Diagnostic Pathology

RESEARCH Open Access

Normalization of HE-stained histological images using cycle consistent generative adversarial networks

Marlen Runz[1,2*], Daniel Rusche[1], Stefan Schmidt[3], Martin R. Weihrauch[4], Jürgen Hesser[2,5,6] and Cleo-Aron Weis[1]

Abstract

Background: Histological images show strong variance (e.g. illumination, color, staining quality) due to differences in image acquisition, tissue processing, staining, etc. This can impede downstream image analysis such as staining intensity evaluation or classification. Methods to reduce these variances are called image normalization techniques.

Methods: In this paper, we investigate the potential of CycleGAN (cycle consistent Generative Adversarial Network) for color normalization in hematoxylin-eosin stained histological images using daily clinical data with consideration of the variability of internal staining protocol variations. The network consists of a generator network G_B that learns to map an image X from a source domain A to a target domain B, i.e. $G_B : X_A \rightarrow X_B$. In addition, a discriminator network D_B is trained to distinguish whether an image from domain B is real or generated. The same process is applied to another generator-discriminator pair (G_A, D_A), for the inverse mapping $G_A : X_B \rightarrow X_A$. Cycle consistency ensures that a generated image is close to its original when being mapped backwards ($G_A(G_B(X_A)) \approx X_A$ and vice versa). We validate the CycleGAN approach on a breast cancer challenge and a follicular thyroid carcinoma data set for various stain variations. We evaluate the quality of the generated images compared to the original images using similarity measures. In addition, we apply stain normalization on pathological lymph node data from our institute and test the gain from normalization on a ResNet classifier pre-trained on the Camelyon16 data set.

Results: Qualitative results of the images generated by our network are compared to original color distributions. Our evaluation indicates that by mapping images to a target domain, the similarity training images from that domain improves up to 96%. We also achieve a high cycle consistency for the generator networks by obtaining similarity indices greater than 0.9. When applying the CycleGAN normalization to HE-stain images from our institute the kappa-value of the ResNet-model that is only trained on Camelyon16 data is increased more than 50%.

Conclusions: CycleGANs have proven to efficiently normalize HE-stained images. The approach compensates for deviations resulting from image acquisition (e.g. different scanning devices) as well as from tissue staining (e.g. different staining protocols), and thus overcomes the staining variations in images from various institutions.

(Continued on next page)

*Correspondence: marlen.runz@medma.uni-heidelberg.de
[1] Institute of Pathology, University Medical Centre Mannheim, Heidelberg University, Mannheim, Germany
[2] Mannheim Institute for Intelligent Systems in Medicine (MIISM), Medical Faculty Mannheim, Heidelberg University, Mannheim, Germany
Full list of author information is available at the end of the article

(Continued from previous page)
The code is publicly available at https://github.com/m4ln/stainTransfer_CycleGAN_pytorch. The data set supporting the solutions is available at https://doi.org/10.11588/data/8LKEZF.

Keywords: Histology stain normalization; HE-stain; Digital pathology; Generative adversarial networks; Unpaired image-to-image translation; Style transfer; Deep learning

Background

In both histology and surgical pathology, the inherent individual appearance of the considered object on the one hand or the different staining protocols on the other hand must be compensated in addition to factors that influence the image acquisition (e.g scanning devices). This demand applies to hematoxylin-eosin (HE) staining being the standard method in pathology but also to all other histochemical and immunohistochemical staining. Regarding HE-staining, solutions and protocols are standardized at first glance. However, even within a single institution, protocols may vary slightly and may not be coordinated with other institutes. Especially when training deep neural networks, for example for image classification, there is a need for stain normalization of images so that models are transferable to other data sets.

The term color normalization is an umbrella term for image processing techniques compensating for effects such as variable illumination, camera setting, etc. This evident request drives an active research. Conventional image processing such as color deconvolution or look-up tables with the need for selecting a reference template slide for normalization are widespread [1–6]. A particular but quite similar issue is stain quantification [7, 8]. Recent publications investigated in the use of deep learning approaches with Generative Adversarial Networks (GANs) and showed the benefits compared to the conventional methods [9, 10]. It was also shown how normalizing images using GANs can highly improve results of image classification [11] or segmentation [12]. Mahapatra et al. [13] integrate self-supervised semantic information such as geometric and structural patterns at different layers to improve stain normalization with CycleGANs.

In this work, we investigate the potential and limitation of a machine learning-based approach for normalization with a cycle consistent Generative Adversarial Network (CycleGAN) which learns the mapping from one HE-stain variant to an other. The approach we follow was proposed by Zhu et al. [14]. An image-to-image mapping is learned between two different HE-stained data sets to generate fake images in each image domain. We apply the technique to two independent data sets: the Mitosis-Atypia-14 challenge which provides two image sets of breast cancer tissue scanned with two different devices, and our *HE-Staining Variation* (HEV) data set, showing follicular thyroid carcinoma slices stained with different protocols. We evaluate the results using the Fréchet Inception Distance (FID) and the Structural Similarity Index Measure (SSIM). In addition, we apply our method to a tumor classification task of lymph node samples, in which we apply stain normalization on pathological lymph node data from our institute and test the gain from normalization on a ResNet classifier pre-trained on the Camelyon16 data set. We show that normalization can increase the confidence of the classifier not only prior to the training, but also when the test data is normalized on the training set and applied to the model after training. In addition to that, we demonstrate the potential of using this normalization for recoloring images with stains that are not present in the original data; and at the same time we achieve a higher accuracy in our classification task.

We provide both, our implementation at https://github.com/m4ln/stainTransfer_CycleGAN_pytorch, and our data set for normalization and classification at https://doi.org/10.11588/data/8LKEZF.

Methods

CycleGAN formulation

The CycleGAN from Zhu et al. [14] consists of two generator and discriminator pairs each of which learns the mapping from one image domain to the other. Given the image domains A and B with training images X_A and X_B, the generator G_B learns the mapping from A to B such that $G_B : X_A \rightarrow X_B$, while the generator G_A learns the mapping in reverse direction, i.e. $G_A : X_B \rightarrow X_A$. A discriminator D is a binary classifier. It decides whether a sample is real (1), i.e. given from the training data set, or fake (0), i.e. produced by the generator. More precisely, discriminator D_B learns to distinguish between real images X_B^{real} and generated ones X_B^{fake}, while in the same way, D_A is trained to discriminate between X_A^{real} and X_A^{fake}.

For training, the objective function to be optimized is modeled by two loss functions: the adversarial loss $\mathcal{L}^{adv}$ [15] and the cycle consistency loss $\mathcal{L}^{cyc}$ [14].

Adversarial loss

Introduced by Goodfellow et al. [15] the adversarial loss refers to the two-player game between the generator and the discriminator networks. More precisely, for the mapping $G_B : X_A \rightarrow X_B$, the discriminator D_B is trained to classify X_B^{real} and X_B^{fake} correctly, while the generator seeks X_B^{fake} being classified as real by the discriminator. In this way, both, the generator and the discriminator try to fool the other. Zhu et al. [14] use the least-squares loss as objective since it ensures stability during training and generates higher quality results. Thus, the adversarial loss function is expressed as follows [14]:

$$\min_{G_B}\max_{D_B}\mathcal{L}_B^{adv} = \mathbb{E}_{X_B^{real}}\left[D_B\left(X_B^{real}\right)^2\right] + \mathbb{E}_{X_A^{real}}\left[\left(D_B\left(G_B\left(X_A^{real}\right)\right)-1\right)^2\right],$$

with $\mathbb{E}$ being the expected value over all instances of X_A^{real} and X_B^{real}. In the same way, we can formulate the adversarial loss for the inverse mapping function $G_A : X_B \to X_A$, i.e.

$$\min_{G_A}\max_{D_A}\mathcal{L}_A^{adv} = \mathbb{E}_{X_A^{real}}\left[D_A\left(X_A^{real}\right)^2\right] + \mathbb{E}_{X_B^{real}}\left[\left(D_A\left(G_A\left(X_B^{real}\right)\right)-1\right)^2\right],$$

Thus, the total adversarial loss $\mathcal{L}^{adv}$ is obtained by the sum of both terms $\mathcal{L}_A^{adv}$ and $\mathcal{L}_B^{adv}$.

Cycle consistency loss

Zhu et al. [14] presented this loss function to enforce that both mapping functions G_A and G_B learned by the generators are inverse functions. In other words, if an image is mapped from one domain to the other domain the backward mapping should bring the image back to its original state. Thus, it must satisfy the cycle $X_A^{real} \to G_B(X_A^{real}) \to G_A(G_B(X_A^{real})) = X_A^{rec} \approx X_A^{real}$ and in the same way for $X_B^{real} \to G_A(X_B^{real}) \to G_B(G_A(X_B^{real})) = X_B^{rec} \approx X_B^{real}$ for the backward mapping. Therefore, the total cycle consistency is given by:

$$\mathcal{L}^{cyc} = \underbrace{\mathbb{E}_{X_A^{real}}\left[\left\|G_A\left(G_B\left(X_A^{real}\right)\right)-X_A^{real}\right\|_1\right]}_{\mathcal{L}_A^{cyc}} + \underbrace{\mathbb{E}_{X_B^{real}}\left[\left\|G_B\left(G_A\left(X_B^{real}\right)\right)-X_B^{real}\right\|_1\right]}_{\mathcal{L}_B^{cyc}},$$

where $\|\cdot\|_1$ denotes the ℓ_1-Norm.

Hence, the total loss function is

$$\arg\min_{G_B,G_A}\arg\max_{D_B,D_A}\mathcal{L} = \mathcal{L}^{adv} + \lambda\mathcal{L}^{cyc},$$

with λ being a regularization factor to control the relative importance of both, adversarial and cycle consistency losses.

Figure 1 illustrates the CycleGAN structure for mapping an image from domain A to domain B by $G_B : X_A^{real} \to X_B^{fake}$ and backwards by $G_A : X_B^{fake} \to X_A^{rec}$. The discriminator D_B tries to identify if an image is generated X_B^{fake} or real X_B^{real}. During training, the network is optimized by computing the adversarial loss $\mathcal{L}^{adv}$ and the cycle consistency loss $\mathcal{L}^{cyc}$. The same process is done for the reverse direction when a real sample image X_B^{real} is mapped from domain B to domain A, i.e $X_B^{real} \xrightarrow{G_A} X_A^{fake} \xrightarrow{G_B} X_B^{rec}$.

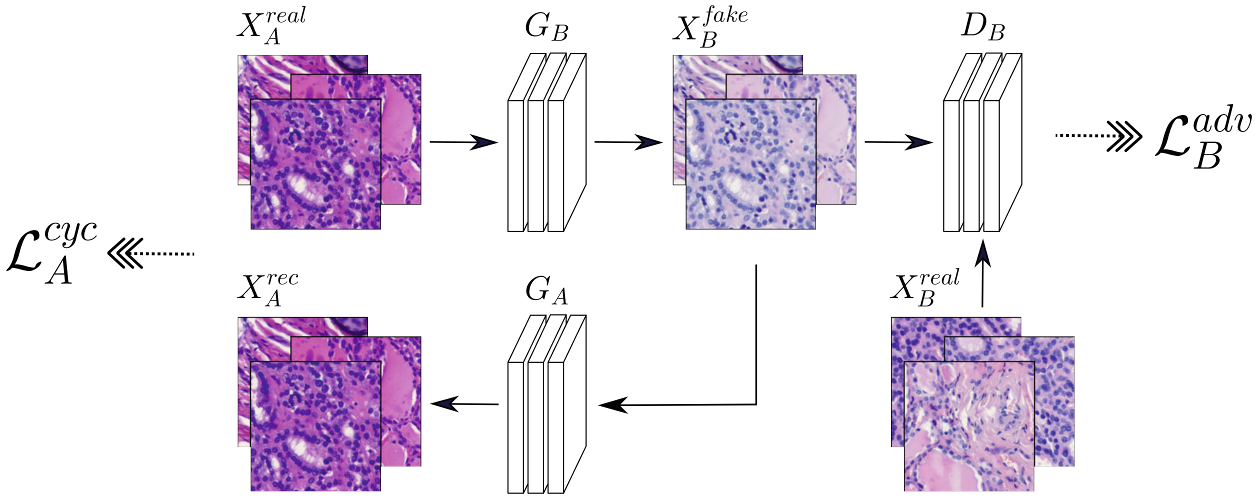

Fig. 1 Illustration of the applied CycleGAN architecture for mapping images from domain *A* to domain *B*. A real sample image X_A^{real} is mapped to domain *B* by the generator $G_B : X_A^{real} \to X_B^{fake}$ and then back to domain *A* by the generator $G_A : X_B^{fake} \to X_A^{rec}$. The discriminator D_B differentiates between the generated image X_B^{fake} and a real sample image X_B^{real}. The same process is done for the reverse direction when mapping a real sample image X_B^{real} from domain *B* to domain *A* and backwards, i.e $X_B \xrightarrow{G_A} X_A^{fake} \xrightarrow{G_B} X_B^{rec}$. During training, the loss is computed by the adversarial loss $\mathcal{L}^{adv}$ and the cycle consistency loss $\mathcal{L}^{cyc}$

Data sets

Two data sets are used to asses the quality of stain normalization using CycleGAN: **(a)** The Mitos-Atypia-14 challenge data set in which the HE-stain in images appears different in color and resolution due to different scanning devices. **(b)** Our clinical HE-Staining Variation (HEV) data set, which contains images of serial sections that were subjected to different staining protocols. In addition, we apply our CycleGAN normalization method in a tumor lymph node classification task trained on the Camelyon16 data set [16] and tested on the TumorLymphnode data set collected at our institute.

Mitos-Atypia-14

This is a publicly available challenge data set containing breast-cancer images [17]. The tissue was HE-stained and scanned by two different whole-slide image (WSI) scanners: the Aperio ScanScope XT and the Hamamatsu Nanozoomer 2.0-HT. Both devices scan images with different resolutions, the Aperio 1539 × 1376 pixels and the Hamamatsu 1663 × 1485 pixels at X20 and X40 magnification. From each scanned set, 7,936 tiles are selected for training and 15,000 tiles for testing. We resize the images to 1024 × 1024 pixels and extract image tiles of 256 × 256 pixels as input to our network.

HE-Staining variation

At the Institute of Pathology, Medical Faculty Mannheim, Heidelberg University we collected serial sections of a follicular thyroid carcinoma and stained them with the following HE-staining variants: standard protocol (of the Institute of Pathology, Mannheim) HE-stain (henceforth *HE*), intentionally stained too short (henceforth *shortHE*), intentionally stained too long (henceforth *longHE*), only stained with hematoxylin (henceforth *onlyH*), and only stained with Eosin (henceforth *onlyE*). Figure 2 shows thumbnails from each WSI. For each set, we extract tiles of 256 × 256 pixels. We collect 10,000 and 15,000 tiles for training and testing, respectively. The whole data set including our training patches is made publicly available [18].

Image classification

The effect of HE-staining normalization is applied on a tumor classification task using two sets of data: **(1)** The Camelyon16 data set [16] containing the classes normal lymph node ($n = 3,318$) and carcinoma infiltration ($n = 3,591$), **(2)** Slides with normal lymph nodes and carcinoma infiltration from an interval of 15a (hereinafter called TumorLymphnode or short TL data set) collected from the Institute of Pathology, Heidelberg University. These are segmented manually and then cropped into tiles ($n = 3605$ with normal lymph node parts, $n = 3,600$ with carcinoma infiltration).

Training details

CycleGAN

In the first setting, we train our model on the Mitos-Atypia-14 challenge to learn the mapping between the two image sets X_A and X_B obtained by the scanners Aperio and Hamamatsu, respectively. We then train four additional models on the HEV data set, with set A being the standard stained tissue (see Fig. 2 **A**) and set B being one of the other stained tissues (see Fig. 2 **B**-**E**). For each model we choose 10,000 images for training. An overview of the experiments on stain normalization are given in the Appendix.

For the classification task, we train a CycleGAN to normalize the Camelyon16 data set to the standard HE and hematoxylin staining of the HEV data set and to the TL data set. In the same way we train another model to normalize the TL data set to the Cameylon16 data set and to the HE and hematoxylin staining of the HEV data set. For each image set we choose 5,000 images for training.

For all models we use the same network architecture as described by Zhu et al. [14]. We train each network for 60

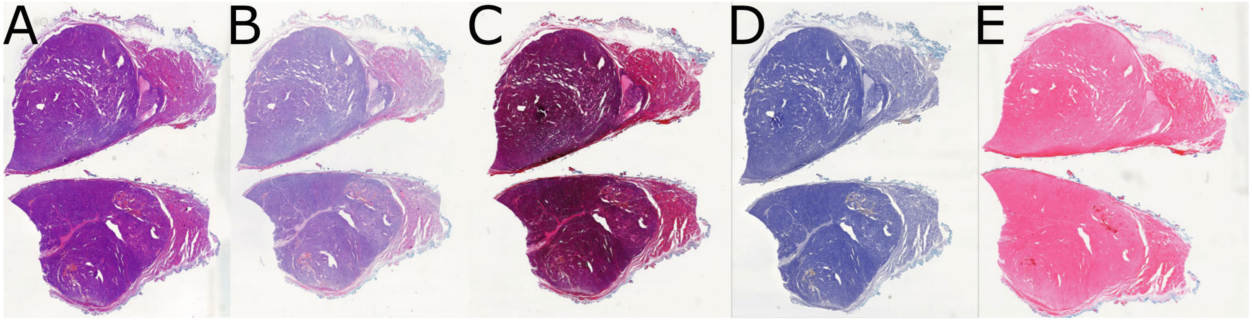

Fig. 2 Exemplary miniature image of the WSI that forms the HEV data set.
Serial tissue sections from a thyroid tissue with a follicular carcinoma with HE-staining. For every slide the staining protocol is intentionally modified: **A**) Standard protocol at the Institute of Pathology, Medical Faculty Mannheim, Heidelberg University (*HE*) **B**) Shortened staining time (*shortHE*) **C**) Prolonged staining time (*longHE*) **D**) Only hematoxylin-stain (*onlyH*) **E**) Only eosin-stain (*onlyE*)

epochs in total where the initial learning-rate is set to $2e^{-4}$ and then decreases to zero after every 30 epochs. The regularization factor λ is set to 10 for all experiments. Adam optimizer is used ($\beta_1 = 0.5, \beta_2 = 0.999$) with a batch size of 1. We train and evaluate the models on an NVIDIA Quadro P6000 graphics card.

ResNet

A ResNet [19] - in its Pytorch-implementation - is trained on the original Camelyon16 data set and its normalized versions to the HEV and TL data sets (using CycleGAN), to classify images into the two classes normal lymph node and carcinoma infiltration. Each network is trained for 100 epochs in total with a learning-rate of $2e^{-4}$.

Evaluation metrics

To assess the quality of the images generated by the CycleGAN we use the Fréchet Inception Distance (FID), to compare the distribution of generated images with the distribution of real images. In addition, the successful training of both generators is assessed by computing the Structural Similarity Index Measure (SSIM) in various windows between a real image and its reconstruction through the generator. For the classification task, we calculate kappa-values from the confusion matrix.

FID

This metric consists of the Fréchet distance also known as Wasserstein-2 distance computed on the basis of feature vectors. Here, a feature vector is the 2048-sized output of a pre-trained inception v3 model applied on one image. For the whole set of input images we get a sample of feature vectors with m_1 as its collective mean and C_1 as its covariance while for the GAN output images we get m_2, C_2 respectively [20]. The Fréchet distance is then applied to calculate the minimum distance between the means and covariances [21]:

$$d^2\ ((m_1, C_1), (m_2, C_2)) = ||m_1 - m_2||^2 + Tr\left(C_1 + C_2 - 2\sqrt{C_1 * C_2}\right).$$

For identical image sets the FID is zero, whereas it increases with noise and disturbances. The python code for computing FID scores is used from [22].

SSIM

For a given original image x and the corresponding output of the GAN y the features luminance $l(x,y)$, contrast $c(x,y)$ and structure $s(x,y)$ are compared on basis of the respective average, variance and covariance. The product of these components with the weighting factors α, β, γ yields the SSIM:

$$SSIM(\mathbf{x},\mathbf{y}) = \left[l(\mathbf{x},\mathbf{y})\right]^{\alpha} \cdot \left[c(\mathbf{x},\mathbf{y})\right]^{\beta} \cdot \left[s(\mathbf{x},\mathbf{y})\right]^{\gamma}.$$

The SSIM metric is applied on image pairs showing the same image content. It ranges from 0 to 1 and equals one only for exact identical images. An SSIM close to zero hardly represents similar images [23]. We use the python code from [24] to compute SSIM scores.

Cohen's kappa

The kappa-values are calculated from the confusion matrix using scikit-learn's function cohen_kappa_score [25]. The kappa score ranges from -1 and 1. A value $\geq$ 0.8 is considered as almost perfect agreement while ≤ 0 means no agreement (i.e. random labels) [26].

Results

We show the visual and evaluation results of our CycleGAN experiments on the Mitos-Atypia-14 and HEV data sets. For generators G_A and G_B, image tiles from image domains A and B can be normalized in both directions such that $X_A^{real} \xrightarrow{G_B} X_B^{fake} \xrightarrow{G_A} X_A^{rec}$ and $X_B^{real} \xrightarrow{G_A} X_A^{fake} \xrightarrow{G_B} X_B^{rec}$. In addition, we show the results of our experiments on the lymph node classification task with HE-stain normalization.

CycleGAN visual results

Mitos-Atypia-14

Example results of on the Mitos-Atypia-14 data set are shown in Fig. 3. Columns **A**-**C** refer to the image tiles scanned by the Aperio scanner (X_A^{real}) being mapped by the generator G_B to produce the corresponding image in the domain of the Hamamatsu scanner (X_B^{fake}) and the reconstruction from mapping the image back to its original domain (X_A^{rec}). The same process is done in the reverse direction for image tiles scanned in domain B being mapped to domain A and backward (columns **D**-**F**). Each row **1**-**4** presents another example image.

HE-Staining variation

Figure 4 presents several test results when mapping a standard stained HE-image X_A^{real} to one of the four stains of domain X_B^{fake}. Each block **A**-**D** shows another example tissue section. The top row of each block represents an exemplary image tile of the stain to be mapped into (*shortHE, longHE, onlyH, onlyE*), while the bottom row depicts the input image (*HE*) and the corresponding output for each stain.

CycleGAN evaluation

For all five experiments FID scores are shown in Fig. 5 **A**. As reference, FID scores of all testing images from data sets A and B (blue) are computed. They range between 31.5 (*MA14*) to 203.68 (*onlyE*). Our experiments achieve on average FID scores for real vs. fake of 7.09 (A) and 6.93 (B), while for real vs. rec we obtain an average of 5.76 (A)

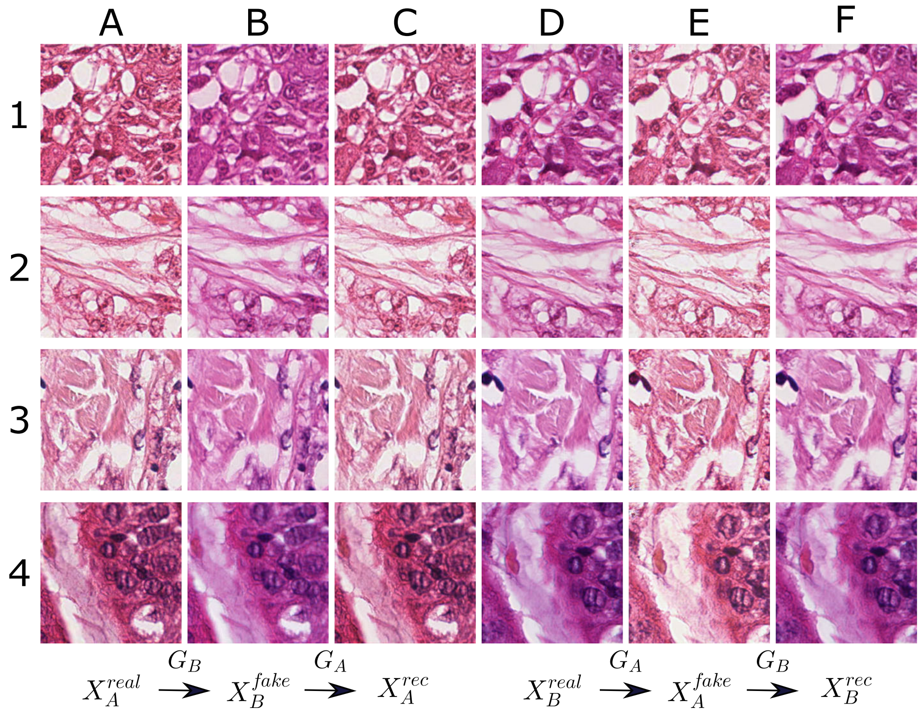
A
B
C
D
E
F
1
2
3
4
$X_A^{real} \xrightarrow{G_B} X_B^{fake} \xrightarrow{G_A} X_A^{rec}$
$X_B^{real} \xrightarrow{G_A} X_A^{fake} \xrightarrow{G_B} X_B^{rec}$

Fig. 3 Results gallery from our experiments on the Mitos-Atypia-14 challenge data set. Columns **A-C** refer to the image tiles scanned by the Aperio scanner (X_A^{real}) being mapped by the generator G_B to produce the corresponding image in the domain of the Hamamatsu scanner (X_B^{fake}) and the reconstruction from mapping the image back to its source domain (X_A^{rec}), i.e $X_A^{real} \xrightarrow{G_B} X_B^{fake} \xrightarrow{G_A} X_A^{rec}$. The same process is done in the reverse direction for image tiles scanned in domain *B*, i.e $X_B^{real} \xrightarrow{G_A} X_A^{fake} \xrightarrow{G_B} X_B^{rec}$ (column **D-F**). Each row **1-4** presents another example tissue section

and 5.58 (*B*). When mapping images from a source domain to a target domain, the FID scores compared to original images from the target domain improve up to 96% (blue vs. orange, red, green and purple). More precisely, for each experiment it is 76.85% (*MA14*), 91.93% (*shortHE*), 89.23% (*longHE*), 95.76% (*onlyH*), 95.57% (*onlyE*). A table with all FID is presented in the Appendix.

In addition, SSIM scores (see Fig. 5 **B**) are computed between the real and their reconstructed images for each image domain *A* (blue) and *B* (orange). Each value refers to

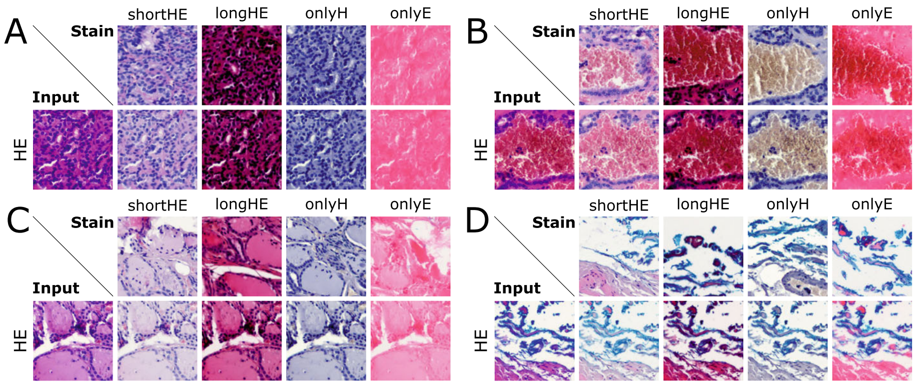

Fig. 4 Results gallery from our experiments on the HEV data set for the mapping $G_B : X_A^{real} \rightarrow X_B^{fake}$. Here, the input image is from domain *A* of the standard stained tissue (*HE*) being mapped to domain *B* corresponding to the image-sets *shortHE*, *longHE*, *onlyH*, *onlyE*. Each block **A-D** shows another example tissue section. The top row of each block represents an exemplary image tile of the stain to be mapped into, while the bottom row depicts the input image and the corresponding output for each stain

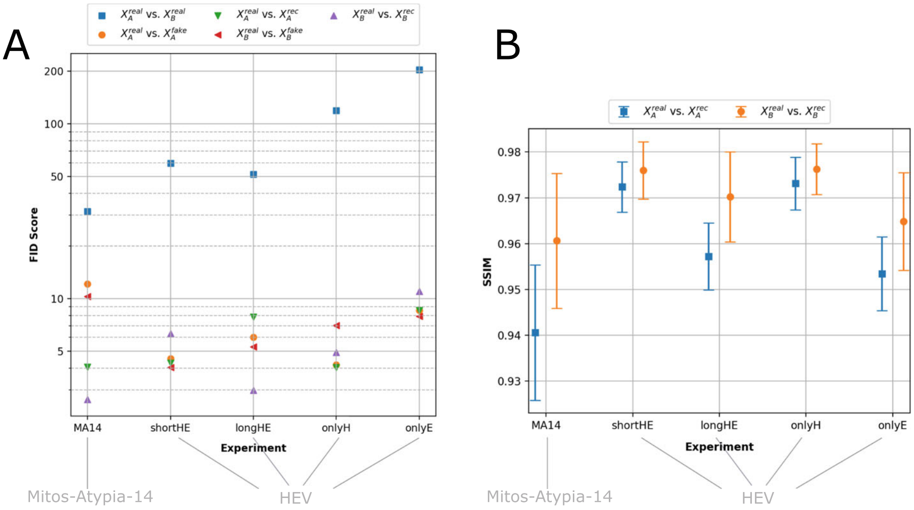

Fig. 5 Evaluation of our experiments using FID and SSIM scores.
A) FID scores between real and generated (fake, rec) images. For identical images the FID is zero, whereas it increases with noise and disturbances.
B) SSIM scores between real vs. rec images. The SSIM scale ranges from 0 to 1 and is close to zero for hardly similar images. A table with all FID and SSIM scores is presented in the Appendix

the average SSIM for all test images and the bars represent the corresponding standard deviation (SD). For each set A we obtain SSIM scores in the range of $SSIM = 0.94$ ($SD = 0.02$) ($MA14$) and $SSIM = 0.97$ ($SD = 0.01$) ($onlyH$), whereas for set B we obtain scores between $SSIM = 0.96$ ($SD = 0.02$) ($MA14$) and $SSIM = 0.98$ ($SD = 0.01$) ($onlyH$). A table with all SSIM scores is presented in the Appendix.

Stain normalization in image classification

In order to sow the impact of stain normalization on a downstream task such as image classification or segmentation, a ResNet [19] for the Camelyon16 data set, and its normalized versions, was trained. It should be noted that a standard color normalization to the mean and standard deviation as well as data augmentation with the albumentations package [27] is carried out in the standard training setting. The model learns to classify the images in lymph node parts with (hereinafter called tumor) and without (hereinafter called normal) carcinoma infiltration. Our three ResNets are tested on the Camelyon16 data set ($n = 830$ images without and $n = 898$ images with tumor) in original, normalized to the HEV data set and normalized to the TL data set as well as to the TL data set ($n = 902$ images without and $n = 900$ images with tumor) in original, normalized to the HEV data set and normalized to the Camelyon16 data set. As shown in Table 1 all ResNet models work best with images of the same HE-staining from the same training data set (Camelyon16 original, normalized to the HEV or TL data set). Furthermore, the models can best be applied to or transferred to the images from the TL data set, which in turn have the same HE-staining or stain normalization. However, with a best kappa-values of 0.55, this transfer from the Camelyon16 data set to the TumorLymphnode data set does not work very well. Also the ResNet-model trained on hematoxylin normalized images is able to classify other hematoxylin images correctly (kappa-value

Table 1 Kappa-values for the ResNet models that were trained on different versions of the Camelyon16 data set

model / data	*Cam_ori*	*Cam_HE*	*Cam_TL*	*TL_ori*	*TL_HE*	*TL_Cam*
ResNet_ori	**0.85**	0.00	0.12	0.00	0.00	**0.54**
ResNet_HE	0.02	**0.79**	0.28	0.13	**0.55**	0.01
ResNet_TL	0.36	0.10	**0.79**	**0.55**	0.08	0.24

The training images (from the Camelyon16 data set) are 1) original (Cam_ori), 2) normalized by the CycleGAN to the HEV data set (Cam_HE) or 3) the TL data set (Cam_TL), respectively. For each training sets, a ResNet model was trained: 1) ResNet_ori, 2) ResNet_HE and 3) ResNet_TL. All models were tested on images from the Camelyon16 data set ($n = 1728$ images) and the TL data set ($n = 1802$ images). There were again three versions of both test data sets: one original version (Cam_ori and TL_ori), one version normalized to the HEV data set (Cam_HE and TL_HE), and one version normalized to the Camelyon16 (TL_Cam) or the TL data set (Cam_TL). The best kappa value obtained for each test set (column-wise) on all models is shown in bold

0.77 for the Camelyon16 data set and 0.54 for the TL data set). The complete table of the results is shown in the Appendix.

Discussion

This paper underlines that CycleGANs for image normalization in the context of digital pathology [12] yield a clear clinical benefit. The trained CycleGAN models show compelling results, both visually (Figs. 3 and 4) and quantitatively (Fig. 5) by obtaining FID scores up to 96% better for images mapped to a target domain. The trained models are able to fully convert to the desired color scheme while preserving the structural contents of the original image due to the cycle consistency constraint leading to SSIM scores greater than 0.9 when mapping generated images back to their source domain.

Furthermore, we could confirm that this normalization approach can be beneficial for image classification tasks. Similar to the results of Shaban et al. [10] for the Camelyon16 data set, Yuan et al. [28] for a number of publicly available data sets or de Bel et al. [12] in the application of image segmentation. Compared to earlier approaches on public data sets, in our work we demonstrate how stain normalization improves the classification accuracy, when applied to multiple internal data on a pretrained model. When normalizing HE-stain images from our institute to the Camelyon16 data set and testing a ResNet-model that is only trained on Camelyon16 data the kappa-value increased more than 50%. Furthermore, we were able to show in a first approach that such a normalization could also enable the use of different colored data sets for downstream tasks, e.g. by converting both data sets to a different staining. However, further work, e.g using more different staining methods, is certainly necessary here.

Some limitations of the used CycleGAN-model can also be seen when mapping images obtained by different scanning devices with varying resolutions. This can cause a loss in structural information despite the consistently good quality of the color normalization. With the HEV data set, the generated images look very realistic compared to the original images in a target domain without any decline in the image content (see Fig. 4).

The CycleGAN approach used here is able to successfully learn every mapping between two image stains and can instantly normalize any unseen image if it is within one of the trained stains. For each other staining, the network needs to be retrained from scratch. However, the network is able to learn even from a small amount of images (1,000-10,000 per set) which can be obtained from a single WSI. In addition, the images do not have to be labeled or paired to learn the mapping between two domains. The network can learn to add a stain to images which is not present in the source domain, e.g we are able to create a full HE-stained image from an image which has only a single stain (*onlyH*, *onlyE*) or vice versa. This can simplify the manual staining process. How this affects other stains besides HE needs further investigation.

Conclusion

In this paper we show that CycleGANs are a powerful tool for normalization of different variants of HE-stains and tissue types. We validated this approach on data sets covering images from different scanning devices, staining protocols and tissue types and additionally showed the impact on stain normalization on the downstream task of image classification. The method has been successfully applied to compensate for variances resulting from image acquisition as well as from tissue staining while preserving structural content of the images. In order to make use of this approach in a clinical manner, the training process should be accelerated, i.e. using transfer learning, an increased batch size and specialized hardware. The method may be added to various image processing frameworks at WSI level to be applied to tasks such as classification or segmentation.

Appendix

Table 2 Overview of our stain normalization experiments

Data set	Experiment Name	Set *A*	Set *B*
Mistos-Atypia-14	*MA14*	Aperio scanner	Hamamatsu scanner
HEV	*shortHE*	standard HE stained	shortened staining time
	longHE		prolonged staining time
	onlyH		only stained with hematoxylin
	onlyE		only stained with eosin
Camelyon16	*Cam_HE*	Camelyon16	standard HE stained
	Cam_TL		TumorLymphnode
TumorLymphnode	*TL_HE*	TumorLymphnode	standard HE stained
	TL_Cam		Camelyon16

Table 3 FID scores for all experiments between real and generated (fake, rec) images for *A* and *B*

FID	*MA14*	*shortHE*	*longHE*	*onlyH*	*onlyE*
X_A^{real} vs. X_B^{real}	31.5017	59.4240	51.4460	119.0061	203.6761
X_A^{real} vs. X_A^{fake}	12.1464	4.5465	6.0007	4.1793	8.5647
X_A^{real} vs. X_A^{rec}	4.0544	4.2877	7.8630	4.0363	8.5685
X_B^{real} vs. X_B^{fake}	10.3222	4.0365	5.3136	7.0321	7.9218
X_B^{real} vs. X_B^{rec}	2.6451	6.3173	2.9931	4.9206	11.0160

Table 4 SSIM scores (SD = standard deviation) for all experiments between real and rec images for *A* and *B*

SSIM (SD)	*MA14*	*shortHE*	*longHE*	*onlyH*	*onlyE*
X_A^{real} vs. X_A^{rec}	0.9406 (0.0147)	0.9724 (0.0055)	0.9572 (0.0073)	0.9731 (0.0057)	0.9534 (0.0080)
X_B^{real} vs. X_B^{rec}	0.9606 (0.0148)	0.9760 (0.0063)	0.9702 (0.0098)	0.9763 (0.0056)	0.9648 (0.0107)

Table 5 Kappa-values for the ResNet models that were trained on different versions of the Camelyon16 data set

model / data	*Cam_ori*	*Cam_HE*	*Cam_onlyH*	*Cam_TL*	*TL_ori*	*TL_onlyH*	*TL_HE*	*TL_Cam*
ResNet_ori	**0.85**	0.00	0.00	0.12	0.00	0.00	0.00	**0.54**
ResNet_HE	0.02	**0.79**	0.01	0.28	0.13	0.00	**0.55**	0.01
ResNet_onlyH	0.00	0.00	**0.77**	0.14	0.03	**0.53**	0.00	0.00
ResNet_TL	0.36	0.10	0.53	**0.79**	**0.55**	0.38	0.08	0.24

The best kappa value obtained for each test set (column-wise) on all models is shown in bold

Abbreviations

HE: Hematoxylin-eosin; GAN: Generative adversarial network; FID: Fréchet inception distance; SSIM: Structural similarity index measure

Acknowledgements

We thank our project partners Smart In Media AG for the collaboration in this project.

Authors' contributions

M. Runz: study design and coordination, data formation, model training, statistical evaluation, manuscript drafting. D. Rusche: statistical evaluation, manuscript drafting. S. Schmidt: manuscript drafting. M. R. Weihrauch: manuscript drafting. J. Hesser: study design and coordination, manuscript drafting. C.A. Weis: study design and coordination, data formation, model training, statistical evaluation, manuscript drafting. The author(s) read and approved the final manuscript.

Funding

Our work is supported by the grant ZIM ZF4689501TS9. Open Access funding enabled and organized by Projekt DEAL.

Availability of data and materials

The code is publicly available at https://github.com/m4ln/stainTransfer_CycleGAN_pytorch. The data set supporting the solutions is available at https://doi.org/10.11588/data/8LKEZF.

Declarations

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Institute of Pathology, University Medical Centre Mannheim, Heidelberg University, Mannheim, Germany. [2]Mannheim Institute for Intelligent Systems in Medicine (MIISM), Medical Faculty Mannheim, Heidelberg University, Mannheim, Germany. [3]Center for Mass Spectrometry and Optical Spectroscopy (CeMOS), Mannheim University of Applied Sciences, Mannheim, Germany. [4]Smart In Media AG, Köln, Germany. [5]Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Heidelberg, Germany. [6]Central Institute for Computer Engineering (ZITI), Heidelberg University, Heidelberg, Germany.

Received: 24 February 2021 Accepted: 5 July 2021
Published online: 06 August 2021

References

1. Bianconi F, Kather JN, Reyes-Aldasoro CC. Experimental assessment of color deconvolution and color normalization for automated classification of histology images stained with hematoxylin and eosin. Cancers. 2020;12(11):. https://doi.org/10.3390/cancers12113337.
2. Bukenya F. A hybrid approach for stain normalisation in digital histopathological images. Multimed Tools Appl. 2020;79(3):2339–2362. https://doi.org/10.1007/s11042-019-08262-0.
3. Vicory J, Couture HD, Thomas NE, Borland D, Marron JS, Woosley J, Niethammer M. Appearance normalization of histology slides. Comput Med Imaging Graph. 2015;43:89–98. https://doi.org/10.1016/j.compmedimag.2015.03.005.
4. Khan AM, Rajpoot N, Treanor D, Magee D. A nonlinear mapping approach to stain normalization in digital histopathology images using image-specific color deconvolution. IEEE Trans Biomed Eng. 2014;61(6): 1729–1738. https://doi.org/10.1109/TBME.2014.2303294.
5. Macenko M, Niethammer M, Marron JS, Borland D, Woosley JT, Guan X, Schmitt C, Thomas NE. A method for normalizing histology slides for quantitative analysis. In: 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro; 2009. p. 1107–1110. https://doi.org/10.1109/ISBI.2009.5193250.
6. Bautista PA, Yagi Y. Staining correction in digital pathology by utilizing a dye amount table. J Digit Imaging. 2015;28(3):283–294. https://doi.org/10.1007/s10278-014-9766-0.
7. Ruifrok A, Johnston D. Quantification of histochemical staining by color deconvolution. Anal Quant Cytol Histol. 2001;23(4):291–9.
8. Reinhard E, Adhikhmin M, Gooch B, Shirley P. Color transfer between images. IEEE Comput Graph Appl. 2001;21(5):34–41. https://doi.org/10.1109/38.946629.
9. Ghazvinian Zanjani F, Zinger S, Ehteshami Bejnordi B, van der Laak J, With P. Stain normalization of histopathology images using generative adversarial networks; 2018. p. 573–577. https://doi.org/10.1109/ISBI.2018.8363641.
10. Shaban MT, Baur C, Navab N, Albarqouni S. StainGAN: stain style transfer for digital histological images. 2018. 1804.01601. Accessed 09 Dec 2020.
11. Swiderska-Chadaj Z, de Bel T, Blanchet L, Baidoshvili A, Vossen D, van der Laak J, Litjens G. Impact of rescanning and normalization on convolutional neural network performance in multi-center, whole-slide classification of prostate cancer. Sci Rep. 2020;10(1):14398. https://doi.org/10.1038/s41598-020-71420-0.
12. de Bel T, Hermsen M, Kers J, van der Laak J, Litjens G. Stain-transforming cycle-consistent generative adversarial networks for improved segmentation of renal histopathology. In: Cardoso MJ, Feragen A, Glocker B, Konukoglu E, Oguz I, Unal G, Vercauteren T, editors. Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning. Proceedings of Machine Learning Research, vol. 102. London: PMLR; 2019. p. 151–163. http://proceedings.mlr.press/v102/de-bel19a.html.
13. Mahapatra D, Bozorgtabar B, Thiran J-P, Shao L. Structure Preserving Stain Normalization of Histopathology Images Using Self-Supervised Semantic Guidance. 2020. 2008.02101. Accessed 09 Dec 2020.
14. Zhu J-Y, Park T, Isola P, Efros AA. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. 2020. 1703.10593. Accessed 09 Dec 2020.
15. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative Adversarial Networks. 2014. 1406.2661. Accessed 09 Dec 2020.
16. Ehteshami Bejnordi B, Veta M, Johannes van Diest P, van Ginneken B, Karssemeijer N, Litjens G, van der Laak JAWM, the CAMELYON16 Consortium. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. JAMA. 2017;318(22):2199–2210. https://doi.org/10.1001/jama.2017.14585. https://jamanetwork.com/journals/jama/articlepdf/2665774/jama_ehteshami_bejnordi_2017_oi_170113.pdf.
17. MITOS-ATYPIA-14 Grand Challenge. https://mitos-atypia-14.grand-challenge.org/. Accessed 22 Mar 2020.
18. Runz M, Weis C-A. Normalization of HE-Stained Histological Images using Cycle Consistent Generative Adversarial Networks [Dataset]. heiDATA. 2021. https://doi.org/10.11588/data/8LKEZF.
19. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. 2015. 1512.03385. Accessed 11 May 2021.
20. Brownlee J. How to implement the frechet inception distance (fid) for evaluating gans. 2019. https://machinelearningmastery.com/how-to-implement-the-frechet-inception-distance-fid-from-scratch/. Accessed 15 Jan 2021.
21. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. 2018. 1706.08500. Accessed 15 Jan 2021.
22. w13b3. SSIM-py Structural Similarity (SSIM) index, where the core dependency is NumPy. 2019. https://github.com/w13b3/SSIM-py. Accessed 20 Dec 2020.
23. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process. 2004;13(4):600–612. https://doi.org/10.1109/TIP.2003.819861.
24. Seitzer M. pytorch-fid: FID Score for PyTorch. 2020. https://github.com/mseitzer/pytorch-fid. Version 0.1.1. . Accessed 18 May 2021.
25. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E. Scikit-learn: Machine learning in Python. J Mach Learn Res. 2011;12:2825–2830.
26. McHugh ML. Interrater reliability: the kappa statistic. Biochemia Medica. 2012;22(3):276–282. 23092060[pmid].
27. Buslaev A, Iglovikov VI, Khvedchenya E, Parinov A, Druzhinin M, Kalinin AA. Albumentations: Fast and flexible image augmentations. Information. 2020;11(2):. https://doi.org/10.3390/info11020125.
28. Yuan E, Suh J. Neural stain normalization and unsupervised classification of cell nuclei in histopathological breast cancer images. arXiv preprint arXiv:1811.03815. 2018.

Publisher's Note