

# Unsupervised Histopathology Image Synthesis

Le Hou<sup>1</sup>, Ayush Agarwal<sup>2</sup>, Dimitris Samaras<sup>1</sup>, Tahsin M. Kurc<sup>3,4</sup>, Rajarsi R. Gupta<sup>3,5</sup>, Joel H. Saltz<sup>3,1,5,6</sup>

<sup>1</sup>Dept. of Computer Science, Stony Brook University

<sup>2</sup>Dougherty Valley High School, California

<sup>3</sup>Dept. of Biomedical Informatics, Stony Brook University

<sup>4</sup>Oak Ridge National Laboratory

<sup>5</sup>Dept. of Pathology, Stony Brook Hospital

<sup>6</sup>Cancer Center, Stony Brook Hospital

{lehhou, samaras}@cs.stonybrook.edu

ayush94582@gmail.com

{tahsin.kurc, joel.saltz}@stonybrook.edu

rajarsi.gupta@stonybrookmedicine.edu

## Abstract

*Hematoxylin and Eosin stained histopathology image analysis is essential for the diagnosis and study of complicated diseases such as cancer. Existing state-of-the-art approaches demand extensive amount of supervised training data from trained pathologists. In this work we synthesize in an unsupervised manner, large histopathology image datasets, suitable for supervised training tasks. We propose a unified pipeline that: a) generates a set of initial synthetic histopathology images with paired information about the nuclei such as segmentation masks; b) refines the initial synthetic images through a Generative Adversarial Network (GAN) to reference styles; c) trains a task-specific CNN and boosts the performance of the task-specific CNN with on-the-fly generated adversarial examples. Our main contribution is that the synthetic images are not only realistic, but also representative (in reference styles) and relatively challenging for training task-specific CNNs. We test our method for nucleus segmentation using images from four cancer types. When no supervised data exists for a cancer type, our method without supervision cost significantly outperforms supervised methods which perform across-cancer generalization. Even when supervised data exists for all cancer types, our approach without supervision cost performs better than supervised methods.*

## 1. Introduction

We propose a method for the synthesis of large scale, realistic image datasets that can be used to train machine learning algorithms for histopathology image analysis in precision medicine. Precision medicine requires the ability to classify patients into specialized cohorts that differ

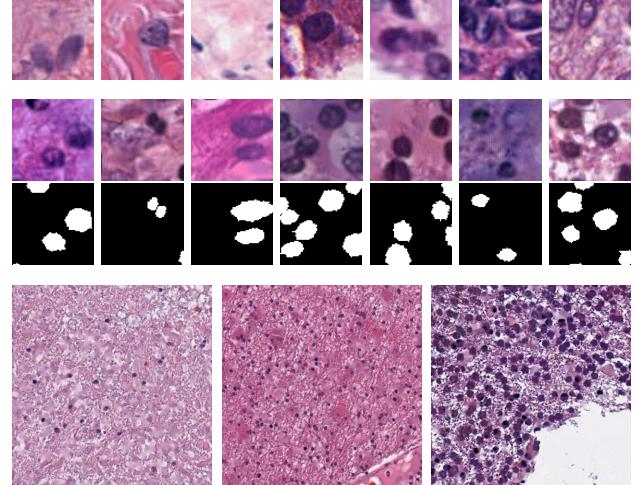


Figure 1. First row: real histopathology image patches at 40X magnification, with unknown nucleus segmentation mask. Center two rows: our synthesized histopathology image patches at 40X and corresponding nucleus segmentation masks. Last row: our synthesized 20X large patches with different cellularity and nuclear pleomorphism.

in their susceptibility to a particular disease, in the biology and/or prognosis of the disease, or in their response to therapy [17, 12]. Imaging data and in particular quantitative features extracted by image analysis have been identified as a critical source of information particularly for cohort classification (imaging phenotypes) and tracking response to therapy. Quantitative features extracted from Pathology and Radiology imaging studies, provide valuable diagnostic and prognostic indicators of cancer [14, 15, 4, 37, 19].

Nucleus segmentation in histopathology images is a central component in virtually all Pathology precision medicine

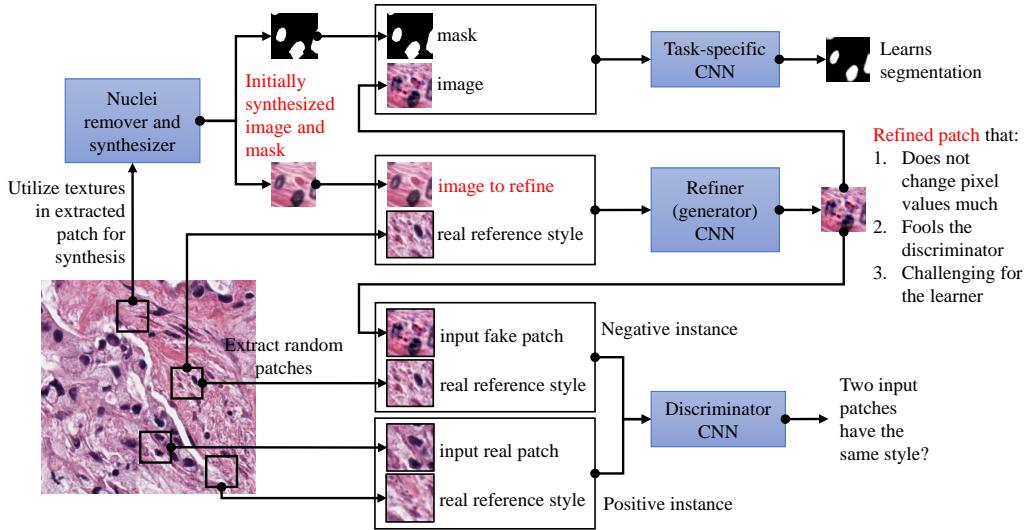


Figure 2. Our method synthesizes histopathology images with desired styles and known information (such as nuclei and their segmentation masks). There are three CNNs in our approach. The refiner (generator) CNN refines initial synthetic image patches synthesized by the “nuclei-remover-and-synthesizer” module according to reference styles. The discriminator learns to criticize the refined patches, so that the refiner can generate realistic patches that match the reference style. The task-specific CNN learns to segment or classify the refined patches and give feedback to the refiner, so that the refiner can generate challenging patches for training. We show details of the “nuclei-remover-and-synthesizer” in Fig. 3.

imaging studies [11, 21, 13, 42]. Existing machine-learning based image analysis methods [5, 50, 48, 49, 9, 52, 51, 23, 33] largely rely on availability of large annotated training datasets. One of the challenges is the generation of training datasets, because it requires the involvement of expert pathologists. It is a time-consuming, labor intensive and expensive process. In our experience, manually segmenting a nucleus in a tissue image takes about 2 minutes. A relatively small training dataset of 50 representative  $600 \times 600$ -pixel image patches has about 7000 nuclei. This corresponds to 225 hours of a Pathologist’s time to generate the training dataset. Such a training dataset is still a very small sampling of a moderate size dataset of a few hundred images – each whole slide tissue image has a few hundred thousand to over a million nuclei. Moreover, the training phase usually should be repeated for different cancer types or even within a cancer type when new images are added. This is because of the heterogeneity of tissue specimens (of different cancer types, sub-types and stages) as well as variations arising from tissue preparation and image acquisition.

We propose a methodology to significantly reduce the cost of generating training datasets by synthesizing histopathology images that can be used for training task specific algorithms. With our methodology a pathologist would only need to help tune the hyperparameters of the unsupervised synthesis pipeline by giving rounds of feedback (synthetic nuclei should be 20% larger, *etc.*). In this way the time cost of human involvement in training dataset

generation would go down from hundreds of hours to under one hour. In our experiments, we synthesized a dataset 400 times larger than a manually collected training set, which would cost 225 hours of a Pathologist’s time. Due to the large volume of training data, segmentation CNNs trained on the synthetic dataset outperform segmentation CNNs trained on the more precise but much smaller manually collected dataset.

Recent works in machine learning for image analysis have proposed crowd-sourcing or high-level, less accurate annotations, such as scribbles, to generate large training datasets by humans [30, 47, 51]. Another approach is to automatically synthesize training data, including pathology images and associated structures such as nucleus segmentation masks. Work by Zhou *et al.* [54] segments nuclei inside a tissue image and redistributes the segmented nuclei inside the image. The segmentation masks of the redistributed nuclei are assumed to be the predicted segmentation masks. Generative Adversarial Network (GAN) [38] approaches have been proposed for generation of realistic images [16, 7, 6, 44, 8, 53, 36]. For example, an image-to-image translation GAN [24, 16] synthesizes eye fundus images. However, it requires an accurate supervised segmentation network to segment eye vessels out, as part of the synthesis pipeline. The S+U learning framework [44] uses physics-based rendering methods to obtain initially synthesized images and refines via a GAN those images to increase their realism. This method achieves state-of-the-art

results in eye gaze and hand pose estimation tasks.

There are several challenges to synthesizing histopathology images. First, state-of-the-art image synthesis approaches [44, 53, 39, 40] require a physics-based 3D construction and rendering model. However, physics in the cellular level is largely unknown, making physics-based modeling infeasible. Second, histopathology images are heterogeneous with rich structure and texture characteristics. It is hard to synthesize images with a large variety of visual features. Moreover, care must be taken to avoid synthesizing images which can easily become biased and easy to classify, despite being realistic and heterogeneous. Our methodology (Fig. 2) addresses these problems for Hematoxylin and Eosin (H&E) stained histopathology images. H&E is the mostly commonly used staining system for disease diagnosis and prognosis.

The **first contribution** is a computer vision-based histopathology image synthesis method that generates initial synthetic histopathology images with desired characteristics such as the locations and sizes of the nuclei, cellularity, and nuclear pleomorphism, as shown in Fig. 3. Our method only needs a simple unsupervised segmentation algorithm that always super-segments nuclei. In “super-segmentation”, the segmented regions always fully contain the segmentation object.

The **second contribution** is that our method can synthesize heterogeneous histopathology images that span a variety of styles, i.e., tissue types and cancer subtypes. Image synthesis methods essentially model the distribution of real data [28]. The joint distribution of real pixel values is very complex and hard to model. We propose to sample images from the real distribution and synthesizes images similar to the sampled real images, thus, simulating the distribution of real samples. Our model takes real images as references and generates realistic images in the reference style using a Generative Adversarial Network (GAN). This can be viewed as an instance of universal style transfer [29, 45].

Our **third contribution** is to train a task-specific model jointly with the image synthesis model. The image synthesis model is aware of the task-specific model and generates adversarial (hard) examples accordingly. Compared with existing hard example mining methods [43, 27] and adversarial data augmentation methods [20], our approach generates different versions of hard or adversarial training examples on-the-fly, according to the snapshot of the current task-specific model, instead of mining for existing hard examples in a dataset or inefficiently adding adversarial noise via slow optimization processes.

We test our method for nucleus segmentation using images from four cancer types. When no supervised data exists for a cancer type, our method without supervision cost significantly outperforms supervised methods which perform across-cancer generalization. Even when supervised

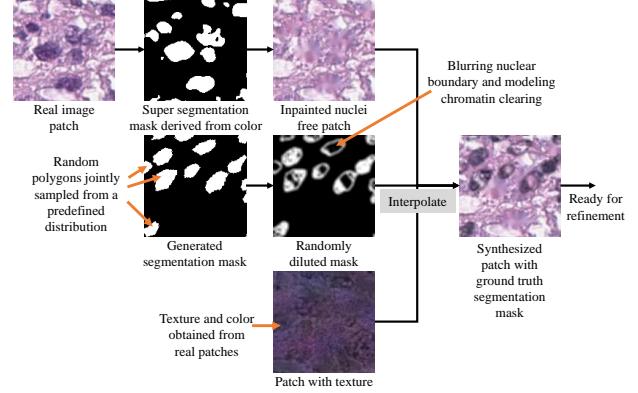


Figure 3. Inside the “nuclei-remover-and-synthesizer” module: the process of synthesizing a histopathology image patch and nucleus segmentation mask in the initial stage. The synthesized image will be refined with GAN.

data exists for all cancer types, our approach performed better than supervised methods.

## 2. Initial Synthesis

We utilize the texture characteristics of real histopathology image patches to generate initial synthetic images patches, in a background/foreground manner, with nuclei as the foreground. The first step of this workflow is to create a synthetic image patch without any nuclei. The second step simulates the texture and intensity characteristics of nuclei in the real image patch. The last step combines the output from the first two steps based on a randomly generated nucleus segmentation mask (see Figure 3 for the initial synthesized image patch). For simplicity, we will refer to image patches as images in the rest of the manuscript. Synthesizing a  $200 \times 200$  pixel patch at 40X magnification takes one second by a single thread on a desktop CPU.

### 2.1. Generating Background Patches

We first remove the foreground (nuclei) in an image patch to create a background image on which we will add synthetic nuclei. We apply a simple threshold-based super-segmentation method on the source image patch to determine nuclear pixels in the source image. In “super-segmentation”, the segmented regions always fully contain the segmentation object. We then remove those pixels and replace them with color and texture values similar to the background pixels via image inpainting [46]. Super-segmentation may not precisely delineate object boundaries and may include non-nuclear material in segmented nuclei. This is acceptable, because the objective of this step is to guarantee that only background tissue texture and intensity properties are used to synthesize the background image.

Hematoxylin mainly stains nucleic acids whereas Eosin

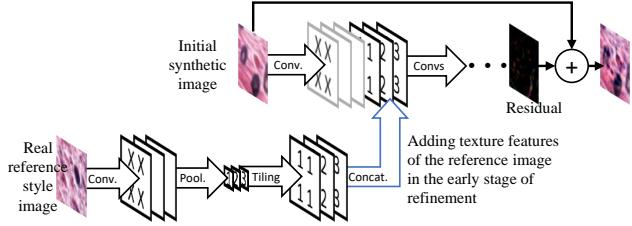


Figure 4. Our refiner (generator) CNN adds the global texture features of the reference image into the early stage of refinement, so that the initial synthetic image will be refined according to the textures of the reference style image.

stains proteins nonspecifically in tissue specimens [18]. We apply color deconvolution [41] to H&E images to obtain the Hematoxylin, Eosin, DAB (HED) color space. We threshold the H channel for nuclei segmentation. Specifically, we first decide the percentage of nuclear pixels,  $p$ , based on the average color intensity  $h$ , of the H channel. For  $h$  in ranges  $(-\infty, -1.25)$ ,  $[-1.25, -1.20)$ ,  $[-1.20, -1.15)$ ,  $[-1.15, -1.10)$ ,  $[-1.10, \infty)$ , we set the percentage of nuclear pixels  $p$  as 15%, 20%, 25%, 30%, 35% respectively. These hyperparameters were selected by visually inspecting super-segmentation results on a set of image patches from all cancer types in the TCGA repository [1]. The segmentation threshold,  $t$ , is the  $p$ -th percentile value of the H channel. After thresholding the H channel with  $t$ , we apply Gaussian smoothing to remove noise such as very small segmented regions. Finally, the segmented pixels are inpainted in a computationally efficient manner [46].

## 2.2. Simulating Foreground Textures

One approach to simulating foreground nuclear textures is to apply a sub-segmentation method and gather nuclear textures from segmented regions. In “sub-segmentation”, the segmentation object always contains segmented regions. The objective of sub-segmentation ensures that pixels within the nuclei are used for nuclei synthesis. Since nuclei are generally small and make up a small portion of the tissue area-wise, sub-segmentation will yield very limited amount of nuclear material which is not enough for existing reconstruction methods to generate realistic nuclear material patches. Thus, our approach utilizes textures in the Eosin channel [18] of a randomly extracted real patch (different from the background source patch in Section 2.1) and combines them with nuclear color obtained via sub-segmentation of the input patch to generate nuclear textures.

We have observed that this method gives realistic textures. To sub-segment, we use the same process as for the super-segmentation approach but with different  $p$  values: For  $h$  in ranges  $(-\infty, -1.25)$ ,  $[-1.25, -1.20)$ ,  $[-1.20, -1.15)$ ,  $[-1.15, -1.10)$ ,  $[-1.10, \infty)$ , we set  $p$  as 10%, 16%, 21%, 27%, 32% respectively.

## 2.3. Combining Foreground and Background

We generate a nuclear mask and combine nuclear and non-nuclear textures according to the mask. First, we randomly generate non-overlapping polygons with variable sizes and irregularities. To model the correlation between the shapes of nearby nuclei, we distort all polygons by a random quadrilateral transform. The resulting nucleus mask is regarded as a synthetic “ground truth” segmentation mask. We then combine foreground and background patches by:

$$I_{i,j} = A_{i,j}M_{i,j} + B_{i,j}(1 - M_{i,j}). \quad (1)$$

Here,  $I_{i,j}$  is the pixel value of the resulting synthetic image. Pixel values at position  $i, j$  in the nuclear texture patch, in the nucleus free patch, and in the nucleus mask are denoted as  $A_{i,j}$ ,  $B_{i,j}$ ,  $M_{i,j}$  respectively.

Applying Eq. 1 naively results in significant artifacts, such as obvious nuclear boundaries. Additionally, clearing of chromatin cannot be modeled. To remedy these issues, we randomly clear the interior and blur the boundaries of the polygons in  $M$ , before applying Eq. 1.

## 3. Refined Synthesis

We refine the initial synthetic images via adversarial training as shown in Fig. 2. This phase implements a Generative Adversarial Network (GAN) model and consists of a refiner (generator) CNN and a discriminator CNN.

Given an input image  $I$  and a reference image  $S$ , the refiner  $G$  with trainable parameters  $\theta_G$  outputs a refined image  $\tilde{I} = G(I, S; \theta_G)$ . Ideally, the output image is:

**Regularized** The pixel-wise difference between the initial synthetic image and the refined image is small enough so that the synthetic “ground truth” remains unchanged.

**Realistic** It has a realistic representation of the style of the reference image.

**Informative/hard** It is a challenging case for the task-specific CNN so that the trained task-specific CNN will be robust.

We build three losses:  $L_G^{\text{reg}}$ ,  $L_G^{\text{real}}$ ,  $L_G^{\text{hard}}$ , for each of the properties above. The weighted average of these losses as the final loss  $L_G$  for training of the refiner CNN is:

$$L_G = \alpha L_G^{\text{reg}} + \beta L_G^{\text{real}} + \gamma L_G^{\text{hard}}. \quad (2)$$

Selection of hyperparameters  $\alpha, \beta, \gamma$  is described in Sec. 6.

The regularization loss  $L_G^{\text{reg}}$  is defined as:

$$L_G^{\text{reg}}(\theta_G) = \mathbb{E}[\lambda_1\|I - \tilde{I}\|_1 + \lambda_2\|I - \tilde{I}\|_2], \quad (3)$$

where  $\mathbb{E}[\cdot]$  is the expectation function applied on the training set,  $\|I - \tilde{I}\|_1$  and  $\|I - \tilde{I}\|_2$  are the  $L$ -1 and  $L$ -2 norms

of  $I - \tilde{I}$  respectively and  $\lambda_1$  and  $\lambda_2$  are predefined parameters. This is the formulation of second order elastic net regularization [55]. In practice, we select the lowest  $\lambda_1$  and  $\lambda_2$  possible that do not result in significant visual changes of  $\tilde{I}$  compared to  $I$ .

The loss for achieving a realistic reference style is:

$$L_G^{\text{real}}(\theta_G) = E[\log(1 - D(\tilde{I}, S; \theta_D))], \quad (4)$$

where  $D(\tilde{I}, S; \theta_D)$ , is the output of the discriminator  $D$  with trainable parameters  $\theta_D$  given the refined image  $\tilde{I}$  and the same reference style image  $S$  as input. It is the estimated probability by  $D$  that input  $\tilde{I}$  and  $S$  are real images in the same style.

The Discriminator  $D$  with trainable parameters  $\theta_D$  has two types of input: pairs of real images within the same style  $\langle S', S \rangle$  and a pair with one synthetic image  $\langle \tilde{I}, S \rangle$ . The loss of  $D$  is defined as:

$$L_D(\theta_D) = -E[\log(D(S', S; \theta_D))] - E[\log(1 - D(\tilde{I}, S; \theta_D))]. \quad (5)$$

The discriminator learns to maximize its output probability for real pairs  $\langle S', S \rangle$  and minimize it for  $\langle \tilde{I}, S \rangle$ . By introducing the reference style image  $S$ , the discriminator can correctly recognize the pair that contains a synthetic image if the synthetic image is not realistic, or it has a different style compared to the reference style image.

**CNN Architecture for Style Transfer** The generator and discriminator both take a reference image and refine or classify the other input image according to textures in the reference image. We implement this feature with a CNN which takes two input images. Existing CNN architectures, such as the siamese network [10, 26], merge or compare the features of two input images at a late network stage. However, the generator must represent the textures in the reference image and use it in the process of refinement at an early stage. To achieve this, our network has two branches: the texture representation branch and the image refinement branch. As is shown in Fig. 4, the texture representation branch takes the reference image as input and outputs a feature vector representing the reference image. The image refinement branch takes both the initial synthetic image and the reference image and generates a refined image.

We show the effect of adding the reference style images in GAN training in Fig. 5. The discriminator is significantly more accurate and gives more feedback in terms of the realism loss  $L_G^{\text{real}}(\theta_G)$ , to the refiner.

## 4. On-the-fly Hard Example Synthesis

The refiner is trained with loss  $L_G^{\text{hard}}$  to generate challenging training examples (with larger loss) for the task-specific CNN. We simply define  $L_G^{\text{hard}}$  as the negative of

the task-specific loss:

$$L_G^{\text{hard}}(\theta_G) = -L_R(\theta_R), \quad (6)$$

where  $L_R(\theta_R)$  is the loss of a task-specific model  $R$  with trainable parameters  $\theta_R$ . In the case of segmentation,  $L_R(\theta_R)$  is the conventional segmentation loss used in deep learning [31, 35]. When training the refiner, we update  $\theta_G$  to produce refined images that maximizes  $L_R$ . When training the task-specific CNN, we update  $\theta_R$  to minimize  $L_R$ .

The underlying segmentation ground truth of the refined images would change significantly if  $L_G^{\text{hard}}(\theta_G)$  overpowered  $L_G^{\text{reg}}(\theta_G)$ . We down weight  $L_G^{\text{hard}}$  by a factor of 0.0001 to minimize the likelihood of this outcome.

**Training process** We randomly initialize the refiner, discriminator and the task-specific networks. During the training process, the realism loss  $L_G^{\text{real}}$  and the task-specific adversarial loss  $L_G^{\text{hard}}$  are fed back to the refiner from the discriminator and the task-specific CNNs respectively. However, because we randomly initialize the discriminator and the task-specific networks, these feedbacks are initially useless for the refiner. Following the existing image refining GAN [44], we initially train each CNN individually before training them jointly. The process is summarized in Alg. 1.

---

### Algorithm 1: Refining and task-specific learning.

---

**Input :** A set of training images. Number of training iterations  $N_G, N_D, N_R, N_{GD}, N_{GDR}, n_G, n_D, n_R$ . Loss parameters  $\alpha, \beta, \gamma, \lambda_1, \lambda_2$ .

**Output:** Trained segmentation/classification CNN  $R$ .

- 1 Randomly initialize the trainable parameters  $\theta_G, \theta_D$  and  $\theta_R$  in  $G, D$  and  $R$  respectively.
- 2 Train  $G$  to minimize  $L_G^{\text{reg}}(\theta_G)$  for  $N_G$  iterations.
- 3 Train  $D$  to minimize  $L_D(\theta_D)$  for  $N_D$  iterations.
- 4 **for**  $n = 1, \dots, N_{GD}$  **do**
- 5     Train  $G$  to minimize  $\alpha L_G^{\text{reg}}(\theta_G) + \beta L_G^{\text{real}}(\theta_G)$  for  $n_G$  iterations.
- 6     Train  $D$  to minimize  $L_D(\theta_D)$  for  $n_D$  iterations.
- 7 **end**
- 8 Train  $R$  to minimize  $L_R(\theta_R)$  for  $N_R$  iterations.
- 9 **for**  $n = 1, \dots, N_{GDR}$  **do**
- 10     Train  $G$  to minimize  $\alpha L_G^{\text{reg}}(\theta_G) + \beta L_G^{\text{real}}(\theta_G) + \gamma L_G^{\text{hard}}(\theta_G)$  for  $n_G$  iterations.
- 11     Train  $D$  to minimize  $L_D(\theta_D)$  for  $n_D$  iterations.
- 12     Train  $R$  to minimize  $L_R(\theta_R)$  for  $n_R$  iterations.
- 13 **end**
- 14 **return**  $R$  with  $\theta_R$ ;

---

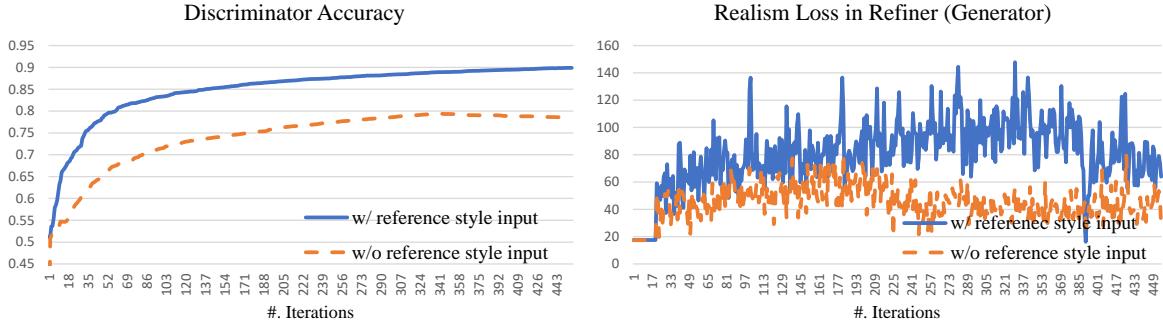


Figure 5. The effect of introducing real reference style images in the GAN training process. To fool the discriminator that “knows” the reference style, the refined images should be in the same style as the reference image, in addition to being realistic. Thus, the discriminator with reference style input is more accurate, and gives significantly more feedback in terms of the realism loss (Eq. 4) to the refiner.

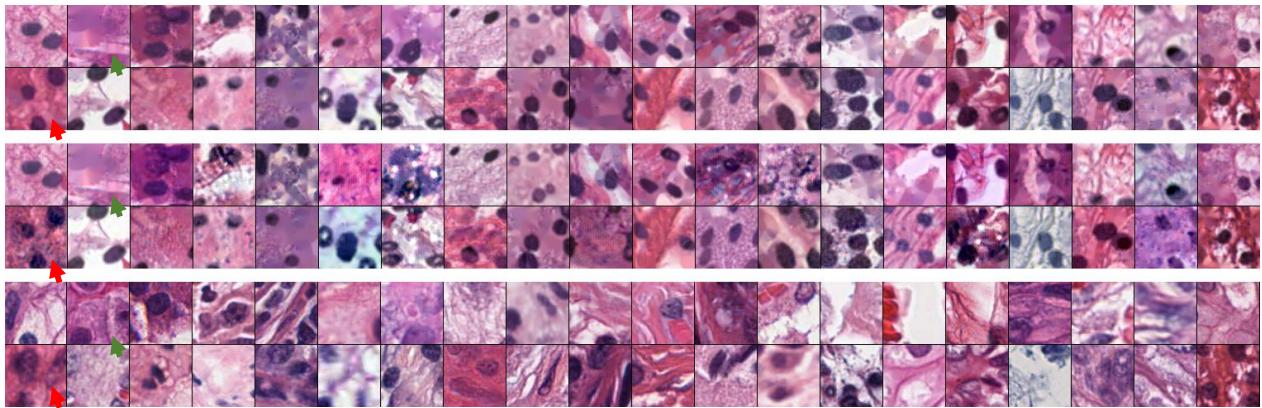


Figure 6. Randomly selected examples of initial synthetic histopathology images (first two rows), refined images (second two rows), and corresponding real reference style images (last two rows). The refiner successfully refines the initial synthetic images to reference styles without modifying the images significantly (example indicated by red arrow). On cases where the refiner fails, this signifies that the initial synthetic images can not be transferred to reference styles without significantly modifying the images (sample indicated by green arrow).



Figure 7. Randomly selected examples of initial synthetic street view house number images (first row), refined images (second row), and corresponding real reference style images (last row).

## 5. Visual Test by Expert

To verify that the synthetic images are realistic, we asked a pathologist to distinguish real versus synthetic images. In particular, we showed the pathologist 100 randomly extracted real patches, 100 randomly selected initial synthetic

patches, and 100 randomly selected refined patches. Out of this set, the pathologist selected the patches he thought were real. We summarize the results in Table 1. A significant number of initial synthetic images (46%) were classified as real by the pathologist. Most of the refined patches (64%) were classified real. Note that 17% of the real patches were classified fake. This is because many of those image patches are out-of-focus or contain no nuclei. In average, the pathologist spend 4.6 seconds classifying on each patch. We show representative examples of synthetic images that appeared real to the pathologist in Fig. 8.

We show randomly selected examples of initial synthetic and refined histopathology images in Fig. 6. The refiner successfully refines the initial synthetic images to reference styles without modifying the images significantly. On cases where the refiner fails, the initial synthetic images can not be transferred to the reference styles without significantly modifying the images.

Ground truth	#. classified real	#. classified fake
Initial synthetic	46	54
Refined	64	36
Real	87	13

Table 1. We show 100 randomly selected and ordered initial synthetic, refined and real patches to a pathologist, and ask the pathologist to classify them as real or fake.

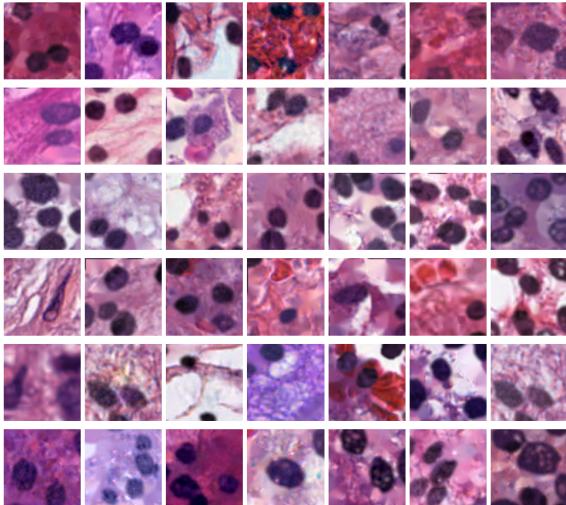


Figure 8. Representative examples of synthetic images that appeared real to the pathologist.



Figure 9. Randomly selected examples of refined synthetic street view house number images.

To demonstrate the generality of our method, and how our method works outside the pathology domain, we synthesize house street numbers using the SVHN database [34]. To generate initial synthetic images from real images, we apply a k-means clustering method to obtain the background and foreground colors in the real images. Then we write a digit in a random font in constant foreground color. The refiner refines the style of the initial synthetic images to the real reference style. We show randomly selected examples in Fig. 7 and Fig. 9.

## 6. Experiments

To evaluate the performance of our method, we conducted experiments with ground-truth datasets generated for the MICCAI15 and MICCAI17 nucleus segmentation challenges [2, 3]. Additionally, we synthesized large pathology image patches for two classes: high/low cellularularity and nuclear pleomorphism and show that a task-specific CNN trained on this dataset can classify glioblastoma (GBM) versus low grade gliomas (LGGs).

### 6.1. Implementation Details

The refiner network, outlined in Fig. 4, has 21 convolutional layers and 2 pooling layers. The discriminator network has the same overall architecture with the refiner. It has 15 convolutional layers and 3 pooling layers. As the task-specific CNN, we implement U-net [40] and a network with 15 convolutional layers and 2 pooling layers, and a semi-supervised CNN [22] for segmentation. We use a 11 convolutional layer network for classification. For hyperparameters in Eq. 2 and Eq. 3, we select  $\alpha = 1.0$ ,  $\beta = 0.7$ ,  $\gamma = 0.0001$ ,  $\lambda_1 = 0.001$ ,  $\lambda_2 = 0.01$  by validating on part of a synthetic dataset. We implement our method using an open source implementation of S+U learning [25, 44]. The methods we test are listed below.

**Synthesis CAE-CNN** Proposed method with the semi-supervised CNN [22] as the task-specific segmentation CNN.

**Synthesis U-net** Proposed method with U-net [40] as the task-specific segmentation CNN.

**Synthesis CNN** Proposed method with a 15 layer segmentation network or a 11 layer classification network.

**CAE-CNN / U-net / CNN with supervision cost** We use the semi-supervised CNN [22], U-net [40] and the 15 layer CNN as standalone supervised networks, trained on real human annotated datasets. We augment the real images by rotating four times, mirroring, and rescaling six times.

### 6.2. Nucleus segmentation

The MICCAI15 nucleus segmentation challenge dataset [2] contains 15 training and 18 testing images extracted from whole slide images of GBM and LGG. The MICCAI17 dataset [3] contains 32 training and 32 testing images, extracted from whole slide images of GBM, LGG, Head and Neck Squamous cell Carcinoma (HNSC) and Lung Squamous Cell Carcinoma (LUSC). A typical resolution is  $600 \times 600$  pixels at 20X or 40X (0.50 or 0.25 microns per pixel) magnifications. Assuming that annotating one nucleus takes 2 minutes, it would take

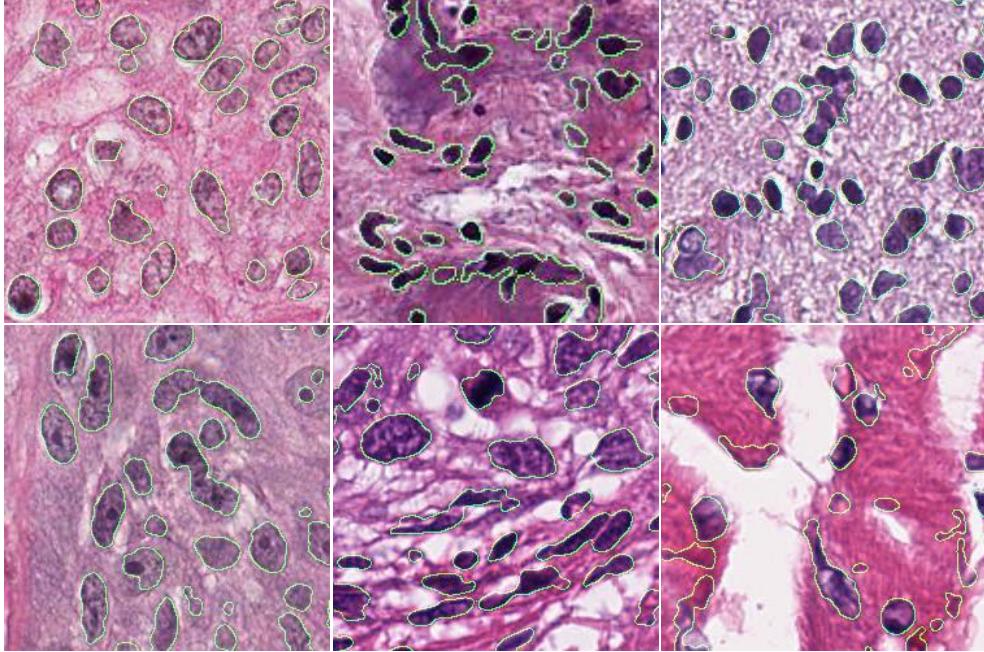


Figure 10. Randomly selected examples of nucleus segmentation results (green contours) on the MICCAI15 and MICCAI17 nucleus segmentation test set.

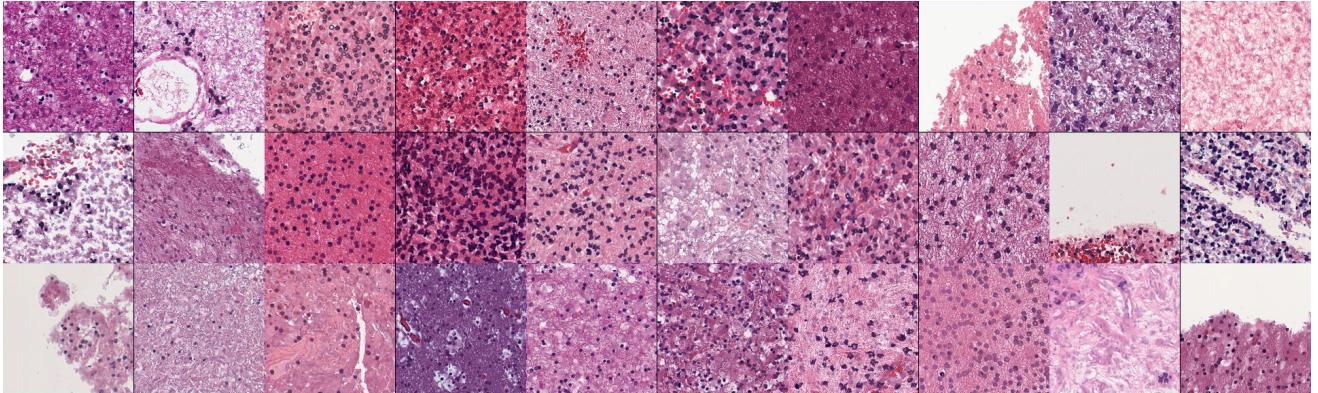


Figure 11. Randomly selected examples of synthetic  $384 \times 384$  pixel 20X histopathology image patches with various levels of cellularity and nuclear pleomorphism.

about 225 man-hours to generate these training datasets. In contrast, it would take just one hour of a Pathologist for us to synthesize a large dataset.

We investigate if the task-specific supervised method performs better in standalone mode when it is trained on a few but real training data or when it is trained with abundant but synthetic training data generated by our synthesis pipeline. We evaluate the supervised segmentation method of Sec. 6.1 under three scenarios:

**Universal** We train one universal segmentation CNN on training images of all two/four (MICCAI15/MICCAI17) cancer types.

**Cancer specific** We train one CNN for each cancer type. During test time, we apply the corresponding CNN based on the cancer type of the input image.

**Across cancer** To evaluate the performance of supervised CNNs on cancer types that lack training data, we train one CNN for each cancer type in the testing set, excluding training images of that cancer type from the training set. During test time, based on the cancer type of the input image, we apply the corresponding CNN that was not trained with that cancer type.

Most cancer types do not have a significant nucleus segmen-

tation training set. Therefore, the third scenario is a very common real world use case. For our method, we generated 200k  $75 \times 75$ -pixel initial synthetic patches at 40X magnification for each cancer type.

Segmentation methods	DICE Avg.
Synthesis CAE-CNN (proposed)	0.8424
CAE-CNN with supervision cost, Universal	0.8362
Synthesis U-net (proposed)	0.8063
U-net with supervision cost, Universal	0.7984
Synthesis CNN (proposed)	0.8254
CNN with supervision cost, Universal	0.8013
CNN with supervision cost, Cancer specific	0.8032
CNN with supervision cost, Across cancer	0.7818
Supervised contour-aware net (challenge winner) [9]	0.812

Table 2. Nucleus segmentation results on the MICCAI15 nucleus segmentation dataset. On cancer types without annotated training data, our approach outperforms the supervised method (CNN with supervision cost, Across cancer) significantly. Even when supervised data exists for all cancer types, our approach improves the state-of-the-art performance without any supervision cost.

Segmentation methods	DICE avg
Synthesis CAE-CNN (proposed)	0.7731
CAE-CNN with supervision cost, Universal	0.7681
Synthesis U-net (proposed)	0.7631
U-net with supervision cost, Universal	0.7645
Synthesis CNN (proposed)	0.7738
CNN with supervision cost, Universal	0.7713
CNN with supervision cost, Cancer specific	0.7653
CNN with supervision cost, Across cancer	0.7314
Challenge winner	0.783

Table 3. Nucleus segmentation results on the MICCAI17 nucleus segmentation dataset. On cancer types without annotated training data, our approach outperforms the supervised method (CNN with supervision cost, Across cancer) significantly. Even when supervised data exists for all cancer types, our approach matches the state-of-the-art performance without any supervision cost.

We use the average of two versions of DICE coefficients. Quantitative evaluation results on the MICCAI15 and MICCAI17 segmentation datasets are shown in Tab. 2 and Tab. 3. With cancer types without annotated training images, our approach outperforms the supervised method (CNN with supervision cost, Across cancer) significantly. Even when supervised data exists for all cancer types, our approach achieves state-of-the-art level performance or better without any supervision cost. We see that the supervised method we incorporated into our pipeline, has comparable performance to the winners of the two challenges.

### 6.3. Ablation study

We evaluate the importance of two proposed components of our method: utilizing a real reference style image for refinement and generating on-the-fly hard examples for CNN training. In particular, we remove one feature at a time and evaluate the performance of nucleus segmentation. Experimental results are shown in Tab. 4. We see that both proposed methods improve the segmentation results. We also show the effect of introducing real reference style images as additional network inputs in Fig. 5.

Segmentation methods	DICE avg
Synthesis CNN (proposed)	0.7738
No reference style during refinement	0.7589
No on-the-fly hard examples	0.7491

Table 4. Ablation study using the MICCAI17 nucleus segmentation challenge dataset. Each proposed method reduces the segmentation error by 6% to 9%.

### 6.4. Glioma classification

We synthesize patches of  $384 \times 384$  pixels in 20X of two classes: relatively low cellularity and nuclear pleomorphism, versus relatively high cellularity and nuclear pleomorphism (Fig. 11). Cellularity and nuclear pleomorphism levels provide diagnostic information. We train the task-specific CNN to classify high versus low cellularity and nuclear pleomorphism patches. The cellularity and nuclear pleomorphism prediction results on real slides can distinguish Glioblastoma (GBM) versus Lower Grade Glioma (LGG) with an accuracy of 80.1% (Chance being 51.3%). A supervised approach [32] trained for the GBM/LGG classification achieved an accuracy of 85% using a domain specific pipeline with nucleus segmentation and counting.

### 6.5. SVHN classification

These experiments evaluate our method with the format1 sub-set in the Street View House Number (SVHN) dataset [34]. The subset contains 68,120 training images and 23549 testing images in  $32 \times 32$  pixels. We synthesized 68,120 images with digits and refined them to reference styles sampled in the format1 training set. Classification errors (1 – accuracy) are shown in Tab. 5.

## 7. Conclusions

Collecting a large scale supervised histopathology image dataset is extremely time consuming. We presented a complete pipeline for synthesizing realistic histopathology images with nucleus segmentation masks, which can be used for training supervised methods. Our method synthesizes images in various styles, utilizing textures and colors

Methods	Training set	Error
Synthesis CNN (proposed)	3,000 syn. training images	29.03%
	5,000 syn. training images	23.24%
	10,000 syn. training images	18.47%
	30,000 syn. training images	17.57%
	68,120 syn. training images	17.08%
CNN with supervision cost	3,000 real training images	24.55%
	5,000 real training images	18.53%
	10,000 real training images	15.22%
	30,000 real training images	12.10%
	68,120 real training images	7.54%

Table 5. Quantitative results on the Street View House Number (SVHN) format1 dataset [34].

in real images. We train a task-specific CNN and a Generative Adversarial Network (GAN) in an end-to-end fashion, so that we can synthesize challenging training examples for the task-specific CNN on-the-fly. We evaluate our approach on the nucleus segmentation task. When no supervised data exists for a cancer type, our result is significantly better than across-cancer generalization results by supervised methods. Additionally, even when supervised data exists, our approach performed better than supervised methods. In the future, We plan to incorporate additional supervised classification and segmentation methods in our framework. Furthermore, we plan to model the texture of nuclei more accurately in the initial synthesis phase.

**Acknowledgements** This work was supported in part by 1U24CA180924-01A1 from the NCI, R01LM011119-01 and R01LM009239 from the NLM, the Stony Brook University SensorCAT, a gift from Adobe, and the Partner University Fund 4DVision project.

## References

- [1] The Cancer Genome Atlas. <https://cancergenome.nih.gov/>. 4
- [2] Miccai 2015 challenge: Segmentation of nuclei in images. <https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=20644646>, 2015. 7
- [3] Miccai 2017 challenge: Segmentation of nuclei in images. <http://miccai.cloudapp.net/competitions/57>, 2017. 7
- [4] H. J. Aerts, E. R. Velazquez, R. T. Leijenaar, C. Parmar, P. Grossmann, S. Cavalho, J. Bussink, R. Monshouwer, B. Haibe-Kains, D. Rietveld, et al. Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. *Nature communications*, 2014. 1
- [5] N. Bayramoglu and J. Heikkilä. Transfer learning for cell nuclei classification in histopathology images. In *ECCV Workshops*, 2016. 2
- [6] N. Bayramoglu, M. Kaakinen, L. Eklund, and J. Heikkila. Towards virtual h&e staining of hyperspectral lung histology images using conditional generative adversarial networks. In *CVPR*, 2017. 2
- [7] L. Bi, J. Kim, A. Kumar, D. Feng, and M. Fulham. Synthesis of positron emission tomography (pet) images via multi-channel generative adversarial networks (gans). In *Molecular Imaging, Reconstruction and Analysis of Moving Body Organs, and Stroke Imaging and Treatment*. 2017. 2
- [8] F. Calimeri, A. Marzullo, C. Stamile, and G. Terracina. Biomedical data augmentation using generative adversarial neural networks. In *International Conference on Artificial Neural Networks*, 2017. 2
- [9] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, and P.-A. Heng. Dcan: Deep contour-aware networks for object instance segmentation from histology images. *Medical Image Analysis*, 2017. 2, 9
- [10] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005. 5
- [11] R. Colen, I. Foster, R. Gatenby, M. E. Giger, R. Gillies, D. Gutman, M. Heller, R. Jain, A. Madabhushi, S. Madhavan, et al. Nci workshop report: clinical and computational requirements for correlating imaging phenotypes with genomics signatures. *Translational oncology*, 2014. 2
- [12] F. S. Collins and H. Varmus. A new initiative on precision medicine. *New England Journal of Medicine*, 2015. 1
- [13] L. A. Cooper, A. B. Carter, A. B. Farris, F. Wang, J. Kong, D. A. Gutman, P. Widener, T. C. Pan, S. R. Cholleti, A. Sharma, et al. Digital pathology: Data-intensive frontier in medical imaging. *Proceedings of the IEEE*, 2012. 2
- [14] L. A. Cooper, J. Kong, D. A. Gutman, F. Wang, S. R. Cholleti, T. C. Pan, P. M. Widener, A. Sharma, T. Mikkelsen, A. E. Flanders, et al. An integrative approach for in silico glioma research. *IEEE Transactions on Biomedical Engineering*, 2010. 1
- [15] L. A. Cooper, J. Kong, D. A. Gutman, F. Wang, J. Gao, C. Appin, S. Cholleti, T. Pan, A. Sharma, L. Scarpace, et al. Integrated morphologic analysis for the identification and characterization of disease subtypes. *Journal of the American Medical Informatics Association*, 2012. 1
- [16] P. Costa, A. Galdran, M. I. Meyer, M. D. Abràmoff, M. Niemeijer, A. M. Mendonça, and A. Campilho. Towards adversarial retinal image synthesis. *arXiv*, 2017. 2
- [17] N. R. Council et al. *Toward precision medicine: building a knowledge network for biomedical research and a new taxonomy of disease*. National Academies Press, 2011. 1
- [18] A. H. Fischer, K. A. Jacobson, J. Rose, and R. Zeller. Hematoxylin and eosin staining of tissue and cell sections. *Cold Spring Harbor Protocols*, 2008. 4
- [19] R. J. Gillies, P. E. Kinahan, and H. Hricak. Radiomics: images are more than pictures, they are data. *Radiology*, 2015. 1
- [20] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 3
- [21] M. N. Gurcan and A. Madabhushi. Digital pathology. *SPIE*, 2013. 2

- [22] L. Hou, V. Nguyen, D. Samaras, T. M. Kurc, Y. Gao, T. Zhao, and J. H. Saltz. Sparse autoencoder for unsupervised nucleus detection and representation in histopathology images. *arXiv*, 2017. 7
- [23] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz. Patch-based convolutional neural network for whole slide tissue image classification. In *CVPR*, 2016. 2
- [24] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 2
- [25] T. Kim. Simulated+unsupervised learning in tensorflow. <https://github.com/carpedm20/simulated-unsupervised-tensorflow>. 7
- [26] G. Koch. Siamese neural networks for one-shot image recognition. In *ICML workshop*, 2015. 5
- [27] J. Lemley, S. Bazrafkan, and P. Corcoran. Smart augmentation-learning an optimal data augmentation strategy. *IEEE Access*, 2017. 3
- [28] C. Li, K. Xu, J. Zhu, and B. Zhang. Triple generative adversarial nets. In *NIPS*, 2017. 3
- [29] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang. Universal style transfer via feature transforms. In *NIPS*, 2017. 3
- [30] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. Scribble-sup: Scribble-supervised convolutional networks for semantic segmentation. In *CVPR*, 2016. 2
- [31] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 5
- [32] H. S. Mousavi, V. Monga, G. Rao, and A. U. Rao. Automated discrimination of lower and higher grade gliomas based on histopathological image analysis. *Journal of pathology informatics*, 2015. 9
- [33] V. Murthy, L. Hou, D. Samaras, T. M. Kurc, and J. H. Saltz. Center-focusing multi-task CNN with injected features for classification of glioma nuclear images. In *Winter Conference on Applications of Computer Vision (WACV)*, 2017. 2
- [34] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop*, 2011. 7, 9, 10
- [35] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *CVPR*, 2015. 5
- [36] A. Osokin, A. Chessel, R. E. C. Salas, and F. Vaggi. Gans for biological image synthesis. In *ICCV*, 2017. 2
- [37] C. Parmar, R. T. Leijenaar, P. Grossmann, E. R. Velazquez, J. Bussink, D. Rietveld, M. M. Rietbergen, B. Haibe-Kains, P. Lambin, and H. J. Aerts. Radiomic feature clusters and prognostic signatures specific for lung and head & neck cancer. *Scientific reports*, 2015. 1
- [38] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016. 2
- [39] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 3
- [40] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 3, 7
- [41] A. C. Ruifrok, D. A. Johnston, et al. Quantification of histochemical staining by color deconvolution. *Analytical and quantitative cytology and histology*, 2001. 4
- [42] J. Saltz, J. Almeida, Y. Gao, A. Sharma, E. Bremer, T. DiPrima, M. Saltz, J. Kalpathy-Cramer, and T. Kurc. Towards generation, management, and exploration of combined radiomics and pathomics datasets for cancer research. *AMIA Summits on Translational Science Proceedings*, 2017. 2
- [43] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *CVPR*, 2016. 3
- [44] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017. 2, 3, 5, 7
- [45] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. *ICLR*, 2017. 3
- [46] A. Telea. An image inpainting technique based on the fast marching method. *Journal of graphics tools*, 2004. 3, 4
- [47] T. Vicente, L. Hou, C.-P. Yu, M. Hoai, and D. Samaras. Large-scale training of shadow detectors with noisily-annotated shadow examples. In *ECCV*, 2016. 2
- [48] S. Wang, J. Yao, Z. Xu, and J. Huang. Subtype cell detection with an accelerated deep convolution neural network. In *MICCAI*, 2016. 2
- [49] Y. Xie, F. Xing, X. Kong, H. Su, and L. Yang. Beyond classification: structured regression for robust cell detection using convolutional neural network. In *MICCAI*, 2015. 2
- [50] J. Xu, L. Xiang, Q. Liu, H. Gilmore, J. Wu, J. Tang, and A. Madabhushi. Stacked sparse autoencoder (ssae) for nuclei detection on breast cancer histopathology images. *Medical Imaging*, 2016. 2
- [51] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Z. Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *MICCAI*, 2017. 2
- [52] Y. Zhang, L. Yang, J. Chen, M. Fredericksen, D. P. Hughes, and D. Z. Chen. Deep adversarial networks for biomedical image segmentation utilizing unannotated images. In *MICCAI*, 2017. 2
- [53] J. Zhao, L. Xiong, K. Jayashree, J. Li, F. Zhao, Z. Wang, S. Pranata, S. Shen, and J. Feng. Dual-agent gans for photorealistic and identity preserving profile face synthesis. In *NIPS*, 2017. 2, 3
- [54] N. Zhou, X. Yu, T. Zhao, S. Wen, F. Wang, W. Zhu, T. Kurc, A. Tannenbaum, J. Saltz, and Y. Gao. Evaluation of nucleus segmentation in digital pathology images through large scale image synthesis. *SPIE Medical Imaging. International Society for Optics and Photonics*, 2017. 2
- [55] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2005. 5