A New Ensemble Learning Framework for 3D Biomedical Image Segmentation

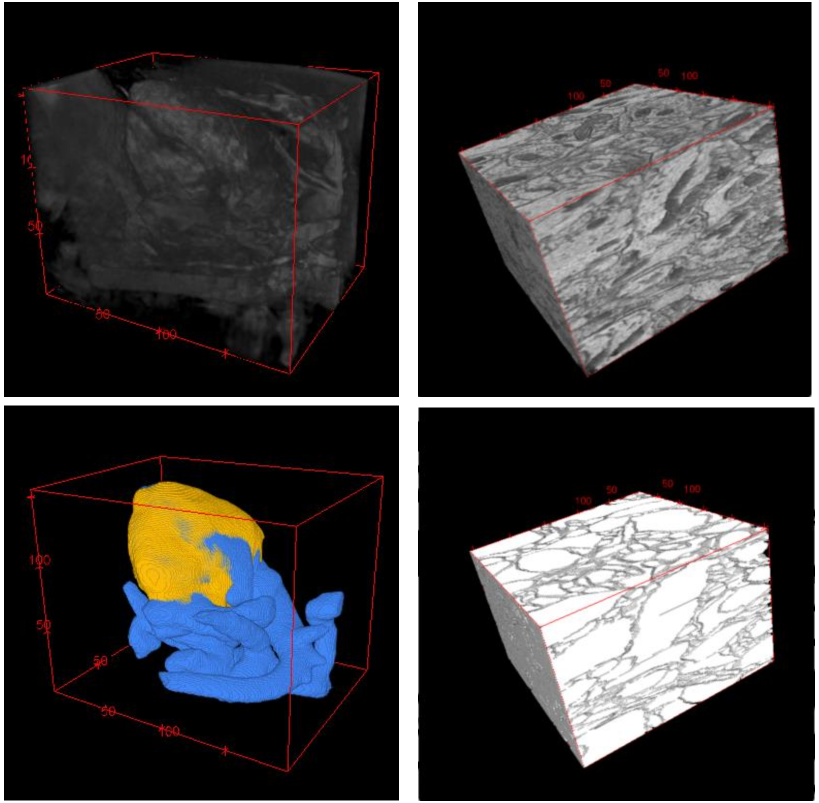
Hao Zheng, Yizhe Zhang, Lin Yang, Peixian Liang,
Zhuo Zhao, Chaoli Wang, Danny Z. Chen

Department of Computer Science and Engineering
University of Notre Dame
Indiana, USA
hzheng3@nd.edu

UNIVERSITY OF NOTRE DAME

3D biomedical image segmentation

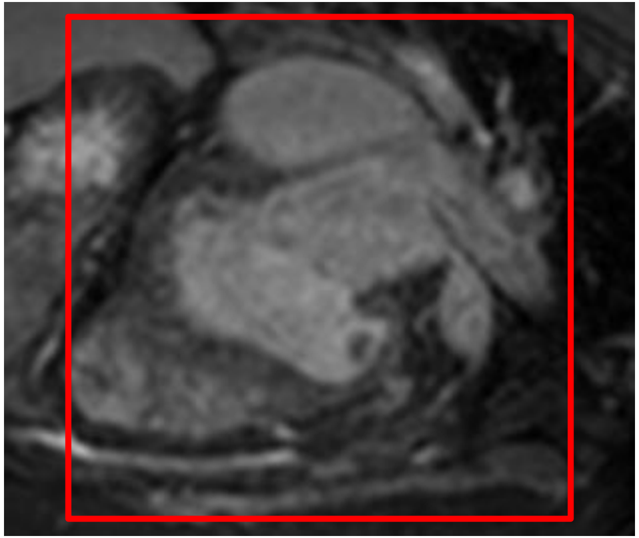
400x400



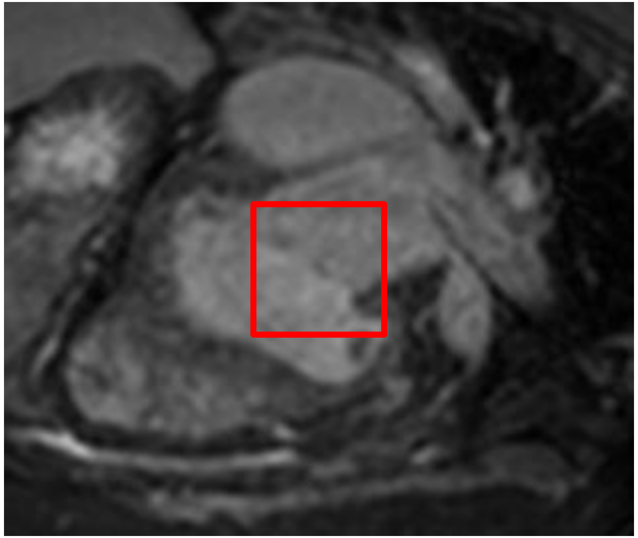
200x150x120 512x512x200

Challenges in 3D image segmentation

- Large image size in 3D and limited GPU memory
- Trade-off between inter-slice information and the size of field of view



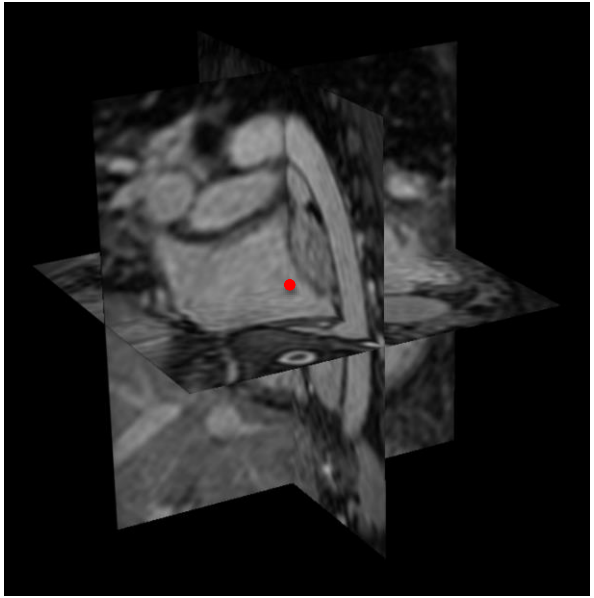
2D: 572 x 572 x 1



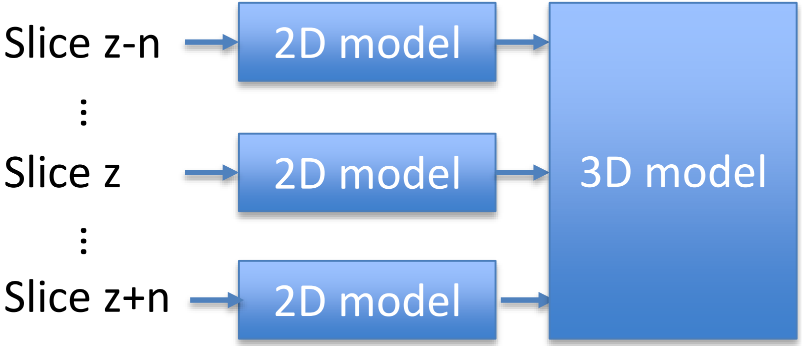
3D: 64 x 64 x 64

Previous methods to circumvent this trade-off

Triplanes

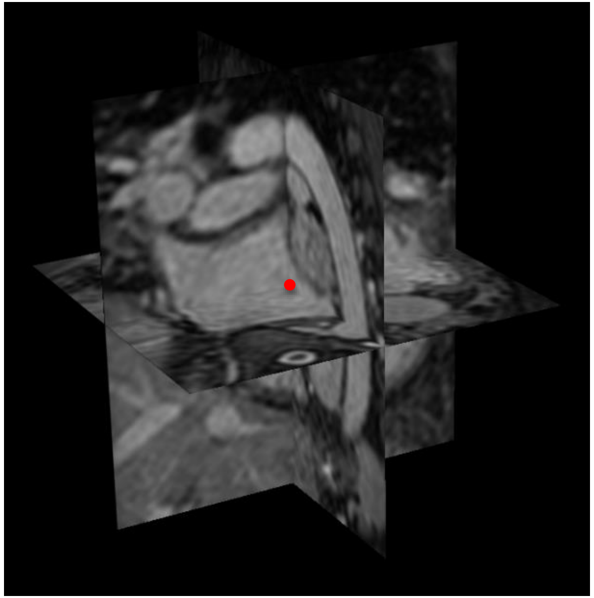


Multi-stage 2D + 3D



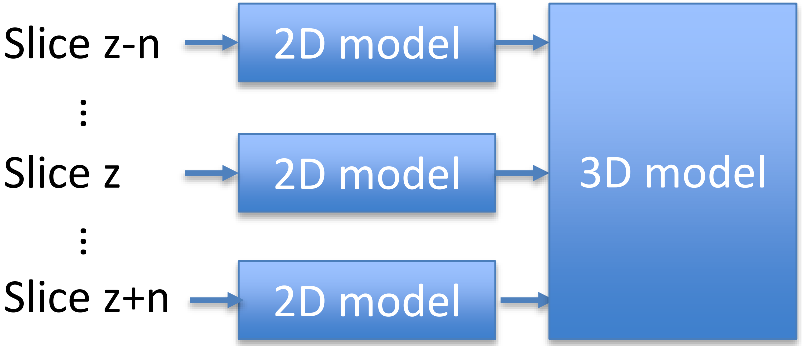
Previous methods to circumvent this trade-off

Triplanes



Still not full 3D

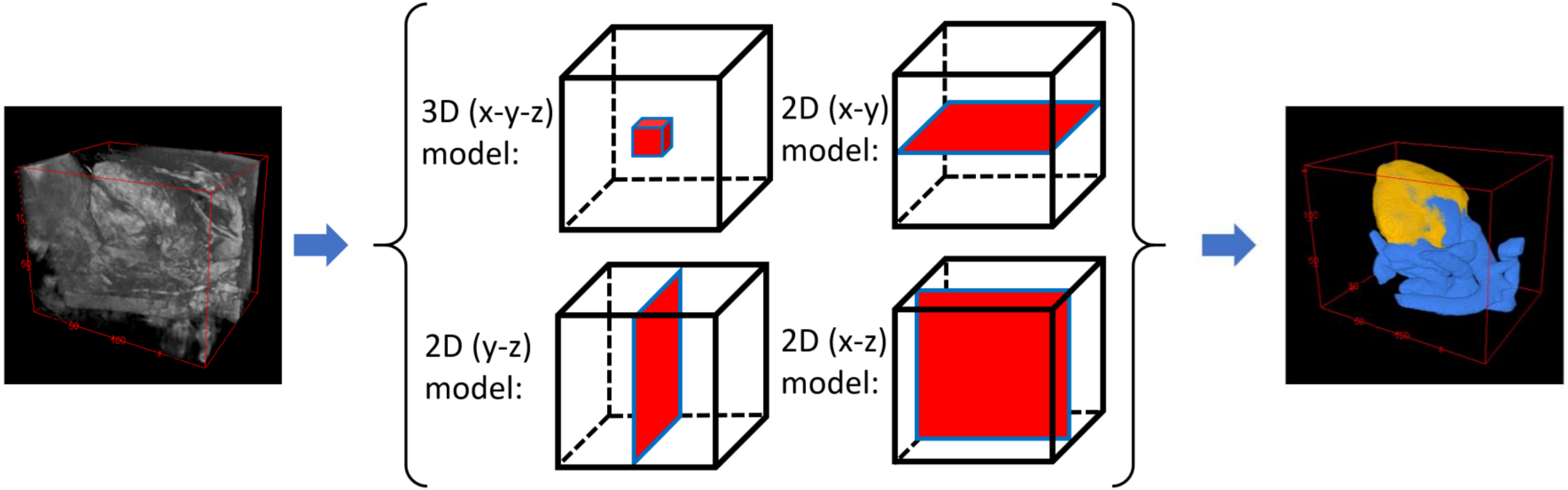
Multi-stage 2D + 3D



2D results may dominate

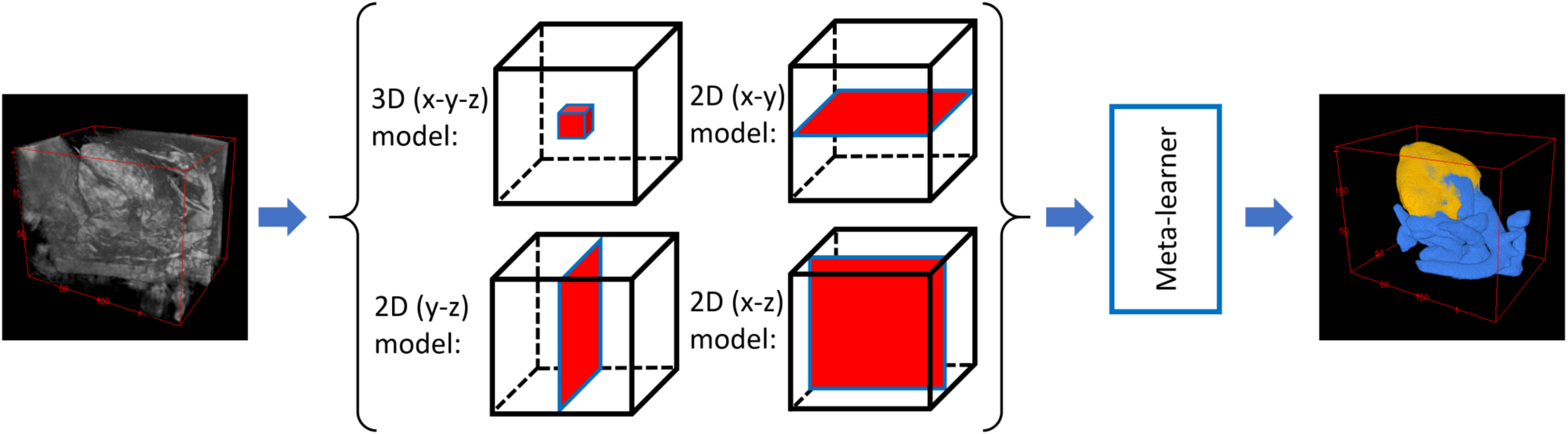
Our idea: Ensemble learning

- Ensemble learning: Unifying the merits of multiple **base-learners**
  - 2D models (large field of view)
  - 3D models (inter-slice information)



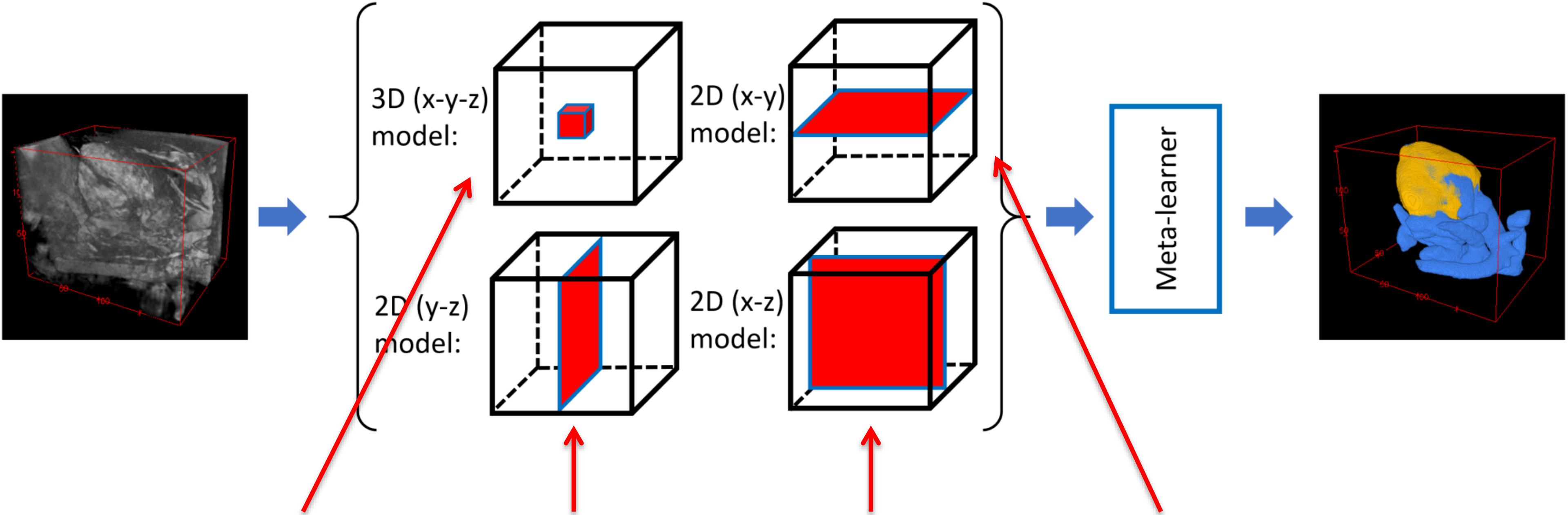
Base-learners

Our approach: Diverse base-learners + meta-learner



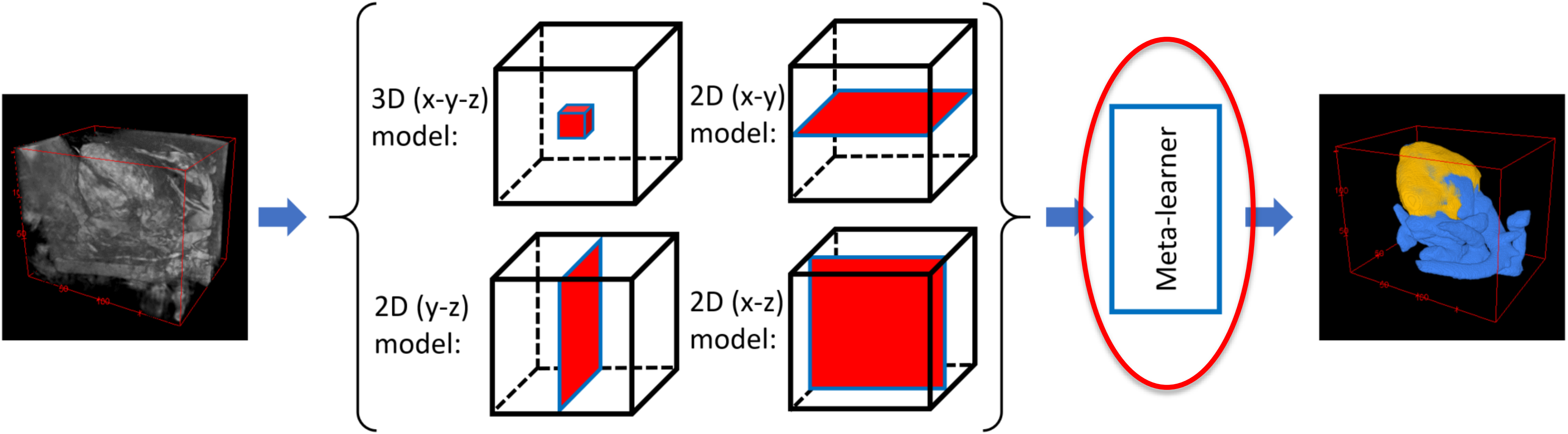
- Diverse base-learners: Each has a unique geometric view of the 3D data
- Meta-learner: Learn how to combine base-learners (instead of simple averaging/voting)

Our approach: Diverse base-learners



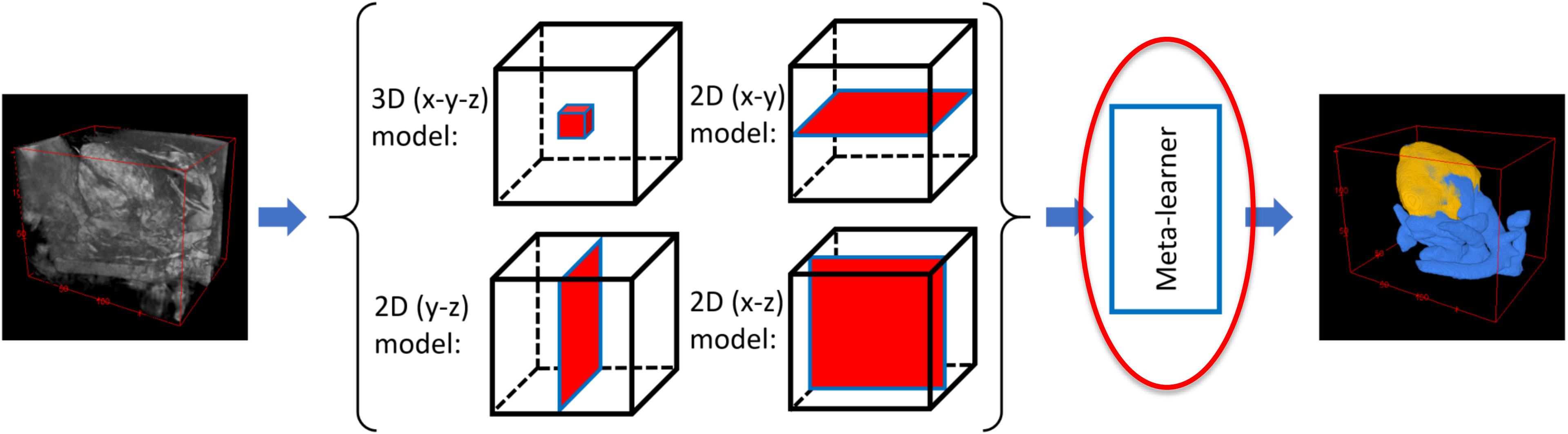
- Diverse base-learners: Each has a unique geometric view of the 3D data
- Meta-learner: Learn how to combine base-learners (instead of simple averaging/voting)

Our approach: Meta-learner



- Diverse base-learners: Each has a unique geometric view of the 3D data
- Meta-learner: Learn how to combine base-learners (instead of simple averaging/voting)

Our approach: We focus on the meta-learner



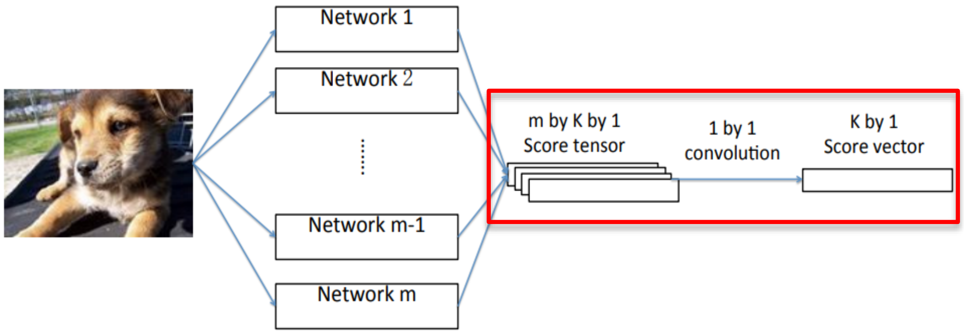
- Diverse base-learners: Each has a unique geometric view of the 3D data
- Meta-learner: Learn how to combine base-learners (instead of simple averaging/voting)
- In our study, we focus on how to design a best possible meta-learner

Previous meta-learners

Learning via fitting the ground truth $$\ell(\theta_{\mathcal{H}})=\sum_{i=1}^{n}\ell_{mce}(\theta_{\mathcal{H}}(x_i,S(PL_i)),GT_i)$$

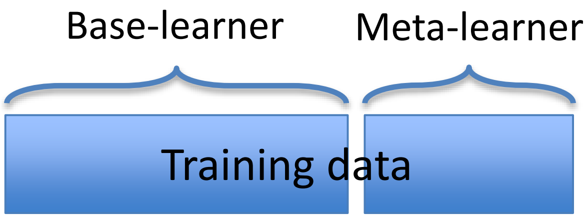
A key issue: Overfitting

Linear combination [1]



Limited capability

Super learner [2]



Not practical when training data are scarce

Our need: A powerful meta-learner with a new scheme to reduce overfitting

[1] The relative performance of ensemble methods with deep convolutional neural networks for image classification
[2] Super learner. Statistical applications in genetics and molecular biology

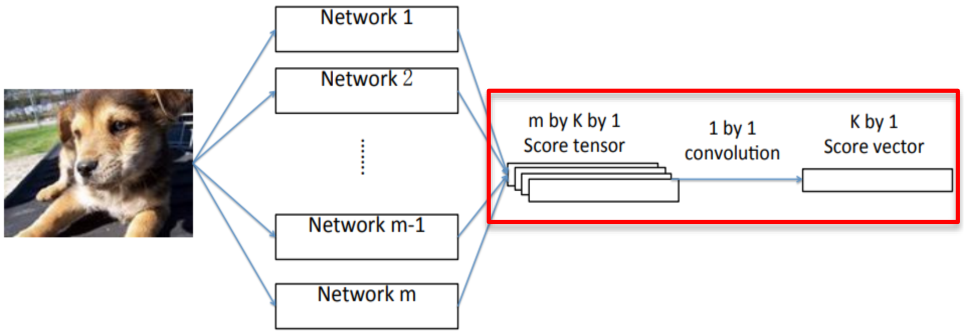
Previous meta-learners

Learning via fitting the ground truth

$$\ell(\theta_{\mathcal{H}})=\sum_{i=1}^{n}\ell_{mce}(\theta_{\mathcal{H}}(x_i, S(PL_i)),GT_i)$$

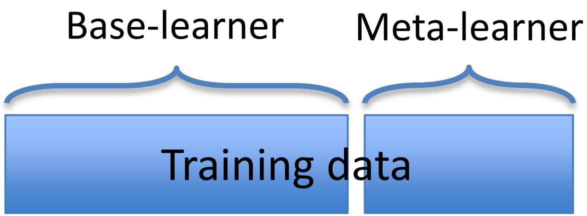
A key issue: Overfitting

Linear combination [1]



Limited capability

Super learner [2]



Not practical when training data are scarce

Our need: A $\boxed{\text{powerful meta-learner}}$ with a new scheme to reduce overfitting

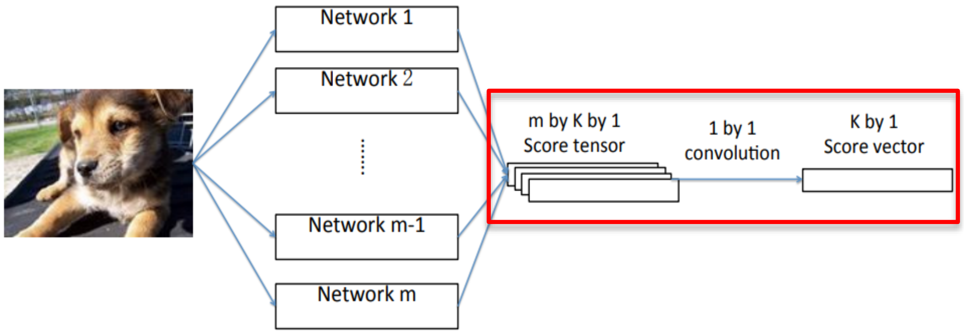
Part I

Previous meta-learners

Learning via fitting the ground truth $$\ell(\theta_{\mathcal{H}})=\sum_{i=1}^{n}\ell_{mce}(\theta_{\mathcal{H}}(x_i,S(PL_i)),GT_i)$$

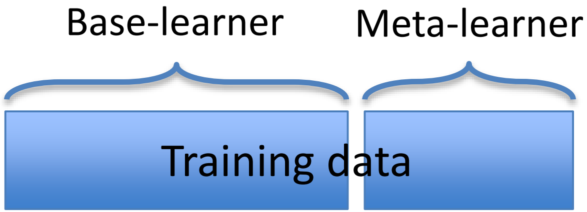
A key issue: Overfitting

Linear combination [1]



Limited capability

Super learner [2]



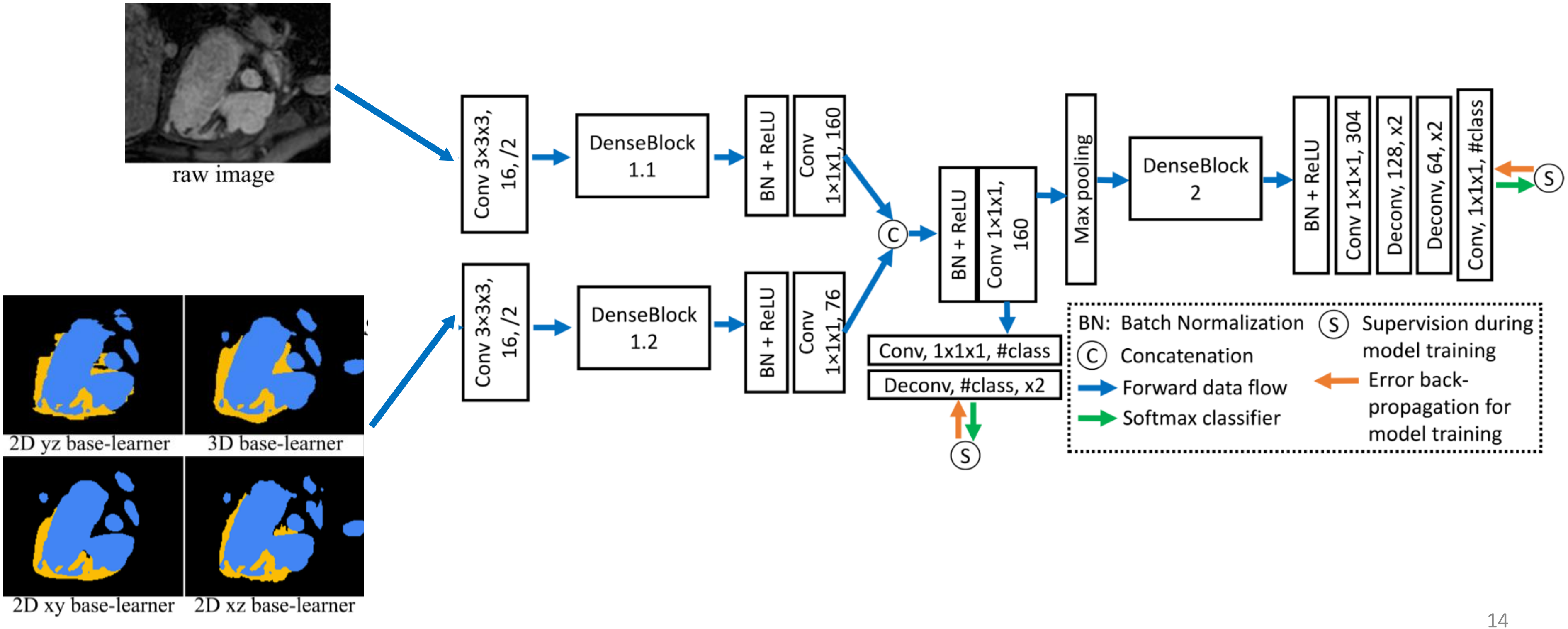
Not practical when training data are scarce

Our need: A powerful meta-learner with a $\boxed{\text{new scheme to reduce overfitting}}$

Part II

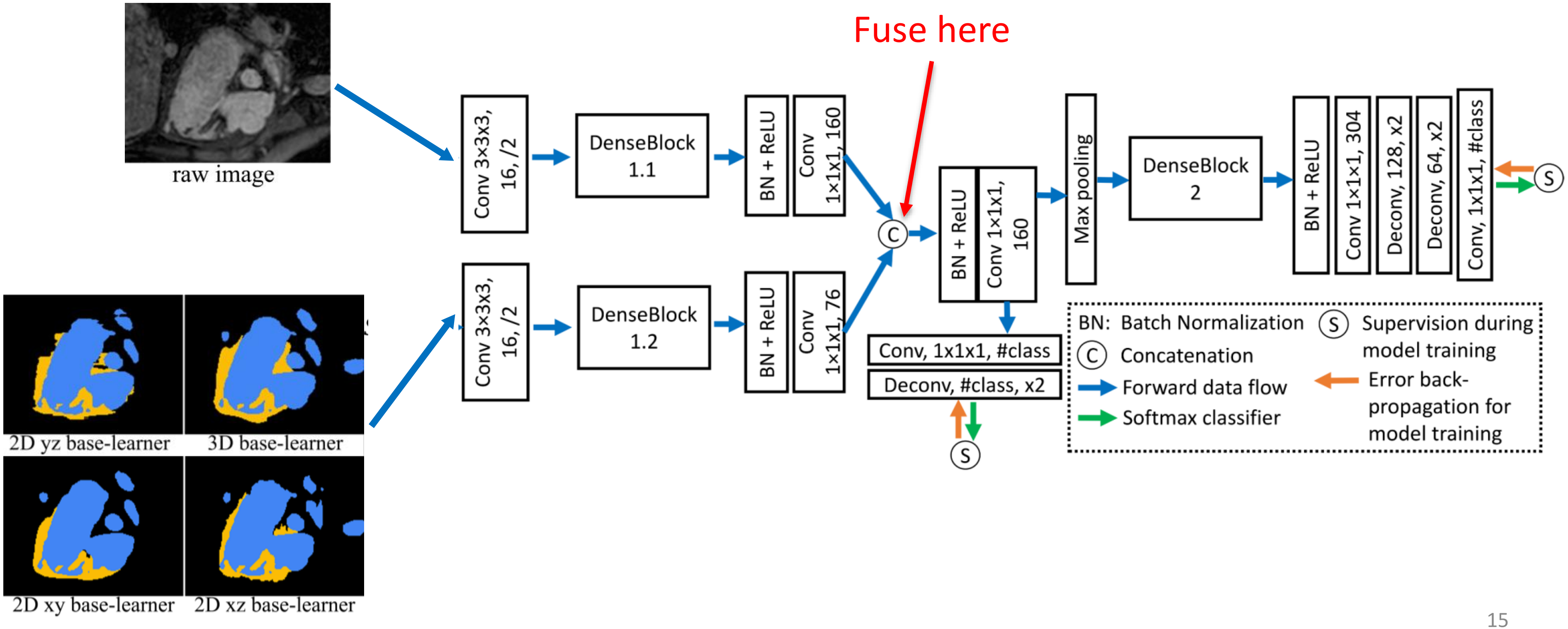
Part I: Our meta-learner structure

- To capture knowledge of each base-learner, we design an FCN based meta-learner



Part I: Our meta-learner structure

- To capture knowledge of each base-learner, we design an FCN based meta-learner



Part II: A new unsupervised training scheme

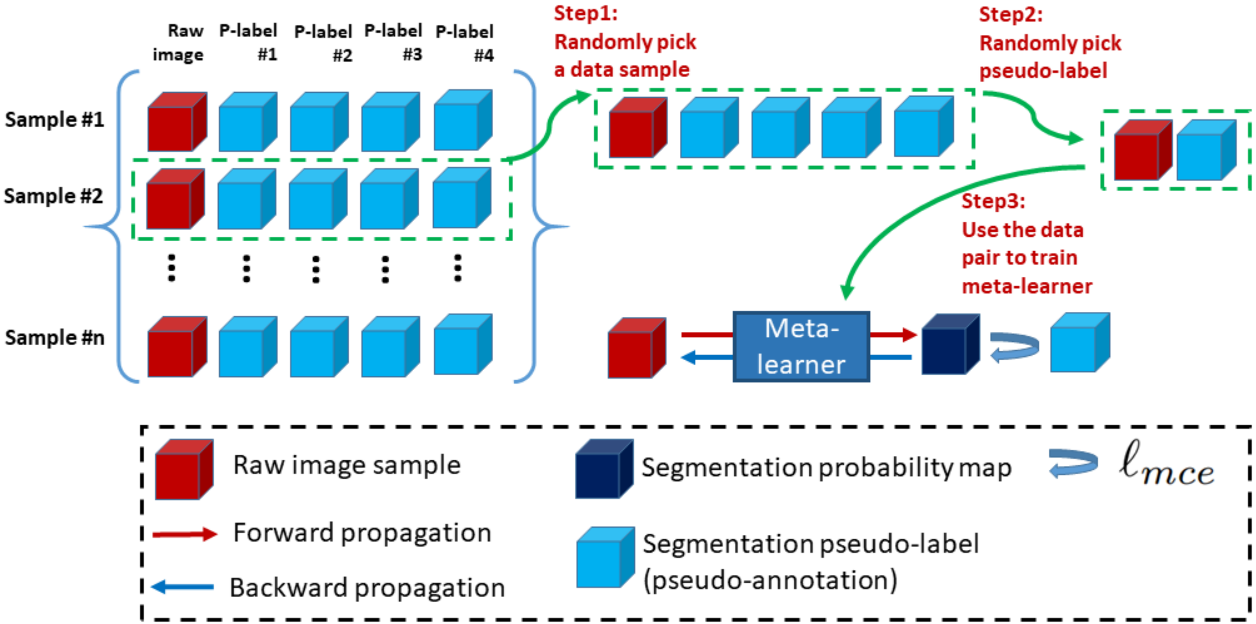
UNIVERSITY OF NOTRE DAME

- Key ideas to **reduce overfitting**
  - Achieve ensemble by finding a balance among base-learners' results (instead of directly fitting the ground truth)
  - Utilize abundant unlabeled raw images
- Our solution:
  - View base-learners' results as pseudo-labels ("fake" ground truth)
  - Find a balance via network training

$$\ell(\theta_{\mathcal{H}}) = \sum_{i=1}^{n} \ell_{mce}(\theta_{\mathcal{H}}(x_i, S(PL_i)), \cancel{GT_i})$$

$$\ell(\theta_{\mathcal{H}}) = \sum_{i=1}^{n} \sum_{j=1}^{m} \ell_{mce}(\theta_{\mathcal{H}}(x_i, S(PL_i)), f_j(x_i))$$

Part II: Training scheme 1 — Random fit



Algorithm 1: Random-fit

Input: $(x_i, PL_i = \{f_1(x_i), f_2(x_i), \ldots, f_m(x_i)\}, S(PL_i))$, $i = 1, 2, \ldots, n$

Output: A trained meta-learner $\mathcal{H}$

initialize a meta-learner $\mathcal{H}$ with random weights;

mini-batch = $\emptyset$

while *stopping condition not met* **do**

  for k = *1 to batch-size* **do**

    $p = rand\text{-}int(1, n)$

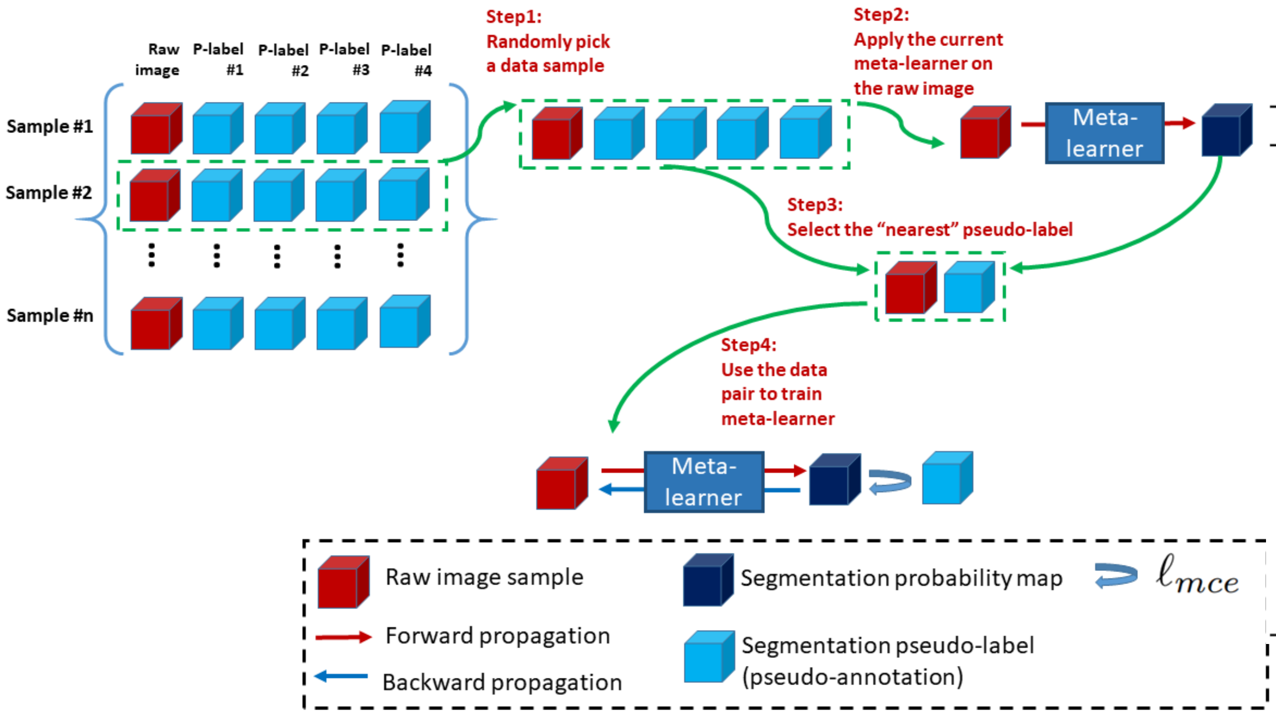
    $q = rand\text{-}int(1, m)$

    add training sample $\{(x_p, S(PL_p)), f_q(x_p)\}$ to the mini-batch

  update $\mathcal{H}$ using training samples in the mini-batch with forward and backward propagation

  mini-batch = $\emptyset$

Part II: Training scheme 2 — Nearest neighbor fit (NN-fit)



Algorithm 2: Nearest-neighbor-fit (NN-fit)

Input: $(x_i, PL_i = \{f_1(x_i), f_2(x_i), \dots, f_m(x_i)\}, S(PL_i))$, $i = 1, 2, \dots, n,$ meta-learner $\mathcal{H}$ (obtained from random-fit)

Output: A refined meta-learner $\mathcal{H}$

mini-batch = $\emptyset$

while *stopping condition not met* **do**

  for k = *1 to batch-size* **do**

    $p = rand\text{-}int(1, n)$

    $\hat{y} = \mathcal{H}(x_p, S(PL_p))$

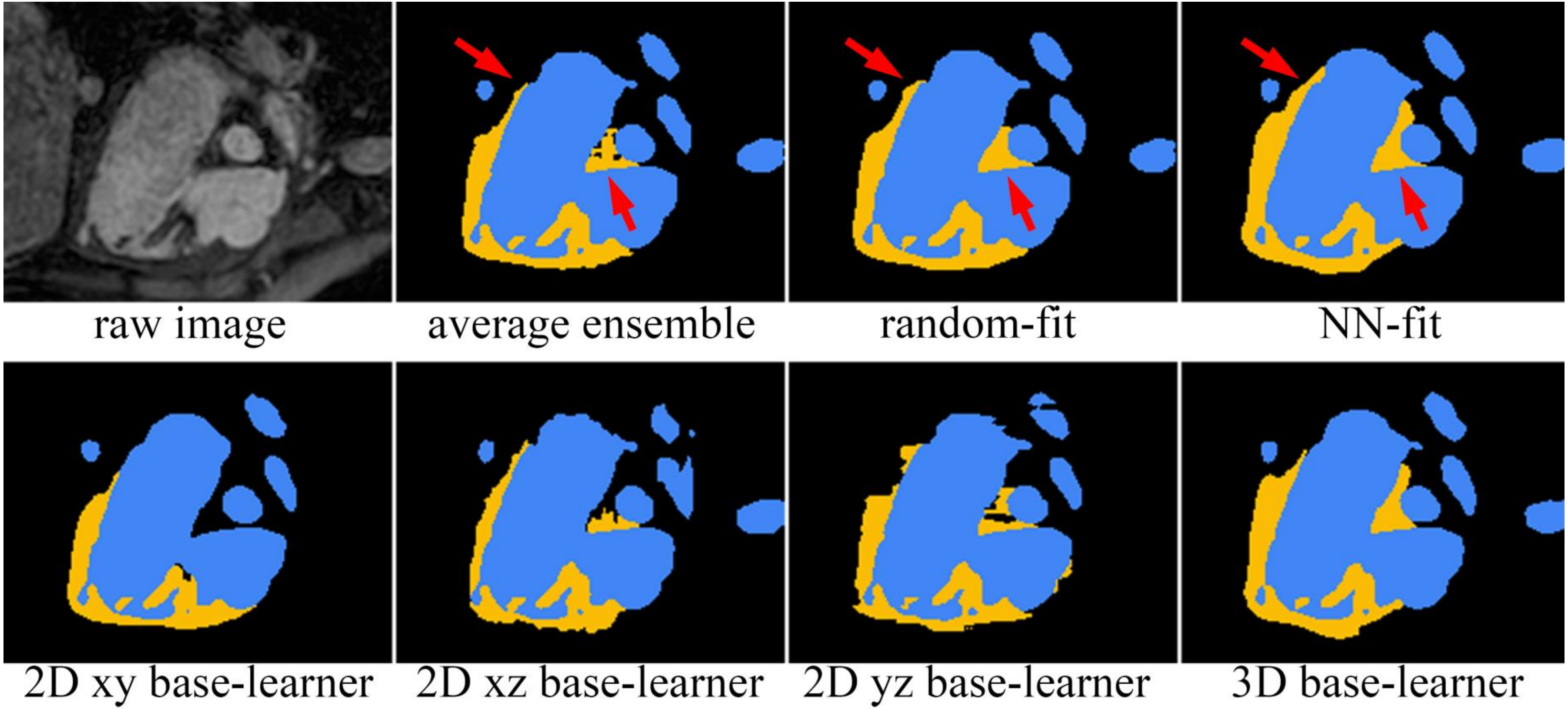
    $\hat{q} = \arg\min_{q=1,2,\dots,m} \mathcal{L}_{mce}(\hat{y}, f_q(x_p))$

    add training sample $\{(x_p, S(PL_p)), f_{\hat{q}}(x_p)\}$ to the mini-batch

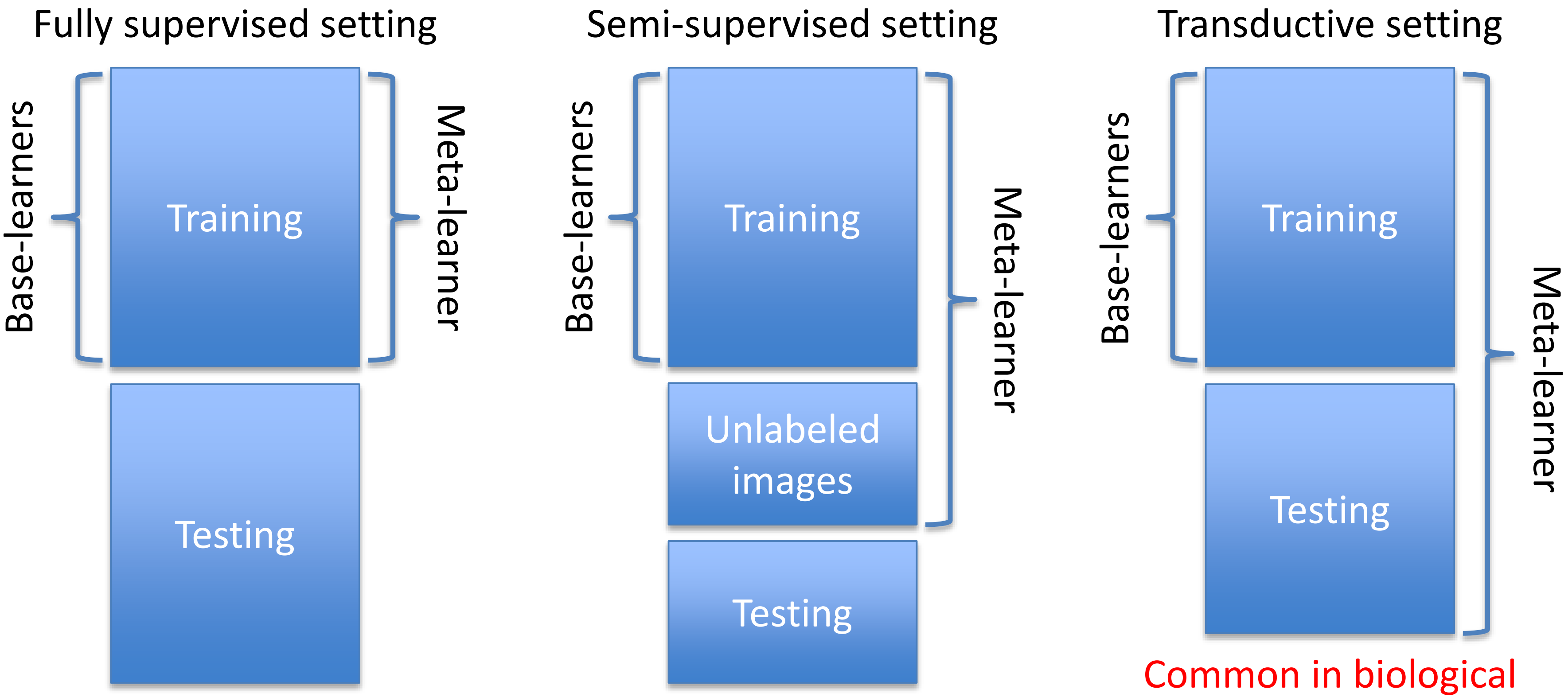
  update $\mathcal{H}$ using training samples in the mini-batch with forward and backward propagation

  mini-batch = $\emptyset$

Visual comparison



Applications



Common in biological discovery applications

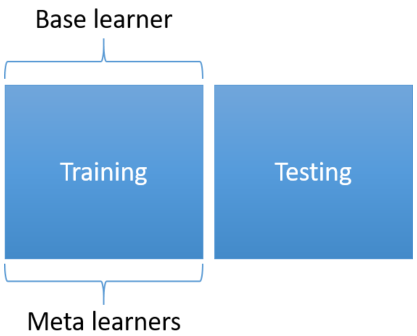
Quantitative comparison — Fully supervised setting

HVSMR 2016 Challenge Dataset

Method	Overall score	
3D U-Net (Çiçek et al. 2016)	-0.419	← 3D models
VoxResNet (Chen et al. 2017)	-0.202	
DenseVoxNet (Yu et al. 2017)	-0.161	
Wolterink *et al.*(Wolterink et al. 2016)	-0.036	← Tri-planar
VFN* (Xia et al. 2018)	0.108	← Multi-stage 2D+3D
Base learner 2D (xy)	0.13	
Base learner 2D (xz)	-0.098	
Base learner 2D (yz)	0.108	
Base learner 3D	0.13	
Average ensemble	0.2	
Our meta-learner (only training data)	0.215	

Mouse Piriform Cortex Dataset

Method	V^{Rand}_{Fscore}
N4 (Ciresan et al. 2012)	0.9304
VD2D (Lee et al. 2015)	0.9463
VD2D3D (Lee et al. 2015)	0.9720
M^2FCN (Shen et al. 2017)	0.9866
Our 2D base-learner	0.9948
Our 3D base-learner	0.9956
Average ensemble of 2D and 3D	0.9959
Random-fit (only training data)	0.9963
NN-fit (only training data)	0.9967



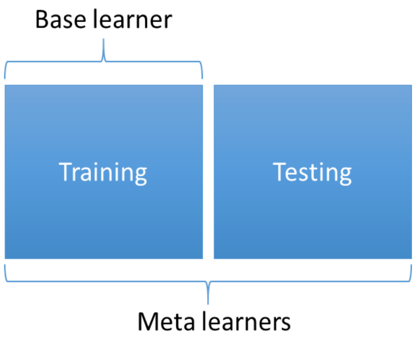
Quantitative comparison — Transductive setting

HVSMR 2016 Challenge Dataset

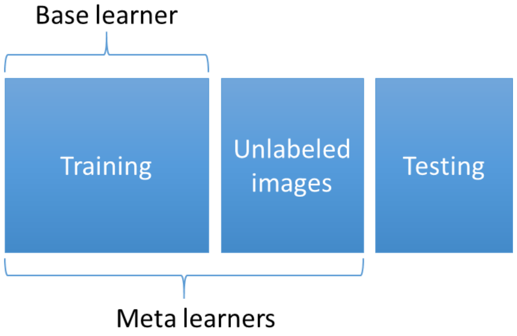
Method	Overall score
3D U-Net (Çiçek et al. 2016)	-0.419
VoxResNet (Chen et al. 2017)	-0.202
DenseVoxNet (Yu et al. 2017)	-0.161
Wolterink *et al.*(Wolterink et al. 2016)	-0.036
VFN* (Xia et al. 2018)	0.108
Base learner 2D (xy)	0.13
Base learner 2D (xz)	-0.098
Base learner 2D (yz)	0.108
Base learner 3D	0.13
Average ensemble	0.2
Our meta-learner (only training data)	0.215
Our meta-learner (transductive)	**0.234**

Mouse Piriform Cortex Dataset

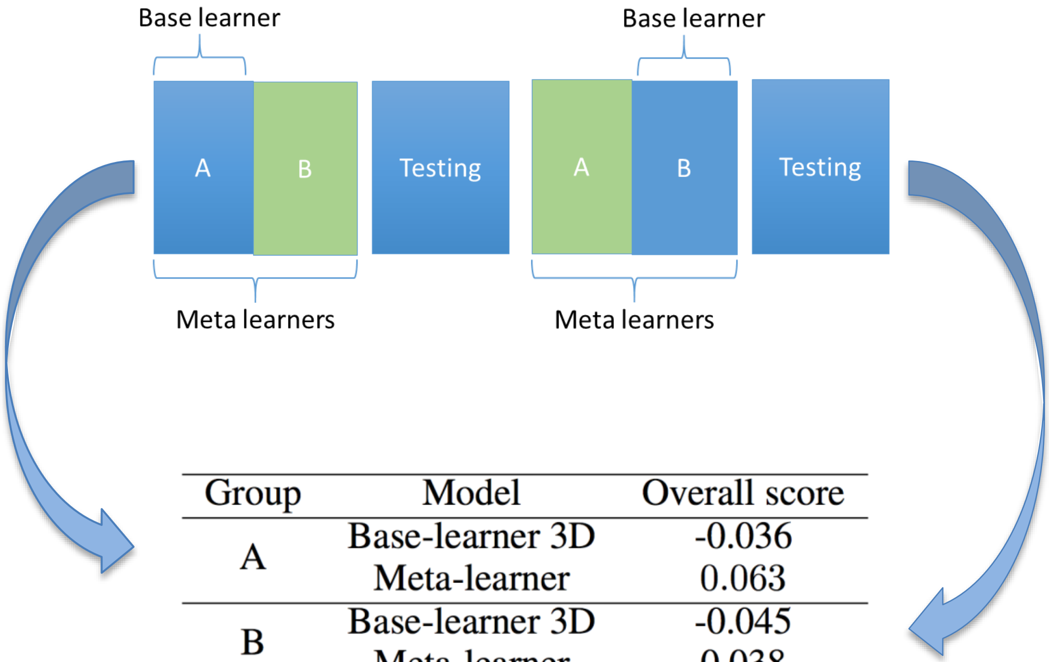
Method	V^{Rand}_{Fscore}
N4 (Ciresan et al. 2012)	0.9304
VD2D (Lee et al. 2015)	0.9463
VD2D3D (Lee et al. 2015)	0.9720
M^{2}FCN (Shen et al. 2017)	0.9866
Our 2D base-learner	0.9948
Our 3D base-learner	0.9956
Average ensemble of 2D and 3D	0.9959
Random-fit (only training data)	0.9963
NN-fit (only training data)	0.9967
Random-fit (transductive)	0.9967
NN-fit (transductive)	**0.9970**



Quantitative comparison — Semi-supervised setting



HVSMR 2016 Challenge Dataset



Group	Model	Overall score
A	Base-learner 3D	-0.036
	Meta-learner	0.063
B	Base-learner 3D	-0.045
	Meta-learner	0.038

Quantitative comparison — Ablation study

- HVSMR 2016 Challenge Dataset

Setting	Inputs		Supervision of training set	Transductive learning	Supervision of testing set	Training		Overall score
	raw image (x_i)	$S(PL_i)$				random fit	NN fit	
S1		✓	GT					0.075
S2	✓	✓	GT					0.192
S3	✓	✓	GT	✓	PL	✓		0.217
S4	✓	✓	GT + PL	✓	PL	✓		0.205
S5	✓	✓	GT + PL	✓	PL	✓	✓	0.224
S6	✓	✓	PL			✓		0.199
S7	✓	✓	PL			✓	✓	0.215
S8	✓	✓	PL	✓	PL	✓		0.218
S9	✓	✓	PL	✓	PL	✓	✓	**0.234**

- Random-fit + NN-fit vs. Random-fit alone (S7 > S6, S5 > S4, S9 > S8)
- Meta training using pseudo-labels vs. ground truth
  - Only training data (S7 > S2)
  - Transductive settings: (1) only GT (S3); (2) GT + PL (S4 & S5); (3) only PL (S8)

Summary & conclusions

UNIVERSITY OF NOTRE DAME

- Introduce a new **ensemble learning framework** to segment 3D biomedical images
- Propose to use **heterogeneous base learners** with different geometric views of 3D data
- Propose to use **pseudo labels** and design a **sophisticated meta learner** to distillate knowledge from base learners
- Propose two optimization strategies (**Random-fit & NN-fit**) for training the meta learner
- Empirical experiments demonstrate that overfitting is alleviated considerably

Thank you!

Q & A