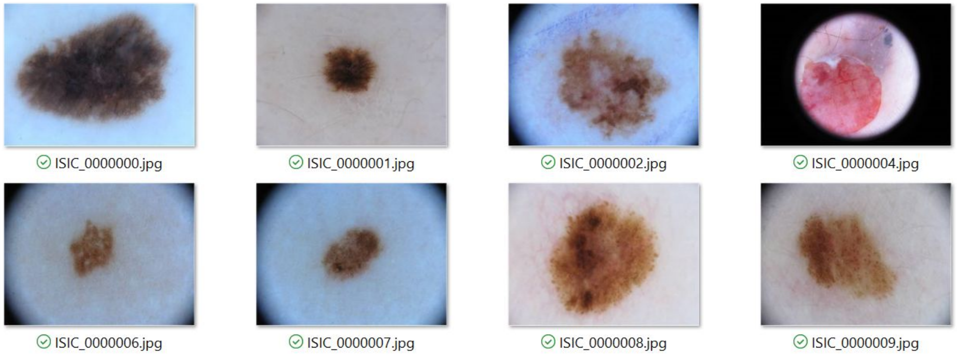
ELEC4010N Final Project

Semi-Supervised Classification
&
Domain Generalization on Fundus Images

Semi-Supervised Classification

Data

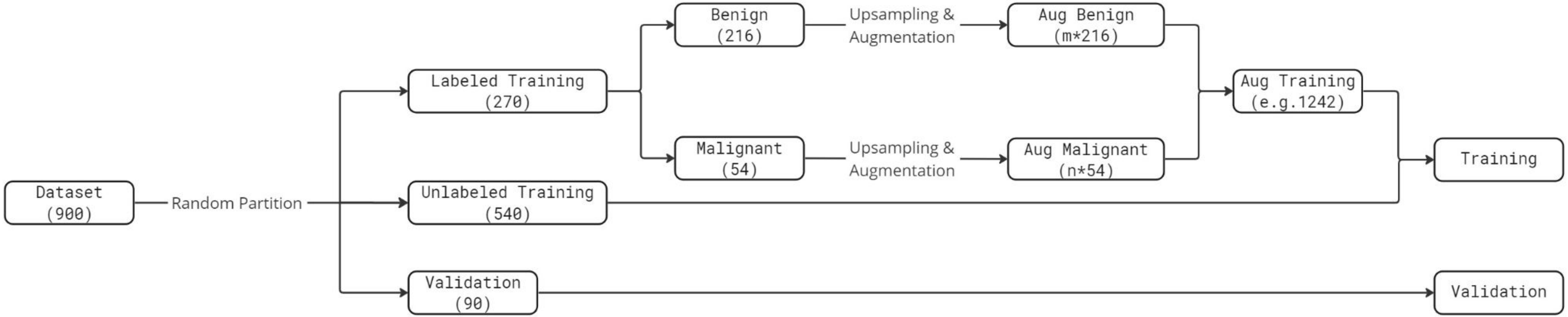


Dataset: ISBI2016_ISIC_Part3_Training_Data

Randomly partition **900** images into **labeled training (270)**, **unlabeled training (540)**, **validation (90)**

Class imbalance problem exists!

We are free to do **upsampling** and **augmentation** only on training data



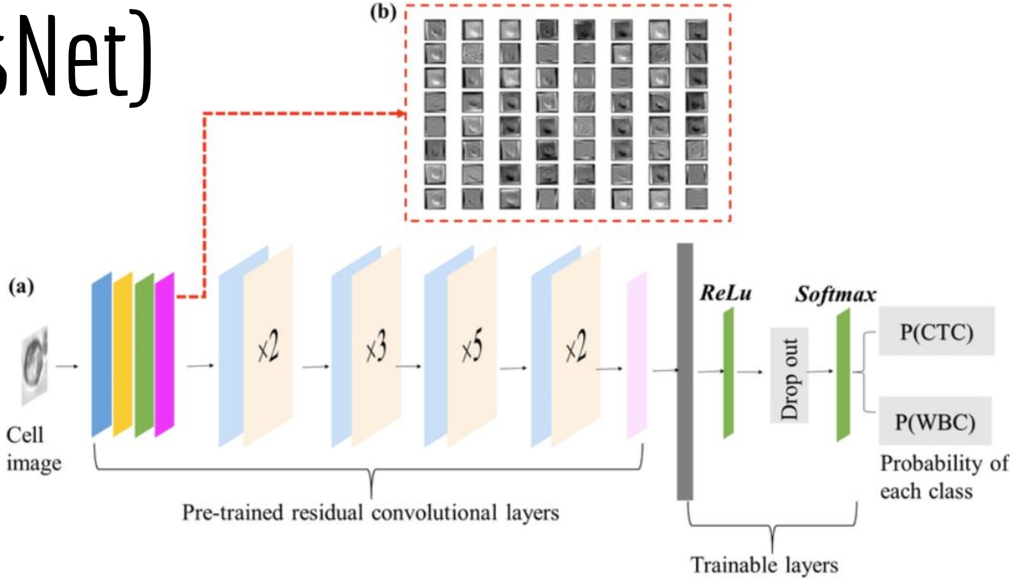
Residual Network (ResNet)

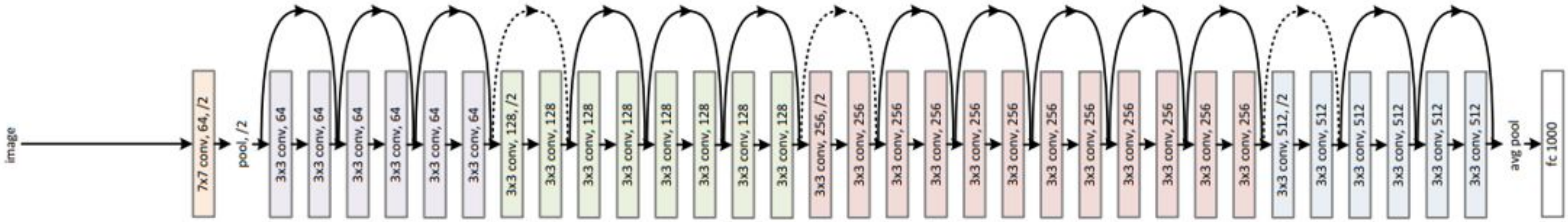
Used for supervised **binary classification**

Will be used as **Student Model**

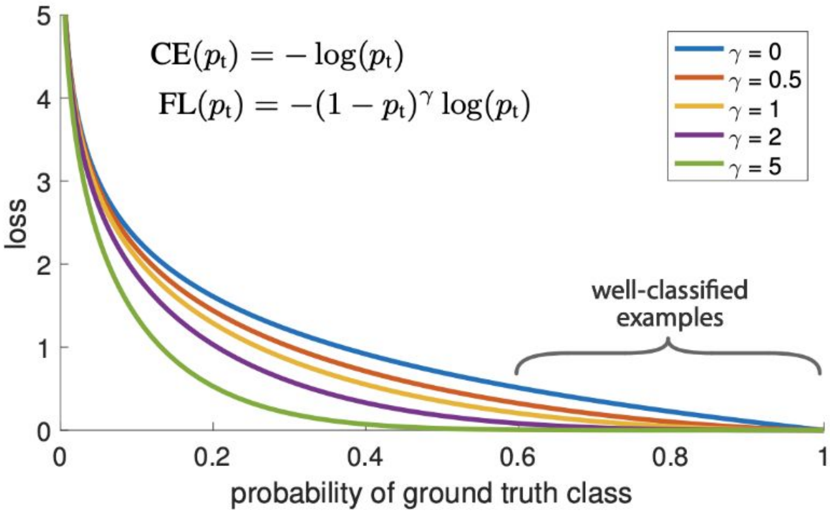
Pretrained

Dropout with **p = 0.5**





BCE Focal Loss



Class imbalance problem exists! **Focal Loss**

BCE Focal Loss = Combination of BCE Loss & Focal Loss

$$\mathrm{BCELoss}(y, \bar{y}) = -(y\log(\bar{y}) + (1-y)\log(1-\bar{y}))$$

$$\mathrm{FocalLoss}(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t)$$

$$\mathrm{BCEFocalLoss}\ (p_t) = -(\alpha_t(1-p_t)^\gamma \log(p_t) + (1-\alpha_t)p_t^\gamma \log(1-p_t))$$

Gamma is used to control **how important the minor class is**

Alpha should be about the **ratio of the classes**, ratio = alpha : 1 - alpha, e.g. alpha = 0.75 for 3:1

Mean Teacher Model

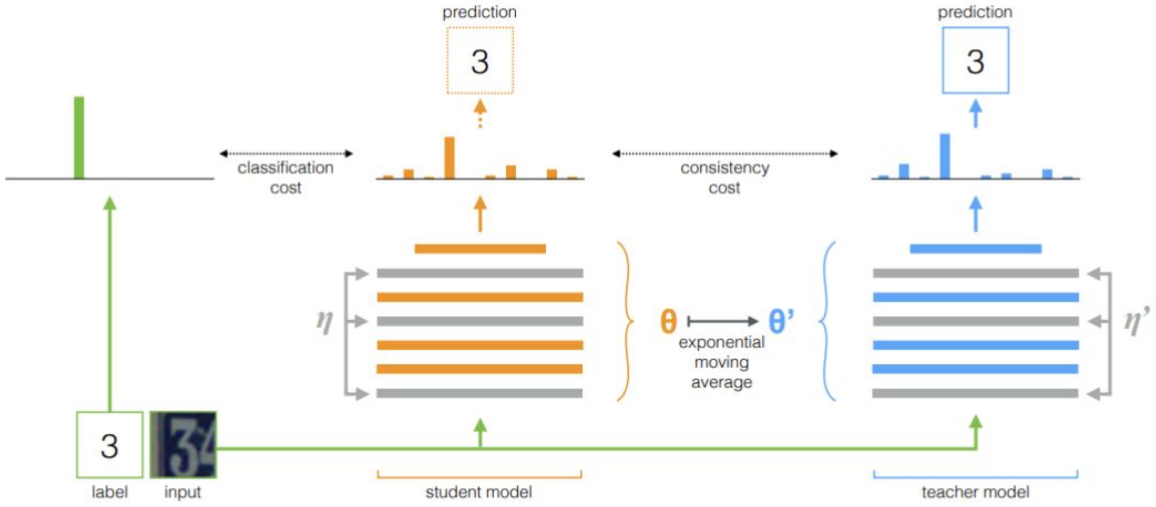
```python
# Mean Teacher Model
# Student model would be ResNet50 model
class MeanTeacherModel(nn.Module):
    # Core
    def __init__(self, student_model, ema_decay):
        super().__init__()
        self.student_model = student_model
        self.teacher_model = copy.deepcopy(student_model)
        self.ema_decay = ema_decay
```

Student Model: ResNet, Supervised Loss, predict for unlabeled data

Teacher Model: Deep-copy of Student Model, update the weight by **Exponential Moving Average**, Consistency Loss

Total Loss = Supervised Loss + Consistency Loss

$$\theta'_t = \alpha\theta'_{t-1} + (1-\alpha)\theta_t$$



Training

Supervised Loss: BCE Focal Loss

Consistency Loss: MSE Loss

Class imbalance problem exists!

Use **sigmoid ramp-up** and **variable momentum** to speed up the loss towards consistency loss

```
def update_teacher_model(self, current_epoch, momentum=0.9995):
    # The momentum increases from 0 to ema_decay
    # Useful for improving quickly at the beginning
    momentum = min(1 - 1 / (current_epoch + 1), self.ema_decay)
    with torch.no_grad():
        for student_params, teacher_params in zip(self.student_model.paramet
            teacher_params.data.mul_(momentum).add_((1 - momentum) * student

# Adjust the weight of the consistency loss to rely on teacher's prediction
# The weight factor decreases from 1 to 0 during the first 5 epochs
def sigmoid_rampup(self, current_epoch):
    current_epoch = np.clip(current_epoch, 0.0, 5.0)
    phase = 1.0 - current_epoch / 5.0
    return np.exp(-5.0 * phase * phase).astype(np.float32)

# The weight decreases from 10
def get_consistency_weight(self, current_epoch):
    return 10 * self.sigmoid_rampup(current_epoch)
```

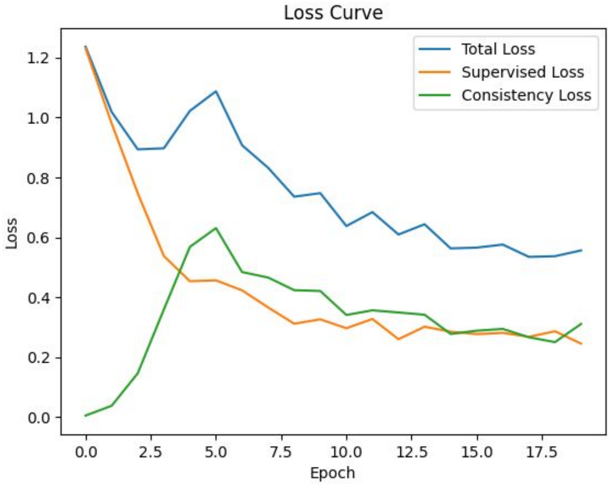
```
# Load ResNet50 as Student model and Mean Teacher model
resnet_model = get_resnet50(pre_trained=True)
base_model = ResnetModel(resnet_model, 1).to(device)
mean_teacher_model = MeanTeacherModel(base_model, ema_decay=0.99).to(device)

# Optimizer, loss functions and scheduler
optimizer = Adam(mean_teacher_model.parameters(), lr=1e-4, weight_decay=1e-5)
supervised_criterion = BCEFocalLoss()
consistency_criterion = nn.MSELoss()

epochs = 15
scheduler = CosineAnnealingLR(optimizer=optimizer, T_max=epochs)
```

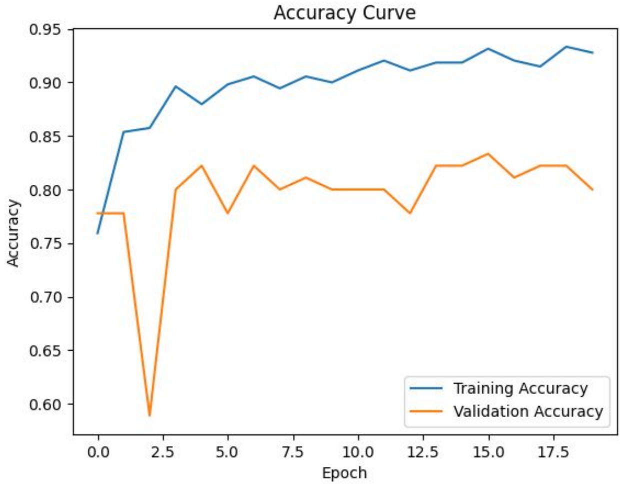
Results

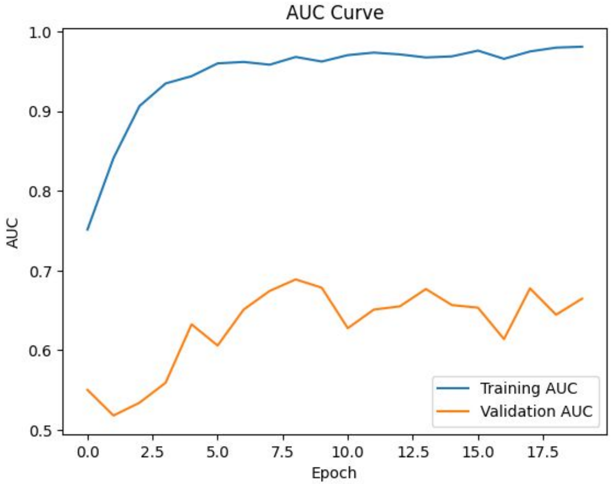
With the **ramp-ups**, the **consistency loss** becomes **significant** fast, better performance

Using upscaling and augmentations, it always shows **good learning in training**, but **sometimes validation is flat**

Balancing the classes might potentially **worsen** the performance somehow







Domain Generalization on Fundus Images

Domain Generalization

- Problems
  - Deep neural network does not generalize too well
  - Out-of-distribution may consider as domain shifting
- Proposed Solution
  - Data Augmentation (Fourier Transform)
    - Phase, Amplitude information
  - Mean Teacher Model
    - Compare student & teacher outputs

Data

Dataset: Fundus Dataset

(Multi-label: Background, Optic Disk, Optic Cup)

Domain 1: Drishti-GS dataset **101** images (50, 51)

Domain 2: RIM-ONE_r3 dataset **159** images (99, 60)

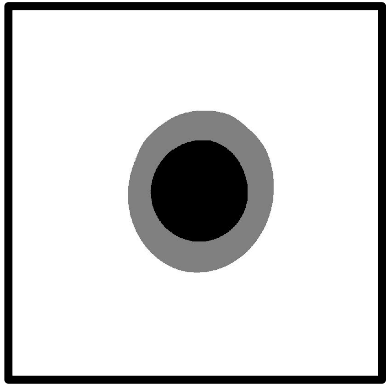
Domain 3: REFUGE training **400** images (320, 80)

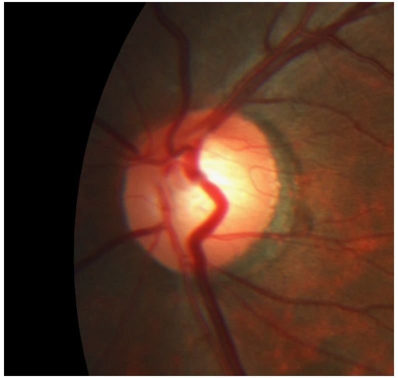
Domain 4: REFUGE val **400** images (320, 80)

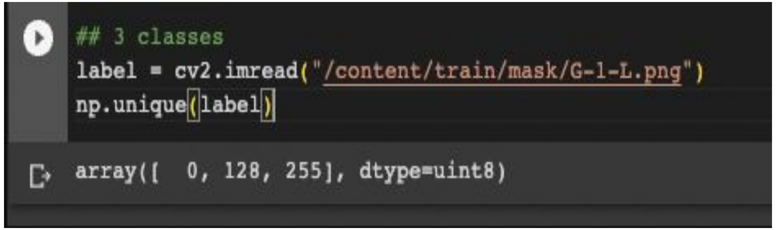
Data Partition:

Train on a **combination of 3** domains and test on the 1 domain

E.g. Train: [Domain1 , Domain2, Domain3], Test: [Domain 4]







Fourier Augmentation

1. Obtain the amplitudes and phases of the images

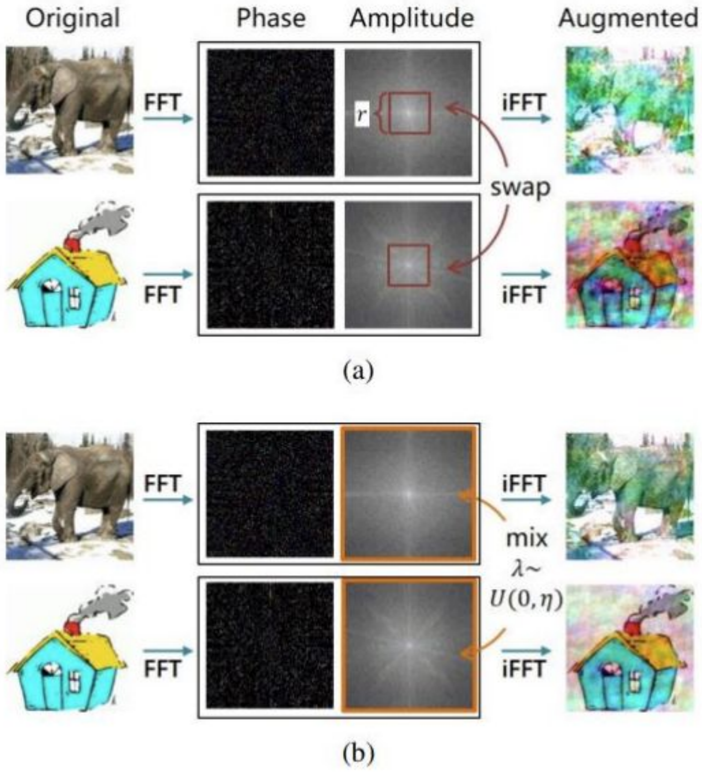
2. **AS strategy (Amplitude Swap)**
   Overwhelming for model to learn

   AM strategy (Amplitude Mixup) by **linear interpolation**

   $$\hat{\mathcal{A}}(x_i^k) = (1-\lambda)\mathcal{A}(x_i^k) + \lambda\mathcal{A}(x_{i'}^{k'})$$

3. Obtain the **soften probability losses** of original & augmented

   $$\mathcal{L}_{\text{cls}}^{\text{ori}} = -y_i^k \log(\sigma(f(x_i^k, \theta)))$$
   $$\mathcal{L}_{\text{cls}}^{\text{ori}} = -y_i^k \log(\sigma(f(x_i^k, \theta)))$$



Fourier Augmented Co-Teacher (FACT)

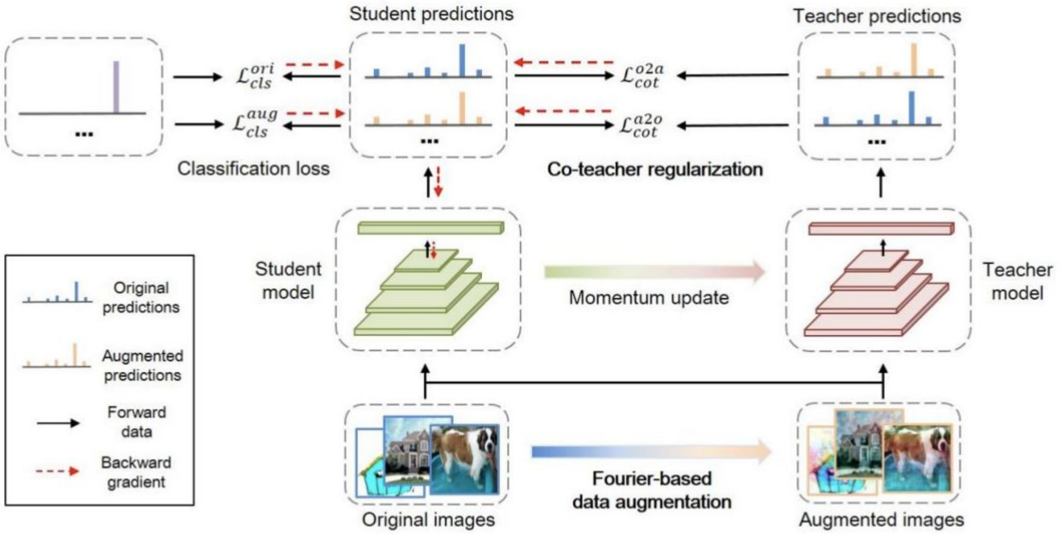
Both **original** data & **Fourier augmented** data were fed into both **Student model** & **Teacher model**

Co-Teacher regularization: Use Kullback-Leibler (KL) divergence to ensure the consistency

$$\mathcal{L}_{\text{cot}}^{a2o} = \mathrm{KL}\left(\sigma\left(f_{\text{stu}}\left(\hat{x}_i^k\right)/T\right) \| \sigma\left(f_{\text{tea}}\left(x_i^k\right)/T\right)\right)$$

$$\mathcal{L}_{\text{cot}}^{o2a} = \mathrm{KL}\left(\sigma\left(f_{\text{stu}}\left(x_i^k\right)/T\right) \| \sigma\left(f_{\text{tea}}\left(\hat{x}_i^k\right)/T\right)\right]\right)$$

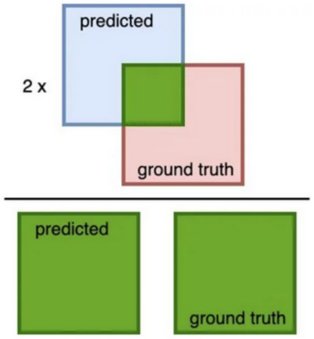
$$\mathcal{L}_{FACT} = \mathcal{L}_{cls}^{ori} + \mathcal{L}_{cls}^{aug} + \beta(\mathcal{L}_{cot}^{a2o} + \mathcal{L}_{cot}^{o2a})$$



Supervised loss: Soft Dice Loss

Total loss = Supervised Loss + Consistency Loss

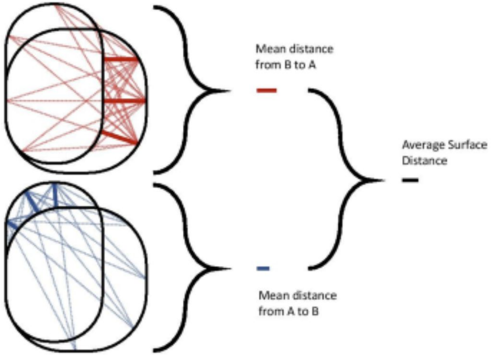
Evaluation Metrics



Dice Loss

$$\text{Dice} = \frac{2TP}{2TP + FP + FN}$$

Average Surface Distance (ASD)



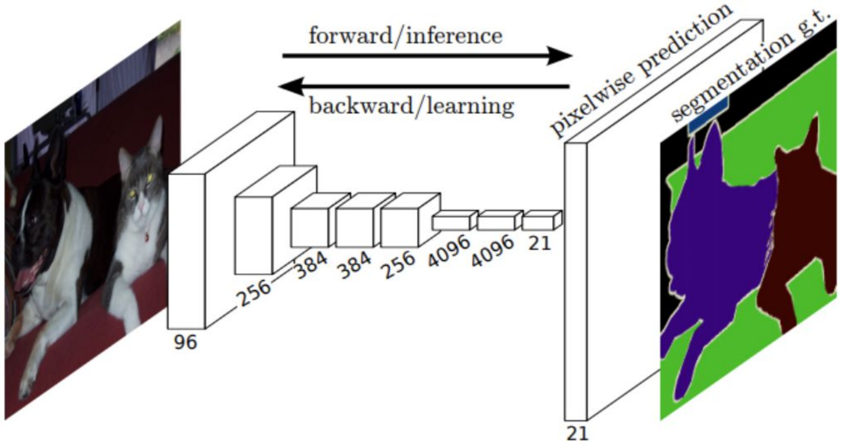
Treat each type of label as binary segmentation mask

$$\text{ASD}(A, B) = \frac{1}{|S(A)| + |S(B)|}\left(\sum_{a \in S(A)} \min_{b \in S(B)} \|a - b\| + \sum_{b \in S(B)} \min_{a \in S(A)} \|b - a\|\right)$$

U-Net

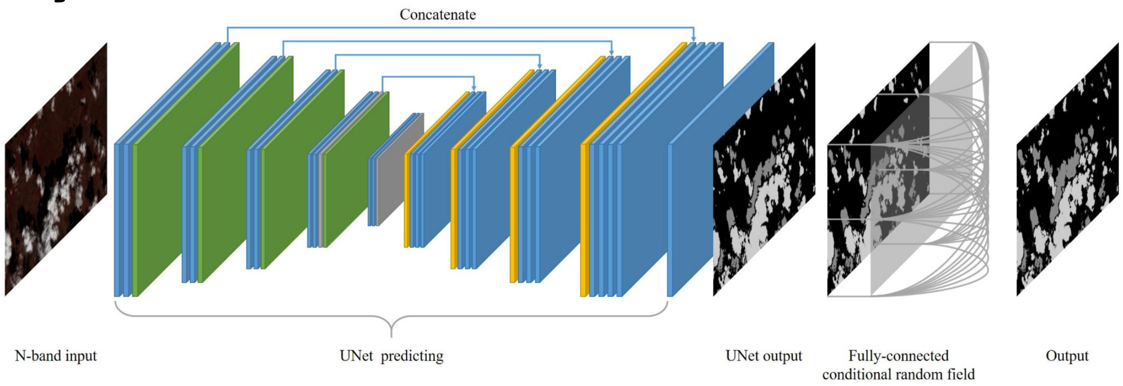
Use for **multi-class segmentation**

- (in_channels=3, out_channels=3)
- Extract important information
- Produce segmentation prediction



As **Student Model** with **pre-trained** encoder weights

- Apply **softmax function**
- [batch_size, **3**, img_size, img_size]



Training

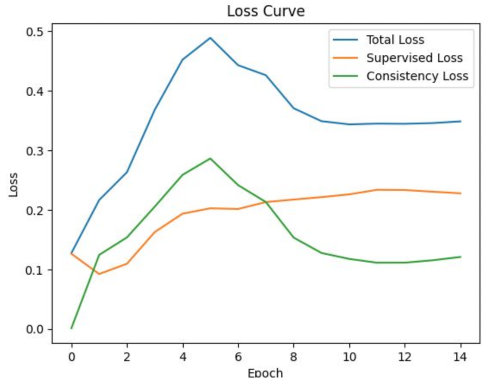
Supervised Loss: MSE Loss

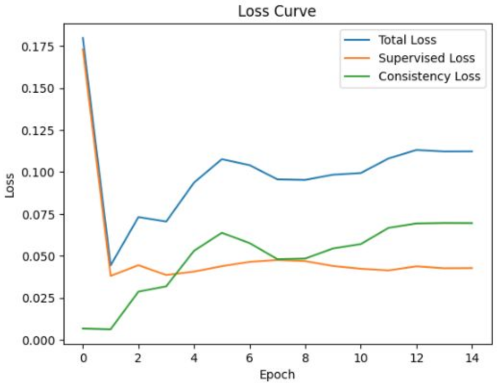
- Taking mean squared error (MSE) between the logit vectors

```
# MSE Loss
loss_ori_tea = consistency_criterion(scores_aug, scores_ori_tea)
loss_aug_tea = consistency_criterion(scores_ori, scores_aug_tea)
```

Consistency Loss: KL Divergence

- Empirically, the consistency losses based on KL divergence are **more able to converge**
- Calculating loss on softened probability distributions

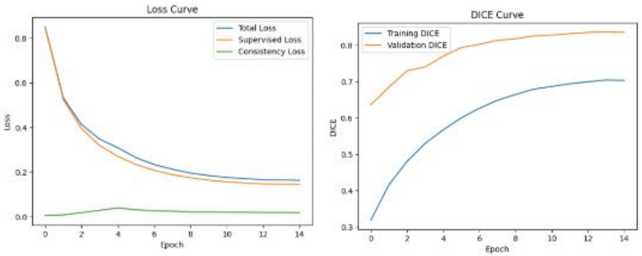




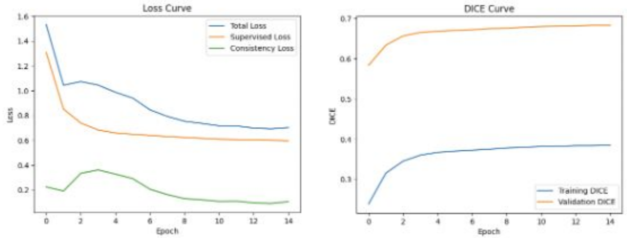
Results

Better than Baseline

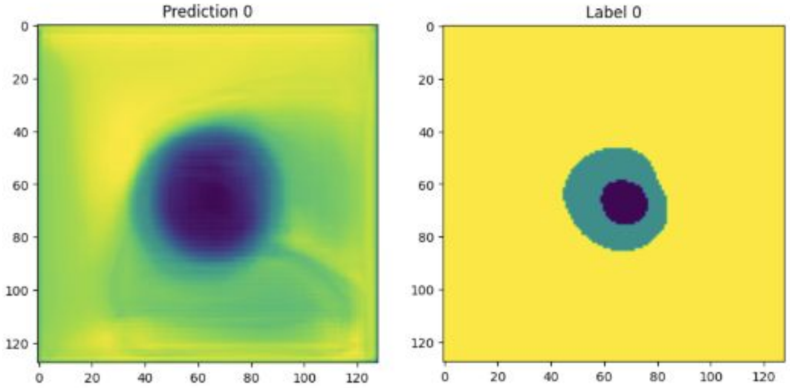
MSE Loss might be **better** than KL Divergence



MSE Loss



KL Divergence



Train	Test	Model	Mean Test Dice	OC Test ASD	OD Test ASD
123	4	Baseline	0.5389	40.0272	31.9848
		FACT	**0.6494**	**36.0975**	**21.2270**
124	3	Baseline	0.5518	**28.2954**	26.8578
		FACT	**0.7138**	31.1686	**5.6759**
134	2	Baseline	0.5512	38.4887	28.7364
		FACT	**0.7476**	**34.0680**	**7.5351**
234	1	Baseline	0.5638	30.8448	24.2397
		FACT	**0.7881**	**25.8820**	**7.3002**

Reference

- Tarvainen, A., Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv:1703.01780*
- Wang, S., Yu, L., Li, K., Yang, X., Fu, C.-W., Heng, P.-A. (2020). DoFE: Domain-oriented Feature Embedding for Generalizable Fundus Image Segmentation on Unseen Datasets. IEEE Transactions on Medical Imaging. (https://github.com/emma-sjwang/Dofe)
- Xu, Q., Zhang, R., Zhang, Y., Wang, Y., Tian, Q. (2021). A Fourier-Based Framework for Domain Generalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). (https://github.com/MediaBrain-SJTU/FACT)
- Laine, S., & Aila, T. (2017). Temporal Ensembling for Semi-Supervised Learning. International Conference on Learning Representations (ICLR). *arXiv:1610.02242*