Robust Methods for Medical Image Segmentation

Neerav Karani
Krishna Chaitanya
Dr. Ertunc Erdil
Dr. Christian Baumgartner
Prof. Ender Konukoglu

Biomedical Image Computing Group
Computer Vision Lab
ETH Zurich

March 2021

Once upon a time...

School exam

Once upon a time...

School exam

<Statement 1>	**TRUE**	FALSE
<Statement 2>	**TRUE**	FALSE
<Statement 3>	TRUE	**FALSE**
<Statement 4>	**TRUE**	FALSE
<Statement 5>	TRUE	**FALSE**
<Statement 6>	TRUE	**FALSE**

Studying at home

Once upon a time...

School exam

<Statement 1>	**TRUE**	FALSE
<Statement 2>	**TRUE**	FALSE
<Statement 3>	TRUE	**FALSE**
<Statement 4>	**TRUE**	FALSE
<Statement 5>	TRUE	**FALSE**
<Statement 6>	TRUE	**FALSE**

Studying at home

<Statement 3>	**TRUE**	FALSE
<Statement 6>	**TRUE**	FALSE
<Statement 1>	TRUE	**FALSE**
<Statement 5>	**TRUE**	FALSE
<Statement 2>	TRUE	**FALSE**
<Statement 4>	TRUE	**FALSE**

In the exam

Neural Networks behave similarly?!

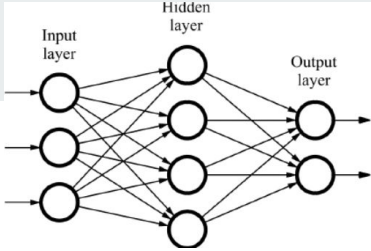
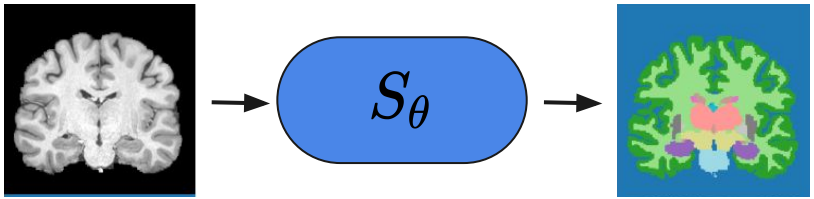
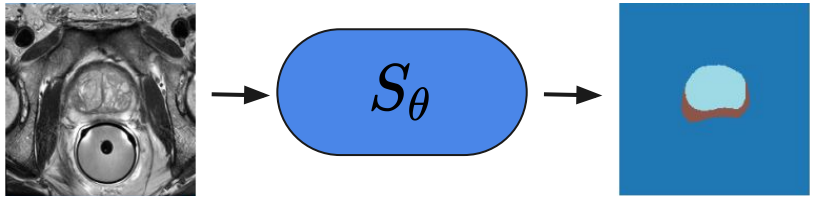


Image segmentation

Training domains





Neural Networks behave similarly?!

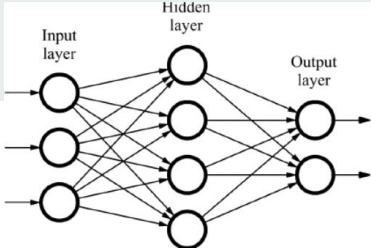
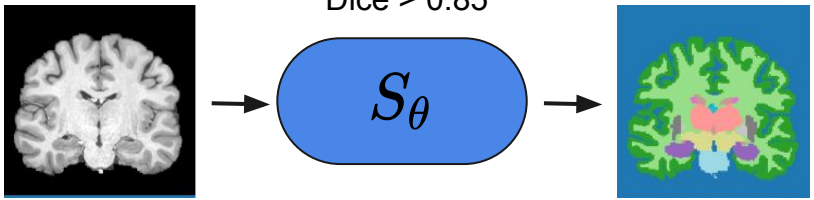


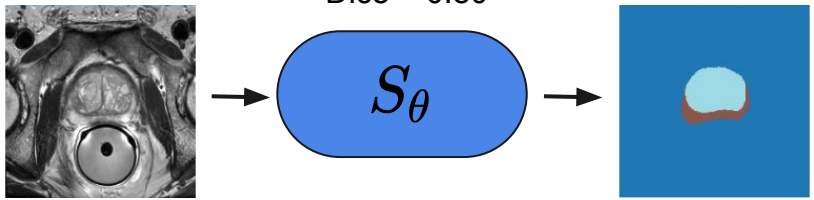
Image segmentation

Training domains

Dice > 0.85



Dice > 0.80



Neural Networks behave similarly?!

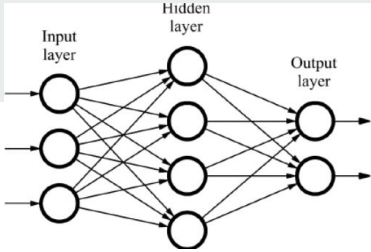


Image segmentation

Training domains

Test domains

Dice > 0.85

Dice > 0.80

Dice ~ 0.60

Dice ~ 0.10

Dice ~ 0.60

S_θ

Neural Networks behave similarly?!

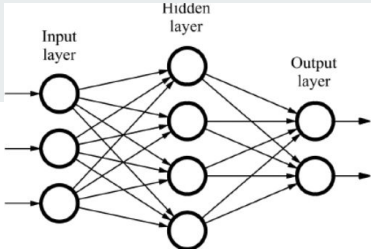
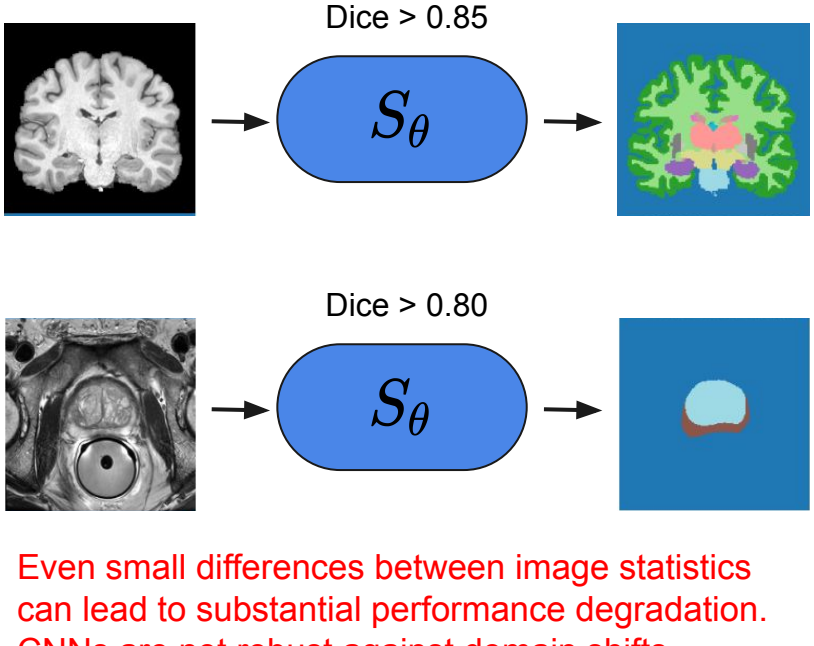


Image segmentation

Training domains

Test domains



Even small differences between image statistics can lead to substantial performance degradation. CNNs are not robust against domain shifts.

Castro, D, at al. "Causality matters in medical imaging." *Nature Communications* (2020)

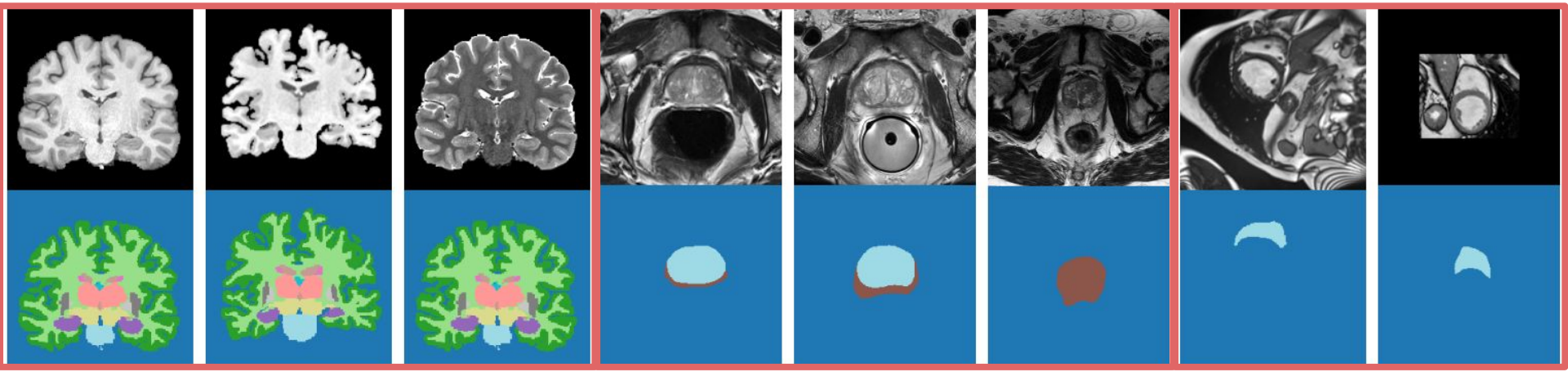
Domain Shift in MRI

Table 2 Types of dataset shift

Type	Direction	Change	Examples of differences
Population shift	causal	$P_D(Z)$	ages, sexes, diets, habits, ethnicities, genetics
Annotation shift	causal	$P_D(Y\|X)$	annotation policy, annotator experience
Prevalence shift	anticausal	$P_D(Y)$	case-control balance, target selection
Manifestation shift	anticausal	$P_D(Z\|Y)$	anatomical manifestation of the target disease or trait
Acquisition shift	either	$P_D(X\|Z)$	scanner, resolution, contrast, modality, protocol

We focus on acquisition-related domain shifts.

Domain Shift in MRI due to differences in acquisition settings



Brain MRIs from different scanners and protocols.

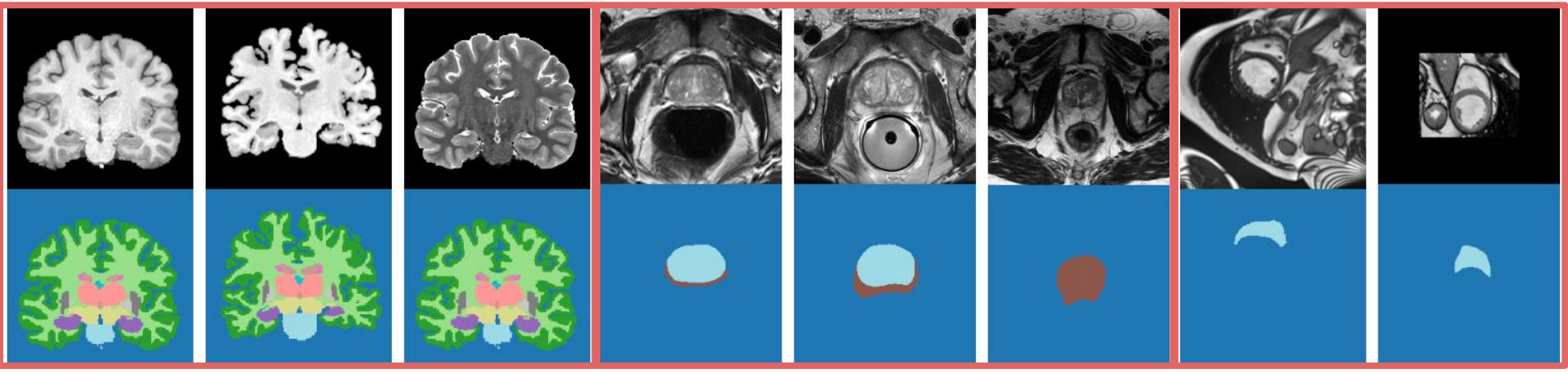
Prostate MRIs from different scanners.

Cardiac MRIs from different scanners.

Introduction

Domain Shift in MRI due to differences in acquisition settings

CNNs do not generalize across such domain shifts.



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806

Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$

Learning setups for dealing with domain shifts

ML Jargon

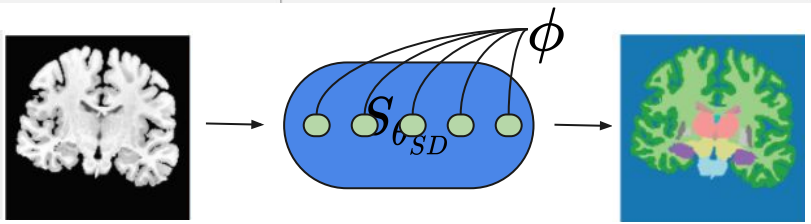
Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$

Learning setups for dealing with domain shifts

Karani, N, et al. "A lifelong learning approach to brain MR segmentation across scanners and protocols." *MICCAI*, 2018.

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$



Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$

Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$
Domain generalization	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Predict $\hat{Y} = S_{\theta^*_{SD}}(X_{TI})$

Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$
Domain generalization	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Predict $\hat{Y} = S_{\theta^*_{SD}}(X_{TI})$
Domain generalization (with test-time adaptation)	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Adapt θ^*_{SD} for each test image (TI). Predict $\hat{Y} = S_{\theta^*_{TI}}(X_{TI})$

Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$
Domain generalization	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Predict $\hat{Y} = S_{\theta^*_{SD}}(X_{TI})$
Domain generalization (with test-time adaptation)	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Adapt θ^*_{SD} for each test image (TI). Predict $\hat{Y} = S_{\theta^*_{TI}}(X_{TI})$
Unsupervised Learning	-	-	(X_{TI}) (one)	Optimize parameters for each TI. Predict $\hat{Y} = argmax_Y P(Y)P(X_{TI}\|Y)$

Learning setups for dealing with domain shifts

ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$
Domain generalization	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Predict $\hat{Y} = S_{\theta^*_{SD}}(X_{TI})$
Domain generalization (with test-time adaptation)	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Adapt θ^*_{SD} for each test image (TI). Predict $\hat{Y} = S_{\theta^*_{TI}}(X_{TI})$

We believe that for medical image segmentation, DG is more suitable than other settings.

Learning setups for dealing with domain shifts

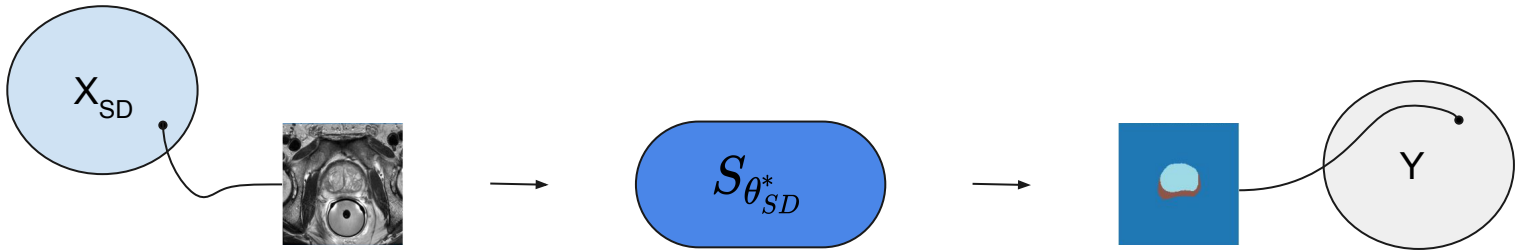
ML Jargon

Setup	At SD		At TD	
	Data	Algorithm	Data	Algorithm
Separate training for each domain	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (many)	$min_\theta L_{TD}$
Transfer learning	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD}$	(X_{TD}, Y_{TD}) (few)	Init. at θ^*_{SD}, $min_\theta L_{TD}$
Unsupervised domain adaptation	-	-	(X_{SD}, Y_{SD}) (many), (X_{TD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$
Domain generalization	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Predict $\hat{Y} = S_{\theta^*_{SD}}(X_{TI})$
Domain generalization (with test-time adaptation)	(X_{SD}, Y_{SD}) (many)	$min_\theta L_{SD} + L_{invariantfeatures}$	(X_{TI}) (one)	Adapt θ^*_{SD} for each test image (TI). Predict $\hat{Y} = S_{\theta^*_{TI}}(X_{TI})$

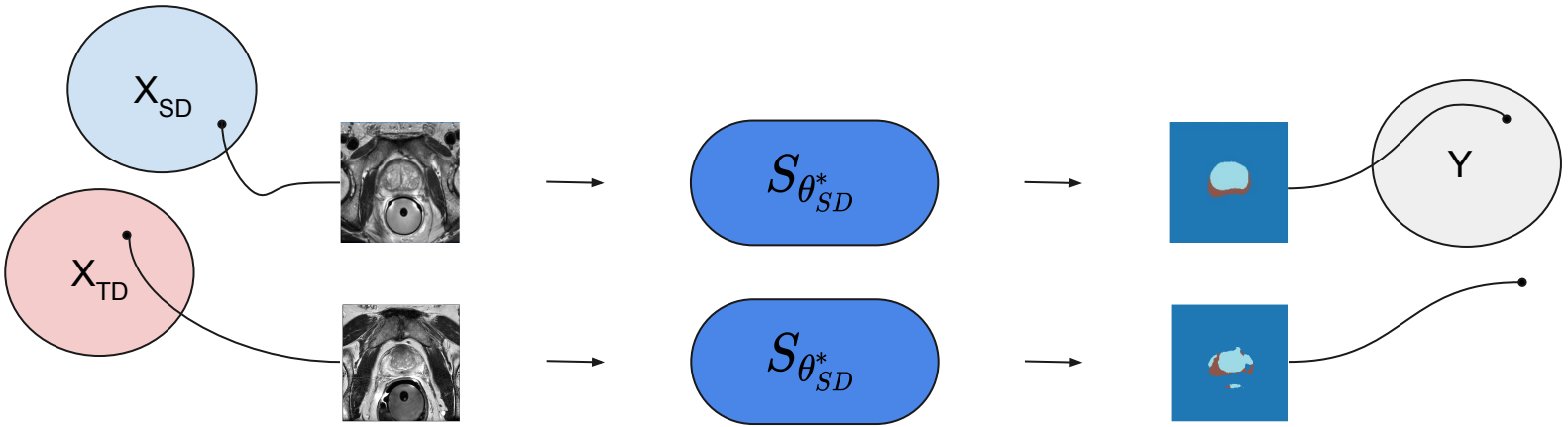
Training strategy

Inference time strategy

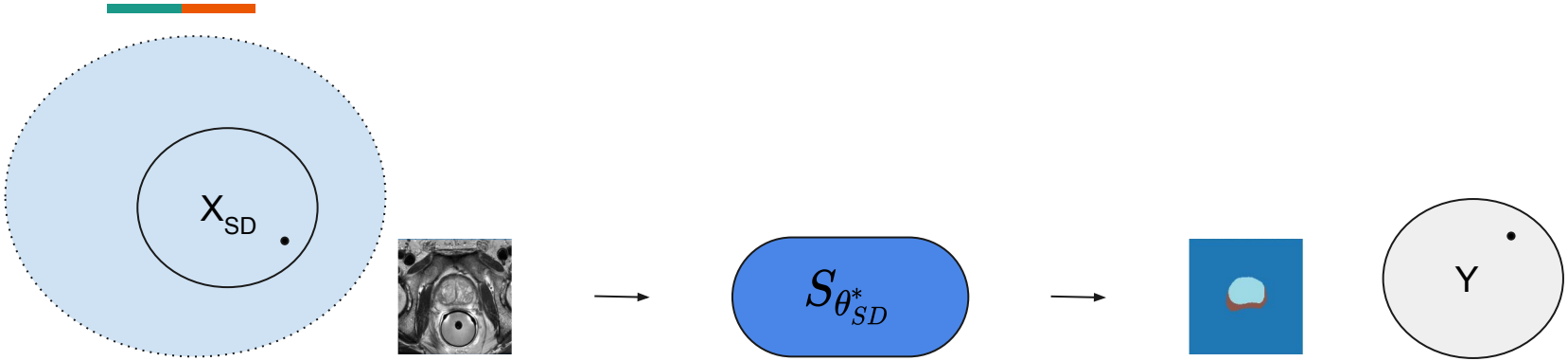
Training strategies for domain generalization



Training strategies for domain generalization

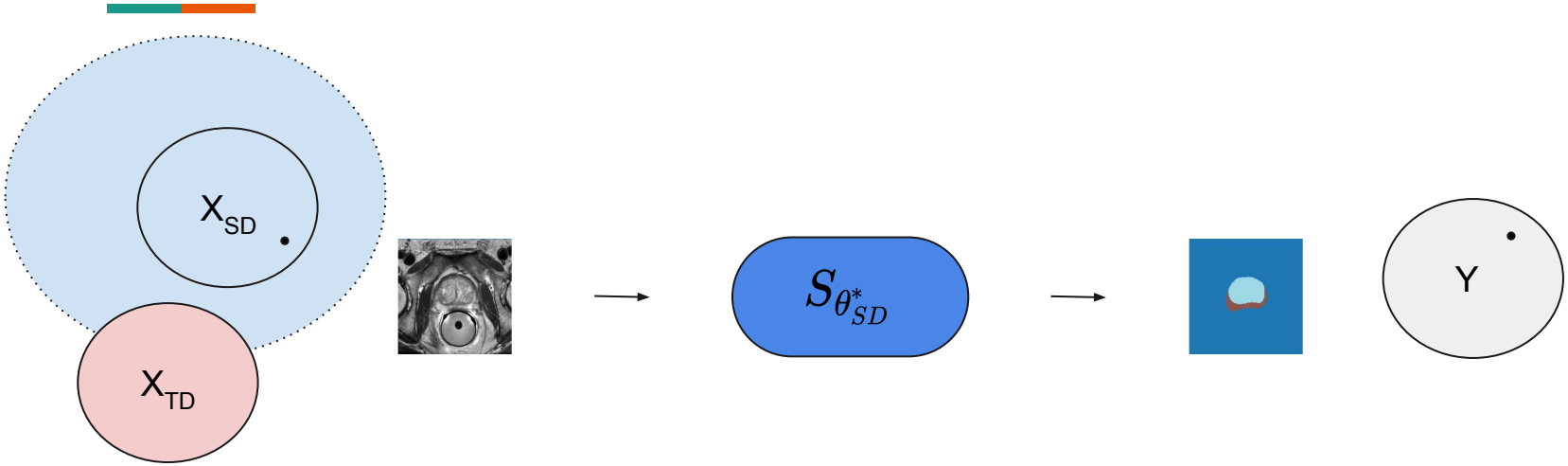


Training strategies for domain generalization - Data Augmentation



- Zhang, L, et al. "Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation." TMI 2020
- Billot, B, et al. "A Learning Strategy for Contrast-agnostic MRI Segmentation." MIDL 2020
- Hendrycks, D, et al. "AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty." ICLR 2019
- Yun, S, et al. "Cutmix: Regularization strategy to train strong classifiers with localizable features." ICCV 2019
- Zhang, H, et al. "mixup: Beyond Empirical Risk Minimization." ICLR 2018
- ...

Training strategies for domain generalization - Data Augmentation

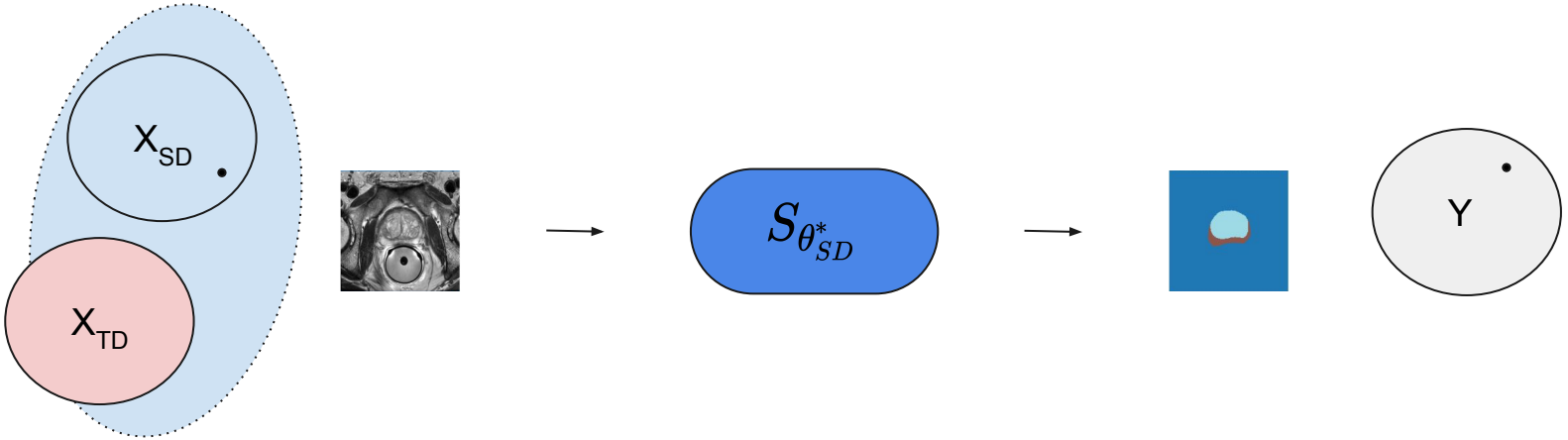


Heuristic data augmentation is a strong baseline.

However, the performance improvement is likely to be limited to the variations covered by the augmentation.

Related Work

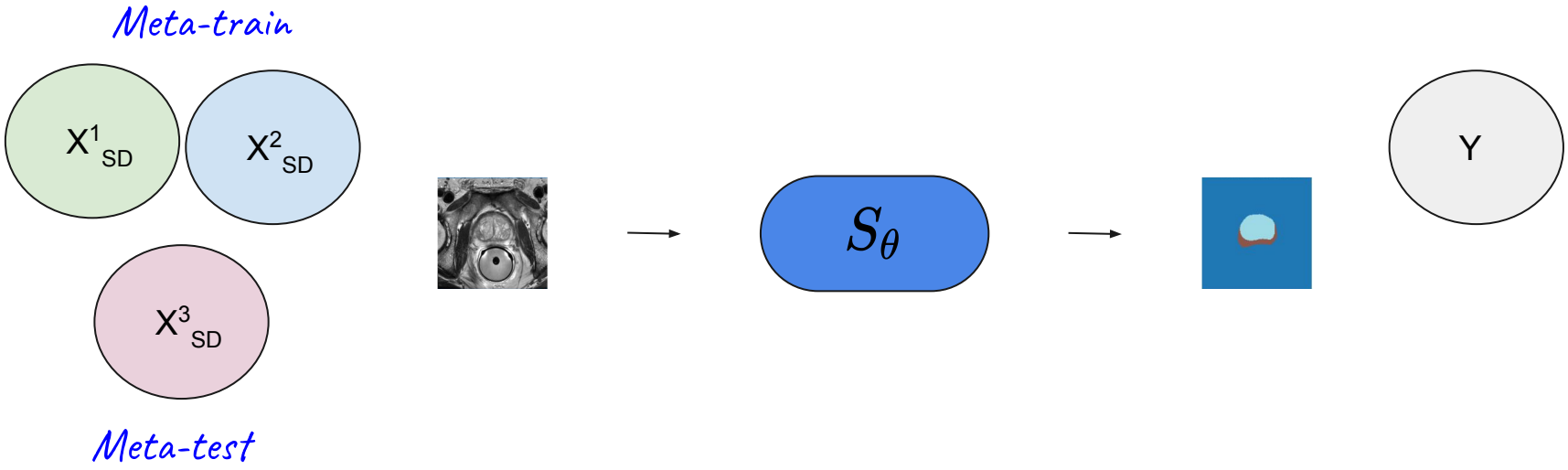
Data Augmentation for unsupervised domain adaptation



- Chaitanya, K, et al. "Semi-supervised and task-driven data augmentation." IPMI 2019.
- Zhao, A, et al. "Data augmentation using learned transformations for one-shot medical image segmentation." CVPR 2019.

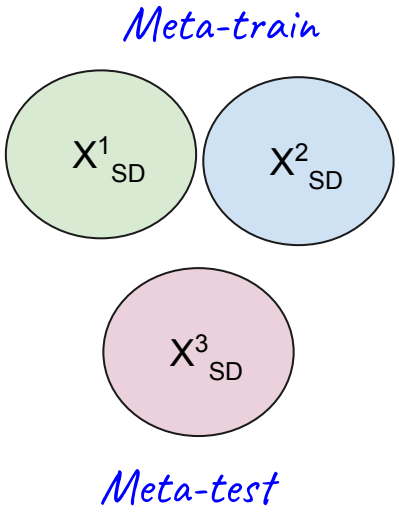
Li, D. et al. "Learning to generalize: Meta-learning for domain generalization." AAAI 2018

Training strategies for domain generalization - Meta Learning



Li, D. et al. "Learning to generalize: Meta-learning for domain generalization." AAAI 2018

Training strategies for domain generalization - Meta Learning



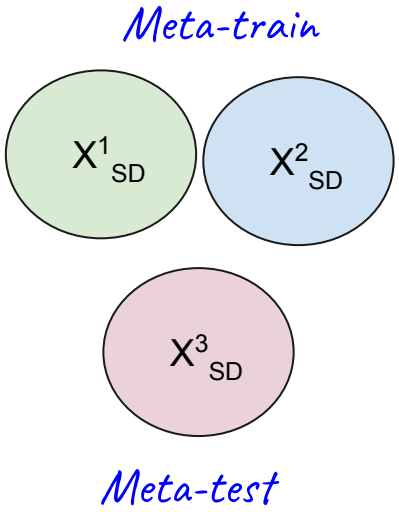
Algorithm 1 Meta-Learning Domain Generalization

procedure MLDG
  Input: Domains $\mathcal{S}$
  Init: Model parameters Θ. Hyperparameters α, β, γ.
  for ite **in** iterations **do**
    Split: $\bar{\mathcal{S}}$ and $\check{\mathcal{S}} \leftarrow \mathcal{S}$
    Meta-train: Gradients $\nabla_\Theta = \mathcal{F}'_\Theta(\bar{\mathcal{S}}; \Theta)$
    Updated parameters $\Theta' = \Theta - \alpha \nabla_\Theta$
    Meta-test: Loss is $\mathcal{G}(\check{\mathcal{S}}; \Theta')$.
    Meta-optimization: Update Θ

$$\Theta = \Theta - \gamma \frac{\partial(\mathcal{F}(\bar{\mathcal{S}}; \Theta) + \beta \mathcal{G}(\check{\mathcal{S}}; \Theta - \alpha \nabla_\Theta))}{\partial \Theta}$$

  end for
end procedure

Training strategies for domain generalization - Meta Learning



Algorithm 1 Meta-Learning Domain Generalization

procedure MLDG
  Input: Domains $\mathcal{S}$
  Init: Model parameters Θ. Hyperparameters α, β, γ.
  for ite **in** iterations **do**
    Split: $\bar{\mathcal{S}}$ and $\check{\mathcal{S}} \leftarrow \mathcal{S}$
    Meta-train: Gradients $\nabla_\Theta = \mathcal{F}'_\Theta(\bar{\mathcal{S}}; \Theta)$
    Updated parameters $\Theta' = \Theta - \alpha \nabla_\Theta$
    Meta-test: Loss is $\mathcal{G}(\check{\mathcal{S}}; \Theta')$.
    Meta-optimization: Update Θ

$$\Theta = \Theta - \gamma \frac{\partial(\mathcal{F}(\bar{\mathcal{S}}; \Theta) + \beta \mathcal{G}(\check{\mathcal{S}}; \Theta - \alpha \nabla_\Theta))}{\partial \Theta}$$

  end for
end procedure

- Li, D. et al. "Learning to generalize: Meta-learning for domain generalization." AAAI 2018
- Dou, Q. et al. "Domain generalization via model-agnostic learning of semantic features." NeurIPS 2019
- Tseng, H. et al. "Cross-Domain Few-Shot Classification via Learned Feature-Wise Transformation." ICLR 2020
- Zhang, J. et al. "Generalizable Semantic Segmentation via Model-agnostic Learning and Target-specific Normalization." arXiv preprint 2020

Training strategies for domain generalization

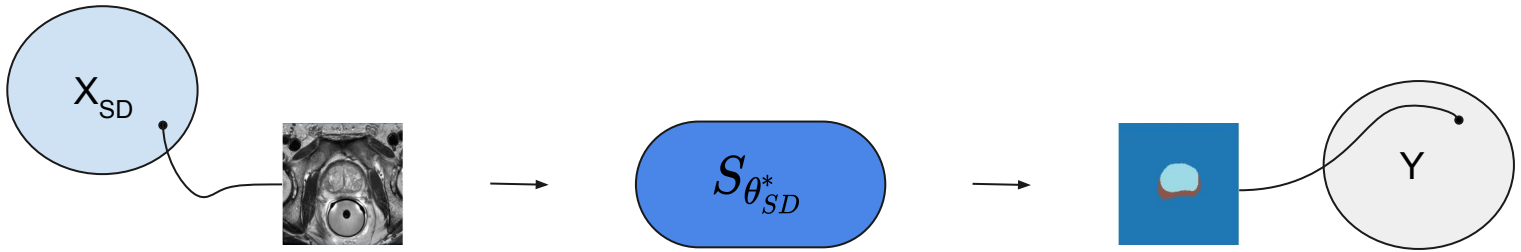
Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Test / Train	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Relevant Methods										
MLDG [Li et al. (2018a)]	0.874	0.686	0.074	0.913	0.772	0.756	0.818	0.658	0.844	0.696
MASF [Dou et al. (2019a)]	0.870	0.693	0.073	0.913	0.751	0.781	0.817	0.640	0.838	0.698
MLDGTS [Zhang et al. (2020b)]	0.876	0.733	0.072	0.912	0.711	0.761	0.815	0.608	0.831	0.361
SD + DA [Zhang et al. (2020)] (Strong baseline)	0.876	0.753	0.083	0.911	0.769	0.786	0.815	0.656	0.834	0.744

Training strategies improve DG performance.

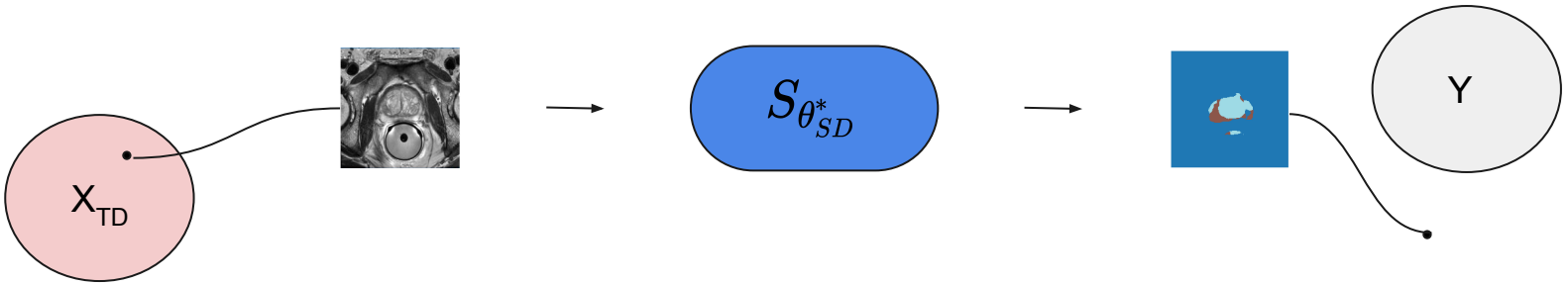
Also, they improve performance within SD!

However, there still remains a gap to the benchmark.

The domain shift problem in machine learning

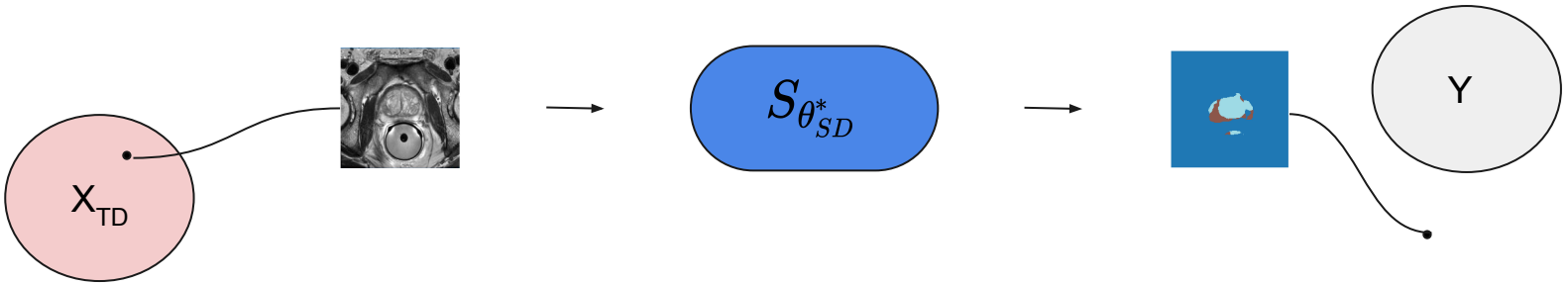


The domain shift problem in machine learning



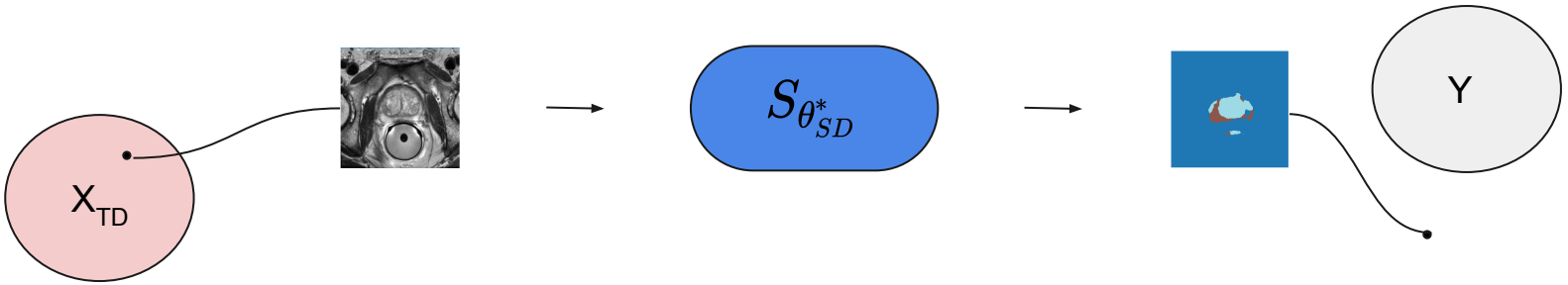
The domain shift problem in machine learning

Hypothesis: For better generalization, we need per-test-image adaptability.



The domain shift problem in machine learning

Hypothesis: For better generalization, we need per-test-image adaptability.

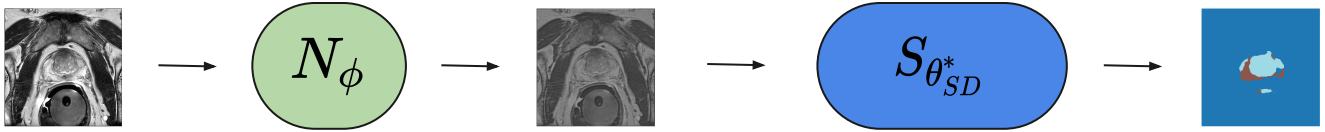


Two questions:

1. Which parameters to adapt?
2. How to drive the adaptation?

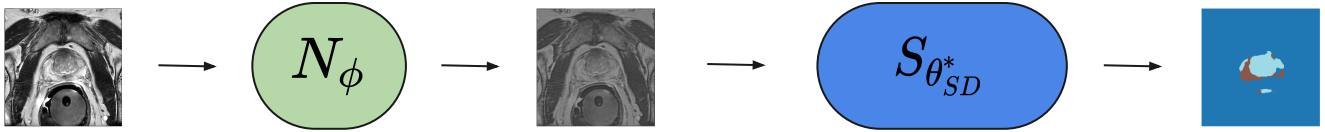
Inference strategy for domain generalization

1. Which parameters to adapt?



Inference strategy for domain generalization

1. Which parameters to adapt?

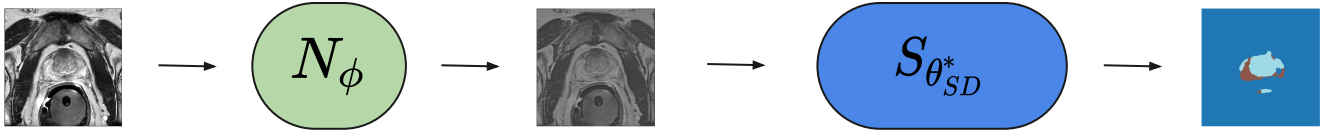


We train N and S jointly, on the SD.

Then, we fix S and adapt N for each test image.

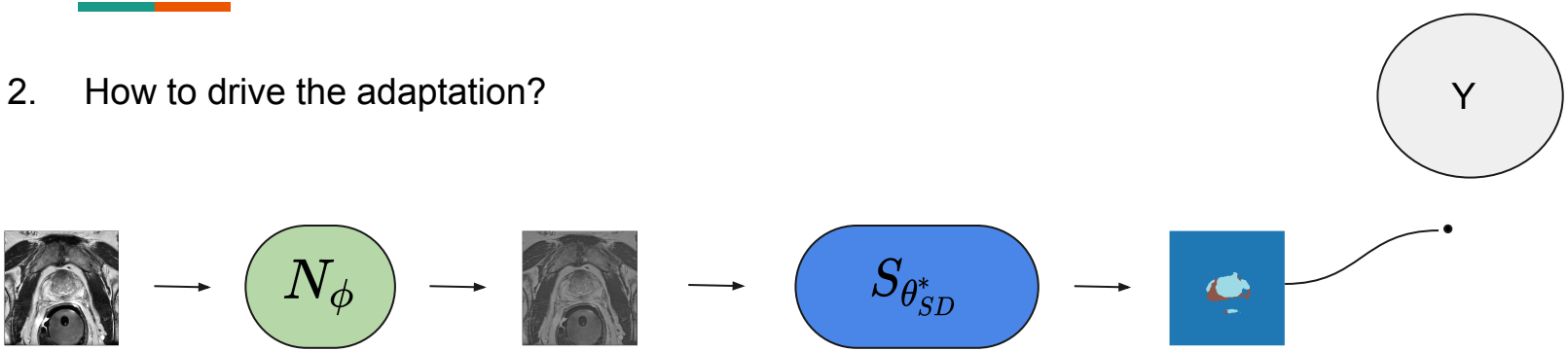
Inference strategy for domain generalization

2. How to drive the adaptation?



Inference strategy for domain generalization

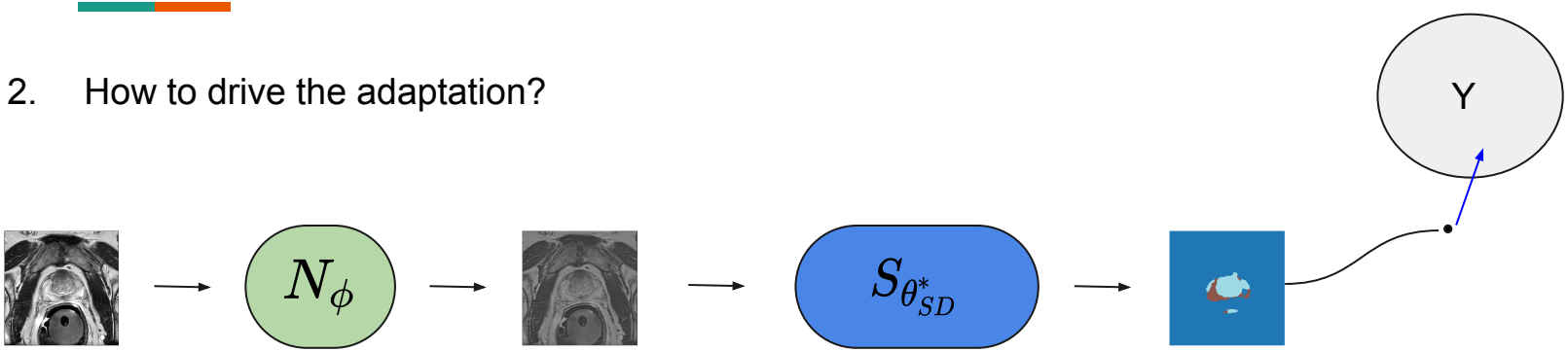
2. How to drive the adaptation?



The predicted segmentation is likely to be outside the distribution of plausible segmentations.

Inference strategy for domain generalization

2. How to drive the adaptation?



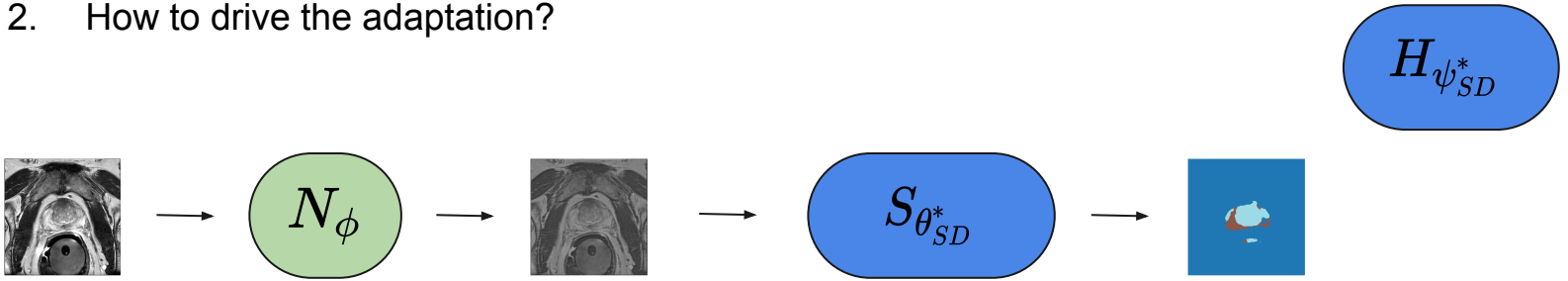
The predicted segmentation is likely to be outside the distribution of plausible segmentations.

We drive the adaptation by encouraging the predicted segmentation to become plausible.

Inference strategy for domain generalization

Helper Network

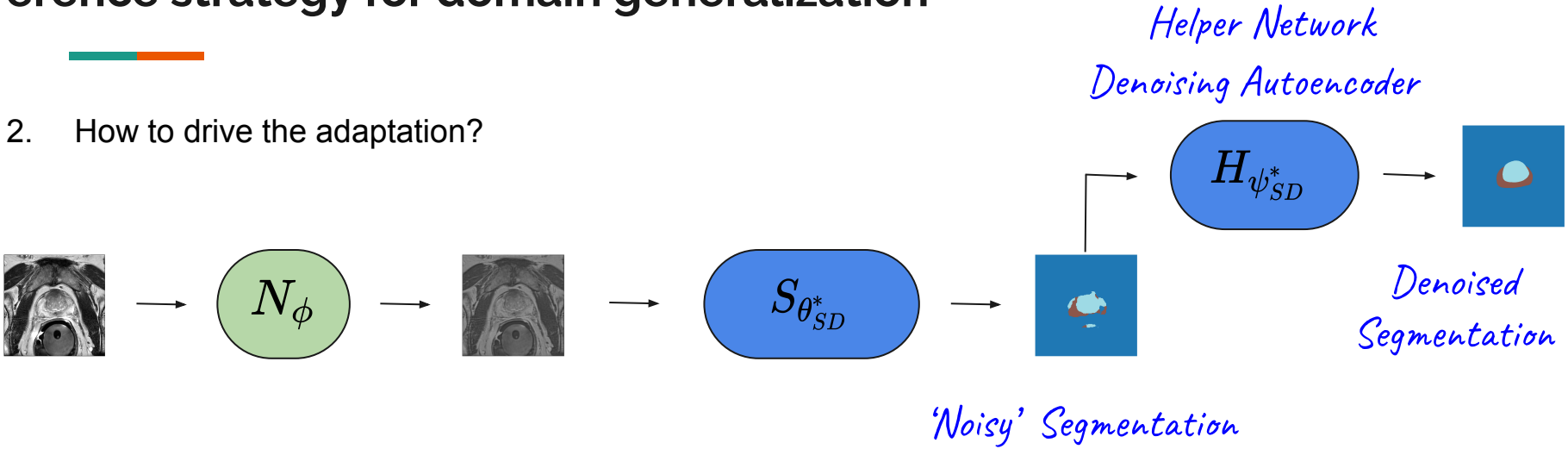
2. How to drive the adaptation?



For this, we introduce a ‘helper’ network.

Inference strategy for domain generalization

2. How to drive the adaptation?

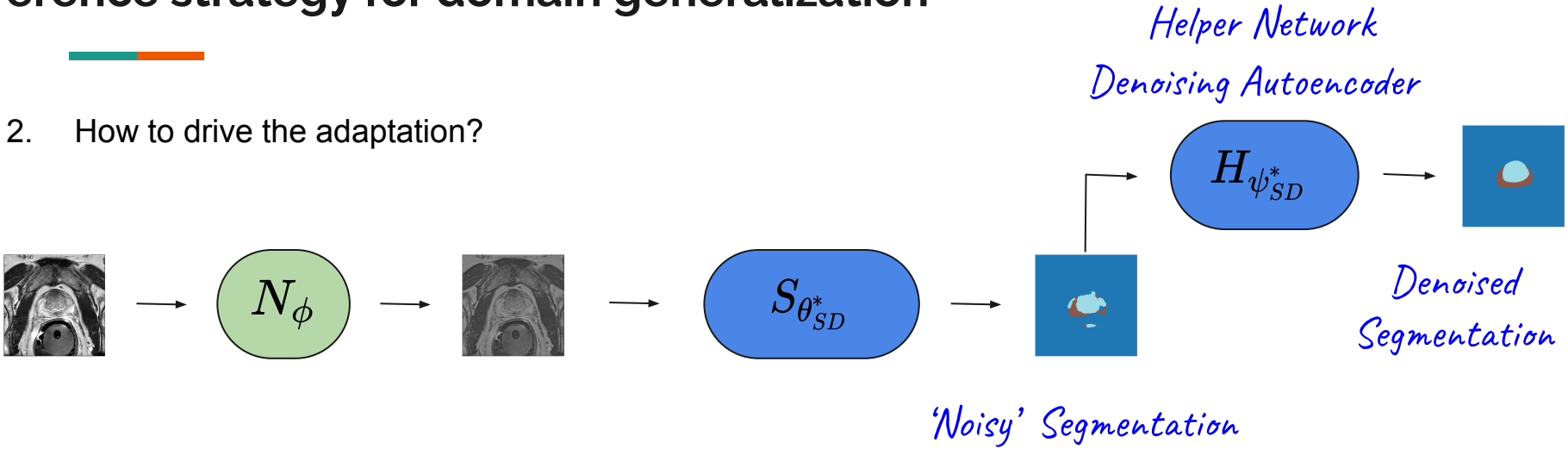


The segmentations predicted for TD images are treated as 'noisy' segmentations.

A DAE is used to denoise these predictions.

Inference strategy for domain generalization

2. How to drive the adaptation?



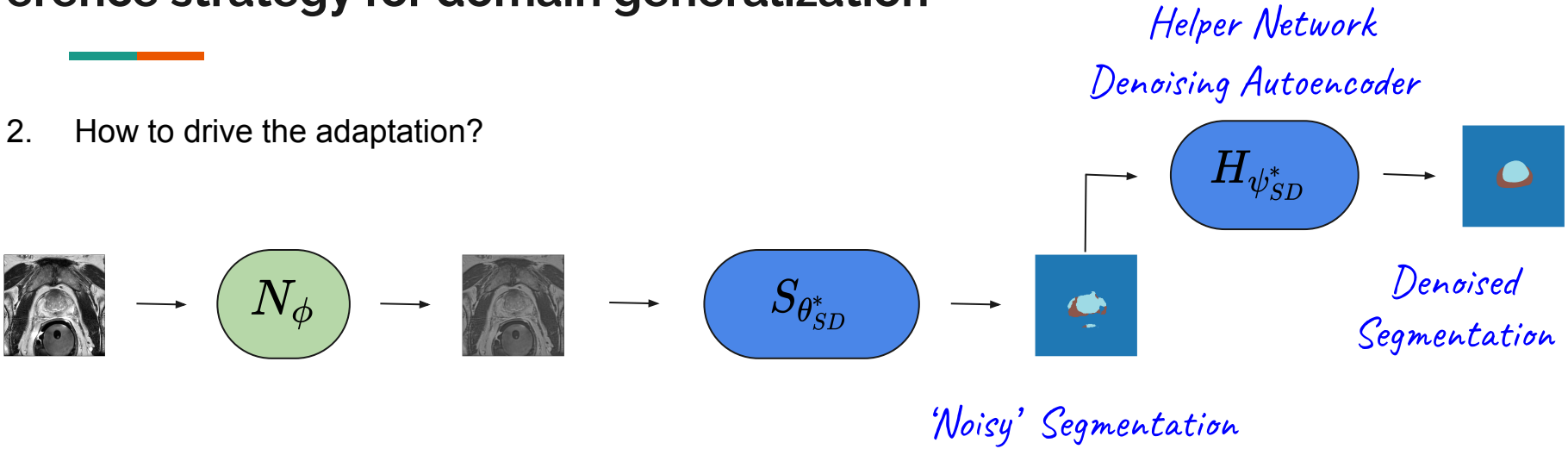
The segmentations predicted for TD images are treated as 'noisy' segmentations.

A DAE is used to denoise these predictions.

We adapt N for each test image, such that the DAE inputs and outputs are similar.

Inference strategy for domain generalization

2. How to drive the adaptation?



The segmentations predicted for TD images are treated as 'noisy' segmentations.

A DAE is used to denoise these predictions.

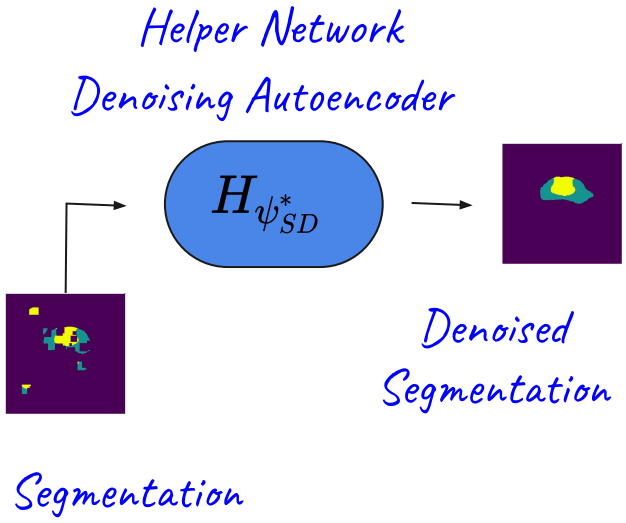
We adapt N for each test image, such that the DAE inputs and outputs are similar.

$$min_\phi L(y_n, H_{\psi^*_{SD}}(y_n))$$
$$y_n = S_{\theta^*_{SD}}(N_\phi(x_{TI}))$$

Inference strategy for domain generalization

For training the DAE, the noisy segmentations are generated by a heuristic noising process.

* the DAE is trained in 3D



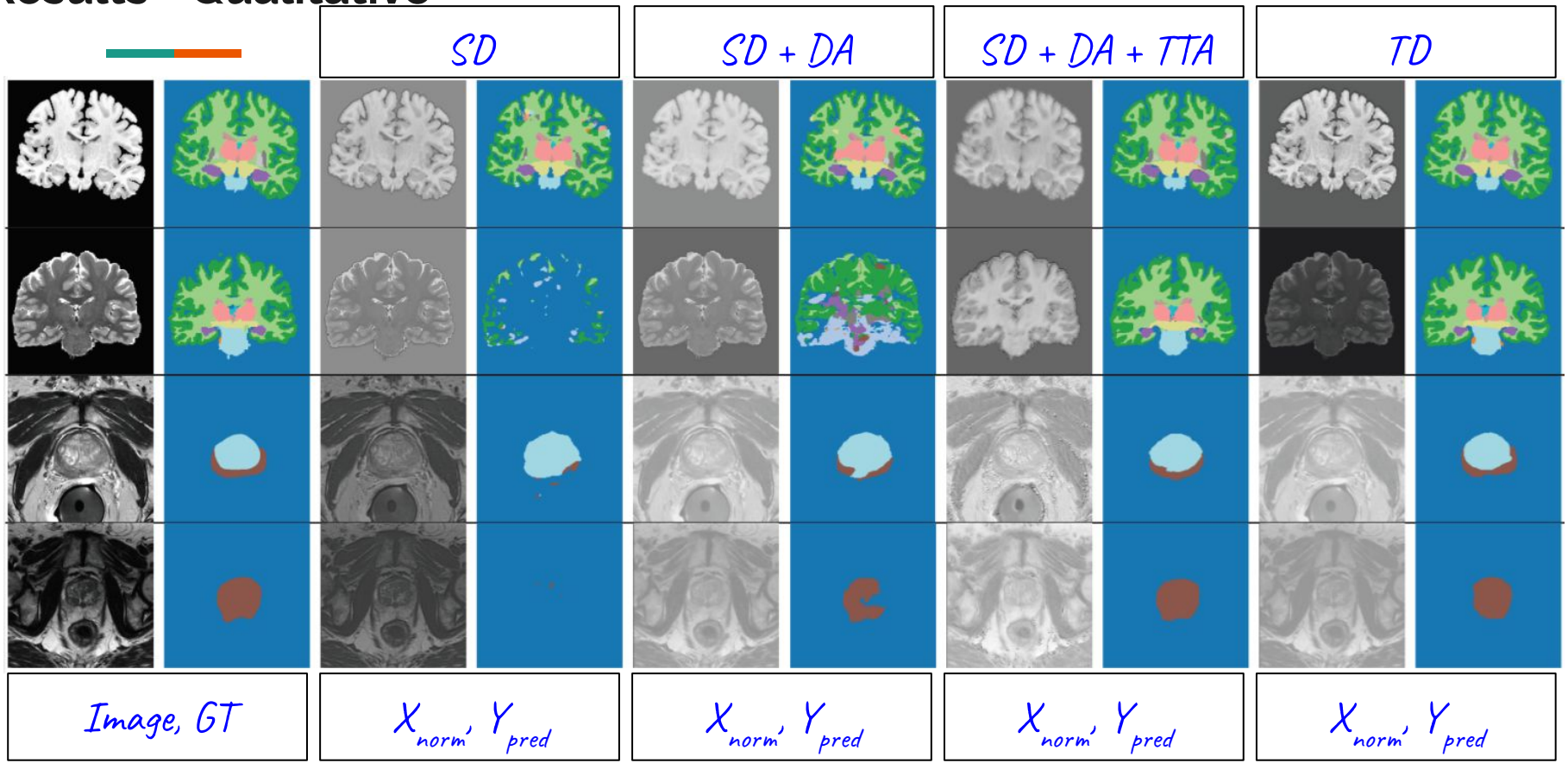
Test-Time Adaptation

Results

Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Relevant Methods										
MLDG [Li et al. (2018a)]	0.874	0.686	0.074	0.913	0.772	0.756	0.818	0.658	0.844	0.696
MASF [Dou et al. (2019a)]	0.870	0.693	0.073	0.913	0.751	0.781	0.817	0.640	0.838	0.698
MLDGTS [Zhang et al. (2020b)]	0.876	0.733	0.072	0.912	0.711	0.761	0.815	0.608	0.831	0.361
SD + DA [Zhang et al. (2020)] (Strong baseline)	0.876	0.753	0.083	0.911	0.769	0.786	0.815	0.656	0.834	0.744
SD + DA + Post-Proc. [Larrazabal et al. (2019)]	-	0.706	0.112	-	0.789	0.823	-	0.678	-	0.746
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742

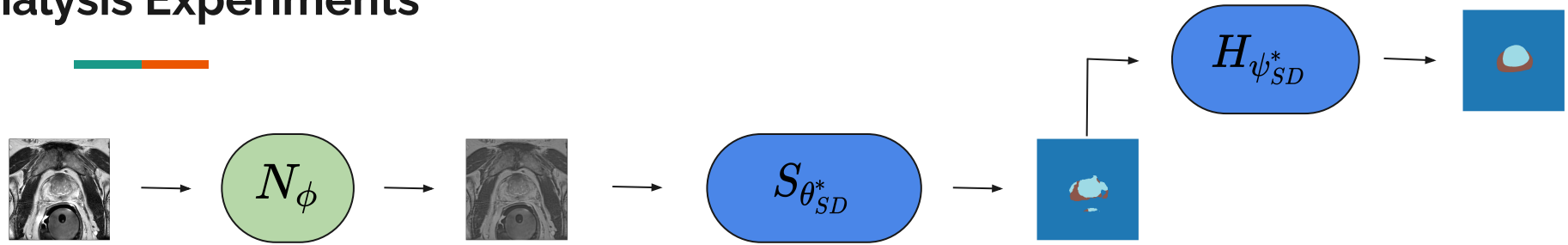
Test-Time Adaptation

Results - Qualitative



Test-Time Adaptation

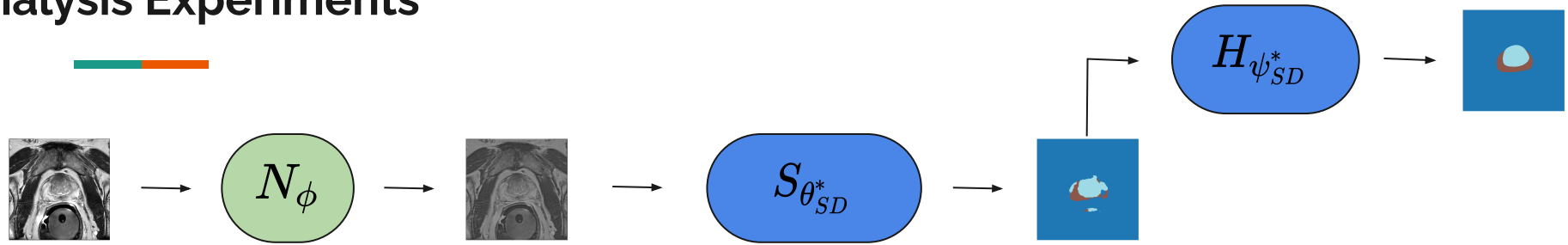
Analysis Experiments



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742
Ablation Studies										
SD + DA + TTA (Adapt ϕ, using DAE) - **Fast**	-	**0.800**	0.728	-	0.790	0.842	-	**0.683**	-	**0.747**
SD + DA + TTA (Adapt ϕ, θ, using DAE)	-	0.671	0.650	-	0.718	0.606	-	0.583	-	0.713
SD + DA + TTA (Adapt ϕ, using GT labels)	-	0.831	0.837	-	0.836	-	-	0.771	-	-
SD + DA + Post.Proc. 10 passes through DAE	-	0.633	0.106	-	**0.791**	0.826	-	**0.683**	-	0.731
SD + DA + Post.Proc. 100 passes through DAE	-	0.529	0.101	-	0.789	0.823	-	0.672	-	0.688

Test-Time Adaptation

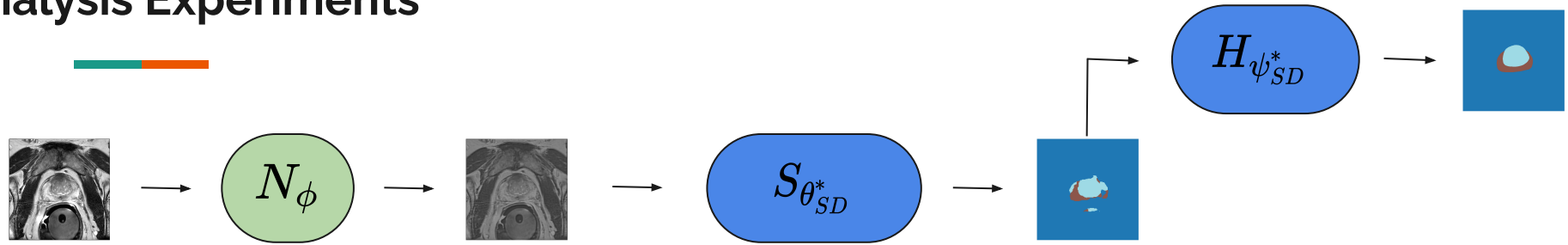
Analysis Experiments



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742
Ablation Studies										
SD + DA + TTA (Adapt ϕ, using DAE) - **Fast**	-	**0.800**	0.728	-	0.790	0.842	-	**0.683**	-	**0.747**
SD + DA + TTA (Adapt ϕ, θ, using DAE)	-	0.671	0.650	-	0.718	0.606	-	0.583	-	0.713
SD + DA + TTA (Adapt ϕ, using GT labels)	-	0.831	0.837	-	0.836	-	-	0.771	-	-
SD + DA + Post.Proc. 10 passes through DAE	-	0.633	0.106	-	**0.791**	0.826	-	**0.683**	-	0.731
SD + DA + Post.Proc. 100 passes through DAE	-	0.529	0.101	-	0.789	0.823	-	0.672	-	0.688

Test-Time Adaptation

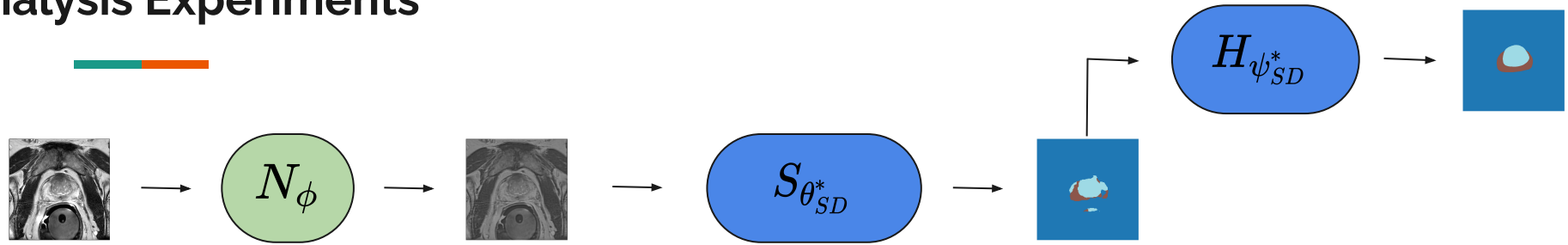
Analysis Experiments



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742
Ablation Studies										
SD + DA + TTA (Adapt ϕ, using DAE) - **Fast**	-	**0.800**	0.728	-	0.790	0.842	-	**0.683**	-	**0.747**
SD + DA + TTA (Adapt ϕ, θ, using DAE)	-	0.671	0.650	-	0.718	0.606	-	0.583	-	0.713
SD + DA + TTA (Adapt ϕ, using GT labels)	-	0.831	0.837	-	0.836	-	-	0.771	-	-
SD + DA + Post.Proc. 10 passes through DAE	-	0.633	0.106	-	**0.791**	0.826	-	**0.683**	-	0.731
SD + DA + Post.Proc. 100 passes through DAE	-	0.529	0.101	-	0.789	0.823	-	0.672	-	0.688

Test-Time Adaptation

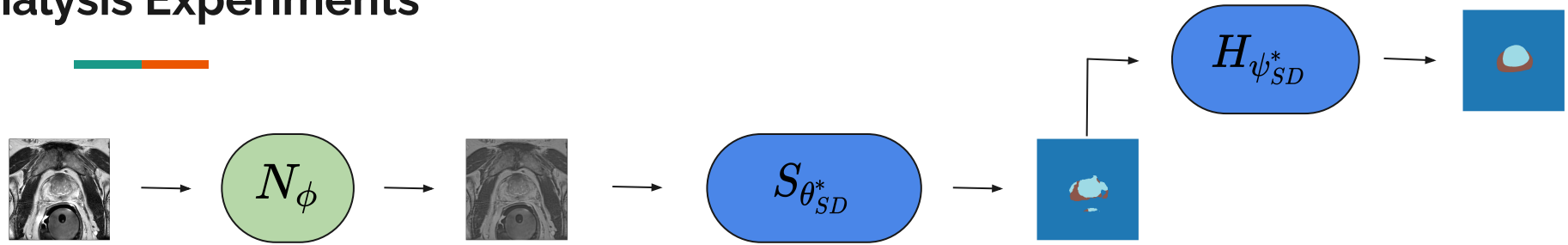
Analysis Experiments



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742
Ablation Studies										
SD + DA + TTA (Adapt ϕ, using DAE) - **Fast**	-	**0.800**	0.728	-	0.790	0.842	-	**0.683**	-	**0.747**
SD + DA + TTA (Adapt ϕ, θ, using DAE)	-	0.671	0.650	-	0.718	0.606	-	0.583	-	0.713
SD + DA + TTA (Adapt ϕ, using GT labels)	-	0.831	0.837	-	0.836	-	-	0.771	-	-
SD + DA + Post.Proc. 10 passes through DAE	-	0.633	0.106	-	**0.791**	0.826	-	**0.683**	-	0.731
SD + DA + Post.Proc. 100 passes through DAE	-	0.529	0.101	-	0.789	0.823	-	0.672	-	0.688

Test-Time Adaptation

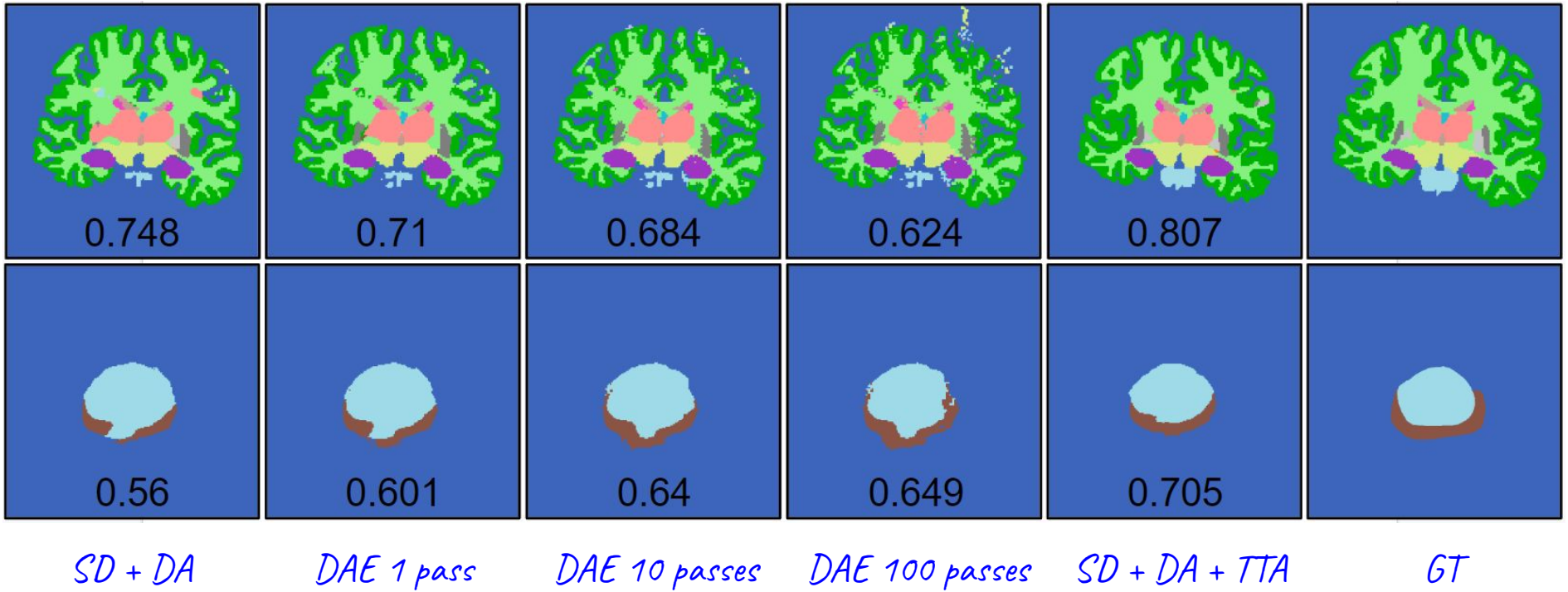
Analysis Experiments



Anatomy	Brain			Prostate (whole)			Prostate (sep.)		Heart	
Train \ Test	SD	TD_1	TD_2	SD	TD_1	TD_2	SD	TD_1	SD	TD_1
Baseline and Benchmark										
SD (Baseline)	0.853	0.588	0.107	0.840	0.586	0.609	0.722	0.544	0.823	0.670
TD_n (Benchmark)	-	0.896	0.867	-	0.817	0.834	-	0.732	-	0.806
Proposed Method										
SD + DA + TTA (Adapt ϕ, using DAE)	-	**0.800***	**0.733***	-	0.790	**0.858***	-	0.676	-	0.742
Ablation Studies										
SD + DA + TTA (Adapt ϕ, using DAE) - **Fast**	-	**0.800**	0.728	-	0.790	0.842	-	**0.683**	-	**0.747**
SD + DA + TTA (Adapt ϕ, θ, using DAE)	-	0.671	0.650	-	0.718	0.606	-	0.583	-	0.713
SD + DA + TTA (Adapt ϕ, using GT labels)	-	0.831	0.837	-	0.836	-	-	0.771	-	-
SD + DA + Post.Proc. 10 passes through DAE	-	0.633	0.106	-	**0.791**	0.826	-	**0.683**	-	0.731
SD + DA + Post.Proc. 100 passes through DAE	-	0.529	0.101	-	0.789	0.823	-	0.672	-	0.688

Test-Time Adaptation

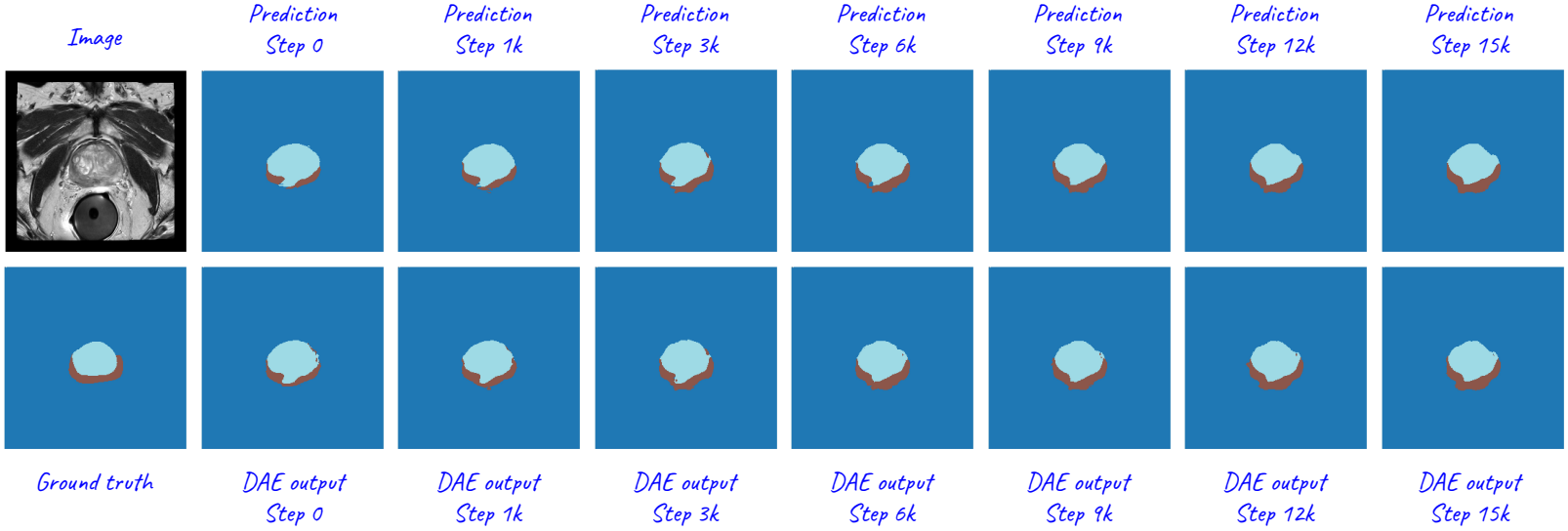
Importance of adaptation v/s using DAE for post-processing

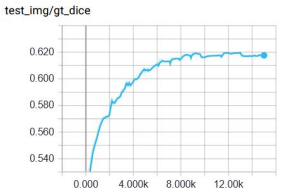


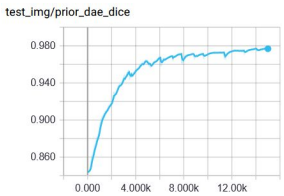
Test-Time Adaptation

Evolution of segmentation over adaptation iterations

Prostate USZ 45



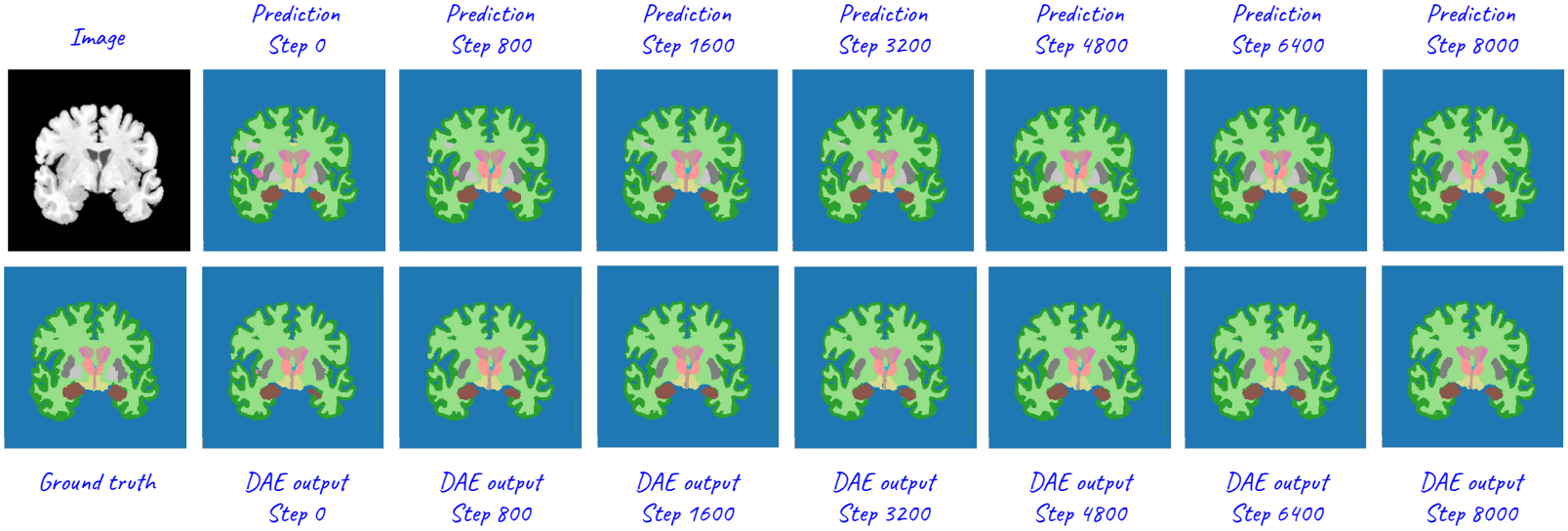


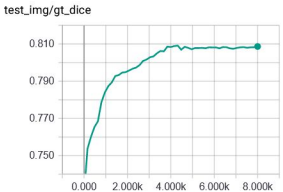


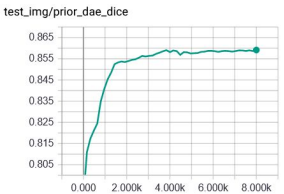
Test-Time Adaptation

Evolution of segmentation over adaptation iterations

Brain Caltech 288







Comparison with Unsupervised Domain Adaptation

Anatomy	Brain				Prostate (whole)				Heart	
Train \ Test	TD_1^{tr}	TD_1^{ts}	TD_2^{tr}	TD_2^{ts}	TD_1^{tr}	TD_1^{ts}	TD_2^{tr}	TD_2^{ts}	TD_1^{tr}	TD_1^{ts}
SD + DA [Zhang et al. (2020)] (Strong baseline)	0.750	0.753	0.081	0.083	0.767	0.769	0.747	0.786	0.715	0.744
UDA - Invariant features [Kamnitsas et al. (2017a)]	0.792	0.798	0.081	0.083	0.789	0.793	0.766	0.802	0.724	0.750
UDA - Image-to-Image translation [Huo et al. (2018)]	0.646	0.639	0.816	0.813	0.607	0.694	0.765	0.747	0.252	0.167
TTA (Proposed)	-	0.800	-	0.733	-	0.790	-	0.858	-	0.742
TD_n (Benchmark)	-	0.896	-	0.867	-	0.817	-	0.834	-	0.806

UDA methods work is a more relaxed scenario - labelled SD dataset is available for each new TD.

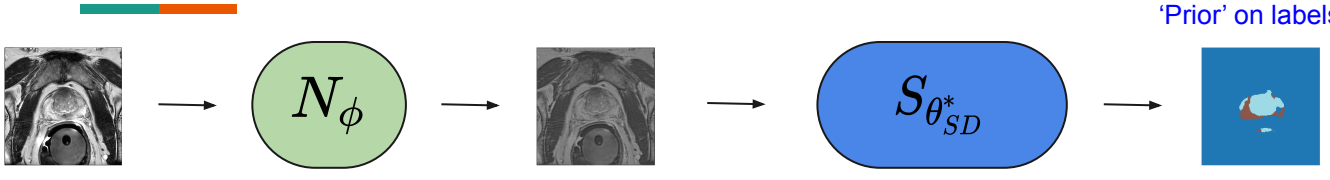
The proposed TTA method is a domain generalization method.

We find that TTA performance is similar to state-of-the-art UDA methods.

Connection with concurrent TTA works

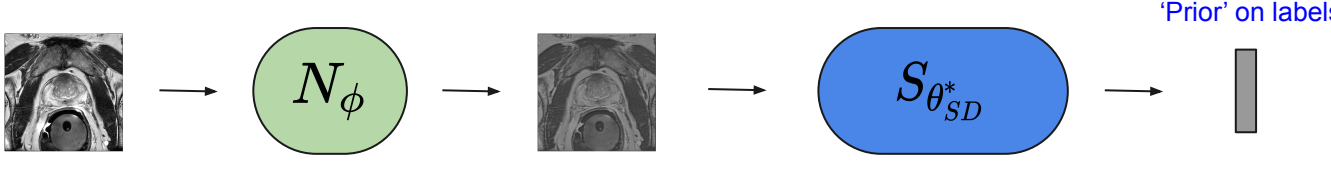
Test-Time Adaptation

Connection with concurrent TTA works



'Prior' on labels

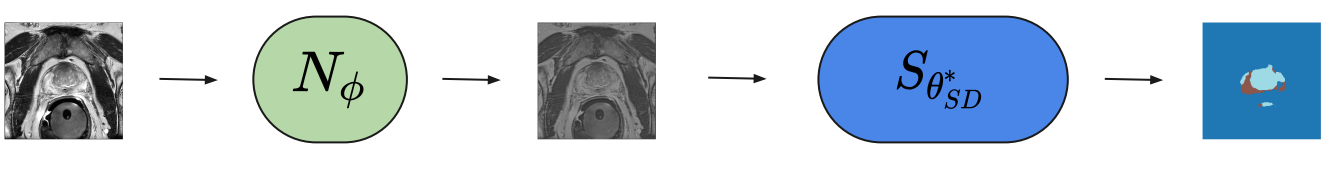
Karani et al. "Test-time adaptable neural networks for robust medical image segmentation." *Medical Image Analysis* 2021.



'Prior' on labels

Wang et al. "Fully Test-time Adaptation by Entropy Minimization." arXiv preprint 2020

'Prior' on intermediate features



Sun et al. "Test-Time Training with Self-Supervision for Generalization under Distribution Shifts." ICML 2020

Yufan et al. Self domain adapted network. MICCAI 2020

Test-Time Adaptation for Robust Medical Image Segmentation

Summary

- We proposed test-time adaptation for robust image segmentation.
- The method consists of two main ideas:
  - An adaptable per-image normalization module in front of the segmentation CNN.
  - The adaptation is driven by denoising autoencoders, that incentivize plausible segmentation predictions.
- Substantial improvement across datasets and anatomies.
- Increased inference time - about 10 minutes for each 3D volume.

Test-Time Adaptation for Robust Medical Image Segmentation

Summary

- We proposed test-time adaptation for robust image segmentation.
- The method consists of two main ideas:
  - An adaptable per-image normalization module in front of the segmentation CNN.
  - The adaptation is driven by denoising autoencoders, that incentivize plausible segmentation predictions.
- Substantial improvement across datasets and anatomies.
- Increased inference time - about 10 minutes for each 3D volume.

Thanks!

Krishna Chaitanya

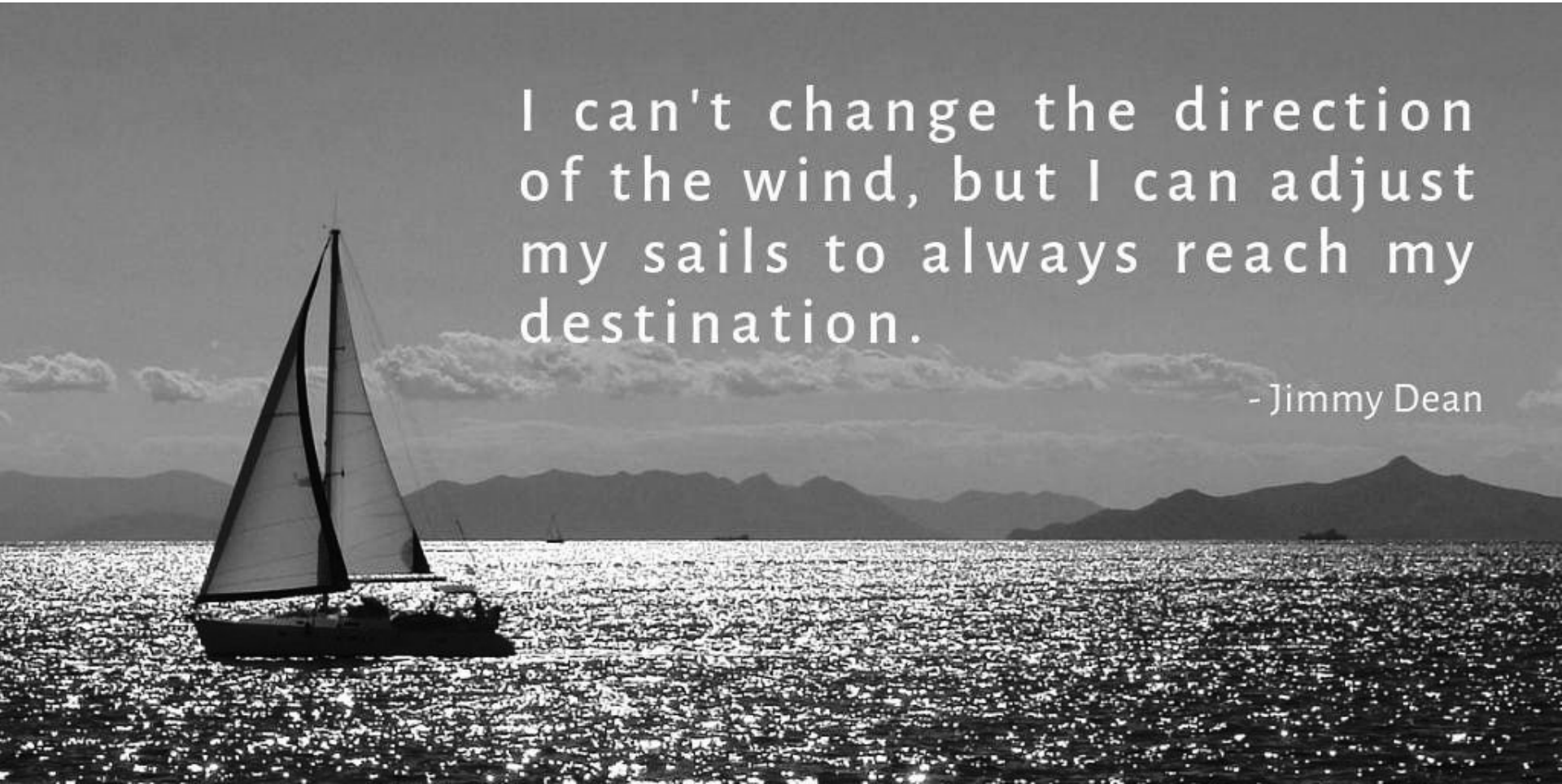
Ertunc Erdil

Christian Baumgartner

Ender Konukoglu

In case of questions, please feel free to write to nkarani@vision.ee.ethz.ch

Thanks!


I can't change the direction of the wind, but I can adjust my sails to always reach my destination.
- Jimmy Dean