MultiPathGAN: Structure Preserving Stain Normalization using Unsupervised Multi-domain Adversarial Network with Perception Loss

Haseeb Nazki*,
School of Computer Science, University of St Andrews
St Andrews, UK
hn33@st-andrews.ac.uk

Ognjen Arandjelović
School of Computer Science, University of St Andrews
St Andrews, UK
oa7@st-andrews.ac.uk

In Hwa Um
School of Medicine, University of St Andrews
St Andrews, UK
ihu@st-andrews.ac.uk

David Harrison
School of Medicine, University of St Andrews
St Andrews, UK
david.harrison@st-andrews.ac.uk

ABSTRACT

Histopathology relies on the analysis of microscopic tissue images to diagnose disease. A crucial part of tissue preparation is staining where a dye is used to make the salient tissue components more distinguishable. However, differences in laboratory protocols and scanning devices result in significant confounding appearance variation in the corresponding images. This variation increases both human error and the inter-rater variability, as well as hinders the performance of automatic or semi-automatic methods. In the present paper we introduce an unsupervised adversarial network to translate (and hence normalize) whole slide images across multiple data acquisition domains. Our key contributions are: (i) an adversarial architecture which learns across multiple domains with a single generator-discriminator network using an information flow branch which optimizes for perceptual loss, and (ii) the inclusion of an additional feature extraction network during training which guides the transformation network to keep all the structural features in the tissue image intact. We: (i) demonstrate the effectiveness of the proposed method firstly on H&E slides of 120 cases of kidney cancer, as well as (ii) show the benefits of the approach on more general problems, such as flexible illumination based natural image enhancement and light source adaptation.

CCS CONCEPTS

• Applied computing → Health care information systems;

KEYWORDS

Digital pathology, multi-domain image translation, generative adversarial networks, semantic structure

*Corresponding Author

ACM Reference Format:
Haseeb Nazki, Ognjen Arandjelović, In Hwa Um, and David Harrison. 2023. MultiPathGAN: Structure Preserving Stain Normalization using Unsupervised Multi-domain Adversarial Network with Perception Loss. In The 38th ACM/SIGAPP Symposium on Applied Computing (SAC ’23), March 27-March 31, 2023, Tallinn, Estonia. ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.1145/3555776.3578612

1 INTRODUCTION

Digitization of pathological and histopathological slides in the medical imaging community has driven a significant increase in the development of computer aided diagnostic (CAD) systems [23]. One of the important steps in the process of tissue preparation for whole slide image (WSI) acquisition is staining, whereby histochemical stains such as hematoxylin and eosin (H&E) are used to highlight the intensity of tissue slice components and thus make tissue structures distinct from each other [2]. Staining facilitates the use of WSIs in manual or automated analysis by pathologists or CAD algorithms as a means of further the insight into the symptoms or mechanisms associated with a particular disease. However, slides collected from different laboratories exhibit diverse stain styles due to even minute variability in staining protocol, dyes, and scanners. This variability may not greatly hinder the analysis by an experienced pathologist; however, it significantly reduces the accuracy of existing CAD systems owing to their limited ability for generalization [29]. Meanwhile, given the massive amount of gigapixel-sized WSI data [7], there is a growing demand to build fast, automated, and scalable pipelines for large-scale image analysis as normalization increases the accuracy of machine learning (ML) algorithms that use stain-normalized images as input to a pre-trained deep network [10, 22].

Considering the importance of the problem, it is not surprising that a number of attempts at solving it have been described in the literature. Simple ones which focused on color matching [25], that is the color-channel alignment of a novel WSI with those of a reference template, lead to unrealistic looking results as a consequence of the lack of learning over a representative corpus and the reliance on a single target image. On the other hand, while stain-separation methods [21] do consider each staining channel independently for normalization, they fail to take into account the spatial features
and the structure of the components in the tissue. Moreover, they rely on a skillfully chosen reference template image which can significantly affect the outcome [28]. A related approach employs a sparse autoencoder [16], dividing an input image into multiple tissue regions and then independently normalizing each region using a single template. It can be readily seen that this approach suffers from the same limitation as that noted before, namely that a single reference template fails to capture the significant variation in the appearance of WSIs within a single lab.

More recently, deep learning-based approaches for stain normalization [27, 28, 32] using style transfer and image to image translation with adversarial learning methods [4, 8, 35] have made strides in first establishing and then repeatedly improving the state-of-the-art. However, all of the existing adversarial network based methods are capable of translating between only two specific domains which makes them unsuitable for the inherently multi-domain challenge posed by the real-world clinical practice. Some of these approaches even rely on supervision, learning a mapping between the two domains given a labelled set of input-output pairs which are not always readily available, particularly for WSI datasets. Moreover, while these methods transfer the style content between the domains efficiently, they too struggle at preserving the structure of the tissue components. Previous efforts to develop algorithms capable of preserving the anatomical structure of tissue in a WSI while also translating the style between domains in an unsupervised setting have shown decent results [18, 22]; nevertheless these methods too remain constrained by being able to deal with two specific domains only. Wagner et al. [33] use a GAN architecture based on disentangled representations [20], which produces realistic looking high level structures but which poorly semantically correspond to the input.

We introduce MultiPathGAN, a unified deep learning adversarial network for translating histopathological WSIs obtained from different labs across multiple domains for stain normalization while preserving the structure of the internal tissue components. The data adaptive nature of the MultiPathGAN means that it can also be readily employed in other applications wherein it is critical to constrain the effects of adversarial learning in a manner that retains the salient input structure, while only translating the style content across multiple domains. This alleviates the problem of variation in the feature representation of training data by having a unified representation that can map many-to-many stain style domains instead of a simple one-to-one representation. Importantly, our network takes in training data from multiple domains, and learns the mappings between all of them using only a single generator. Unlike StarGAN [3] which translates between multiple domains without effectively preserving the semantic content, we also achieve the preservation of the fine salient anatomical structure using an auxiliary feature extraction network and perception loss to minimize the perceptual distance between a real and the corresponding generated (synthetic), fake image.

**2 METHOD**

One of the key ideas in MultiPathGAN is the use of an auxiliary feature extraction network. This network is used to integrate perceptual loss in a multi-domain adversarial network, thereby introducing a semantic relationship between the input and the output, and drives the network to learn how to retain the salient structural elements of the input.

MultiPathGAN trains a single generator $G$ to learn mappings between multiple domains using target labels (Fig. 1). To achieve this, we train $G$ to translate an input image $x$ into an output image $y$ conditioned on the randomly generated target domain label ($y_{trg}$), such that $G(x, y_{trg}) = y$. The target domain label $y_{trg}$ is generated randomly so that $G$ learns to translate the input image flexibly [3]. Further, an auxiliary classifier allows a single discriminator to control multiple domains by producing probability distributions over both sources and domain labels [3]. Note that the single generator is used twice, first to translate the original image into the target domain and then to reconstruct the original image from the translated image. Simultaneously, the input image $x$ and output image $y$ are fed to a pre-trained deep convolutional classifier $C$ to obtain feature maps $x^\phi$, $G(x, y_{trg})^\phi$ and obtain the mean squared distance between these activations [19]. This distance is used in an additional perception loss function to help preserve the perceptually salient content in the input image as it is translated to the output domain [19, 24]. In the subsections which follow we provide detailed descriptions of all loss functions used in training the network that make this preservation possible.

**Adversarial Loss.** As in previous work, $G$ tries to generate an image $G(x, y_{trg})$ conditioned on both the input image $x$ and the target domain label $y_{trg}$. At the same time, a discriminator $D$ tries to distinguish between real and fake images, $x$ and $G(x, y_{trg})$ respectively. To stabilize the training process and generate higher quality output of each mapping that matches the empirical distribution of the target domain, while focusing on the source domain, we adopt

\[
C = \delta(x^\phi, G(x, y_{trg})^\phi)
\]
the Wasserstein adversarial loss [1] with gradient penalty [9]:
\[
L_{\text{Adv}} = \mathbb{E}_x[D_N(x)] - \mathbb{E}_{y_{\text{trg}}}[D_N(G(x, y_{\text{trg}}))] - \lambda_g \mathbb{E}_{\hat{x}}[(\|\nabla_D S_{\hat{x}}\|_L^2 - 1)^2]
\]  
(1)
where \(\lambda_g\) is the coefficient for gradient penalty and \(\hat{x}\) is uniformly sampled along a straight line between a real and a generated image pair [9]. \(D_N\) refers to the probability distribution over the sources given by discriminator \(D\).

**Domain Classification Loss.** We also adopt the domain classification loss [3] using an auxiliary classifier on top of \(D\) when optimizing \(G\) and \(D\) to ensure that the input \(x\) translated into an output image \(y\) is properly classified to the target domain \(y_{\text{trg}}\). This objective can be broken down into two terms: a domain classification loss calculated using real images to optimize \(D\), and a domain classification loss calculated using fake images to optimize \(G\):
\[
L_C^f = \mathbb{E}_{x, y_{\text{trg}}}[\log D_C(y_{\text{trg}}|x)]
\]  
(2)
\[
L_C^r = \mathbb{E}_{x, y_{\text{org}}}[\log D_C(y_{\text{org}}|G(x, y_{\text{trg}}))]
\]  
(3)
where \(D_C\) is the probability distribution calculated by \(D\) over domain labels. In (2), \(D\) tries to minimize the objective by learning to classify a real image \(x\) to its corresponding original domain \(y_{\text{org}}\). At the same time, in (3), \(G\) tries to minimize the objective to generate images that could be classified as the target domain \(y_{\text{trg}}\).

**Reconstruction Loss.** To ensure that the translated images preserve high level features of the input while changing the domain specific characteristics, we use a cycle consistency loss with the generator:
\[
L_C = \mathbb{E}_{x, y_{\text{org}}}[\|x - G(G(x, y_{\text{trg}}), y_{\text{org}}))\|_1]
\]  
(4)
where \(G\) takes in the translated image \(G(x, y_{\text{trg}})\) \(\rightarrow y\) and the original domain label \(y_{\text{org}}\) as input and tries to reconstruct the original image \(x\) using the L1 norm in the loss function.

**Perceptual Loss.** In addition to the reconstruction loss used to retain the high level features, we also include a perceptual loss which preserves the fine structure of the input image during domain transfer as the perceptual distance is minimized [17] between the generated \(G(x, y_{\text{trg}})\) and the original image \(x\). This distance is calculated by feeding both the original and the transformed image to a pre-trained classification network \(C\) and extracting their feature maps \(x^\phi\), \(G(x, y_{\text{trg}})^\phi\) respectively, at spatial resolution \(\phi\):
\[
L_P = \delta(x^\phi, G(x, y_{\text{trg}})^\phi)
\]  
(5)
The mean squared distance is calculated for each feature map and the total perceptual distance is then a weighted sum of the \(L_2\) distances. By minimizing this distance, we implicitly compel the transformed image to contain the same structural meaning as the original image as perceived by a pre-trained classification network.

**Final Objective.** Putting everything together, the final training objective can be written as:
\[
L_D = -L_{\text{Adv}} + \lambda_C L_C^r + \lambda_C L_C^f
\]  
\(\lambda_C\) is the hyper-parameters to regulate the individual loss functions \(L_{\cdot}\).

**Implementation.** In our network, the generator \(G\) is based on the six block ResNet [11] backbone and a pair of convolutional and transposed convolutional layers with the stride size of two on either side, respectively. We use instance normalization for the generator, which has been shown to improve performance [30]. For the discriminator \(D\), we use PatchGANs [15] with no normalization and Leaky ReLU with a negative slope of 0.01 to classify the overlapping real and fake patches. Finally, for our feature extraction network \(C\), we utilize a pre-trained 34-layer residual network [11]. We use the activations from the last convolutional layer of the ResNet feature extraction network to concentrate on the fine changes in these higher levels, propagated by its skip connections, which thus allows us to determine a better perceptual distance between the generated \(G(x, y_{\text{trg}})\) and original image \(x\).

We set \(\lambda_g = 10, \lambda_C = 1, \lambda_{C_{yc}} = 10 & \lambda_P = 0.75\), use a mini-batch size of 16 and a base learning rate of \(10^{-5}\) for both \(G\) and \(D\) which decays after every 10 epochs, and Adam optimizer [6] with \(\beta_1 = 0.5\) & \(\beta_2 = 0.999\). Following Gulrajani et al. [9], we perform one \(G\) for every five \(D\) updates over 80 epochs, taking 21h to train on Nvidia DGX-1.

### 3 EVALUATION

**Data.** H&E slides were prepared from a cohort of 120 cases of kidney cancer obtained from the Pathology Archives, Lothian NHS. Sections were cut at 3 \(\mu\)m and routinely stained for H&E before being cover-sliped. Slides containing both cancerous and non-cancerous “normal” tissue were distributed to different laboratories and scanned locally on their scanners of choice before all images were collated at <removed for review>. Ethics approval was provided by Lothian NHS Biorepository (ES/15/0094). We used WSI acquired from Hamamatsu NanoZoomer S60 at 20 \(\times (0.42\mu m/pix)\) magnification, Hamamatsu NanoZoomer 2.0-HT at 20 \(\times (0.45\mu m/pix)\) and Leica Aperio AT2 at 20 \(\times (0.50\mu m/pix)\) magnifications, respectively, to create three dataset domains. The WSIs were then split into 256 \(\times 256\) non-overlapping patches. For training we used 10 WSIs for each domain and randomly extracted 1250 training and 200 testing sample patches for each class. We also used an additional domain from Hitachi HV-F202SCL at 20 \(\times (0.22\mu m/pix)\) magnification, unseen to the network in training. This domain was used to evaluate the generalization of our network and its ability to adapt to the high variance of WSIs across histopathological corpora.

Fig. 2 shows examples of WSIs and their respective tiles from our dataset. Even though the variation in the acquired data is readily apparent to the naked eye, we corroborate this in Fig. 2(c) using t-Distributed Stochastic Neighbor Embedding (t-SNE), which helps visualize high dimensional data by assigning each datum a location in a 2D or 3D map [31]. In our case, to concentrate on the specific features resulting in these variations in the WSI domains, the image embeddings are extracted from the penultimate layer of a pre-trained deep convolutional neural network trained on images of paintings by various artists. We chose the painters dataset specifically, to aid the network in distinguishing between the color spaces of different WSI domains, as well as to help us analyse the performance of our method by facilitating a meaningful visualization of WSI clusters and the distances between them. Although the primary motivation behind our work was biomedical in nature,
in order to demonstrate the wider effectiveness of the proposed method, we also evaluated it on Virtual Image Dataset for Illumination Transfer [12].

We derive five domains from the illumination setting at five different (2500K, 3500K, 4500K, 5500K and 6500K) color temperatures. All the domains have 2400 samples and we use the train:test split of 9:1 at 256, 512 and 1024 input-output resolutions.

### 3.1 Experimental Results

**Metrics.** To quantify how well different methods preserve the salient structures present in the original patches, we use Peak Signal-to-Noise Ratio (PSNR), Structural Similarity (SSIM) [14], Multi-Scale Structural Similarity Index (MS-SSIM) [34] and Haar Wavelet-Based Perceptual Similarity Index (HaarPSI) [26] measures. While PSNR depends purely on the pixel-wise differences in pixel value between the two compared images, SSIM, MS-SSIM and HaarPSI are more correlated with the perceptual differences between them. These metrics have high sensitivity to detect distortions between images as well as the differences arising due to luminance and contrast changes.

**Qualitative Analysis.** We train and evaluate MultiPathGAN with three WSI domains. Fig. 3 shows typical examples of translation results. As confirmed independently by a pair of experienced pathologists, our method provides high translation quality, appropriately adapting style (specific domain characteristics), while correctly preserving salient anatomical structures (cell type and morphology, etc.). In addition, we also translate patches from the fourth unseen WSI domain to the three domains from our dataset used for training. As can be seen in Fig. 4, MultiPathGAN can effectively translate, and hence normalize, data from an alien subset of the WSI space into a known domain. This shows that normalizing unseen subsets of WSI color space using MultiPathGAN could be achieved without retraining the whole network. However, considering that this is an unseen domain, the inverse translation cannot be achieved without domain specific training, i.e. without the parameters which correspond to this domain, along with the corresponding target labels.

We also use t-SNE to visualize the effect of our translation upon the normalization of WSI patches in the deep neural network feature space. From Fig. 5 it can be seen that the patches normalized using MultiPathGAN match well with the distinctive distribution of the target class. This can provide an intuition for the improved performance in terms of evaluation metrics as discussed in the next subsection. In addition to Fig. 4, we can also use Fig. 5(d) to extrapolate the effectiveness of MultiPathGAN in translating an unseen WSI domain into a known WSI domain without the need to retrain the entire network.

Finally, in Fig. 6 we compare our cross-domain MultiPathGAN to the existing methods in the literature and observe that our method not only provides high quality and perceptually meaningful WSI normalization results, but also that it can normalize multiple WSI domains simultaneously, using a single generator in a single training session. While the method of Runz et al. [27] effectively translates the style between two domains, it fails at preserving the quality of the final image and resulting in blur. Mahapatra et al. [22] have shown that using semantic guidance in addition to adversarial and cycle reconstruction loss in a CycleGAN, we can preserve the detailed structural information, however, their method is restricted to only two WSI domains. In their attempt, Vasiljević et al. [32] have used StarGAN to translate between multiple staining modalities, but when we consider stain variations in a single modality, it fails at preserving the fine structure of the tissue components in the translated output. In addition to the StarGAN, we also show results from its newer variant by Choi et al. [4] and found that the generator produces images that are perceptually valid but slightly blurry and completely out of context with regards to the input WSI tissue structure. Translated samples from Wagner et al. [33] look realistic when compared to StarGAN but it struggles at preserving
Figure 3: Multi-domain translation results using MultiPathGAN on our WSI dataset for inter-domain normalization. The column on the left shows the input WSI patch while the remaining columns on the right show the translated output for the three domains as per the target label. This is achieved using a single generator.

Figure 4: Translations from the unseen Hitachi-SCL (leftmost column) domain to our three dataset domains used for training MultiPathGAN generator.

the detailed structural information and introduces objects that are out of context from the original image.

Quantitative Analysis. We first report the performance metrics for translations between two domains, namely those of the generated image $G(x, y_{trg})$ and of the original image $x$, for different methods in Table 1. We compute the evaluation metrics between a real image and the same image translated to other domains in all comparative assessments. As can be seen, our model performs according to all metrics, evidencing that it produces the most realistic WSI patches by preserving the fabrication of the tissue as well as normalizing the stains. These outputs match the style of the target domain and preserve the structure of the input patch, thereby corroborating the findings of our previous qualitative analysis.

To evaluate our model on the unseen data, we translate between the three test domains including the additional unseen class and compare them with each other using the similarity metrics. From Table 2 we can notice that our model performs well with the domain transfer between the test data for all the cases. However, when we translate an unseen domain (here HitachiSCL) to one of the domains used at the time of training MultiPathGAN, there is a decrease in performance in terms of quantitative evaluation metrics. This difference is due to the lack of knowledge about the distance of the parameter distribution of our trained network from the distribution of the unseen domain and would remain constant for all such domains, provided they come from a similar locus. Nevertheless, these translations for unseen domains still deliver a good result and increasing the number of data domains to train MultiPathGAN would make the generator invariant to these unseen cohorts. This in return would further improve the performance of MultiPathGAN at normalizing unseen WSI domains.

Loss Analysis. To evaluate the effect of individual loss components, we add them successively to our general baseline architecture (comprising our generator with a ResNet backbone, a PatchGAN discriminator, and adversarial loss $L_{Adv}$). We illustrate the effects in Fig. 7, translating Hamamatsu-HT $\rightarrow$ Hamamatsu-S60. Moreover, Table 3 provides a comparison in terms of quantitative metrics for each configuration.

As can be seen from Fig. 7(a) and Table 3, using adversarial loss helps translate the style of the input to the target output. However,
it is also clear that without any semantic supervision it cannot produce a result which retains the correct semantic content, that is the relevant anatomical, structural information in the present case. Adding semantic guidance using reconstruction loss effects an improvement, see Fig. 7(b) and Table 3; nevertheless, it fails to preserve fine-grained semantic information. This can be seen
Table 2: Comparison of translation performance of MultiPathGAN between seen and unseen domain pairs (HT: Hamamatsu-HT, S60: Hamamatsu-S60, AT2: Leica-AT2 and SCL: Hitachi-SCL).

| Domain Transfer | PSNR | SSIM | MS-SSIM | HaarPSI |
|-----------------|------|------|---------|---------|
| HT-to-S60       | 22.96| 0.96 | 0.98    | 0.94    |
| AT2-to-S60      | 27.21| 0.95 | 0.95    | 0.96    |
| S60-to-HT       | 26.98| 0.94 | 0.95    | 0.95    |
| AT2-to-HT       | 35.69| 0.97 | 0.99    | 0.98    |
| S60-to-AT2      | 22.63| 0.96 | 0.97    | 0.93    |
| HT-to-AT2       | 36.47| 0.98 | 0.99    | 0.98    |
| SCL-to-S60      | 18.50| 0.90 | 0.94    | 0.85    |
| SCL-to-HT       | 23.77| 0.90 | 0.93    | 0.87    |
| SCL-to-AT2      | 19.32| 0.88 | 0.94    | 0.87    |

Table 3: Comparison of effects of different MultiPathGAN losses.

| Loss                | PSNR | SSIM | MS-SSIM | HaarPSI |
|---------------------|------|------|---------|---------|
| $L_{Adv}$           | 22.04| 0.85 | 0.93    | 0.90    |
| $L_{Adv} + L_{CyC}$ | 14.05| -0.22| 0.03    | 0.56    |
| $L_{Adv} + L_{CyC} + L_{C}$ | 37.76| 0.97 | 0.99    | 0.99    |
| $L_{Adv} + L_{CyC} + L_{P}$ | 22.79| 0.93 | 0.97    | 0.91    |
| $L_{Adv} + L_{CyC} + L_{C} + L_{P}$ | 22.96| 0.96 | 0.98    | 0.93    |

In Fig. 8, we show the generator producing images with artifacts when we only use reconstruction loss. Although adding classification loss forces the network to classify the input image into the corresponding target image, which aids in rendering it into the style of the target domain, it again fails in preserving the structure of the input image; see Fig. 7(c) and Table 3. However, the effectiveness of classification loss can be observed in its absence in Fig. 7(d) where the unified effect of reconstruction and perception loss can be seen. The resulting images highly correlate with the input in terms of both content and style, resulting in the least perceptual distance (see Table 3), demonstrating that the network fails to translate to the target domain without classification loss.

**Qualitative results on VIDIT.** While emphasizing that our primary goal was that of strain translation in digital pathology, herein we also include experimental results which demonstrate the potential of our method on entirely different kinds of data. In particular, we show empirically that our method can maintain structural consistency between the original images and the transferred images irrespective of the discrepancy in domain composition and style more generally using the VIDIT data set. As can be seen from Fig. 9, MultiPathGAN can synthetically render a scene as if re-illuminated using an unseen lighting setup, which is useful in a variety of tasks, for example in reference-based image relighting.

**Mean Color Difference.** To measure the color difference between the input image, output image and the target image, we calculate the difference of euclidean distances between their corresponding pixels in the CIELAB color space. [5, 13]. The resultant mean difference with higher values indicates a better match between the generated fake image and the target image. This value for MultiPathGAN comes out to be -0.0036 followed closely by Vasiljevic et al. [32] [-0.0039]. For Runz et al. [27], Mahapatra et al. [22] and Wagner et al. [33] this metric value was -0.0054, -0.0055 and -0.0056 respectively.

### 4 DISCUSSION AND CONCLUSIONS

Considering the widespread use of convolutional neural networks in CAD systems, stain normalization plays a crucial part in interoperability and recommendation accuracy. MultiPathGAN provides a means of removing variations in acquired images by providing clinically meaningful output normalized at the pixel resolution. There are several reasons for its robustness, flexibility, and accuracy in WSI normalization when compared to all previously proposed methods. First, unlike these, MultiPathGAN learns the mapping from the entire dataset instead of relying on a single reference image, thus avoiding the complexity of choosing reference images. Second, most existing methods that use adversarial learning methods focus on using GANs to translate only between two domains, thus limiting the potential to translate between multiple domains simultaneously using a single generator and style code. Moreover, all existing methods are ineffective in that even if multiple generators are used to translate between different domain pairs, the learning of global features available to each generator is done separately. Hence, the entirety of information available within the training data corpus is not used to its full extent. This inability to fully utilize training data by jointly training across domains inevitably limits the quality of generated samples and the ability of the generator to successfully normalize unseen WSI domains. In MultiPathGAN the shared data from each domain helps to learn domain-invariant
features which produces a regularization effect, thus facilitating better generalization to unseen WSI samples.

Lastly, we analysed different loss functions and introduced a pre-trained auxiliary ResNet feature extraction network to calculate a perceptual distance. This distance is employed in the additional perceptual loss function used while training the generator to preserve fine detail of input WSI patches in the synthetic output. Finer features are better propagated to higher convolutional levels in residual networks owing to skip connection and an implicit weighing between local and global features. We use the activations from the final layers of the ResNet feature extraction network to enable us to retain the finer details of the original image during the training process. This helps us to produce images which preserve the fine structure of the input image while effectively translating to the style of the target image domain.

REFERENCES

[1] Martin Arjovsky, Soumith Chintala, and Leon Bottou. 2017. Wasserstein generative adversarial networks. In ICLR.
[2] Peter D Caie, Neofytos Dimitriou, and Ognjen Arandjelovic. 2021. Precision medicine in digital pathology via image analysis and machine learning. In Artificial intelligence and deep learning in pathology, Elsevier.
[3] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sung hun Kim, and Jaegul Choo. 2018. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In CVPR.
[4] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. 2020. Stargan v2: Diverse image synthesis for multiple domains. In CVPR.
[5] Christine Collins and Thomas Flessa. 1997. A study of efficiency and accuracy in the transformation from RGB to CIELAB color space. IEEE transactions on image processing 6, 7 (1997), 1046–1048.
[6] Kingma Da. 2014. A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
[7] Neofytos Dimitriou and Ognjen Arandjelovic. 2021. Magnifying networks for images with billions of pixels. arXiv preprint arXiv:2112.06121 (2021).
[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. NIPS (2014).
[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. NIPS (2017).
[10] Laxmi Gupta, Barbara M Klinkhammer, Peter Boor, Dorit Merhof, and Michael Gudernay. 2019. GAN-based image enrichment in digital pathology boosts segmentation accuracy. In MICCAI.
[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In CVPR.
[12] Majed El Helou, Ruofan Zhou, Johan Barthas, and Sabine Süsstrunk. 2020. VIDIT: Virtual image dataset for illumination transfer. arXiv preprint arXiv:2005.05460 (2020).
[13] Bernhard Hill, Th Roger, and Friedrich Wilhelm Vorhagen. 1997. Comparative analysis of the quantization of color spaces on the basis of the CIELAB color-difference formula. ACM Transactions on Graphics (TOG) 16, 2 (1997), 109–154.
[14] Alain Hore and Djemel Ziou. 2010. Image quality metrics: PSNR vs. SSIM. In IJIP.
[15] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In CVPR.