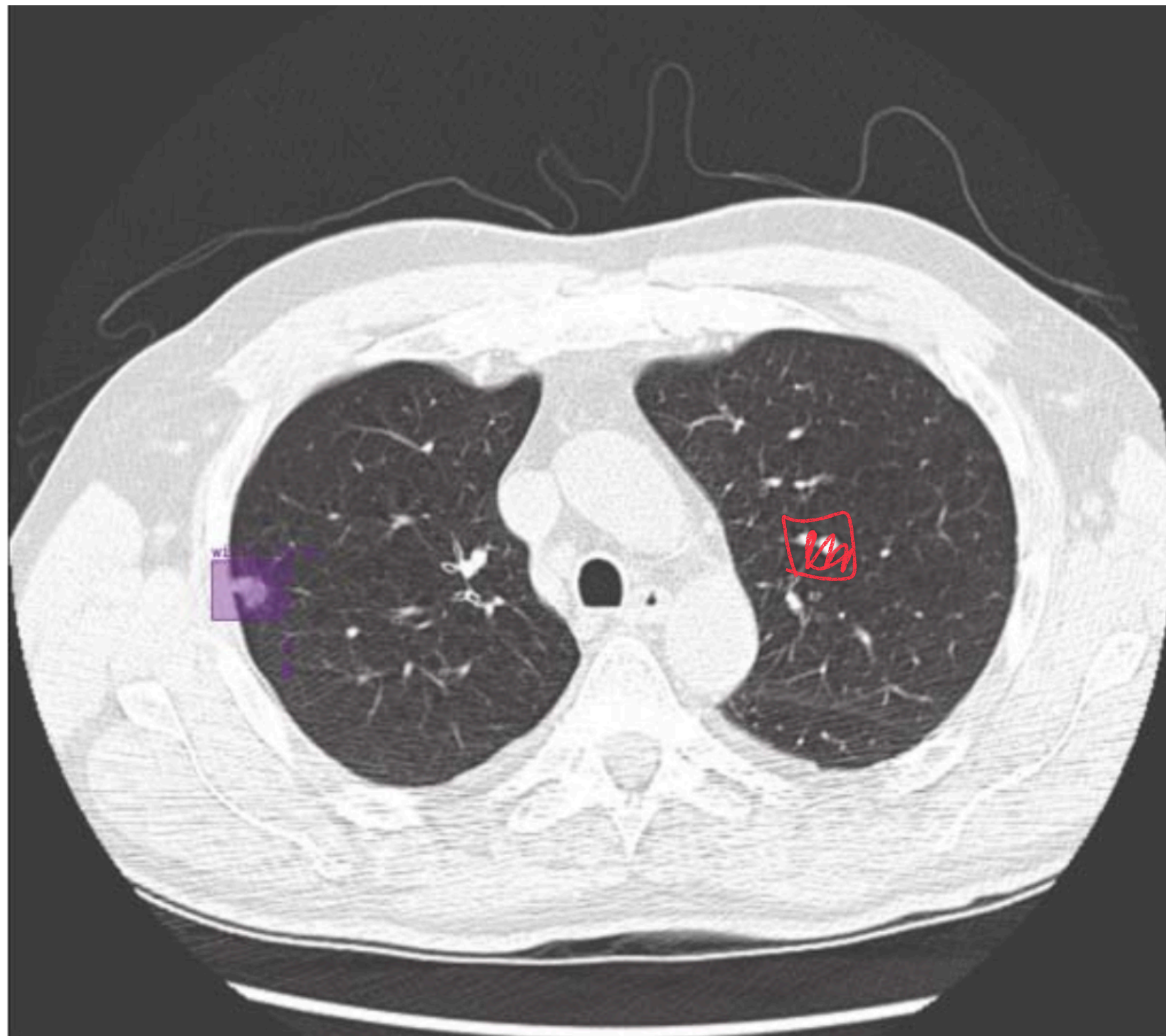


$$L = L_{\text{Risk}} + \lambda_{\text{Attn}} L_{\text{Attn}}$$

$$L_{\text{Risk}} = -\sum y_n \log(p_n(x)) + (1-y) \log(1-p_n(x))$$

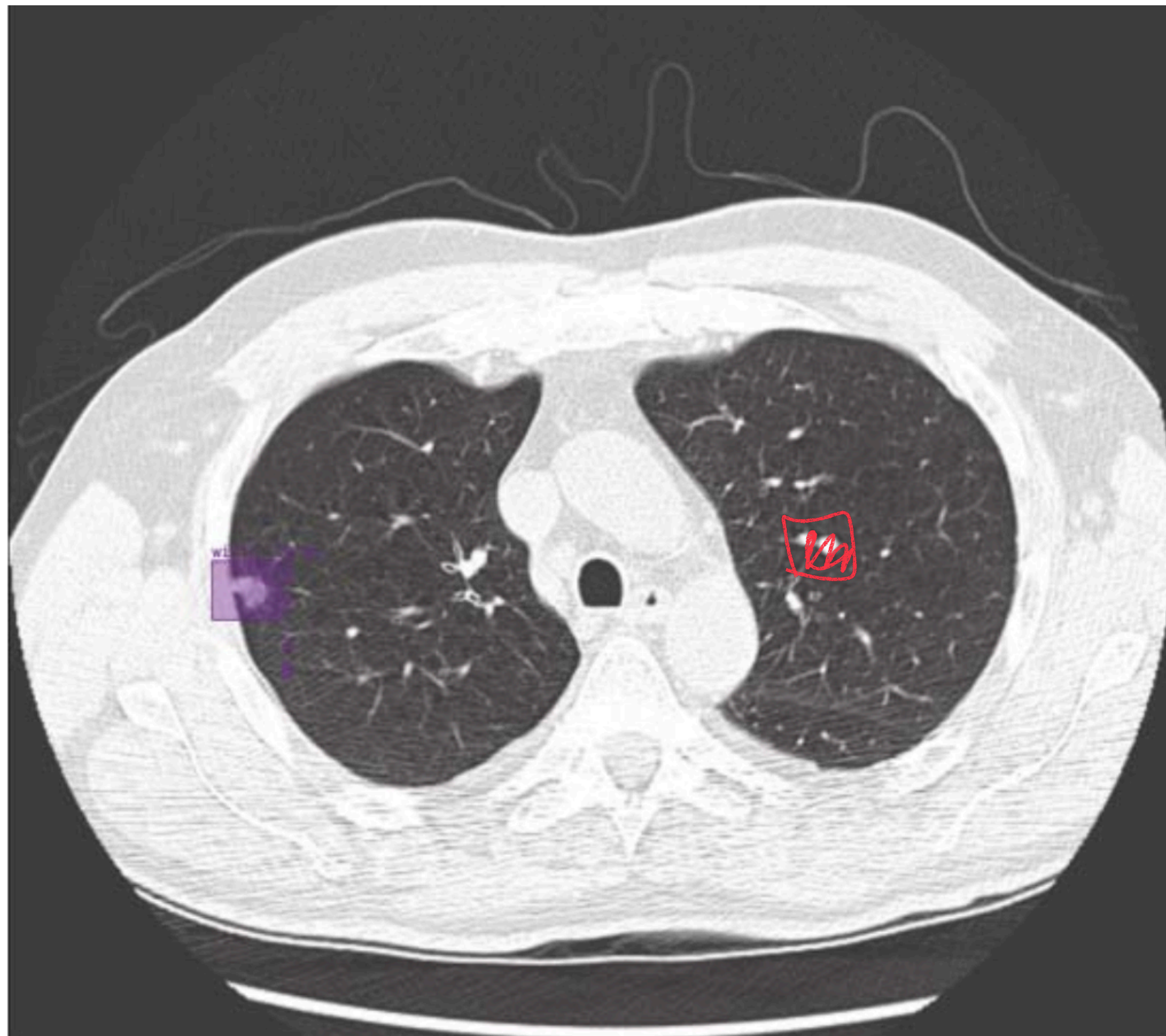
$$L_{\text{Attn}} = -\log(\text{CoA})$$

What's missing there? Beyond regularization



$$\mathcal{K} = \begin{bmatrix} 0.1 & 0.23 \\ 0.43 & \dots \\ \vdots & \vdots \end{bmatrix}$$

What's missing there? Beyond regularization

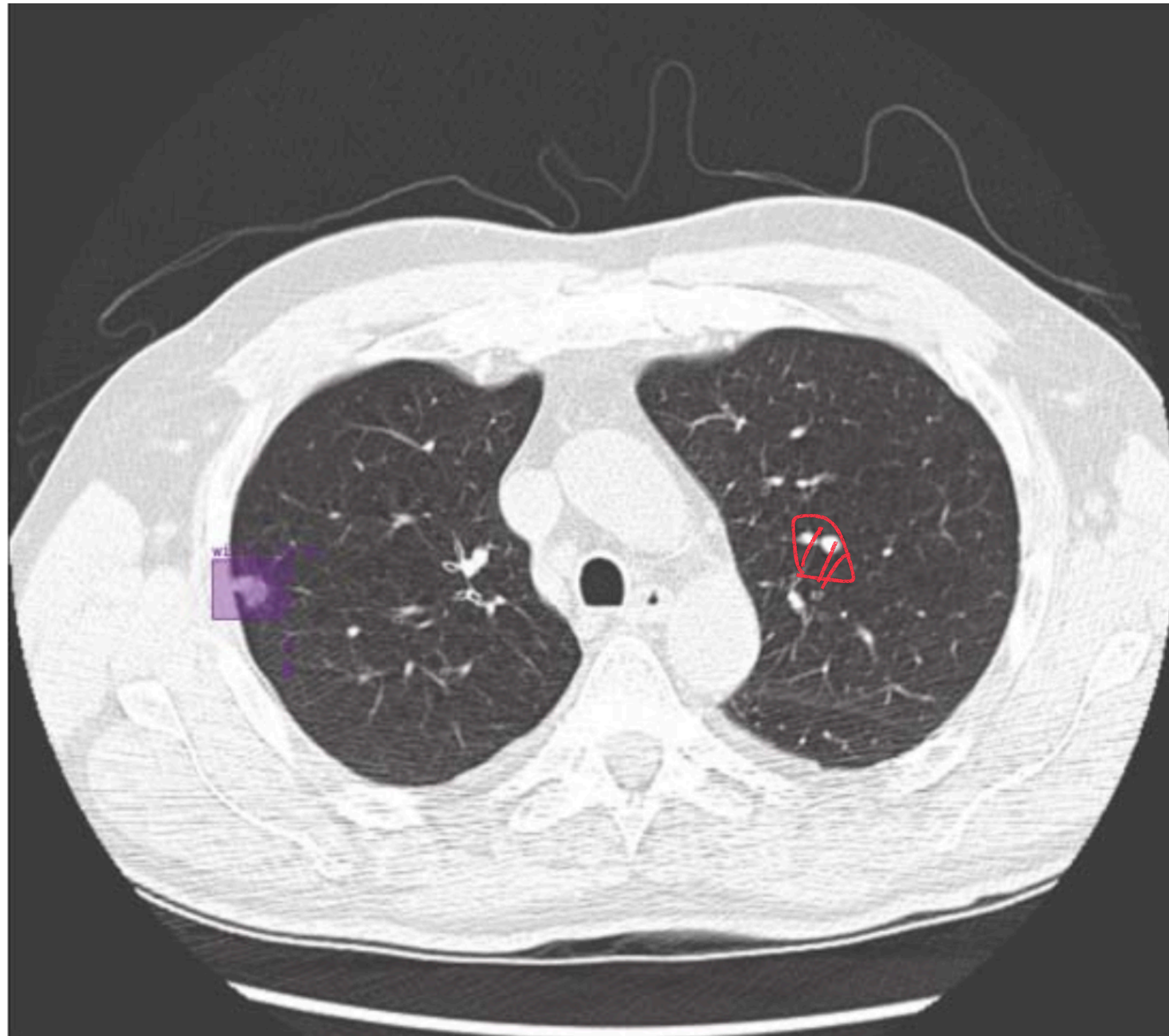


$$\mathcal{K} = \begin{bmatrix} 0.1 & 0.23 \\ 0.43 & \dots \\ \vdots & \vdots \end{bmatrix}$$

→ Not designed to give crisp bounding boxes

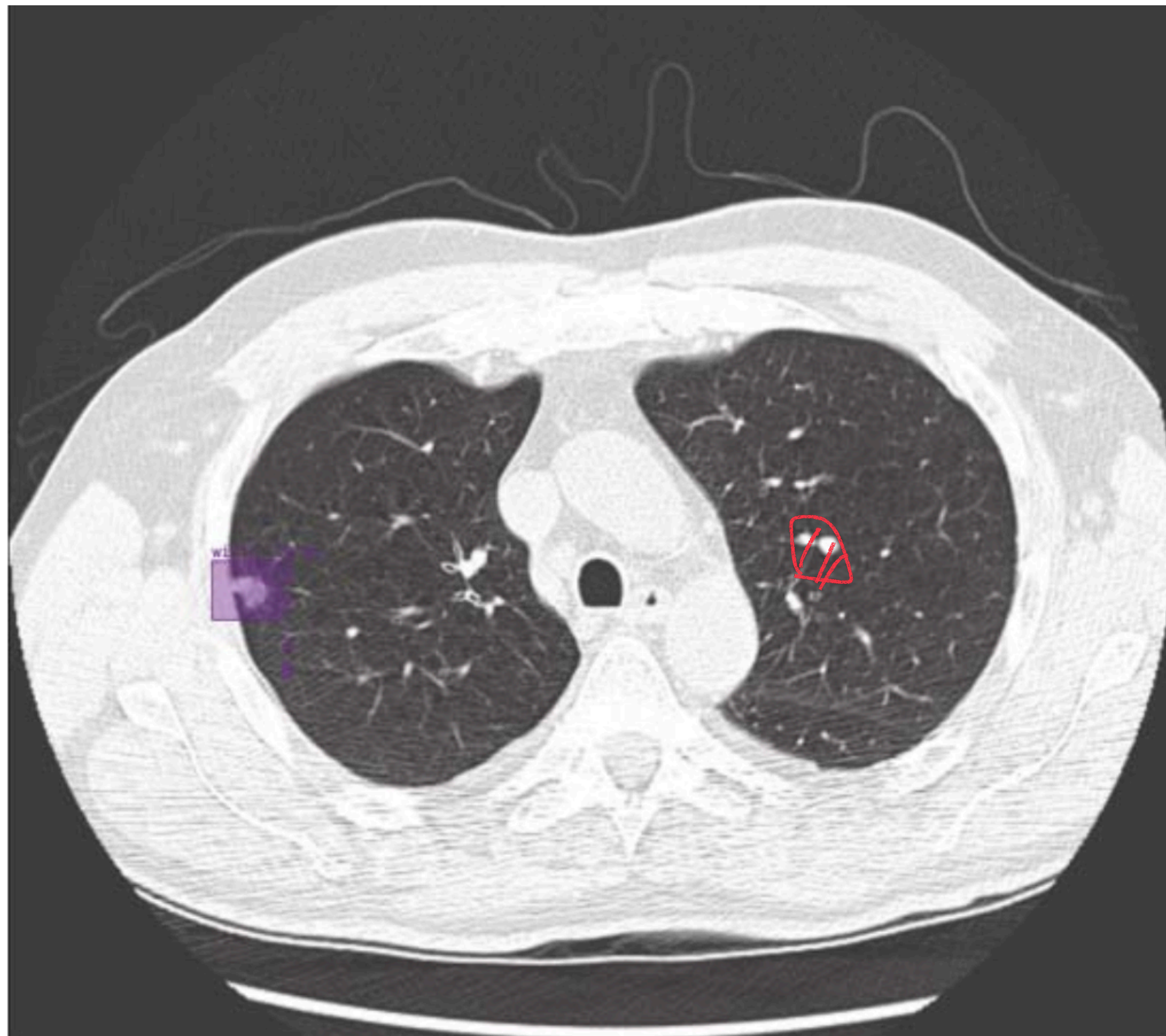
→ Under constrained for Multi Object!

What's missing there?



$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

What's missing there?



$$A = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$
$$v_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$
$$v_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$
$$v_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

⌞ Attention Equivalent! Want multiple boxes!

Agenda

Recap

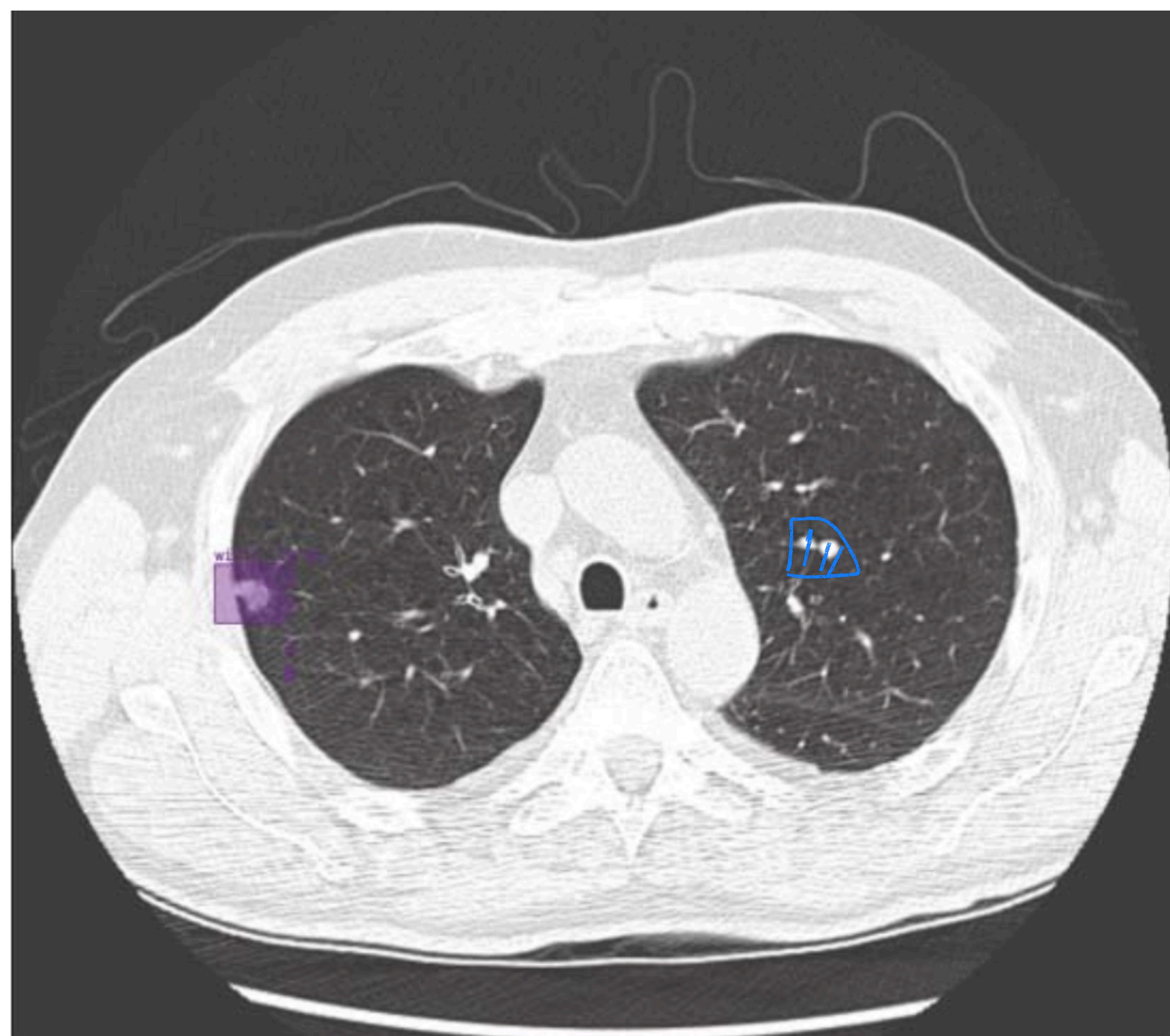
Motivation for Localization

Localization as Attention

Bounding box prediction

Segmentation

Problem Setting



$$X \in \mathbb{R}^{\begin{matrix} w & H & D \\ \downarrow & \downarrow & \downarrow \\ 512, 512, 200 \end{matrix}}$$

Radiologist drew bounding boxes for each cancer

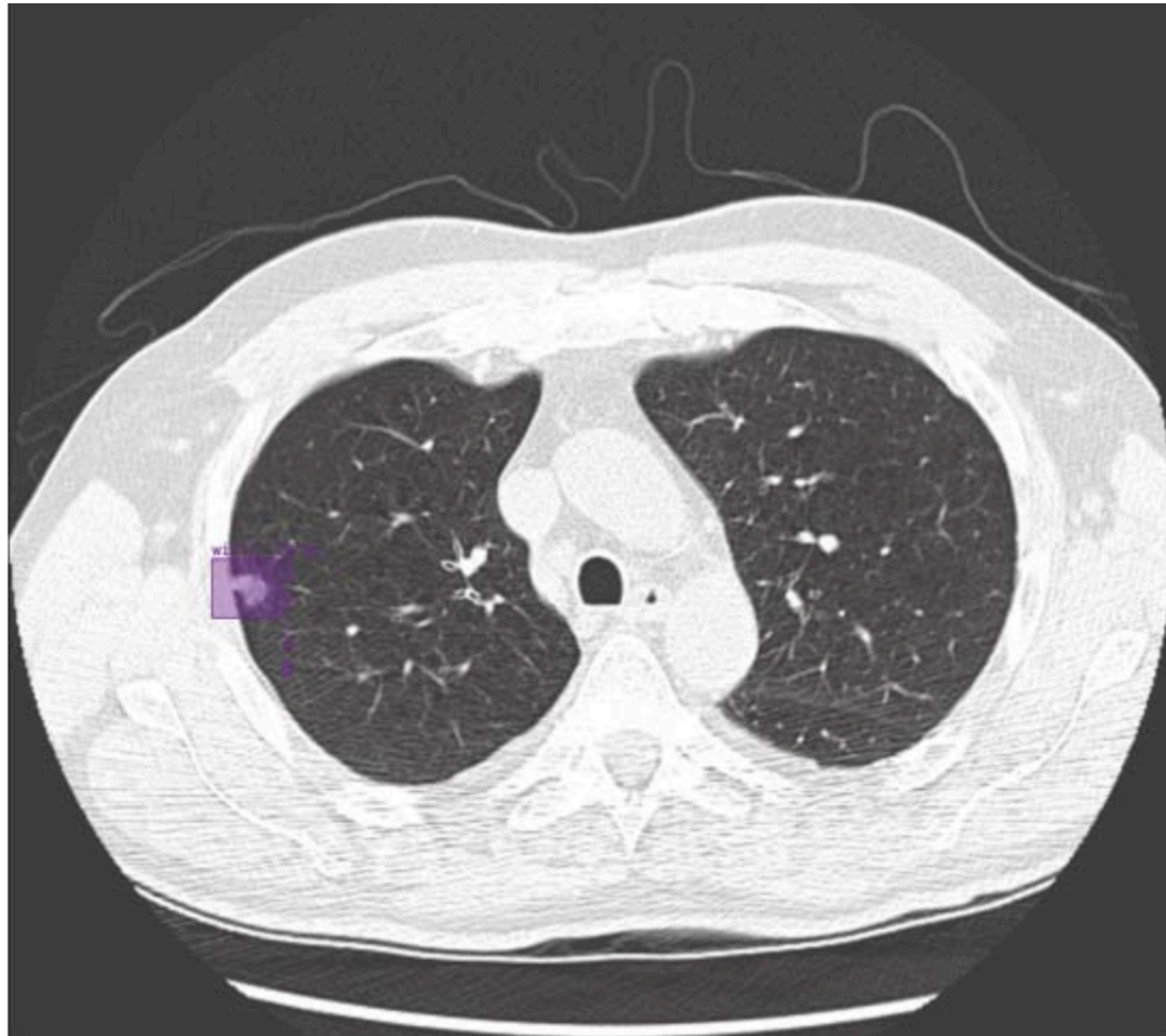
$$b_i = (p, c_x, c_y, c_z, w, h, d)$$

is it cancer?

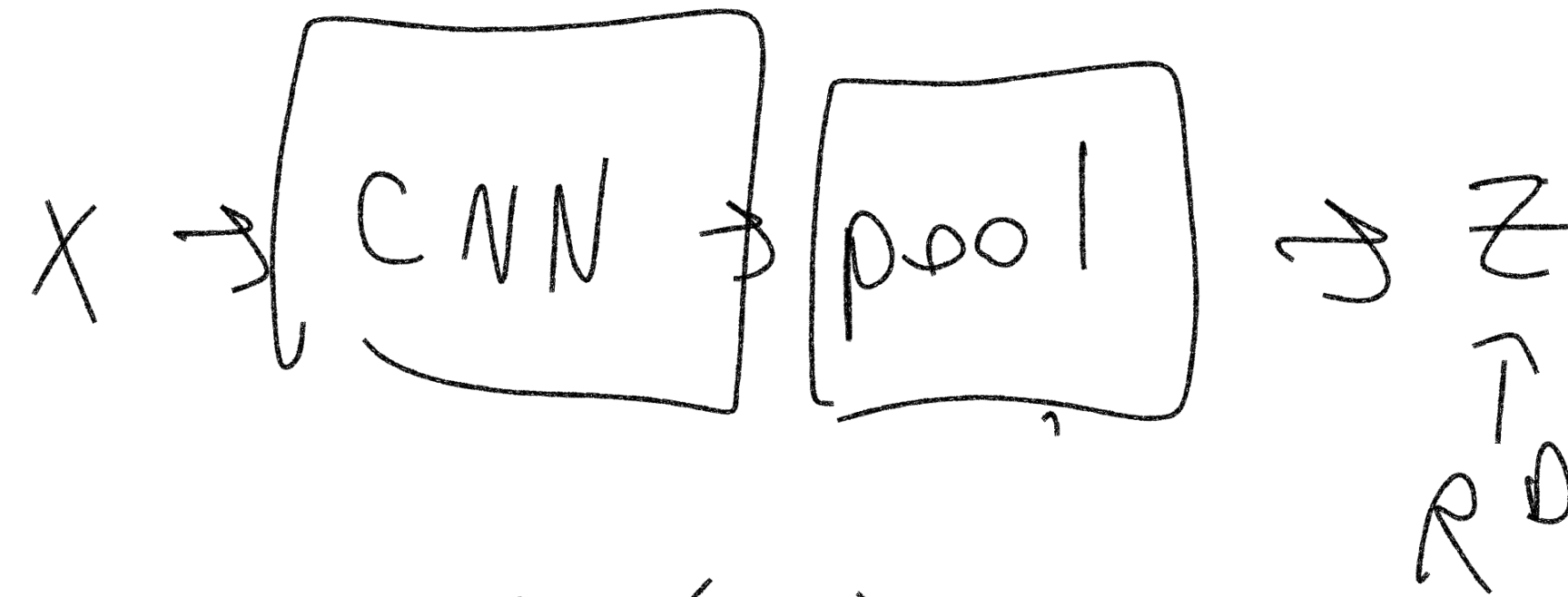
center coordinate

box dimensions

Simple baseline



$$b_i = (p, c_x, c_y, c_z, w, h, d)$$



$$p = \text{cls}(Z)$$

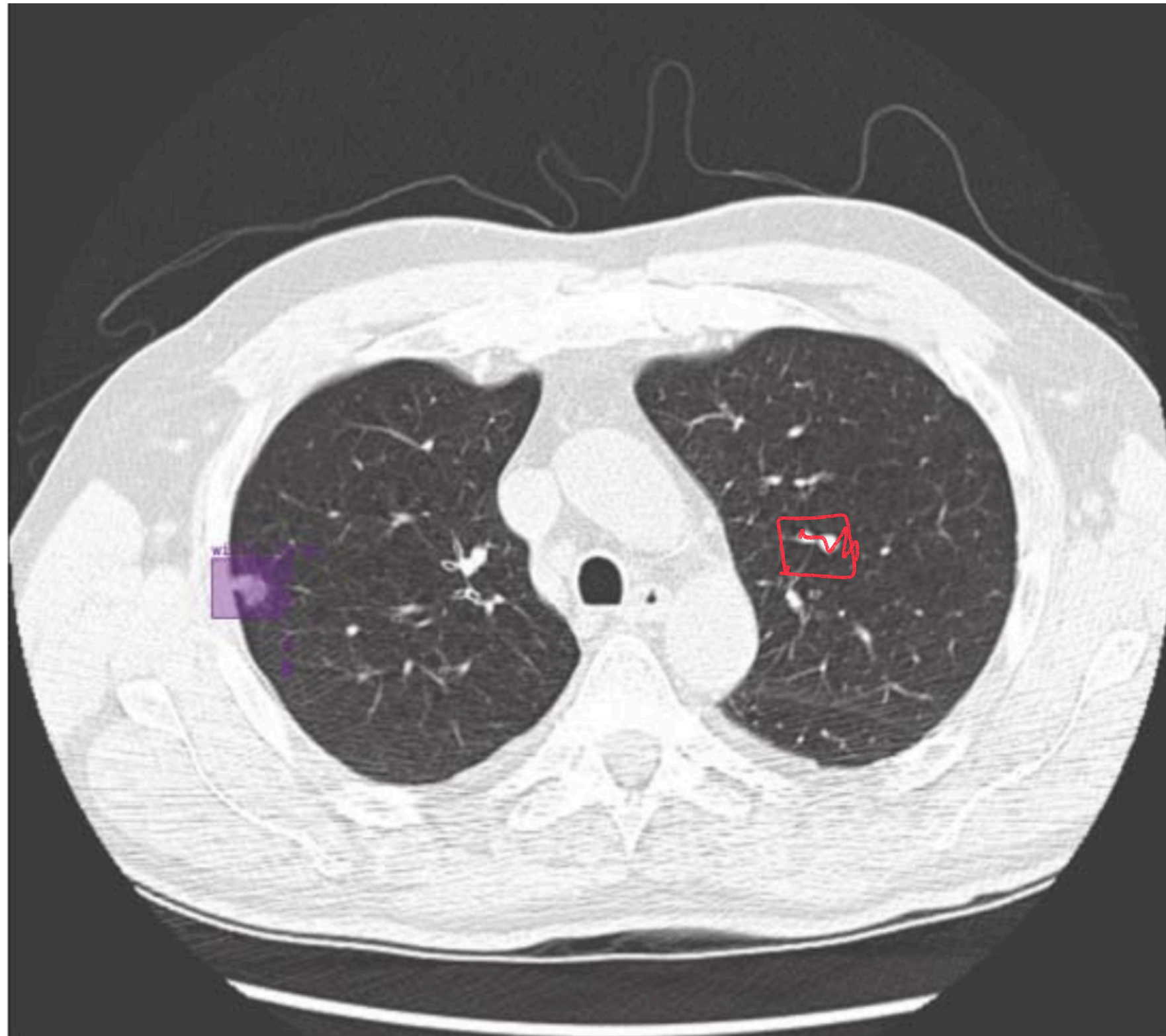
$$(\underline{c_x, c_y, c_z}) = \text{regress}(Z)$$

$$(\underline{w, h, d}) = \text{regress}(Z)$$

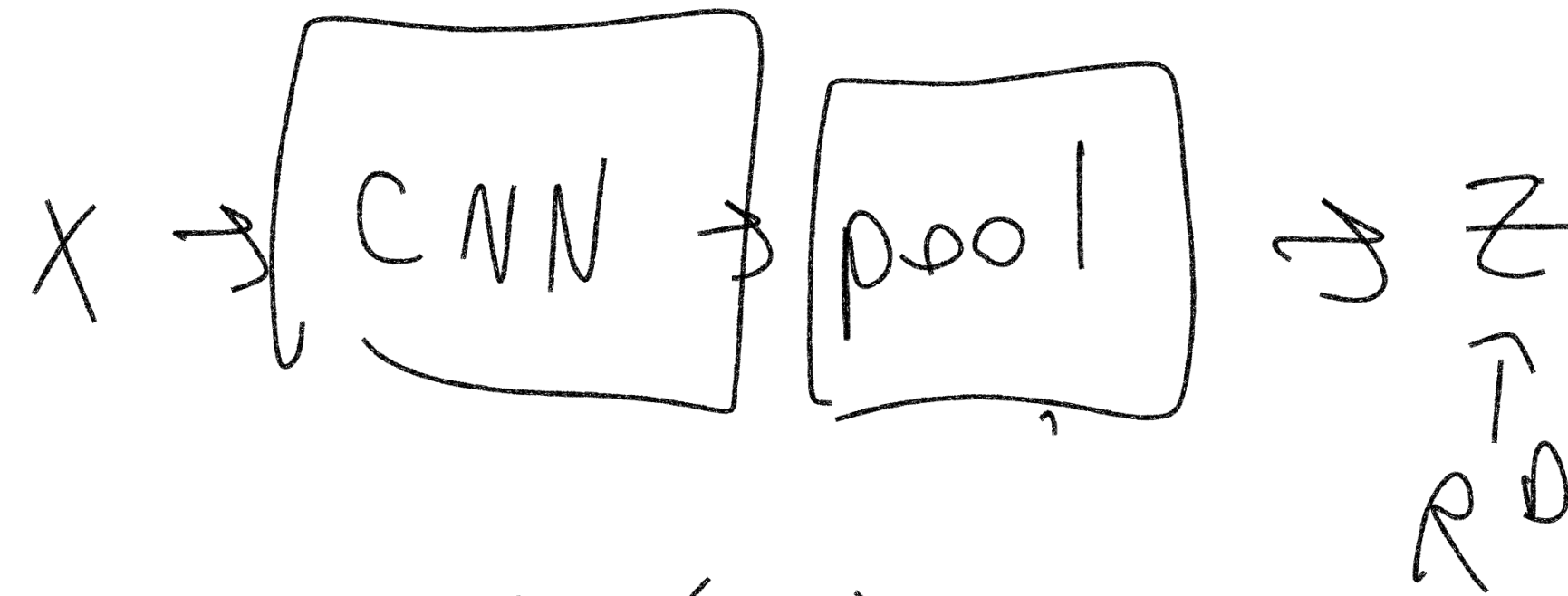
Will
this
work?

$$L = L_p + L_c + L_{\text{size}} \quad \leftarrow \text{MSE}$$

Simple baseline



$$b_i = (p, c_x, c_y, c_z, w, h, d)$$



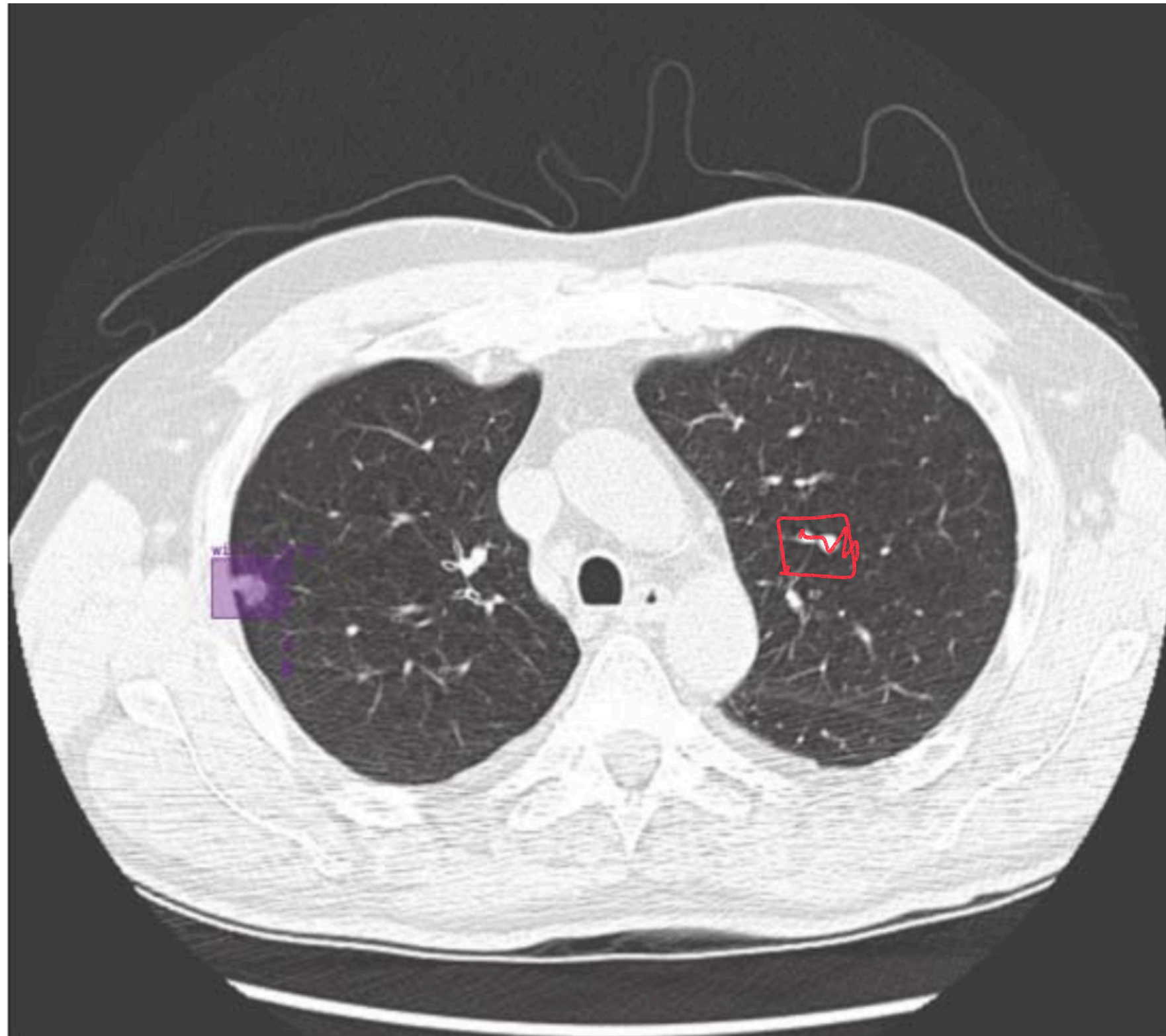
$$p = \text{cls}(Z)$$

$$(\underline{c_x, c_y, c_z}) = \text{regress}(Z)$$

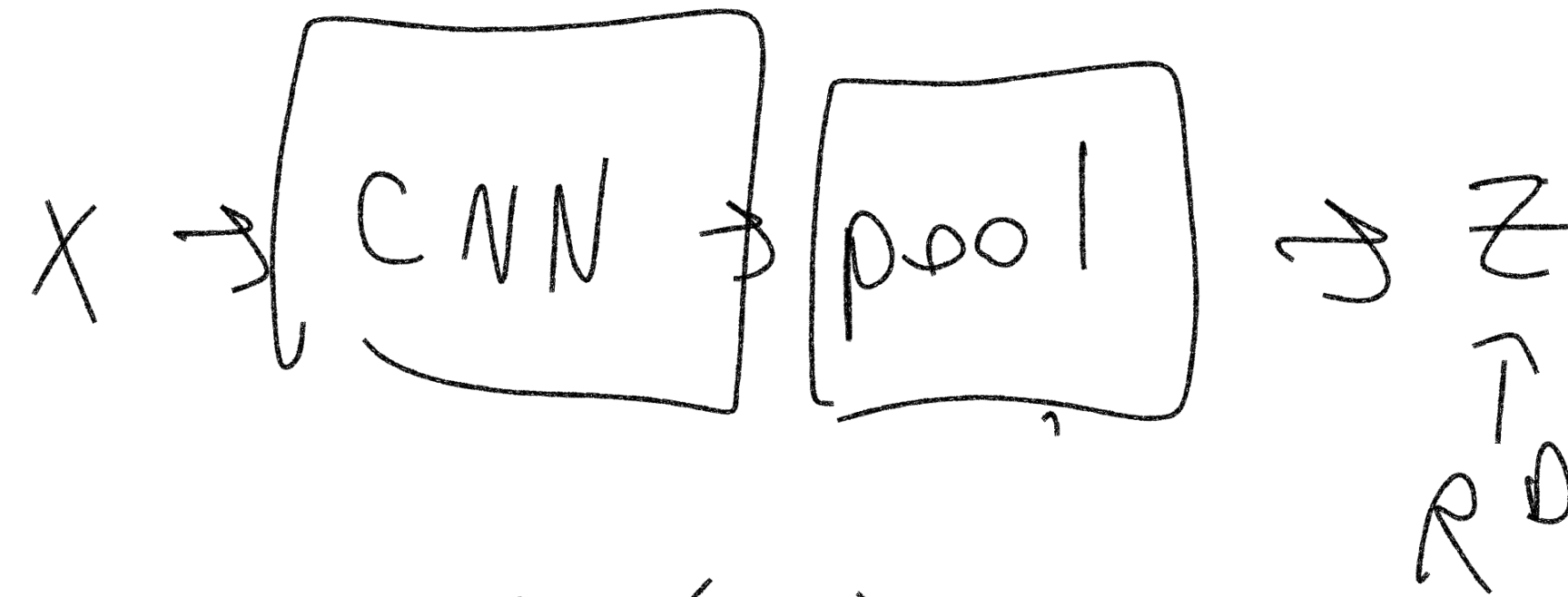
$$(\underline{w, h, d}) = \text{regress}(Z)$$

$$L = L_p + p L_c + p L_{\text{size}} \leftarrow \text{MSE}$$

Simple baseline



$$b_i = (p, c_x, c_y, c_z, w, h, d)$$



$$p = \text{cls}(Z)$$

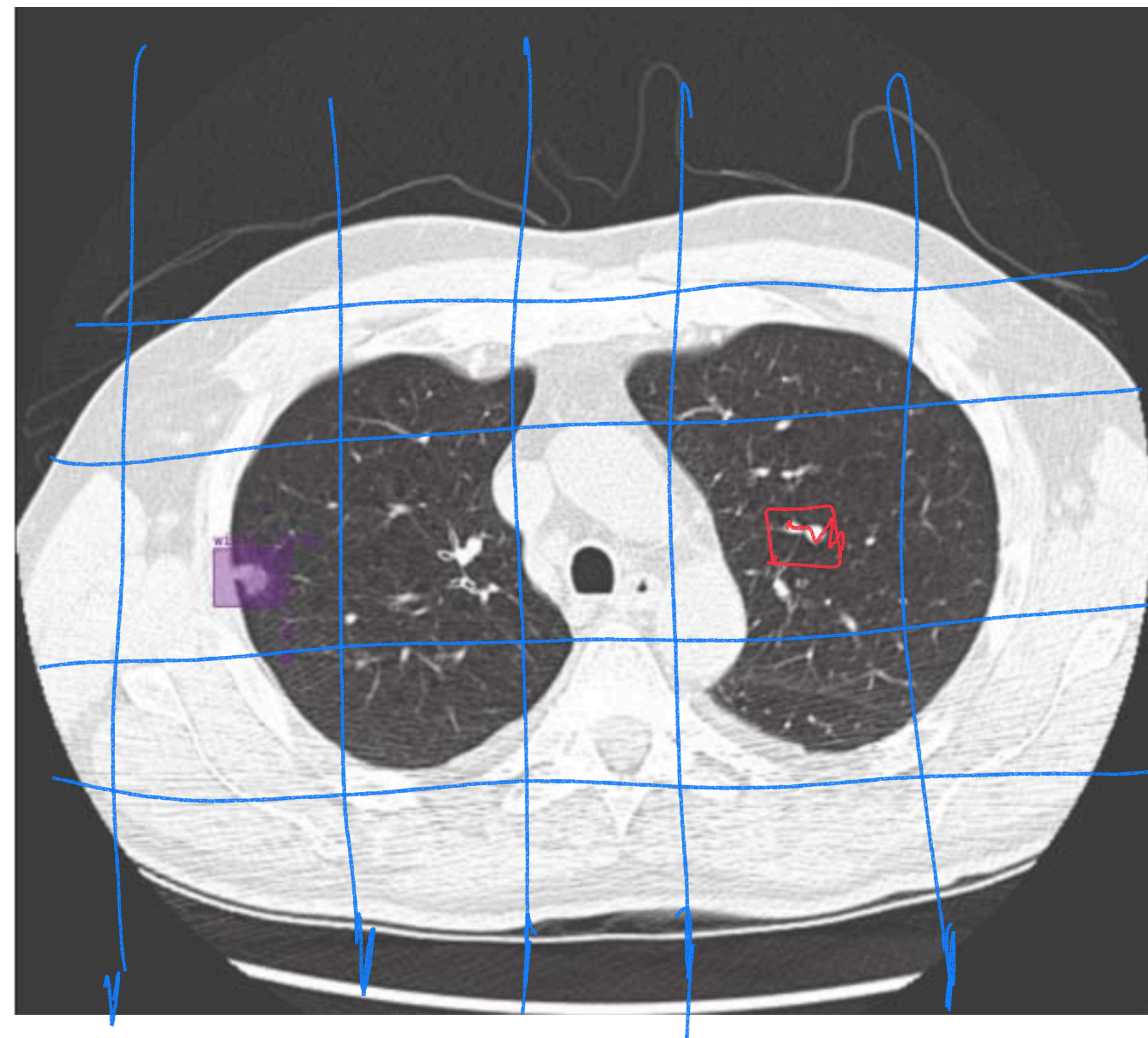
$$(\underline{c_x}, \underline{c_y}, \underline{c_z}) = \text{regress}(Z)$$

$$(\underline{w}, \underline{h}, \underline{d}) = \text{regress}(Z)$$

How
can
we
support
multiple
objects?

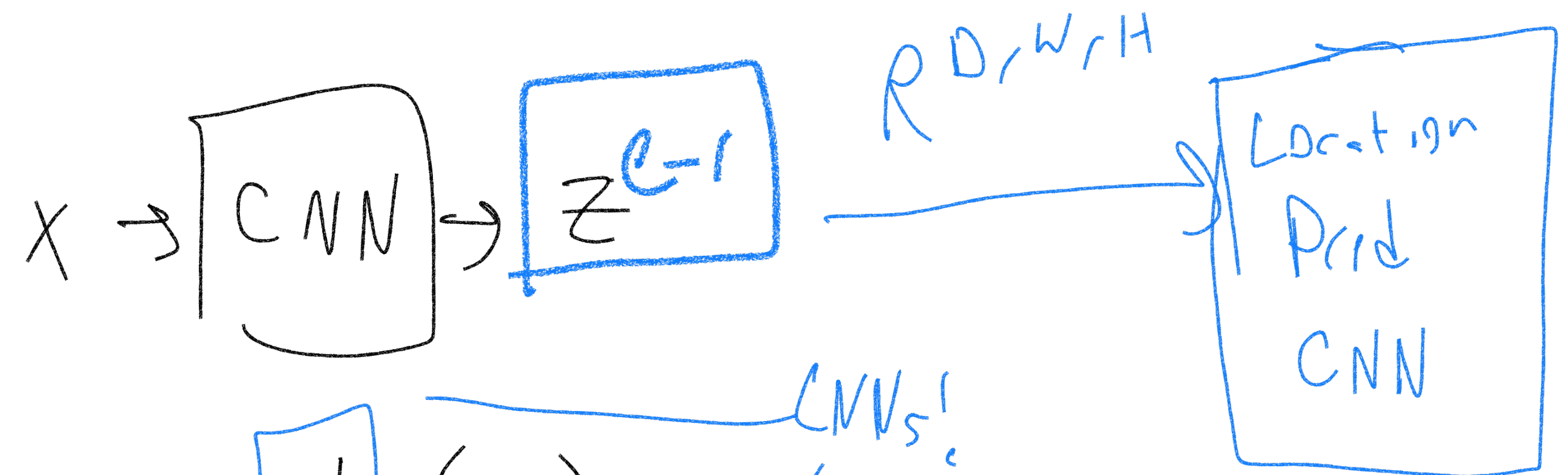
$$L = L_p + p L_c + p L_{\text{size}} \leftarrow \text{MSE}$$

Supporting multiple objects



relative offset!

$$b_i = (p, [c_x, c_y, c_z], w, h, d)$$



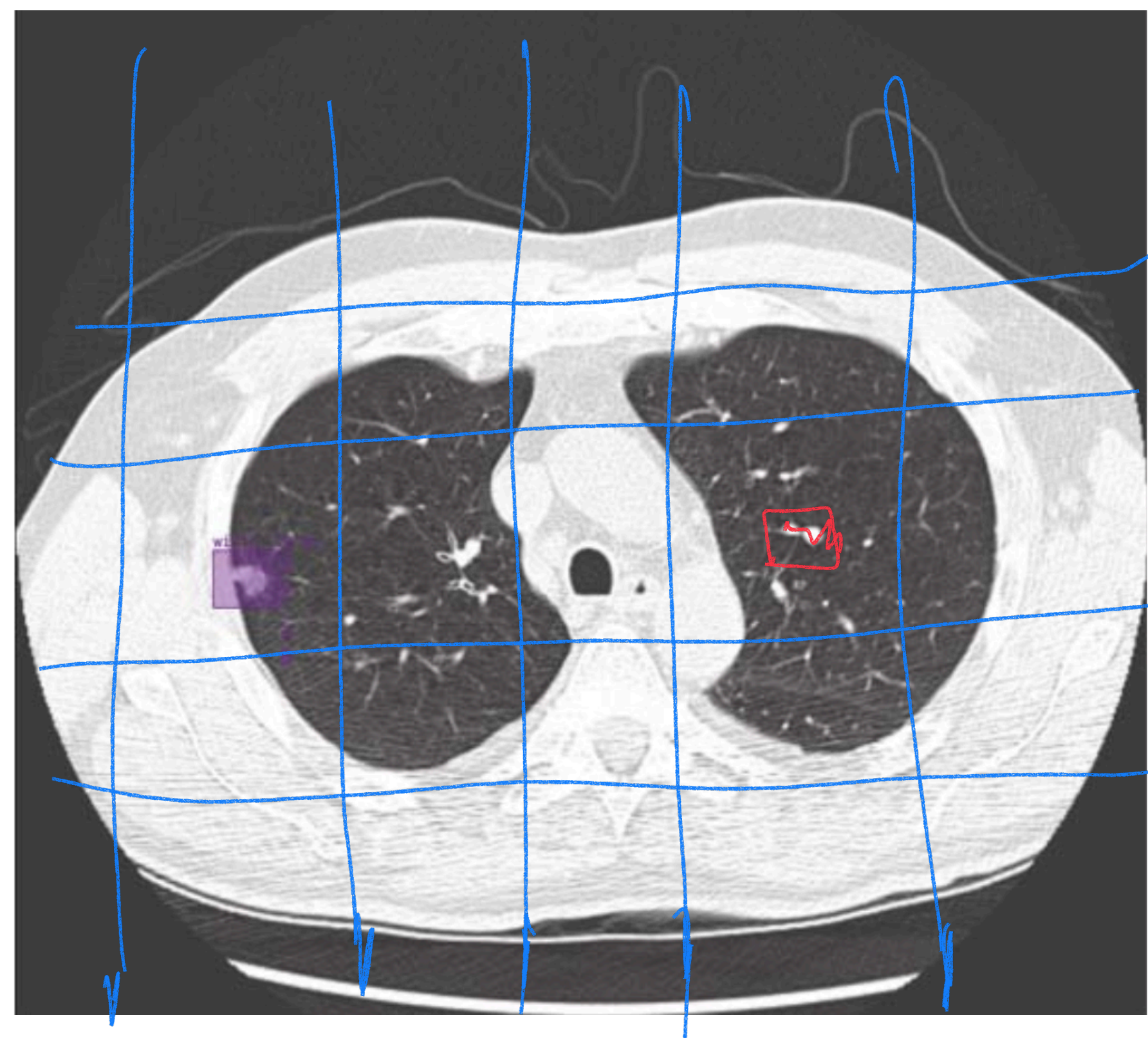
$$p = \text{cls}(z)$$

CNNs!

$$(c_x, c_y, c_z) = \text{regress}(z)$$

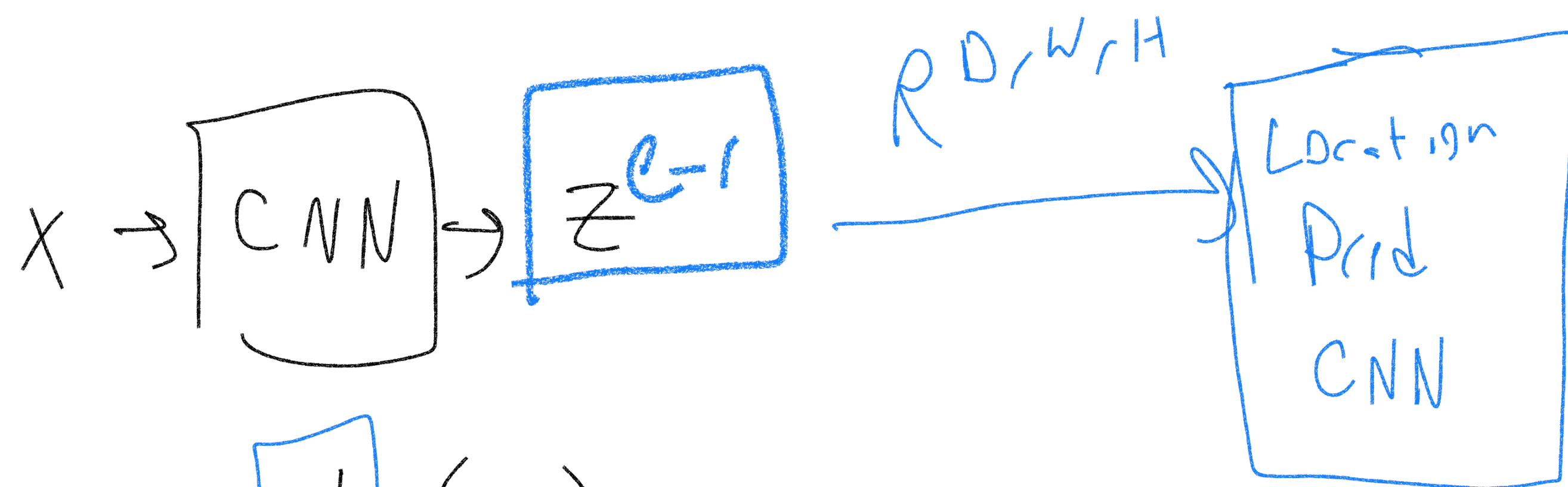
$$(w, h, d) = \text{regress}(z)$$

Supporting multiple objects



relative offset!

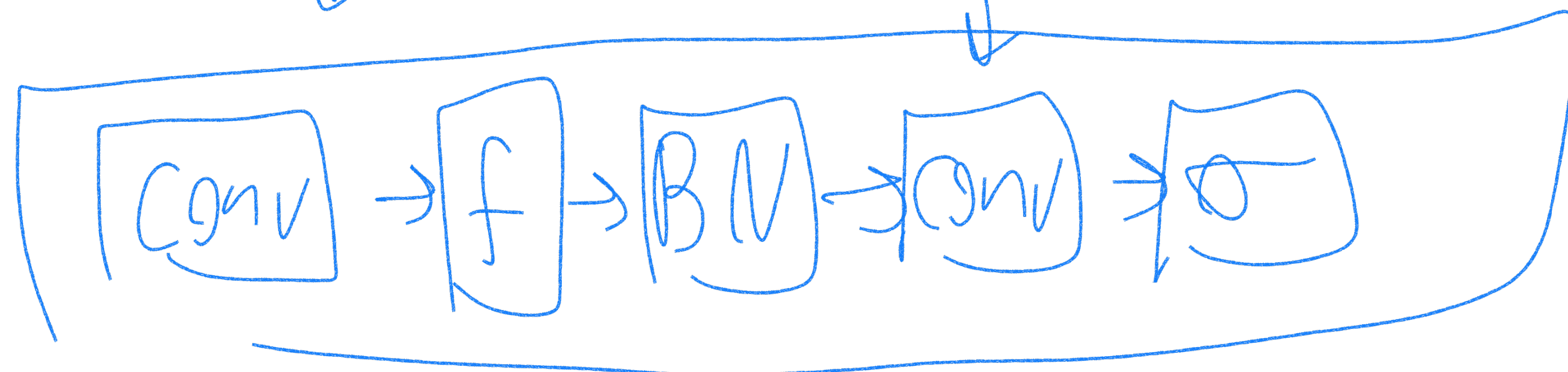
$$b_i = (p, \boxed{c_x, c_y, c_z}, w, h, d)$$



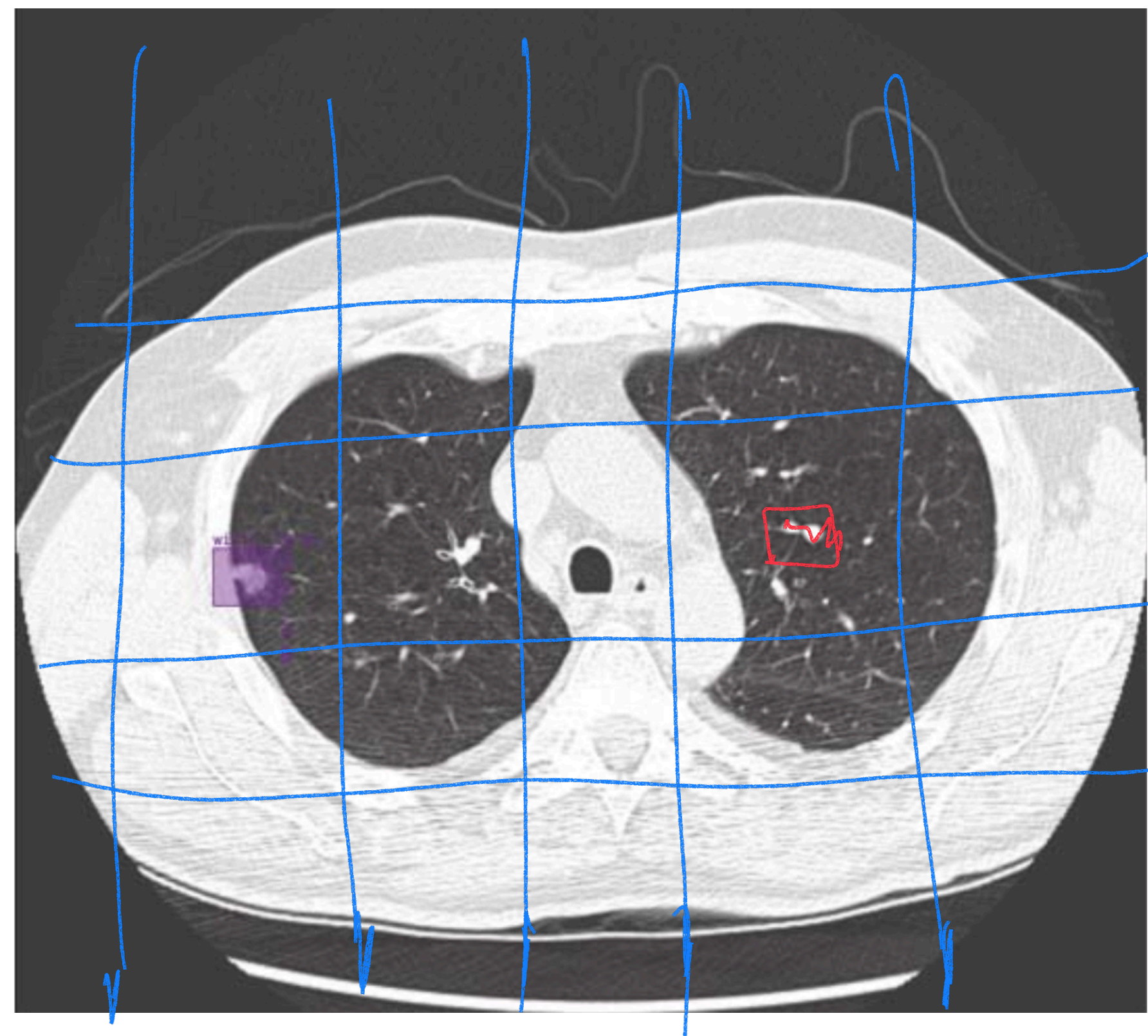
$$p = \boxed{\text{cls}}(z)$$

$1 \times 1 \times 1 \downarrow$

$1 \times 1 \times 1 \downarrow$

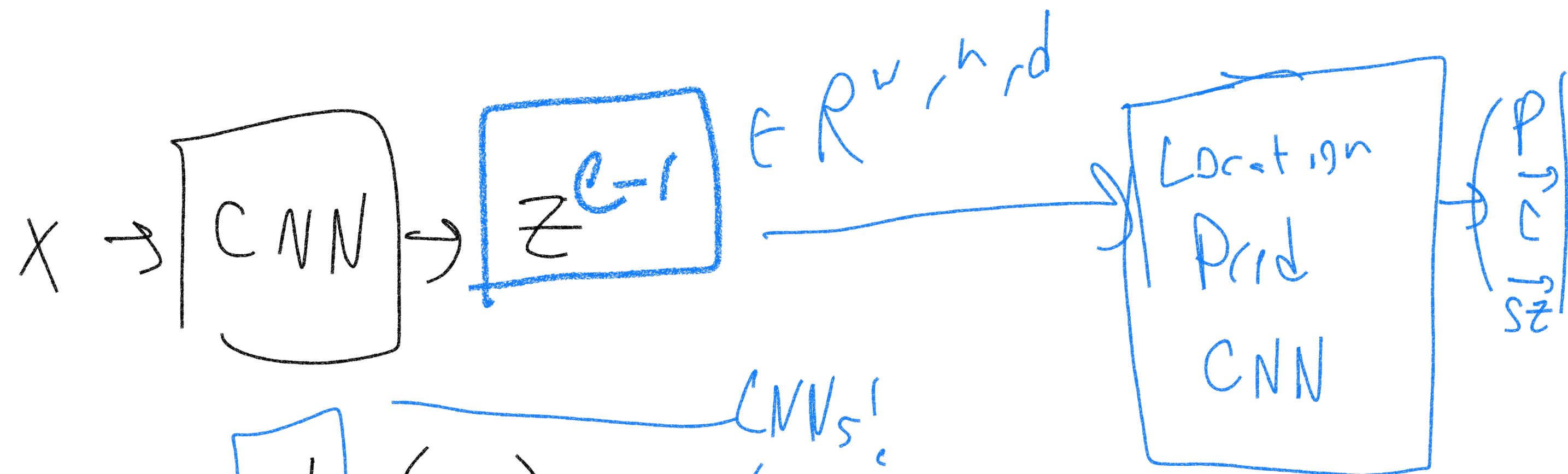


Supporting multiple objects



relative offset!

$$b_i = (p, [c_x, c_y, c_z], w, h, d)$$



$$p = \text{cls}(z) \quad \leftarrow \text{CNNs!}$$

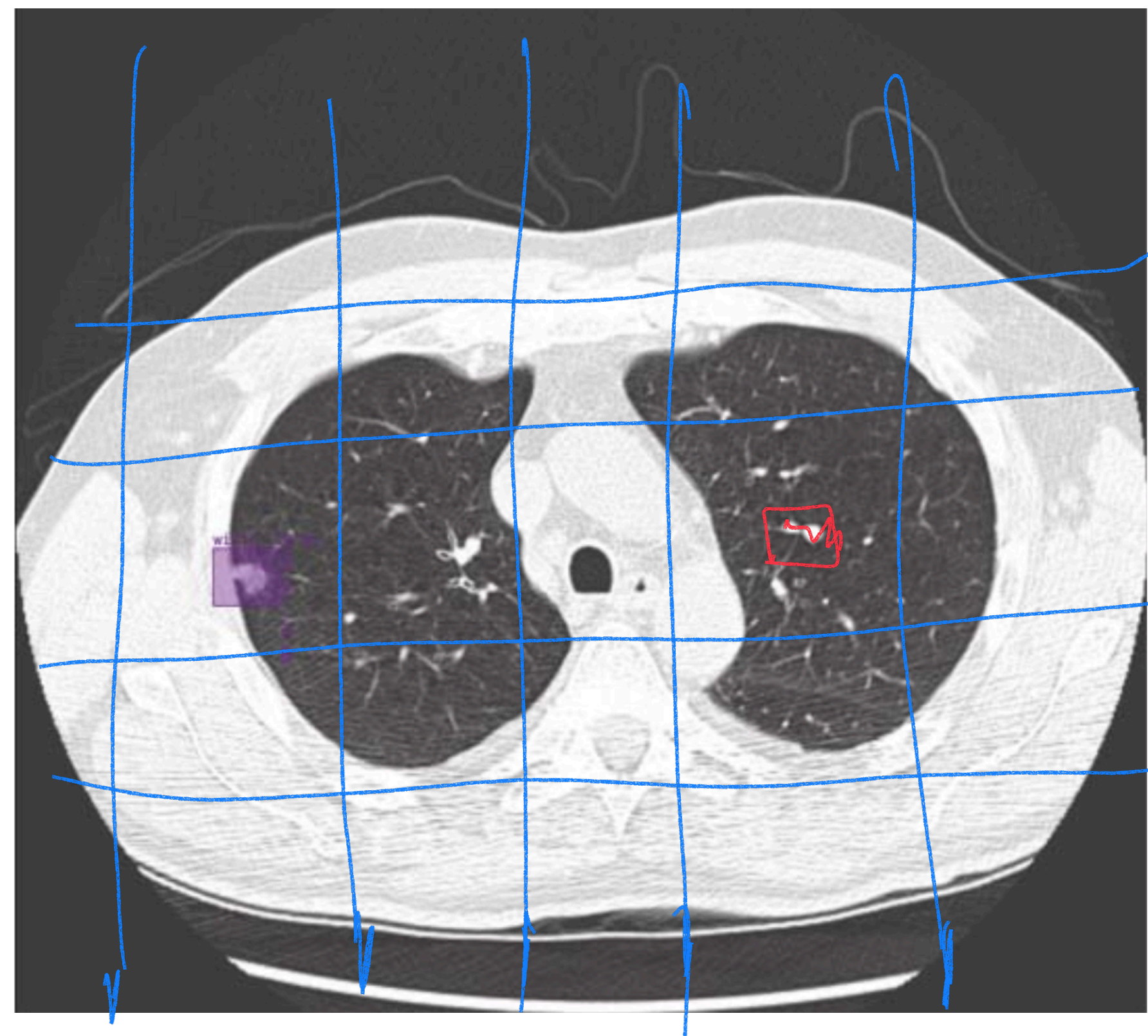
$$(\underline{c_x}, \underline{c_y}, \underline{c_z}) = \text{regress}(z)$$

$$(\underline{w}, \underline{h}, \underline{d}) = \text{regress}(z)$$

How many preds?

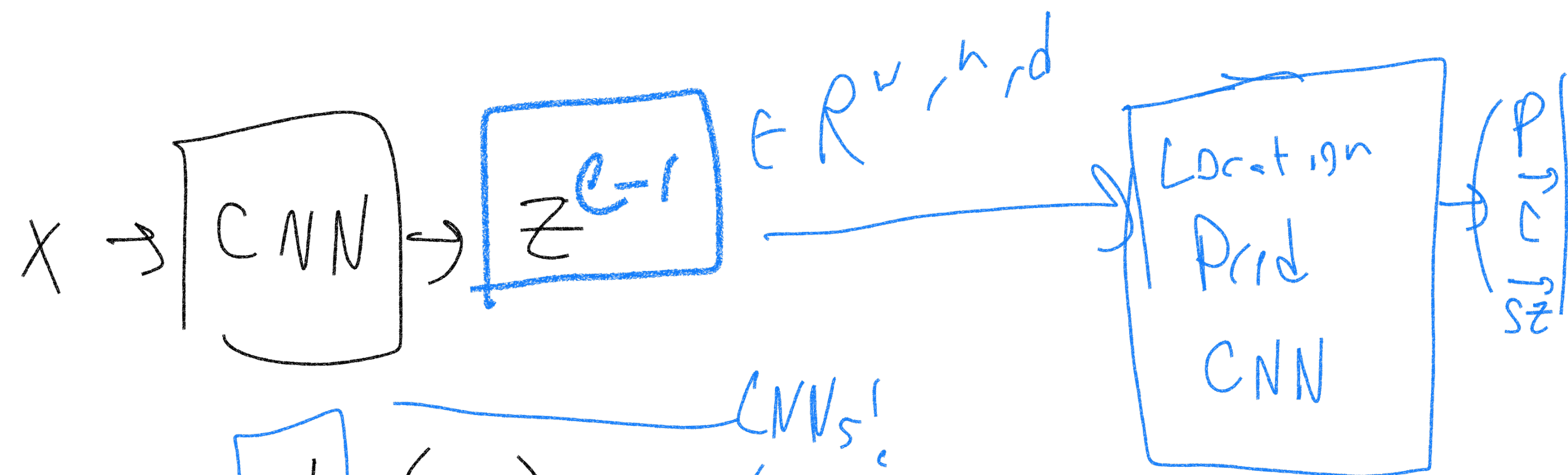
$$L = L_p + p L_c + p L_{size} \quad \leftarrow \text{MSE}$$

Supporting multiple objects



relative offset!

$$b_i = (p, [c_x, c_y, c_z], w, h, d)$$



Classification:

$$p = \text{cls}(z)$$

← CNNs!

Coordinate regression:

$$(c_x, c_y, c_z) = \text{regress}(z)$$

Size regression:

$$(w, h, d) = \text{regress}(z)$$

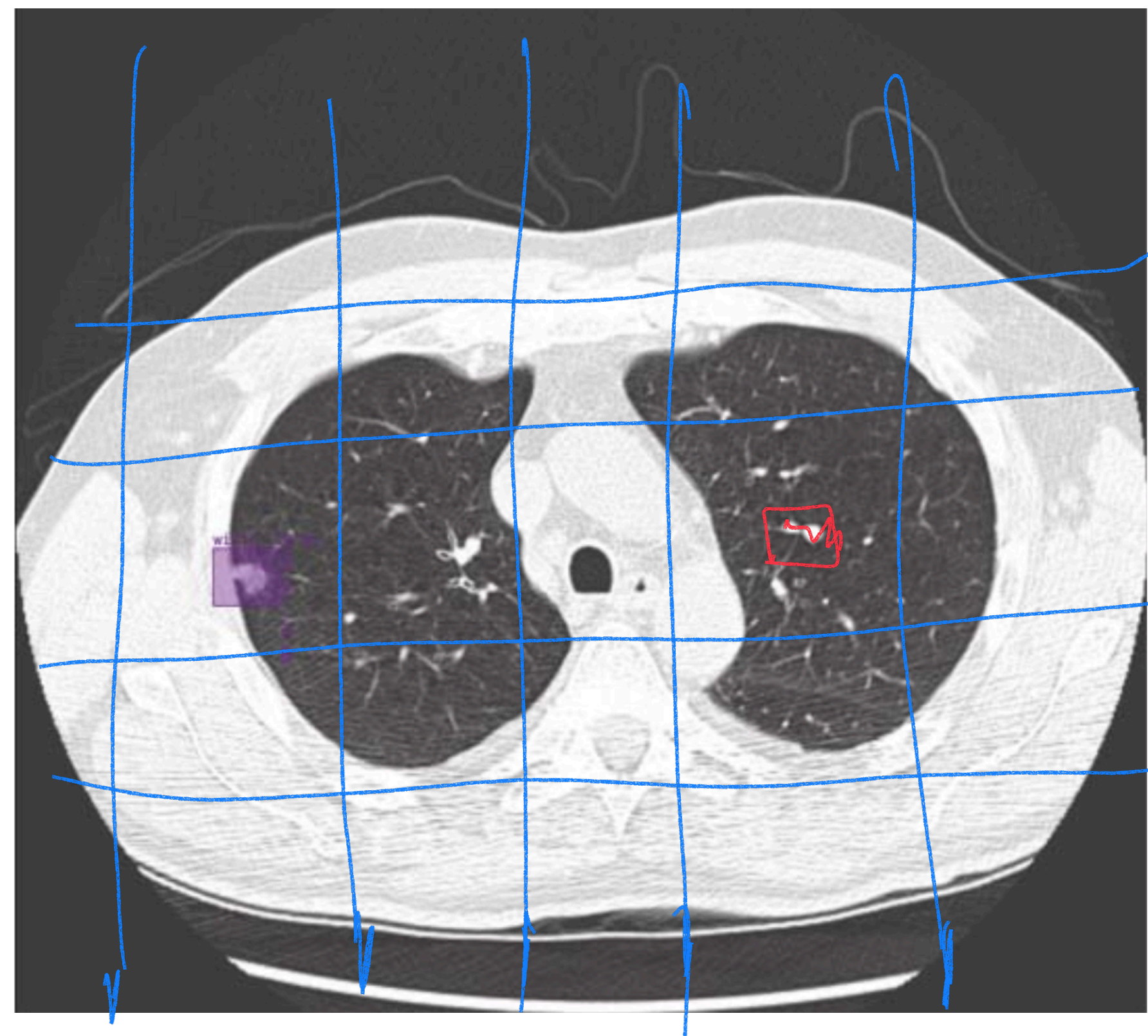
Loss function:

$$L = L_p + p L_c + p L_{\text{size}} \leftarrow \text{MSE}$$

How many preds?

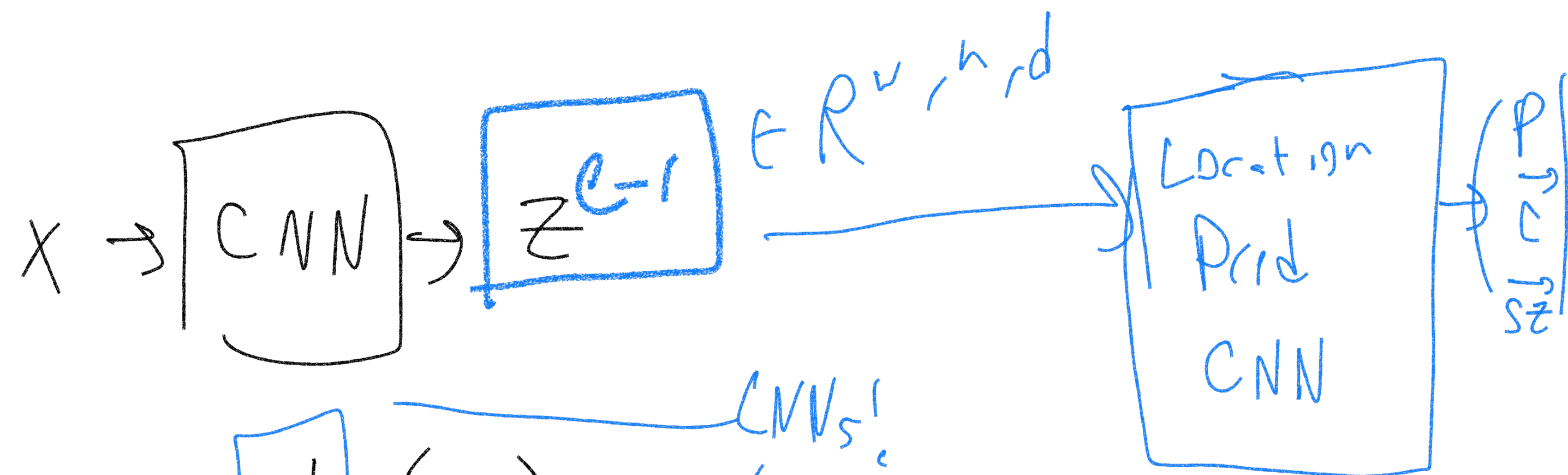
$(whd) \cdot 7$

Supporting multiple objects



relative offset!

$$b_i = (p, [c_x, c_y, c_z], w, h, d)$$



$p = \text{cls}(z)$ (CNNs!)

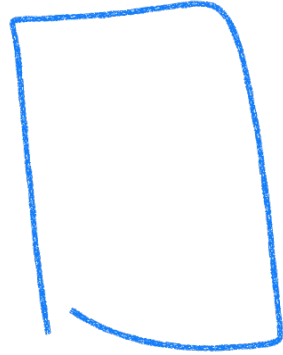
$(\underline{c_x}, \underline{c_y}, \underline{c_z}) = \text{regress}(z)$

$(\underline{w}, \underline{h}, \underline{d}) = \text{regress}(z)$

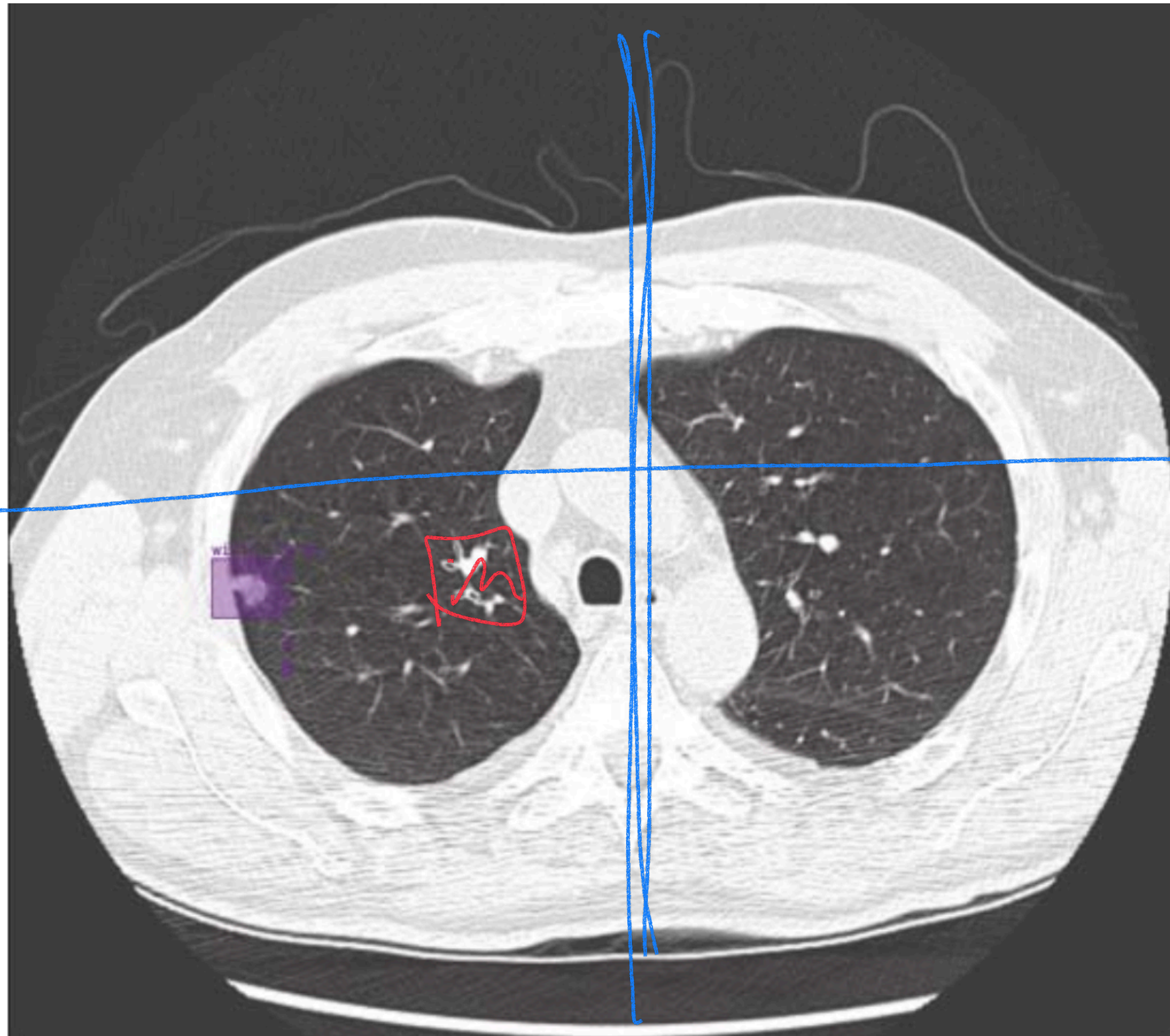
$$L = L_p + p L_c + p L_{\text{size}} \leftarrow \text{MSE}$$

Will this work?
Assumptions?

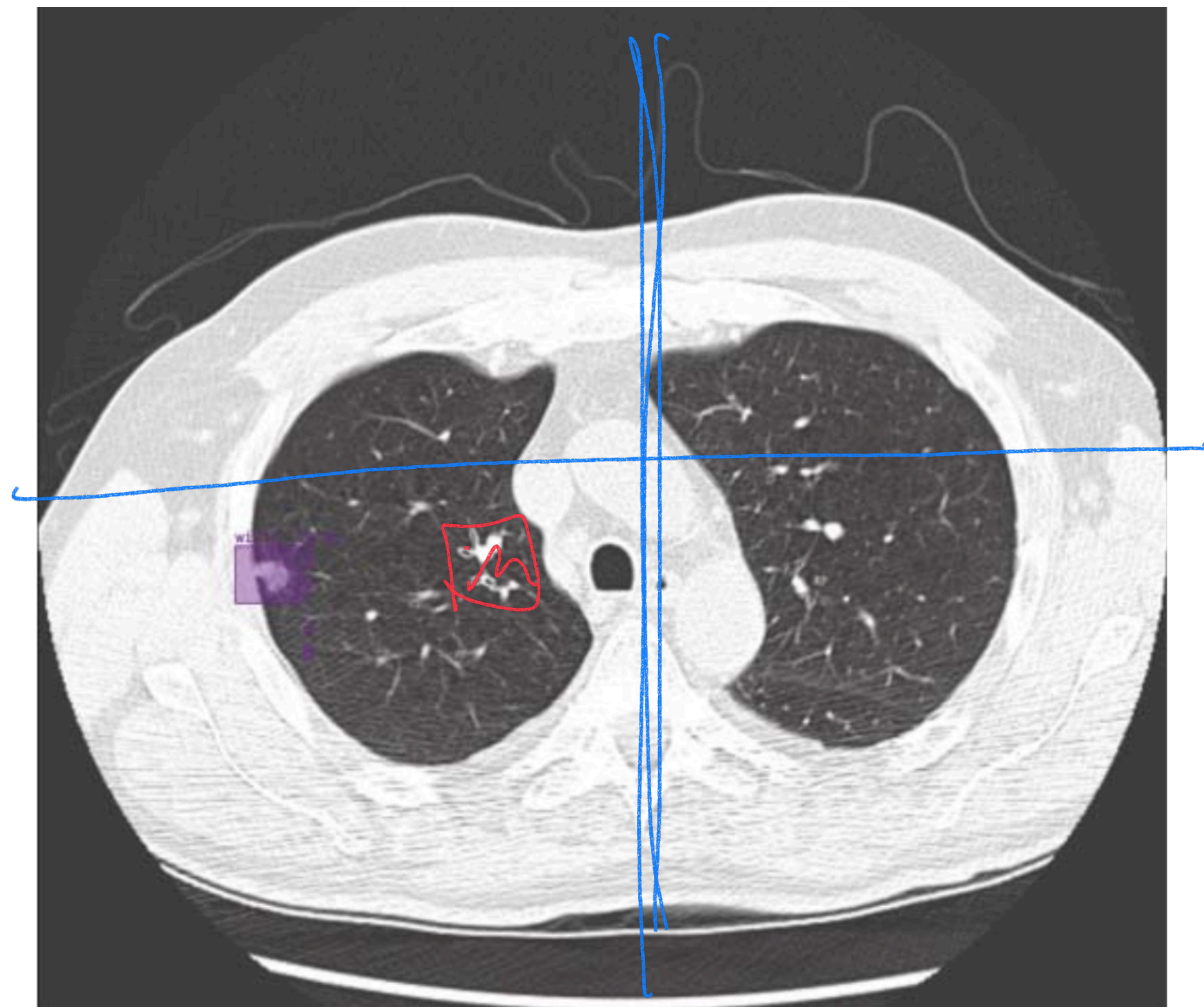
Supporting multiple objects

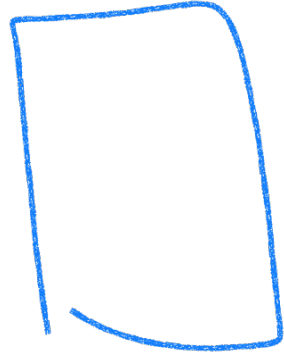
 \neq voxel size

What if voxel too big?

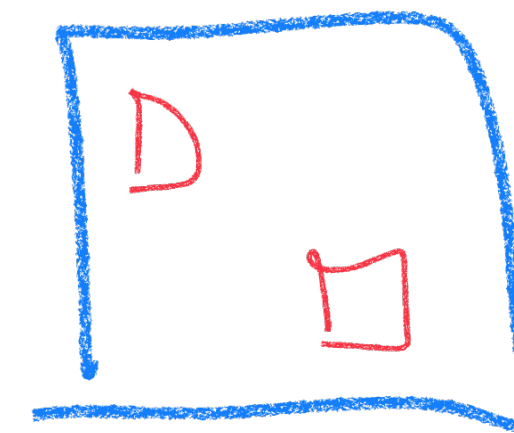


Supporting multiple objects



 \neq voxel size

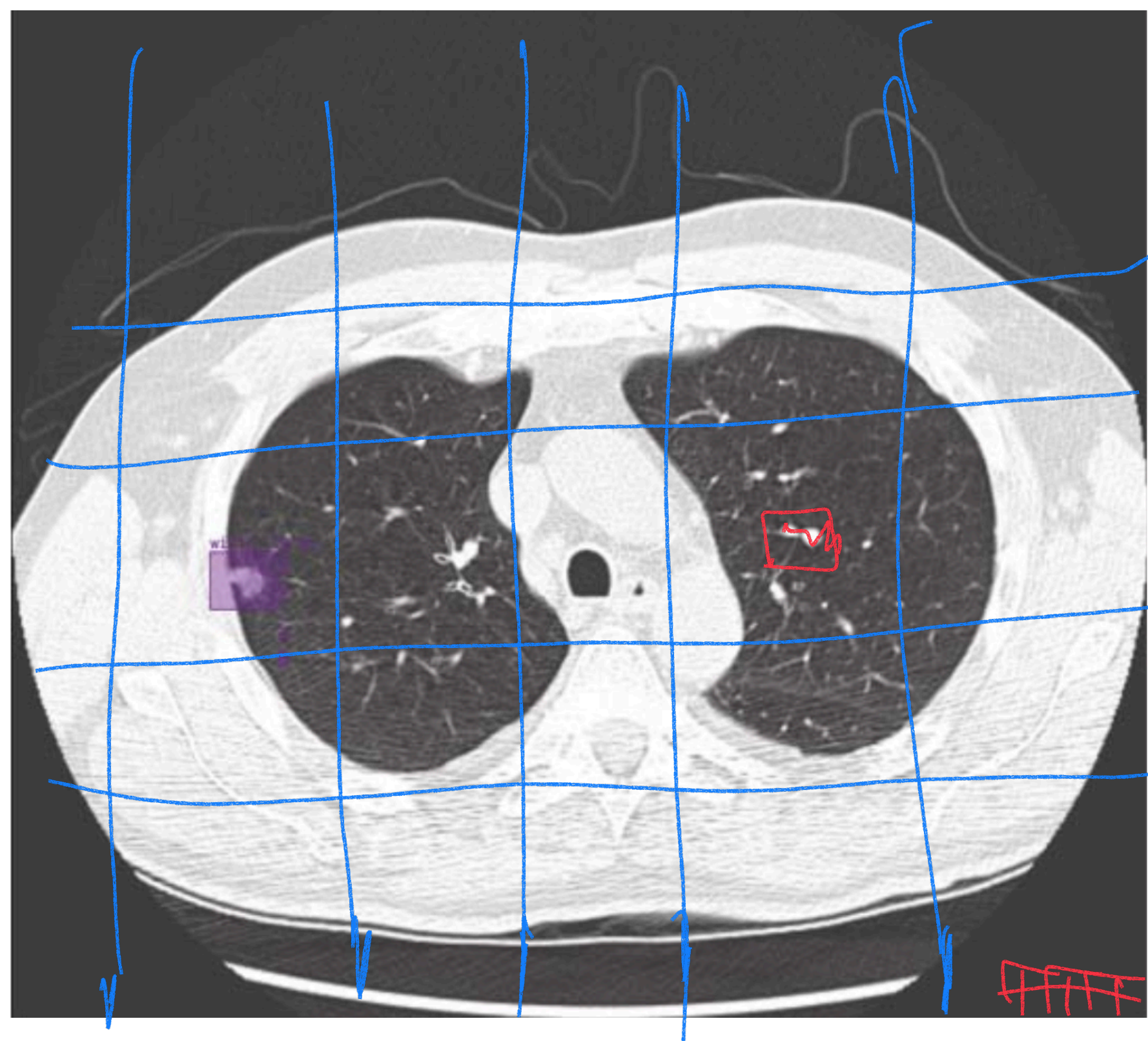
What if voxel too big? Bad!

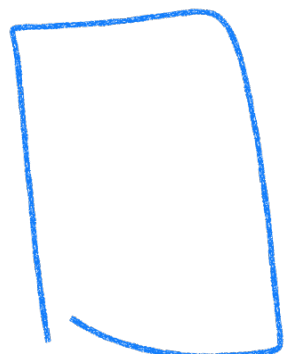


← can't separate out objects!

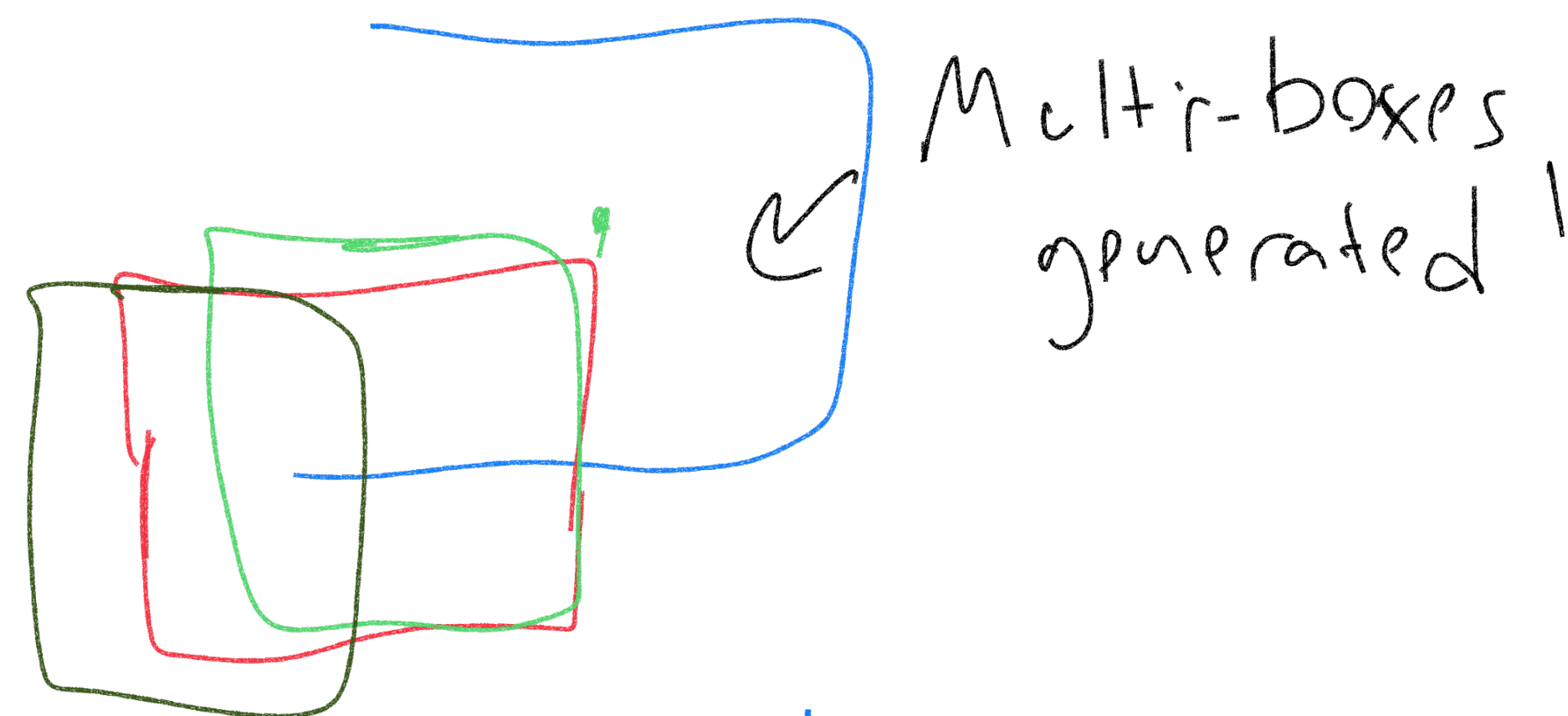
Equivalent to Baseline ^{oo}_o

Supporting multiple objects

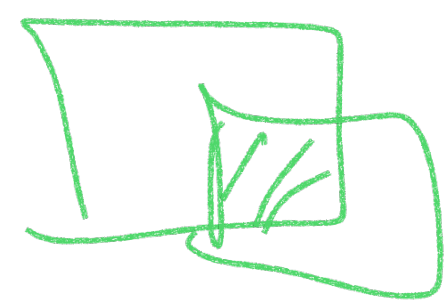



 \approx voxel size

Voxel too small?

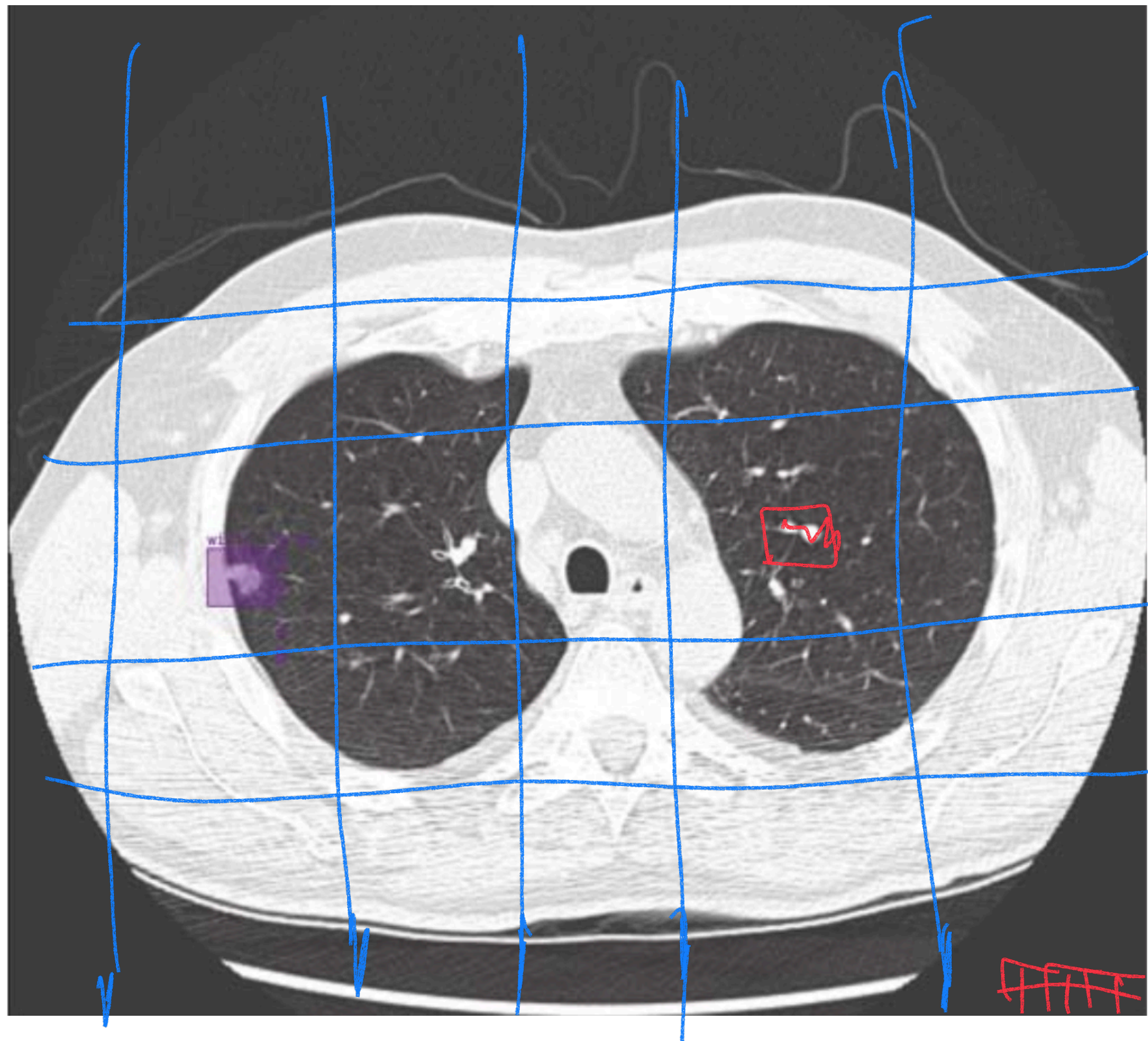


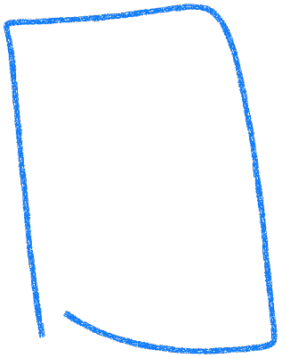
Solution: Merge boxes heuristically



IOU, Non Max Suppression 

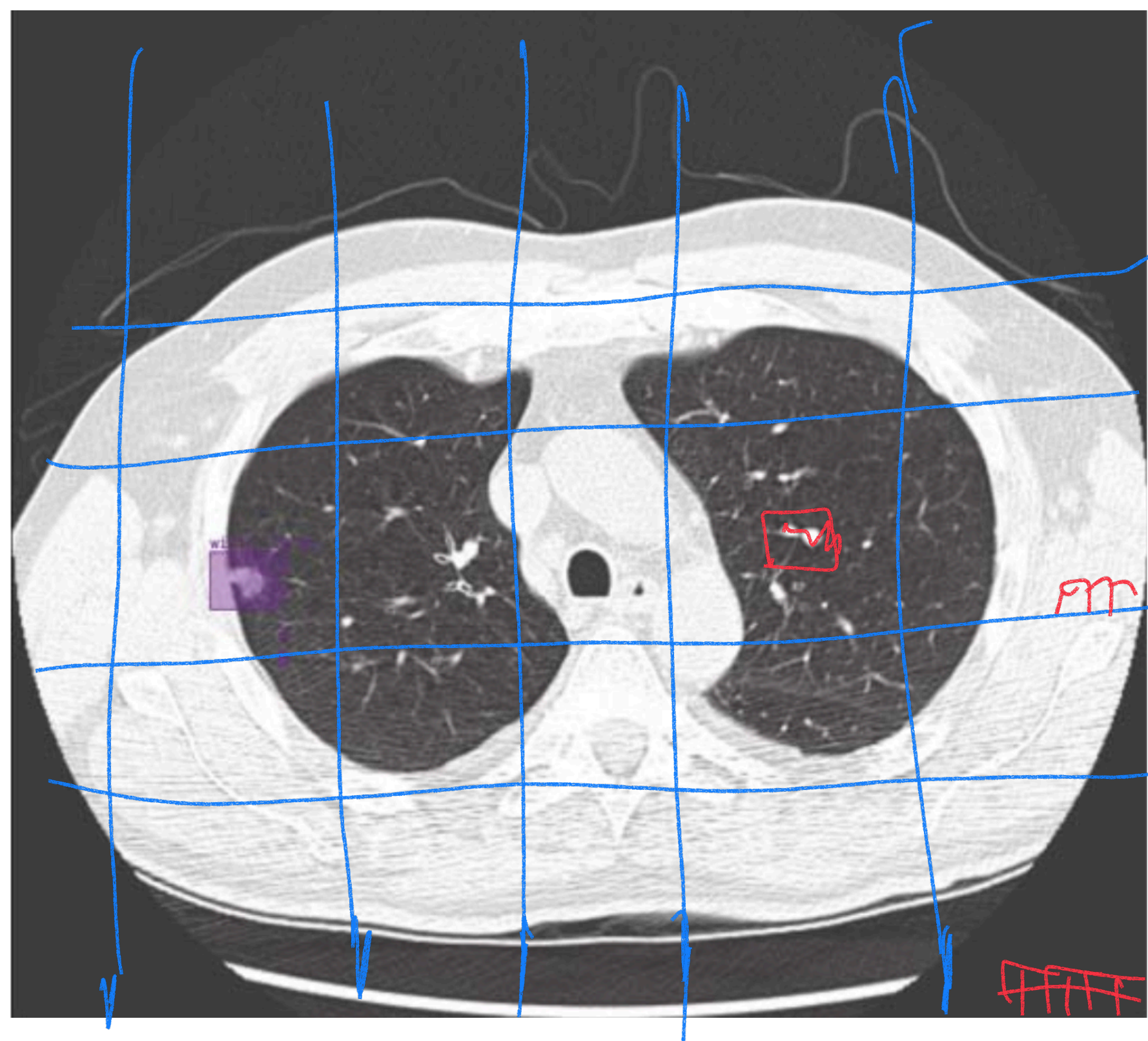
Supporting multiple objects



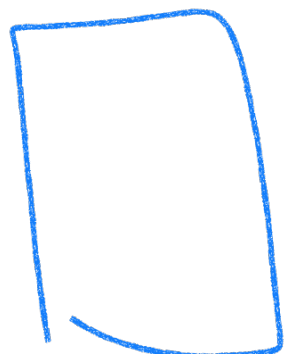
 \approx voxel size

Voxel too small?

Supporting multiple objects



scales!

 \approx voxel size

Voxel too small?

Not enough context!

\Rightarrow All black!

Use features for multiple

$$Z = z_{\text{coarse}} + z_{\text{mid}} + z_{\text{fine}}$$

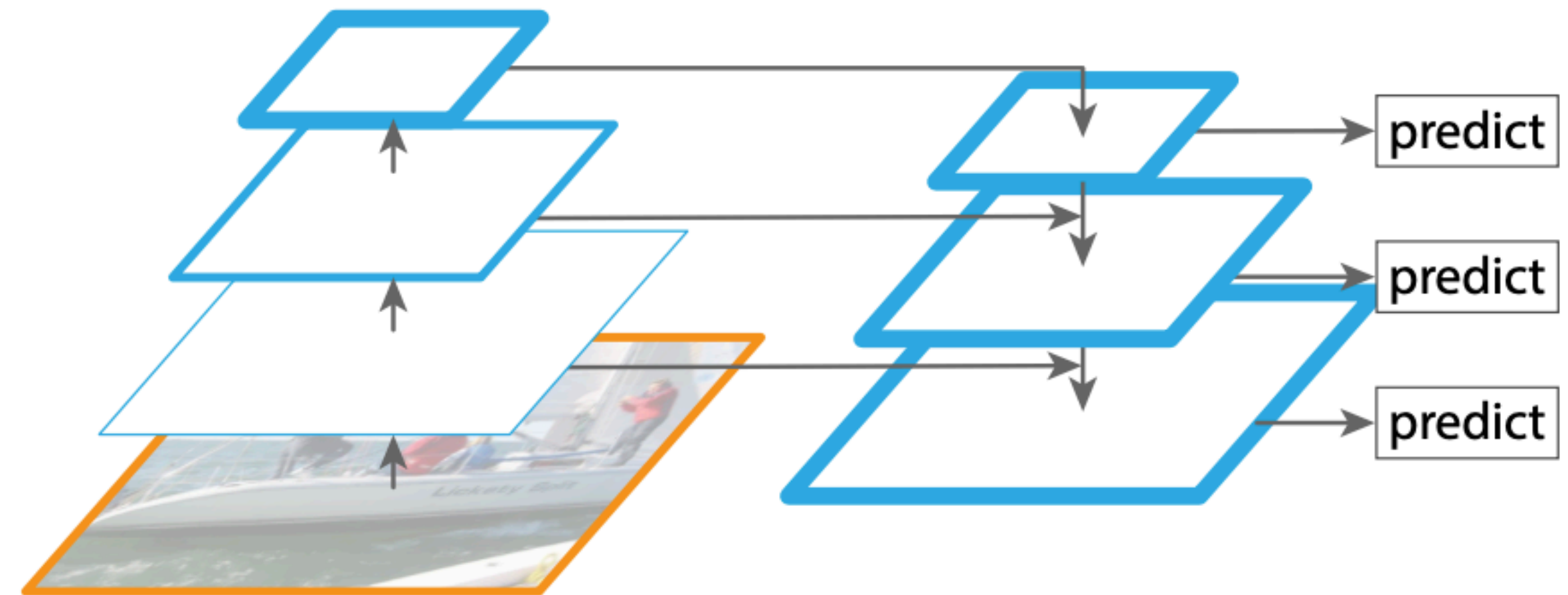
Feature pyramids networks

Feature Pyramid Networks for Object Detection

Tsung-Yi Lin^{1,2}, Piotr Dollár¹, Ross Girshick¹,
Kaiming He¹, Bharath Hariharan¹, and Serge Belongie²

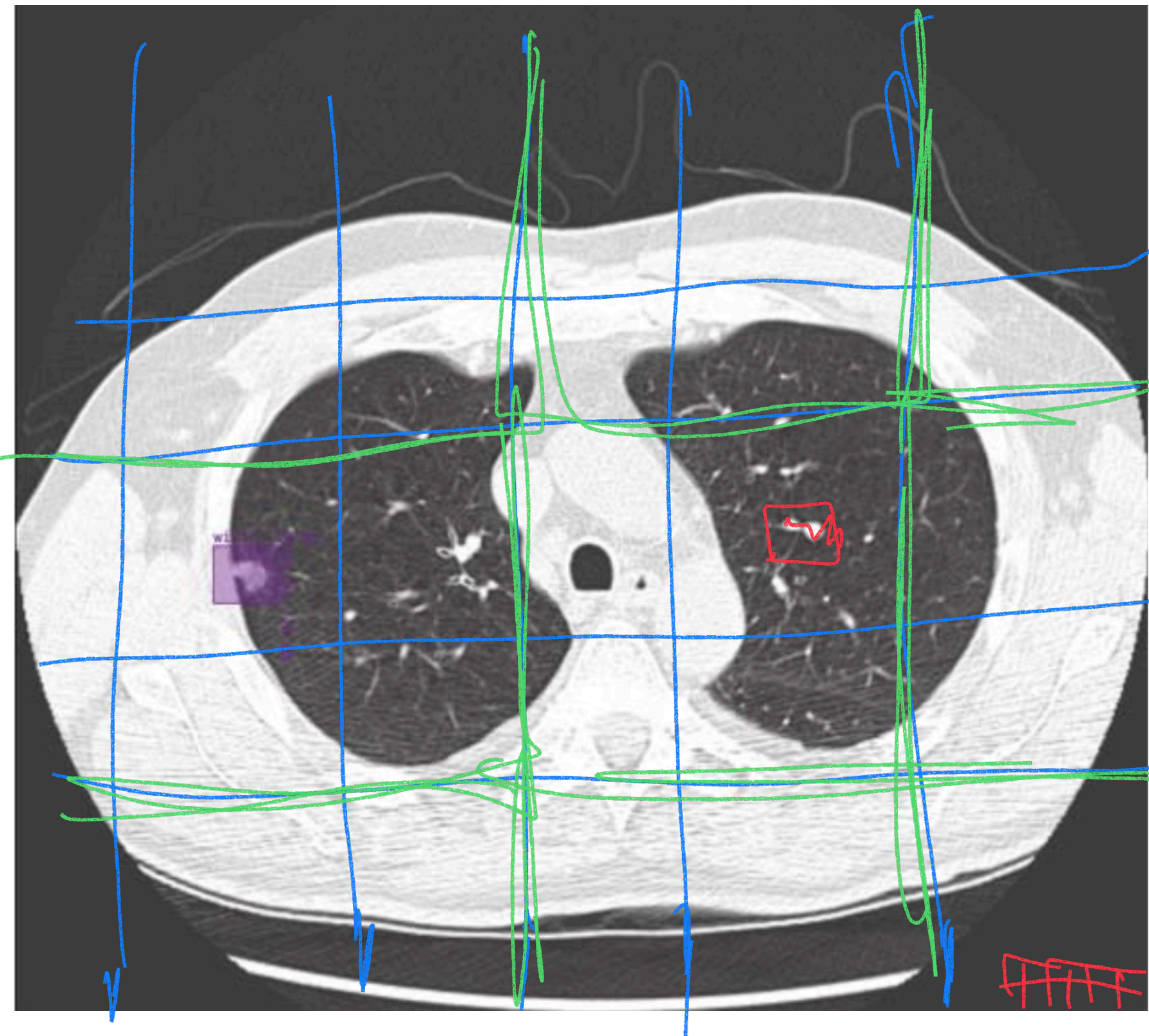
¹Facebook AI Research (FAIR)

²Cornell University and Cornell Tech



(d) Feature Pyramid Network

Supporting multiple objects and scales



Voxel too small?

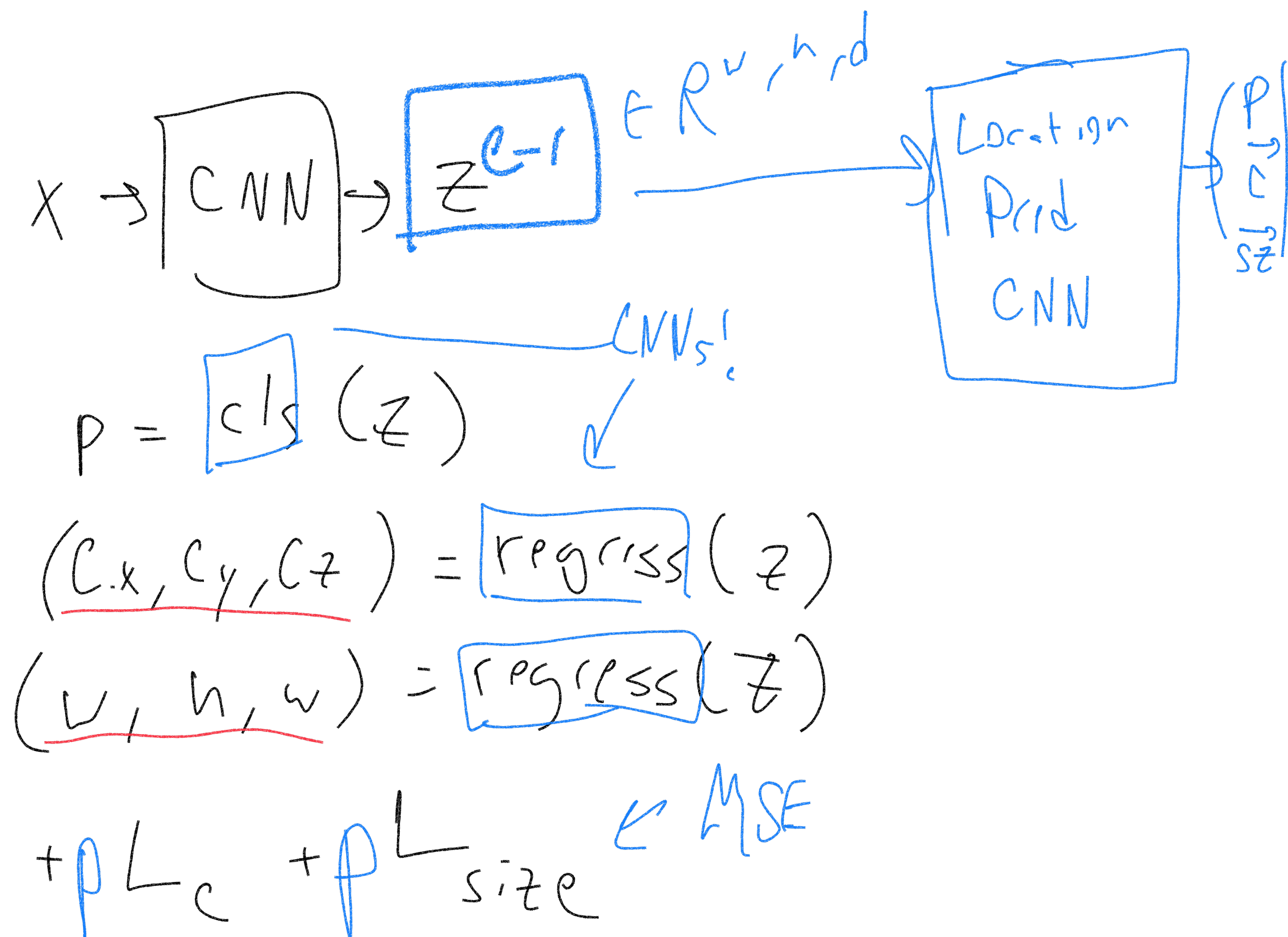
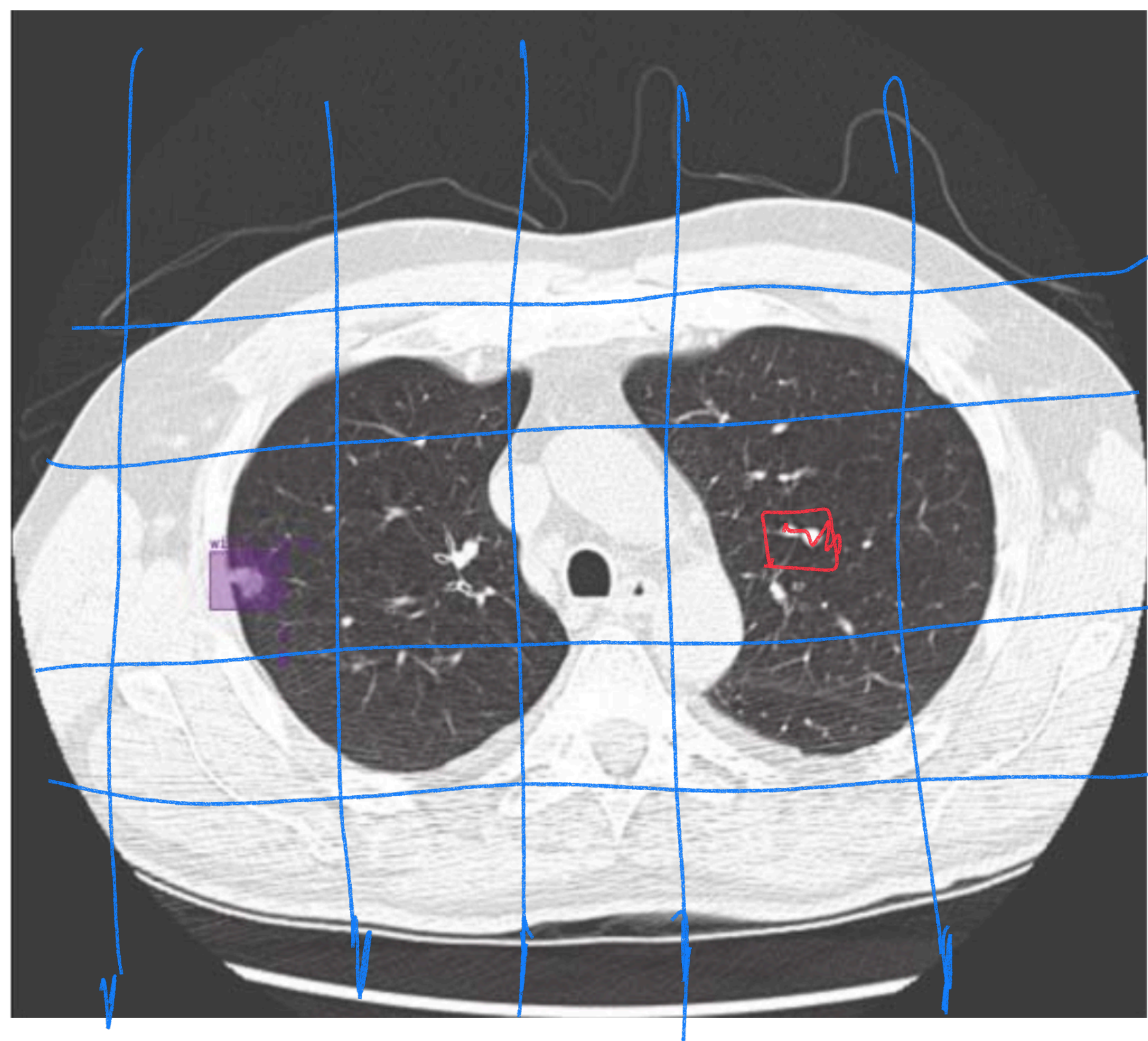
↳ Consider k scales!

Leads to

$(k) \cdot (width) \cdot 7$

predictions

Putting it all together



Classic papers in this area

Multi-stage approaches:

R-CNN

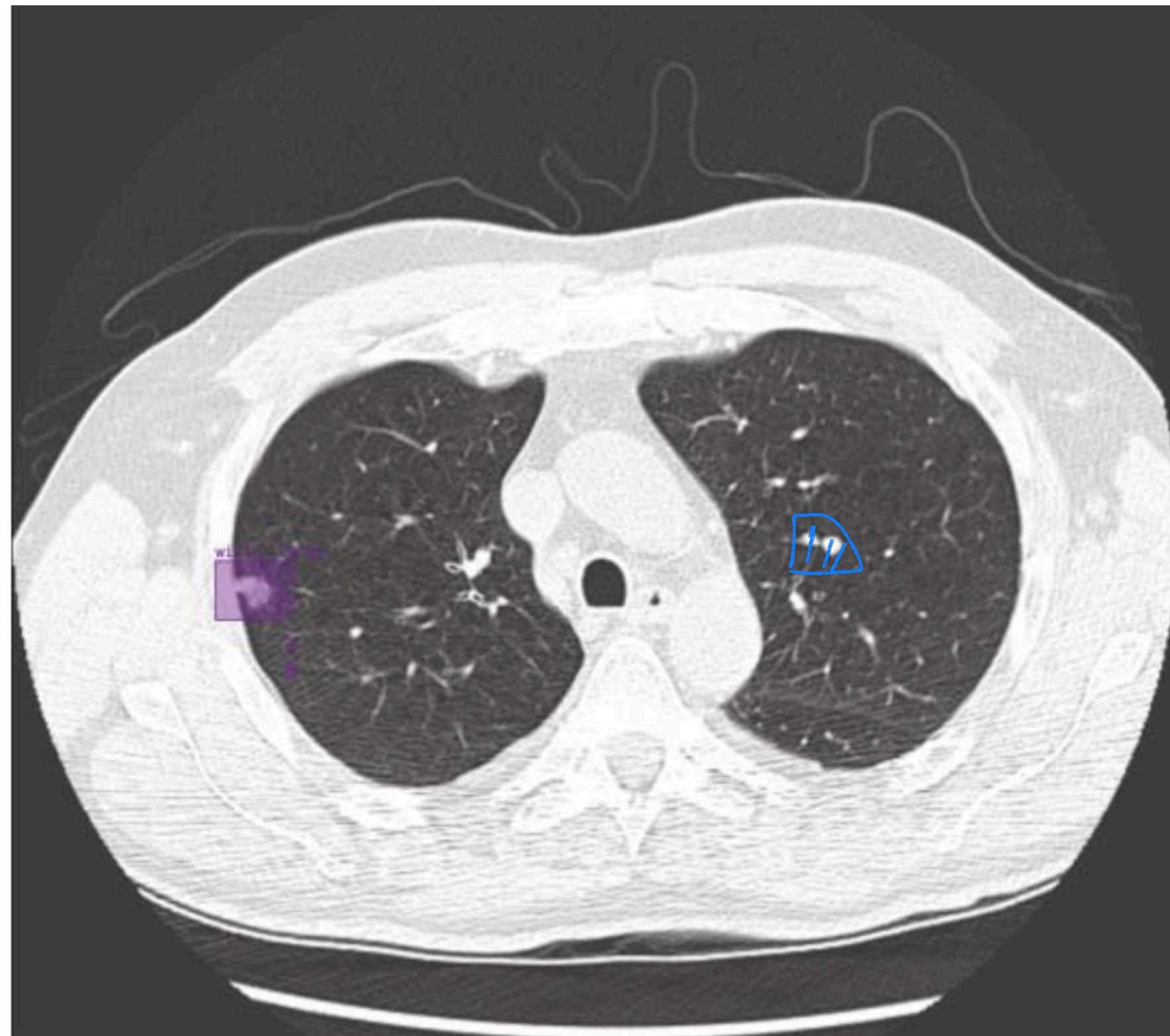
Fast R-CNN

Faster R-CNN

Single-stage approaches:

YOLO

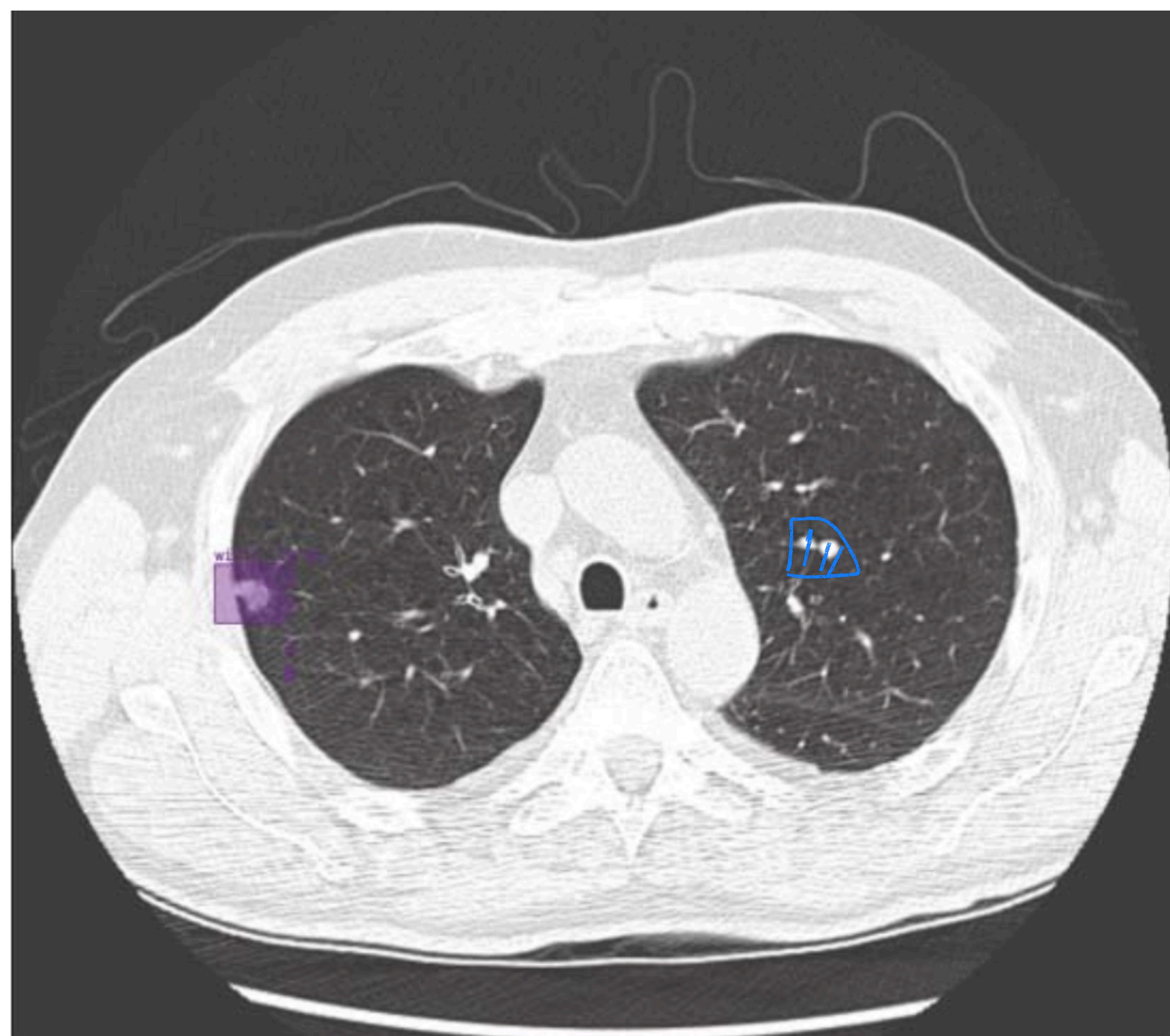
What is this missing?



Radiologist drew bounding boxes
for each cancer

$$b_i = (p, c_x, c_y, c_z, w, h, d)$$

What is this missing?



Radiologist drew bounding boxes
for each cancer

$$b_i = (p, c_x, c_y, c_z, w, h, d)$$

↳ Can't measure tumor
volume!

↳ boxes for coarse

↳ Treatment monitor / Radiation Planning

Agenda

Recap

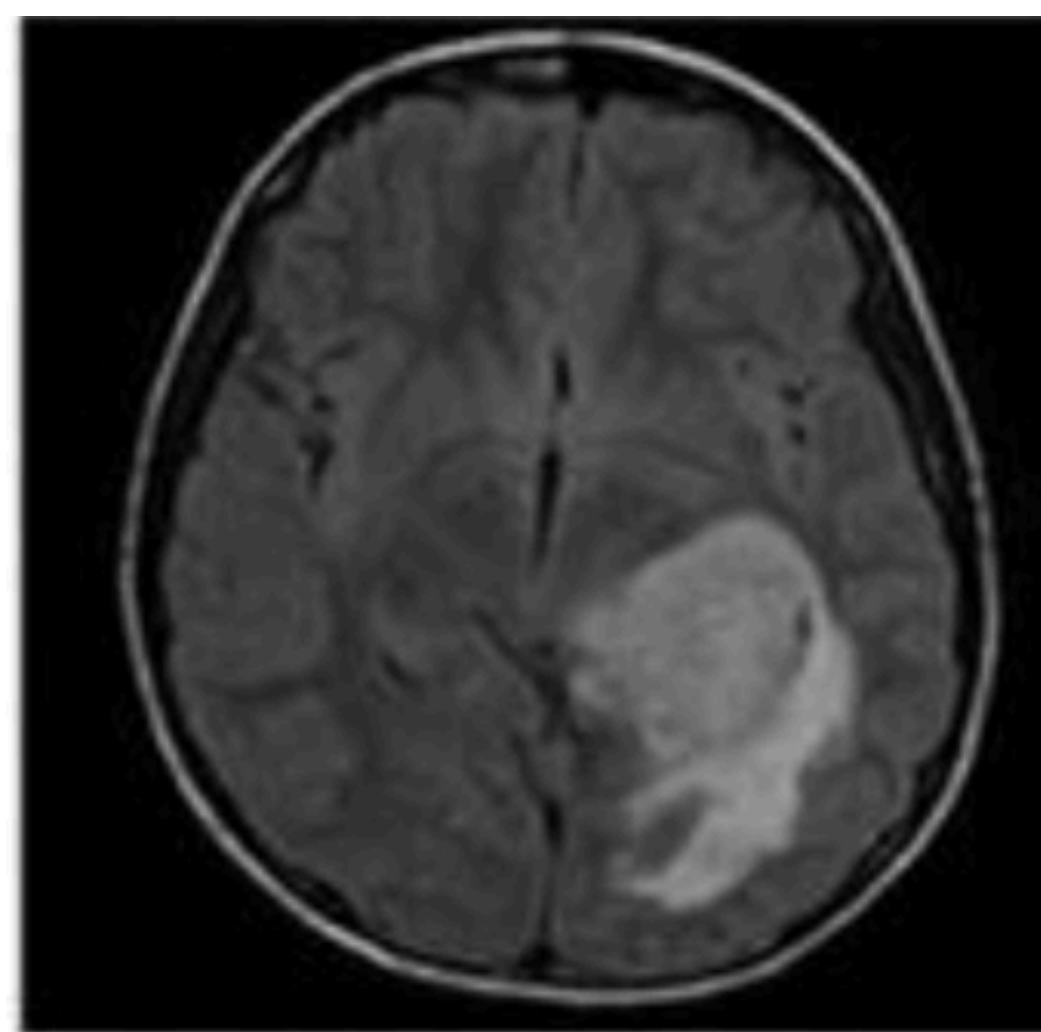
Motivation for Localization

Supervising Attention pooling

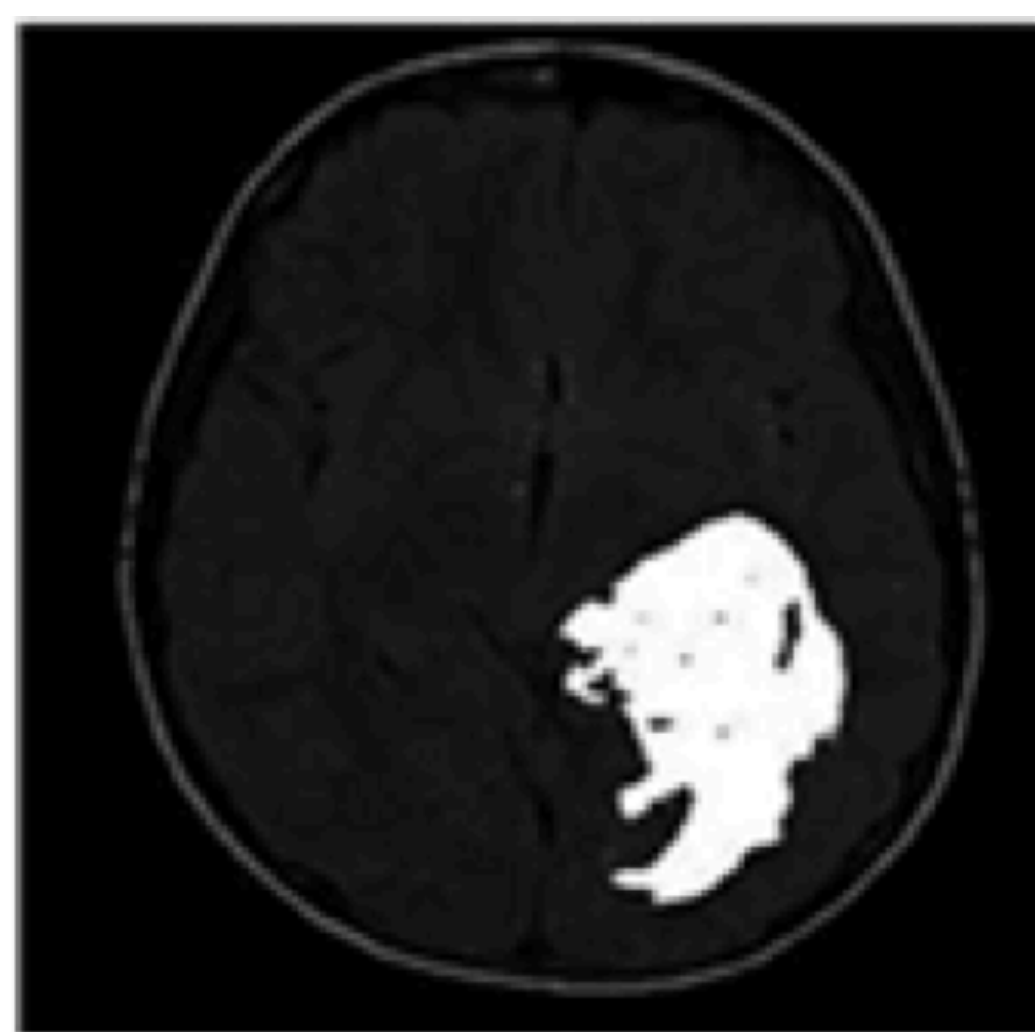
Bounding box prediction

Segmentation

Problem Setting



i



ii

Measure exact tumor
volume!

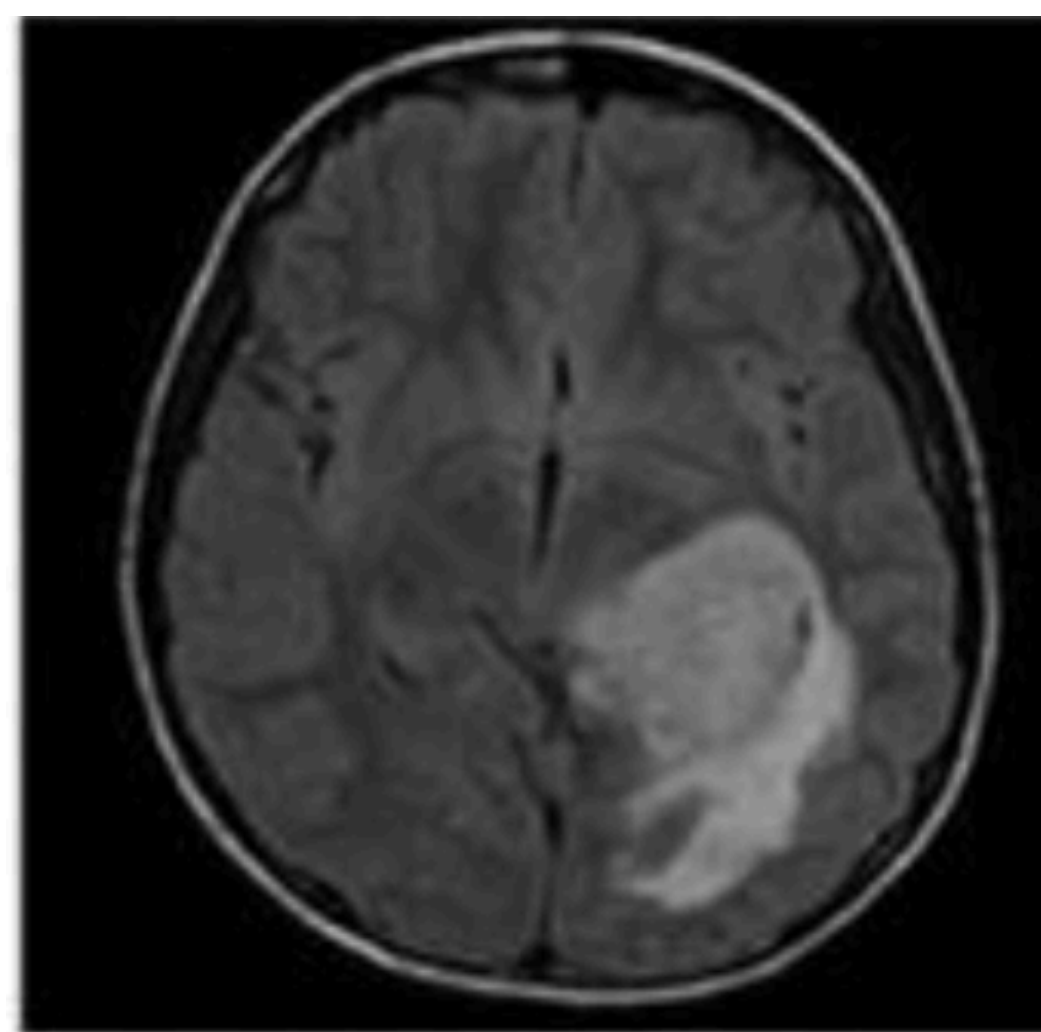
$$X \in \mathbb{R}^{H, W, D}$$

$$A \in \mathbb{R}^{H, W, D}$$

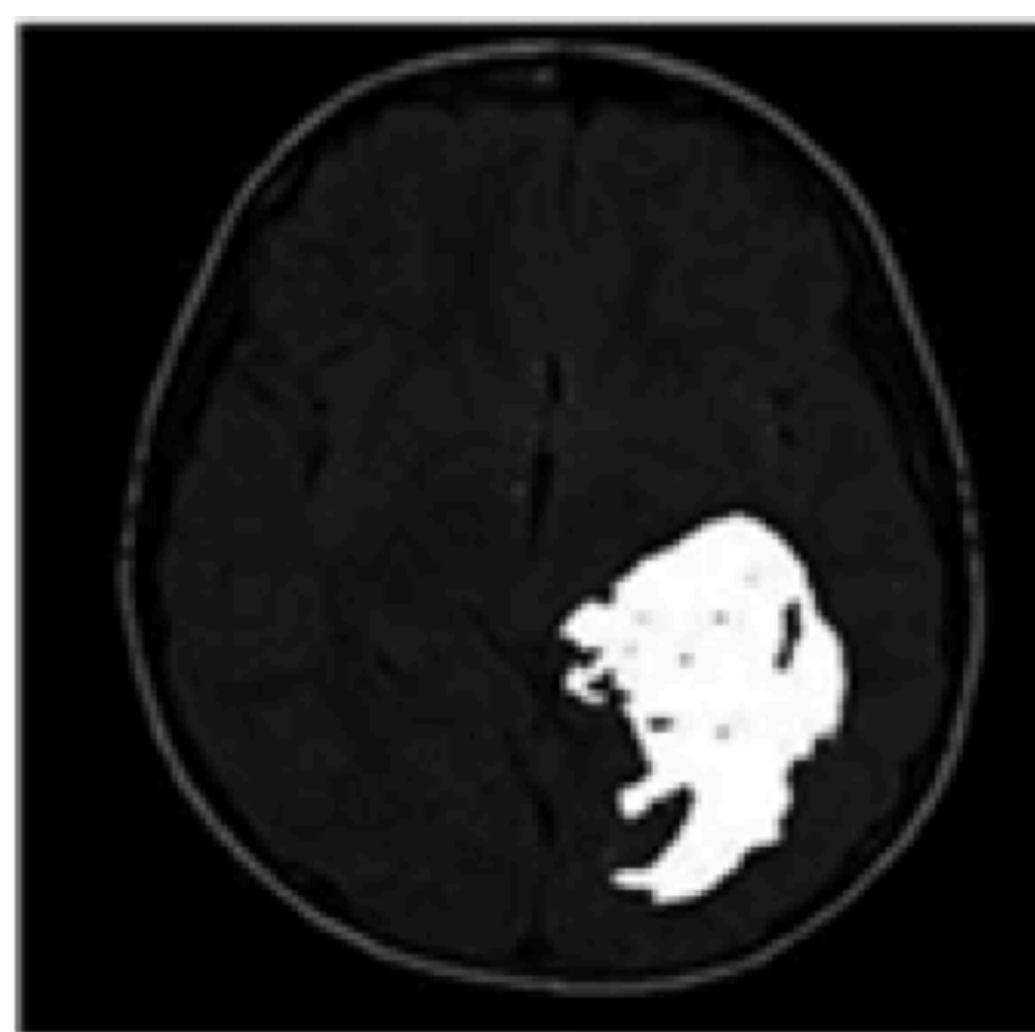
↑
Binary Mask!

Why not attention?

Problem Setting



i



ii

Measure exact tumor
volume!

$$X \in \mathbb{R}^{H, W, D}$$

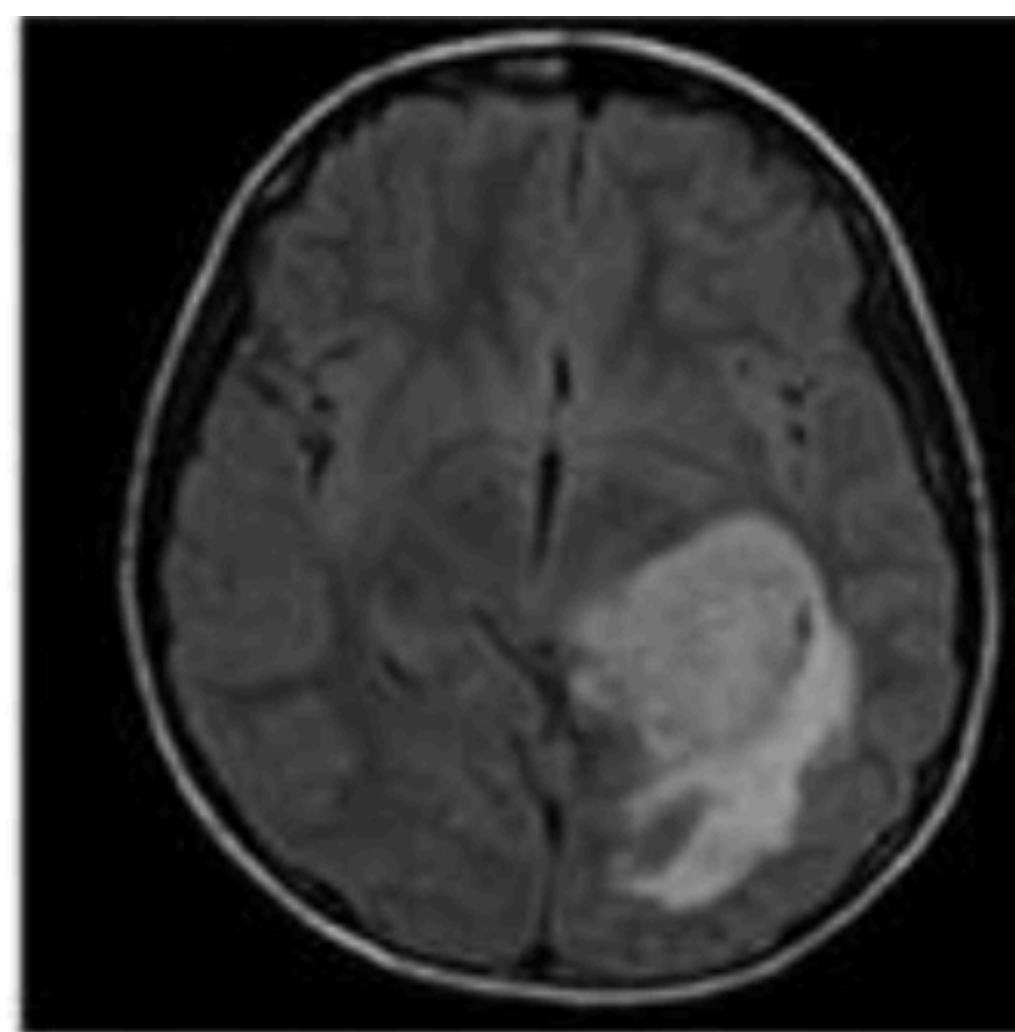
$$A \in \mathbb{R}^{H, W, D}$$

↑
Binary Mask!

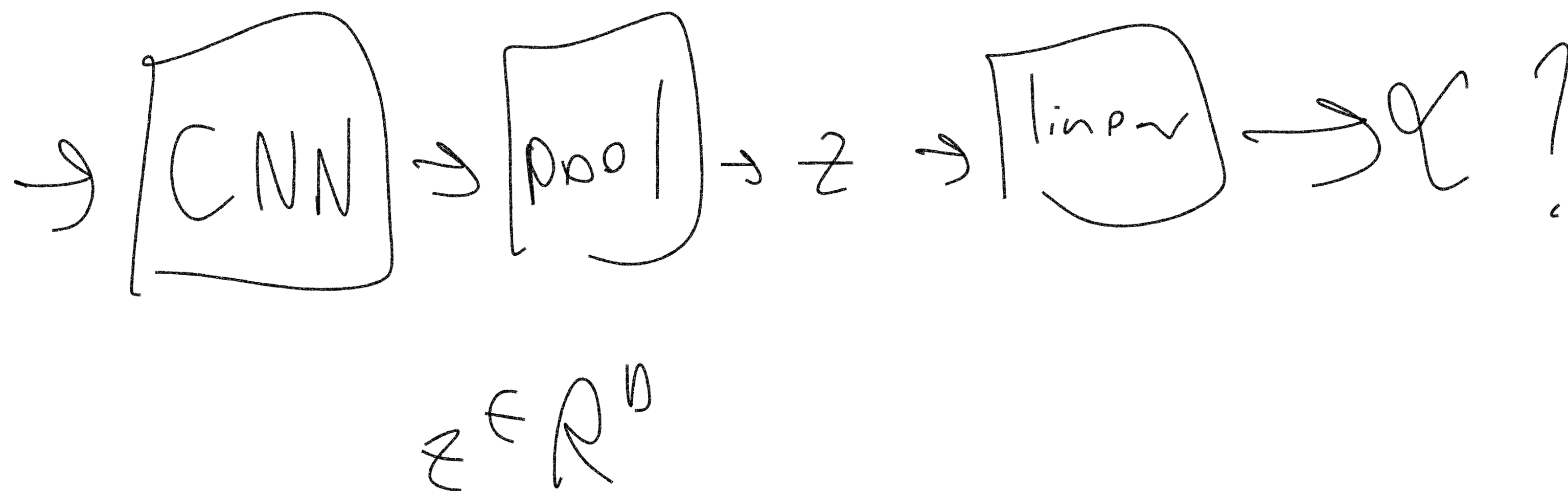
Why not attention?
softmax! coarse!

Baseline

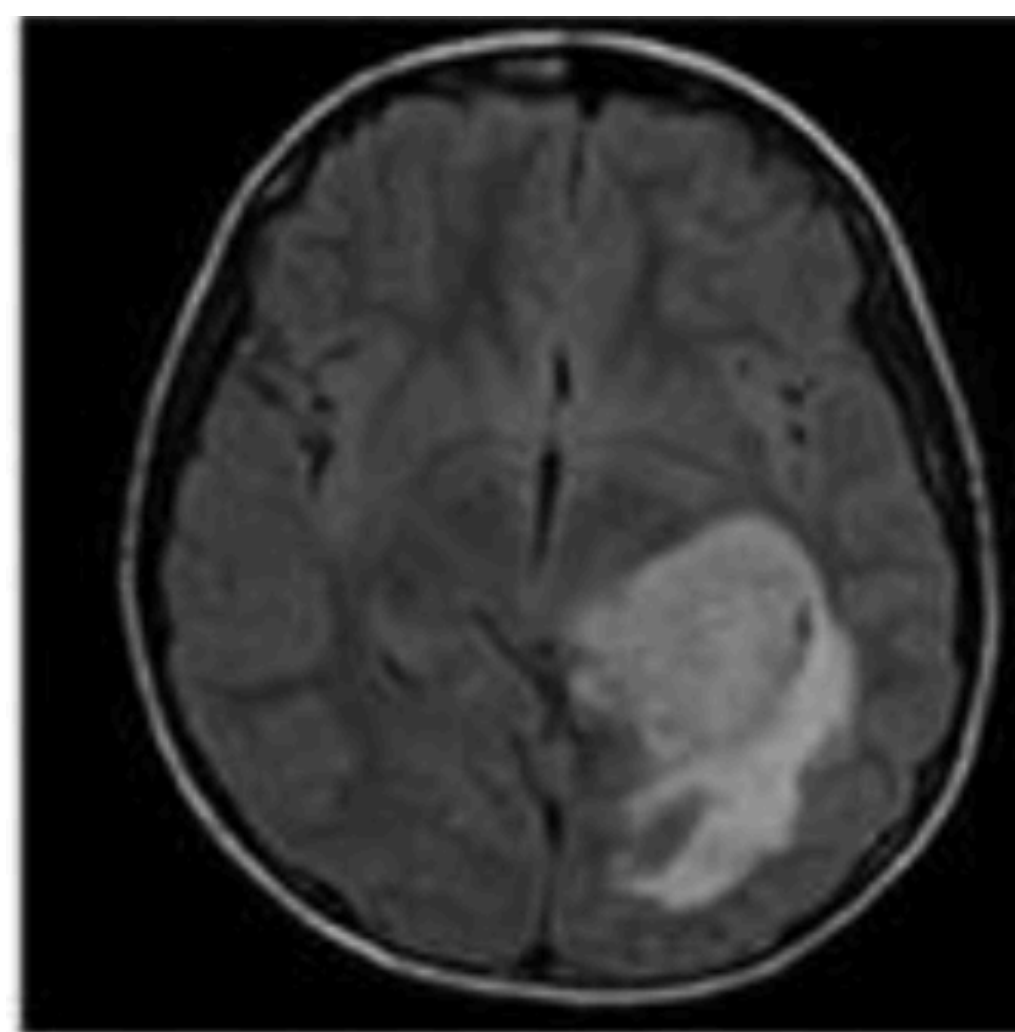
What's wrong?



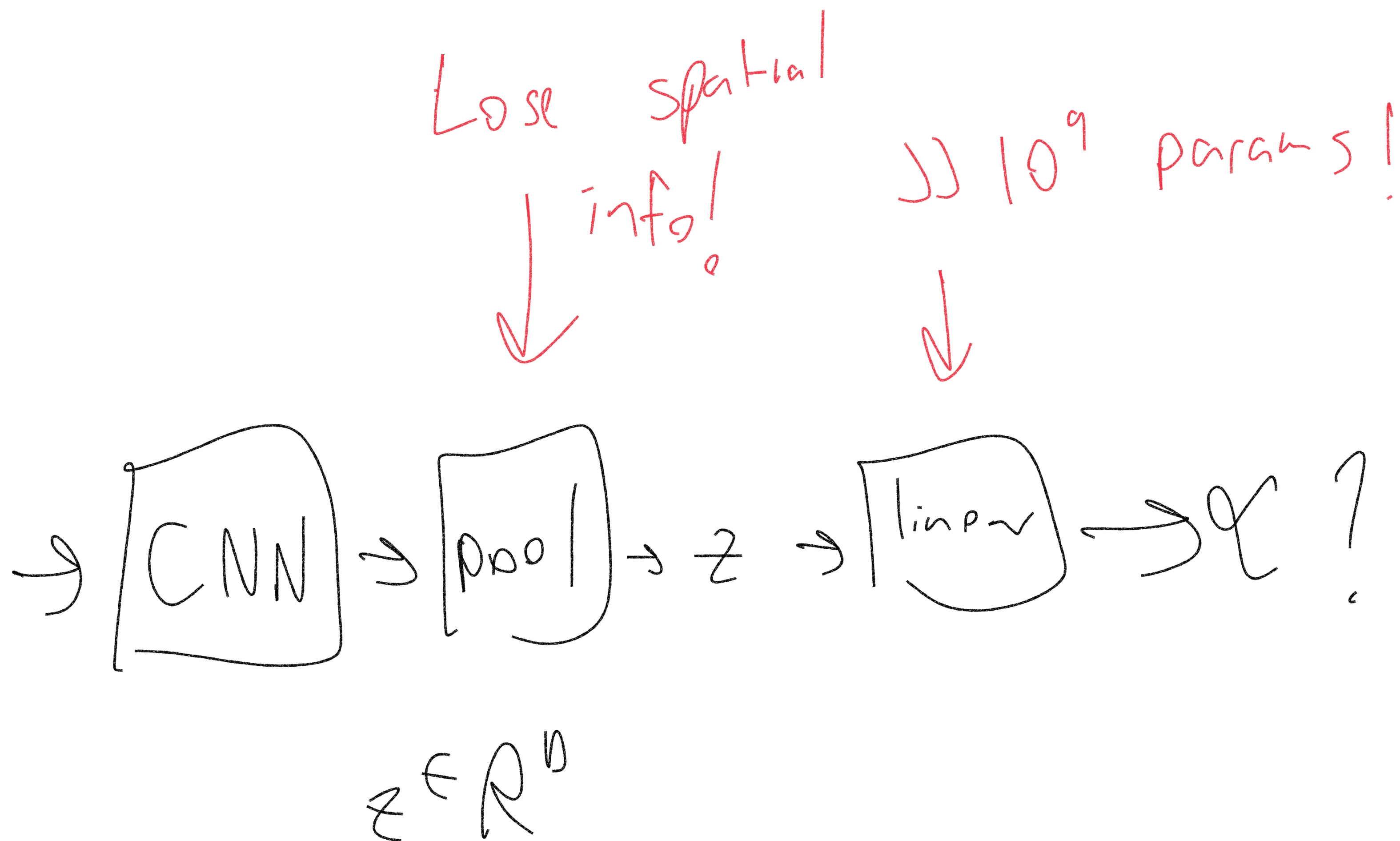
i



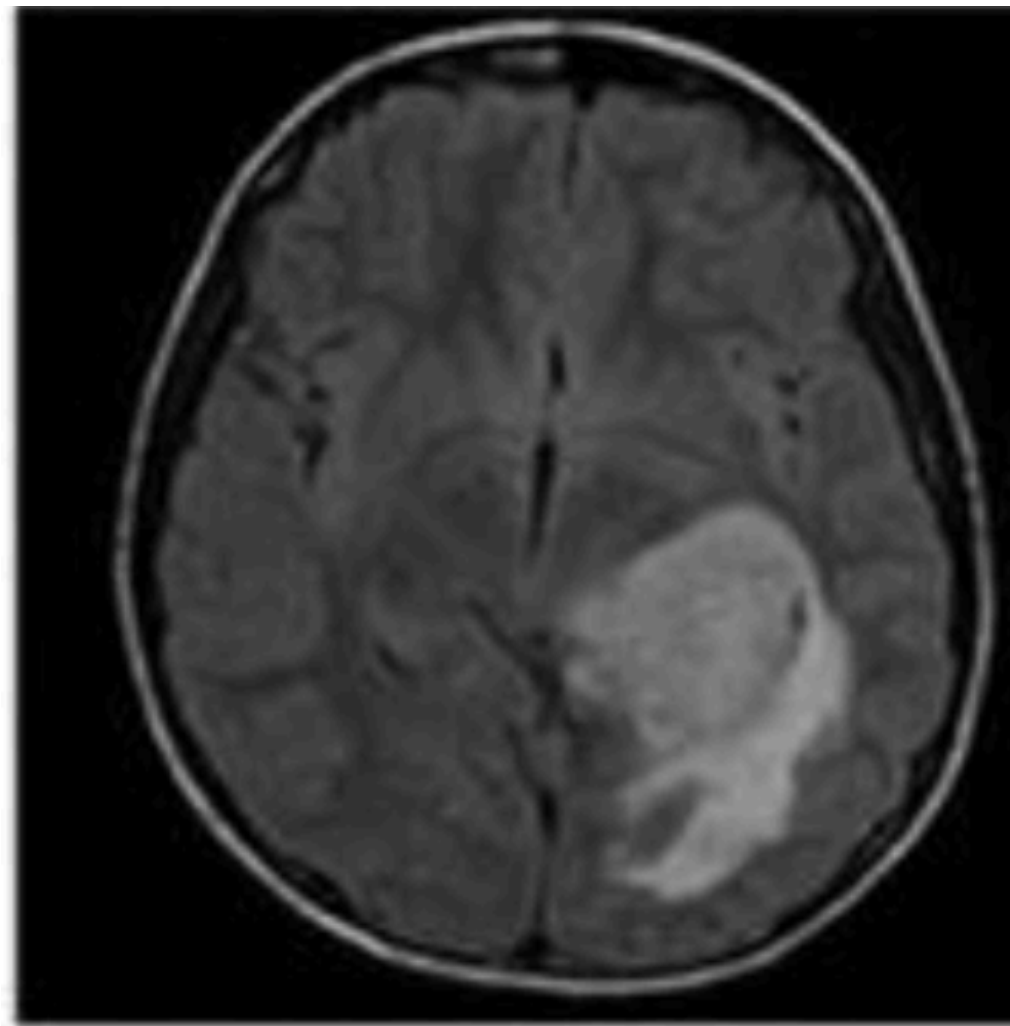
Baseline



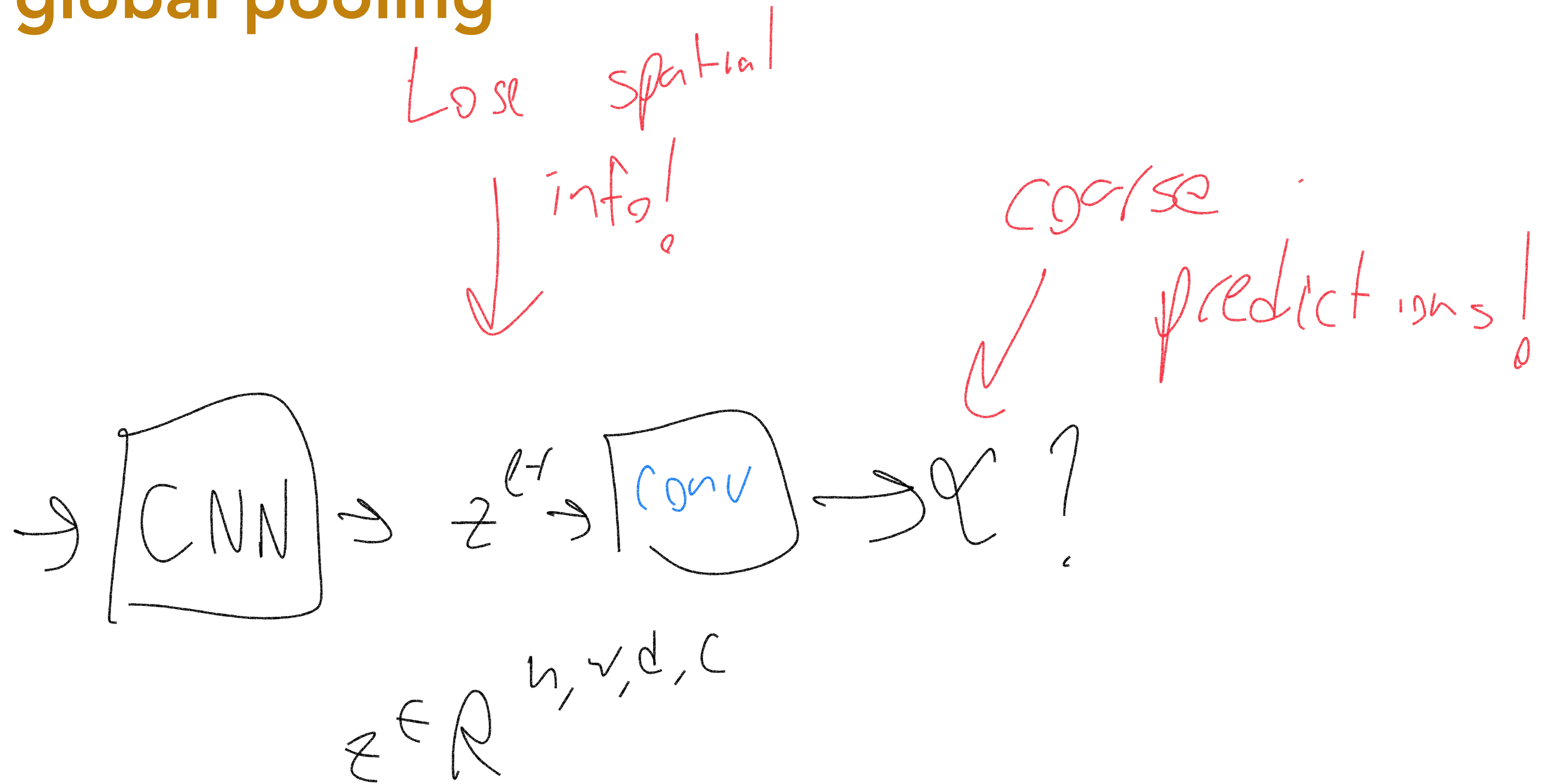
i



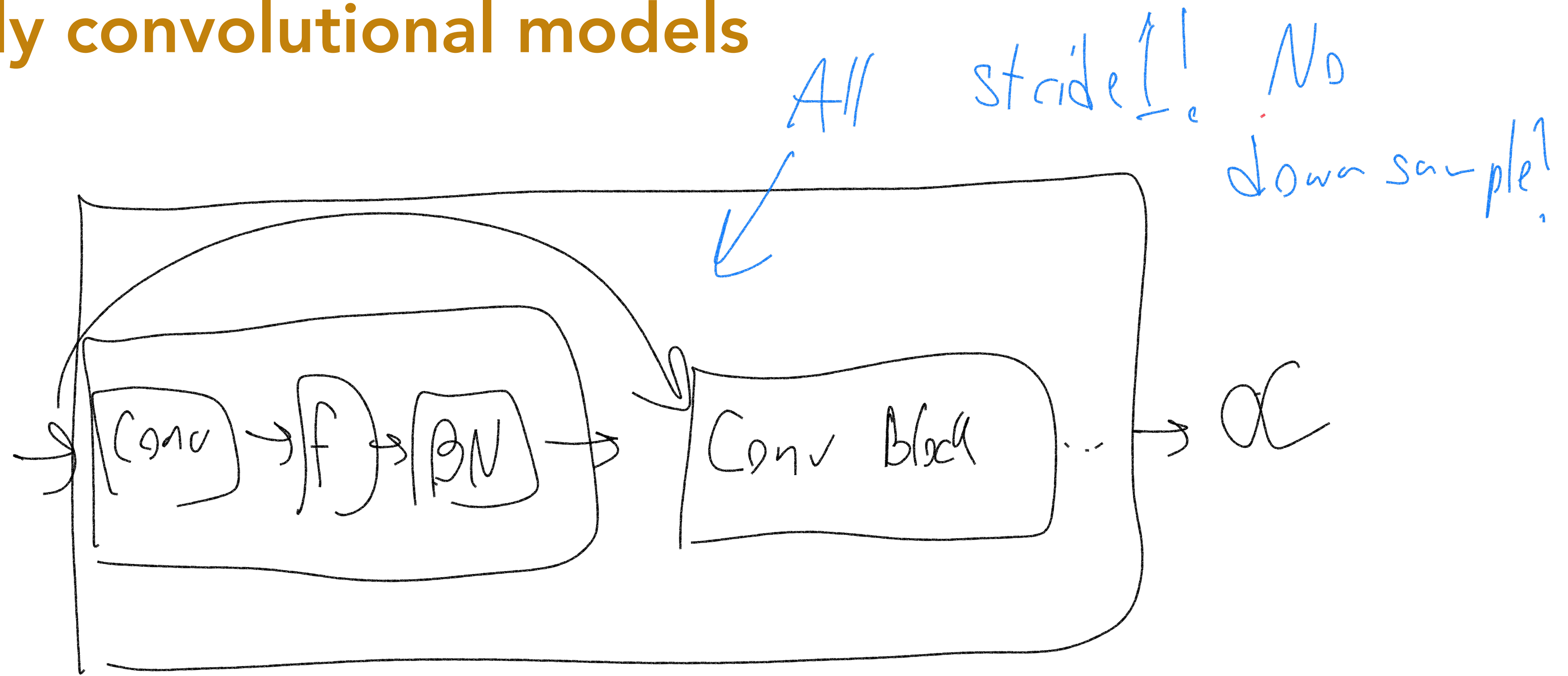
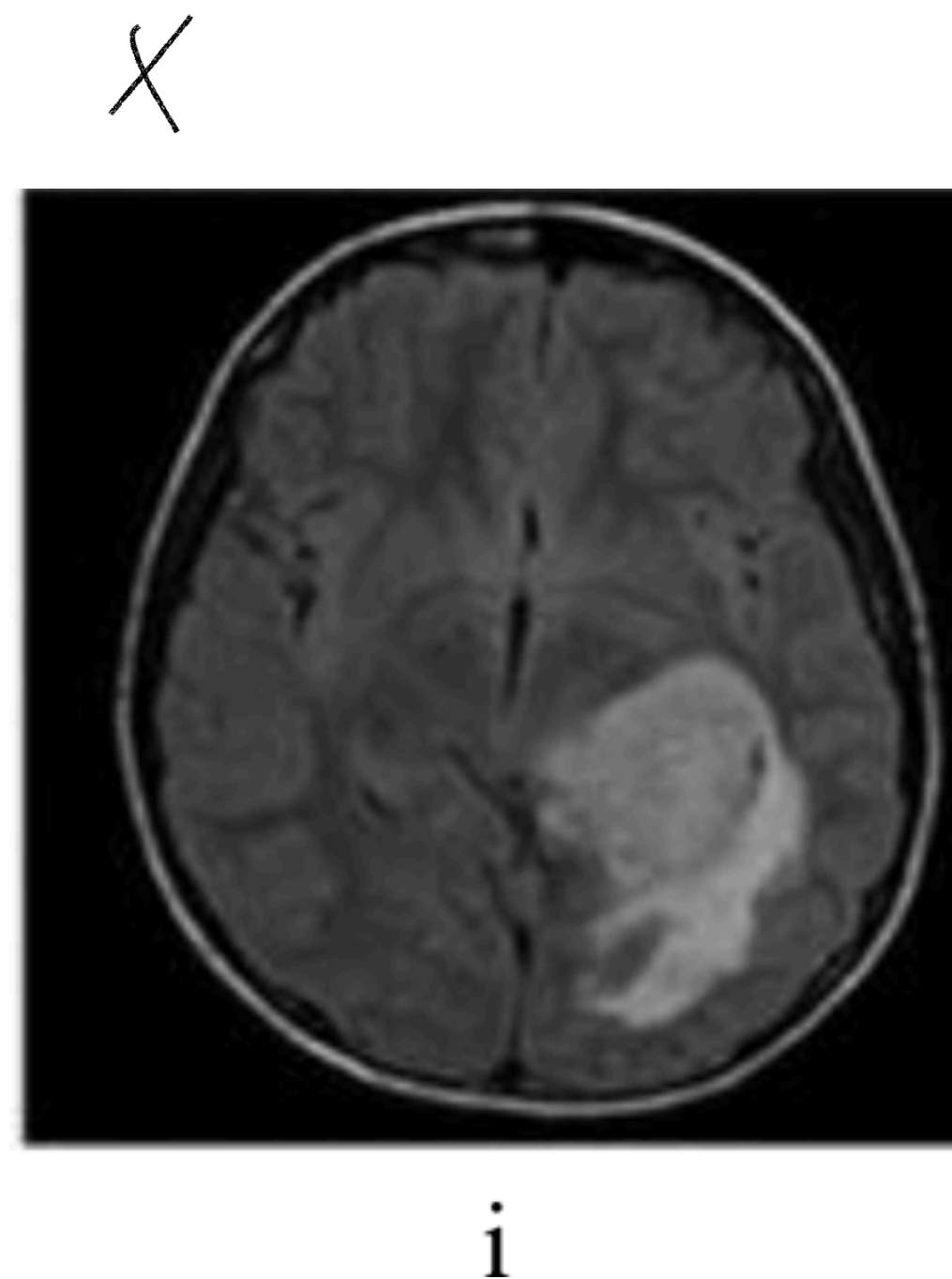
Baseline - No global pooling



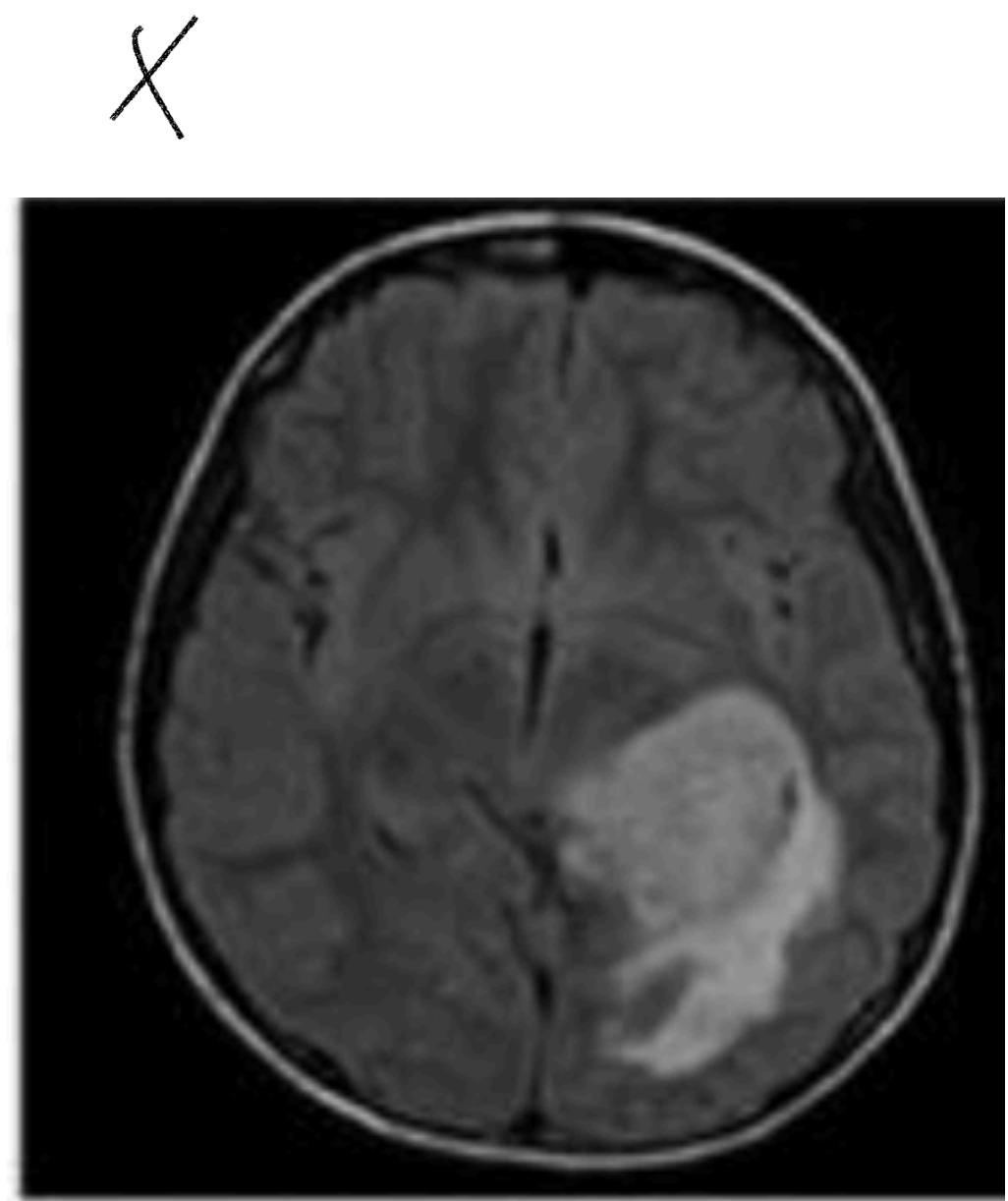
i



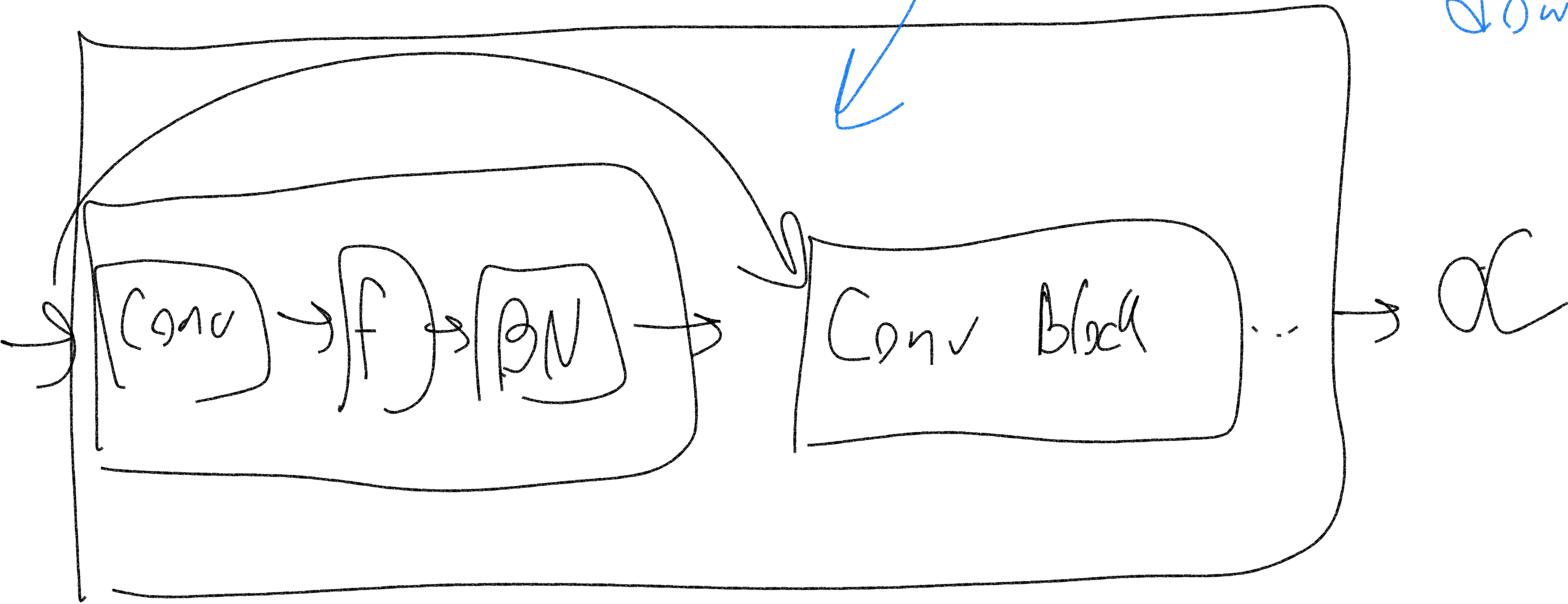
Baseline: fully convolutional models



Baseline: fully convolutional models



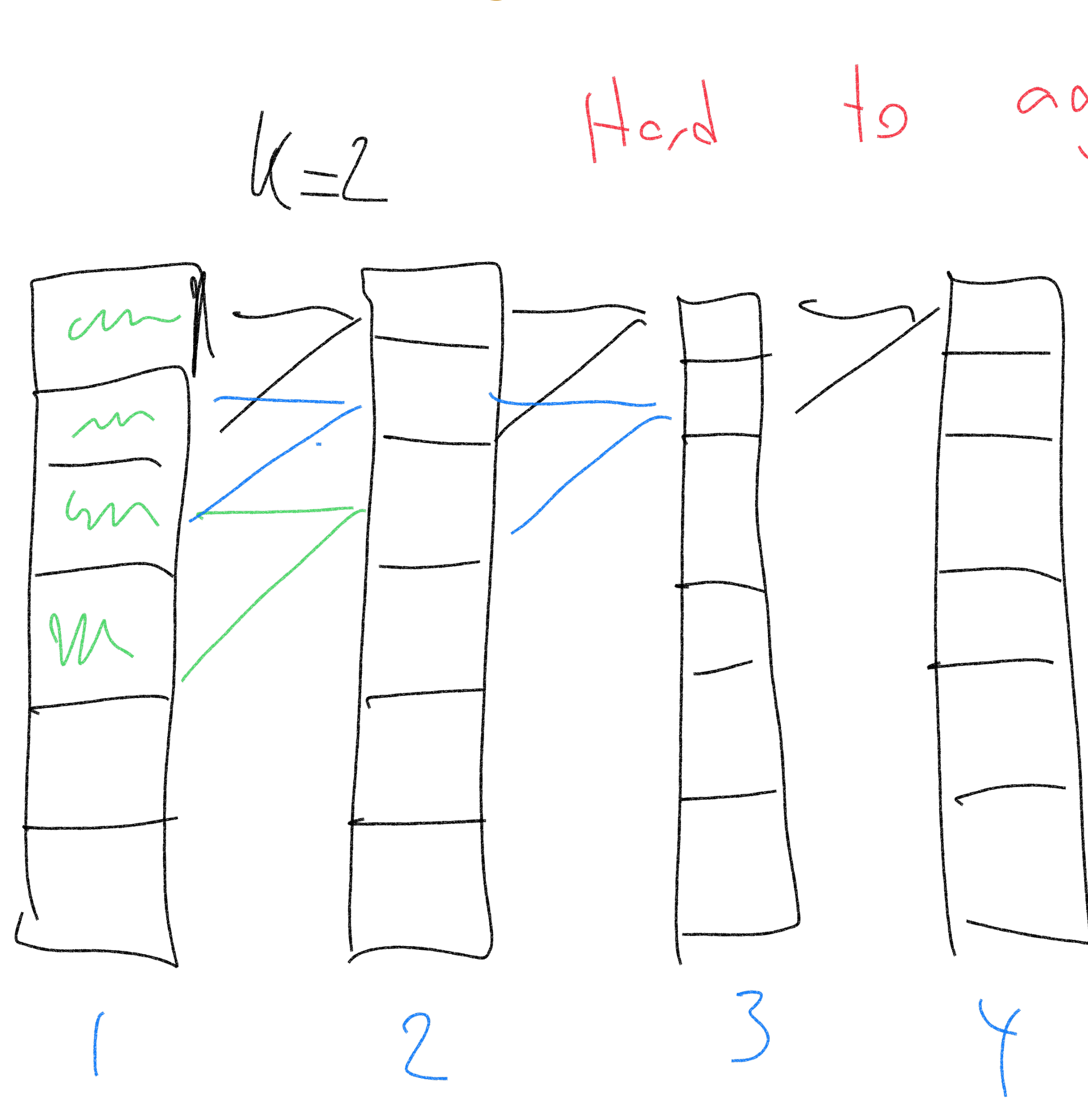
i



All stride 1! No down sample!

Hard to aggregate spatial context!
Low "receptive field"

Baseline: fully convolutional models

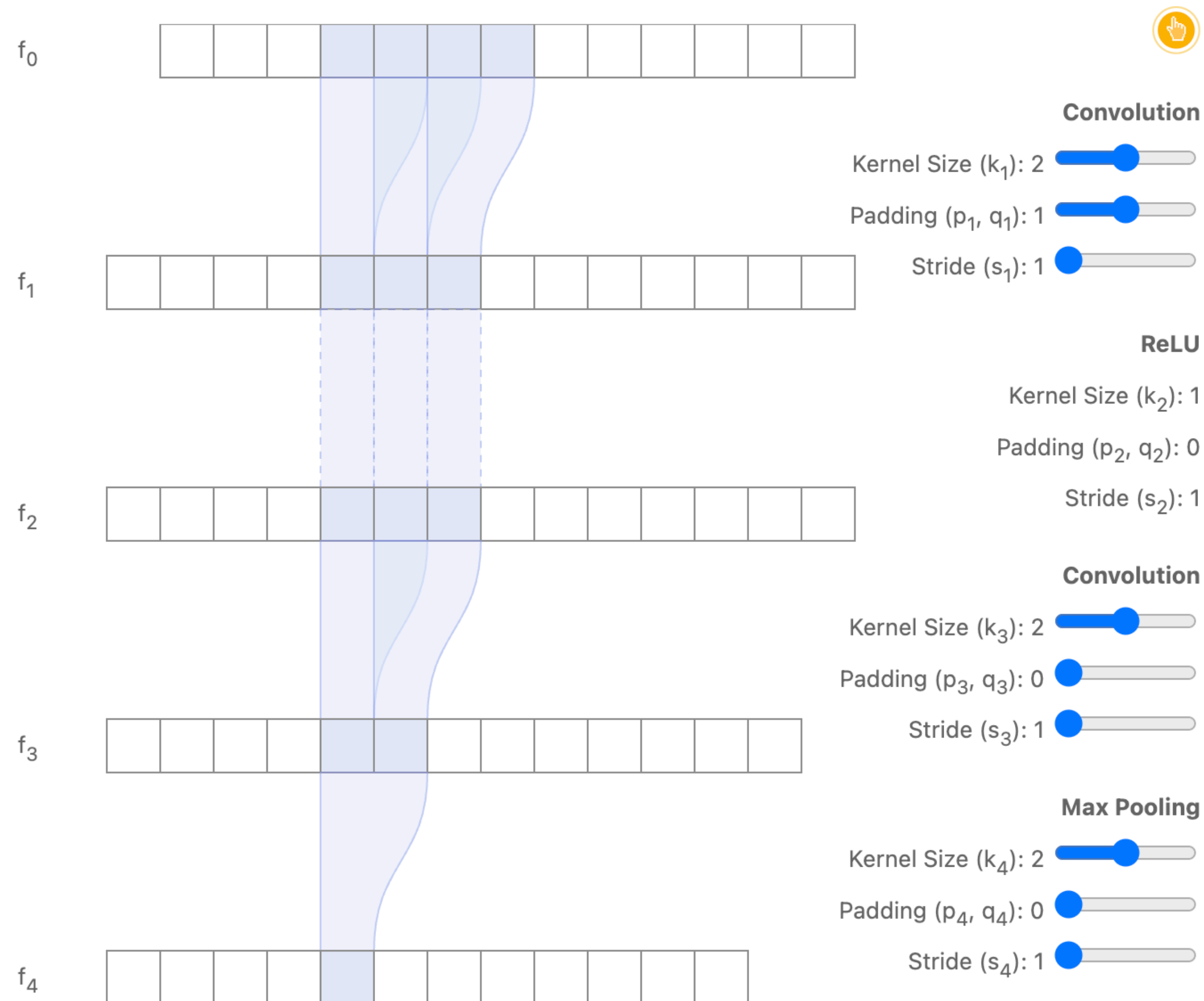


Hard to aggregate spatial context!

Low "receptive field"

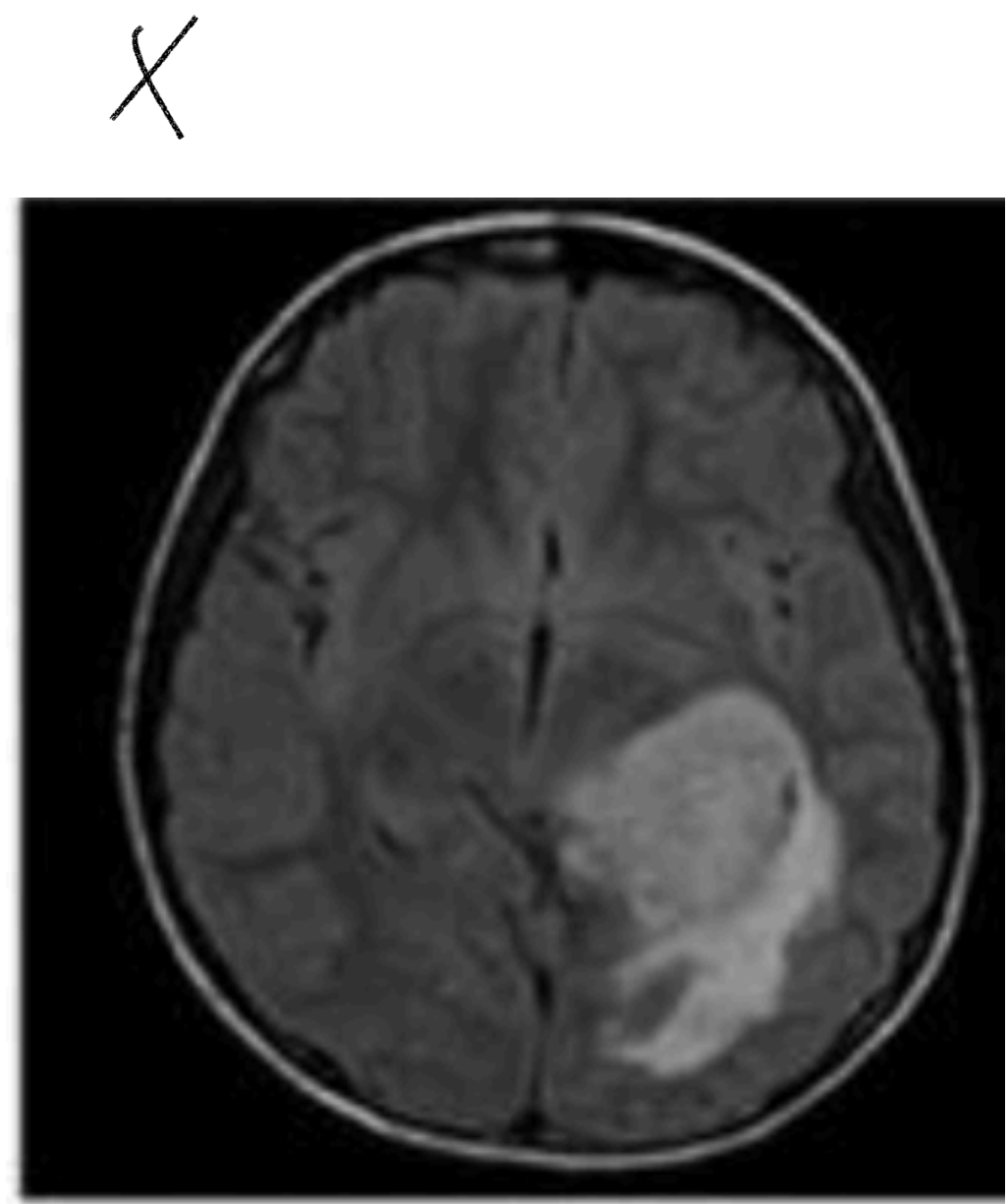
$$R_l = \sum_{i=1}^L (k_i - 1) + 1$$

Understanding receptive field in CNNs

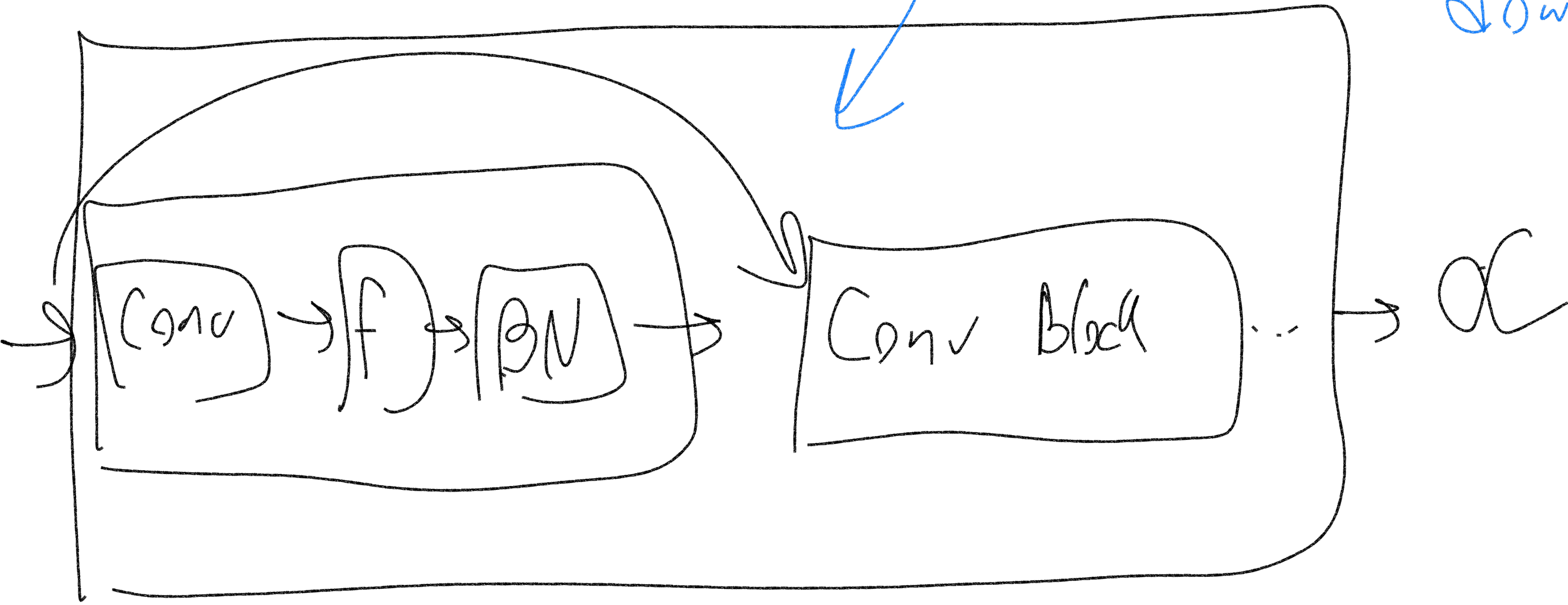


<https://distill.pub/2019/computing-receptive-fields>

Baseline: fully convolutional models



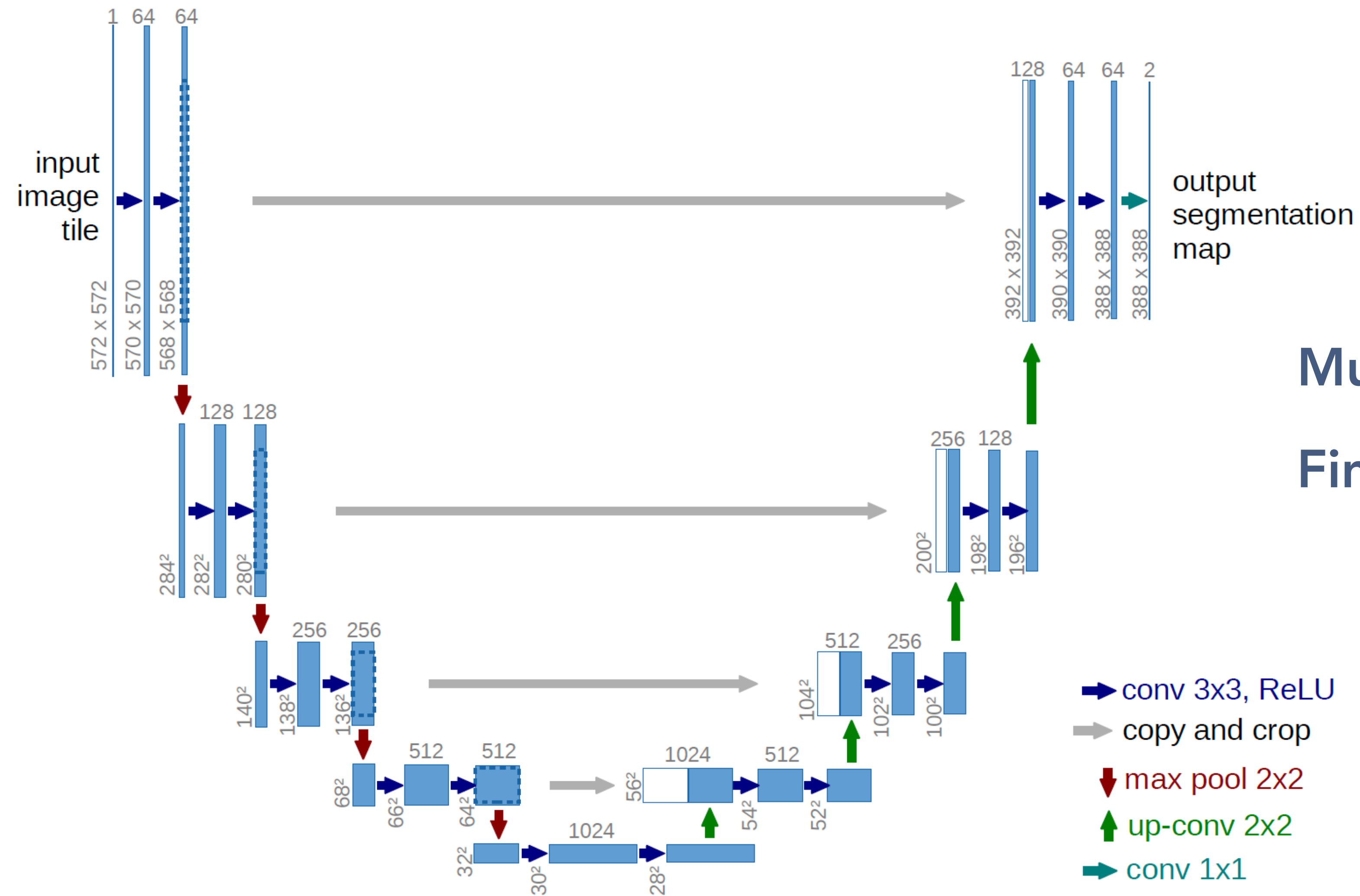
i



Has to be crazy deep ☹

inefficient

UNets: Multiscale representations



Mutli-scale representations!

Fine-grain representations!

Upsampling with convolutions: TransposeConv

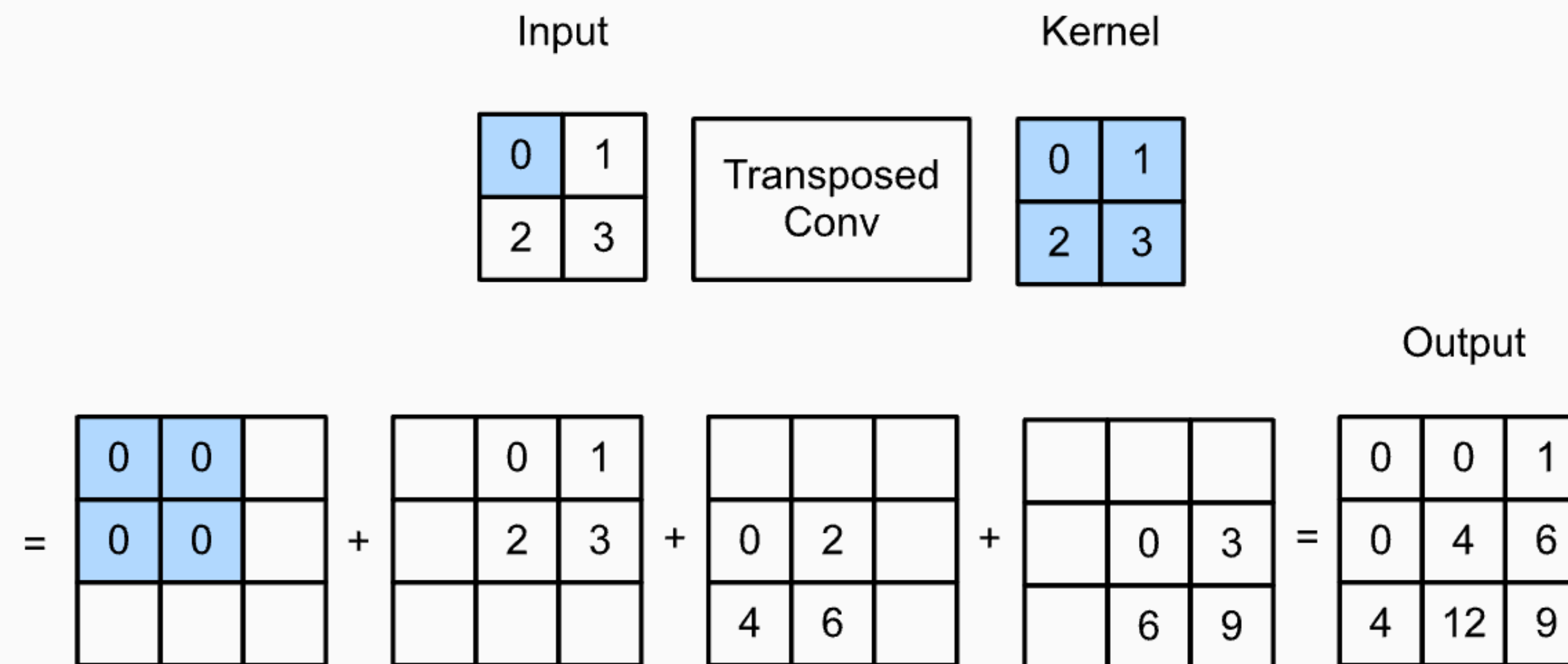
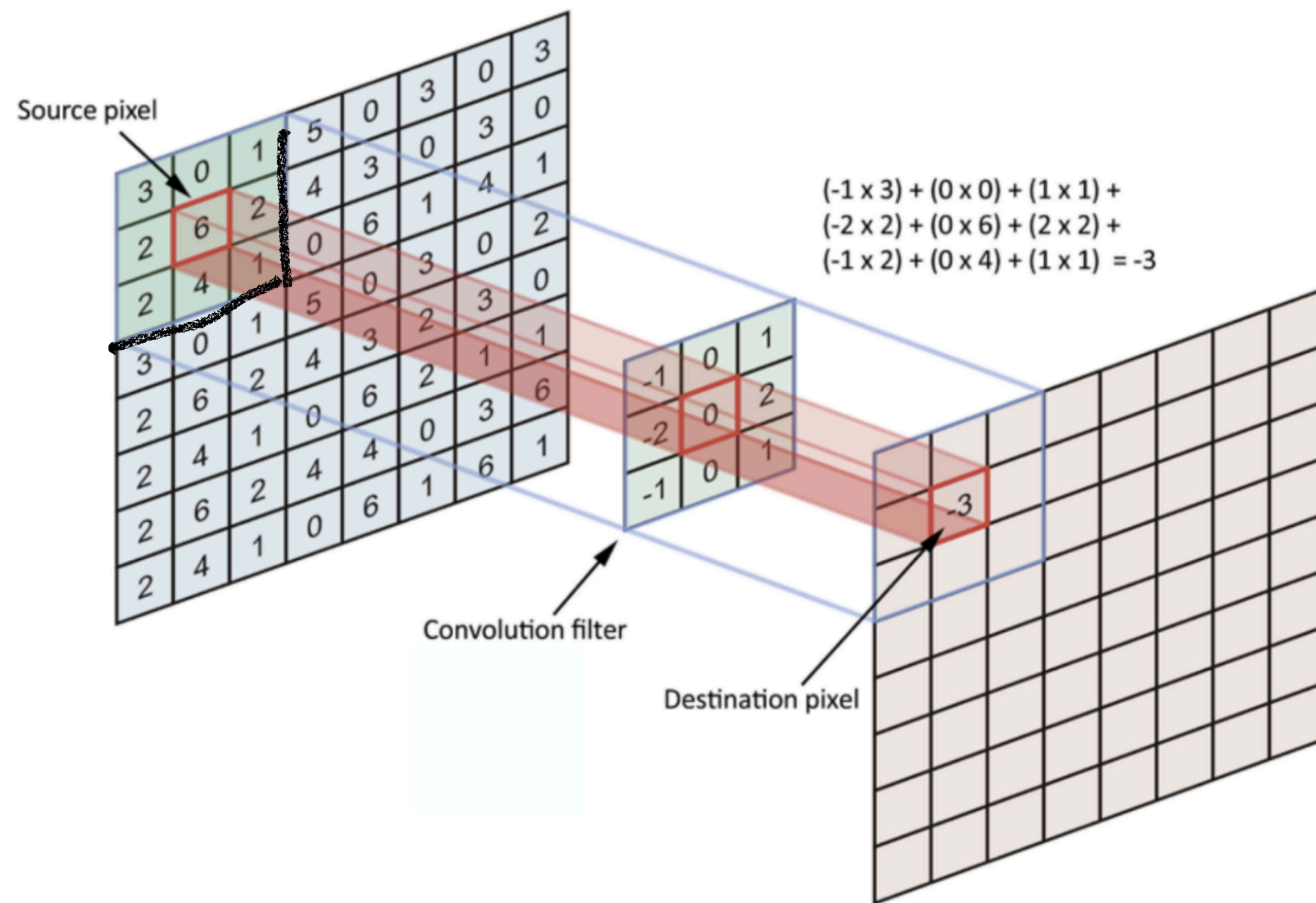


Fig. 14.10.1 Transposed convolution with a 2×2 kernel. The shaded portions are a portion of an intermediate tensor as well as the input and kernel tensor elements used for the computation.

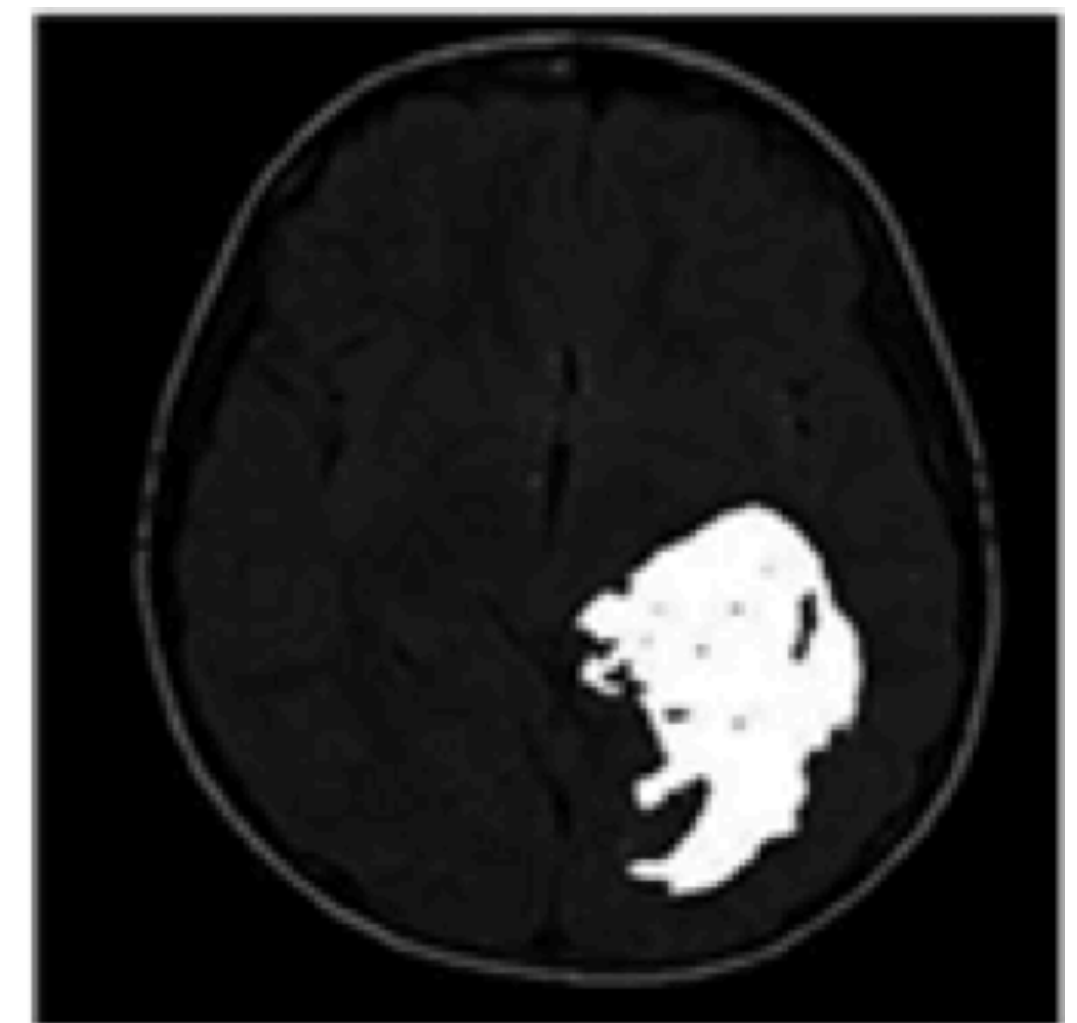
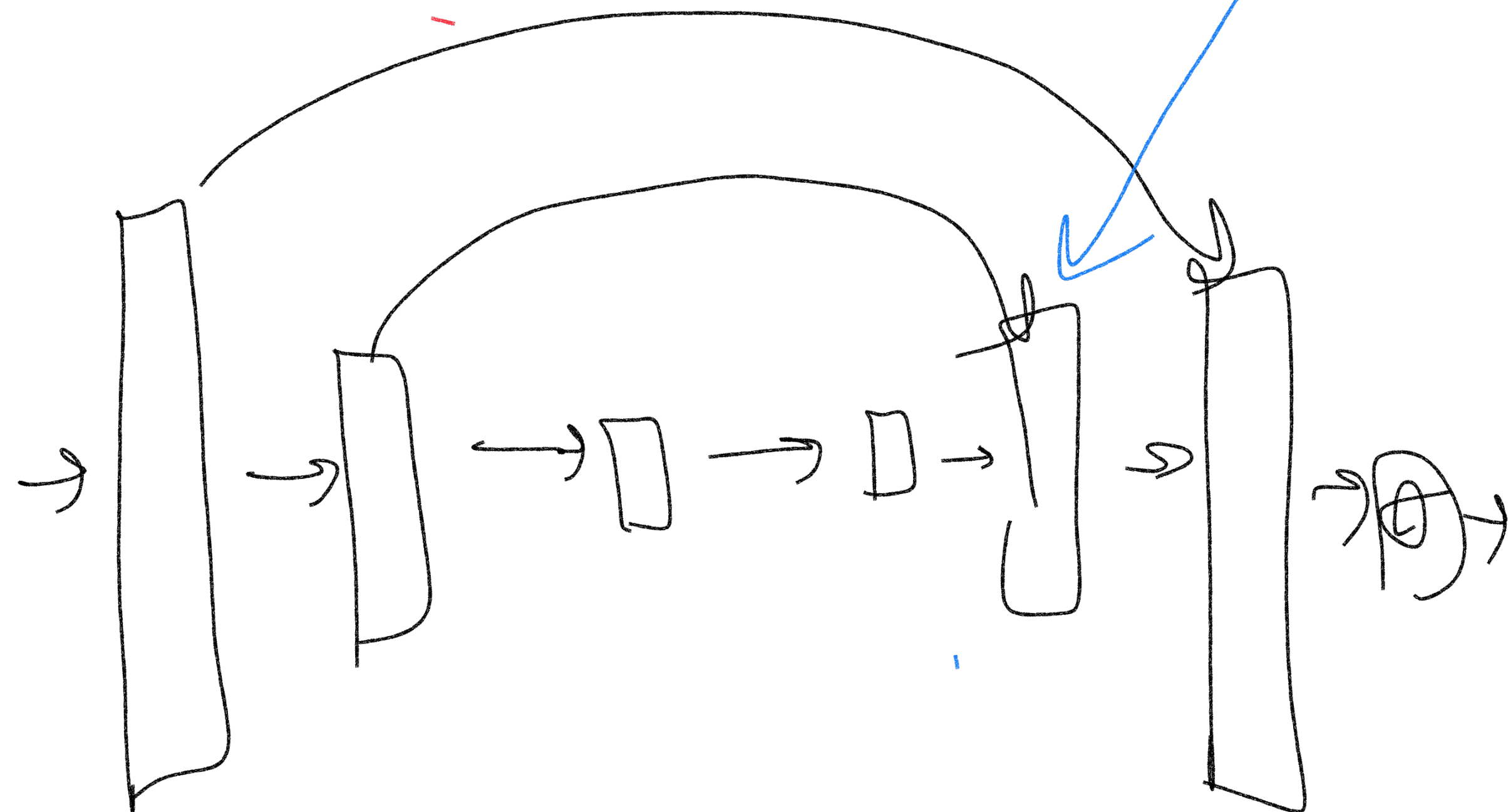
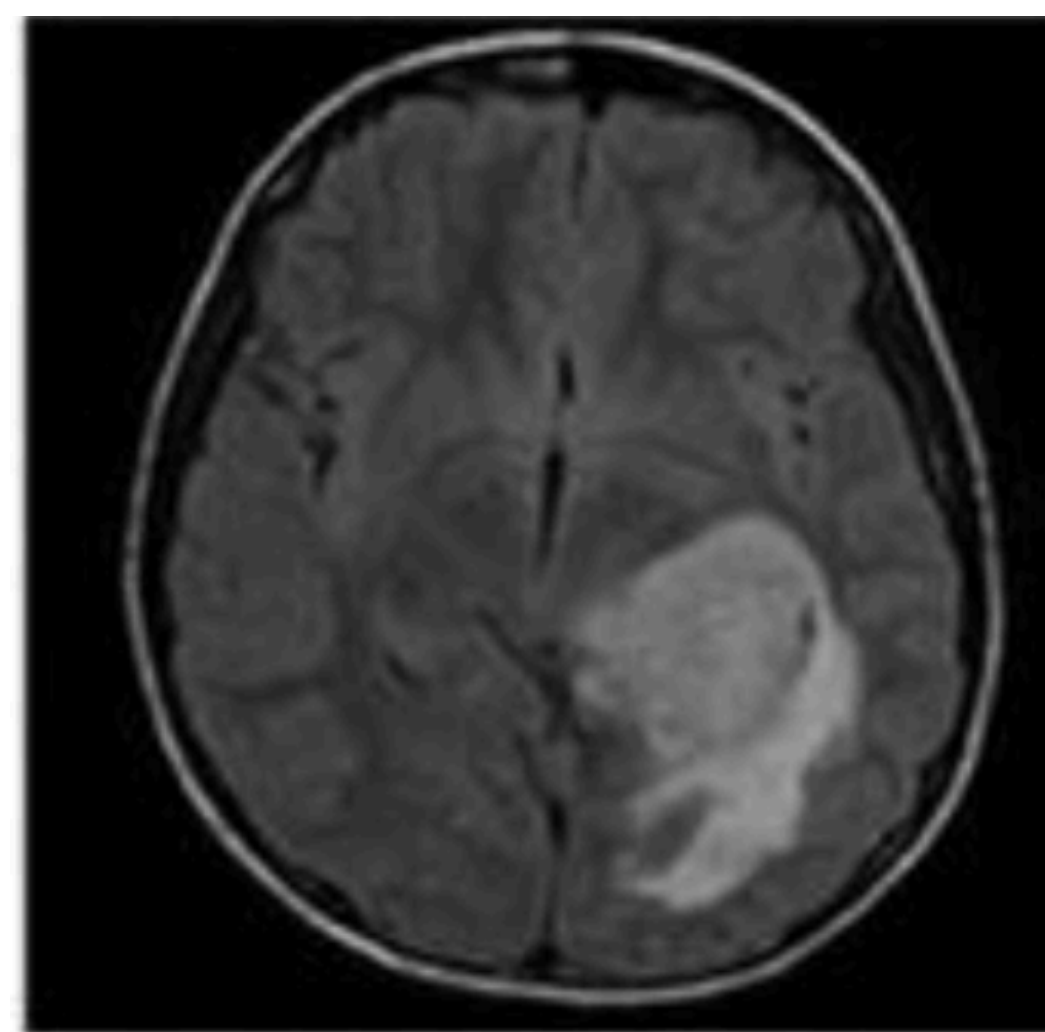
https://d2l.ai/chapter_computer-vision/transposed-conv.html

2D Convolutions



The convolution operation.

Putting it all together



$$L = \frac{1}{WHN} \sum a_{ijk} \log(a_{ijk}) + (1 - a_{ijk}) \log(1 - a_{ijk})$$

Summary

Why localization?

- Key output for some CPH interventions (e.g. Where to biopsy)
- Effective regularizer

Methods for localization:

- Loose Supervision: Constraining attention maps
- Bounding box regression: Capturing multiple objects of different sizes
- Segmentation: Pixel level predictions and UNets

Questions?