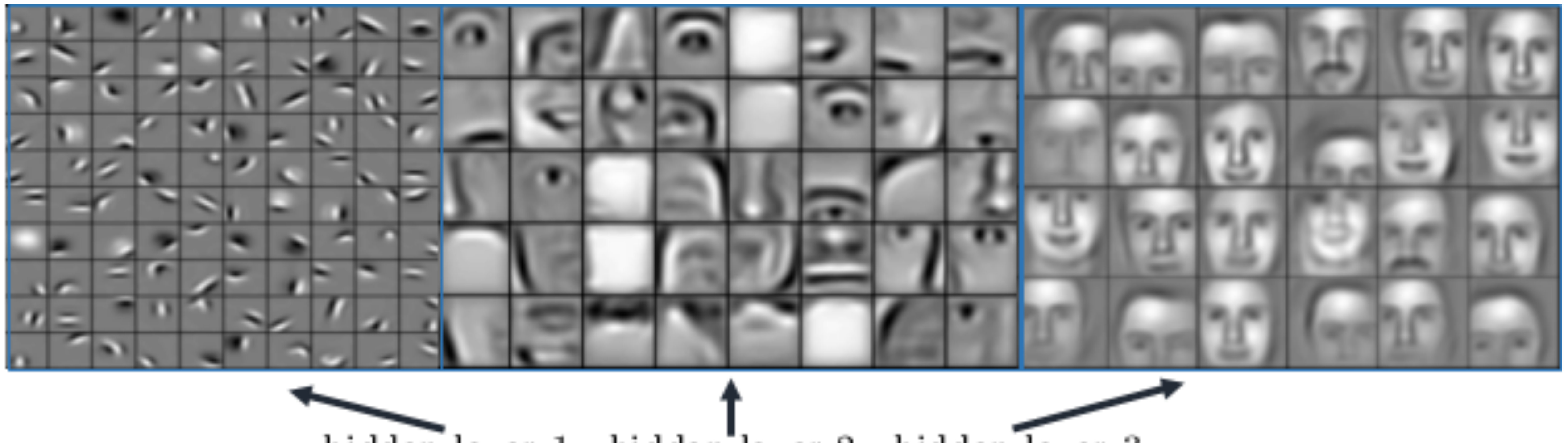
Deep Learning for Medical Imaging

Miguel Monteiro - INESC-ID
mab.mtr@gmail.com

Deep neural networks learn hierarchical feature representations



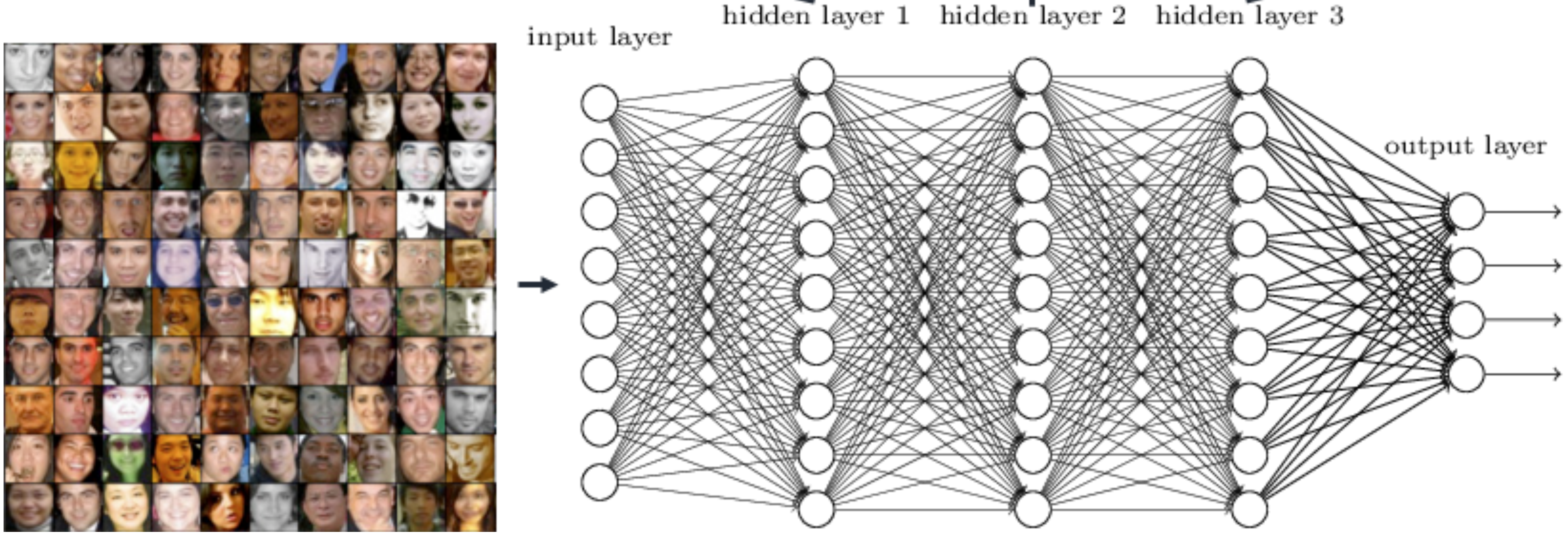
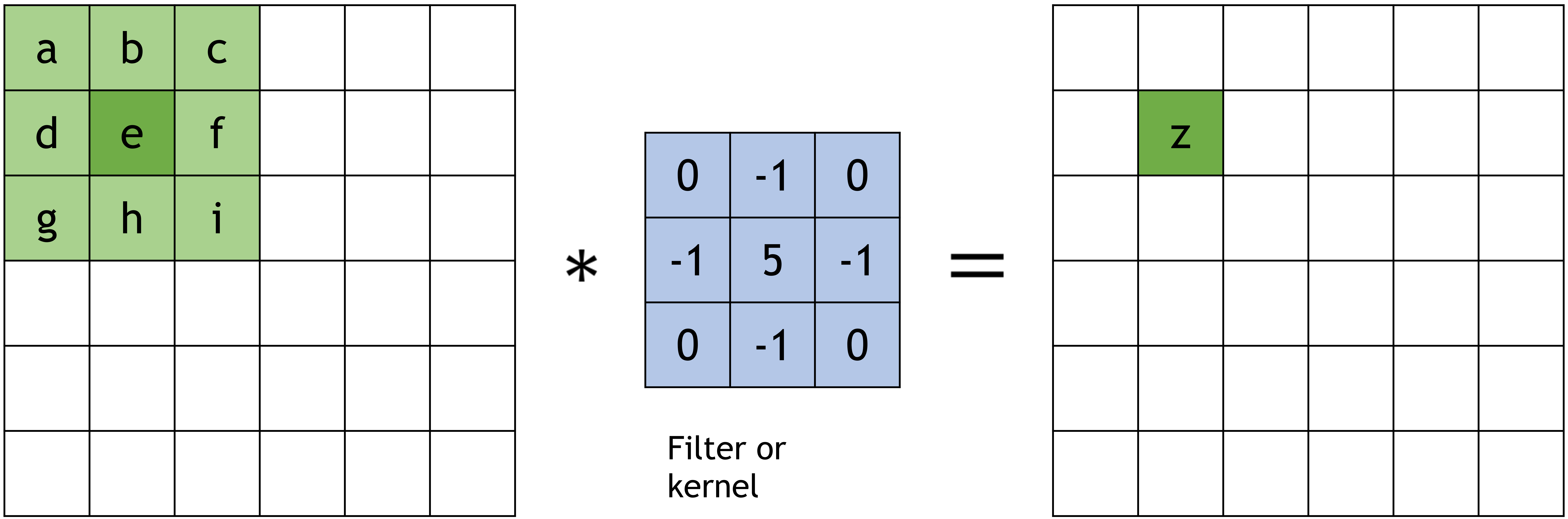


Image Convolution (1)



$$z = 0\times a + (-1)\times b + 0\times c + (-1)\times d + 5\times e + (-1)\times f + 0\times g + (-1)\times h + 0\times i$$

Image Convolution (2)

1	2	4	1	0	2
0	1	0	0	1	1
1	0	3	2	3	0
4	3	4	1	0	1
2	4	1	1	2	0
4	2	5	2	6	4

$*$

0	-1	0
-1	5	-1
0	-1	0

Filter or kernel

$=$

	3				

$$3 = 1\times 0 + 2\times(-1) + 4\times 0 + 0\times(-1) + 1\times 5 + 0\times(-1) + 1\times 0 + 0\times(-1) + 3\times 0$$

Image Convolution (3)

$$\begin{bmatrix}1&2&4&1&0&2\\0&1&0&0&1&1\\1&0&3&2&3&0\\4&3&4&1&0&1\\2&4&1&1&2&0\\4&2&5&2&6&4\end{bmatrix} * \begin{bmatrix}0&-1&0\\-1&5&-1\\0&-1&0\end{bmatrix} = \begin{bmatrix} & & & & & \\ & 3 & -8 & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \end{bmatrix}$$

Filter or kernel

Image Convolution (4)

1	2	4	1	0	2
0	1	0	0	1	1
1	0	3	2	3	0
4	3	4	1	0	1
2	4	1	1	2	0
4	2	5	2	6	4

$*$

0	-1	0
-1	5	-1
0	-1	0

Filter or kernel

$=$

	3	-8	-4		

Image Convolution (5)

1	2	4	1	0	2
0	1	0	0	1	1
1	0	3	2	3	0
4	3	4	1	0	1
2	4	1	1	2	0
4	2	5	2	6	4

*

0	-1	0
-1	5	-1
0	-1	0

Filter or kernel

=

	3	-8	-4	1	

Image Convolution (6)

1	2	4	1	0	2
0	1	0	0	1	1
1	0	3	2	3	0
4	3	4	1	0	1
2	4	1	1	2	0
4	2	5	2	6	4

*

0	-1	0
-1	5	-1
0	-1	0

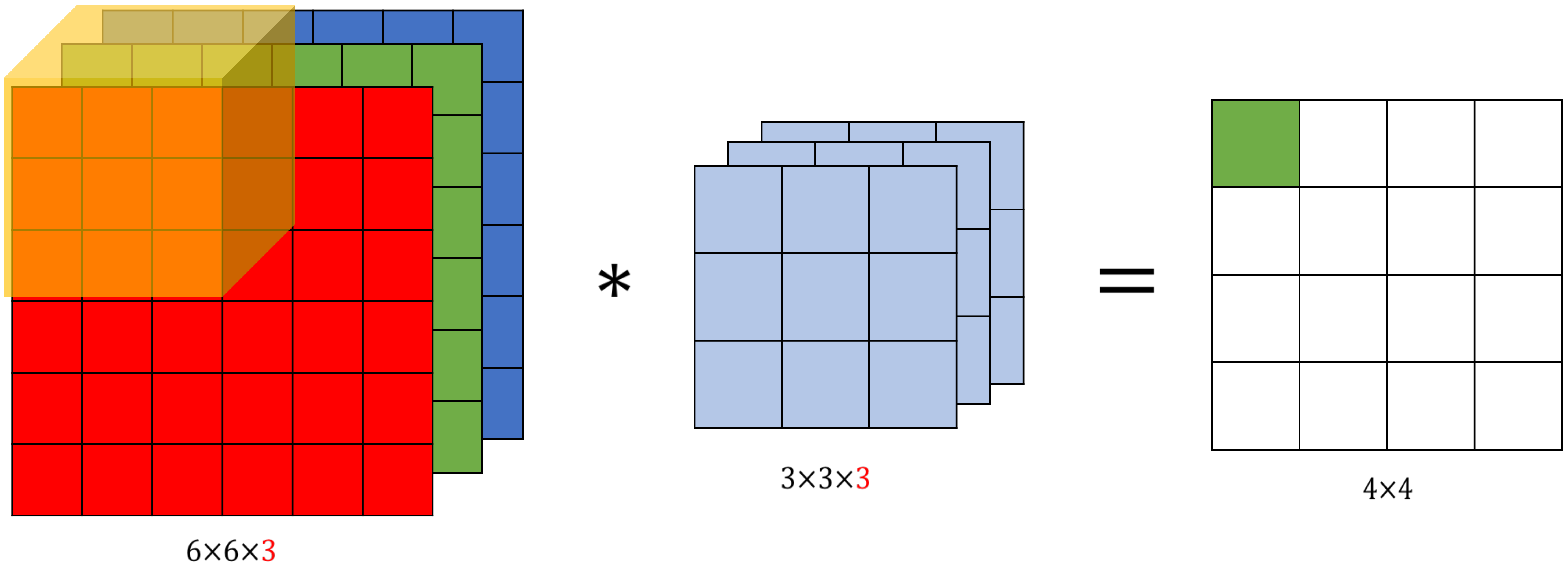
Filter or kernel

=

	3	-8	-4	1	
	-8				

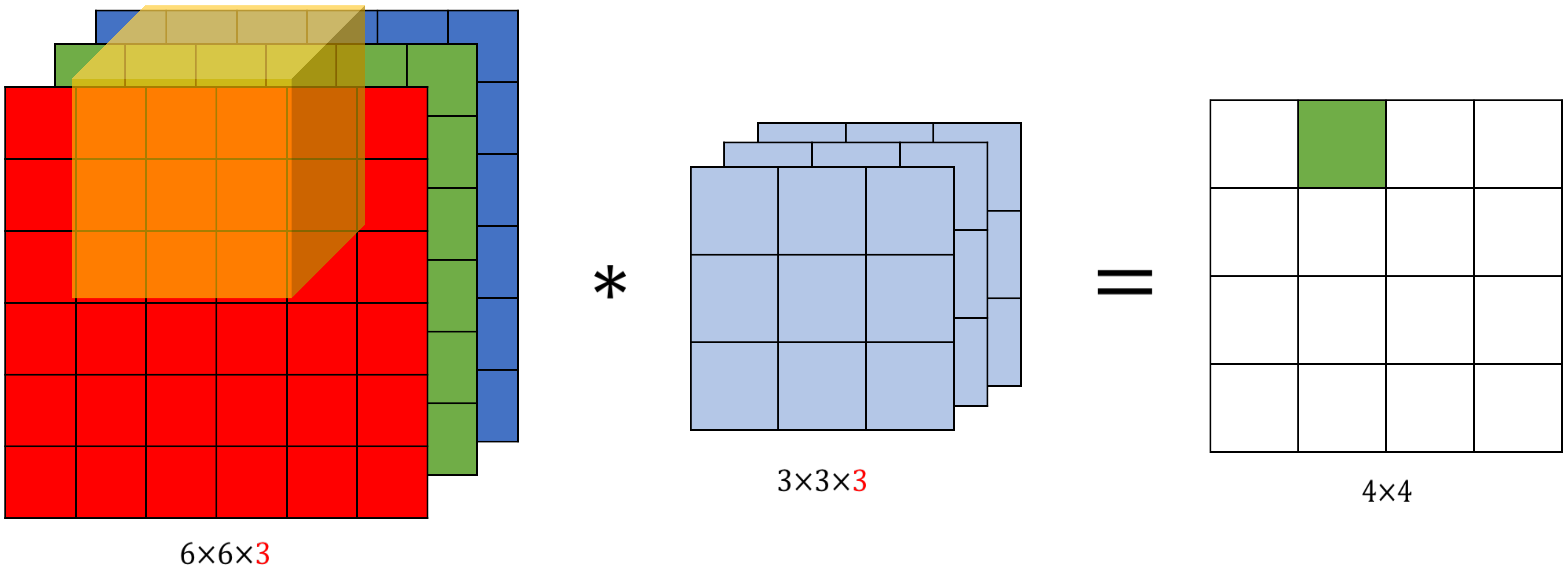
And so on...

Convolutions with multiple input channels (RGB images) (1)



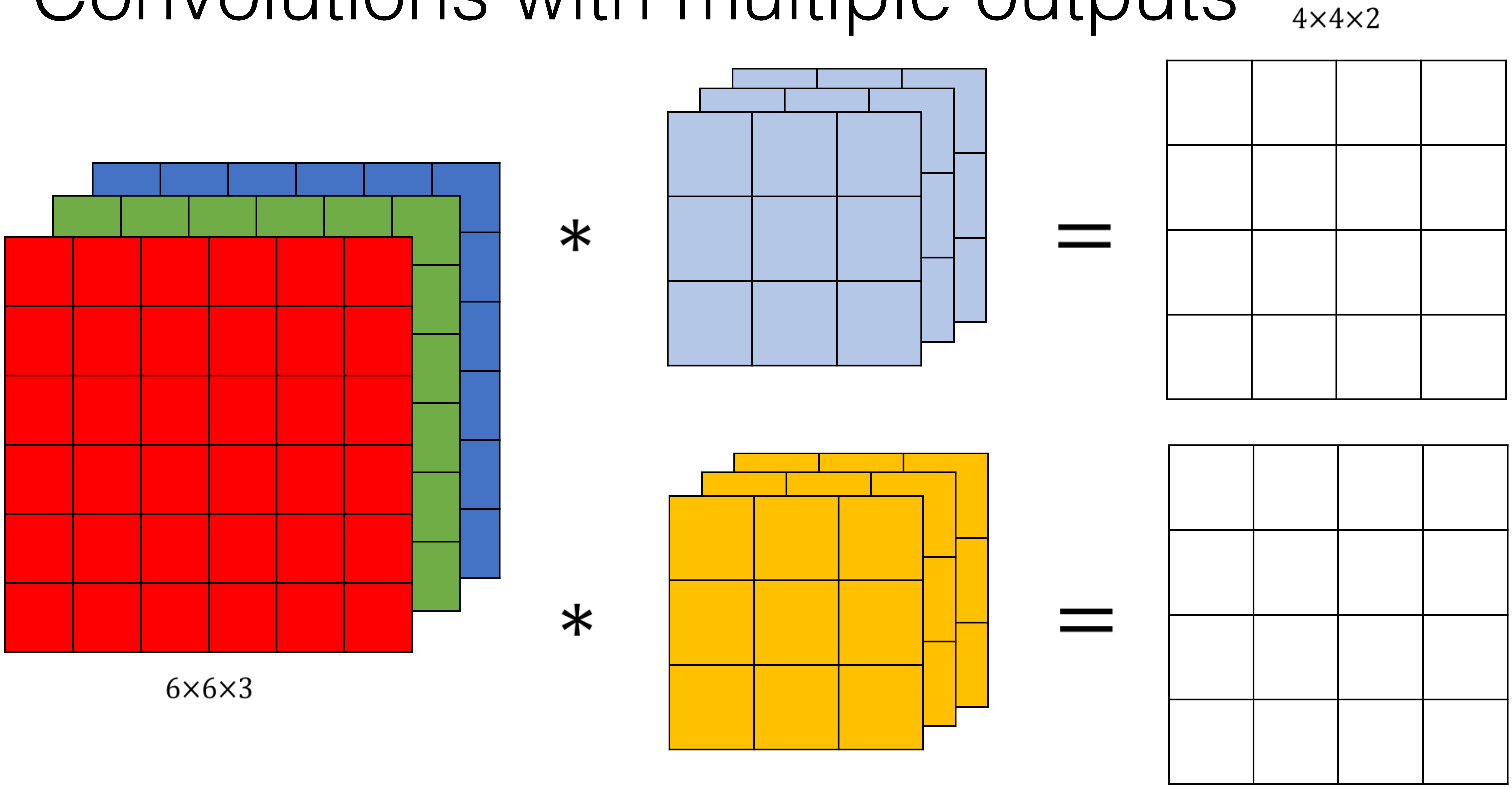
The red 3 is the number of input channels

Convolutions with multiple input channels (RGB images) (2)

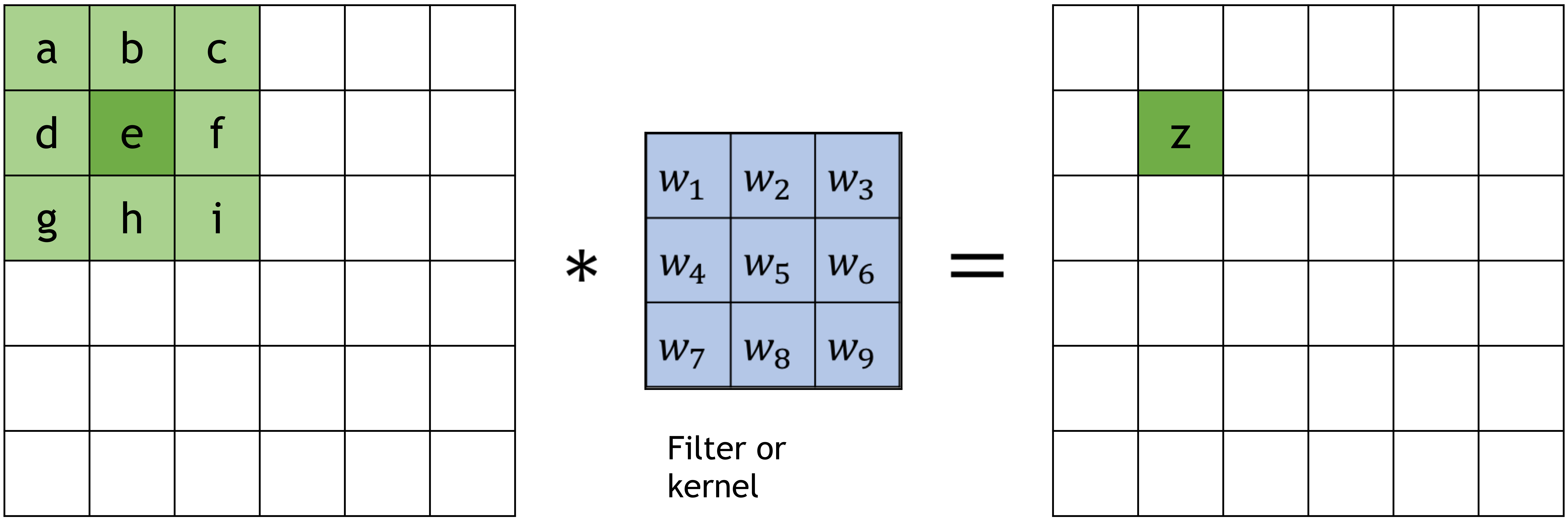


The red 3 is the number of input channels

Convolutions with multiple outputs

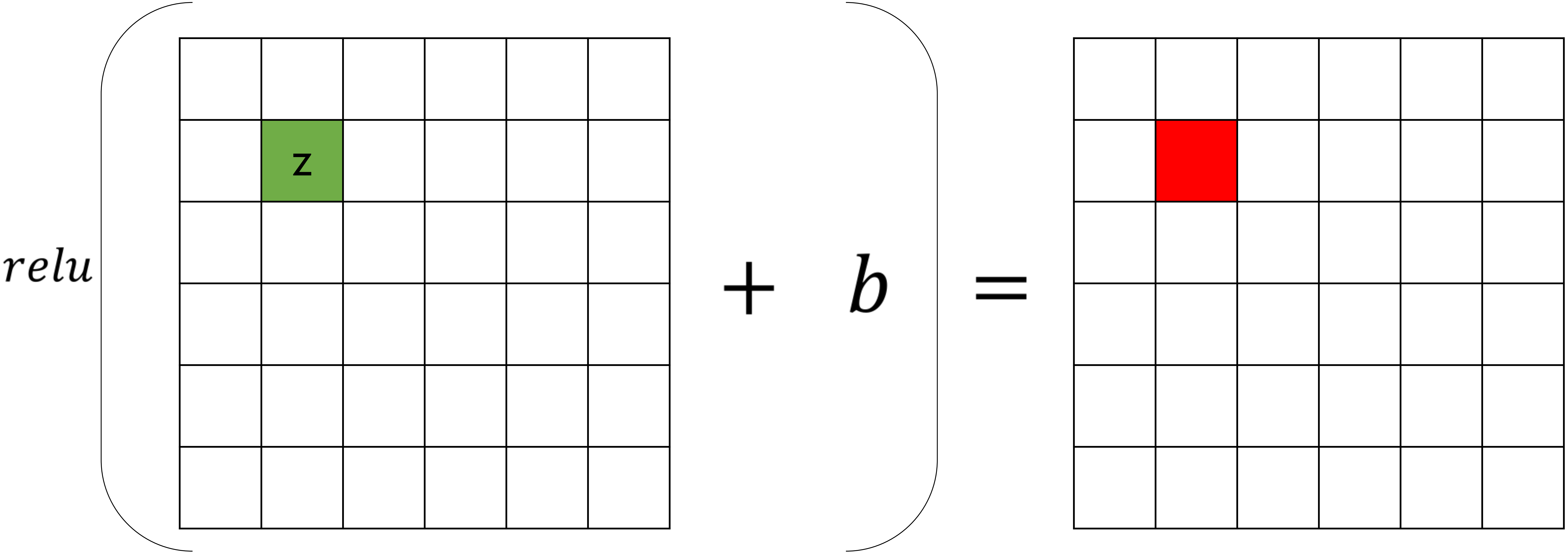


Convolution Layer (2)



$$z = relu(w_1 a + w_2 b + w_3 c + w_4 d + w_5 e + w_6 f + w_7 g + w_8 h + w_9 i + b)$$

Convolution Layer (3)

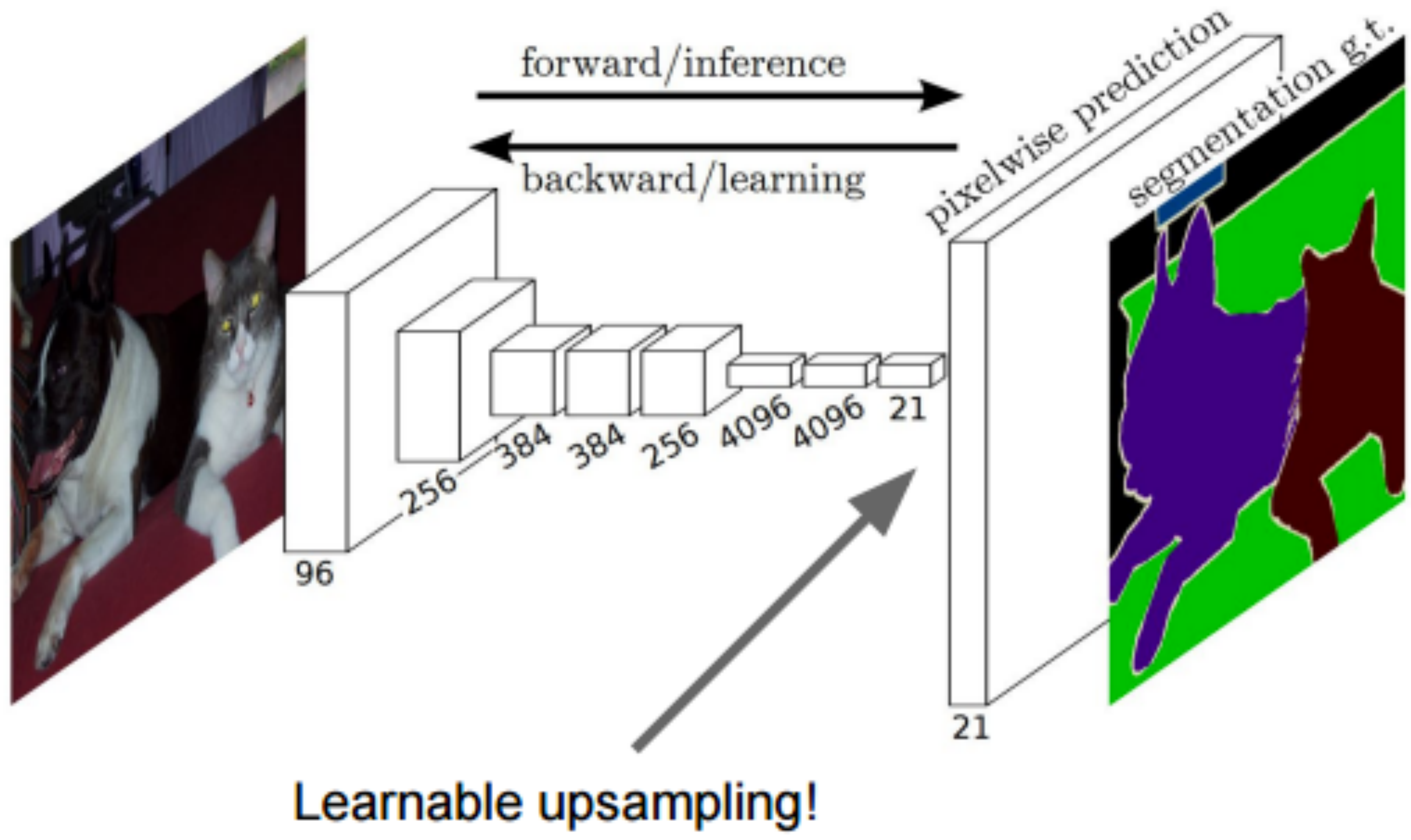


$$activation = relu(w_1 a + w_2 b + w_3 c + w_4 d + w_5 e + w_6 f + w_7 g + w_8 h + w_9 i + b)$$

http://cs231n.github.io/assets/conv-demo/index.html

Convolution Neural Network (CNN)

Fully Convolutional Neural Network (FCNN)



Deep Learning for Prediction of Ischemic Stroke Lesion Segmentation

Miguel Monteiro - INESC-ID
mab.mtr@gmail.com

Motivation (1)

- Ischemic Stroke is one of the leading causes of death and disability worldwide [Truelsen et al.];
- Patient outcome is extremely dependent on treatment decisions physicians make during the acute phase (around less than 3 hours after stroke);
- Stroke treatment procedures always involve medical imaging techniques:
  - Computed Tomography (CT);
  - Magnetic resonance imaging (MRI).

Motivation (2)

- The ischemic stroke lesion seen in the image can be seen by trained physicians;
- What is less clear is how the lesion will evolve over time and react to treatment;
- Automatic semantic segmentation and prediction methods can help;
- **Goal**: To predict the segmentation of an ischemic stroke lesion after it has stabilized (90 days after the stroke) using only MRI data collected at the time of the stroke.

Data Description

- MRI sequences for 75 patients (training/test = 43/32 patients), the data was provided as part of the ISLES 2017 challenge;
- 6 different maps/channels (previously skull stripped, anonymized and co-registered):
  - Apparent Diffusion Coefficient (ADC);
  - Cerebral Blood Volume (CBV);
  - Cerebral Blood Flow (CBF);
  - Mean Transit Time (MTT);
  - Time To Peak (TTP) - concentration of contrast agent;
  - Time at which the residue function reaches its maximum value (Tmax).
- Ground truth provided by manually outlined follow-up scans acquired when the stroke lesion had stabilised.

Pre-Processing

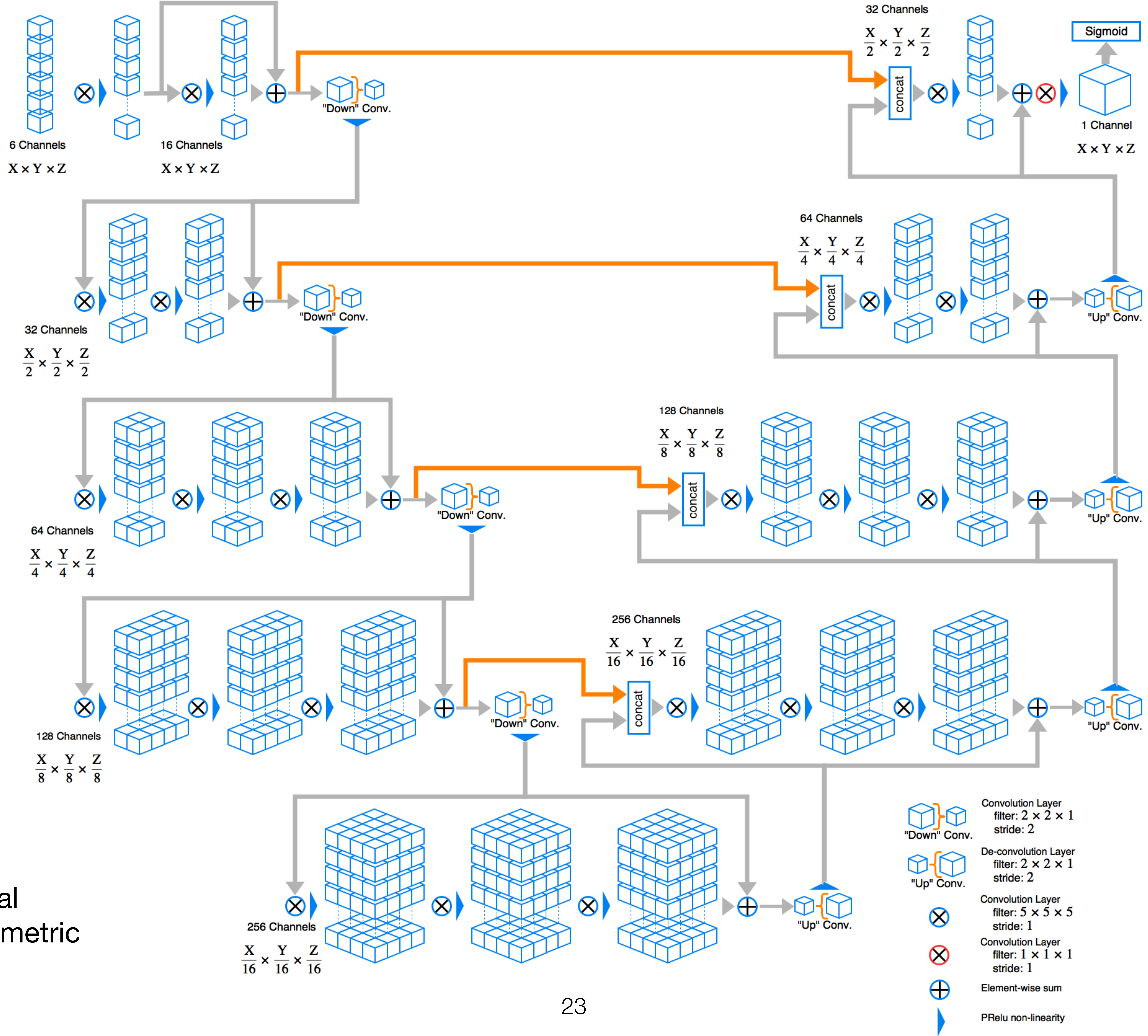
- Individually scale each channel to have zero mean and unit variance;
- Stack the 6 channels to create a 4 dimensional image:

$$width \times height \times depth \times 6$$

- No re-scaling or re-sampling of the images was performed meaning the network was fed images of different sizes and spatial resolutions (competition driven choice - maybe not ideal).

Methods (1)

- Fully Convolutional Neural Network [Long et al.] with a “V-Net” architecture [Milletari et al.];
- Trained in Tensorflow with asynchronous Stochastic Gradient Descent (SGD) in Google Cloud ML Engine:
  - This allows to take advantage of multiple machines with GPUs which considerably speed up training and the iterative process of building a neural network.
- Code available at https://github.com/MiguelMonteiro/ISLES2017



Milletari F., Navab N. & Ahmadi S. A. (2016). V-Net: Fully convolutional neural networks for volumetric image segmentation.

Methods (3)

- Custom Loss function: sum of the cross-entropy loss with the “soft” Dice loss
  - N is the number of voxels:
  - y_n is the correct label;
  - $\hat{y}_n$ is the predicted probability.

$$L_{\text{cross-entropy}} = -\frac{1}{N}\sum_{N}^{n=1} y_n \log(\hat{y}_n) + (1 - y_n)\log(1 - \hat{y}_n)$$

$$L_{\text{Dice}} = -\frac{2\sum_{N}^{n=1} y_n \hat{y}_n}{\sum_{N}^{n=1} y_n + \sum_{N}^{n=1} \hat{y}_n}$$

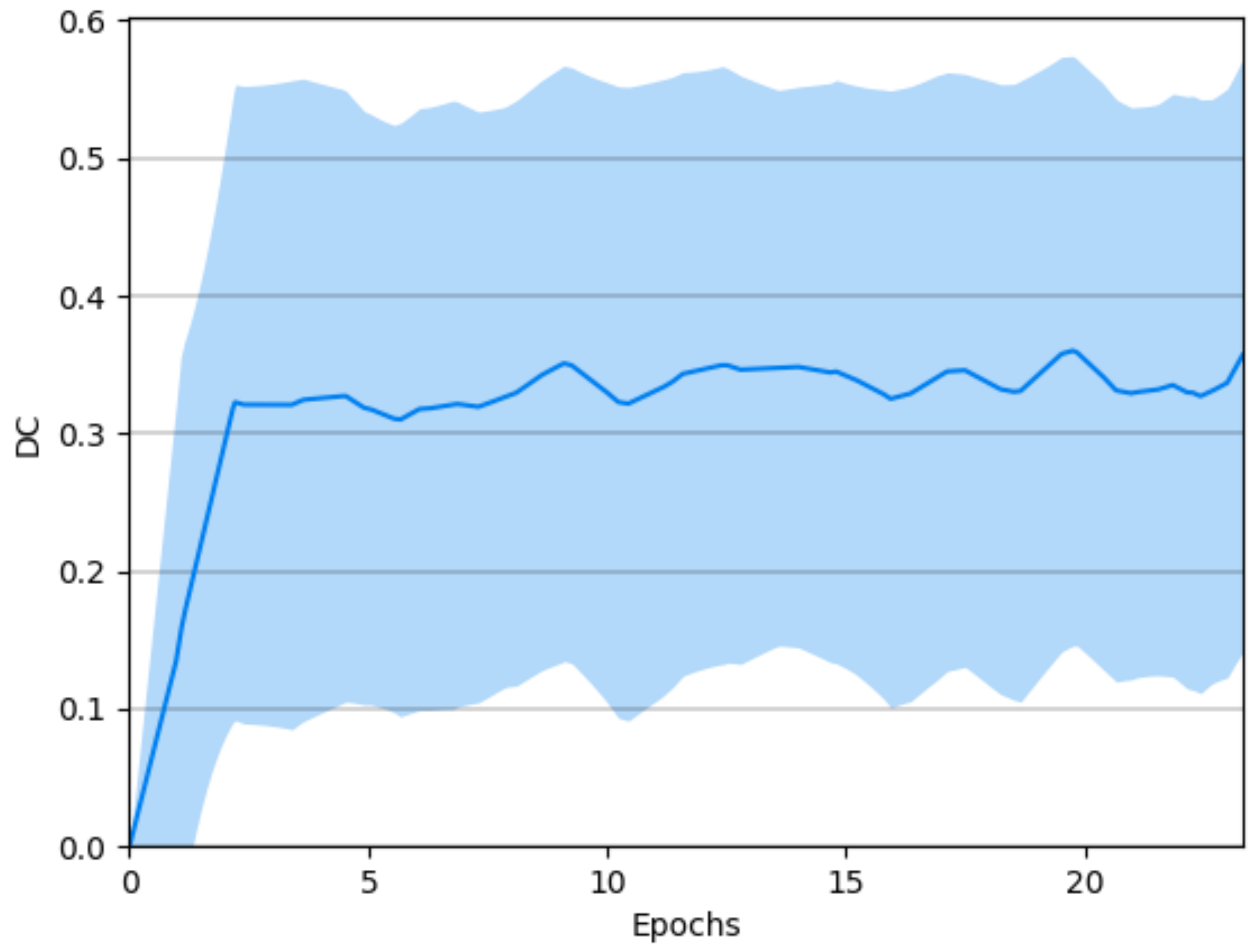
$$L = L_{\text{cross-entropy}} + L_{Dice}$$

Evaluation (1)

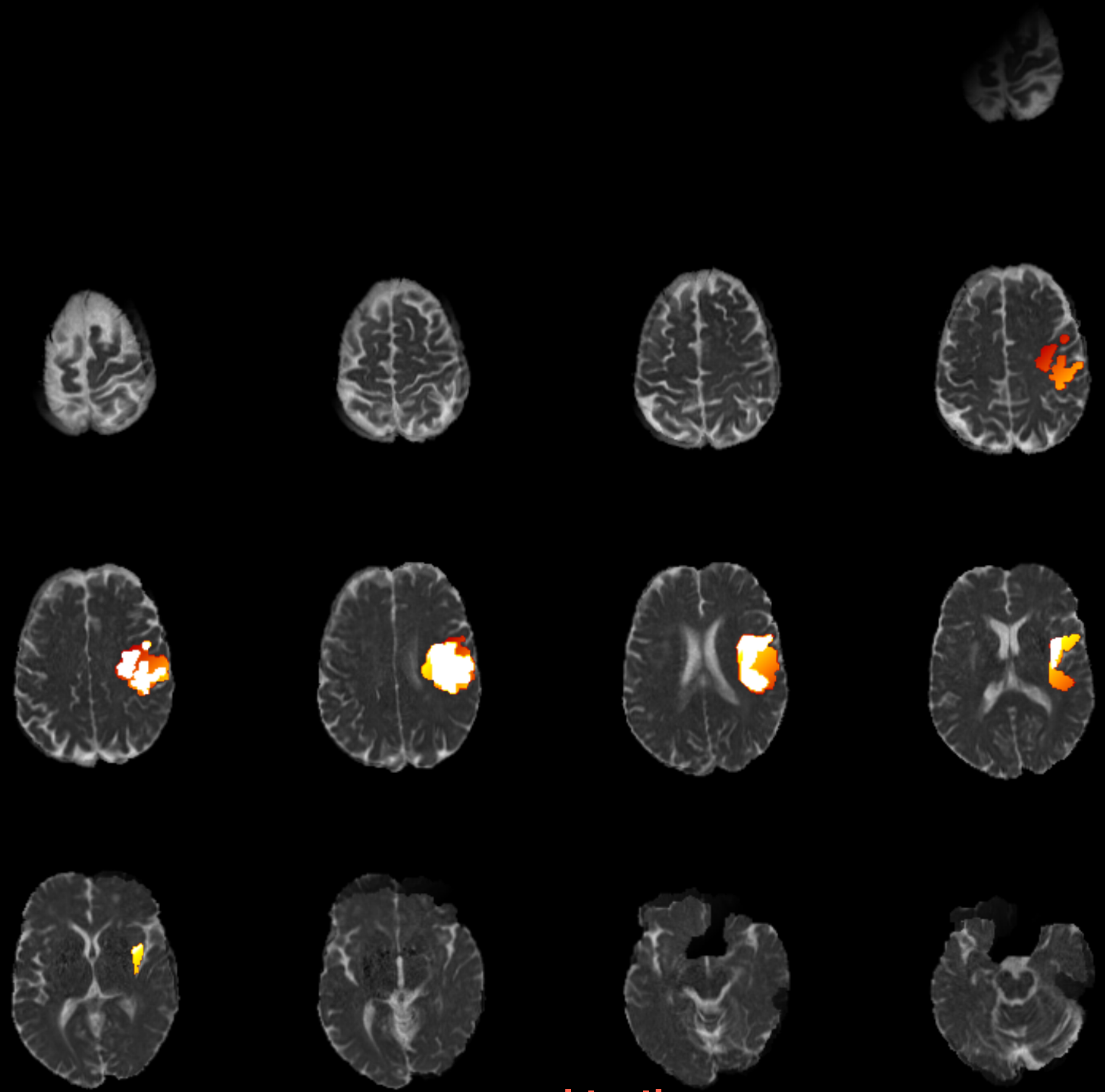
- Evaluation Metrics:
  - Dice Coefficient (DC) - overlap between ground truth and predicted volumes;
  - Hausdorff Distance - measures outliers;
  - Average Symmetric Surface Distance (ASSD) - measures overall surface deformity.
- Training with 5-fold cross-validation for hyper-parameter search.

Evaluation (2)

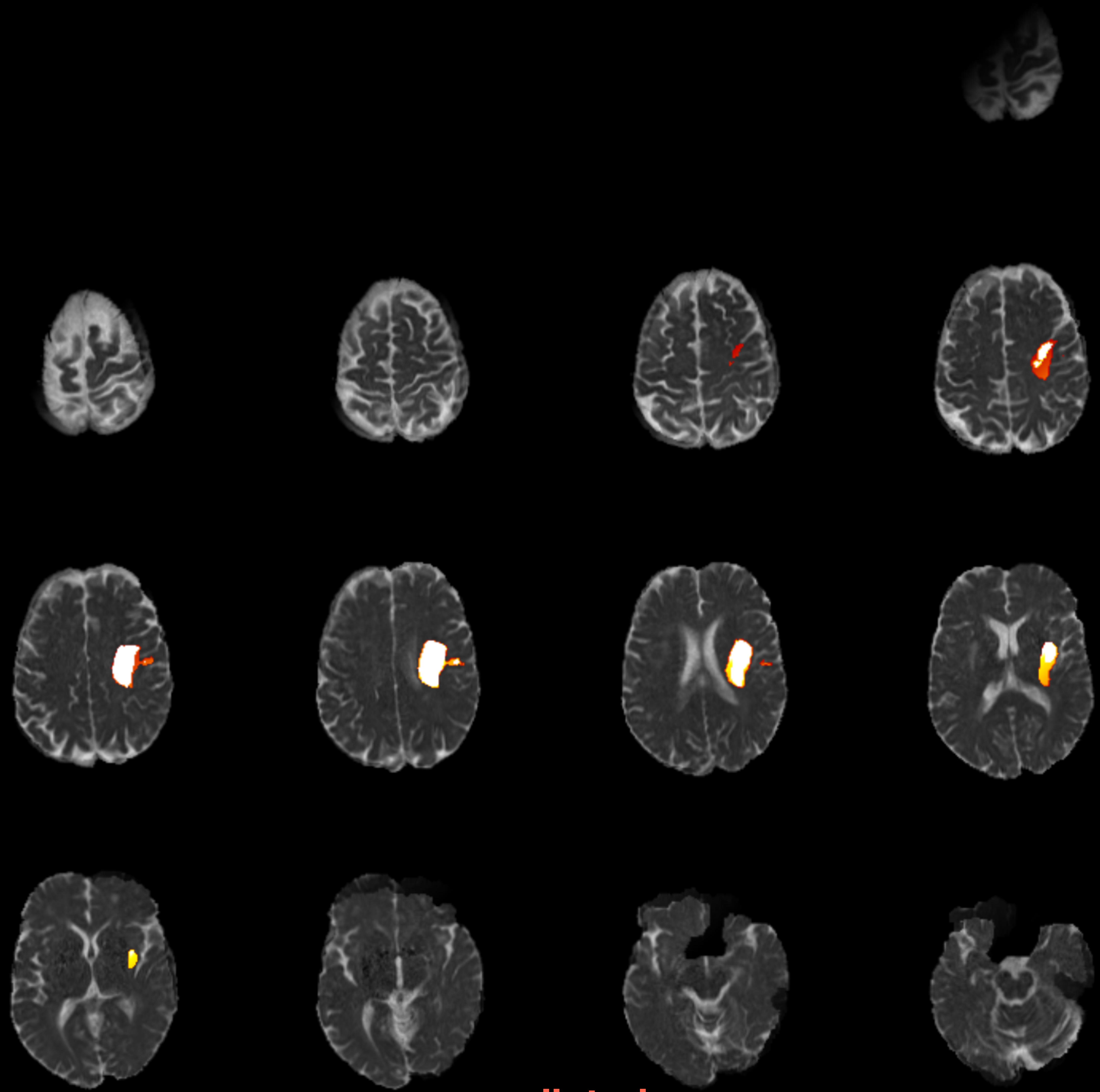
	DC	HD	ASSD
Proposed	$\mathbf{0.357 \pm 0.216}$	$\mathbf{30.823 \pm 18.512}$	$\mathbf{4.426 \pm 3.546}$
Cross-Entropy Loss	0.179 ± 0.204	54.383 ± 61.122	35.265 ± 67.517
Dice-Loss	0.000 ± 0.000	190.512 ± 42.516	190.512 ± 42.516



Median case (DC=42%)

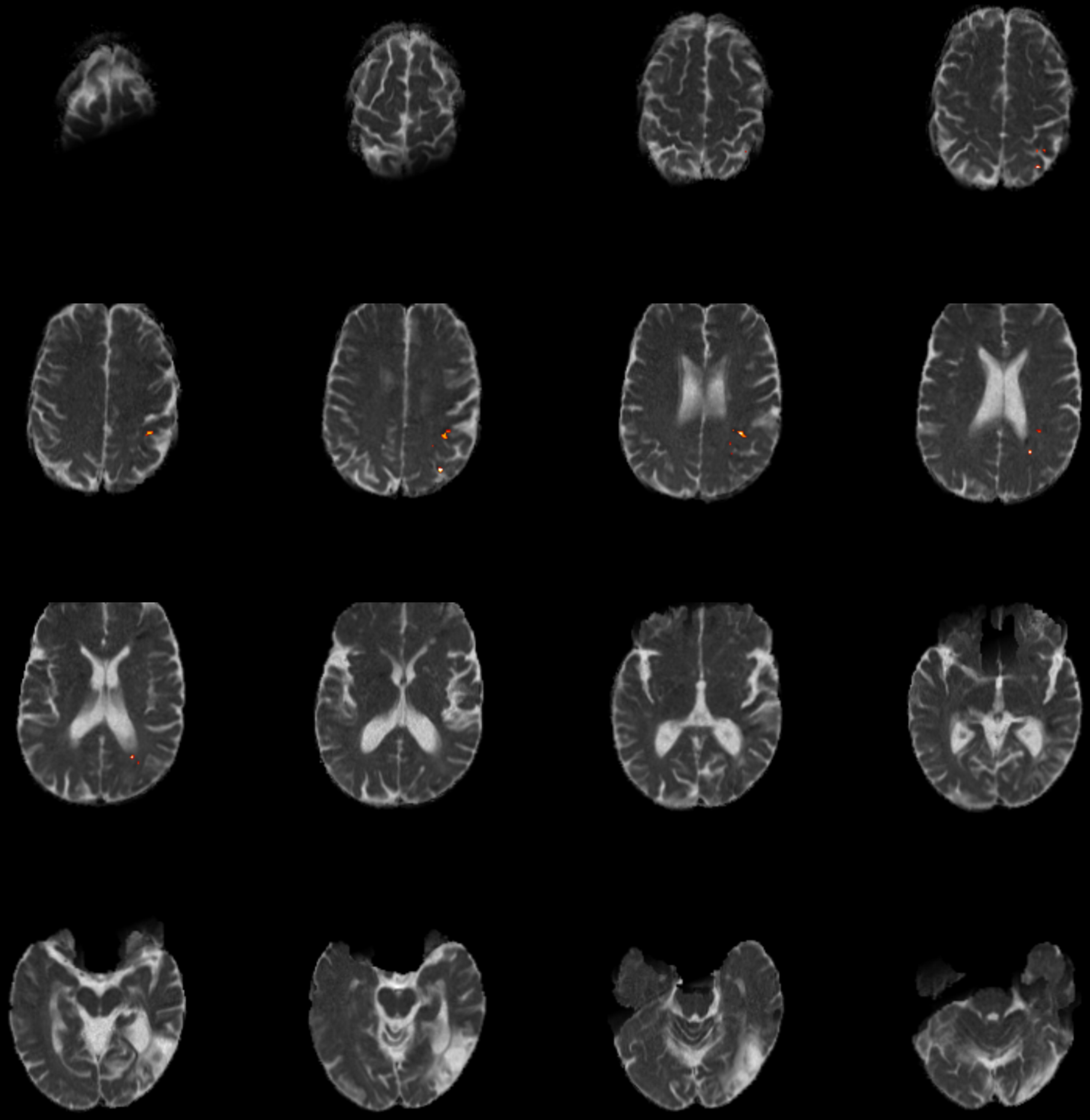


ground truth

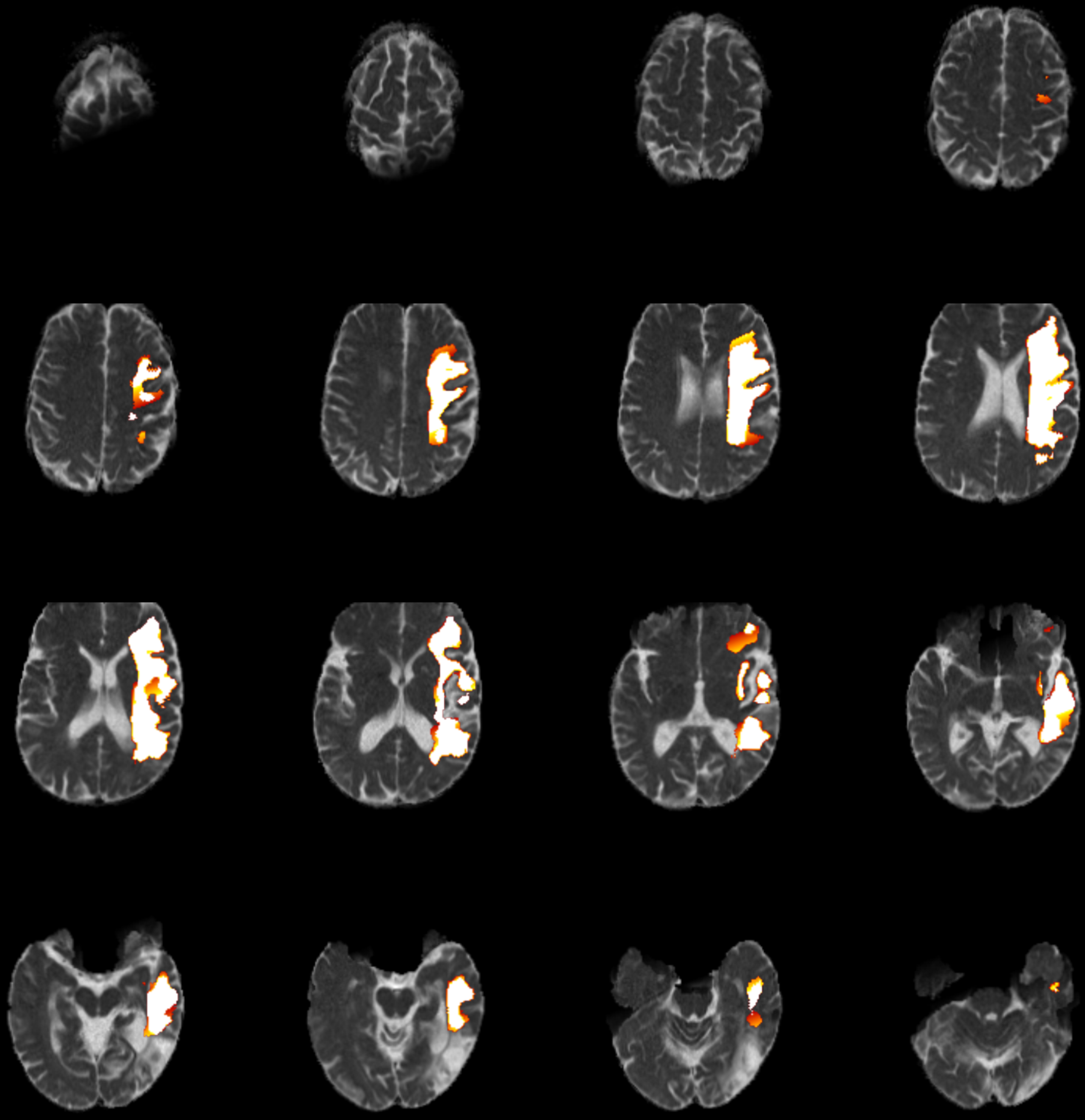


predicted

Worst case (DC=0%)

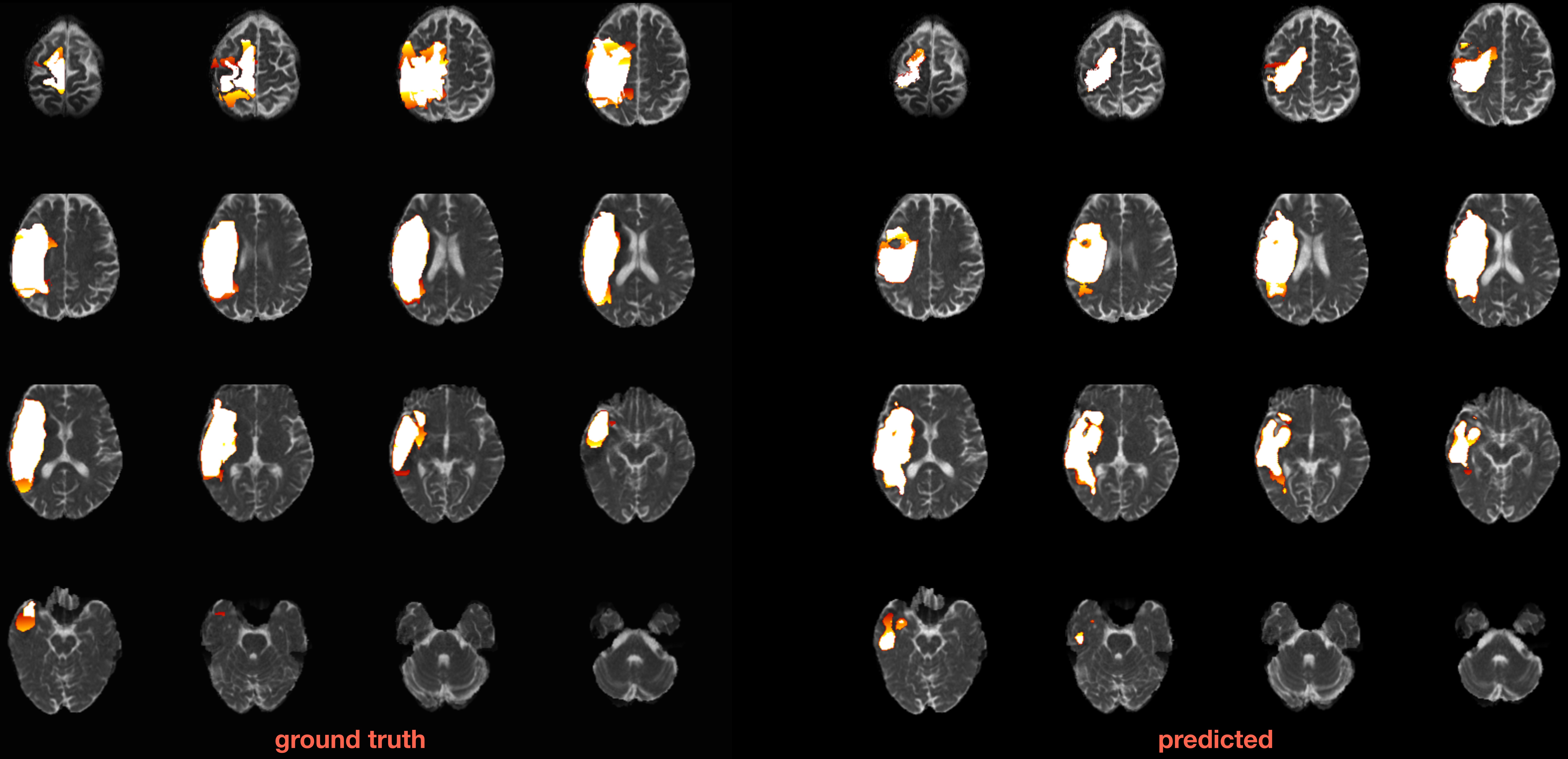


ground truth



predicted

Best case (DC=73%)



Future Challenges

- Incorporating tabular data such as the treatment information into the voxel-wise prediction;
- Better metrics to evaluate the prediction result.

Other applications

- Segmentation (organs, lesions, tumours etc…);
- Object Detection and classification;
- Image reconstruction;
- Image registration;
- MRI Pulse Sequence Generation;
- Image Synthesis for label generation;
- and more…

Questions?