Automatic Thoracic CT Image Segmentation using Deep Convolutional Neural Networks

Xiao Han, Ph.D.

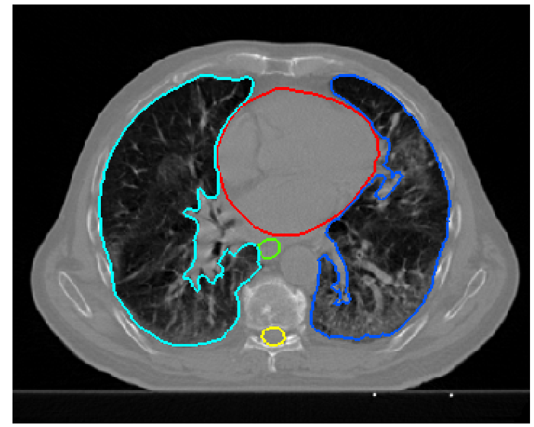
Elekta

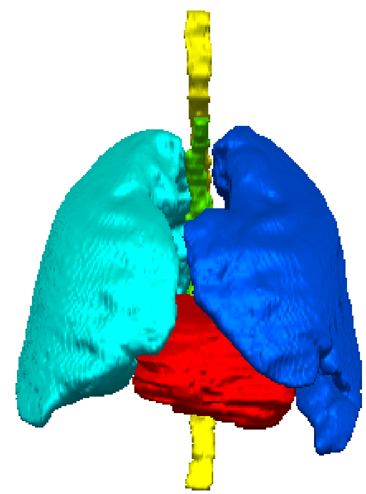
Outline

- Background
- Brief Introduction to DCNN
- Method
- Results

Structure Segmentation For RTP

- A prerequisite for radiotherapy treatment planning
- Manual Segmentation
  - Tedious and time-consuming
  - Suffers from large inter- and intra- rater variability
- Automatic Segmentation Methods
  - Atlas-based methods have been popular
  - Deep learning (DL) methods very likely be the method of choice for future



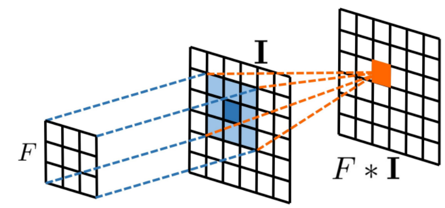


Deep Convolutional Neural Networks (DCNN)

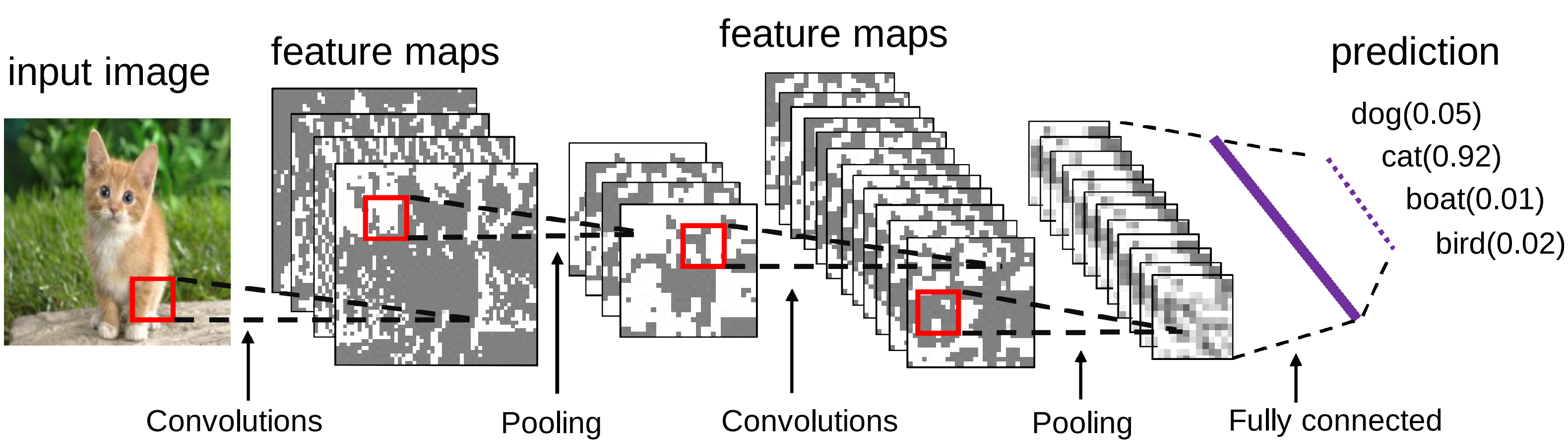
– well suited for image related problems

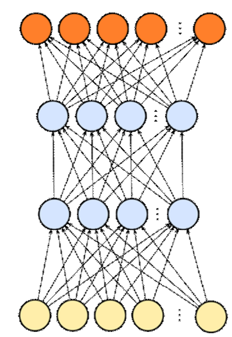
[LeCun’1998, Hinton’2006]

- Take advantage of spatial structures of image data
  - Local receptive fields
  - Shared weights
  - Multi-scale, hierarchical feature learning



convolution

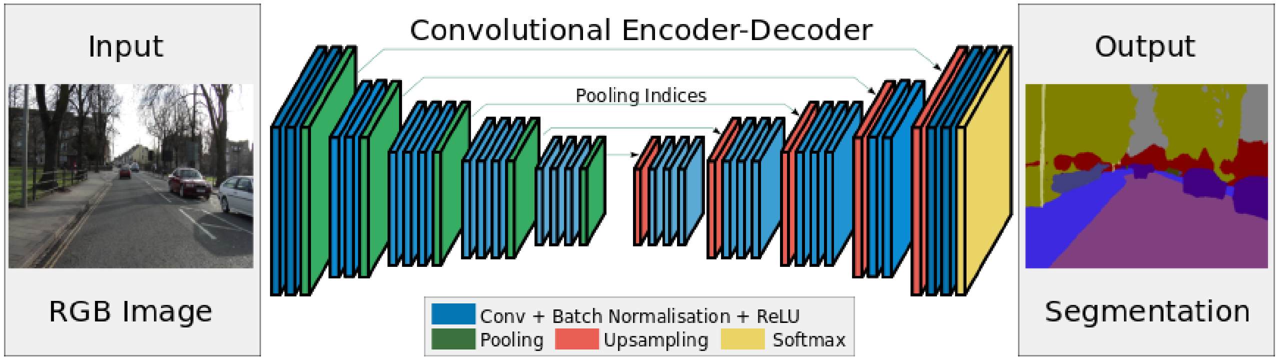




Elekta

Encoder-Decoder DCNNs

– for semantic image segmentation (pixel-by-pixel labeling)



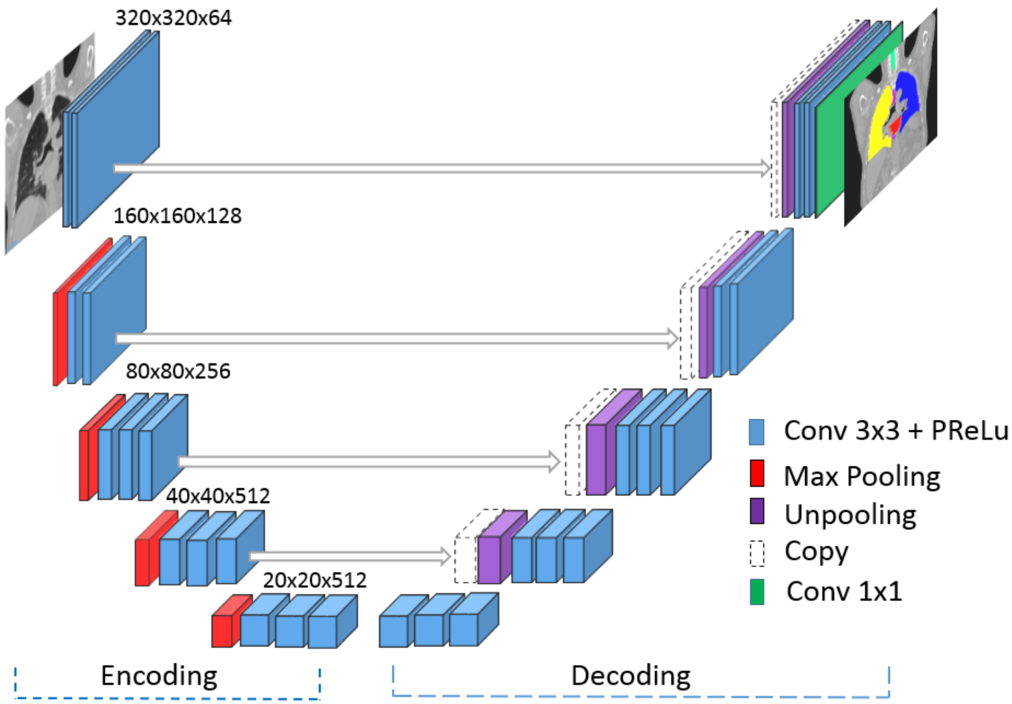
[Badrinarayanan'2015]

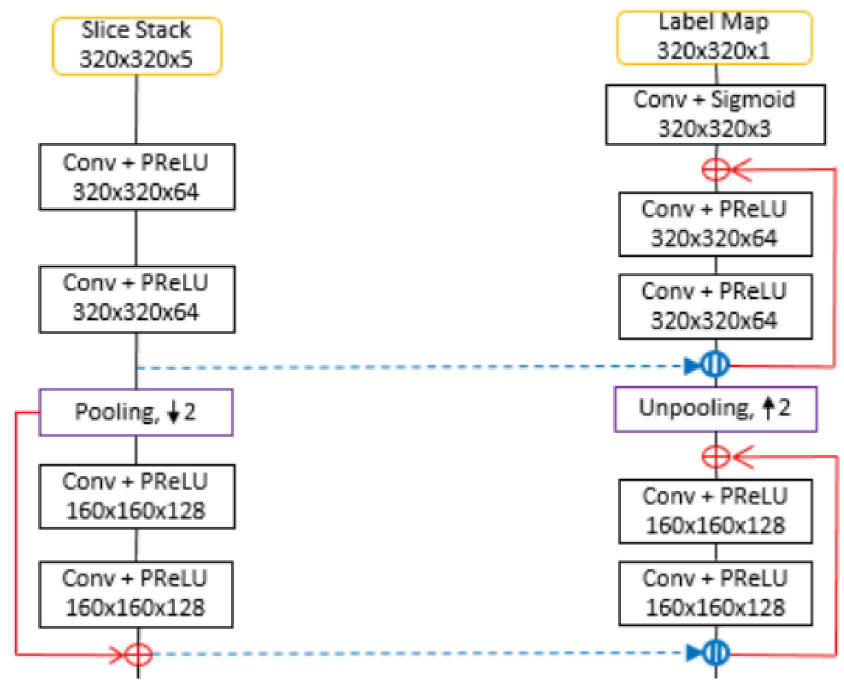
- FCN (2015, Long et al, Univ. of California, Berkeley)
- DeconvNet (2015, Noh et al, POSTECH, Korea)
- U-Net (2015, Ronneberger et al, Univ. of Freiburg)
- SegNet (2015, Badrinarayanan et al, Univ. of Cambridge)

Elekta

DCNN Architecture For Thoracic Image Segmentation

- A modified U-Net, added with *residue connections* from ResNet [He'2015]





Residue-UNet

27 Convolutional Layers
34.9 Million Parameters

Elekta

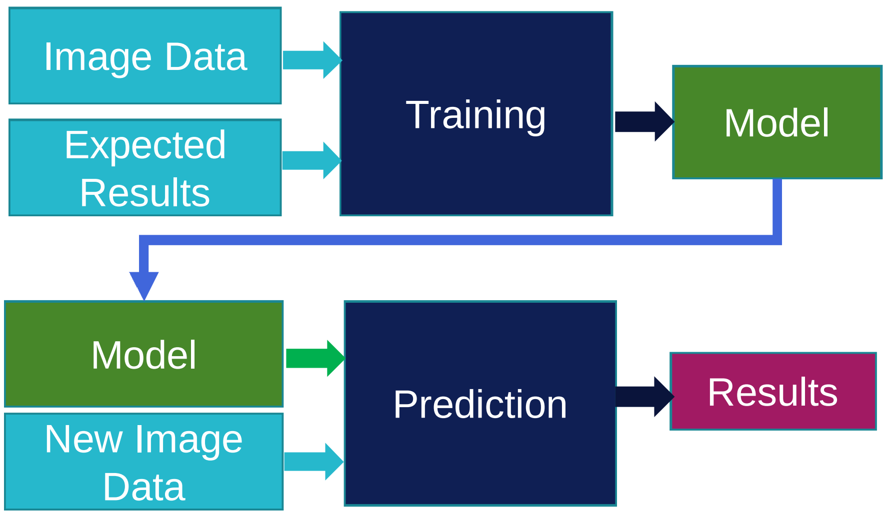
A Two-Model Scheme

for better computation efficiency

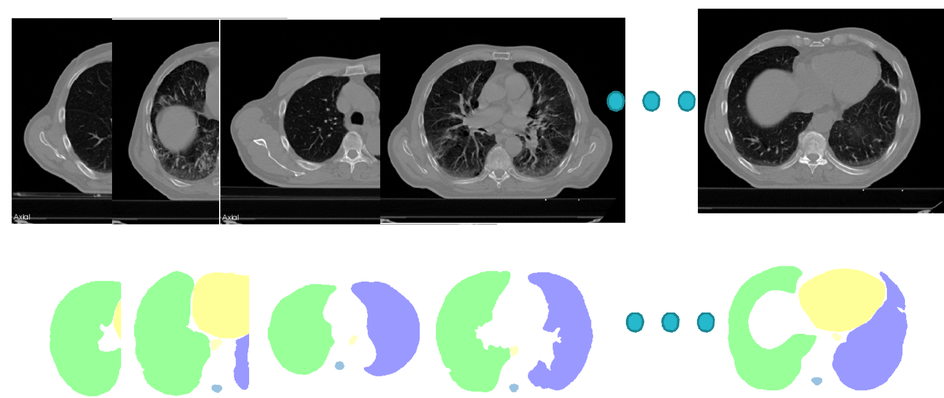
- A 2.5D model for large structures – the lungs
  - Input is 5 adjacent axial slices of size 360x360, output is 2D segmentation map corresponding to the center slice
  - Fast, but accuracy limited for thin, elongated structures – e.g. esophagus
- A 3D model for small structures – heart, esophagus, spinal cord
  - Input is a 128x128x32 sub-volume, output is 3D segmentation map of same size
  - Very slow if applied to process a whole 3D volume
- The 2.5D model is first applied, the result is used to automatically define a smaller ROI, within which the 3D model is applied

DCNN Workflow

- Two stages



- Training data



$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=0}^{5} w_i^c y_i^c \log P_i^c(\theta)$$

cross-entropy loss

$$\theta = \{W_1, b_1, W_2, b_2, \cdots\}$$

Elekta

Some Implementation Details

- Models implemented using *Caffe* package, trained using stochastic gradient descent with momentum algorithm
- Data augmentation very important during training when training data are limited
  - Apply random deformations to each training sample (image and label map) on the fly
- Hardware: PC with a NVIDIA Titan X GPU with 12GB memory
- Training a model from scratch takes about 3 days
- Applying the two trained models to process a new 3D image takes ~30 seconds each

Cross Validation Using AAPM Challenge Data

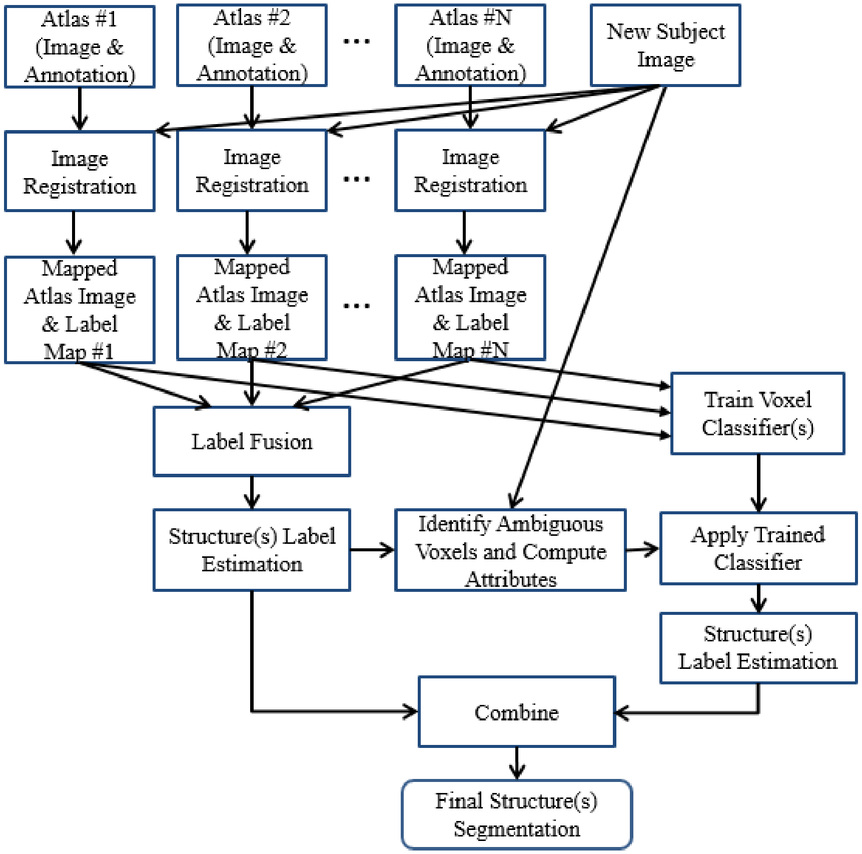
- 36 patients, 12 from each of three institutions
  - Randomly select 3 subjects from each institution as test data
  - Using the remaining 27 (9x3) as training data for DCNN model training
  - DCNN results on 9 test subjects are compared with ground truth (manual) segmentation

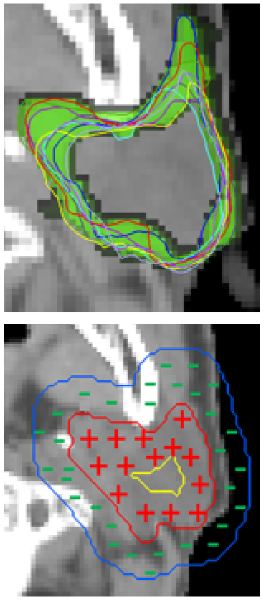
Comparison is also made with respect to an ABAS method we previously developed

Comparing Method – Atlas-based Auto-segmentation

[Han'MLMI2013]

- Multi-atlas atlas-based auto-segmentation using online RF (Random Forest) enhanced label fusion
- Using the 9 training subjects as atlases for the 3 test subjects from the same institution

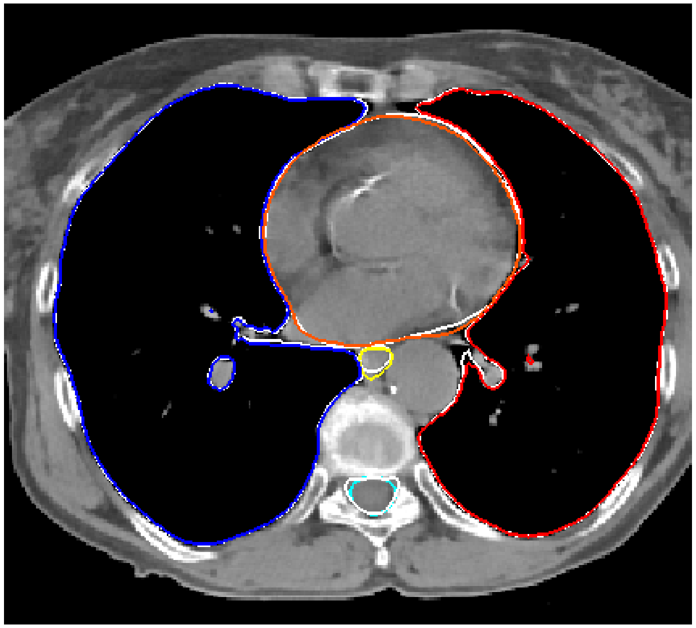
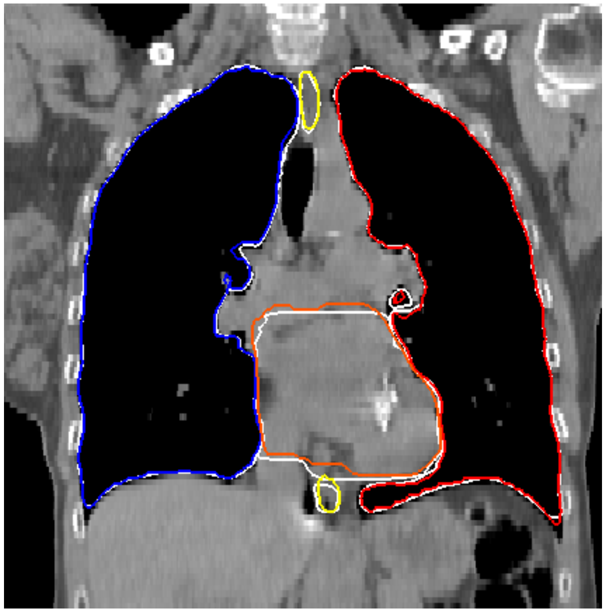
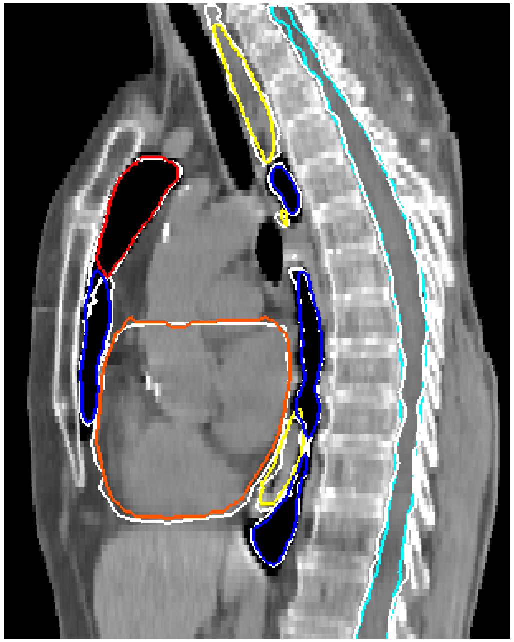




Elekta

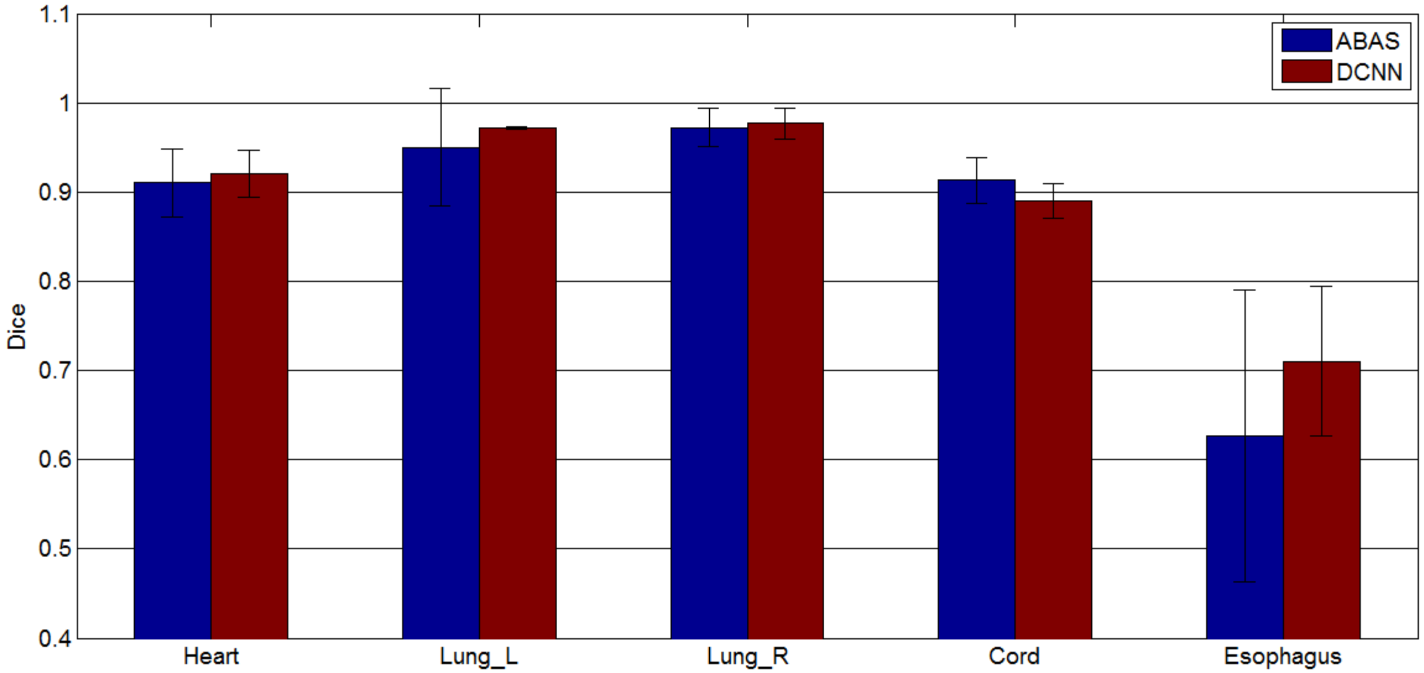
Auto-segmentation using the DCNN method

A sample result

Color: DCNN result
White: manual segmentation

Comparison – DCNN vs ABAS



- DCNN more accurate for most structures
- DCNN only takes ~1m, atlas-based takes ~6 minutes

Elekta

Discussion

Advantages of Deep Learning

- DCNN method produces fast and accurate auto-segmentation results even with limited training data
- Accuracy should improve further with more training data
  - DL greatly benefits from big data due to high model capacity
- DCNN can easily accommodate large amount of training data
  - Only training time may increase, applying the model takes the same time
  - Computation time for ABAS increases with number of atlases

Elekta

Thank you

Email: xiao.han@Elekta.com

Elekta

A 3D U-Net based thoracic segmentation framework using cropped images

Xue Feng[1], Kun Qing[2], Craig H. Meyer[1,2]

[1]Biomedical Engineering, University of Virginia, Charlottesville, VA

[2]Radiology, University of Virginia, Charlottesville, VA

Introduction: Deep Learning

- Deep learning models have shown their superiority in classification, object detection and (medical) image segmentation

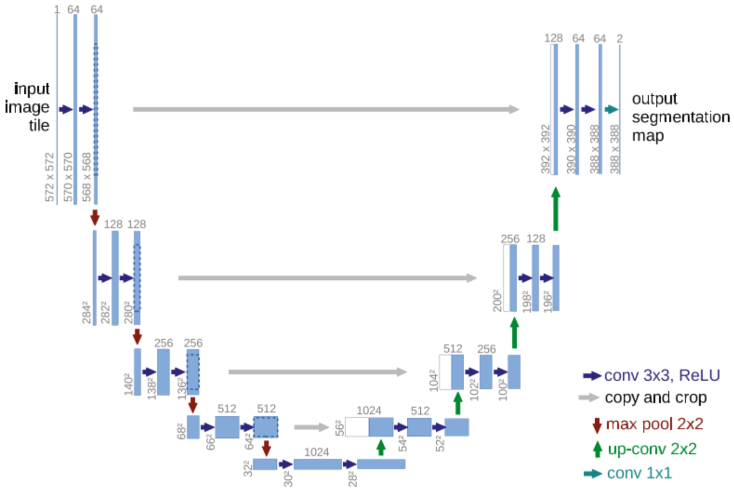
- Models:
  - Patch-based CNN model: predicts the class label of the center pixel
  - U-Net: fully convolutional networks trained end-to-end [1, 2]

- Data:
  - The more, the better
  - Key: how well can the training data represent the task distribution?

[1] Ronneberger et al., *arXiv:1505.04597* (2015)
[2] Cicek et al., *arXiv:1606.06650* (2016)

Methods: General Model

- U-Net is a better suited model
  - Efficiency
  - Captures information of intensity, position, shape, etc.
- 3D vs. 2D (?)
  - Pros:
    - Additional input information (slice)
    - Consistent output
  - Cons:
    - Less training data
    - Memory (hard to fit 200 * 512 * 512 into GPU)
- We choose 3D U-Net and deal with the challenges



Methods: Image Pre-processing

- Why? If more data is difficult, it's better to limit the variability in the dataset

- Steps:
  - Intensity normalization (crop to -1000~600 in Hounsfield scale)
  - Unify the pixel spacing and slice thickness (resizing)
  - Crop the images to the same in-plane dimension (#slice * 512 * 512)
  - Generate masks with ROI labels (0-Background, 1-SpinalCord, 2-Lung_R, 3-Lung_L, 4-Heart, 5-Esophagus

Methods: U-Net with Cropped Images

- Challenges:
  - GPU Memory (Titan X: 12 GB) limits the size of input images
  - Due to small data set, U-Net model does not perform good when trained end-to-end (full image -> all ROI labels)

- Solution: crop the input images to separate regions containing one ROI each (organs don't overlap!)
  - Step 1: train a U-Net model on scaled images for end-to-end segmentation and extract the bounding boxes for each ROI
  - Step 2: train one U-Net model for each ROI with cropped images and combine the results (including resolve of conflicts)

Methods: Step 1

- Objective: extract bounding boxes for each ROI

- Network structure:
  - Input: scale to 72x256x256, then crop to 72x208x208 (uniform slice thickness is broken)
  - Encoding path: 72x208x208x24 -> 36x104x104x48 -> 18x52x52x96
  - Decoding path: 18x52x52x96 -> 36x104x104x48 -> 72x208x208x24
  - Loss: weighted cross entropy (background: 1.0, SpinalCord: 2.0, Lung_R: 1.0, Lung_L: 1.0, Espophagus: 3.0)

- Data augmentation
  - Random 3D translation, rotation, scaling is applied on the fly

Methods: Step 1 - Continued

- Training
  - 200 epochs (8 hours on Titan X)

- Post-processing (bounding boxes extraction):
  - Clean the contour by removing isolated regions (keep only one connected region for each ROI)
  - Transform to original shape via padding and scaling
  - Calculate the range for each ROI and extraction cropped images with slightly enlarged bounding boxes

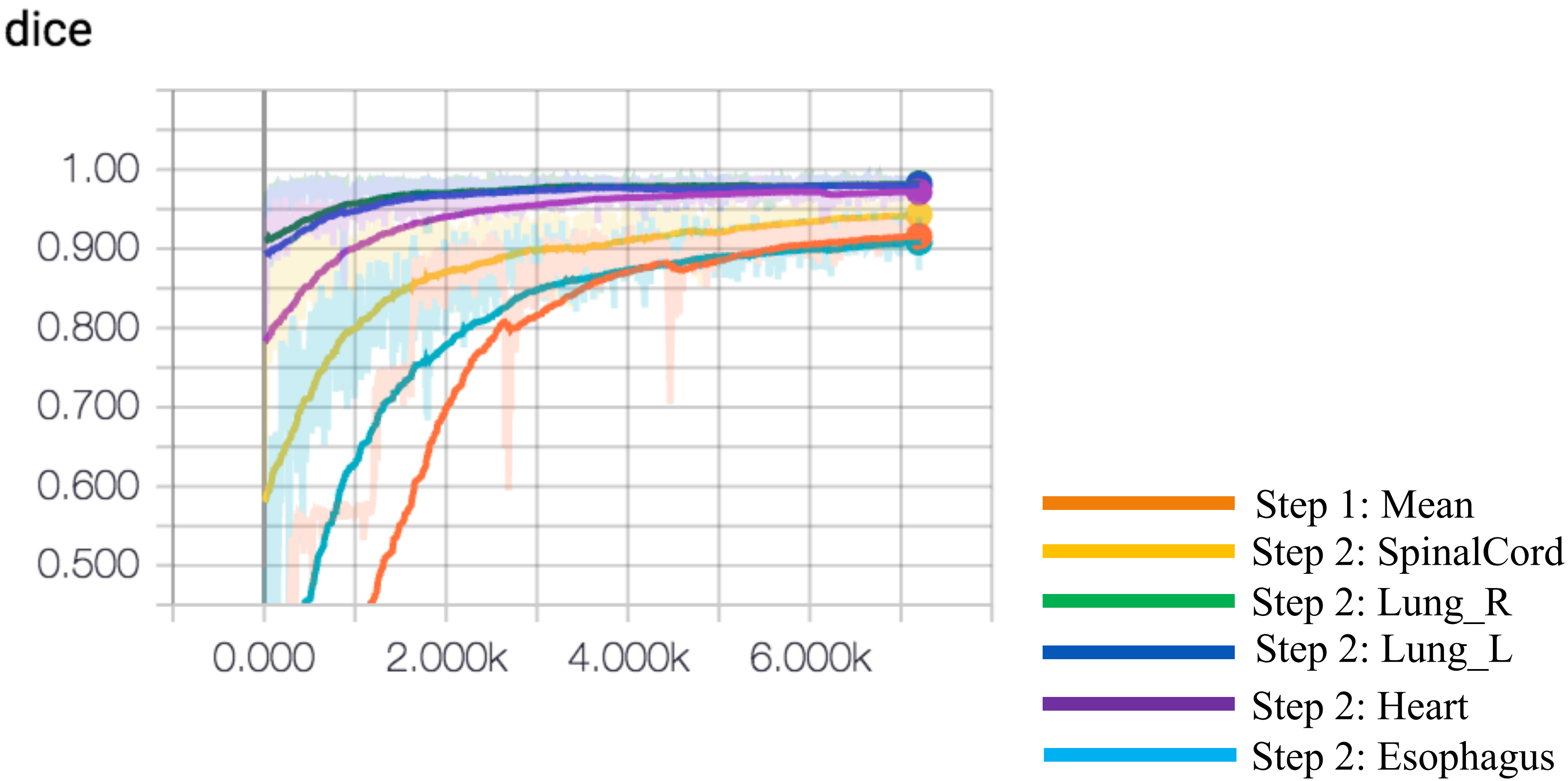
Methods: Step 2

- Objective: get the label maps for each ROI
- Network structure:
  - Input: fixed size for each ROI estimated from the mean sizes (uniform slice thickness and pixel spacing are broken)
  - Wider than step 1 model (48 filters in the first layer)
  - Output: foreground (ROI) and background
  - Loss: weighted cross-entropy for SpinalCord and Esophagus
- Data augmentation:
  - Random 3D rotation and shear, variations of bounding boxes

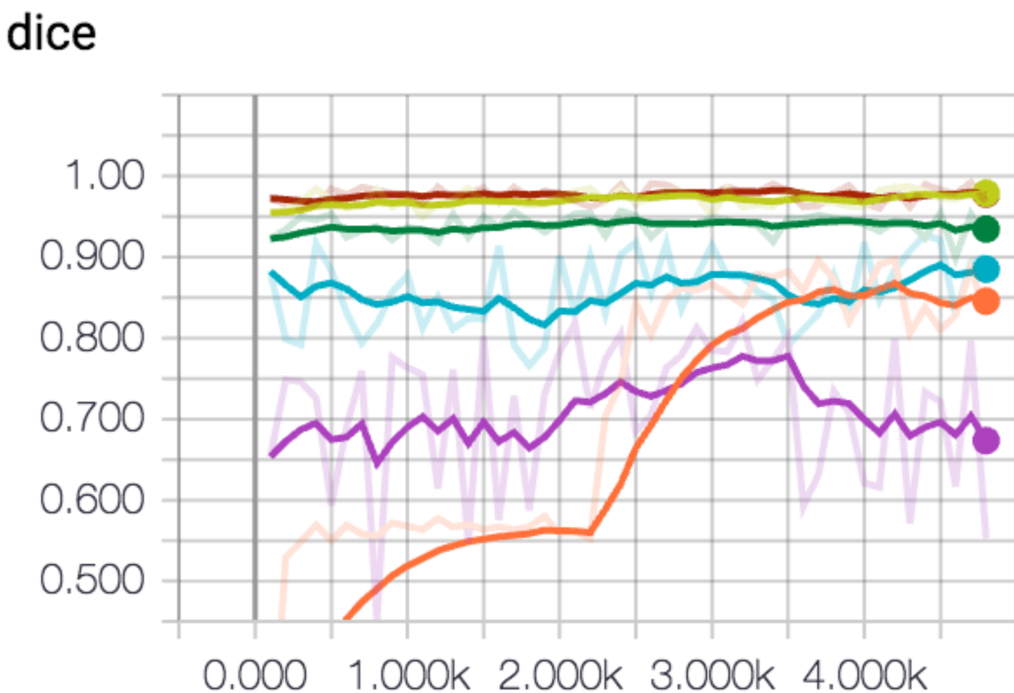
Methods: Step 2 - Continued

- Training
  - 200 epochs (7-10 hours on Titan X depending on the input size)
- Post-processing:
  - Clean the contour by removing isolated regions (keep only one connected region for each ROI)
  - Transform to original shape via padding and scaling
  - If conflicts exist (e.g. multiple models predict foreground for the same voxel), choose the result with the highest probability from softmax output

Results: Training



Results: Validation (Splitted Training)



Step 1: Mean
Step 2: SpinalCord
Step 2: Lung_R
Step 2: Lung_L
Step 2: Heart
Step 2: Esophagus

Step 1 Mean Dice: 0.84
Step 2 Mean Dice: 0.88

Results: Validation

	SpinalCord	Lung_R	Lung_L	Heart	Esophagus
Dice	0.91	0.97	0.97	0.89	0.75
Hausdorff Distance	1.73	4.83	3.29	13.40	11.45
Average Distance	0.59	1.05	0.82	4.11	3.58

Discussion and Conclusion

- U-Net structure fits this task well
- Intensity normalization is important
- The easier the task, the better the performance (label all ROIs from all images vs. separate foreground and background from cropped images)
- More work needs to be done for Esophagus (e.g. post processing using shape constraint)

Devil is in the Detail

Source code:

https://github.com/xf4j/aapm_thoracic_challenge

Thanks

Automatic Multi-organ Segmentation in 3D Computed Tomography

Surgical Sciences

Bruno Miguel Gomes Oliveira

Master in Biomedical Engineering

E-mail: brunooliveira@med.uminho.pt

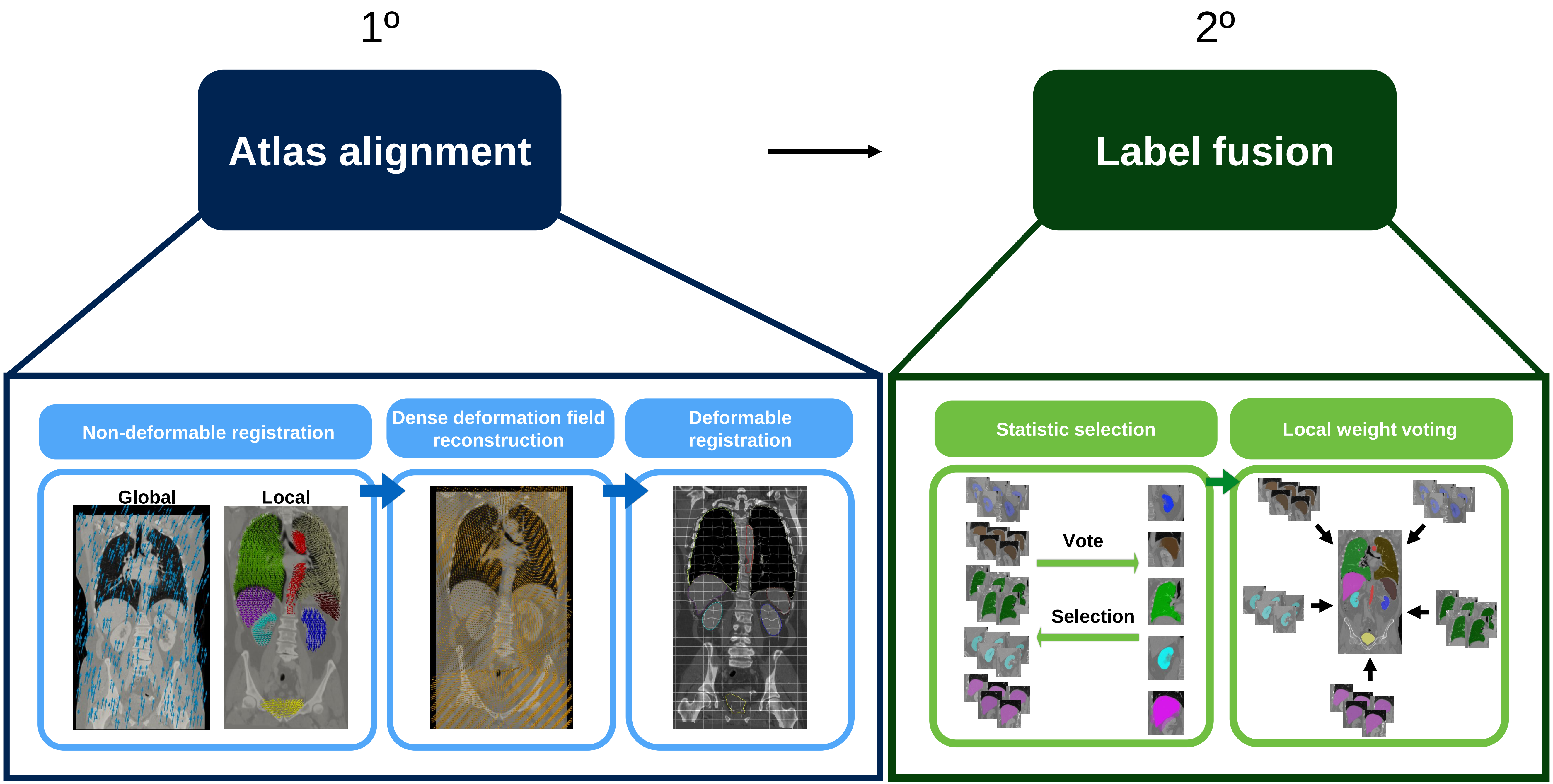
ICVS Life and Health Sciences Research Institute
Instituto de Investigação em Ciências da Vida e Saúde

July - 2017

Universidade do Minho
Escola de Ciências da Saúde

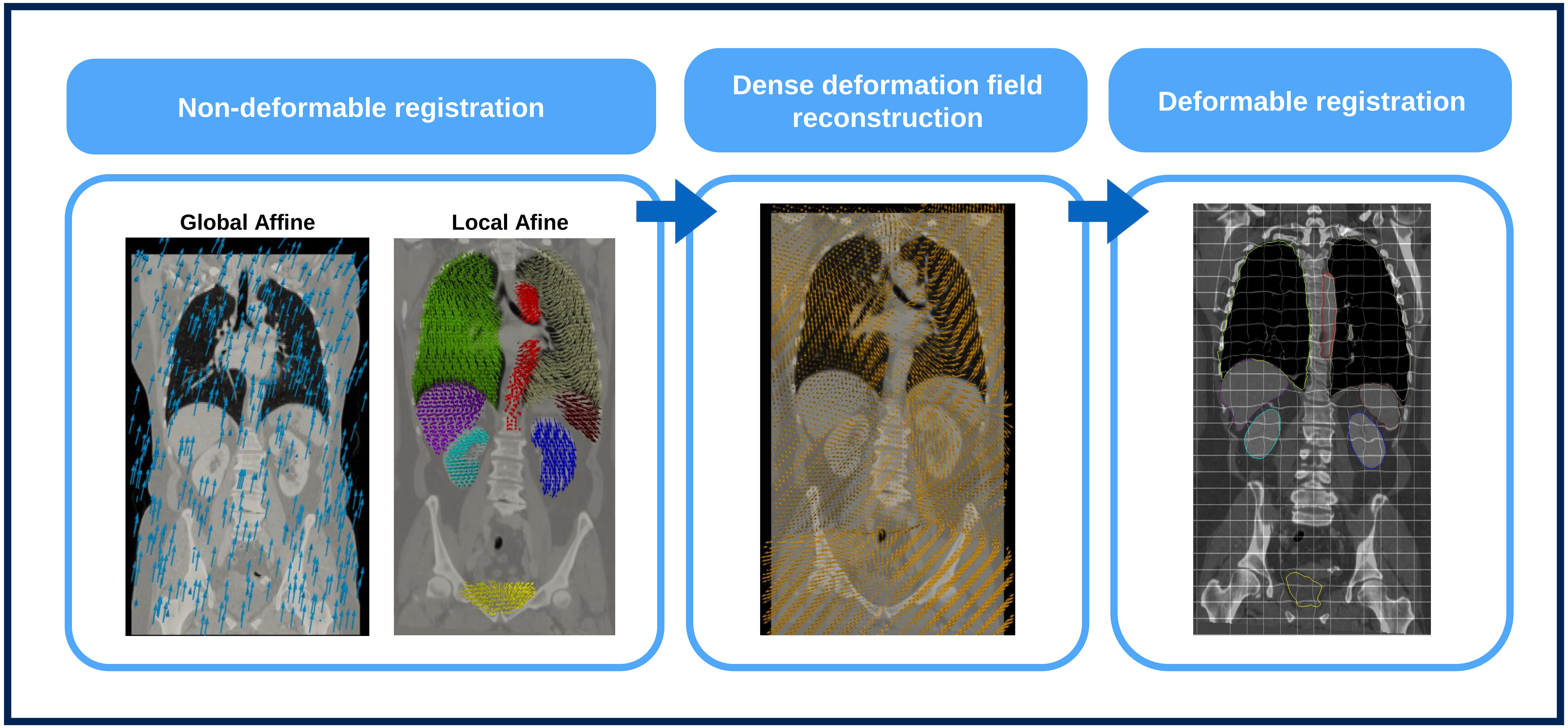
ICVS/3B's
Associate Laboratory

Methodology



Coarse-to-fine strategy

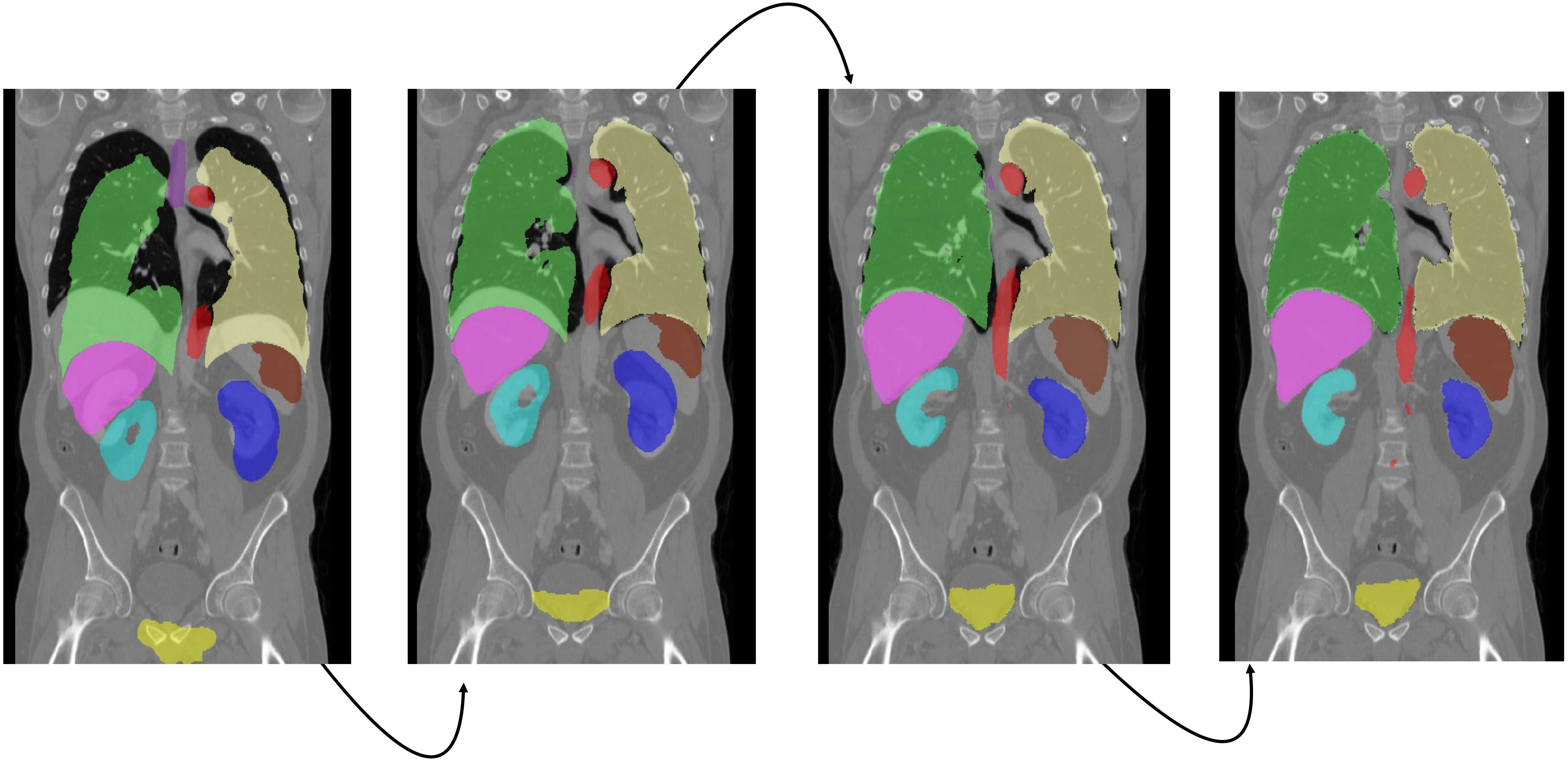
Atlas alignment



Methodology - **Atlas alignment**

Example, coronal slice from one atlas

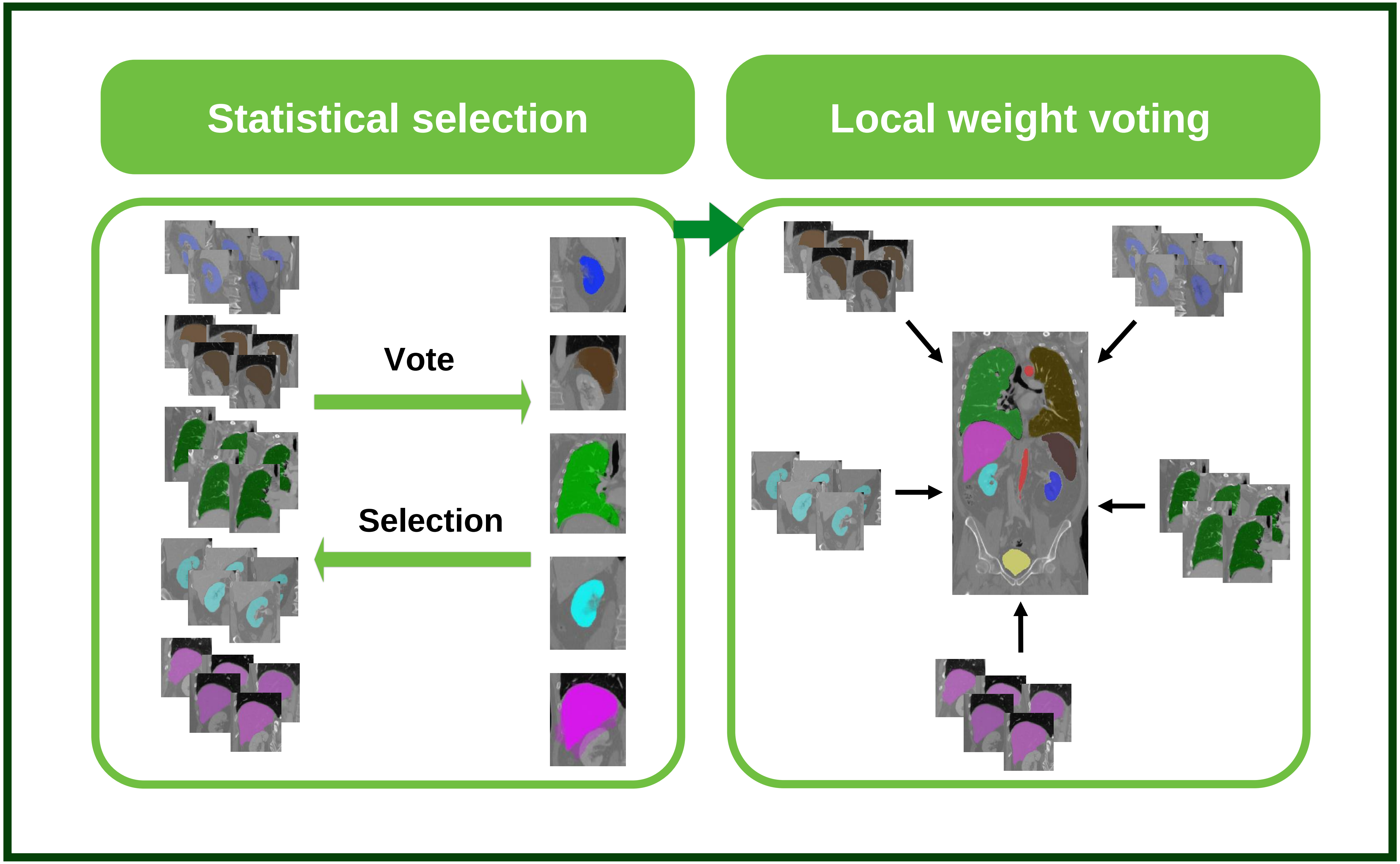
2º Local affine registration
+ Dense deformation field reconstruction



1º Global affine registration

3º Deformable registration

Label fusion



Results

AD	DICE	HD	
Esophagus	0.64	7.6	2.4
Lung_L	0.96	1.7	0.90
Lung_R	0.97	5.9	1.3
Heart	0.90	10.8	3.3
Spinal Cord	0.91	1.8	0.60

Challenge score: 46.4

Questions: brunooliveira@med.uminho.p

Circumscriptio ex machina
A step-change for auto-contouring

Paul Aljabar, Devis Peressutti, Mark Gooding

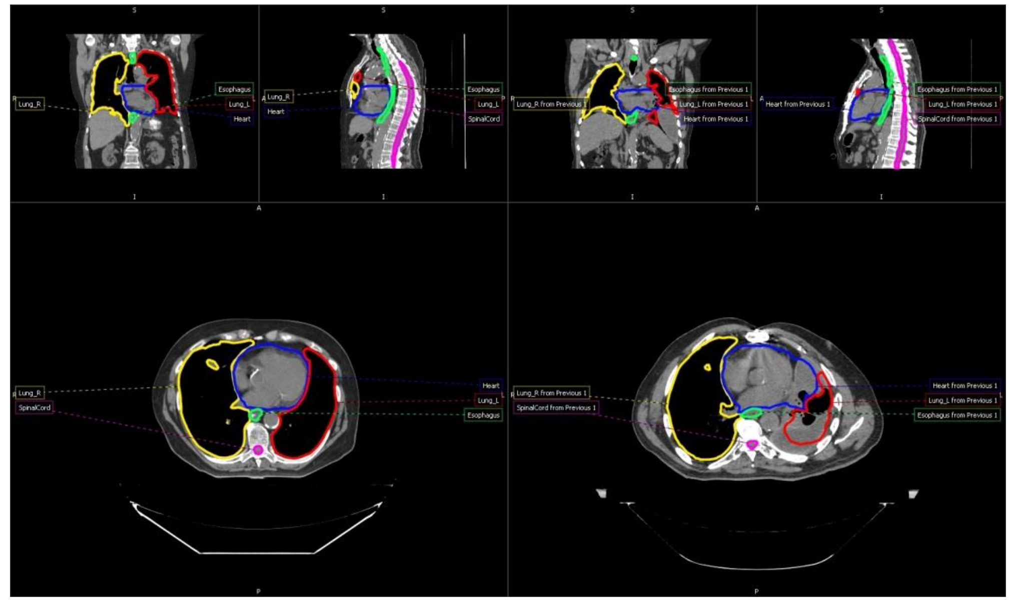
Mirada Medical Ltd.

Deus ex machina

- an unexpected power or event saving a seemingly hopeless situation

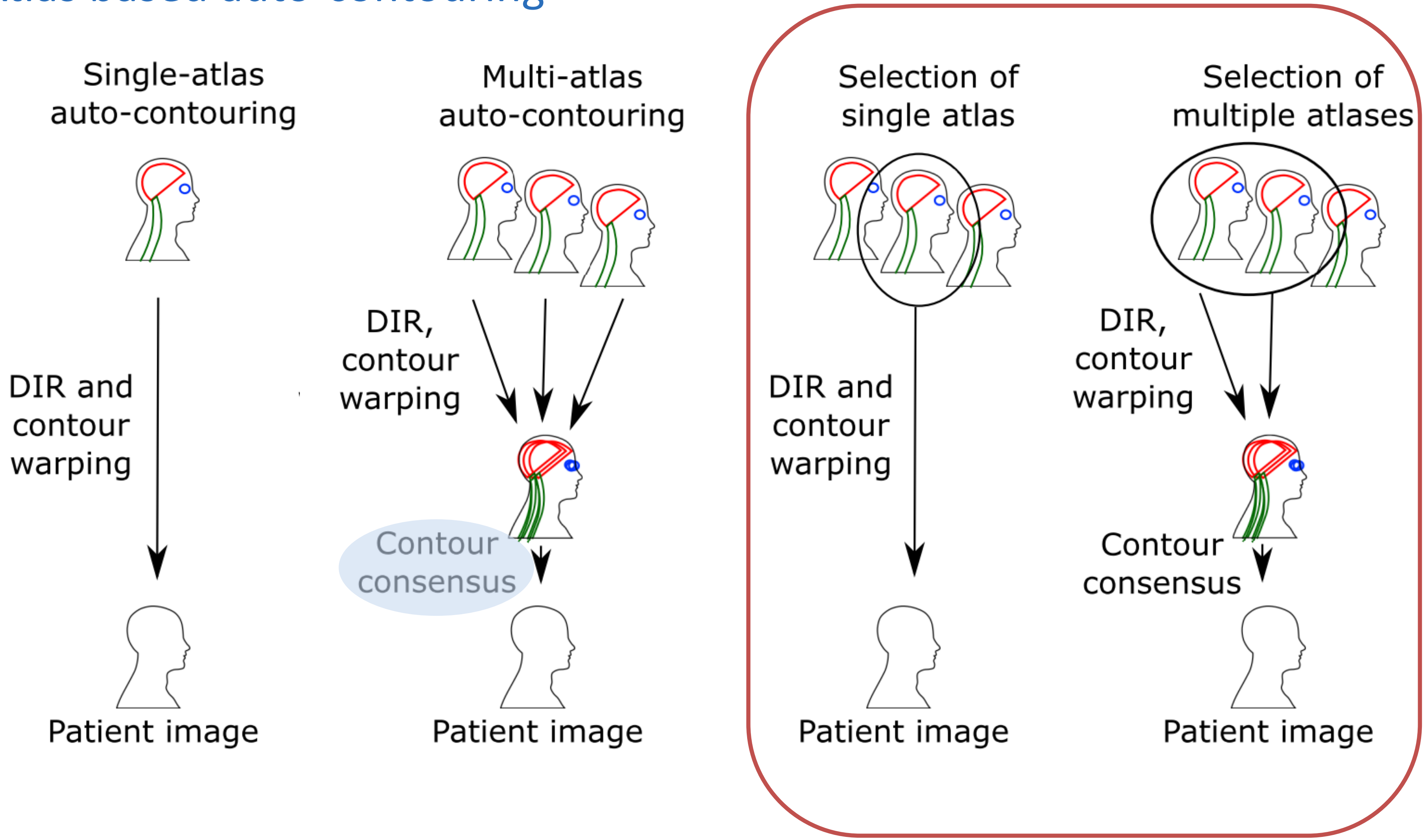
A story about auto-contouring

- Basic segmentation methods
- Prior-knowledge segmentation
  - ASM
  - Atlas



Sharp, Gregory, et al. "Vision 20/20: Perspectives on automated image segmentation for radiotherapy." *Medical physics* 41.5 (2014).

Atlas based auto-contouring



What's achievable with atlas-based auto-contouring

	Single atlas			Multi atlas		
	DSC	HD*	AD	DSC	HD*	AD
Esophagus	0.81	7.0	1.19	0.87	4.9	0.8
Heart	0.94	10.8	1.72	0.96	7.8	1.3
Lung_L	0.99	10.8	0.45	1.0	10.3	0.4
Lung_R	0.99	11.2	0.47	1.0	8.4	0.3
SpinalCord	0.93	3.6	0.50	0.95	3.0	0.4

* Calculated differently to the challenge

Extreme Value Theory

H&N results presented at ICCR 2016

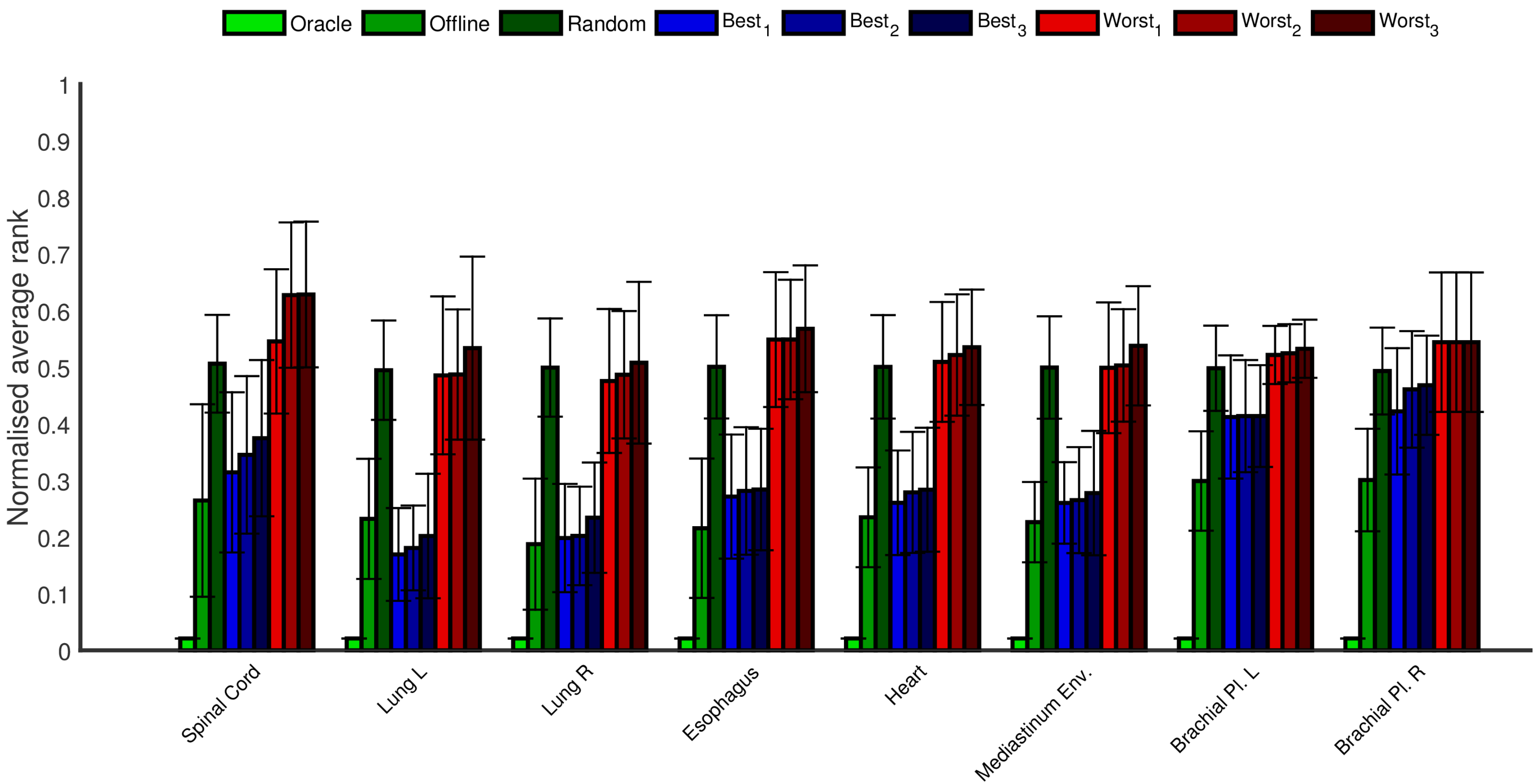
The problem

Estimates the *potential* performance for a very large atlas database…

…assuming perfect atlas selection by an oracle with fore-knowledge of the output performance

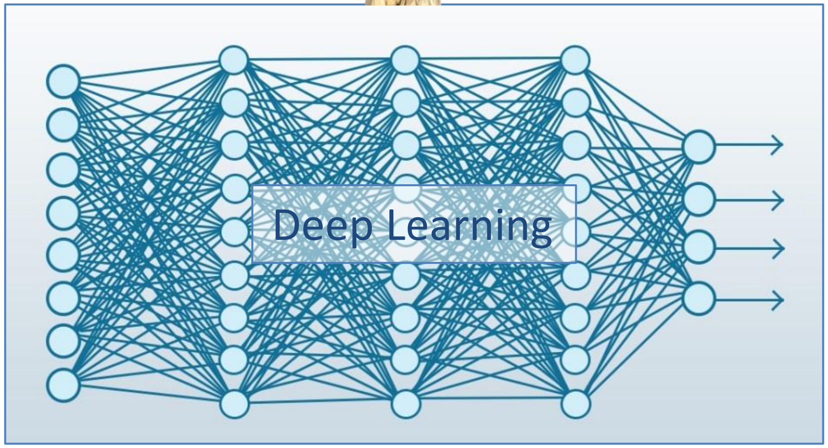
MIRADA medical

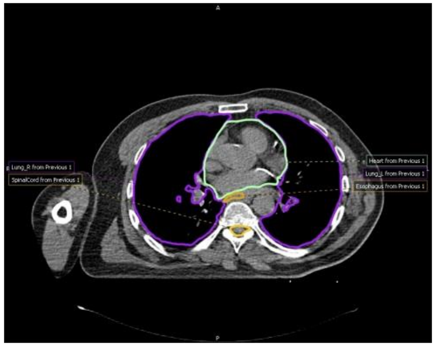
Atlas selection doesn't work so well in practice



H&N results presented at AAPM 2016

Circumscriptio ex machina





Deep learning

- "Architecture"
- Initialisation
- Epochs and Iterations
- Momentum
- Jittering

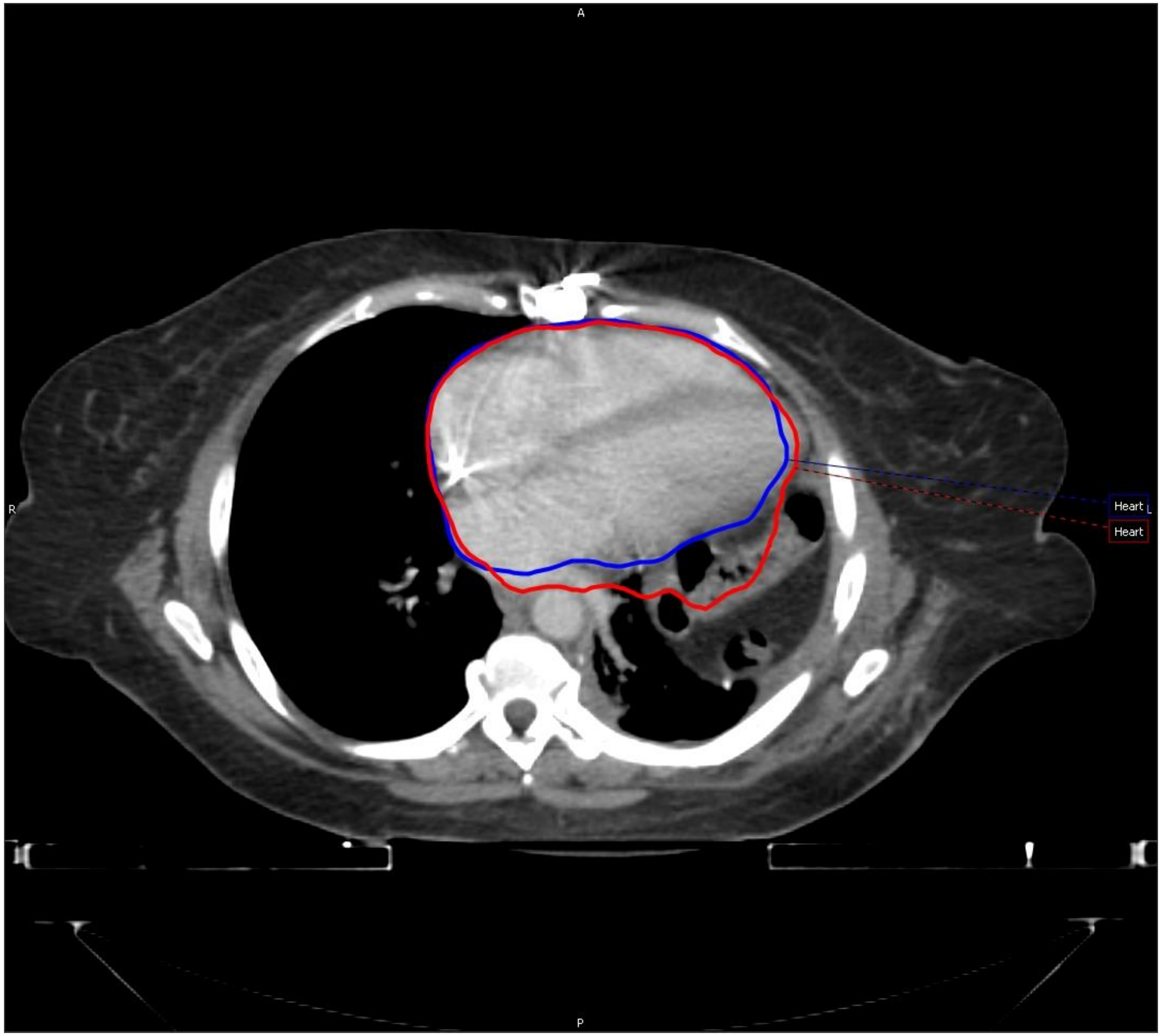
- DATA – Lots of data!
  - Pre-trained on 450 cases
  - Refined on training set

Comparing results

	Atlas contouring			Deep learning		
	DSC	HD	AD	DSC	HD	AD
Esophagus	0.56	10.2	3.0	0.76	5.9	1.8
Heart	0.89	11.6	3.9	0.90	10.8	3.6
Lung_L	0.94	6.9	1.7	0.97	2.8	0.71
Lung_R	0.96	6.3	1.3	0.97	4.2	0.91
SpinalCord	0.88	2.2	0.75	0.91	1.6	0.58

Challenge score	39.5	54.2

An example from the test data



MIRADA medical

Difficulties with the challenge

- Limited data
- Institutional variation
- Quantitative scoring

Can you tell the difference from a clinician?

www.auto-contouring.com

Mirada Medical USA, Inc.
999 18th Street Suite 2025N Denver, CO 80202
enquiries@mirada-medical.com | 877.872.2617