Deep Adversarial Training for Multi-Organ Nuclei Segmentation in Histopathology Images

Lucky Simarda
2106779503

Problems

Find learning mapping functions G and S between two domains: **M (nuclei mask)** and **N (H&E images)**

1. **Generate synthetic H&E images** that model the distribution of cellular and extracellular spatial featuresrepresented in multiple organs.

$G : M \rightarrow N$

(for synthetic data generation)

2. Use both the synthetic and real histopathology data for training a context-aware CNN that can accurately **segment nuclei**.

$S : N \rightarrow M$

(for nuclei segmentation)

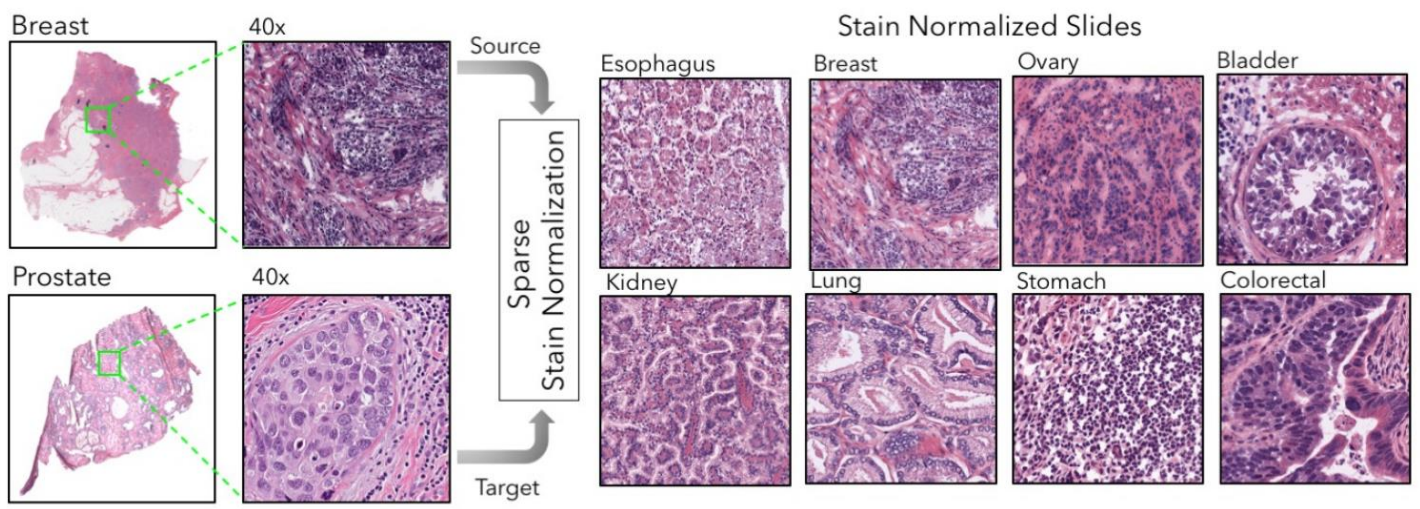
We denote m and n as training examples where $m \in M$ and $n \in N$.

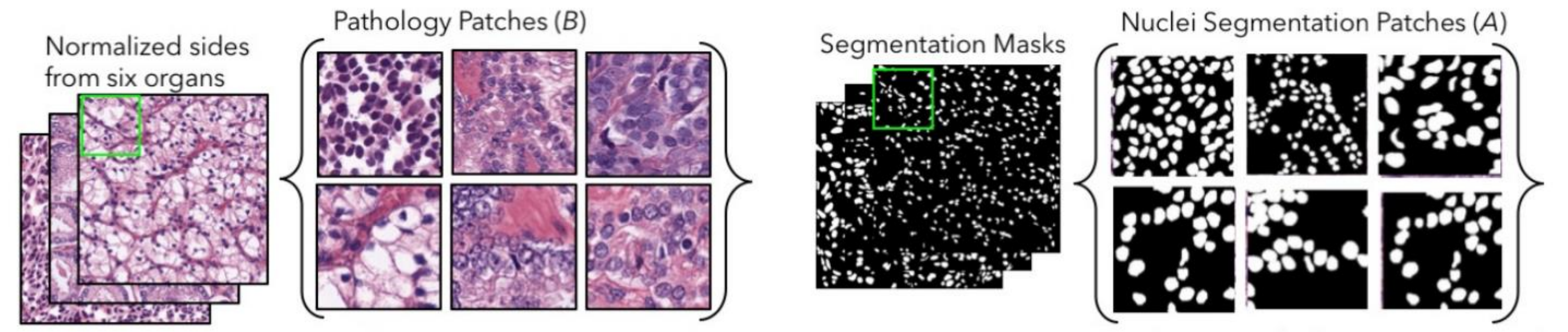
Datasets

Multi organ pathology images (with annotations) of 34 1000 × 1000 pathology images from nine different organs (bladder, colon, stomach, breast, kidney, liver, and prostate, ovary, esophagus)

preprocessing

- Sparse stain normalization
- Decomposition into large patches of 256 x 256
- Training images: four slides from breast, liver, kidney and prostate
- Testing images: two slides from breast, liver, kidney, prostate, bladder, colon and stomach and one slide from esophagus and ovary
- Adding 4.650 synthetically generated images to training images (nuclei segmentation)

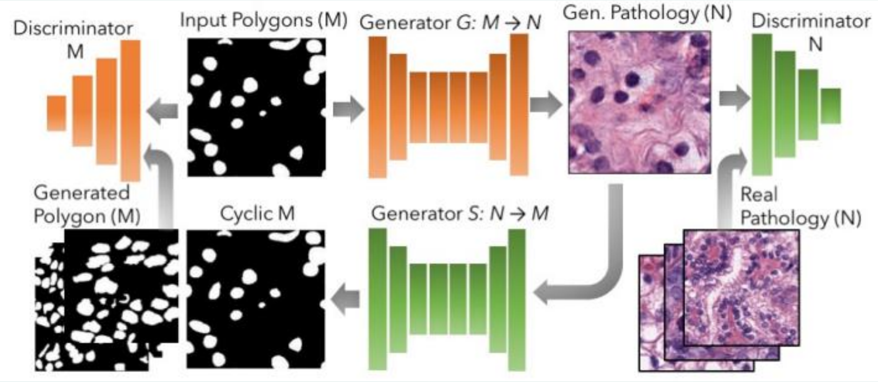

Breast
40x
Source
Prostate
40x
Target
Sparse Stain Normalization
Stain Normalized Slides
Esophagus
Breast
Ovary
Bladder
Kidney
Lung
Stomach
Colorectal


Normalized sides from six organs
Pathology Patches (*B*)
Segmentation Masks
Nuclei Segmentation Patches (*A*)

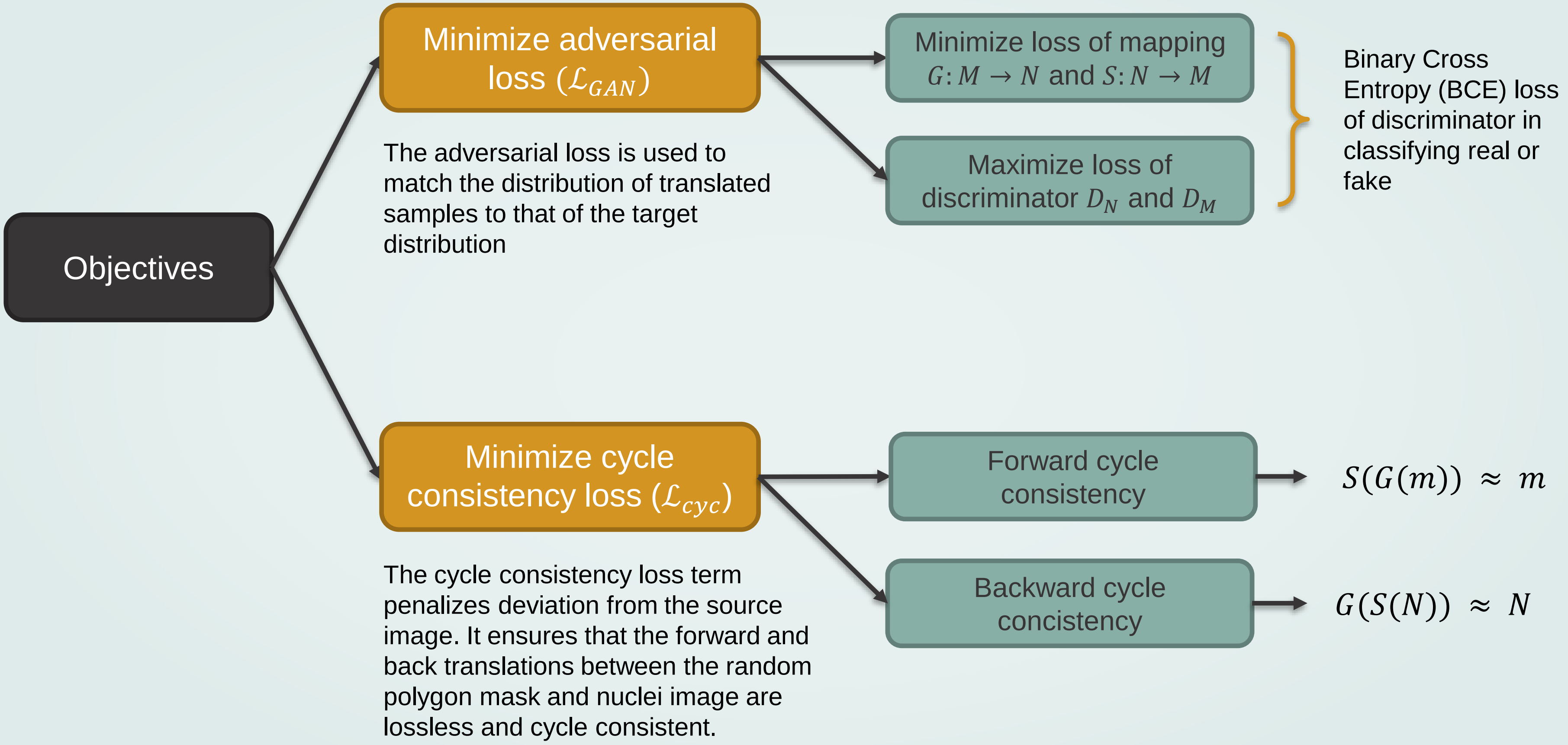
Problem 1: Synthetic data generation

Architectures: dual-GAN (with cycle consistency loss) → cycleGAN
Normalization: spectral normalization (to prevent mode colapse)

The cycleGAN framework learns a mapping between randomly generated polygon masks and unpaired pathology images. Since cycleGAN is based on consistency loss, the setup also learns a reverse mapping from pathology images to corresponding segmentation or polygon masks. **The reverse mapping is used only to train the forward mapping more effectively.**



G: random polygon mask to pathology image generator
S: pathology image to polygon mask generator
D_N: discriminator for G
D_M:discriminator for S



Objectives

Minimize adversarial loss ($\mathcal{L}_{GAN}$)

The adversarial loss is used to match the distribution of translated samples to that of the target distribution

Minimize loss of mapping $G: M \rightarrow N$ and $S: N \rightarrow M$

Maximize loss of discriminator D_N and D_M

Binary Cross Entropy (BCE) loss of discriminator in classifying real or fake

Minimize cycle consistency loss ($\mathcal{L}_{cyc}$)

The cycle consistency loss term penalizes deviation from the source image. It ensures that the forward and back translations between the random polygon mask and nuclei image are lossless and cycle consistent.

Forward cycle consistency

$S(G(m)) \approx m$

Backward cycle concistency

$G(S(N)) \approx N$

Loss of mapping $G: M \rightarrow N$

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, D_N) = \; & \mathbb{E}_{n \sim p_{\text{data}}(n)}[\log D_N(n)] \\ & + \mathbb{E}_{m \sim p_{\text{data}}(m)}[\log(1 - D_N(G(m)))],\end{aligned}$$

Loss of mapping $S: N \rightarrow M$

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(S, D_M) = \; & \mathbb{E}_{m \sim p_{\text{data}}(m)}[\log D_M(m)] \\ & + \mathbb{E}_{n \sim p_{\text{data}}(n)}[\log(1 - D_M(S(n)))].\end{aligned}$$

Loss of mapping $S: N \rightarrow M$

$$\begin{aligned}\mathcal{L}_{cyc}(G, S) = \; & \lambda_n \mathbb{E}_{n \sim p_{\text{data}}(n)}[||G(S(n)) - n||_1] \\ & + \lambda_m \mathbb{E}_{m \sim p_{\text{data}}(m)}[||S(G(m)) - m||_1]\end{aligned}$$

λ controls the importance of the forward and backward cycle constraints. For synthetic data generation, the λ_m term is relaxed.

full objective for synthetic data generation

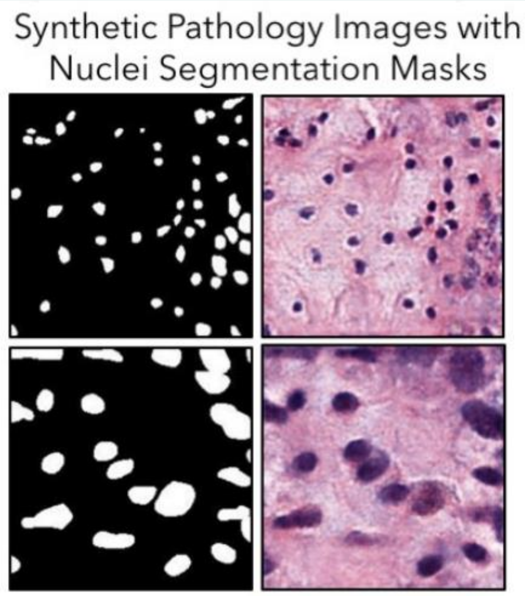
$$\arg\min_{G,S} \arg\max_{D_N, D_M} \mathcal{L}_{\text{GAN}}(G, D_N) + \mathcal{L}_{\text{GAN}}(S, D_M)$$

$$+ \mathcal{L}_{cyc}(G, S)$$

Training Details

- Pytorch 4.0
- $\lambda_n = 70, \lambda_m = 10$
- Optimizer: Adam solver
- Number of epoch: 300
- Normalization: spectral normalization
- Learning rate: 0.0002 (for first 150 epochs) → linearly decayed to 0 (for remaining epochs)
- Weight initialized from a Gaussian distribution with a mean 0 and standard deviation 0.02

Output

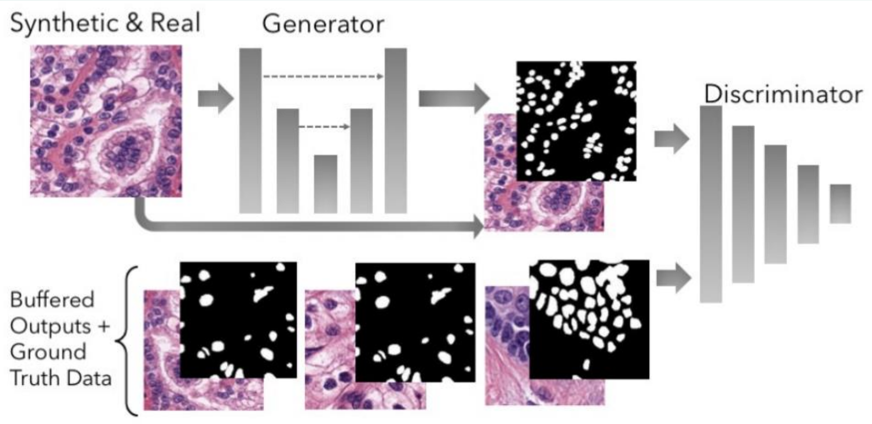


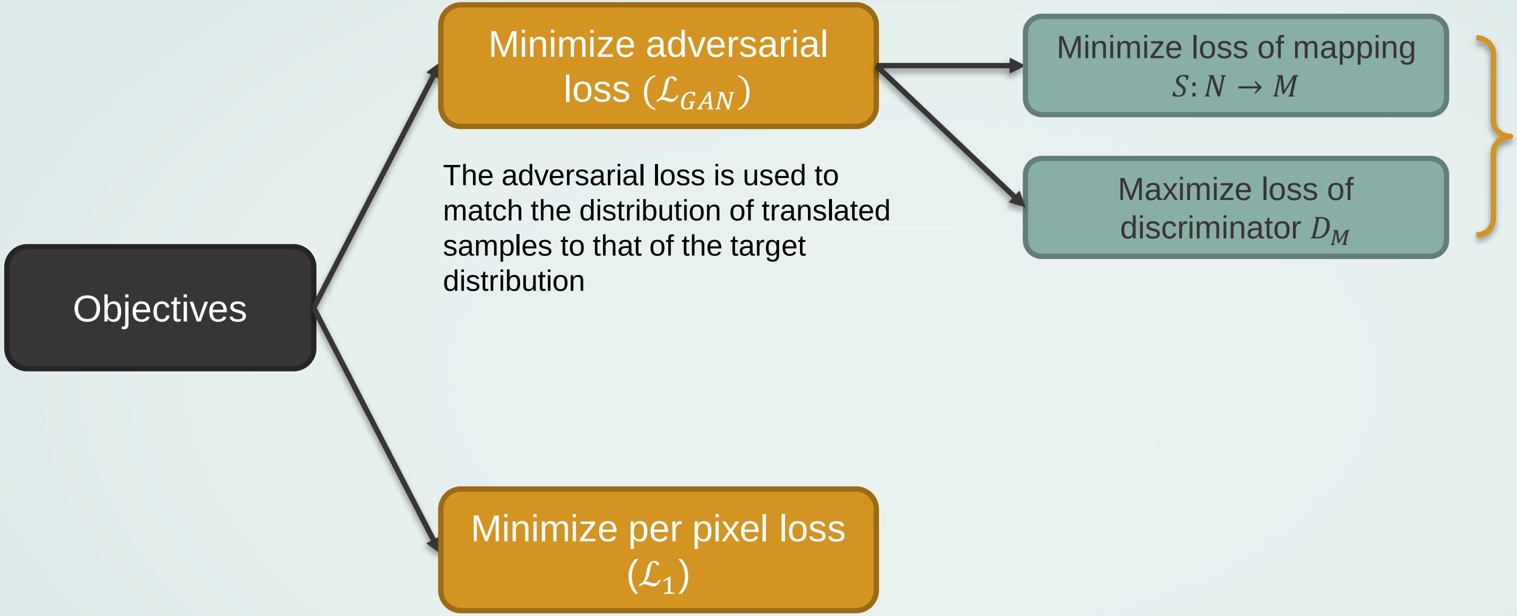
Synthetic Pathology Images with Nuclei Segmentation Masks

Problem 2: Nuclei segmentation

Architectures: cGAN (conditional GAN)
Normalization: spectral normalization (to prevent mode colapse)

Conditional GANs can learn a **structured context-aware loss** considering a larger receptive field. The cGAN framework learns a mapping S for nuclei segmentation, in which S can adapt H&E nuclei images to their segmentation masks.





Objectives

Minimize adversarial loss ($\mathcal{L}_{GAN}$)

The adversarial loss is used to match the distribution of translated samples to that of the target distribution

Minimize loss of mapping $S: N \rightarrow M$

Maximize loss of discriminator D_M

Binary Cross Entropy (BCE) loss of discriminator in classifying real or fake

Minimize per pixel loss ($\mathcal{L}_1$)

to penalize both the joint configuration of pixels and segmentation errors

Loss of mapping $S: N \to M$

S is penalized if the group configuration of the pixels in the predicted mask is unrealistic, i.e. masks that look like salt-and-pepper noise.

$$\begin{aligned}\mathcal{L}_{\mathrm{GAN}}(S, D_M) = {} & \mathbb{E}_{m,n\sim p_{\mathrm{data}}(m,n)}[\log D_M(m,n)] \\ & +\mathbb{E}_{n\sim p_{\mathrm{data}}(n)}[\log(1 - D_M(m, S(n)))]\end{aligned}$$

Pixel loss

$$\mathcal{L}_1(S) = \mathbb{E}_{m,n\sim p_{\mathrm{data}}}(m,n)[||m - S(n)||_1].$$

full objective for conditional GAN segmentation

$$\arg\ \min_S \max_{D_M} \mathcal{L}_{\mathrm{GAN}}(S, D_M) + \mathcal{L}_1(S).$$

Training Details

- Pytorch 4.0
- Ukuran 512 x 256
- Optimizer: Adam optimizer
- Number of epoch: 400
- Normalization: spectral normalization
- Learning rate: 0.0002 (for first 200 epochs) → linearly decayed to 0 (for remaining epochs)
- Weight initialized from a Gaussian distribution with a mean 0 and standard deviation 0.02
- Randomly select 64 patch pairs of segmented output and ground truth to be used in the discriminator

Output

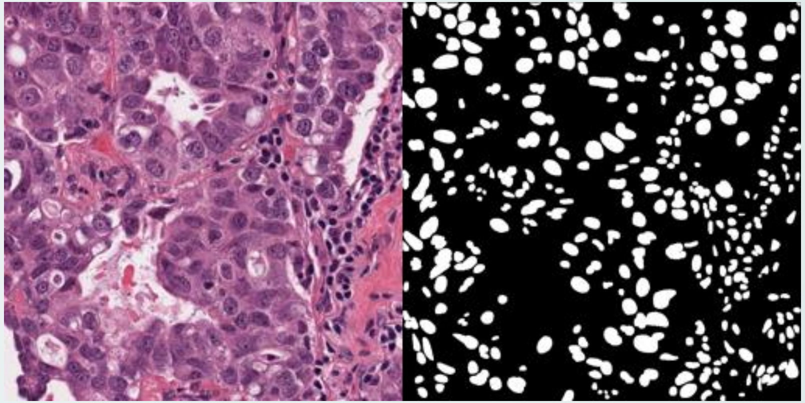
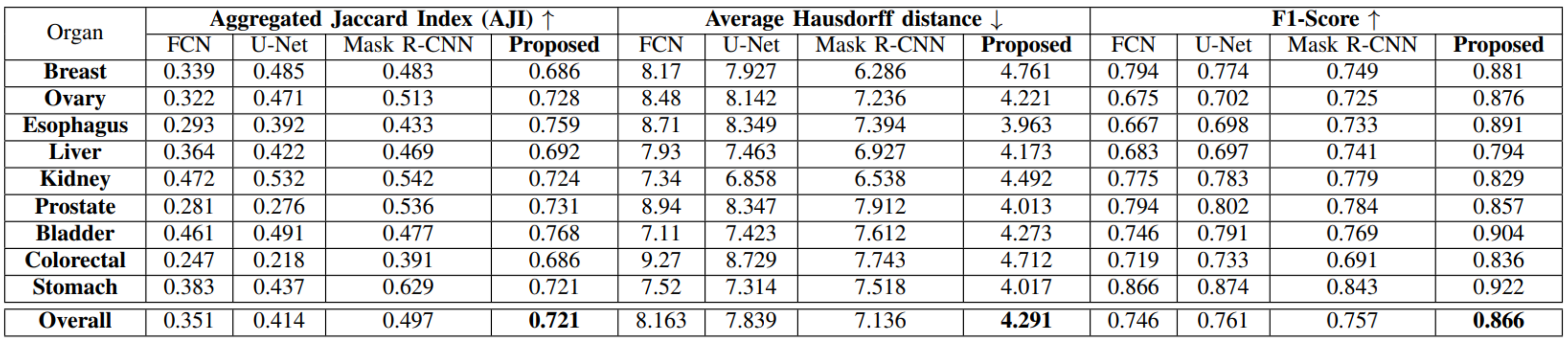


Image of preprocessing result

Results



Organ	Aggregated Jaccard Index (AJI) ↑				Average Hausdorff distance ↓				F1-Score ↑			
	FCN	U-Net	Mask R-CNN	**Proposed**	FCN	U-Net	Mask R-CNN	**Proposed**	FCN	U-Net	Mask R-CNN	**Proposed**
Breast	0.339	0.485	0.483	0.686	8.17	7.927	6.286	4.761	0.794	0.774	0.749	0.881
Ovary	0.322	0.471	0.513	0.728	8.48	8.142	7.236	4.221	0.675	0.702	0.725	0.876
Esophagus	0.293	0.392	0.433	0.759	8.71	8.349	7.394	3.963	0.667	0.698	0.733	0.891
Liver	0.364	0.422	0.469	0.692	7.93	7.463	6.927	4.173	0.683	0.697	0.741	0.794
Kidney	0.472	0.532	0.542	0.724	7.34	6.858	6.538	4.492	0.775	0.783	0.779	0.829
Prostate	0.281	0.276	0.536	0.731	8.94	8.347	7.912	4.013	0.794	0.802	0.784	0.857
Bladder	0.461	0.491	0.477	0.768	7.11	7.423	7.612	4.273	0.746	0.791	0.769	0.904
Colorectal	0.247	0.218	0.391	0.686	9.27	8.729	7.743	4.712	0.719	0.733	0.691	0.836
Stomach	0.383	0.437	0.629	0.721	7.52	7.314	7.518	4.017	0.866	0.874	0.843	0.922
Overall	0.351	0.414	0.497	**0.721**	8.163	7.839	7.136	**4.291**	0.746	0.761	0.757	**0.866**

The segmentation networks demonstrates a 29.19% improvement in AJI as compared to DIST and 42.98% as compared to CNN-3C. In terms of standard architectures there is a 44.27% improvement over standard Mask R-CNN and 73.19% over a U-Net.

Resources

Mahmood, Faisal dkk. (2018). *Deep Adversarial Training for Multi-Organ Nuclei Segmentation in Histopathology Images*.