Department of Physics and Astronomy
University of Heidelberg

Master Thesis in Physics
submitted by

Marco Hübner

born in Schorndorf (Germany)

2021

Deep Learning-Based Synthesis of Surgical Hyperspectral Images

This Master Thesis has been carried out by Marco Hübner
at the
German Cancer Research Center (DKFZ) in Heidelberg

under the supervision of

Prof. Dr. Carsten Rother
and
Prof. Dr. Lena Maier-Hein

Abstract

Postoperative death within 30 days after surgical intervention is the third largest contributor to mortality globally. Causes of postoperative mortality are manifold but also comprise challenging perception and the inability to estimate physiological tissue parameters during interventions. To capture data emanating from underlying physiological tissue properties, hyperspectral imaging (HSI) together with machine learning-based analyses has been proposed as a solution in recent literature. However, HSI data in the clinical setting is sparse, as its acquisition is crucially limited by a small number of approved devices and the need for clinical trials. Therefore, the present work investigates common deep learning frameworks for HSI and proposes a two-step image generation pipeline to synthesize hyperspectral tissue images. To validate the image generation pipeline, spectral correctness and textural realism were assessed both qualitatively and quantitatively. Results of the textural Kernel Inception Distance (KID) exhibited state of the art (SOTA) performance for both paired and random generated HSI patches. Furthermore, the feasibility of using the synthetic, unlabelled data for an image segmentation task was tested and found to not lead to improvement. From the conducted experiments it can be concluded that RGB image synthesis can be adapted to the HSI domain, while synthetic additional data has to be tailored for individual tasks.

Überblick

Postoperative Todesfälle bis zu 30 Tage nach chirurgischen Eingriffen sind weltweit die dritthäufigste Todesursache. Die Gründe für postoperative Sterblichkeit sind vielfältig, aber umfassen auch schwierige Sichtverhältnisse und das Unvermögen, physiologische Gewebeparameter während der Intervention bestimmen zu können. Für die Gewinnung solcher Gewebedaten, welche aus den physiologischen Gewebeeigenschaften resultieren, wurde in der Literatur hyperspektrale Bildgebung (HSI) zusammen mit auf Machine Learning basierenden Analysen als Lösungsansatz vorgestellt. Allerdings sind HSI Bilder im medizinischen Bereich rar, da nur wenige zugelassene Aufnahmegeräte zur Verfügung stehen und klinische Studien erforderlich sind. Deshalb untersucht die vorliegende Arbeit Deep Learning Ansätze für HSI und stellt eine Bildgenerierungspipeline mit zwei Schritten für hyperspektrale Gewebeaufnahmen vor. Um die Bildgenerierungspipeline zu überprüfen wurde hierfür die Korrektheit von Spektren und Texturen wurde qualitativ und quantitativ bewertet. Ergebnisse der Textur-messenden KID zeigten sowohl für gepaarte, wie auch für zufällige synthetische Aufnahmen SOTA Resultate. Darüber hinaus wurde die Nutzbarkeit der generierten, nicht annotierten HSI Daten für Bildsegmentierung untersucht, welche zu keiner Verbesserung führten. Aus den durchgeführten Experimenten wird geschlossen, dass RGB Bildgenerierung für HSI Daten adaptiert werden kann, sowie dass künstliche Bilddaten auf die individuelle Aufgabe zugeschnitten werden müssen.

Table of Contents

Page

List of Abbreviations

BRISQUE	Blind/ Referenceless Image Spatial Quality Evaluator
CE	Conformité Européenne
CT	Computed Tomography
DCGAN	Deep Convolutional GAN
DISTS	Deep Image Structure and Texture Similarity
EMD	Earth Mover's (first Wasserstein) Distance
FID	Fréchet Inception Distance
GAN	Generative Adverarial Network
LR-GAN	Latent Regressor GAN
HSI	Hyperspectral Imaging
IQA	Image Quality Assessment
IS	Inception Score
INN	Invertible Neural Network
KID	Kernel Inception Distance
KL	Kullback-Leibler (- Divergence)
MI	Mutual Information
MMD	Maximum Mean Discrepancy
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
MSI	Multispectral Imaging
MUNIT	Multimodal Unsupervised Image-to-Image Translation
NIR	Near Infrared
OT	Optimal Transport
PCA	Principal Component Analysis
PET	Positron Emission Tomography
PSNR	Peak Signal to Noise Ratio
ReLU	Rectified Linear Unit (Activation Function)
RGB	Red-Green-Blue (Colourspace)
SOTA	State of the Art
SSIM	Structural Similarity
UMAP	Uniform Manifold Approximation and Projection
VAE	Variational Autoencoder
VGG	Visual Geometry Group
WAE	Wasserstein Autoencoder

List of Figures

Page

Acknowledgements

First of all, I would like to thank Prof. Dr. Lena Maier-Hein, who accepted me into her research team and always helped out with further ideas, hints and revisions, which shaped this work.

I would also like to thank Prof. Dr. Carsten Rother, who supervised this work from the side of the faculty of physics and astronomy.

This work further would not have been possible without my direct supervisor, Leonardo Ayala, who guided me through my time at the DKFZ with feedback, suggestions as well as lots of meetings with valuable discussions. The same also applies for Jan Sellner, who largely contributed to evaluations of the downstream task as author of a hyperspectral image segmentation network [4]. He also helped me with CPU-efficient implementations and again many constructive discussions.

I would further like to acknowledge Dr. Alexander Seitel, Dr. Minu Tizabi and Patrick Godau, who proof-read parts of this work and thus largely contributed to its final form.

Lastly, I want to thank Mascha Sokolova for additional corrections and her patience during the last year(s) as well as my parents for their unlimited support throughout my entire studies.

Introduction

Motivation

Postoperative death within 30 days after surgical intervention is with 4,2 million fatalities annually the third largest cause for global death, with half of the casualties occurring in low- and middle-income countries [5]. Contributors to postoperative mortality include among others challenging intraoperative perception and the inability to estimate physiological tissue parameters with the human eye, since both human vision and hence standard (laparoscopic) imaging modalities operate with broad spectral responses in the red, blue and green wavelength region [6]. This inaccurate spectral recognition leads to a loss of detailed information, including that on characteristic chromophores such as hemoglobin, melanin [7, 8] or bilirubin [9]. Multispectral imaging (MSI) with 10s of wavelengths or even hyperspectral imaging (HSI) with 100s of wavelengths [1] can provide accurate spectral measurements of scattering-induced tissue reflectance spectra [8, 10] and hence allow to obtain information on physiological parameters obtainable as well as make informed classification decisions.

Since the acquired data surpasses the limitations of the human eye, it can not directly be evaluated or interpreted by medical personnel and hence requires additional information extraction steps. Therefore, the medical imaging community has proposed myriad machine and deep learning approaches which transform HSI spectral characteristics into applicable knowledge, with the potential to support decision-making during minimally invasive surgeries. Examples of such machine learning applications are cancerous tissue detection [11, 12, 13], better image segmentation [4] or real-time physiological parameter estimation [14, 3]. The latter would allow for non-invasive instead of fluorescence marker-based[15] perfusion estimation, while hyperspectral data generally facilitates classification and segmentation decisions [4].

However, the introduced machine learning methods require HSI data for training, which is sparse [3, 16]. Among the several reasons for this are the necessity for clinical trials, supervision of domain experts and strict privacy regulations which often hinder data distribution across e.g. different research facilities [17]. Furthermore, only few HSI devices are approved or available for clinical usage and thus the imaging modality itself is rare.

Motivated by recent successes of RGB surgical image synthesis [2, 16, 18], this work investigates surgical spectral image synthesis. Key points are the exploration of hyperspectral data synthesis, for which we are not aware of any prior work, with particular focus on overcoming data sparsity and privacy concerns. The primary objective of the thesis was to investigate the following hypothesis:

Hypothesis

Deep learning can enable realistic hyperspectral image synthesis.

Research Questions

This work extends already existing deep-learning image synthesis frameworks to the previously rarely investigated hyperspectral image deep learning domain. Therefore, the objectives of this thesis encompass beyond qualitative and image quality metric assessment of the generated data also spectral evaluation. Furthermore, the application of generated HSI data was tested on a downstream task. The main research questions investigated thus are:

Can the proposed deep learning pipeline generate hyperspectral image patches that...

... look realistic in terms of imaging effects like specular highlights or shadows and physiological attributes such as blood vessels?

... feature pixel spectra similar to those extracted from real data?

... generalize beyond the training data?

... feature realistic textures?

... improve a downstream organ segmentation task?

Outline

This work begins with an introduction of *Principles and Theoretical Background* to explain underlying physics of tissue reflectance spectra and, later, applied deep learning methods. *Related Work* presents works of literature which this thesis builds on, especially with regards to methods which are detailed in the subsequent *Materials and Methods* section. Conducted experiments and their results are presented in the *Experiments and Results* section, while the *Discussion and Conclusion* afterwards assesses outcomes and gives an outlook on future work. Lastly, the *Appendix* contains additional results.

Part I.
Principles and Theoretical Background

This chapter presents background behind *Radiative Transport in Tissue* to motivate image synthesis by means of later presented *Deep Learning* methods and their building blocks. For a more detailed introduction of biomedical optics, please consult e.g. Wang et al. [8]. Principles of the used *Spectral Imaging Devices* are introduced to illustrate their incorporation in clinical settings. Overviews of spectral imaging in the surgical domain, with particular focus on imaging hardware and techniques, can be found in Lu et al. [19] and Clancy et al. [1].

1. Radiative Transport in Tissue

HSI serves the purpose of obtaining improved qualitative and quantitative knowledge of physiological parameters without additional usage of biomarkers, in a non-invasive manner. To gain access to this tissue content information, HSI records light reflected by the tissue. For conversion of the acquired information into applicable knowledge, it is therefore important to understand how and why tissue responds in a spectrally specific way to light.
The main properties of light propagation in and out of tissue such as scattering, absorption and beam divergence are hence discussed in the upcoming paragraphs.

1.1. Tissue Scattering

Scattering effects dominate the reflectance spectrum of tissue in an optical and NIR 'window', which is displayed in Figure 1. This window ranges from 600 - 1000 nm [9, 20] or 400 - 1350 nm according to other sources [8] and depends on specific tissue scattering parameters. For HSI relevant optical and near-infrared wavelength range, cellular nuclei and mitochondria are besides melanin the main scatterers of (human) tissue. This property results from their size, which is similar to the wavelength of the incident light, as well as their refractive index, which is slightly higher than the refractive index of the embedding cytoplasm [8]. The resulting free photon path in tissue is therefore approximately 0.1 mm [8]. Melanin is mostly neglected in this discussion, since this work is mainly concerned with internal organs where the melanin content is much lower than for (human) skin.

When looking at possible scattering events inside of a tissue volume element dV, two kinds of events are possible: *into-* and *out-of-beam-scattering*. *into-beam-scattering* refers to light with incoming unit direction $\hat{s}'$, which leaves the tissue element in 'beam-axis' direction $\hat{s}$. Opposite, a light ray with incoming direction along the 'beam-axis $\hat{s}$ leaving in direction $\hat{s}' \nparallel \hat{s}$ is referred to as *out-of-beam-scattering*. Following Wang et al. [8], these contributions are treated separately, as the *into-beam-scattering* requires the consideration of the scattering phase function $p(\hat{s}', \hat{s})$, which often is only depending on the angle θ between incoming and outgoing ray.

The energy flux per unit time dP into an infinitesimal solid angle element $d\Omega$ can be calculated with phase function p, incoming light direction $\hat{s}' \in \Omega'$ and outgoing light direction $\hat{s}$. To do so, the number of scatterers N has to be multiplied with the individual scatterers cross section σ_s

to obtain an overall cross-section. Second, the energy influx per area and unit time needs to be calculated. Therefore, the radiance L is integrated over all possible incoming infinitesimal solid angle elements $d\Omega'$, where the phase function p as probability density function for direction combinations accounts for the possibility. Multiplying both terms, this becomes

$$dP_{sca} = \underbrace{(n_s dV)}_{\#\text{ scatterers } N} \; \sigma_s \underbrace{\left(\int_{\Omega'} L(\vec{r},\hat{s}',t)p(\hat{s}',\hat{s})d\Omega'\right) d\Omega}_{\text{energy flux } W\cdot(m^2\cdot\text{sr})^{-1}} = \mu_s \left(\int_{\Omega'} L(\vec{r},\hat{s}',t)p(\hat{s}',\hat{s})d\Omega'\right) d\Omega \; dV. \quad (1)$$

Combining scatterer's density n_s and cross section σ_s in above's equation yields the scattering coefficient μ_s.

1.2. Absorption and Extinction

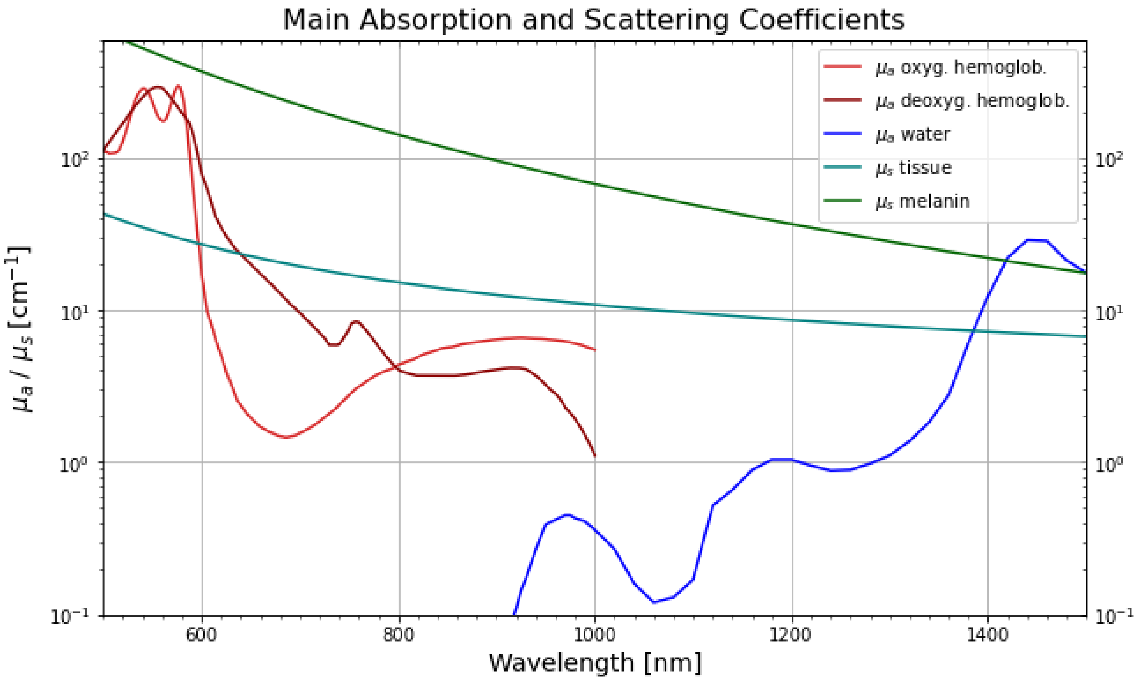


Figure 1: Contributors to the spectral response of tissue with scattering coefficient μ_s and absorption coefficients μ_a as function of the wavelength. Characteristic responses of oxygenated hemoglobin in the region of 500 - 600 nm and absorption band of deoxygenated hemoglobin at around 760 nm are clearly visible. Hemoglobin data for a concentration of 150 g/l [21], water absorption [22] as well as melanin scattering properties [23] are taken from literature. For generic tissue scattering μ_s *tissue*, a combination of Mie and Rayleigh-scattering is assumed [24]. Hemoglobin data from literature only covers wavelengths up to 1000 nm, while water absorption and scattering was plotted for up to 1500 nm to display the optical window.

The *out-of-beam-scattering* together with absorption makes up the so-called extinction. Exchanging incident light direction to $\hat{s}$ and outgoing direction to $\hat{s}'$ in equation Equation 1 allows to calculate *out-of-beam-scattering*, which returns a much easier equation. This is due to the fact, that the now $d\Omega'$ independent radiance L can be pulled out of the integral and only the phase function p remains to be integrated over all incoming ray directions. Since the phase function is a probability density, integrating it over the whole space returns a factor of 1. Together with the volume-element specific absorption this yields the extinction term

$$dP_{ext} = (\mu_a + \mu_s)L(\vec{r},\hat{s},t)d\Omega \; dV. \quad (2)$$

The absorption and scattering coefficients μ_a and μ_s sum up to the overall extinction coefficient, in Wang et al. [8] mentioned as $\mu_t = \mu_a + \mu_s$. From a biomedical point of view, this term of the radiative transport equation is the most important and interesting one, as the main absorbers shown in Figure 1 are oxygenated and deoxygenated hemoglobin, water [8] and also substances like bilirubin [9]. They hence reveal crucial physiological parameters such as tissue oxygenation, blood volume fraction and water content.

1.3. Sources, Beam Divergence and Radiative Transport Equation

To obtain the full radiative transport equation, the beam divergence term and the source term need to be incorporated. The source term is here introduced with a black-box notation S as

$$dP_{src} = S(\vec{r}, \hat{s}, t) d\Omega \; dV \tag{3}$$

for abstract discussion. In a concrete case of a calculation or simulation the source needs to be specified, which also would require defining spectral properties. These have so far not been discussed as this would lead to far, but are always implicitly contained in the radiance L, which already is integrated over all wavelengths.

The divergence of the beam is calculated by computing the divergence of the radiance along the rays' direction. This yields

$$dP_{div} = \nabla_{\hat{s}}(L(\vec{r}, \hat{s}, t)\hat{s}) d\Omega \; dV, \tag{4}$$

the last term of the radiative transport equation. Using overall energy conservation for the change in energy per unit time P in the volume element dV from solid angle $d\Omega$ results in

$$\begin{aligned} dP &= \frac{1}{c}\frac{\partial L(\vec{r}, \hat{s}, t)}{\partial t} d\Omega \; dV \\ &= dP_{sca} - dP_{div} - dP_{ext} + dP_{src}. \end{aligned} \tag{5}$$

Putting equations 1 - 4 into Equation 5 finally gives the full radiative transport equation:

$$\frac{1}{c}\frac{\partial L(\vec{r}, \hat{s}, t)}{\partial t} = \mu_s \int_{\Omega'} L(\vec{r}, \hat{s}', t) P(\hat{s}', \hat{s}) d\Omega' - (\hat{s}\nabla_{\hat{s}} + \mu_t) L(\vec{r}, \hat{s}, t) + S(\vec{r}, \hat{s}, t) \tag{6}$$

This equation in theory allows to calculate tissue reflectance spectra, emanating from underlying physiological and optical tissue properties as well as source specifications, when additionally taking into account boundary conditions on the air-tissue border. The complexity of the individual contributors of this integro-differential equation makes Equation 6 only in special cases analytically solvable and thus requires numerical solutions. Several Monte-Carlo approaches [14, 25] have thus been implemented to generate reliable reflectance spectra, as other numerical approximations and solutions of the radiative transport equation are less accurate [8].

Simulating many photons propagating through tissue, however, is computationally intensive. Especially due to the large absorption length of 10 - 100 mm compared to the mean free path of 0.1 mm many scattering events occur for every single photon [8]. This time-intensity makes Monte Carlo simulations for full images impractical and is the technical motivation for learning HSI tissue, as learned models would allow much faster gathering of synthetic data once they are trained.

2. Spectral Imaging Devices

HSI tissue data for machine learning-based analysis or, in this case, training of the deep learning models, is acquired with special cameras: MSI or HSI cameras capture 10s to 100s of wavelengths at a time [1] and their recordings thus contain much more detailed implicit information on scattering and absorption, which grants specific insights into biological and optical tissue parameters. The MSI or HSI devices used for said optical measurements can be separated into three classes, according to their image acquisition modes. In the following, these acquisition modes are discussed and where possible compared to their RGB counterparts in terms of construction for the integration into the clinical setting.

Spectral Scanning

Spectral scanning is the first presented acquisition technique and can be achieved in a myriad of ways. One of the most simplistic approaches is a mechanical filter wheel, dedicated to either only allow transmission of certain wavelengths to the imaging sensor or to adjust the illumination spectrum to the desired optical region, as shown in Figure 2. This allows to record one image per specified wavelength sector, which can be composed into a n-channel MSI or HSI data cube after a full imaging cycle. However, the downsides of this technique are slow recording times and mechanical vibrations besides their overall larger device size [19].

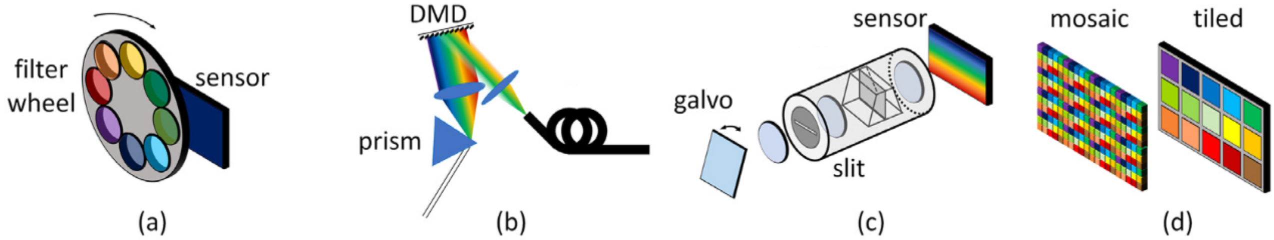


Figure 2: MSI/HSI devices: (a) and (b) use spectral filtering techniques for either the reflectance spectrum or the illuminating light source by means of a mechanical filter wheel, electro-optical band filtering or a digital micromirror device (DMD). (c) sketches the optical apparatus used for spatial pushbroom scanning while (d) shows snapshot specific filter grids, mounted directly onto the sensor. Adapted from [1], permitted by *Creative Commons Attribution 4.0 International* (CC BY 4.0) license.

Previous problems of the filter wheel can be overcome with electronically controlled bandpass filters. Tunable liquid crystal or acousto-optic filters allow for faster switching but come with their own disadvantages: The liquid crystal filters suffer from lower optical transmittance [1] whilst in the second case the resulting image quality is worsened due to the imperfect, nonlinear susceptibility of the acousto-optic filter, which causes undesired frequency mixing [26]. Besides the previously mentioned filtering approaches, also (linear) variable optical filters [27] and grating-like digital micromirror devices for flexible bandpass shaping can be used [28].

Lastly, advances of light-emitting diodes allow for a high-intensity, monochromatic and also rapidly switchable light source and therefore a filter-less setup. In clinical practice however, this is hindered by ambient light, which compromises the images' quality by contaminating wavelength-specific intensities as well as inefficient fibre-coupling, which largely reduces intensity [1].

RGB imaging devices also occasionally use monochromatic sensors for spectrally separated light; however, this is then done with help of beam splitting and three separate sensors [29], a concept which is unfeasible for a higher amount of optical bands. While the structure of the imaging modalities is hence in this case not similar, e.g. electronically tunable wavelength filters can be incorporated into laparoscopes, which yields compatibility of this MSI/ HSI acquisition technique with existing medical equipment.

Spatial Scanning

Spatial scanning uses the two-dimensional light sensor for recording of only singular spatial regions (whiskbroom) or lines (pushbroom) of a scene. Figure 2 depicts such a pushbroom setup which uses pre-attached optical components to disperse the light onto the full area of the sensor. The spectral information is then read out along the pixels of the second imaging sensor axis. Figure 2 also showcases an electromechanically turnable mirror which allows to capture the whole scene without having to move the camera for both whiskbroom and pushbroom devices. Scanning the scene with mirror galvanometers or other electro-mechanical mechanisms usually results in high spectral resolution while often lacking recording speed, which can cause issues *in-vivo*, if swift movement or pulsation is involved [1].

While there is no similar imaging technique for RGB recording, laparoscopic imaging devices are also compatible with this image acquisition technique. The optical fibre which transmits the light can easily guide the captured scene to such an optical apparatus and hence poses no additional burden in the clinical setting.

Snapshot Systems

Modern snapshot systems use a more complex version of the Bayer filter pattern [30]. The filter pattern adds optical wavelength filters in front of the sensor's pixels in an equally distributed manner and thus allows to record full images with several optical bands at the same time [1]. Yet this manufacturing approach comes with the disadvantage of trading spatial for spectral resolution and vice versa, since the amount of pixels on the sensor is limited and thus must be divided according to the number of desired spectral bands.

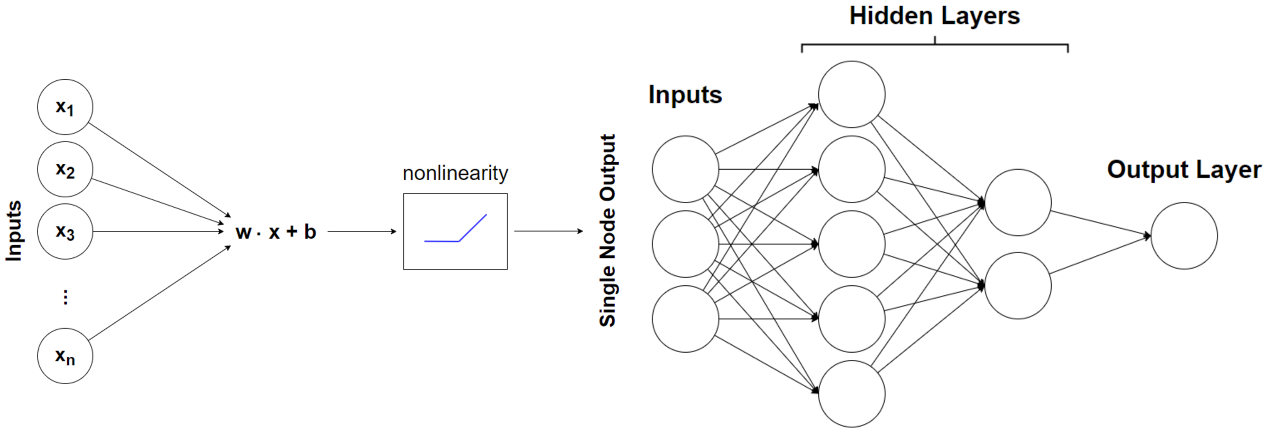
This acquisition technique via the Bayer filter pattern is standard for RGB imaging and can seemlessly be transferred into the medical setting, by only exchanging the imaging sensor.

3. Deep Learning

Image generation in this thesis builds on *deep learning* to overcome limitations of solutions to the radiative transport equation. Deep learning itself belongs under the umbrella of *machine learning* methods when following the classification of Goodfellow et al. [31]. Its consisting of simple but hierarchically stacked building blocks is characteristic and distinguishes it from most classical machine learning approaches. Automated pattern recognition and generation can be achieved via learning of filters [31] where the automation allows for the participation of non-domain experts [32]. The upcoming section introduces basic building blocks of neural networks and their fundamental optimization procedure.

3.1. Neural Network Basic Building Blocks

The idea of the 'deepness' of deep learning can easily be grasped when looking at the graph of a fully connected neural network in Subfigure 3(b), which's schematic nodes in turn can consist of diverse basic building blocks.



(a) Sketch of inner workings of a so-called neuron, containing weights and biases to apply to the inputs, followed by an activation layer.

(b) Stacking of simple building blocks, distinctive for (deep) neural networks and deep learning. Hidden layers transform the input.

Figure 3: Artificial neural network nodes with inner workings of a neuron are shown in the left part. A sketch of the hierarchy and 'deepness' of a neural network is visualized by stacking of schematic building blocks on the right side, where the hidden layers create a non-linear transformation.

Before continuing with the building blocks themselves, it should be mentioned that neural networks as a class of universal approximators [33] can also be seen as natural extensions of more classical machine learning methods like linear and logistic regression, as the whole network prior to the final output layer can be recognized as non-linear transformation while the regression task in the final layer remains unchanged [34].

Since linear and convolutional layers are the primarily used building blocks of the present work, they are introduced in the upcoming paragraph. Linear layers can as an *'affine transformation controlled by learned parameters'* [31] be written down like

$$z_j^l(\vec{x}) = w_{jk}^l x_k + b_j^l, \tag{7}$$

where z_j^l is the j^{th} neuron before activation in layer l of the network and $\vec{x}$ is either input or activation of the previous l-1^{th} layer. w_{jk}^l are the inherent weights and b_j^l is the bias of the respective neuron in the l^{th} layer [34]. In a practical setting the affine transformation is often followed some form of normalization $n(\cdot)$ (like batch normalization) to improve trainability [35, 36], before the outputs are fed into a non-linear activation function to complete one network layer. Common choices for activation functions are the Sigmoid function and different kinds of Rectified Linear Units (ReLU) such as parametric or leaky ReLU. Sometimes also more exotic functions such as Swish [37] are chosen, all of which are ordinarily referred to with $\sigma(\cdot)$ and have a differentiable implementation to allow for gradient-based optimization [34]. One layer of a in this case fully connected neural network is thus complete after passing the linear transformed input through optional normalization and in the whole network at least one non-linearity to obtain the activation

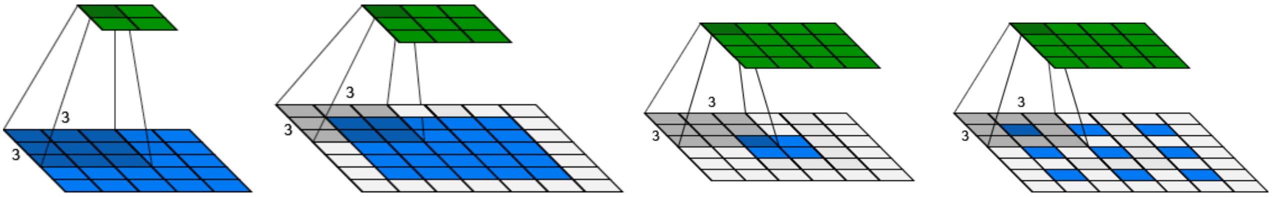
$$a_j^l = \sigma(n(z_j^l(\vec{x}))), \tag{8}$$

which then can be evaluated or fed into the next layer.

Besides the linear layers this work will mainly use so-called convolutional layers. As the name states, the j^{th} two-dimensional preactivation $\boldsymbol{Z}$ of layer l is calculated by convolution (cross-correlation for computational purposes) of image or activation $\boldsymbol{X}$ and kernel $\boldsymbol{W}$ [31]

$$\boldsymbol{Z}^l_j(x,y) = \sum_k (\boldsymbol{X} * \boldsymbol{W}^l_j)(x,y) + b^l_j = \sum_k \left(\sum_m \sum_n \boldsymbol{W}^l_j(m,n,k) \boldsymbol{X}(x+m, y+n, k) \right) + b^l_j \tag{9}$$

with an additional bias b per set of filters j. The typical ordering of the convolution sum, usage of tensor notation as well as naming conventions have been changed in comparison to Goodfellow et al. [31] for visualization purposes. This allows to display the similarity to the linear case of Equation 7 while the dummy index k in Equation 9 serves as a reminder of the usually non-singular amount of input filters involved. Depending on convolutional layer parameters like stride, padding and dilatation, the sequence of used x, y as well as m and n varies and hence allows to increase or decrease the resolution. Depending on the way of implementing the resolution increase, such a layer is often sloppily called transposed- or deconvolution, while the more precise term for a convolutional layer with dilation unequal to zero is fractionally strided convolution [38].



(a) Convolutional layers, which are used for decreasing the input's resolution. Depiction with and without padding at the borders of the input.

(b) Increasing the resolution by padding to the border of the input. The setting with dilation $\neq 0$ and thus creating so-called fractional stride is more common.

Figure 4: Convolutional layers with varying padding and dilatation parameters. The input layer is depicted in blue, the kernel with a kernel size of 3 × 3 for all convolutions in grey and the output of the convolutional layer in dark green. Stride $s = 2$ is used on the right side while the left side keeps the stride to $s = 1$. Padding in white, which is most often achieved by extension with zeros.

As a last addition to convolution parameters, three-dimensional convolutions introduce three-dimensional kernels and thus have a limited spectral range. A three-dimensional convolution inserts one more sum for the kernel's spectral range in above's Equation 9, which also further increases computational complexity.

The more complex affine transformation of a convolutional layer is like in the linear case followed by optional normalization and in the whole network at least one activation layer for non-linearity. For more details on convolutional layer parameters, please have a look into Dumoulin et al. [38].

For both image synthesis and pattern recognition, the filter-weight optimization approach of convolutional layers has two crucial advantages over linear layers:

Learning specific spatially - and in the case of 3D-convolutions also spectrally - finite kernels which are applied to input or hidden layers not only aid with the higher computational burden of the linear case where every node is connected to every other node, but also introduce a natural sense of locality and translational invariance [34].

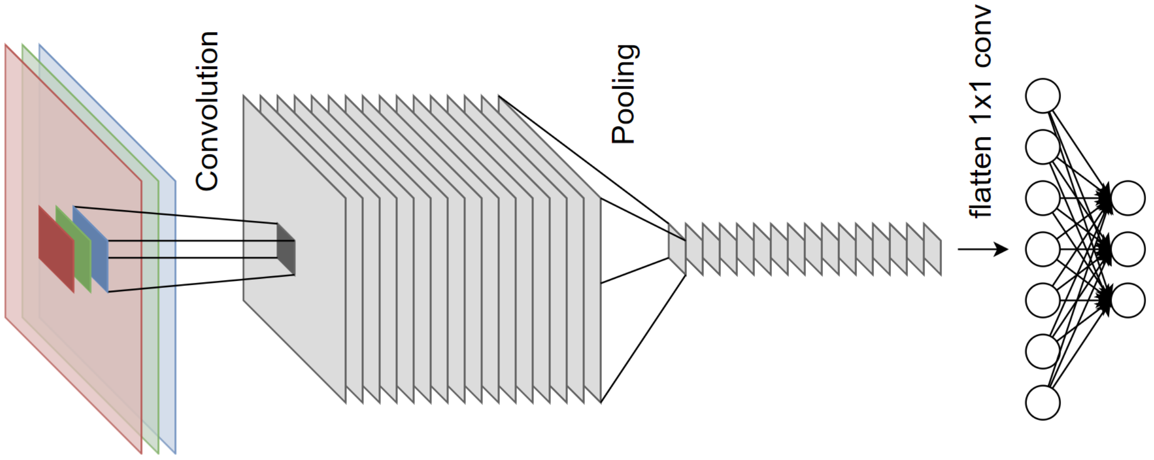


Figure 5: Example graph of a convolutional neural network with RGB input. Equation 9 describes the highlighted convolution operation on the left. Input layers convoluted with a two-dimensional kernel are summed afterwards and result in one feature layer of the next deeper layer. If deep hidden layers are flattened or reduced to a linear dimension e.g. by pooling, these layers can be used as inputs for linear layers. The fully connected output layers on the right illustrate such a case and can be trained to e.g. learn probabilities for a classification task.

This spatial knowledge along with the hierarchical structure allows to gather feature maps which coincide well with human judgement after proper training [39] and also surpass previously hand-crafted features e.g. in terms of classification accuracy [32]. Last, the hierarchical structure together with common decrease of the layers' spatial resolution in deeper layers of the network allows to grasp the global structure of the input image, even when only using small kernels in each layer.

3.2. Neural Network Training

For a more detailed introduction of neural network training and the aforementioned link of neural networks as natural extension to regression, please have a look at Mehta et al. [34] as well as Nielsen et al. [40], which also lay basis for the following subsection.

Training neural networks can roughly be splitted into two steps: The forward pass with error calculation and subsequent backpropagation of the error. In combination, they allow learning of correct representations by means of numerical weight and bias optimization [41].

The forward pass is quite simple and can be achieved by chaining activations of the involved layers as in Equation 8. After obtaining the results from the final layer, calculating a difference between learned and expected outcome is required. Such a cost function $C(\cdot, \cdot)$ needs to compute a meaningful measure of discrepancy between learned and desired outcome and be differentiable, as optimizing the weights of the network demands the ability of gradient computations.

The steps for efficient network parameter optimization are called backpropagation and are explained in the following: Backpropagation traces the difference calculated by means of the cost function C consecutively back to single layers' weights and biases, with the help of four iterative equations. Starting from the end of the network, the first cost change Δ_j^L is defined, which can be attributed to the preactivation z_j^L in the last layer L. Here notation of equations 7 and 8 as well as the assumption of an L layer network are made. This cost change is related to the computable overall cost $C(\vec{a}^L, \vec{y})$ of received result $\vec{a}^L$ and expected result $\vec{y}$ by

$$\Delta_j^L = \frac{\partial C}{\partial z_j^L} = \frac{\partial C}{\partial a_j^L}\frac{\partial a_j^L}{\partial j_j^L} = \frac{\partial C}{\partial a_j^L}.\sigma'(z_j^L). \tag{10}$$

In a similar manner, the cost changes attributed to previous earlier preactivations z_j^l with $0 < l < L$ can be obtained by the recurrect relation

$$\Delta_j^l = \frac{\partial C}{\partial z_j^l} = \frac{\partial C}{\partial z_k^{l+1}}\frac{\partial z_k^{l+1}}{\partial j_j^l} = \left(\sum_k \Delta_k^{l+1} w_{kj}^{l+1}\right)\sigma'(z_j^l). \tag{11}$$

Preactivation influence on the final cost can be traced back to the very first layer with this relation, only lacking a link to the influence of the actual weights and biases. The required link is established via

$$\frac{\partial C}{\partial b_j^l} = \frac{\partial C}{\partial b_j^l}\underbrace{\frac{\partial b_j^l}{\partial z_j^l}}_{=1} = \frac{\partial C}{\partial z_j^l} = \Delta_j^l \qquad \text{and} \tag{12}$$

$$\frac{\partial C}{\partial w_{jk}^l} = \frac{\partial C}{\partial z_j^l}\frac{\partial j_j^l}{\partial w_{jk}^l} = \Delta_j^l a_k^{l-1}, \tag{13}$$

containing weight w_{jk}^l and bias b_j^l. Therefore, individual weights and biases are now related to the computable quantities of cost change Δ and the in the forward pass calculated activations a [34].

After having established a calculation method for the cost changes of the individual weights and biases on the outcome, both can be optimized to improve the outcome of the neural network. The method of choice for the optimization usually is gradient descent, due to higher order derivatives being computationally more costly with only slightly improved minimization behaviour [41]. For sake of brevity, the work of Mehta et al. [34] contains more details on not only more sophisticated methods for gradient descent like Adam [42], which often is the optimizer of choice in this work, but also on the vanishing and exploding gradients problem as well as techniques like batch normalization and Dropout to tackle these training issues.

With the basics of neural network building blocks and training behind, the more technical parts of the *Related Work* and *Materials and Methods* sections can now be dealt with and a full methodological understanding of the presented image generation pipeline is possible.

Part II.
Related Work

Recalling the main objective of investigating realistic hyperspectral tissue patch generation, the following related work presents already existing deep-learning frameworks for *Medical Image Synthesis* and *Image Synthesis Beyond Medicine*. *Image Realism Quantification* finally presents metrics for quantitative comparison of image textures which allow evaluation of the present objective.

1.1. Medical Image Synthesis

In the following, synthesis methods are ordered by the strictness of physical limitation imposed within the framework or method.

Physics-based tissue diffuse reflectance data can be generated by means of Monte-Carlo simulations [25], which have to be fine-tuned by implementing more realistic tissue models. Such models comprise multiple layers or dedicated vessels [43, 44, 45], to meet real tissue responses. Monte-Carlo simulation is the gold-standard [1] for biomedical data generation when ground truth tissue properties like physiological parameters are required [14, 46], as other numerical solutions of the radiative transfer equation and also approximations like the diffusion approximation lack accuracy [8]. However, while this approach might be the physically most precise one, it is timewise costly to simulate due to the large mean absorption length of photons [8] and it is crucially dependent on modeling of tissue geometry and e.g. vessel contents [47], which restricts photo-realism.

Rendering-based image generation provides a link between strictly physics-based and organ-based synthesis, where the latter method only provides a segmentation map or organ model as input. Pfeiffer et al. [16] have explored unpaired image-to-image translation from rendered laparoscopic computer simulations to realistic image data by means of the MUNIT framework [48]. Rendering-based image generation is the link between physics and organ-based modeling, since it uses physically correct rendering of a scene at some point in the image generation pipeline. This includes depth and surface normal maps which are additional, physically correct information, provided on top of segmentation maps as organ ground truth data. While this is impressive both in effort and outcome, the results of Pfeiffer et al. [16] have been limited to the RGB domain and besides the rendered image, the depth maps as well as further rendering details have not been used in the image-to-image translation step. Rivoir et al. [2] have built on this previous work and utilized the provided depth map in the data translation step. So-called neural textures [49], which are differentiably projectable by means of classical image rendering, have been learned in their approach. The combination of learnable textures and model-based depth maps has allowed to create photorealistic and long-term temporally consistent video sequences of abdominal surgeries.

Organ model-based approaches are another alternative for medical image synthesis. As introduced in the previous paragraph, they integrate explicit constraints via an input ground truth to restrict the scene to physiologically meaningful (organ) structures. Marzullo et al. [18] have transformed segmentation maps using the *pix2pix* framework [50] to photorealistic

RGB images, Figure 6 shows example results. The simplicity of the required organ map input data for this case has the advantage, that it can be generated automatically with learning algorithms and help from some expert annotations [51, 52], while the data meshes for the previously described rendering-based models have to be constructed and rendered customly, which requires more work from the expert side [2].

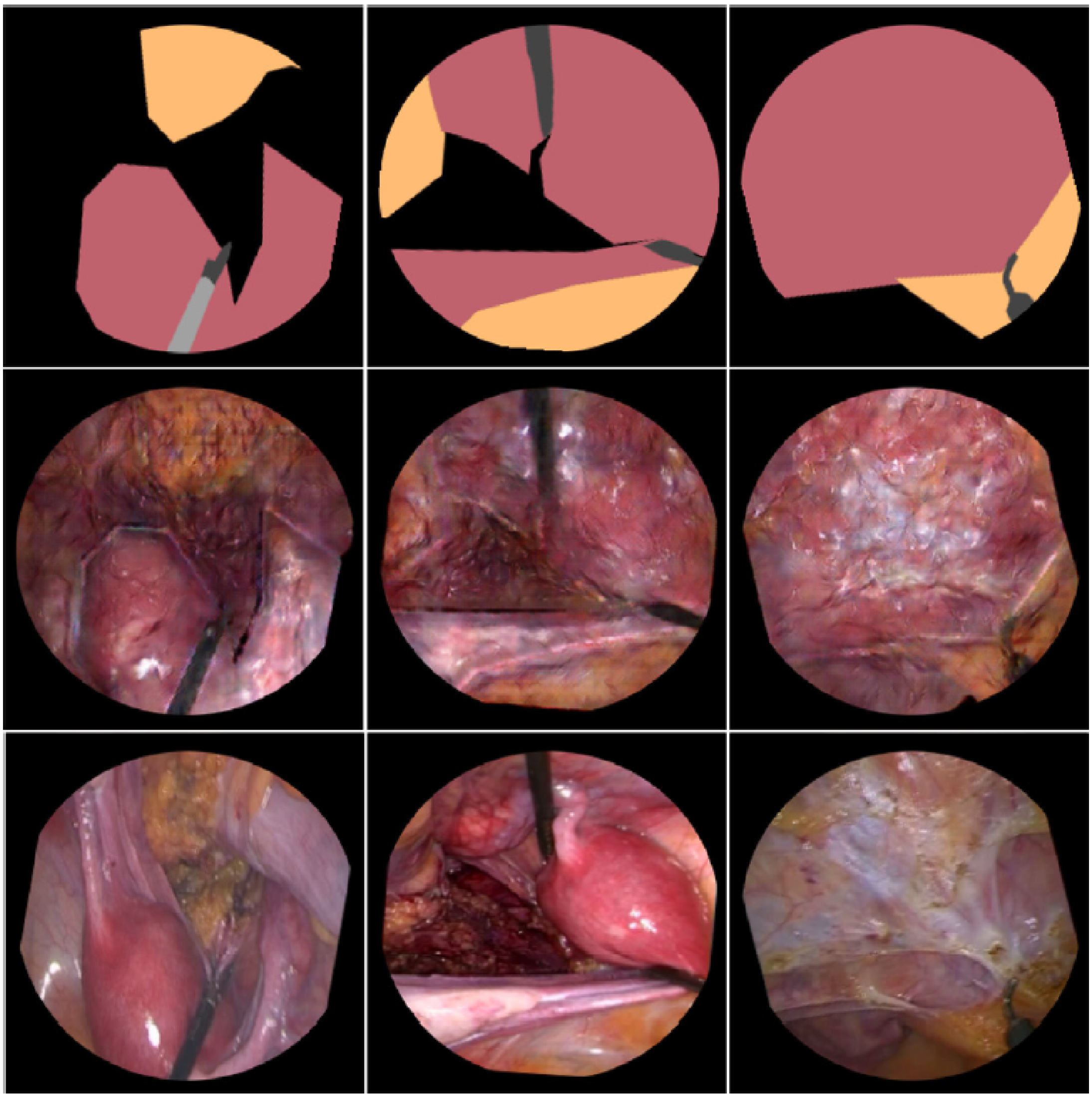


Figure 6: Image-to-image translation from segmentation maps to realistic laparoscopic image. The top row shows ground truth segmentation maps, middle generated results and bottom row ground truth images. Artefacts are visible on the borders of different labelled regions. Furthermore, the leftmost generated result shows signs of repetition or checkerboard artefacts on the yellowish, fatty tissue. Instruments sometimes appear blurred and jittered when looking closer. Reprinted from Computer Methods and Programs in Biomedicine, Vol. 200, Marzullo et al., *Towards realistic laparoscopic image generation using image-domain translation*, p. 105834, 2021, with permission from Elsevier.

Unsupervised concepts try to leverage unlabeled data and still allow partaking of non-experts to a certain degree. In case of an image synthesis tasks, the data obtained by the unsupervised framework is in the evaluation to be inspected and contextualized by humans, which does not necessary require previously mentioned expert knowledge, since qualitative and quantitative comparisons with the real image data can be made. Still, quality assessment of

generated results by experts is often a viable evaluation method [53].
There are several domain transfer approaches in the medical imaging domain, which can be classified as something in between implicitly model-based, conditional and unsupervised learning algorithms. Examples encompass methods which try to minimize radiation exposition by learning *Computed Tomography* (CT) images from *Magnetic Resonance Imaging* (MRI) ground truth [54, 55], methods which transfer between the domains of *Positron Emission Tomography* (PET), CT and MRI [54, 56, 57] or even methods which convert T1- to T2-weighted MRI images and vice versa [58]. Although these concepts are still loosely restricted by their inputs, the control over the input has been decreased and it can not be easily manipulated in a specific way, but on the positive side only sparse to no additional expert annotations are required. To solve this kind of problem, CycleGANs are in literature often preferred over the *pix2pix* framework, as they empirically have shown superior performance, especially in an unpaired data setting or when registration is challenging [55, 59].

Unconditional data synthesis examples from the medical imaging domain can be found for brain MRI synthesis [53] as well as for skin lesions [60, 61]. Yi et al. [60] have aimed to utilize the learned features, which they extract from their synthesis task, for improvement of a downstream classification task whereas Qin et al. [61] try to directly use the newly generated images for the same task.

Synthesis of hyperspectral reflectance images is an unexplored research field, as opposed to plenty segmentation and classification approaches on hyperspectral data in biomedical literature [1, 19]. Occasionally, RGB and sparse HSI data are used to reconstruct full hyperspectral images [62, 63], which comes closest to the intended implementation, but no comparable full image synthesis approaches are found in literature.

1.2. Image Synthesis Beyond Medicine

In the computer vision community, several possible meta-learning frameworks have been proposed for image synthesis. This subsection is going to present some of the larger conceptual frameworks, before preselecting and then presenting approaches in more detail.

Starting off with more recent approaches, neural rendering has shown impressive results for semantic image synthesis or novel view generation [64], but it requires a three-dimensional mesh or equivalent depth map for rendering and associating the textures to the physical, three-dimensional scene. Depending on the implementation, also several images from different viewpoints might be necessary to be able to synthesize new images or specifically novel viewpoints [49], which is further impractical in case of already sparse data. INNs as another recent approach try to mitigate image diversity issues of GANs [65] as well as the blurriness and mode mixing of VAEs [66, 67], which is partly attributed to poor low-dimensional latent space conditioning [68], which causes intermediate representations of poor quality. Keeping the resolution of both latent space and singular network layers large, resolves mentioned problems; however, at the cost of a very large latent space, scaling with the number of spectral bands and resolution.

Due to the drawbacks of previous meta-learning solutions, this paragraph will introduce some influential architecture types useful for the preselected two standard learning approaches GAN and VAE. Radford et al. [69] have proposed simultaneously up- or downsampling the resolution, which they refer to as *Deep Convolutional GAN* (DCGAN). Their proposed method comes usually with results of lower quality but due to the simplicity has better training behaviour. Three other very influential architectures are *VGG* [70], *ResNet* [71] and *U-Net* [72]. All mentioned architectures build on additional layers to learn more detailed filters. *ResNet* and *U-Net* additionally use skip-connections and concatenation of activation or pre-activation results on different levels which aids trainability [73]. Lastly, *U-Net* was specifically designed for biomedical image segmentation and in contrast to the other two influential architectures has image input and output, which is specifically for domain translation helpful [72]. Furthermore, newer architectures which were shown to work in the medical domain [61] and incorporate detailed information on the image's content more explicitly [74] have been considered. While discussing architectures, it is important to keep current findings in literature in mind, which suggest that training strategies and data augmentations rather than specific architectures are the cause of image synthesis improvements [75, 76].

Image generation for hyperspectral data is as for the specific medical hard to find in literature. Work on inpainting, denoising and super-resolution [77, 78] of multispectral satellite images can be found; however, they only enhance existing images rather than generating them from scratch. Sidorov et al. have presented the interesting finding that three-dimensional convolutions do not outperform two-dimensional convolutions [78] which coincides with own early findings and allows to keep the basic building blocks and overall network architecture similar to the popular RGB-based approaches.

1.3. Image Realism Quantification

Central part of the present work are possibilities to quantify received results, particularly since the hyperspectral data extends beyond human recognition, but also to compare results to results from literature. Several metrics used and proposed in works of image synthesis thus are mentioned here, while the details are contained in the *Image Quality Assessment* subsection of the *Materials and Methods* chapter.

Grouping quantification methods coarsely into paired and unpaired metrics, simple **paired methods** like the *mean squared error* (MSE) and *peak signal to noise ratio* (PSNR) allow for global scene comparisons in terms of difference to an original image and sharpness of the generated images' features. More intricate image statistic extractions like *structural similarity* (SSIM) [79] further grant a reliable measure on image texture content [80, 81], although only partially correlating with human judgement [82]. To make up for this flaw, approaches such as the DISTS score [83] try to compose a proper metric which matches human visual assessment by utilizing pretrained *VGG* features for their perceptual features [39]. Most widely used in the imaging community [2, 75, 84] are *Fréchet Inception Distance* (FID) [85] and *Kernel Inception Distance* [86], which are calculated from the pre-activation outputs of an Inception v3 architecture [87] for real and fake image samples.

In the **unpaired methods** sector there exist some methods which try to apply knowledge on image statistics for quality assessment [82, 88]. Mittal et al. [82] for example extract normalized scene statistics from windows of the given image and do regression on a fitted feature statistic function to compute an image realism score. However, most known [48, 84, 89] is the *Inception Score* (IS). Similar to the FID, the IS uses high-level features of the Inception v3 network to predict the realness by calculating the exponential of the Kullback-Leibler (KL) divergence between conditional and marginal class distribution [90], which can also be interpreted as exponential of their mutual information [89].

Summary

Neural rendering requires intricate preprocessing and modeling of three-dimensional surgical scene reconstructions, which is unfeasible in case of non-existence of such data. Thinking of presented, already existing work in the medical and specifically laparoscopic imaging domain led the focus towards frameworks using synthesis from ground truth segmentation maps [18] and domain transfer of simulations [16], especially since the building blocks of neural networks seem to work as well with HSI data [78]. Since synthesis from noise or a latent space is otherwise common in works of state of the art image synthesis, the decision was made to not use the segmentation maps as a basis for image generation like Marzullo et al. [18] did. With the idea of incorporating image manipulations and guided novel synthesis by means of altering latent embedding properties of real HSI data in a conditional or unconditional manner, INNs also become unfeasible. This is due to their high-resolution, high-dimensional latent space, which seemed much harder to supervise. Overall, image synthesis implementations in medicine show a tradeoff between what input (modeling) data is available and obtained quality of results, which is especially visible for the different laparoscopic synthesis papers [2, 16, 18] depicted in Figure 6 and Figure 7.

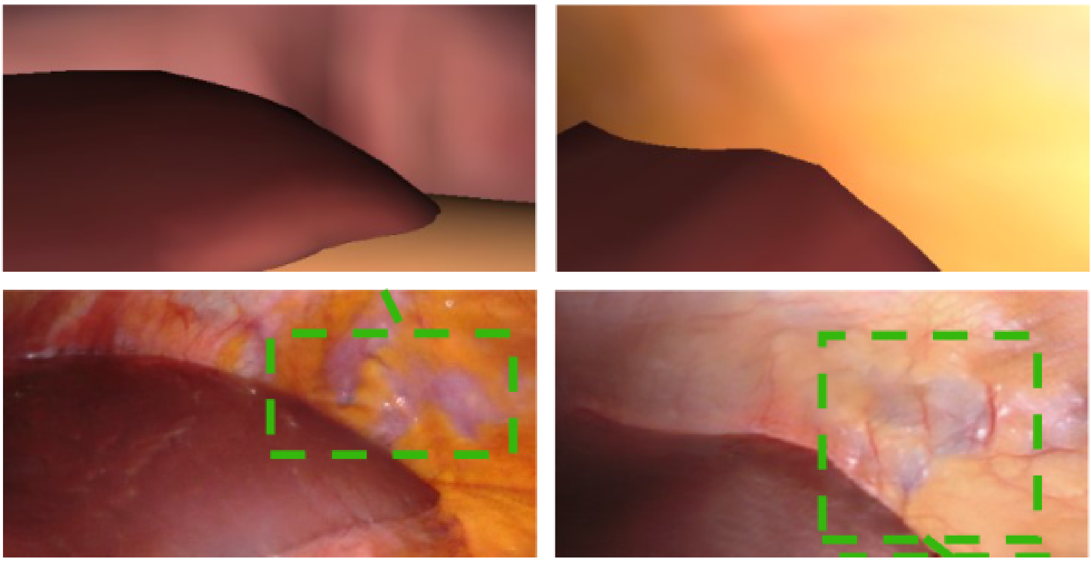


Figure 7: Neural rendering (bottom) of three-dimensional, rendered laparoscopic simulations (top). Green boxes in the original publication of Rivoir et al. [2] outline consistent video frames, which are not shown here. Obtained lighting of the surgical scene and physiological structures like vessels and organ borders look realistic in this unpaired domain translation approach. Adapted from [2], permitted by *Creative Commons Attribution 4.0 International* (CC BY 4.0) license.

Part III. Materials and Methods

After having introduced principles of *Radiative Transport in Tissue*, devices to record the observed reflectance spectra as well as basic deep learning and neural network building blocks, this methods chapter elaborates the conclusions derived from *Related Work*. Before describing the proposed *Image Generation Pipeline* in detail, the *Concept Overview* and the *Hyperspectral Imaging Datasets* with acquisition and contents of the dataset used throughout this work are presented. Furthermore, central calculations for the *Image Quality Assessment* are introduced, before continuing with the experiments and computing their results.

Concept Overview

To enable realistic hyperspectral tissue synthesis, the chosen deep learning models have to ensure physiological and thus also spectral realism. To guarantee both points and also being able to work with minimal data, an unsupervised pipeline was developed.

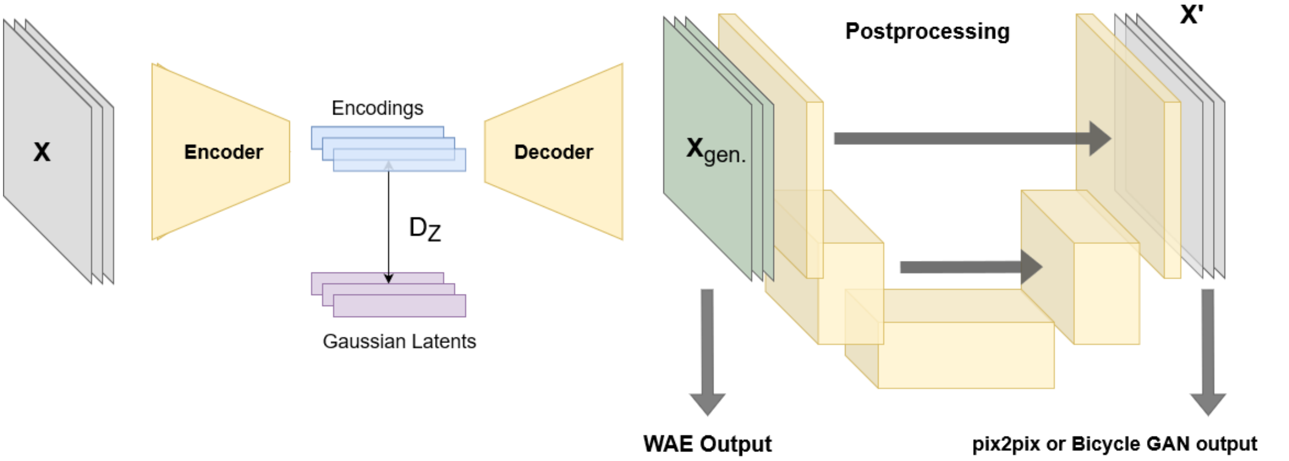


Figure 8: WAE on the left, with Gaussian latent embedding. In a second step, the intermediate WAE results $\boldsymbol{X}_{gen.}$ are postprocessed by means of a *pix2pix* approach or Bicycle GAN, which both use a U-Net generator network. While the WAE can generate paired image results, postprocessing visually improves the obtained WAE results for a more realistic texture.

Physiological constraints enter the proposed deep learning pipeline not explicitly but implicitly in the first step of Figure 8, when a so-called *Wasserstein Autoencoder* (WAE) [67] learns the image data manifold [31]. The WAE was chosen over a classical VAE for proposed better visual quality as it solves issues with low-quality encoding which are attributed to the in comparison to GANs worse VAE performance while avoiding hallucinations [91]. Another big advantage of Autoencoders is the existence of a corresponding original image, which is required for spectral one-on-one comparisons. Otherwise, the precise spectral quality comparison would not be possible, as there are no further labels involved.
This was partially due to the larger, unlabelled dataset, which restricted available labelled data and is introduced in the next section, but mostly since conditional WAE results returned results

of lower quality in initial attempts. Lower quality in this case referred to both lower visual quality as well as exhibition of strong correlations between label input and latent vector, which resulted in non-meaningful image patches when images were generated from not corresponding random label vectors and random latent vectors. Example visualizations of a conditional approach can be found in *Training Result: WAE Different Decoder Results*. Similarly, approaches with three-dimensional convolutions or approaches which incorporated generation of segmentation maps along with HSI data suffered from low-quality results and are not presented.

Outcomes of the WAE as the first pipeline step can then be used in domain transfer, postprocessing and one-to-many approaches [48, 92] as proposed in literature [2, 16, 18]. For the postprocessing of the HSI data, generated by the WAE, two frameworks are presented and compared. One of the frameworks utilizes image-to-image translation as Marzullo et al. [18] while the second one treats the intermediate images of the WAE as a domain adaptation problem.

1. Hyperspectral Imaging Datasets

The foundation for usage of the presented deep learning frameworks in *Image Generation Pipeline* are two HSI datasets. The semantic dataset was mainly used for the presented work and contains fully labelled hyperspectral images, while the masks dataset only contains partially labelled polygon 'masks'. The masks dataset was initially used and acquired with different cameras, which leads to it only reappearing in the *Discussion and Conclusion*. Before referring to further details of the *Datasets, Preprocessing and Data Loading*, the *TIVITA® Tissue* HSI-camera, used for image acquisition, is introduced.

1.1. TIVITA® Tissue

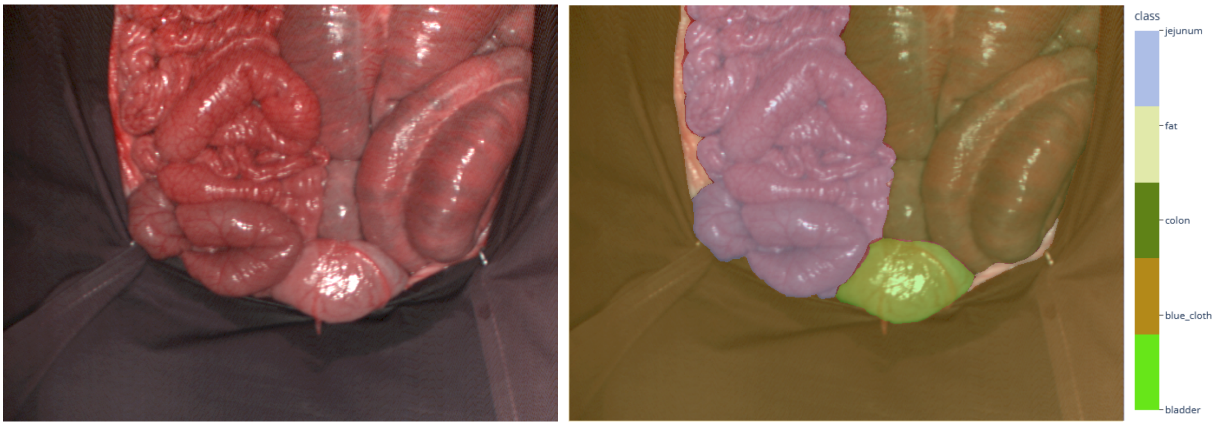
The CE (Conformité Européenne) certified TIVITA® Tissue (Diaspective Vision GmbH, Am Salzhaff, Germany) HSI-camera was utilized for acquiring spectral data in the wavelengths from 500 - 1000 nm by means of a pushbroom imaging spectrograph (*Spectral Scanning*) along the y-axis of the sensor [9]. This wavelength range was chosen by the manufacturer, as it comprises absorption bands of important physiological properties as seen in Figure 1. Furthermore, properties of different tissue depths were observed by utilizing the wavelength range:
A typical light penetration depth at 500 nm is due to higher absorption by chromophores only approximately 0.8 mm, whereas the penetration depths increases up to 2.6 mm for 1000 nm [9, 93]. Therefore, Holmer et al. [9] exploited wavelengths from 500 - 650 nm to extract surface tissue properties, while the near-infrared wavelengths allowed to obtain physiological responses from deeper tissue layers.

On the technical side, a complementary metal oxide semiconductor sensor of resolution 2048 × 1088 was used for taking the images [9]. The first internal preprocessing step of the camera runs an initial calculation on a selected region of interest of size 960 × 780 [9]. This calculation returns 500 wavelengths between 500 - 1000 nm [9]. By means of a binning algorithm, this spatial×spectral resolution is reduced to 480 × 100 [9]. A stepper motor moves the optical slit of the pushbroom device to acquire 640 line images, yielding the final hyperspectral image cube size of 640 × 480 × 100 [9]. For more technical details please consult Kulcke et al. [94].

In clinical application, the camera covers an area of 20 × 30 cm, when measuring at distance of 50 cm with acquisition times of approximately 5 s per image [94].

1.2. Datasets, Preprocessing and Data Loading

Both semantic organ segmentation and masks dataset were acquired at the Heidelberg University Hospital after approval by the Committee on Animal Experimentation of the regional council Baden-Württemberg in Karlsruhe, Germany (G-161/18 and G-262/19) and were also used in data-wise similar HSI segmentation work of Seidlitz and Sellner et al. [4]. Hyperspectral images were taken for 20 pigs that were managed according to the German laws for animal use and care and in agreement with the directives of the European Community Council (2010/63/EU).



(a) Example image from the train dataset.

(b) Segmentation map, blended over the RGB image.

Figure 9: Example RGB visualization of a full training image on the left side. The image is the first one taken for pig 41 and the corresponding annotations are displayed on the right. For the semantic dataset, every pixel in the image is labelled.

18 organ class annotations for 506 images from 20 pigs were acquired during the course of work of Seidlitz and Sellner et al. [4], with one example image depicted in Figure 9. In this work, the semantic segmentation maps were not utilized for the training of the model, due to worse early experimental results, and only during organ-specific evaluation taken advantage of. For more details on the image and annotation acquisition, please consult Seidlitz and Sellner et al. [4]. For the larger mask dataset, 11.860 images from 90 pigs were acquired and partially or fully annotated.

As proposed by the manufacturer, the HSI data cubes were corrected with white reference and dark current corrections [4, 9]. Besides clipping the individual HSI data cube values to the range of $[0, 1]$, no additional preprocessing has been applied to the reflectance data.

Conversion of HSI data into RGB images is an additional processing step, which is required for visualization as in Figure 9, but also for some evaluation methods that involve RGB data. Beneath the transformation to RGB images, the camera manufacturer implemented additional algorithms for calculation of tissue oxygenation StO_2, perfusion ν, tissue hemoglobin index and tissue water index parameter images [9].

From the overall 20 pigs of the semantic HSI dataset with individual identifiers, images from the five pigs with identifiers 43, 46, 62, 68 and 72 were always used as test dataset and thus remained untouched until the final evaluations. When not artificially limiting the training data, images from pigs with identifiers 48, 57 and 58 made up the validation set and images from the remaining 12 pigs were used as training data.

In the special case of the artificially limited data setting for the *Downstream Task: Image Segmentation* experiment, only data from pigs with identifier 47, 50 and 57 was utilized as training data for the *Image Generation Pipeline*. For clarification: The test dataset in this instance was kept as introduced before.

The decision to use HSI patches of shape 64 × 64 × 100 was motivated by findings that the patch size correlated with discriminability [50] as well as considerations to limit the complex data to an easier handlable problem. If not mentioned differently, the HSI patches were loaded with the same dataloader of Seidlitz and Sellner et al. [4] which randomly selects image patches from the HSI dataset in a CPU efficient manner. Albumentations' [95] `ShiftScaleRotate` with default parameters and augmentation probability $p = 0.9$ as well as `Flip` with probability of $p = 0.5$ were by default applied to the full hyperspectral images before randomly cropping the patches. The selection criterion for the named augmentations was their naturalness, meaning that they do not alternate the data distribution beyond what would be physiologically possible.

2. Image Generation Pipeline

The in *Concept Overview* presented image generation pipeline is explained in this section, which comprises detailed concept, architecture and loss descriptions. Following the order of the pipeline itself, the *Wasserstein Autoencoder* is introduced first, continued with a brief definition of the *GAN Learning Approach* and ended with *pix2pix Postprocessing* and *Bicycle GAN Postprocessing*.

2.1. Wasserstein Autoencoder

The *Wasserstein Autoencoder* (WAE) is the first step of the image generation pipeline and allows to approach the synthesis task in an unsupervised manner. WAEs represent a progression in optimization compared to VAEs and are the generalization of adversarial autoencoders. Tolstikhin et al. [67] have claimed stable training, a structured latent manifold and improved sample quality as their pivotal properties, which are introduced with architectural details in the next subsections that also include architectural details.

Optimal Transport Learning Approach

WAEs take a new approach on generative models, motivated by a similar optimization method as the Wasserstein GAN [96]. Instead of minimizing the negative log-likelihood by means of optimizing the KL divergence as in VAEs [66], properties of the optimal transport (OT) cost are exploited [67]. OT is an initially by Monge introduced and later by Kantorovich relaxed problem [97], which searches for the best transport plan between two distributions according to a given measure. The following paragraphs outline the main theoretical steps described in the work of Tolstikhin et al. [67], since the underlying idea is quite different from omnipresent GAN image synthesis.

As an introduction to the concepts behind the WAE, Kantorovich's formulation of the OT problem defines the optimal transport cost W_c corresponding to a measureable cost function $c(x, y) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$. Optimal transport then seeks to minimize

$$W_c(P_X, P_G) := \inf_{\Gamma \in \mathcal{P}(\boldsymbol{X} \sim P_X, \boldsymbol{Y} \sim P_G)} \mathbb{E}_{(\boldsymbol{X},\boldsymbol{Y}) \sim \Gamma}[c(\boldsymbol{X}, \boldsymbol{Y})]. \tag{14}$$

Γ is a coupling from within the set $\mathcal{P}$ of all joint distributions of $(\boldsymbol{X}, \boldsymbol{Y})$ with marginal distributions P_X, P_G, which can be thought of as a transport map. The marginal distributions P_X and P_G are already named intuitively after real image distribution $\boldsymbol{X} \sim P_X$ and generated image distribution $\boldsymbol{Y} \sim P_G$.

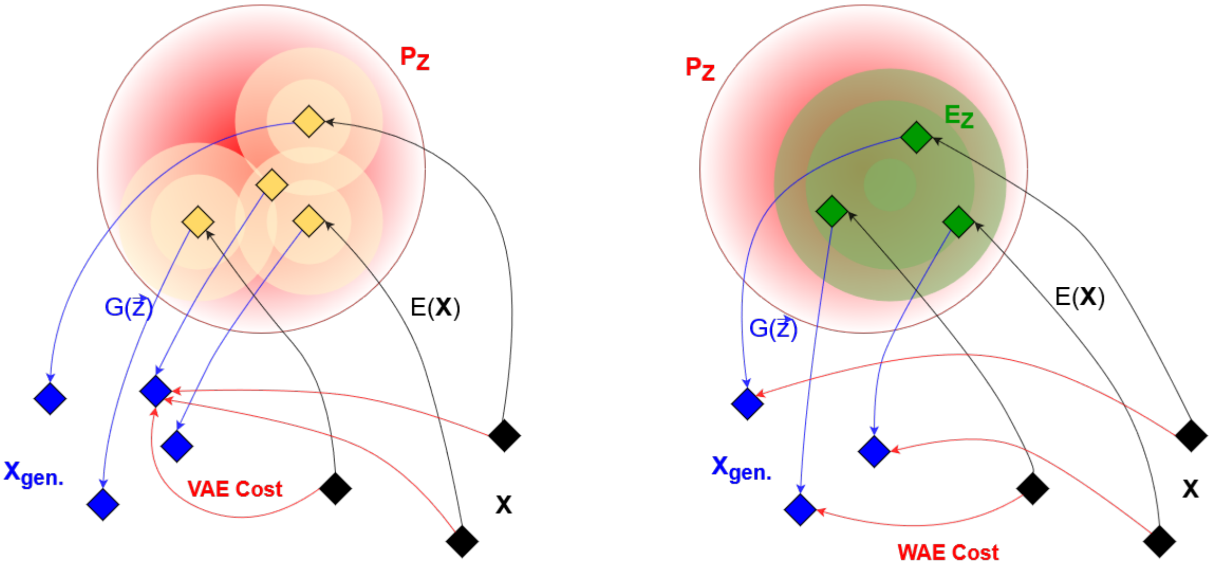


Figure 10: Comparison of VAE and WAE image reconstruction procedure. The OT cost minimization and usage of random decoders G are claimed to lead to better results. This is achieved through a different embedding approach: The VAE on the left embeds images as a distribution, which all have to match the latent distribution P_Z. Contrary, singular WAE embeddings only have to be likely samples from the said distribution. The reconstruction hence becomes easier, as the WAE embeddings are less restricted and not reconstructed from overlapping regions which would form a mixture of images.

Latent space sampling and decoding are the two steps of the WAE generation procedure: First a latent code $\vec{z} \in \mathcal{Z}$ is sampled from a fixed distribution P_Z and afterwards mapped to an image $\boldsymbol{X} \in \mathcal{X}$ by means of a deep learning decoder $G : \mathcal{Z} \to \mathcal{X}, \boldsymbol{X} = G(\vec{z})$. While this sounds familiar when the concept behind VAEs is known, the ingenious part of the WAE concept is the factoring of the transport plan through the decoder G: The search for good visual results in image space $\mathcal{X}$ can be reduced to a search for a good embedding distribution $E(\vec{z}|\boldsymbol{X})$ of the encoder E, which's marginal distribution $E_Z(\vec{z})$ has to be identical to the prior distribution P_Z. This big difference between the VAE and WAE reconstruction procedure is visualized in Figure 10.

Returning to the mathematical description, the OT problem from Equation 14 can in a first step be rewritten as

$$\inf_{\Gamma \in \mathcal{P}(\boldsymbol{X} \sim P_X, \boldsymbol{Y} \sim P_G)} \mathbb{E}_{(\boldsymbol{X},\boldsymbol{Y}) \sim \Gamma}[c(\boldsymbol{X}, \boldsymbol{Y})] = \inf_{E : E_Z = P_Z} \mathbb{E}_{P_X} \mathbb{E}_{E(\vec{z}|\boldsymbol{X})}[c(\boldsymbol{X}, G(\vec{z}))], \tag{15}$$

with proof in the work of Tolstikhin et al. [67]. The novel formulation in Equation 15 concretizes the computation to cost calculation and optimization over the encoder distribution E while enforcing $E_Z = P_Z$.

Further relaxation of this problem then allows numerical solution: In this relaxation, optimization is done over the encoder distribution $E(\vec{z}|\boldsymbol{X})$ for any nonparametric set of probabilistic encoders $\mathcal{E}$ [67] and only a regularizer term enforces $E_Z = P_Z$. The final WAE objective with encoding distribution regularizer $\mathcal{D}_Z(E_Z, P_Z)$ reads

$$D_{\text{WAE}}(P_X, P_G) := \inf_{E(\vec{z}|\boldsymbol{X})\in\mathcal{E}} \mathbb{E}_{P_X}\mathbb{E}_{E(\vec{z}|\boldsymbol{X})}[c(\boldsymbol{X}, G(\vec{z}))] + \lambda \cdot \mathcal{D}_Z(E_Z, P_Z). \tag{16}$$

An additional hyperparameter $\lambda > 0$ determines the strength with which $E_Z = P_Z$ is enforced and the regularizer term $\mathcal{D}_Z$ can be chosen as an arbitrary divergence, but should be able to distinguish E_Z and P_Z. The authors propose both a GAN-based and a maximum mean discrepancy (MMD) [98] divergence $\mathcal{D}_Z$; however, also remind the reader that meaningfulness of the MMD depends on latent space dimension as well as sample amount. In the present case this is especially important, since the latent dimension was increased for accurate hyperspectral image embeddings, while the sample amount stays fixed and at a much lower amount than the dimensionality of the latent space.

Encoder

The work of Tolstikhin et al. [67] utilized a DCGAN-like architecture for encoding, which worked with four layers of the sequence convolution, batch normalization and ReLU activation. The first convolution generated 128 channels and the following ones each doubled this number. Kernel size for all convolutions was 5 with stride 2, which effectively decreased the resolution without the need for pooling. Having reached the deepest layer, the output was flattened and encoded into a latent space of dimension 64 for the CelebA dataset, which comes closest to the used HSI data, as the original WAE paper scaled the images to a resolution of 64 × 64 × 3.

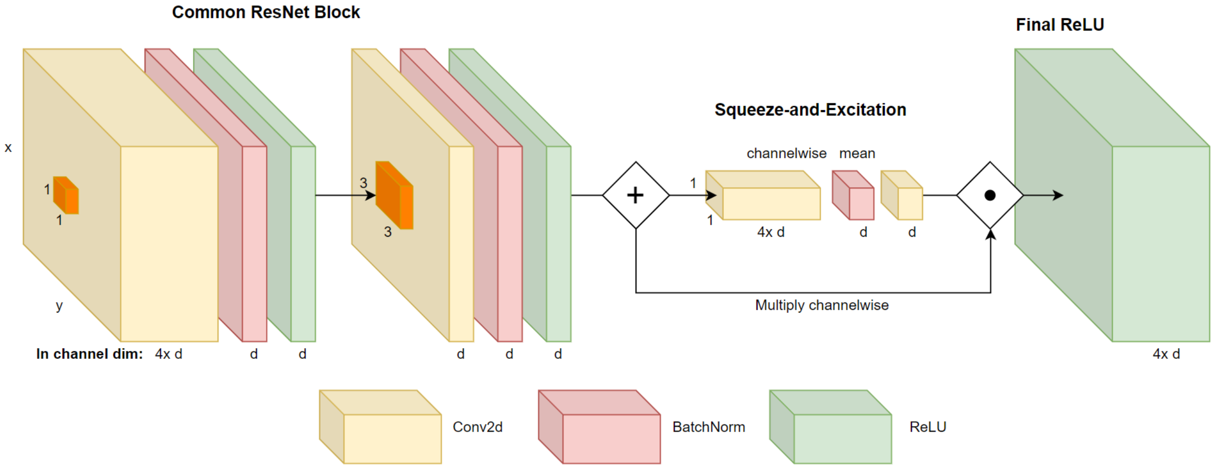


Figure 11: Example ResNet-RS block used in the encoder with channel dimension multiplier d. Multiplier values for the four different stages of the ResNet-RS 50 encoder blocks, which follow the repetition pattern 3-4-6-3, are 64, 128, 256 and 512 respectively. In the Squeeze-and-Excitation Block the input of the previous activation is average-pooled channelwise, processed and after passing through a sigmoid function multiplied with the initial layers.

While the simplistic architecture showed to work quite well for initial results, the presented implementation exchanged the simplistic DCGAN-like encoding for the much deeper ResNet-RS 50 [76], 50 layers deep and mainly consisting of four stages of ResNet-RS building blocks, which are schematically visualized in Figure 11. The stacking scheme of the building blocks was as is the original ResNet 50 [71] 3-4-6-3, meaning the stem of the network with some initial two-dimensional convolutions and downsampling was followed by 3 blocks of stage one, then by

4 blocks of stage two until the fourth stage blocks were passed. Before applying global pooling, the network output was taken, flattened and with one linear layer and reparametrization trick [66] embedded into the latent space. Pytorch Image Models' [99] implementation of the ResNet-RS 50 was used and the global pooling part was discarded and replaced with the previously mentioned custom fully connected layers, to be more similar to the original WAE architecture [67].

Decoder

DCGAN-like decoding was used in the original work, with an initial large fully convolutional, reshaped to $8 \times 8 \times 1024$ before upsampling the resolution. Fractionally strided convolutions with a kernel size of 5, followed by batch normalization and ReLU activations were used for upsampling and the channel dimension was halved after every layer.

Since there are quite a few GANs with novel architecture concepts like adaptive instance normalizations [75, 74], several options for updates on the decoder part exist in comparison to the original implementation. In Figure 12 the VGG-inspired structure used in the present work is shown, which adds convolutional layers before each fractionally strided convolution and applies batch normalization as well as the activation function. Additional decoder concepts which yielded visually worse or at best similar results are listed in the *Appendix.*

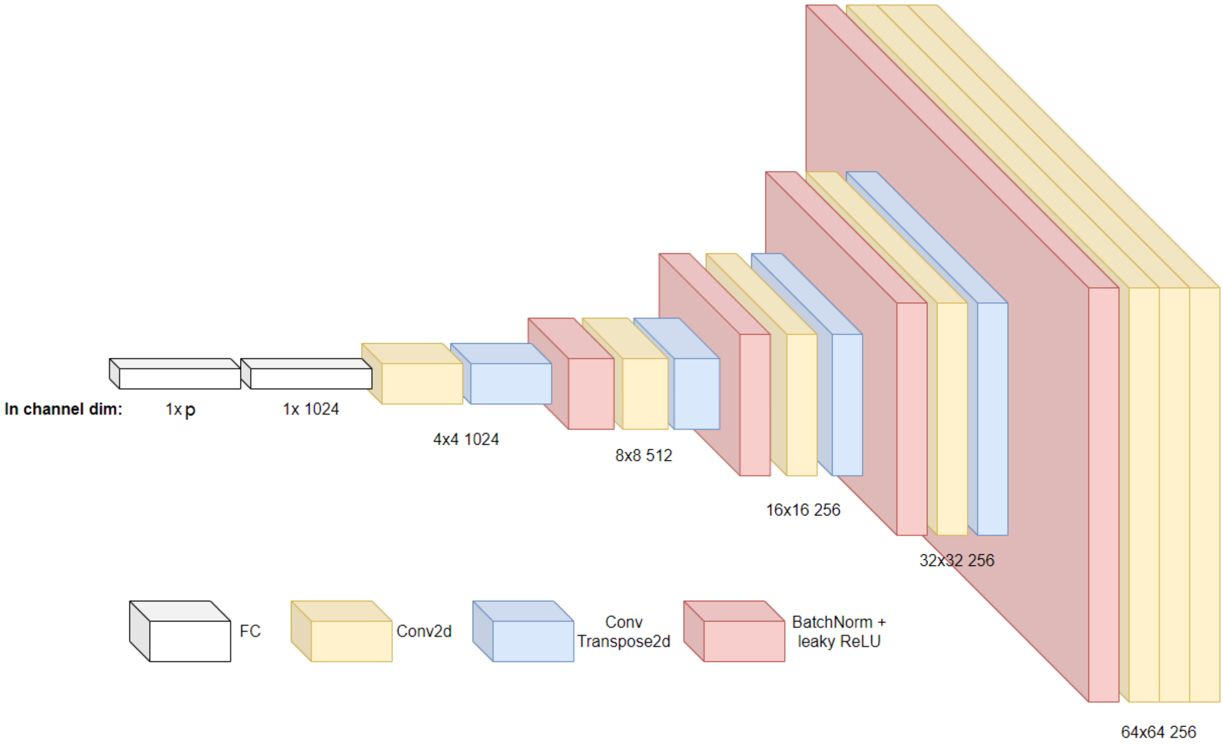


Figure 12: Decoder of the implemented WAE with latent dimensionality p. The first two convolutional layers use a kernel size of 3, the second two a kernel size of 5 and the last three convolutions kernel sizes of 5, 3 and 1 from left to right. While the resolution is kept constant in the convolutional layers by choice of the according padding values, the fractionally strided or 'transposed' convolutions use a kernel size of 4 and stride of 2 to double the resolution.

Training Procedure

The theoretical WAE objective from Equation 16 included one hyperparameter and two general choices of cost function and regularizer. Choices regarding the hyperparameter and regularizer of the Tolstikhin et al. [67] are shown in Table 1. Detailed choices of this works' implementations are listed in the *Overall Training Details* section. The authors [67] chose mean squared error as cost, provided examples for both regularizers and used Adam [42] with lowered $\beta_1 = 0.5$ and default $\beta_2 = 0.999$. Building on previous findings from literature [50, 100], this work used a cost function combined from mean absolute error and SSIM. In contrast to the original WAE implementation, the latent dimension was increased for the grown image contents in terms of spectral features, which increased from 3 for the RGB CelebA images to 100 for the HSI data. Only the GAN regularizer was used as a consequence, since a reliable MMD estimation requires the number of samples to roughly match the dimensionality of the input [101].

Dataset	Spatial Resolution	Latent Dimension	λ	Regularizer
CelebA	64×64	64	1	GAN
CelebA	64×64	64	100	MMD

Table 1: Hyperparameter choices for WAE according to Tolstikhin et al. [67].

The regularizer implementation was the same as in the original work and consisted of altering fully connected layers with 512 nodes and ReLU activations, stacked four times. Using the mean absolute loss

$$\mathcal{L}_1(\boldsymbol{X}, G(E(\boldsymbol{X}))) = \mathbb{E}_{\boldsymbol{X}\sim P_X}[||X - G(E(\boldsymbol{X}))||_1], \tag{17}$$

the SSIM introduced in Equation 30, the GAN regularizer and the shorter notation $\boldsymbol{X}_{gen.} = G(E(\boldsymbol{X}))$, the final WAE loss term reads

$$D_{\text{WAE}}(\boldsymbol{X}, \boldsymbol{X}_{gen.}) = \mathcal{L}_1(\boldsymbol{X}, \boldsymbol{X}_{gen.}) + \lambda_{\text{SSIM}} \cdot \text{SSIM}(\boldsymbol{X}, \boldsymbol{X}_{gen.}) - \lambda \cdot \mathbb{E}_{\boldsymbol{X}\sim P_X}[\log(D(E(\boldsymbol{X}))] \tag{18}$$

when utilizing the log-trick for GAN regularizer. In above's equation G, D and E are decoder, GAN regularizer (discriminator) and encoder respectively.

Discriminator training was achieved via standard GAN discriminator training with binary cross entropy loss. Encoded real images $\vec{z}_{\text{real}} = E(\boldsymbol{X})$ for $\boldsymbol{X} \sim P_X$ and $\vec{z}_{\text{real}} \sim E_Z$ were in this step enforced to be similar to prior samples $\vec{z}_{\text{fake}} \sim P_Z = \mathcal{N}_p(0, 1)$ from a p-dimensional Gaussian with unit variance to regulate the latent manifold. For more details on GAN discriminator training, please have a look at the *GAN Learning Approach* paragraph.

The presented implementation also used Adam optimizer [42] with customized parameters, as objections regarding generalization performance like those presented by Wilson et al. [102] were not observed since the problem is likely not overparametrized.

2.2. Post-Processing with Generative Adversarial Networks

For the second half of the image generation pipeline, image patches generated or reconstructed by the WAE are fed into a *Generative Adversarial Network* (GAN) to improve visual quality. Before presenting the two different implementations used for this task, the general *GAN Learning Approach* is presented.

GAN Learning Approach

Generative Adversarial Networks were initially proposed by Goodfellow et al. [65] and have not only revolutionized image synthesis but also led to an astonishing variety of models grounded on their idea with advancing architectures, loss terms and training procedures.

GANs consist of two neural networks, which are rewarded for contrary tasks. At first, there is the generator G, decoding input noise $\vec{z}$, most often coming from a p-dimensional Gaussian $P_Z \sim \mathcal{N}_p(0, 1)$, into an image. Second, there is the discriminator D trying to distinguish real image samples $\boldsymbol{X}$ from synthetically generated fake ones $G(\vec{z})$. This encourages the generator to generate image patches or whole images which are similar to the given data, whilst the discriminator slowly improves in telling real and fake data apart. Initially [65] the problem was formulated as minimax game with value function V

$$\min_G \max_D V(D, G) = \mathbb{E}_{\boldsymbol{X} \sim P_X}[\log D(\boldsymbol{X})] + \mathbb{E}_{\vec{z} \sim P_Z}[\log(1 - D(G(\vec{z}))], \tag{19}$$

where previous notation for the corresponding data distributions was used. The dependence of V on input values, $\boldsymbol{X}$ and $\vec{z}$ in this example, is left out to make clear that these contrary to D and G are not part of the optimization. In a more implementation friendly way, the samples synthesized by the generator get judged by the discriminator according to the generator's cost function $c_G(G, \vec{z})$

$$c_G(G, \vec{z}) = \log(1 - D(G(\vec{z})) \approx -\log(D(G(\vec{z})), \tag{20}$$

where in the approximation the so-called log-trick was introduced. The log-trick serves the purpose to improve early training, where gradients tend to vanish when calculated with the first formulation. The discriminator is provided with information 'real' and 'fake' for its inputs and trained according to the discriminator cost $c_D(D, \boldsymbol{X}, \vec{z})$

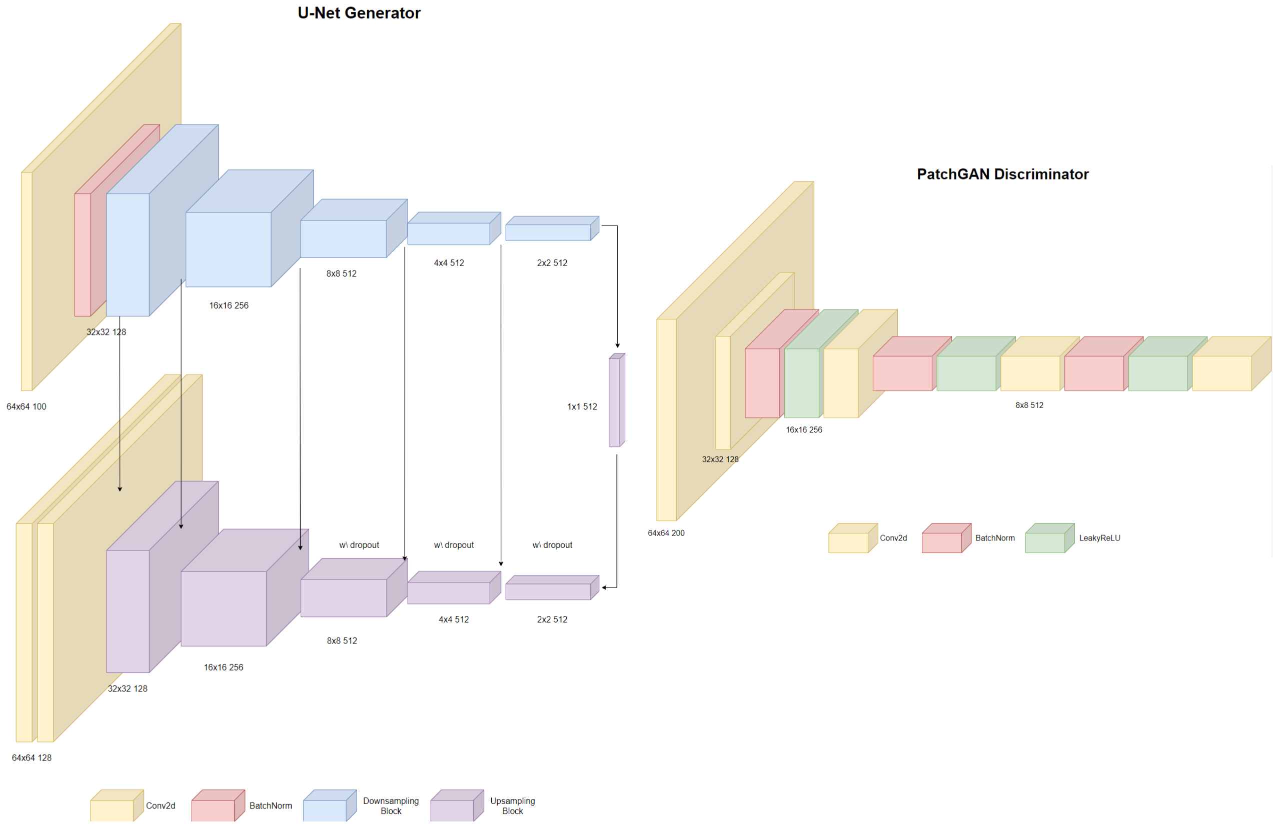
$$c_D(D, \boldsymbol{X}, \vec{z}) = \log(D(\boldsymbol{X})) + \log(1 - D(G(\vec{z})) \approx \log(D(\boldsymbol{X})) - \log(D(G(\vec{z})), \tag{21}$$

where the same trick as for the generator is introduced. While the problem formulation is quite easy, solving the minimax game often leads to ending up in unfavourable local optima and thus bad training results. The computer vision community has therefore put a lot of effort into improvements such as the Wasserstein GAN [96], training GANs with two time-scale update rules [85] or the even more recent U-Net discriminator [103] to name only a few. The in the thesis incorporated advancements of improved loss terms and architectures are described in more detail in the next two paragraphs. Since training still often led to unfavourable results, also quite some effort has been put into hyperparameter optimization.

pix2pix Postprocessing

Implementation-wise, the first approach is similar to the work of Marzullo et al. [18], which was introduced in the *Related Work* section. The output image patches of the pipelines' first WAE stage are treated as coarse data map which has to be made (more) realistic. An adapted lightweight and lower-resolution version of their U-Net [72] generator network is shown in Figure 13, where also the PatchGAN [50] discriminator with additional conditional guidance [104] is displayed.

The generator was trained in similarly to the initial image-to-image translation work by Isola et al. [50] which was also used by Marzullo et al. [18]. Both works translated image label maps into real images and further have in common, that they added a ℓ_1-regularizer term into the GAN optimization setting. The latter approach additionally used another ℓ_1-regularizer to assure sharp medical tool borders [18].



(a) Simplified generator with Down- and Upsampling Blocks described in the overall figure caption. The last two convolutions use kernel size 3 and 1.

(b) Simplified PatchGAN discriminator. The output is fed to a sigmoid function to assess conditional realness of the concatenated inputs.

Figure 13: *pix2pix* generator and discriminator networks. Given channel dimensions always refer to the input dimensions. Downsampling blocks consist of two-dimensional convolutions with kernel size 4, stride 2 and padding 1 followed by batch normalization and leaky ReLU. Upsampling blocks consist of fractionally strided convolutions with the same parameters as the downsampling convolutions, followed by batch normalization and ReLU. Similarly, the first three convolutional discriminator layers downsample with kernel size 4 and stride 2, the last two keep the resolution constant (kernel size 3).

Inspired by further work [80, 105], this work further used pretrained VGG features to achieve better texture results. Formulating the mentioned aspects mathematically gives with channel dimension C_l, width W_l and height H_l of the individual feature maps the VGG-loss term

$$\mathcal{L}_{VGG}(\boldsymbol{X}, G(\boldsymbol{X}_{gen.})) = \sum_l \frac{50}{C_l W_l^2 H_l^2} \mathbb{E}_{\boldsymbol{X}\sim P_X, \boldsymbol{X}_{gen.}\sim P_{\mathrm{WAE}}}[(\Phi_l(\boldsymbol{X}) - \Phi_l(G(\boldsymbol{X}_{gen.})))^2] \tag{22}$$

with VGG-features Φ_l, intermediate image $\boldsymbol{X}_{gen.} \sim P_{\mathrm{WAE}}$ and original image $\boldsymbol{X} \sim P_X$. A pretrained torchvision [106] implementation of the VGG19 was used, where the input layer was adapted for the HSI data by repeating the pretrained weights. As in previous work [80, 105] the five post activation outputs before each downsampling step in the VGG network were utilized, in above's formula referred to with the layer index l, at which the output was extracted from the network. With absolute loss term from Equation 17 and GAN objective from Equation 19 referred to as $\mathcal{L}_{GAN}$ the final objective reads

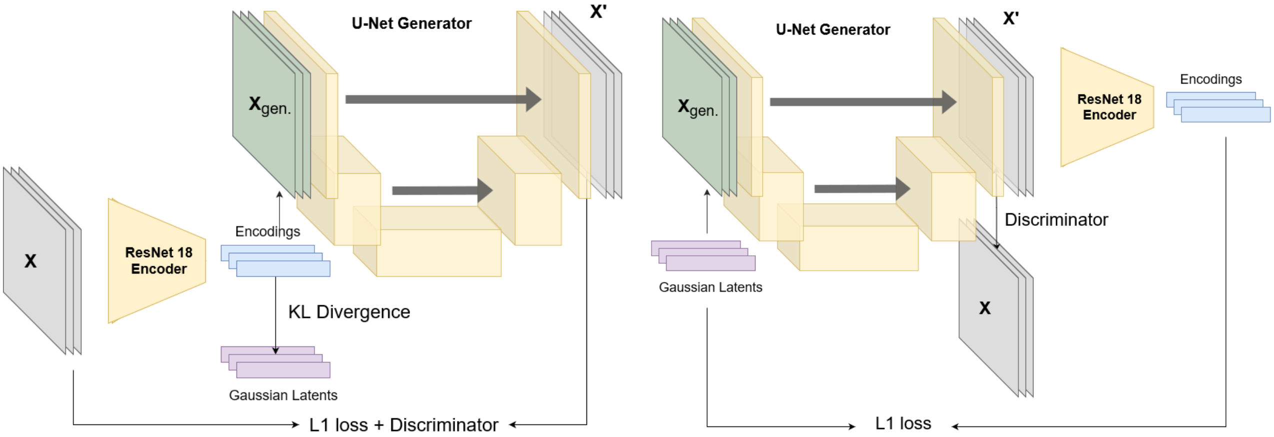
$$\min_{G}\max_{D} V(D,G) = \lambda_{GAN} \cdot \mathcal{L}_{GAN}(D, G, \boldsymbol{X}, \boldsymbol{X}_{gen.}) + \mathcal{L}_1(\boldsymbol{X}, G(\boldsymbol{X}_{gen.})) \\ + \lambda_{VGG} \cdot \mathcal{L}_{VGG}(\boldsymbol{X}, G(\boldsymbol{X}_{gen.})). \qquad (23)$$

During training, the GAN term $\mathcal{L}_{GAN}$ rather than the ℓ_1-regularizer term $\mathcal{L}_1$ was scaled with the hyperparameter λ_{GAN}. Furthermore, also the VGG-loss term $\mathcal{L}_{VGG}$ is scaled with a respective hyperparameter λ_{VGG}.

Lastly, since there are multiple networks from which the presented network inherits different parts, no specific hyperparameters or optimizers could be derived from literature. Details on the chosen hyperparameters for the depicted implementation can be found in the *Experiments and Results* section and Adam [42] was chosen as the optimizer for the same reasons as before.

Bicycle GAN Postprocessing

When treating the refinement of the intermediate WAE result as a domain adaptation problem, Cycle GAN approaches or usage of the UNIT [107] and MUNIT framework [48] were the most common implemented solutions. This work chose Bicycle GAN [92] over the MUNIT framework, since paired data from both 'domains' was available by construction of the image pipeline with the WAE as the first stage and MUNIT was explicitly designed for the unpaired data setting [48].



(a) Scheme of the first cycle: (c)VAE-GAN.

(b) Scheme of the second cycle: (c)LR-GAN.

Figure 14: Schematic visualization of the two cycles of Bicycle GAN. In the first step, which is termed conditional (c)VAE-GAN [92], the ResNet-18 Encoder embeds the original image $\boldsymbol{X}$ and the U-Net domain adaptation network uses the embeddings as additional input. At the same time, the encoder and thus the style encoding are optimized to match a Gaussian distribution. The second step is termed conditional latent regressor (c)LR-GAN [92] and generates domain adaptations of the WAE output $\boldsymbol{X}_{gen.}$ with help of Gaussian random style vectors. The obtained domain adapted result is re-encoded to compute the ℓ_1 loss on the style vector input. For architectural details of Generator, Discriminator and Encoder, please have a look into Zhu et al. [92].

Beyond adaptation of the architectures' input and output dimensions as well as exchanging the hyperbolic tangent output with a sigmoid layer, no additional changes were applied to the architecture. Both variations were necessary to make the network suit the HSI data and to work within the same image value range $\boldsymbol{X}_{i,j,k} \in [0,1]$ as for the other approaches. For brevity, this section focuses on the concept behind Bicycle GAN, depicted in Figure 14. The mostly unmodified architectural details are explained in Zhu et al. [92].

Central concepts of the work of Zhu et al. [92] were multimodal outputs by incorporating additional style noise $\vec{z}_s$ in a VAE-like manner as well as cycle-consistency to encourage both diverse but also invertible and thus related results. As Figure 14 shows, the image $\boldsymbol{X}_{gen.} \sim P_{\text{WAE}}$ from the input domain together with an additional style vector $\vec{z}_s$ is fed into the generator G, which transfers domains and outputs the image $\boldsymbol{X}' = G(\boldsymbol{X}_{gen.}, \vec{z}_s)$. In similarity to VAEs, the first of the two cycles contained encoding the style vector $\vec{z}_s$ from image $\boldsymbol{X} \sim P_X$ with encoder E and restricted it by means of the KL divergence to a standard normal distribution $P_Z \sim \mathcal{N}_8(0,1)$, to learn a structured feature space. Furthermore, a PatchGAN discriminator was incorporated into the cycle to encourage realism beyond a simple absolute error loss function. The first half of Bicycle GAN's loss term coming from the VAE-GAN construction thus reads

$$\begin{aligned}\mathcal{L}_{\text{VAE-GAN}}(D,G,E,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s) =&\mathcal{L}_{GAN}(D,G,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s)\\ &+\lambda_{abs.}\cdot\mathcal{L}_1(\boldsymbol{X},G(\boldsymbol{X}_{gen.},\vec{z}_s))\\ &+\lambda_{KL}\cdot\mathbb{E}_{\boldsymbol{X}\sim P_X}[D_{KL}(E(\boldsymbol{X})|P_Z)]\end{aligned} \tag{24}$$

with KL divergence D_{KL}. To complete the second cycle of the Bicycle GAN, an image from the input domain with sampled style code $\vec{z}_s \sim P_Z$ was generated and the encoder afterwards tried to recover this style code - beneath the same PatchGAN discriminator for overall quality judgement. This gives the second part of the loss term

$$\begin{aligned}\mathcal{L}_{\text{cLR-GAN}}(D,G,E,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s) =&\mathcal{L}_{GAN}(D,G,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s)\\ &+\lambda_{\text{latent}}\cdot\mathcal{L}_1(Z,E(G(\boldsymbol{X}_{gen.},\vec{z}_s))\end{aligned} \tag{25}$$

where an absolute loss is used to regress the initially used feature code. For completeness, the optimizable value function combined from both terms reads

$$\begin{aligned}\min_{G,E}\max_{D} V(D,G,E) =&\mathcal{L}_{\text{VAE-GAN}}(D,G,E,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s)\\ &+\mathcal{L}_{\text{cLR-GAN}}(D,G,E,\boldsymbol{X},\boldsymbol{X}_{gen.},\vec{z}_s).\end{aligned} \tag{26}$$

Zhu et al. [92] used standard Adam [42] as the optimizer with a batch size of 1 and a style code dimension of 8. It further used $\lambda_{abs.} = 10$, $\lambda_{latent} = 0.5$ and $\lambda_{KL} = 0.05$ as hyperparameter values, which served as orientation for the hyperparameter search space of the loss weights. As before, the optimizer parameters were customized and the batch size was like the three loss weights λ_i part of the hyperparameter search.

3. Image Quality Assessment

Sophisticated ways of quantifying the realism of generated data are central for robust decisions on the constitution of model results. Accordingly, this section presents several metrics, including calculation details, which were mentioned in *Related Work* and are used for the assessment of the synthesized HSI results. According to the required data, the section is split into full reference, no reference and feature-based metrics.

3.1. Full Reference Metrics

The first category of so-called full reference metrics is most demanding in regard to the involved data: Synthetic samples, which are to be inspected, need to be matched one-on-one with original data. Except for the DISTS score, this not only refers to a comparison of two images but also to required same contents of said images to compute meaningful results, which becomes more clear with the first example in the next paragraph.

While being a very basic assessment, **Median Spectra** reveal especially for HSI important information, which otherwise goes unnoticed by the human eye. The median is calculated on the intensities for each wavelength to obtain the median reflectance along the spectral axis. This allows for a precise assessment of spectral correctness, which is otherwise not possible with the human eye. To receive comparable spectral results, the median spectra have to be acquired from images with the same overall scene content, which is easiest obtained from paired image data of original and corresponding HSI patches.
By construction, median spectra are robust to outliers and hence a good way of quantification with special focus on the spectral component, which HSI aims to improve. In the *Experiments and Results* section, the median spectra are often ℓ_1-normalized to allow for better comparison of different scene illumination. Computing differences between median spectra then can serve as spectral quality metric.

Mean Squared Error (MSE) and **Peak Signal to Noise Ratio** (PSNR) are further examples for computation-wise simple metrics, which deliver easy to interpret values of the overall scene agreement (MSE) and scene sharpness (PSNR). As the name implies, MSE is calculated as

$$\mathrm{MSE}(\boldsymbol{X}, \boldsymbol{Y}) = \sum_{a=1}^{\alpha} \dots \sum_{z=1}^{\omega} ||\boldsymbol{X}_{a,\dots,z} - \boldsymbol{Y}_{a,\dots,z}||_2^2 / \left(\alpha \cdot \dots \cdot \omega\right) \tag{27}$$

the entry-wise meaned, squared difference of an in this case z-dimensional tensor with individual sizes of $\alpha, \dots, \omega$. This allows evaluation of the overall scene quality and does not lay focus on individual image proportions. The MSE is also required for the PSNR, since it compares the maximum ('peak') of the image against the squared difference to the original and thus quantifies the signal quality or sharpness. PSNR is defined as

$$\mathrm{PSNR}(\boldsymbol{X}, \boldsymbol{Y}) = 10 \cdot \log \frac{I_{max.}^2}{\mathrm{MSE}(\boldsymbol{X}, \boldsymbol{Y})} \mathrm{dB} = (2 \cdot \log I_{max.} - \log \mathrm{MSE}(\boldsymbol{X}, \boldsymbol{Y})) \cdot 10 \text{ dB} \tag{28}$$

which simplifies to

$$\mathrm{PSNR}(\boldsymbol{X}, \boldsymbol{Y}) = -\log \mathrm{MSE}(\boldsymbol{X}, \boldsymbol{Y}) \cdot 10 \text{ dB} \tag{29}$$

for the used data range of $\boldsymbol{X}_{i,j,k} \in [0, 1]$ in the given case with $I_{max.} = 1$.

Structural Similarity (SSIM) was introduced by Wang et al. [79] and combines luminance $l(\cdot, \cdot)$, contrast $c(\cdot, \cdot)$ and structure $s(\cdot, \cdot)$ measures. It is generally defined as

$$\mathrm{SSIM}(\boldsymbol{X}, \boldsymbol{Y}) = l^{\alpha}(\boldsymbol{X}, \boldsymbol{Y}) \cdot c^{\beta}(\boldsymbol{X}, \boldsymbol{Y}) \cdot s^{\gamma}(\boldsymbol{X}, \boldsymbol{Y}). \tag{30}$$

However, its most well-known form uses the parameters $\alpha = \beta = \gamma = 1$, which simplifies to

$$\mathrm{SSIM}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{2\mu_X \mu_Y + c_1}{\mu_X^2 + \mu_Y^2 + c_1} \frac{2\sigma_{XY} + c_2}{\sigma_X^2 + \sigma_Y^2 + c_2}. \tag{31}$$

Additional constants c_i are added for numerical stability and the μ_i and σ_{ij} describe the corresponding (localized) means, variances and covariances [79, 108].

DISTS is probably the least known of the so far presented metrics and also was proposed the most recent [83]. Although it uses trained VGG network features, the classification into full reference metrics of the PyTorch Image Quality package [108] was adopted, since the DISTS metric does not compare whole datasets but individual images and the used VGG network features are put together in a handcrafted way. It is inspired by SSIM and combines luminance l and structure term s for different pretrained VGG terms in a weighted way [83]. The resulting distance measure is the square root $d(\cdot,\cdot) = \sqrt{D(\cdot,\cdot)}$ of

$$D(\boldsymbol{X}, \boldsymbol{Y}; \alpha, \beta) = 1 - \sum_{i=0}^{m} \sum_{j=1}^{n_i} \left(\alpha_{ij} l(\Phi_j^{(i)}(\boldsymbol{X}), \Phi_j^{(i)}(\boldsymbol{Y})) + \beta_{ij} s(\Phi_j^{(i)}(\boldsymbol{X}), \Phi_j^{(i)}(\boldsymbol{Y})) \right) \tag{32}$$

with individual VGG layer Φ, output luminance l and structure s weights α and β of layer i and channel j. DISTS as well as SSIM are indeed metrics in the mathematical sense, fulfilling non-negativity, symmetry and the triangle inequality.

3.2. No Reference Metrics

No reference metrics aim to provide image quantification grounded on solely natural scene statistics, which makes them in contrast to previous full reference metrics free of paired data for comparison.

To achieve this goal, **BRISQUE** [82] first calculates natural statistics of the channel-wise image intensity $\boldsymbol{I}(i,j)$ at position (i,j) via mean subtraction and normalization like

$$\hat{\boldsymbol{I}}(i,j) = \frac{\boldsymbol{I}(i,j) - \mu(i,j)}{\sigma(i,j) + C}, \tag{33}$$

with spatial indices i and j and a small constant C for numerical stability. $\mu(i,j)$ and $\sigma(i,j)$ are localized mean and standard deviation [82]. The in this way obtained, so-called mean subtracted contrast normalized (MSCN) coefficients $\hat{\boldsymbol{I}}(i,j)$ are supposed to follow characteristic statistical properties, because pairwise products of the MSCNs were shown to follow a certain distribution in absence of distortion [82]. To predict the naturalness, Mittal et al. [82] fitted the received distribution for computational purposes with an asymmetric generalized Gaussian and extracted the fit parameters. Evaluation of the fit parameters using a learned support vector machine regressor, for which the present work used a trained version contained in the PyTorch Image Quality toolkit [108], then yielded the final score. In this way, obtainable scores range from 0-100 with lower scores characterizing better results.

3.3. Feature-Based Metrics

Last mentioned but probably most often used [2, 75, 48, 84] are feature-based metrics, which utilize feature responses of pretrained neural networks. While this introduces the recognition of intricate features and has the possibility to reach far beyond what simple handcrafted metrics like PSNR and SSIM are able to recognize, it is obviously also crucially dependent on the pretraining of the network from which the features are extracted [89].

All of the following metrics commonly use a pretrained Inception network [87]; however, only the

Inception Score (IS) [90] is able to operate with unpaired data. The Inception v3 architecture from PyTorch Image Models [99] was retrained from scratch to suit the medical HSI data, as suggested in literature [89]. Following Barratt et al. [89], the splits parameter for the score was omitted such that the results were reported for the overall image distribution. Additionally, the IS without exponentiation was also reported, which is equivalent to the mutual information (MI) of individual label and overall label distribution [89]. The calculation

$$\mathrm{IS}(\boldsymbol{X}) = \exp\left(\mathrm{MI}(\boldsymbol{X})\right) = \exp\left(\mathbb{E}_{\boldsymbol{X}}[\mathrm{KL}(p(\vec{y}|\boldsymbol{X})|p(\vec{y}))]\right) \tag{34}$$

returns the results for both MI and IS with KL divergence and input HSI patch $\boldsymbol{X}$ from either real or synthetic domain. $p(\vec{y}|\boldsymbol{X})$ is the label distribution obtained from the pretrained Inception network, while $p(\vec{y})$ is the overall label distribution.

For paired feature-based assessment, two more metrics allow to compare data distributions. Heusel et al. [85] introduced the **Fréchet Inception Distance**, which works with preactivations of the previously mentioned Inception architecture and returns a (biased) estimate of their similarity. The distance is obtained by computing

$$d^2((\vec{\mu}_X, \boldsymbol{C}_X), (\vec{\mu}_G, \boldsymbol{C}_G)) = ||\vec{\mu}_X - \vec{\mu}_G||_2^2 + \mathrm{tr}\left(\boldsymbol{C}_X + \boldsymbol{C}_G - 2(\boldsymbol{C}_X \cdot \boldsymbol{C}_G)^{1/2}\right) \tag{35}$$

with mean μ_i along the sample axis of the received individual activations and their covariance matrices $\boldsymbol{C}_i$. The overall concept is based on the assumption that the preactivations follow a Gaussian. Since the in this way obtained comparison of supposed Gaussians is a biased estimate [84], the FID has to be computed on the same sample amount across different models for a meaningful comparison, which is in literature conventionally chosen to be 50.000 samples.

As a solution to the issues of the FID, the **Kernel Inception Distance** [86] was proposed, which computes the maximum mean discrepancy (MMD) of the preactivations and allows for an unbiased estimate. It is implemented in the same way as proposed in the original MMD publication [98] and uses a polynomial kernel $k(\cdot,\cdot)$ of third degree

$$k(\vec{x}, \vec{y}) = \left(\frac{1}{d} \cdot \vec{x}^T \vec{y} + 1\right)^3 \tag{36}$$

with an additional dimensional scaling parameter d, which refers to the preactivations' size $d = 2048$. Although there is also criticism in literature regarding the convergence of the KID to its true value for low sample sizes [109], it overall is expected to be more consistent than the FID and well-interpretable, which also held true within this work.

Part IV. Experiments and Results

The upcoming experiments explore and evaluate results of the HSI generation pipeline and apply generated patches from the developed pipeline in a real-world test case of the *Downstream Task: Image Segmentation*. Subdivision of this part into *Overall Training Details* and experiments with corresponding results provides all training and tuning details of the *Image Generation Pipeline* in one place and afterwards assesses the research questions in dedicated subsections. The experimental sections subsequent to the implementation details chronologically present results for the...

... *Imaging Effect Analysis*: Generated images are visualized as RGB images for qualitative discussion. The focus within this experiment lies on structures such as vessels and instruments as well as imaging effects like specular highlights and shadows.

... *Spectral Features*: Spectral consistency of the generated hyperspectral image patches against the real image patches is tested, which allows for both qualitative and quantitative assessment of spectral and thus physiological correctness.

... *Embedding Analysis*: The latent space of the WAE is evaluated to aid explainability, check for confounders and provide insights into generalization capabilities of the implemented deep learning method.

... *Texture Analysis*: Image quality metrics are used to calculate quantitative scores of generated textures. This allows both quantitative conclusions on texture quality as well as comparison to state of the art results.

... *Downstream Task: Image Segmentation*: Reconstructed patches from the image generation pipeline are used as additional data for an image segmentation task. This experiment is operating in a limited data setting where the training data is reduced to roughly a quarter of the overall training set, which allows to test the ability of synthetic patches from the image generation pipeline to solve the artificially created data bottleneck.

1. Overall Training Details

Before heading into the experiments, all implementation details of the four used deep learning networks are given in the upcoming section to avoid redundancy and be able to focus the experiment descriptions to analysis-relevant parts.

Global Data Loading and Network Initialization

After many early experiments with a lower patch resolution of $32 \times 32 \times 100$, all final experiments used a higher spatial resolution of $64 \times 64 \times 100$ to improve discriminability [50], which per epoch roughly took 50% longer due to the additional high-resolution layers. All models were trained with around 8192 (2^{13}) samples per epoch, which came from the dataloader already described in *Datasets, Preprocessing and Data Loading*. Augmentations on full images were already incorporated in the dataloader, before randomly cropping a patch of the desired resolution. The selected augmentations for all training setups were implemented using Albumentation [95] and comprised shifting, scaling, rotation and flipping of the images, since they did not influence physiological meaning beyond actually observable results.

Further, all models were initialized with Kaiming normal initialization and fixed random seeds, which was specifically proposed for networks with rectifier non-linearities [110] with the non-linearity parameter selected according to the respective activation function [106]. If not mentioned otherwise, training was done on the in *Datasets, Preprocessing and Data Loading* described training part of the dataset, which consisted of images from 12 pigs. The validation part was used for the hyperparameter optimizations and visual RGB results during training, which determined stopping. The test subset stayed untouched until the evaluations, carried out in the upcoming sections.

Hardware Report

A NVIDIA RTX 2070 TI with 11.7 GB of VRAM together with an Intel® Xeon® E5-1620 CPU was used during hyperparameter optimization and most of the training. For multiple runs on a GPU cluster, either a NVIDIA RTX 2080 with 10.7 G VRAM or a TitanX with 11.9 G VRAM together with an Intel® Xeon® E5-2620 CPU were utilized.

Wasserstein Autoencoder

For the WAE with final resolution of $64 \times 64 \times 100$, more than 15 hyperparameter searches with at least 20 parameter trials each were carried out. Automated hyperparameter searches were conducted with the *Ray Tune* [111] framework during model architecture optimization. The optimization was focused on the selection of the best results after 15 epochs.
Previous results of architecture-wise different models, which also contained label-conditional models, as well as models with lower image resolution both have found similar best hyperparameters such as an encoder learning rate of approximately 10^{-5}, a decoder learning rate larger than the encoder learning rate by a factor of 10 to 20, batch sizes around 16 to 64 and lower than default β's for the Adam optimizer. Furthermore, a latent space dimension of 1024 was shown to deliver good results during previous searches and experiments. Therefore, the hyperparameter search was carried out with the latent dimensionality fixed to this value to

reduce the amount of tunable hyperparameters. Since the embedding space dimensionality is crucial for the overall model results, it was reintroduced during the first two experiments, where different latent dimensionalities and their effects on spectral correctness as well as embedding consistency were compared.

Hyperparameter sampling for the optimizer parameters within *Ray Tune* was done from a uniform distribution, while the larger search intervals for learning rate and loss weights utilized the `loguniform` method to guarantee even distribution over several orders of magnitude.

Hyperparameter	Search Space	Optimized Value
$\text{lr}_{\text{encoder}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-3}]$	$9.15 \cdot 10^{-5}$
$\text{lr}_{\text{decoder}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-3}]$	$1.68 \cdot 10^{-4}$
λ	$\mathcal{U}[\log 10^{-2}$ - $\log 1]$	0.098
λ_{SSIM}	$\mathcal{U}[\log 10^{-2}$ - $\log 1]$	0.165
β_1	$\mathcal{U}[0.1 - 0.7]$	0.41
β_2	$\mathcal{U}[0.5 - 0.99]$	0.76
$\beta_{1,\text{discriminator}}$	$\mathcal{U}[0.1 - 0.7]$	0.19
$\beta_{2,\text{discriminator}}$	$\mathcal{U}[0.5 - 0.99]$	0.68

Table 2: Hyperparameter search space and optimized hyperparameters of Wasserstein Autoencoder.

Separately, smaller and bigger batch sizes were tested for training. While bigger batch sizes were able to increase the GPU utilization to its maximum, visual results during validation steps became worse and hyperparameter searches dedicated specifically to bigger batch sizes were unsuccessful in obtaining better results. On the opposite end, smaller batch sizes decreased GPU utilization and slowed training down with no improvement in loss or visual quality. Overall, results were worse when either low or high batch sizes were used, if judging fairly after an equal amount of optimization steps taken, which lead to the fixation of batch size to 32.

For the final version of the network presented here, 25 trials were carried out in the search space given in Table 2. Cosine annealing periodic learning rate scheduler [112] was used with parameters $T_0 = 1$, $T_{\text{mult.}} = 2$ and $\eta_{\text{min.}} = 10^{-6}$ for the Adam optimizer with previously mentioned custom $\beta's$ as optimizer [106]. The rationale behind the chosen learning rate scheduler parameters is its ability to leave local minima by means of the periodic learning rate, which leads to better optimization results. The learning rate λ_i gets decreased with a cosine scaling to $\eta_{\text{min.}}$ from epoch $T_0 = 1$ until the next epoch of the learning rate cycle. After completion of one learning rate cycle, the learning rate is increased to the default value, which is equal to a 'warm restart' of the neural network with initial high learning rate but already tuned parameters. Doubling the period of the learning rate cycle with the parameter $T_{\text{mult.}}$ increases the training cycle duration to optimize network results over elongated periods in the long run.

The WAEs described in the next sections were all trained for 500 epochs or roughly 100.000 training steps similar to the original publication by Tolstikhin et al. [67]. Assessed by validation loss, this training duration showed decent results with no artefacts or signs of overfitting. Additional observations substantiating this choice can be found in the *Training Result: WAE Exhaustive Training* section in the appendix. This section contains longer, exhaustive training results for up to 1750 epochs.

pix2pix

Similar to the WAE optimization, several hyperparameter searches with at least 20 parameter trials each were conducted. Again, early results from within the first 10 epochs were compared. For the *pix2pix* network only learning rate and β_1 parameter of the used Adam optimizer were optimized, since both implementations which were used as source [18, 50] used the default $\beta_2 = 0.999$ parameter. In contrast to literature and motivated by hyperparameter search results, a larger batch size than in both publications [18, 50] was chosen.

The final network was trained for 400 epochs and used the cosine annealing learning rate scheduler with the same parameters $T_0 = 1$, $T_{\text{mult.}} = 2$ as well as $\eta_{\text{min.}} = 10^{-6}$ such as already implemented in the WAE. Since the ℓ_1 loss was already low from the optimized WAE, the small loss weights were justified to receive similar contributions to the total loss. Overall, the search space was more narrow when compared to the WAE search space, which granted to achieve finer parameter space coverage after initial searches on a larger search space.

Hyperparameter	Search Space	Optimized Value
$\text{lr}_{\text{generator}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-4}]$	$1.94 \cdot 10^{-5}$
$\text{lr}_{\text{discriminator}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-4}$	$3.43 \cdot 10^{-6}$
λ_{VGG}	$\mathcal{U}[\log 10^{-4}$ - $\log 10^{-1}]$	0.00018
λ_{GAN}	$\mathcal{U}[\log 10^{-3}$ - $\log 10^{-1}]$	0.0079
batch size	4 - 64	8
β_1	$\mathcal{U}[0.1 - 0.7]$	0.66
$\beta_{1,\text{discriminator}}$	$\mathcal{U}[0.1 - 0.7]$	0.70

Table 3: Hyperparameter search space and optimized hyperparameters of the *pix2pix* framework.

Bicycle GAN

As for the *pix2pix* approach, Adam's default $\beta_2 = 0.999$ was kept and the batch size increased in agreement with hyperparameter search results, in contrast to the original implementation of Zhu et al. [92]. The search space for the loss weights was chosen in accordance with original loss weights, which were mentioned in the *Bicycle GAN Postprocessing* section, due to fact, that the Bicycle GAN architecture was mostly left default. No further learning rate scheduling was applied and the overall search space was narrowed for the same reasons as for the *pix2pix* approach. The final network was trained for 300 epochs, in which the validation loss mostly stayed constant or even slightly increased, while the visual results kept improving.

Hyperparameter	Search Space	Optimized Value
$\text{lr}_{\text{generator}} = \text{lr}_{\text{encoder}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-4}]$	$3.27 \cdot 10^{-5}$
$\text{lr}_{\text{discriminators}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-4}]$	$1.33 \cdot 10^{-6}$
$\lambda_{abs.}$	$\mathcal{U}[\log 1$ - $\log 100]$	23
λ_{latent}	$\mathcal{U}[\log 10^{-1}$ - $\log 10]$	0.74
λ_{KL}	$\mathcal{U}[\log 10^{-3}$ - $\log 10^{-1}]$	0.012
batch size	4 - 32	8
$\beta_{1,\text{gen.}} = \beta_{1,\text{enc.}}$	$\mathcal{U}[0.2 - 0.7]$	0.55
$\beta_{1,\text{discriminators}}$	$\mathcal{U}[0.2 - 0.7]$	0.58

Table 4: Hyperparameter search space and optimized hyperparameters of Bicyle GAN.

Inception v3

The Inception v3 network [87] was required [89] for computation of Inception Score, Fréchet Inception Distance and Kernel Inception Distance. Therefore, several hyperparameter searches were launched to obtain a well trained Inception v3 network for the custom HSI dataset. Received hyperparameter results are presented in Table 5.

34 organ classes were contained in the dataset on which the customized Inception v3 network was trained, as opposed by 1.000 classes for the original implementation [87]. The model was loaded with pretrained ImageNet weights from the PyTorch Image Models library [99]. To obtain labels to train on, the individual organ coverage percentages were calculated from semantic segmentation maps which corresponded to training patches. The obtained percentages of the contained organs were used as label vector, such that the Inception v3 network aims to predict the proportions of organs present in the input HSI patch. For more details on organ distribution, please consult Seidlitz and Sellner et al [4].

Hyperparameter	Search Space	Optimized Value
$\text{lr}_{\text{discriminator}}$	$\mathcal{U}[\log 10^{-6}$ - $\log 10^{-4}]$	$2.87 \cdot 10^{-5}$
batch size	8 - 96	96
β_1	$\mathcal{U}[0.1 - 0.7]$	0.25
β_2	$\mathcal{U}[0.5 - 0.99]$	0.74

Table 5: Hyperparameter search space and optimized hyperparameters of Inception v3 network.

2. Imaging Effect Analysis

The imaging effects analysis section covers results concerning qualitative realism in terms of imaging effects like specular highlights and shadows, as well as physiological attributes such as blood vessels and organ borders.

Experimental Design

For this purpose, several reconstruction and postprocessed images were synthesized and a manual preselection was made. The preselection focuses on choosing image patches which allow to answer the research question, *'whether the proposed deep learning pipeline can generate hyperspectral image patches that look realistic in terms of imaging effects like specular highlights or shadows and physiological attributes such as blood vessels'*. At the same time, the selected samples tried to stay representative and present both positive and negative results. Additionally, multimodal postprocessing results of one image with several input styles for Bicycle GAN were computed. More encompassing image grids can be found in the *Imaging Effect Analysis: Additions* section in the appendix.

Method Details

Image reconstructions of randomly selected, real HSI patches from the test dataset were generated using the WAE with a latent dimension of 512. Afterwards, the WAE reconstructions were postprocessed with one of the two postprocessing networks. To be able to visualize the results, the HSI results were translated to RGB images according to the reimplementation of the manufacturer's conversion [4]. The obtained image grids were manually preselected to aggregate patches in a content-wise structured way, which outline important qualitative, visual results.

Example Cases

The first visual comparison in Figure 15 showcases scenes with globally similar illumination, containing spleen and skin. Reconstruction with the WAE and particularly postprocessing with both GAN types worked well in these cases, as the overall colour was correct and finer structures of the skin as well as specular highlights for the spleen were visible. Also, shadows could be recovered, which were caused by the organ structure. In addition, the generated HSI samples were able to meet the different organ class expectations in terms of imaging effects, as for the spleen samples many specular highlights were visible, while the skin reconstructions in Figure 15 correctly did not contain specular highlights.

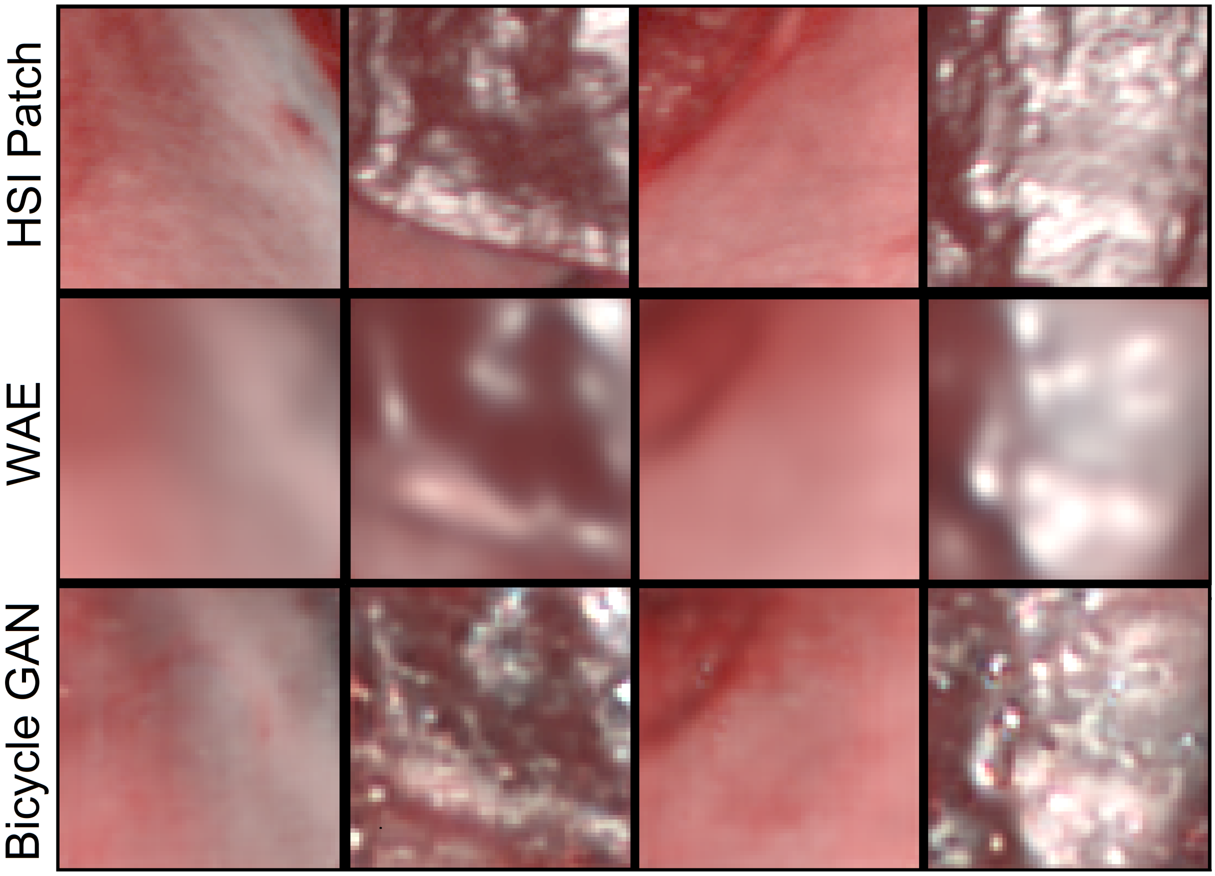


Figure 15: Reconstruction and postprocessing of RGB patches with WAE and Bicycle GAN. The top row depicts the to RGB converted patches from the test dataset, the middle row the WAE reconstructions and the bottom row shows the final postprocessed results. Overall, real image patches and postprocessing results look similar, with comparable colour and structure contents.

Since the focus of this experiment was the qualitative, visual realism, the postprocessing models were not directly compared as their results were perceivably similar. More visualizations for the *pix2pix* postprocessing can be found in the *Imaging Effect Analysis: Additions* section in the appendix.

Special Case: Heart When computing the results for certain organs with more intricate structure, such as the heart, larger visual differences were observed. Figure 16 displays three similar examples of patches containing the heart, their respective reconstruction and postprocessing with Bicycle GAN. In the presented cases, the WAE first pipeline step was not able to capture many high-frequency structural details, which also the Bicycle GAN was then not able to recover. Furthermore, the Bicycle GAN in these cases seemed to generate different amounts of specular highlights rather than recovering high-frequency structural details.

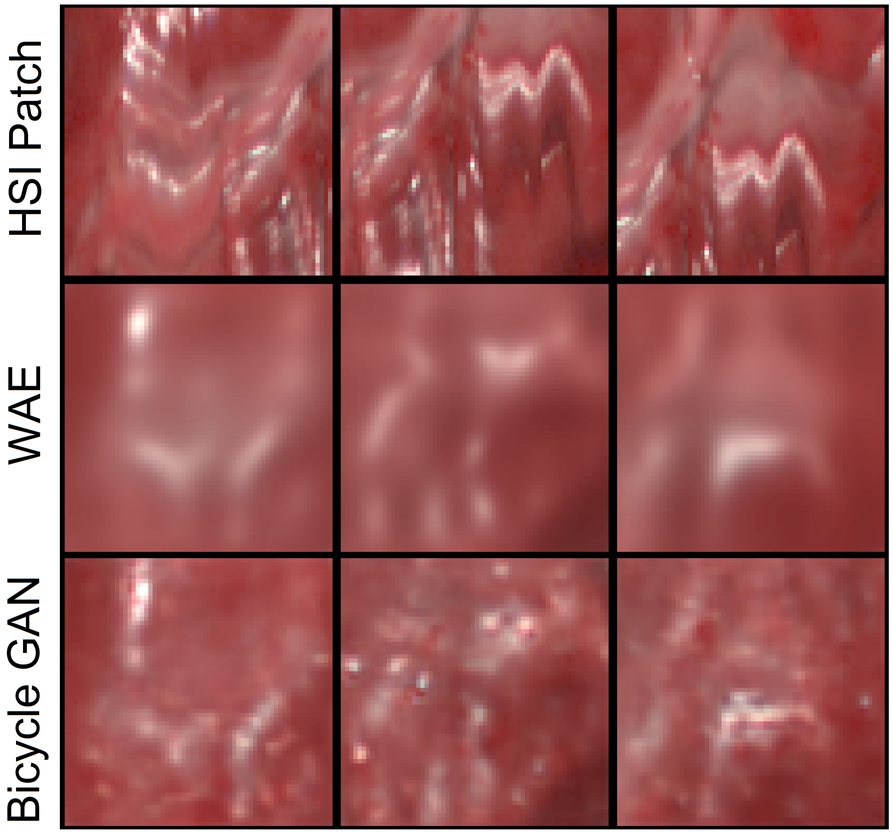


Figure 16: Reconstruction and postprocessing of RGB patches with WAE and Bicycle GAN. Both WAE and postprocessing are not able to depict the intricate high-frequency structure of the heart, which comes from tissue movement during the image acquisition time of roughly five seconds.

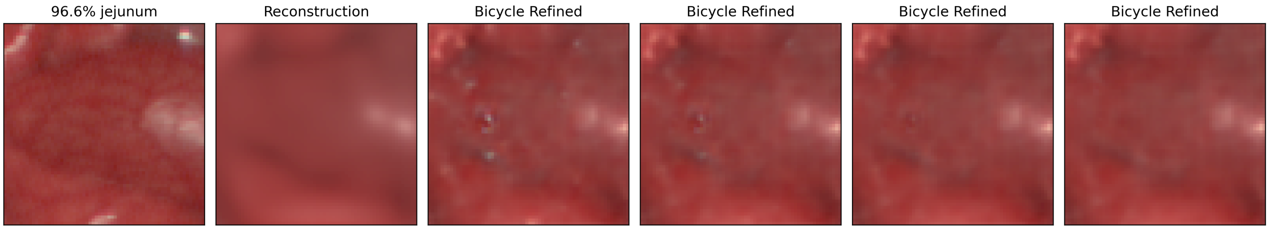


Figure 17: An RGB patch containing mostly jejunum is visible on the left. Next to it, the WAE reconstruction is depicted, followed by different postprocessed Bicycle GAN styles. Style vectors are linearly interpolated between the first and last style. The first style on the left contains three more additional specular highlights, which vanish when interpolating to the second style. Other image properties stay perceptionally unchanged.

Motivated by the previous patches of the heart, Figure 17 depicts the effect of different Bicycle GAN style inputs. Style inputs to the Bicycle GAN mostly affect specular highlights, while e.g. desired multimodality aspects such as lighting or physiological contents stay visually unchanged. An additional two-dimensional style interpolation grid, which highlights effects in style space, can be found in the *Imaging Effect Analysis: Additions* section in the appendix.

Special Case: Image Patterns A further observation that was made, is the missing of the finer, camera-specific patterns. Presented in Figure 18, shadows on the background and at organ borders appear to be recognized and reconstructed mostly correct. In contrast, the stripish-wavy camera pattern, especially well visible on the darker cloth background, looked rather edgy in the postprocessed image patches. Figure 18 additionally shows some of Bicycle GANs typical, additional specular highlights in columns four and five.

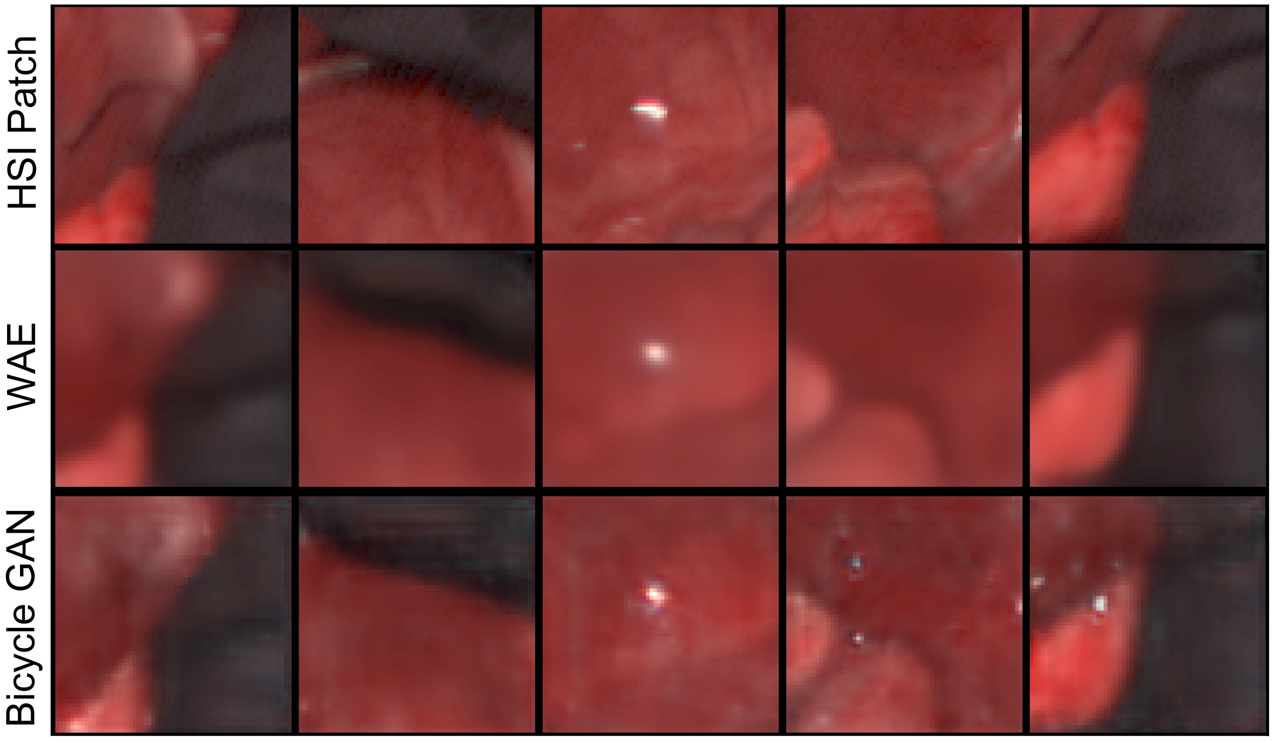


Figure 18: Camera-specific patterns, caused by the stepper motor, are best observable on top of the dark cloth-background in the top row. The middle row contains WAE reconstructions, the bottom row Bicycle GAN postprocessed patches. Image patterns are blurred out in the WAE results and not correctly reproduced by the postprocessing GANs.

Special Case: Vessels Missing details in the WAE reconstruction were found to also apply for finer physiological structure aspects like veins. Depicted in Figure 19, vessels are completely lost in the intermediate WAE reconstruction and also not clearly visible in the final postprocessed outcome. When examining vessels on internal organs, the postprocessed reconstructions often did not have a meaningful structure of veins but met the overall tone and lighting of the real image patch. Lack of meaningful structure as shown in Figure 19 refers to the colour variations which at first sight look similar to real vessels but at second glance are rather colour jitter than a physiological pattern. However, the created jitter still respects the physiological borders e.g. depicted by shadows.

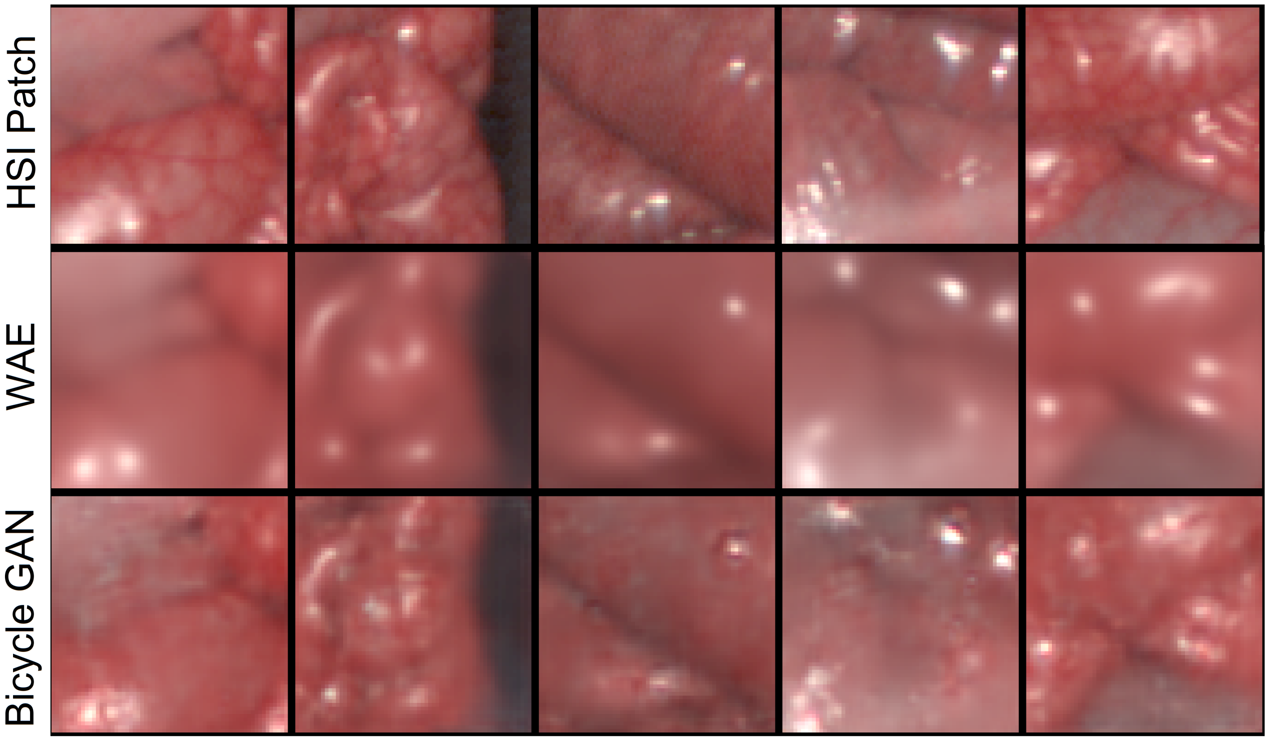


Figure 19: Reconstruction and postprocessing of RGB patches with WAE and Bicycle GAN. The top row depicts the RGB patches containing vessels on internal organs, the middle row displays the blurry WAE reconstructions and the bottom row shows the final postprocessed results. Recovered vessel structure in the bottom row is not mesh-like as in the top row but rather looks like colour jitter.

Special Case: Instruments Figure 20 also illustrates issues with the sharpness of the WAE reconstruction, in this instance visible for the appearance of surgical instruments and especially the correctness of their borders. As a positive example, the left reconstruction and especially the postprocessing was able to recover the instrument's shape and shine. The postprocessing was even able to retrieve instrument contamination with some red blood stains. In a scene with worse illumination, as presented in the right part of Figure 20; however, the results looked worse. While the reconstruction was globally correct, the instrument appeared much more noisy, the border between organ and instrument vanished in the WAE reconstruction and was hardly visible in the postprocessed patch. Still, the specular highlights on the instrument and tissue were correct and also detailed red blood stains from the surgical procedure were visible on the instrument.

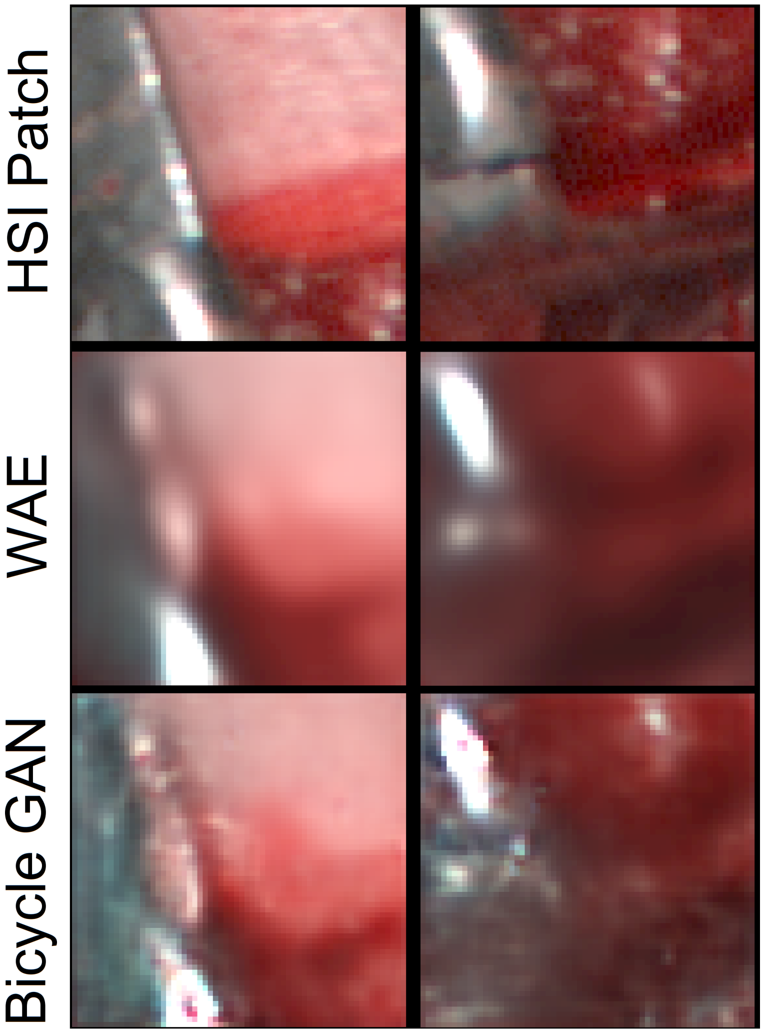


Figure 20: Reconstruction and postprocessing of RGB patches with WAE and Bicycle GAN. In the scene with better illumination on the left, the instrument is recovered clear and with correct gloss effects. In cases of worse illumination as on the right, the borders are blurred and the instrument loses some of its metallic gloss properties.

***pix2pix*: Checkerboard Patterns** Figure 21 demonstrates another issue, in this intensity only observable for the *pix2pix* postprocessing approach: Especially for background cloth, checkerboard patterns were observed.

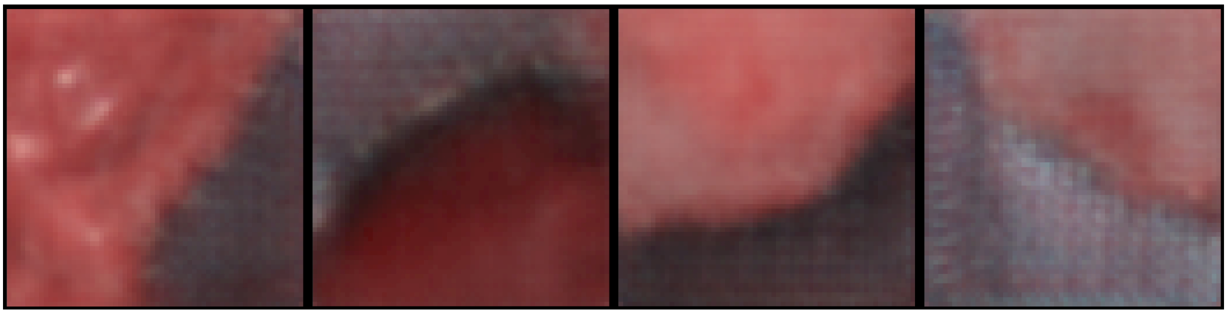


Figure 21: *pix2pix* specific checkerboard artefacts. While similar patterns are sometimes also weakly visible for Bicycle GAN results, they are more often and much more pronounced for the *pix2pix* approach, also in literature [18]. The occurrence is specifically strong over cloth in the background.

3. Spectral Features

To assess the spectral quality and *'whether the generated samples of the image generation pipeline feature pixel spectra similar to those extracted from real data'*, this experiment evaluates median spectra of WAE reconstructions as well as their postprocessed counterparts qualitative and quantitative. Furthermore, the latent dimensionality of the WAE is assessed in terms of spectral consistency.

Experimental Design

For an overall understanding of the spectral landscape, the median spectra of several image patches of one selected organ as well as examples from the overall latent space manifold were visualized with *Principal Component Analysis* (PCA).

Afterwards, changes in the crucial latent space dimension of the WAE and their effect on spectral consistency were computed. The latent space dimensionality is a central hyperparameter, since it regulates the correctness of image details by the way and the amount of features, which are encoded. Spectral results were computed in form of organ-weighted median spectra for physiological correctness. On the qualitative side, the organ-weighted median spectra of randomly sampled patches were presented for the different latent dimensionalities, to illustrate quantitative findings.

To make sure that the postprocessed pipeline results did not lose physiological accuracy, also the differences between median spectra of real HSI patches and the final postprocessed pipeline results were computed and the same example spectra as for the WAE were plotted.

Method Details

For a first intuitive visualization, PCA was used to compare median spectra of real image patches and the results obtained from the image pipeline. At first, 200 image patches were collected from the HSI test dataset, with restriction to patches that contained one specific single organ in at least 60% of the pixel content. For the three types of real HSI patch, reconstruction and postprocessed patch, the organ-weighted median spectra were computed. The organ-weighted median spectrum $\tilde{\vec{x}}_w$ of image patch $\boldsymbol{X}$ was calculated by

$$\tilde{\vec{x}}_w = \sum_{i\in\{\text{organs}\}} f_i \cdot \tilde{\vec{x}}_i, \tag{37}$$

where f_i is the pixel fraction covered by the organ i in the HSI patch and $\tilde{\vec{x}}$ is the median of each of the 100 spatial planes. $\tilde{\vec{x}}_i$ restricted the median to the pixels of one organ label i, meaning that there is one median spectrum for each organ in the HSI patch. Organ-wise median spectrum extraction served the purpose to preserve physiological meaning on larger patches which often contained several organs. Comparison of median spectra computed on full patches thus led to overall image statistic comparisons which do not leverage all available information and hence results in a weaker spectral quality assessment. Afterwards, PCA was applied to the spectra received from computations on the real HSI patches and reconstruction and postprocessed spectra were transformed accordingly.
Furthermore, a higher amount of 400 low discrepancy latent space samples [113] was reconstructed, post-processed and compared to 400 random samples, aggregated from the test dataset to compare overall distribution. Due to missing semantic segmentation data for the reconstructions from latent space, overall median spectra were calculated for all randomly selected patches. Similar to the single-organ PCA visualization, PCA was applied to the gathered real patch median spectra and the reconstruction and postprocessed spectra were transformed accordingly.

The first quantitative method (beyond the validation loss) used for the hyperparameter selection was the comparison of organ-coverage weighted median spectra. For the comparison, the KL-divergence and the Wasserstein or Earth Mover's distance (EMD), referred to as d_k for $k \in \{\text{KL, EMD}\}$, were utilized to compute the difference of ℓ_1-normalized, organ-wise median spectra. The difference on the whole patch D_k between the real HSI patch $\boldsymbol{X}$ and the synthetic patch $G(\vec{z})$ is obtained by calculating the organ-coverage weighted sum, according to the relative image fraction f_i of each organ i as in Equation 37. Writing it down yields

$$D_k(\boldsymbol{X}, G(\vec{z})) = \sum_{i\in\{\text{organs}\}} f_i \cdot d_k(\tilde{\vec{x}}_i, \tilde{G}(\vec{z})_i) \tag{38}$$

with $\tilde{\cdot}_i$ again referring to the organ-wise median.

The in Equation 38 defined way to measure spectral differences was used to receive the quantitative results of this experiment. Statistics were obtained by calculating the divergence on 1.000 HSI patches, received from the dataloader. To be precise, the HSI patches were sampled without augmentations from the test dataset, thus the spectral consistency for the real test data distribution was assessed. The sample size of 1.000 was kept, since increasing it to 5.000 samples did not improve the resulting standard deviation of the metric, which gave a first glimpse at inherent, large differences which will come up again in the *Downstream Task: Image Segmentation* and *Discussion and Conclusion* sections. As the computation of scores for 1.000 samples already took around 10 minutes and time scaled roughly linear with sample size, sample amounts as high as 50.000 for later feature-based *Texture Analysis* were timewise unfeasible.

On the qualitative side, eight image patches were randomly sampled from the dataloader and fed to the image pipeline for reconstruction and postprocessing. Afterwards their organ-coverage weighted median spectrum according to Equation 37 was calculated and also standard deviations for each wavelength were reported. The median spectra and standard deviations were ℓ_1 normalized before plotting, to make them illumination independent.

Results

PCA Visualization Results In between the quantitative results of thousands of samples and comparing the organ-weighted median spectra of single examples, PCA decomposition allowed to compare several images at once in an intuitive way. Figure 22 presents a kernel density plot of the PCA of organ-weighted median spectra from 200 HSI patches. Additional restriction for the selection of HSI patches was, that the individual pixels had to be labelled 'liver' more than 60% of the time and hence depicted a spectral manifold of one organ. As a reminder, the HSI patches for PCA were not sampled from the dataloader but iteratively extracted from the test images to ensure diversity. On top of the density, 20 randomly chosen HSI patches, their corresponding WAE reconstructions and *pix2pix* postprocessed results were shown to keep the visualization clear.

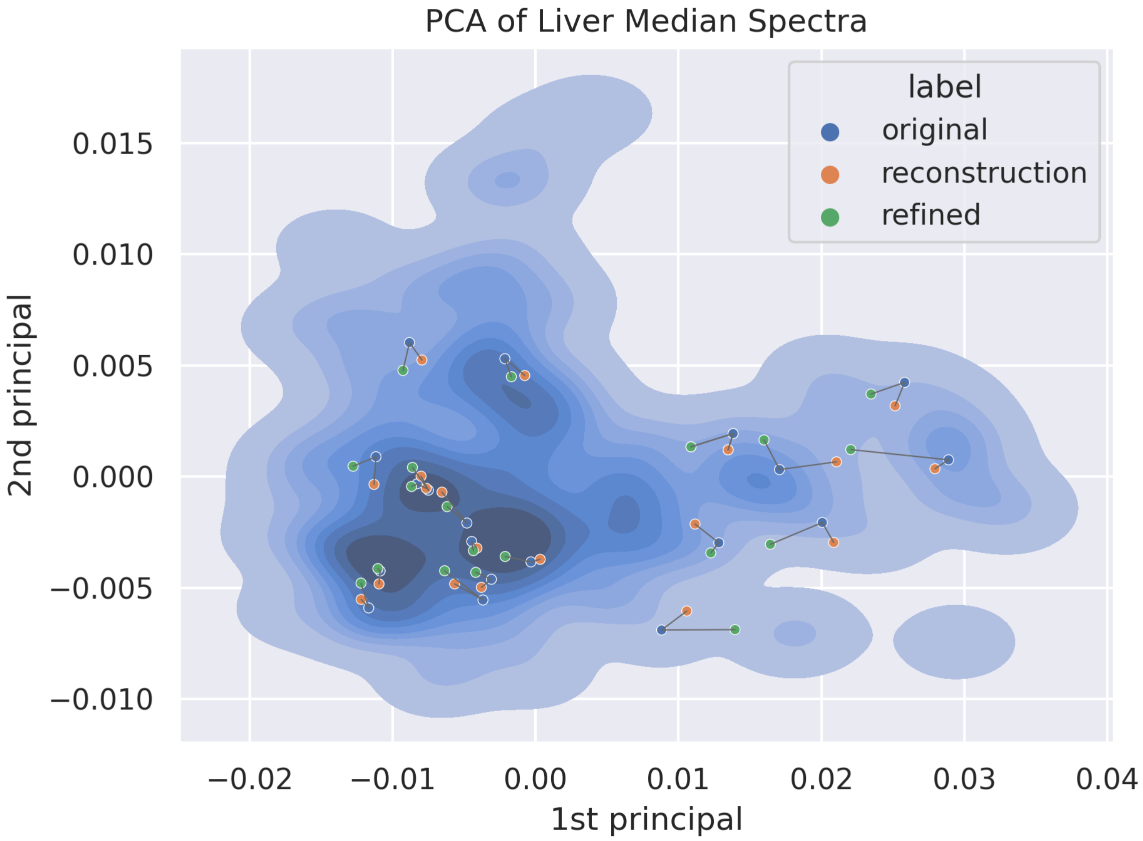


Figure 22: Kernel density plot of the first two PCA components of organ weighted medians, stemming from 200 HSI patches. Explained variance for the first two components of the real HSI patch data is 87.46%. Real HSI patch embeddings in blue, WAE reconstructions in orange and postprocessed results in green. Postprocessing for this plot was done with the *pix2pix* framework. Corresponding reconstructions and postprocessed PCA embeddings are attached to the real HSI embedding with grey lines. Reconstructions and refined results are in general similar distant to the original PCA result.

No specific tendency regarding distance or direction to real HSI patch PCA is seen in Figure 22 for the reconstructions or refined image results. Furthermore, all obtained results lay within the expected density. However, there seemed to be a tendency that results in less dense regions lay further apart.

Upcoming Figure 23 is more concerned with the overall image manifold rather than only one organ. 400 random HSI patches were cropped from test set images and 400 low-discrepancy samples from a Sobol sequence [113] were Box-Muller transformed [114, 115] and decoded to cover the latent space of the WAE evenly. The decoded latent vectors were also postprocessed. Figure 23 depicts the PCA of the median spectra with the density belonging to the test data, and 'refined' referring to the *pix2pix* model. Bicycle GAN postprocessings created a similar distribution, which is displayed in the *Spectral Features: Additions* section in the appendix.

The rough distribution of PCA embeddings was similar, but several real sample regions were not covered. Furthermore, the random reconstructed and postprocessed samples significantly extended beyond where real samples could be found in the top right of the density plot.

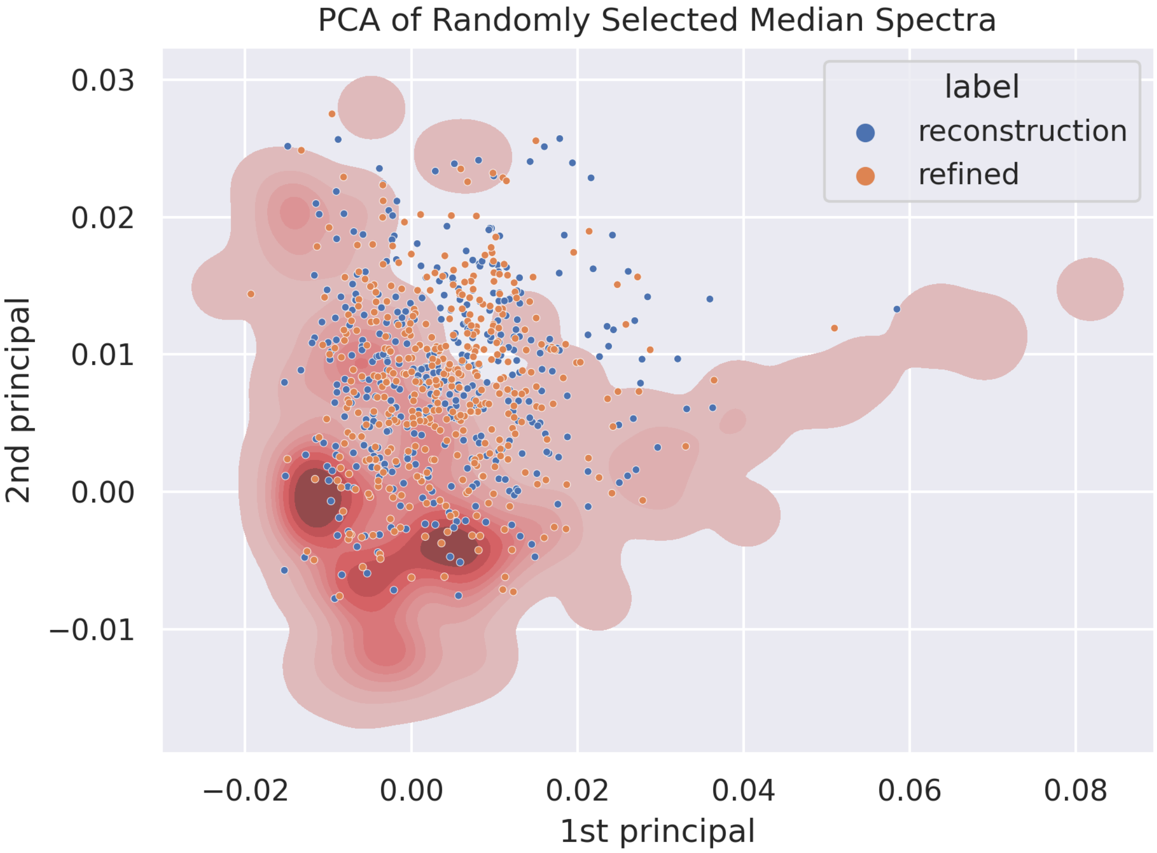
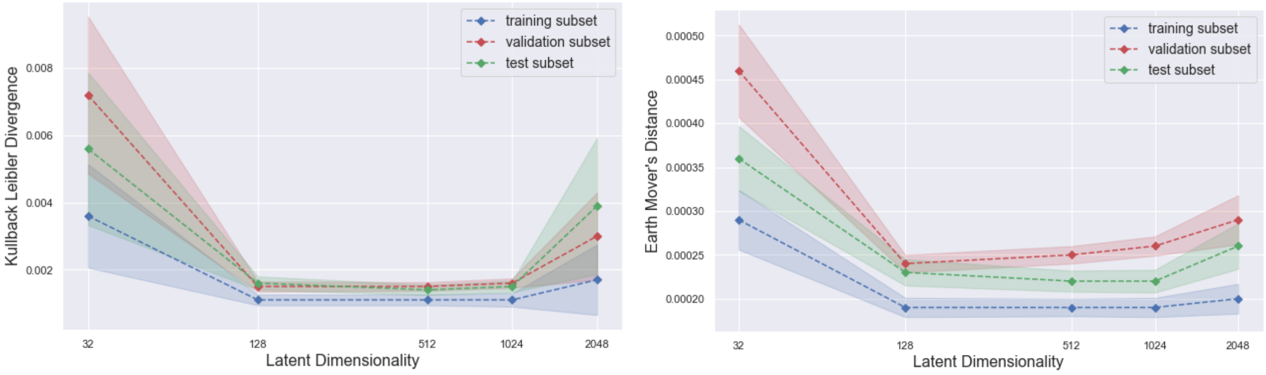


Figure 23: Kernel density plot of the first two PCA components of medians stemming 400 HSI patches. Explained variance for the first two components of the real HSI patch data is 88.65%. WAE reconstructions are displayed in blue and the postprocessed PCA embeddings in orange. Postprocessing for this plot was done with the *pix2pix* framework. A shift in distribution of the generated patch point cloud from latent space samples and corresponding refined *pix2pix* results from the test data, visualized as background density, is visible.

Quantitative WAE Results Figure 24 displays the KL- and EMD-based divergence measure for organ-weighted median spectra of 1.000 train, validation and test dataset samples and their corresponding WAE reconstructions across five different latent space dimensionalities. To better visualize the results, the standard deviation bands were reduced to one-tenth of their real size.

A first observation, which came as no surprise, was that the spectral difference for reconstructions on training data was lower than on validation and test data, which were roughly similar. However, all results lay within the large standard deviation of the organ-wise distance. Second, for both distances the results formed some kind of 'valley', where the latent dimensions 128, 512 and 1024 made up the trough.

For later interpretation it is important to keep in mind what both underlying divergences computed: Apart from the information theory interpretation, the KL divergence computed a distribution weighted, logarithmic difference between two distributions. The EMD measured the difference between two distributions by how much difference had to be transported how far, such that the distributions were equal.



(a) Organ-wise, ℓ_1-normalized median difference. The individual median spectrum divergence was calculated with the KL-divergence.

(b) Organ-wise, ℓ_1-normalized median difference. The individual median spectrum distance was calculated with the EMD (Wasserstein) distance.

Figure 24: Organ-weighted median spectra divergences with standard deviation bands. The standard deviations were divided by a factor of 10 for visualization purposes, which means that the results are not significantly different. Evaluation of training dataset HSI patches and their reconstructions returns the most spectrally consistent results. Intermediate large latent space dimensions (128-1024) show the lowest spectral discrepancy of original HSI patch and WAE reconstruction across training, validation and test dataset.

Qualitative WAE Results Figure 25 visualizes the median spectra differences by plotting organ-weighted median spectra with their standard deviations for real HSI patches from the test dataset and the corresponding WAE reconstructions with different latent dimensionalities.

Beyond a latent dimension of 128, the disagreement of different patches was hard to distinguish for the human eye. It is further notable, that while some organ spectra improved when the latent dimensionality was further increased, others diminished at the same time. An example for improvement is the 50.2% liver patch in the first column, a diminishing example the 100.0% colon patch in the last column of Figure 25.

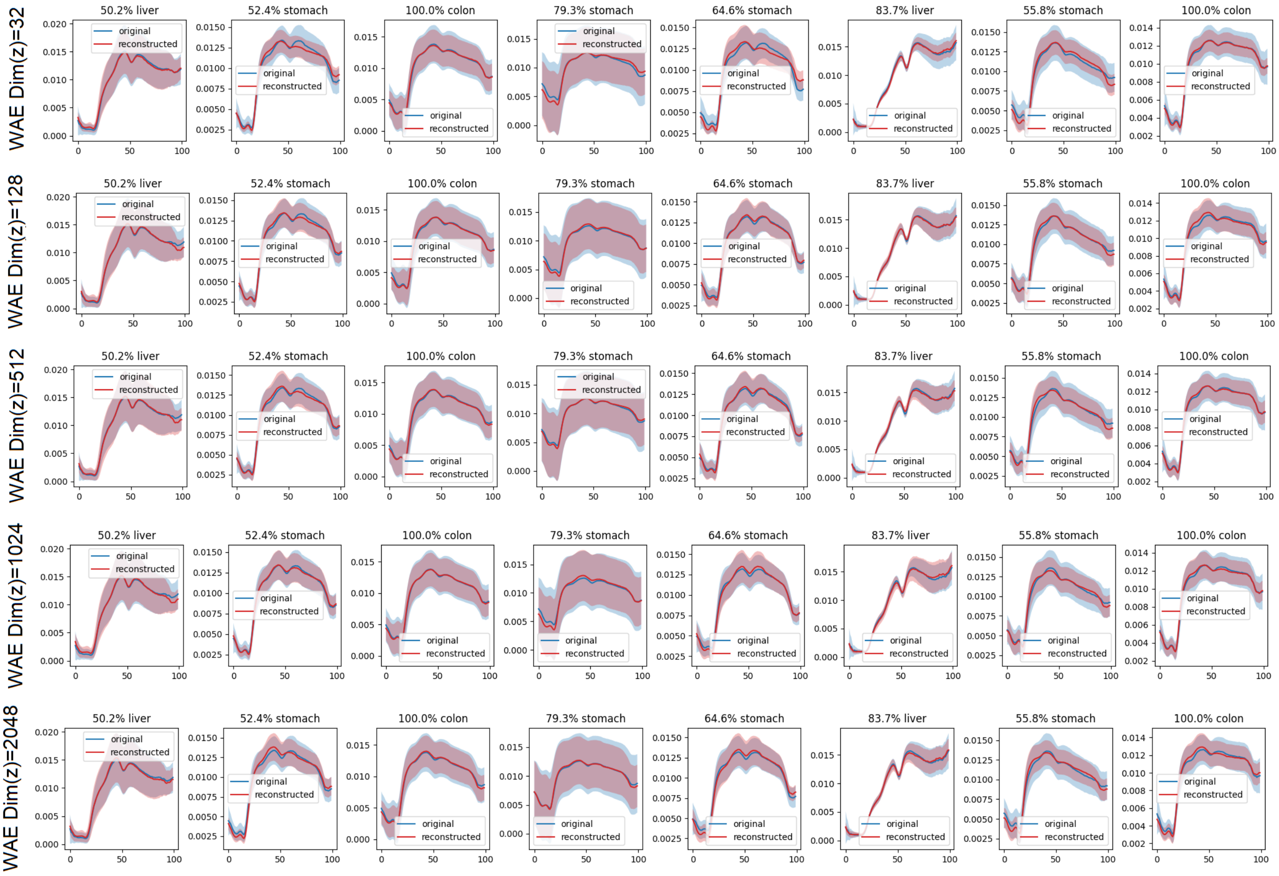


Figure 25: Example ℓ_1-normalized, organ-weighted median spectra for different latent dimensionalities. Spectral band 0 corresponds to a wavelength of 500 nm while band 100 corresponds to 1000 nm. Percentages in the plot headings give the coverage of the mentioned organ within the patch. Same patches across different latent dimensions are shown column-wise. Beyond a latent dimension of 128, no general further improvements are visible.

Post-Processing Results Due to the previous quantitative spectral, but also because of upcoming quantitative embedding results, the postprocessing GANs were trained with a WAE with latent dimension of 512. Table 6 compares the organ-weighted median divergence results for the postprocessing GANs with the previous WAE results, which were obtained from a WAE with latent dimension of 512 for patches from the test dataset.

The spectral consistency for both postprocessing approaches slightly decreased when compared to median spectra results, which were obtained from the first image generation pipeline stage. At the same time, the error of the spectral divergences also increased slightly. Due to the large error, the increase was not significant.

Model	Kullback-Leibler-Divergence	Earth-Mover's-Distance
WAE	0.0014 ± 0.0015	$(2.2 \pm 1.2) \cdot 10^{-4}$
pix2pix	0.0021 ± 0.0024	$(2.7 \pm 1.5) \cdot 10^{-4}$
Bicycle GAN	0.0018 ± 0.0023	$(2.6 \pm 1.5) \cdot 10^{-4}$

Table 6: ℓ_1-normalized median spectra divergences of 1.000 samples, coming from the test dataset and their reconstructed counterparts. Due to the large error, the increase is overall not significant for both applied divergence measures.

The qualitative, spectral comparison of the same example patches as for Figure 25 with postprocessing is shown in Figure 26. Additionally to the postprocessing GANs, the intermediate WAE results for latent dimension 512 are displayed to illustrate the observation of quantitative median spectra divergence increase.

Some median spectra were closer to the real HSI patch spectra than they were for the WAE; however, this was counterbalanced by fail cases, which in some wavelength ranges suddenly differed from the original HSI patch spectra. Exemplary fail cases were depicted in columns six for Bicycle GAN and seven for the *pix2pix* approach of Figure 26.

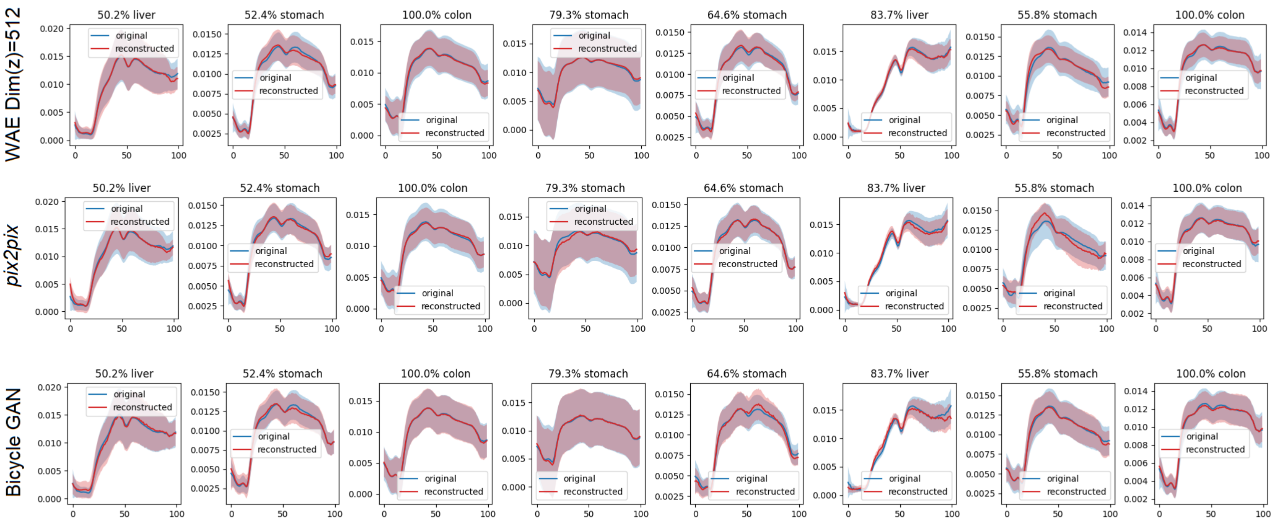


Figure 26: Example ℓ_1-normalized, organ-weighted median spectra for WAE with latent dimension 512 and both postprocessing GAN approaches. Spectral band 0 corresponds to a wavelength of 500 nm while band 100 corresponds to 1000 nm. Percentages in the plot headings give the coverage of the mentioned organ within the patch. The spectral agreement for the postprocessing approaches either improves beyond the WAE results, or shows large deviations in certain wavelength ranges, as seen in columns six and seven.

4. Embedding Analysis

Besides evaluation of spectral consistency, the comparison of WAE latent space embeddings of both real and generated image patches give further insight into the physiological correctness and generalization capabilities. This is especially the case, since the optimal transport concept of the WAE relies on the quality of the embedding [67] and the latent space structure as well as the organ label structure of the latent space thus allow to judge, *whether the deep learning pipeline can generate hyperspectral image patches that generalize beyond the training data.*

Experimental Design

Cosine similarity and EMD were computed from HSI patch and generated patch latent space encodings, for the purpose of analyzing embedding consistency quantitatively. For qualitative assessment of the latent space structure, results for latent space interpolations, noisy latent space reconstructions and a *Uniform Manifold Approximation and Projection* (UMAP) of approximately 2.000 latent space vectors were plotted and evaluated.

Method Details

Similar to the last experiment, UMAP was used to display global embedding structure as first result to grasp an intuitive understanding. UMAP was chosen over t-SNE since it was claimed to preserve global structure better than t-SNE [116]. A hyperparameter search for the clustering parameters was carried out on 2.001 encoded HSI patches from the test dataset. The resulting UMAP parameters, which showed well interpretable clustering results, were 8 neighbours and a euclidean distance of 0.1. For the visualization, all HSI patches were provided with the organ label of the largest organ in the image patch corresponding to the embedding. The 2.001 HSI patches also were preselected according to their organ labels and largest organ coverage. Only samples from the organs heart, lung, liver, colon, jejunum, stomach, spleen, gallbladder, peritoneum, pancreas, kidney and kidney with peritoneum were picked. Additionally, it was enforced that one of the reported organs had to make up more than 70% of the patch content, to reduce patches containing many equally large organs. This served the purpose to obtain a clearer as well as easier interpretable visualization.

To keep differences between latent space embeddings dimension independent, cosine similarity

$$\mathrm{CS}(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{||\vec{x}|| \cdot ||\vec{y}||} = \cos(\theta) \tag{39}$$

and EMD were chosen over Manhatten or Euclidean distance. The reported distances were calculated between a latent vector $\vec{x}$ of a real HSI patch and the encoding (reembedding) of a reconstructed or postprocessed patch $\vec{y}$. The HSI data for the calculations was provided by the dataloader, which loaded 1.000 sample patches from the test dataset without augmentations.

Interpolations between embeddings of randomly sampled HSI patches were reconstructed, postprocessed and plotted for visual assessment, or examination of the latent space in a sample-based, qualitative manner. To account for the properties of the high-dimensional latent space, namely its norm following the χ-distribution and thus forming a bubble in high-dimensional space, the simple linear interpolation was compared with a spherical interpolation similar to *Slerp* [117]. Contrary to the linear interpolation, the spherical interpolation kept the norm

closer to spherically symmetric, χ-distributed, embedding norm 'bubble'.
Additionally, qualitative smoothness of latent space environments was assessed by adding noise to latent embeddings and comparing the to RGB transformed outcomes. Warping was introduced for this aspect and is not to be confused with the truncation trick [118]. Warping renormalized the latent vectors, to which noise was added for the latent neighbourhood exploration, to make them follow the latent norm distribution again. This was helpful, since otherwise latents which were from even more unlikely regions of the latent space would have been sampled with increasing amplitude of the noise. The goal to 'warp' the generated noisy latent $\vec{z}$ back on to the latent space bubble was achieved by at first ℓ_1-normalization of the latent vector, before it was multiplied with the χ-distribution mean μ_χ. Afterwards, a randomly $s \sim \mathcal{N}_p(0,1)$ scaled χ-distribution standard deviation σ_χ was added, which gives

$$\vec{z}_{\text{warped}} = \frac{\vec{z}}{||\vec{z}||_1} \cdot (\mu_\chi + s \cdot \sigma_\chi). \tag{40}$$

Both mean and standard deviation of the χ-distribution were calculated for the specific latent space dimension externally once, because the computation required the gamma function Γ of large float values, for which e.g. scipy's implementation failed.

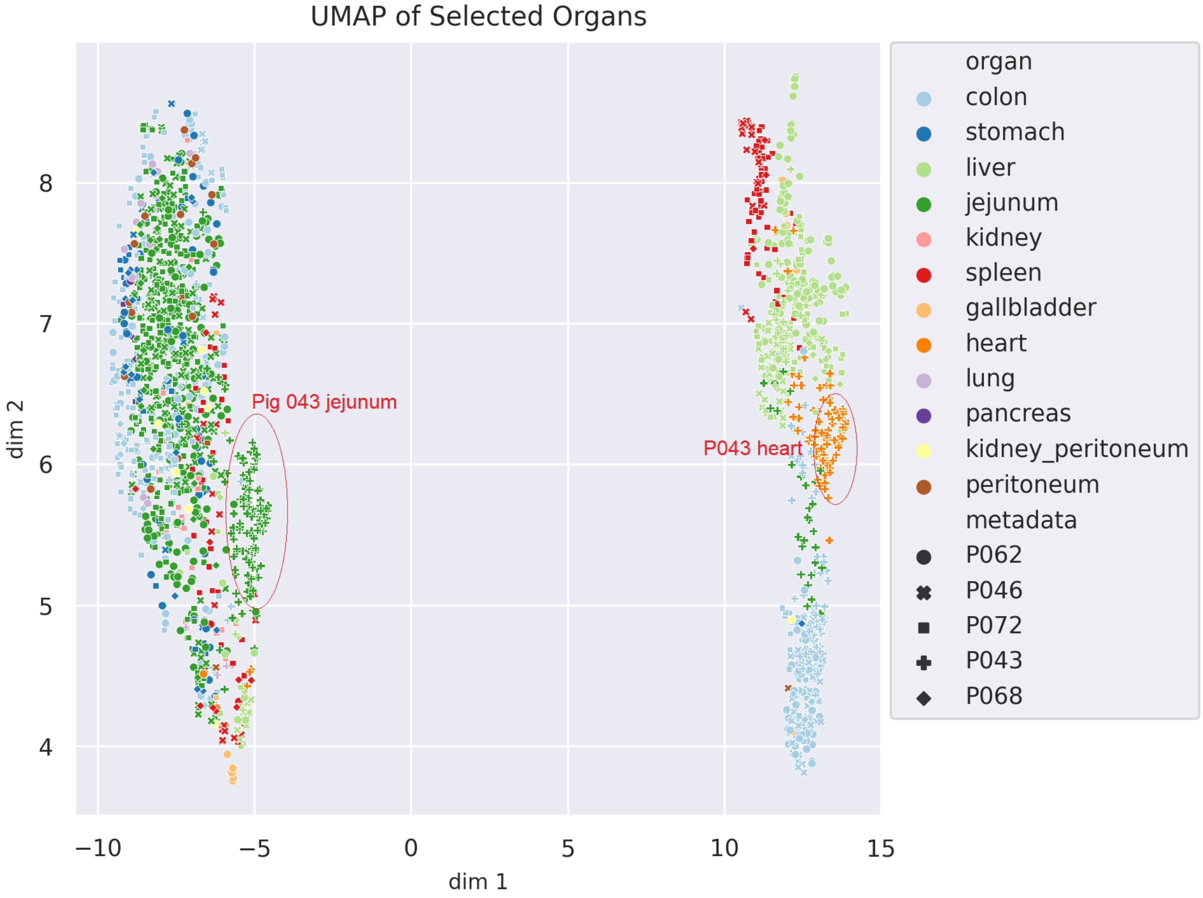


Figure 27: UMAP of 2.001 latent space embeddings. Indicated organ colours refer to the largest organ in the HSI patch, markers refer to the respective pig, which is source of the HSI patch. Anatomically close organs, which are likely to be within one HSI patch, are also close in the latent space embedding. Two minor pig and organ-specific subclusters are outlined in red.

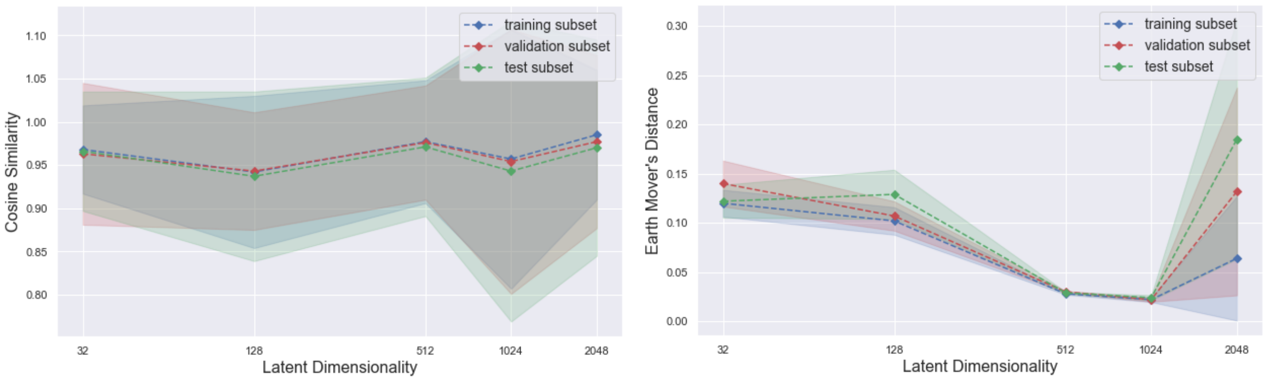
Results

UMAP Embedding Visualization Figure 27 shows the UMAP for 2.001 HSI patch embeddings from the test dataset, where colours refer to the largest organ in the patch. Different markers indicated the five different pigs, which were the source of the respective HSI patch from the test dataset.

No individual confounder clusters such as e.g. pig-specific clusters were visible in Figure 27. While there were no separate pig clusters, there were two specific groups, for jejunum of pig 43 and for heart patch embeddings from the same pig. Further, neighbouring organs were close in the UMAP visualization of high-dimensional space. Examples for the closeness of anatomically nearby samples in Figure 27 were kidney peritoneum, peritoneum and colon; jejunum, colon and stomach; spleen with liver, jejunum and gallbladder and as a last example gallbladder and liver. Overall, two large clusters were seen, the one on the right with distinct organ clustering and the one on the left with patch samples, clustered around jejunum samples.

Quantitative Analysis Plots of the embedding distances are depicted in Figure 28. Cosine similarity and EMD were used to assess the difference of 1.000 reembeddings of WAE reconstructions from the embeddings of the real HSI patches.

While the cosine similarity delivered stable results within its large error margins, the best overall results were observed for a latent dimension of 512. For the EMD the situation was a little bit different: The values dropped until they stabilized for latent dimensions of 512 and 1024 and afterwards skyrocketed for the higher latent dimensionality of 2048. Again, the error here was so large, that it was reduced by a factor of 10 for visualization purposes.



(a) Cosine Similarity of 1.000 latent space embeddings with their corresponding reembeddings of WAE reconstructions.

(b) EMD of 1.000 latent space embeddings and corresponding WAE reconstructions. Error bands are divided by a factor of 10.

Figure 28: Distances between HSI patch latent vectors and respective reconstructed patch latent vectors for different latent dimensions. Both distance measures have large error margins, but while the cosine similarity results are comparable across all latent dimensions, the EMD favours latent dimensionalities of 512 and 1024.

Table 7 additionally shows the embedding distances for real HSI patch embeddings from their postprocessed counterparts. The computed values showed a non-significant decrease in cosine similarity and an increase of nearly one standard deviation in EMD, when compared to the results for the WAE with latent dimension of 512. Decrease in cosine similarity and increase in EMD came together with an increase of the respective embedding distance error.

Model	Cosine Similarity	Earth-Mover's-Distance
WAE	0.98 ± 0.07	0.029 ± 0.010
pix2pix	0.96 ± 0.08	0.036 ± 0.014
Bicycle GAN	0.97 ± 0.07	0.036 ± 0.013

Table 7: Embedding differences between 1.000 HSI patches, sampled from the test dataset, and their complementary reconstructed and postprocessed patches. WAE results refer to a latent dimensionality of 512 and are the same used as input for the postprocessing GANs.

Qualitative Analysis Two randomly received HSI patches were sampled from the test dataset, with restriction that a unique organ must cover at least 80% of their pixels. The obtained patches were encoded and the latent vectors interpolated and reconstructed. Two ways of interpolation, namely linear and spherical interpolation, were utilized.

In Figure 29, slightly less clear shadows are seen for the linear interpolation when comparing it to the spherical interpolation; however, no big differences or flaws are discovered. The postprocessed spherical interpolations also provided smooth interpolation results without visual artefacts for the RGB visualizations of interpolation results. Two-dimensional, linear interpolation grids for all three networks can be found in the *Embedding Analysis: Additions* section.

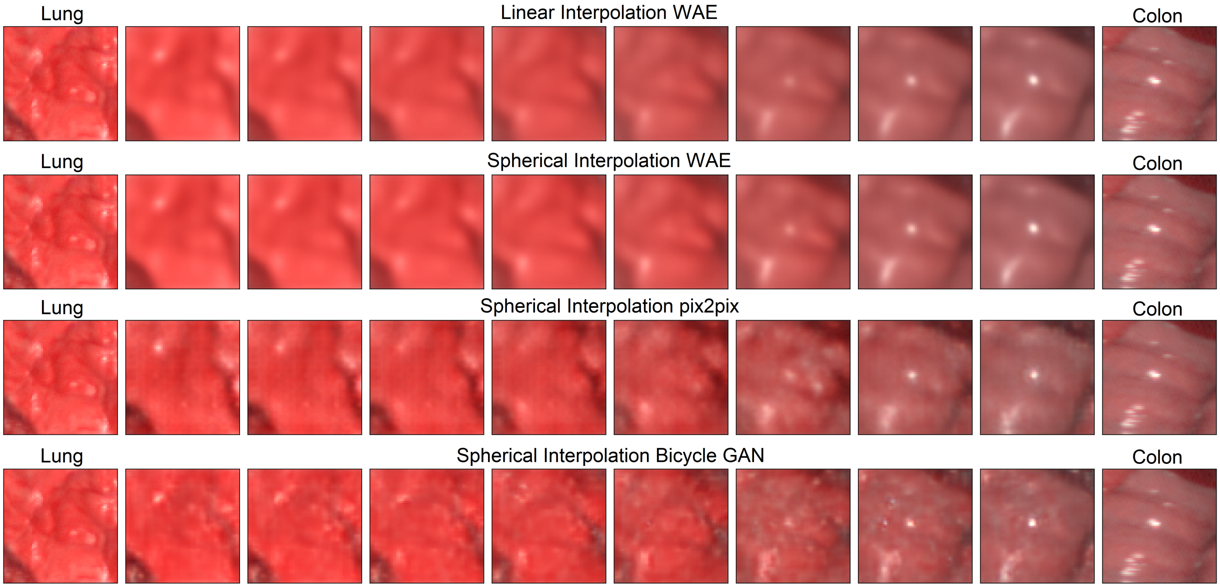

Lung
Linear Interpolation WAE
Colon
Lung
Spherical Interpolation WAE
Colon
Lung
Spherical Interpolation pix2pix
Colon
Lung
Spherical Interpolation Bicycle GAN
Colon

Figure 29: Interpolation from a lung to a colon sample. Top to bottom: Linear interpolation, spherical interpolation, spherical interpolation postprocessed with *pix2pix* and spherical interpolation postprocessed with Bicycle GAN. Both left- and right-most columns show the real RGB image patches. All interpolations return smooth image transitions with visually realistic contents.

Lastly, latent space environments were qualitatively explored on a sample basis. For this, Gaussian noise $n \sim \mathcal{N}_p(0, 1)$ was added to latent space vectors of HSI patches from the test dataset. Except for the last column in Figure 30, the warped reconstructions yielded a visually much more appealing result than reconstructions of the latents with only added noise. While reconstructions from noisy latent vectors without warping either looked largely different or not physiologically meaningful at all, the warped reconstructions usually deviated in only minor aspects such as texture, rotation or translation from the 'clean' reconstruction.

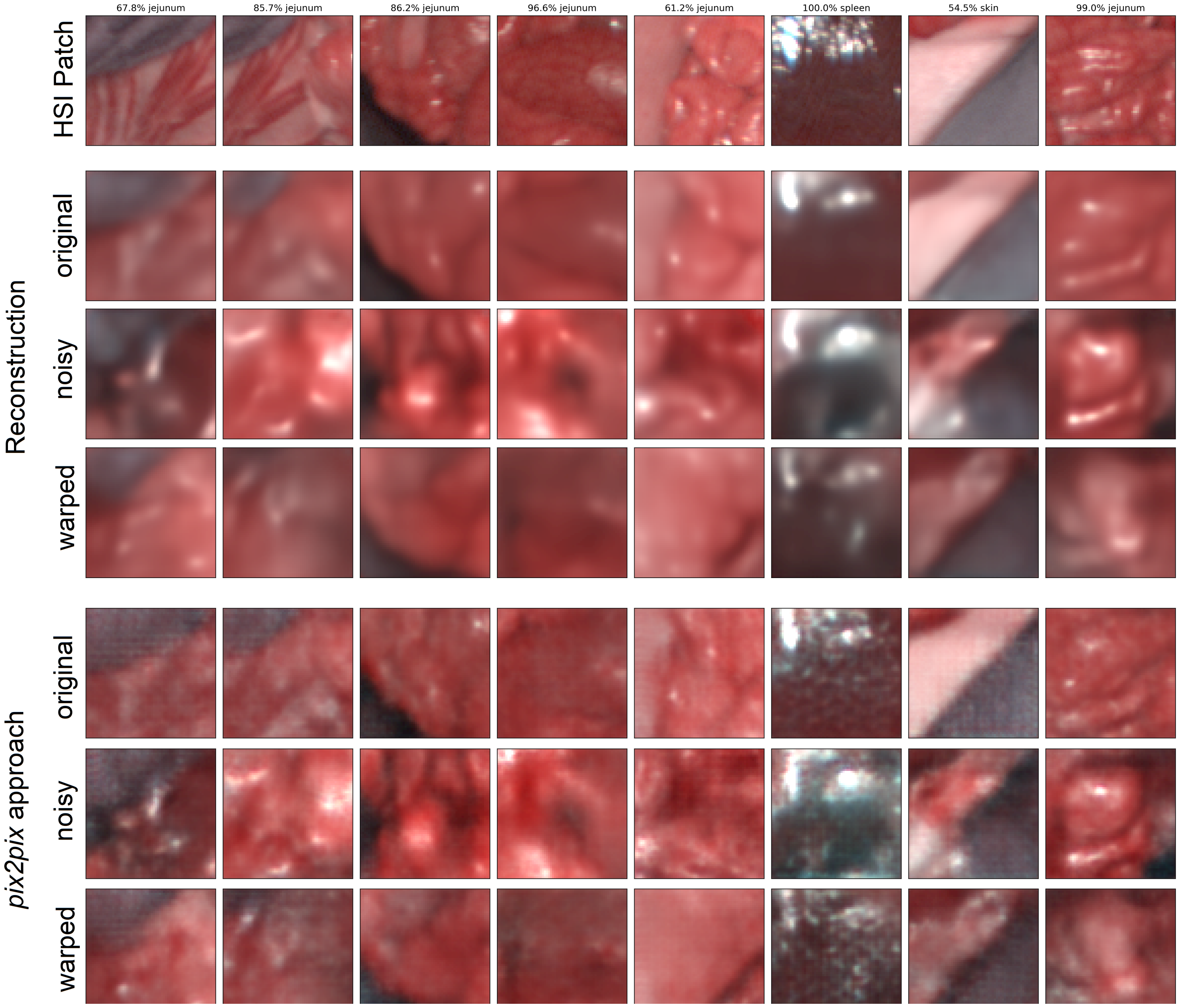


Figure 30: Results split into RGB visualization of HSI patch from the test dataset, three kinds of reconstructions and three kinds of postprocessings. Top to bottom: Real image patches, WAE reconstruction, noisy WAE reconstruction, warped noisy WAE reconstruction, reconstruction post-processed, noisy reconstruction post-processed and warped noisy reconstruction post-processed. Post-processing is done with the *pix2pix* approach, Bicycle GAN results can be found in the appendix. Except for results from the last column, warping massively improves the outcome and results look physiologically much more plausible. With the warped results, the environment of latent space vectors shows minor changes in shape or texture.

5. Texture Analysis

To quantify realism and answer, *'whether the proposed deep learning pipeline can generate hyperspectral image patches that feature realistic textures'*, this section presents results of *Image Quality Assessment* metrics. Standard IQA metrics also allow to compare results of the proposed image generation pipeline to state of the art (SOTA) implementations.

Experimental Design

FID and KID were calculated between train, validation and test datasets to obtain a baseline for achievable IQA metric scores. The three datasets have similar contents and thus yielded low scores for both metrics, which were used as expected differences between datasets. Afterwards, all presented metrics for reconstructed and postprocessing patches, corresponding to an underlying HSI 'ground truth' patch from the test dataset, were calculated. Patches sampled from the WAE latent space were due to the lack of comparable data only evaluated with the unpaired and overall dataset metrics. Commonly in literature used metrics and obtained scores in SOTA work were collected and compared to metric scores achieved with the image generation pipeline of this work.

Method Details

Calculation of MSE and PSNR was straightforward and the presented results were computed from 50.000 HSI patches, sampled from the respective dataset without augmentations. Results were aggregated before calculating mean and standard deviation. For DISTS score and SSIM was proceeded in the same way and the dataloader provided random samples without augmentations for all metric calculations.

For additional assessment of metric stability of FID and KID, calculations were run on different sample amounts [84]. FID was calculated on 10.000 and 50.000 aggregated Inception preactivations. The first result served as control quantity and the score on 50.000 preactivations could be compared to literature results. $\text{FID}_{10.000}$ splitted the calculation previously done on 50.000 preactivations into five scores from 10.000 non-overlapping samples, which thus enabled to compute a standard deviation for the five individual scores. The unbiased KID claimed reliability, even when calculated on smaller sample amounts [86]. Therefore, the KID was calculated on randomly selected subsets of size 1.000 ($\text{KID}_{1.000}$) and 10.000 ($\text{KID}_{10.000}$). Computations were repeated 50 and 5 times for $\text{KID}_{1.000}$ and $\text{KID}_{10.000}$, respectively.

The unpaired metrics, Inception Score (IS) and mutual information (MI) from Equation 34, were calculated in analogy on 50.000 aggregated classification probabilities of the same, on HSI data trained Inception v3 network. Lastly, BRISQUE was calculated for 10.000 samples and the score results were aggregated for overall mean and standard deviation computation.

As already mentioned, the image patches were sampled from the dataloader and then reconstructed and postprocessed. Contrary, the random patch samples were decoded and postprocessed from low-discrepancy Sobol sequences [113] that were Box-Muller transformed [114, 115], to achieve uniform latent space coverage for sophisticated, latent space-covering evaluation.

Results

Inter-Dataset IQA The inter-dataset results are shown in Table 8. The FID score improved (decreased) with increasing sample size and showed quite large deviations for the different datasets. Results for the KID were much more stable both among datasets as well as across different sample sizes. Notably, the KID results were still more than three standard deviations away (significant) from a score of zero, which implied that the datasets have a small but significant deviation in the underlying distribution.

Datasets	$\mathrm{FID}_{10.000}$ (↓)	$\mathrm{FID}_{50.000}$ (↓)	$\mathrm{KID}_{1.000}$ (↓)	$\mathrm{KID}_{10.000}$ (↓)
train-validation	40 ± 6	37	0.021 ± 0.007	0.0189 ± 0.0012
train-test	40 ± 9	36	0.021 ± 0.008	0.0197 ± 0.0018
validation-test	27.9 ± 1.9	24.0	0.021 ± 0.008	0.0187 ± 0.0025

Table 8: Results from the dataset metrics FID and KID. Arrows mark the direction of improving scores. While the FID scores deviate largely and are due to the inherent bias harder to interpret, the KID distances are stable and equal across different datasets and sample amounts.

Paired Data IQA Table 9 reports the metric scores from paired reconstructions and postprocessed image patches for the WAE with latent dimension 512 and HSI patches from the test dataset without augmentations.

Low-level metrics such as MSE, PSNR and SSIM preferred the results of the WAE first pipeline step, while Inception-based metrics returned better results for the postprocessed patches. It is notable, that the reported KID values were close to zero; however, as for the dataset comparison significantly different when judging by the standard deviation. When additionally the reconstruction and postprocessed KID scores were compared with the inter-dataset scores, better scores for the deep learning generated results were observed.

IQA metric	WAE	pix2pix	Bicycle GAN
MSE (↓)	**0.0015 ± 0.0015**	0.0022 ± 0.0021	0.0023 ± 0.0021
PSNR (↑)	**29.3 ± 3.2**	27.7 ± 3.1	27.6 ± 3.0
SSIM RGB (↑)	**0.75 ± 0.08**	0.68 ± 0.08	0.65 ± 0.09
SSIM HSI (↑)	**0.83 ± 0.06**	0.78 ± 0.07	0.76 ± 0.07
DISTS (↓)	0.28 ± 0.03	**0.25 ± 0.03**	**0.25 ± 0.03**
$\mathrm{FID}_{10.000}$ (↓)	82.4 ± 3.6	28.6 ± 2.2	**14.0 ± 1.2**
$\mathrm{FID}_{50.000}$ (↓)	81.4	27.3	**12.9**
$\mathrm{KID}_{1.000}$ (↓)	0.0381 ± 0.0026	0.0078 ± 0.0020	**0.0025 ± 0.0006**
$\mathrm{KID}_{10.000}$ (↓)	0.0389 ± 0.0005	0.0076 ± 0.0005	**0.0026 ± 0.0002**
MI real (↑)	1.8 ± 0.7	1.8 ± 0.7	1.8 ± 0.7
MI synth. (↑)	1.5 ± 0.6	1.6 ± 0.6	1.6 ± 0.6
IS real (↑)	7.8 ± 9.6	7.8 ± 9.5	7.8 ± 8.9
IS synth. (↑)	5.4 ± 4.7	6.5 ± 7.1	6.4 ± 7.8
BRISQUE real (↓)	42 ± 10	42 ± 10	- ± -
BRISQUE synth. (↓)	64 ± 7	**43 ± 10**	- ± -

Table 9: IQA metric results for reconstructed and postprocessed HSI patches. Arrows mark the direction of improving scores. While classic IQA metrics favour the WAE result, feature-based or no-reference metrics heavily prefer postprocessed and in specific Bicycle GAN results.

For metrics such as DISTS, based on other pretrained models (VGG), or metrics like BRISQUE, based on image statistics, the postprocessed results also delivered improvements over the WAE results. For Bicycle GAN, at least one parameter of the asymmetric generalized Gaussian distribution (AGGD) was calculated to be smaller than zero and thus the calculation of the BRISQUE metric failed in an intermediate step of the PIQ [108] implementation and was therefore not reported.

Unpaired Data IQA In comparison to the paired WAE reconstructions and postprocessed image patches, the FID and KID values in Table 10 have risen significantly and the KID values were reported to be on one level with the inter-dataset scores. The MI and IS dropped but due to the large standard deviation were still similar to the previous scores on paired data. For WAE reconstructions of low-discrepancy samples, the BRISQUE score improved to the same level as for real samples and *pix2pix* postprocessing. For Bicycle GAN, again at least one parameter of the AGGD was calculated to be smaller than zero and thus the calculation failed in an intermediate step of the PIQ [108] implementation and was therefore not reported.

IQA metric	WAE	pix2pix	Bicycle GAN
$\text{FID}_{10.000}$ (↓)	122 ± 5	91 ± 5	**73 ± 7**
$\text{FID}_{50.000}$ (↓)	121	89	**71**
$\text{KID}_{1.000}$ (↓)	0.045 ± 0.004	0.032 ± 0.003	**0.0230 ± 0.0021**
$\text{KID}_{10.000}$ (↓)	0.0455 ± 0.0006	0.0316 ± 0.0012	**0.0224 ± 0.0006**
MI (↑)	0.9 ± 0.6	1.1 ± 0.6	1.1 ± 0.6
IS (↑)	3.2 ± 3.3	3.7 ± 5.8	3.8 ± 4.4
BRISQUE (↓)	43 ± 10	42 ± 11	- ± -

Table 10: IQA metric results for reconstruction and their postprocessed counterparts, sampled from the WAE latent space. Arrows mark the direction of improving scores. Again, Bicycle GAN results are rated best while the Inception score is indecisive.

SOTA Comparison Table 11 displays collected results from above's paired and unpaired quality assessment and compares them to reported outcomes from other image synthesis work. MSE and MI were not reported, as they are implicitly incorporated in the PSNR and IS.

As literature does rarely use DISTS and BRISQUE scores, realistic values were taken from the corresponding original publications: Good DISTS values were around and below ≈ 0.2 [83], while BRISQUE values on real images were not reported in the original publication [82], but theoretically go as low as zero.

SSIM values were computed channelwise and meaned afterwards, which allowed to compare the HSI SSIM result with SSIM results from literature. The PSNR results were on par with results from literature, although best models for specific datasets obtained results better by 3 dB and thus a factor of two in MSE. A similar picture was obtained for the structural similarity, where achieved scores lay in between high and low values reported in literature.

For the feature-based metrics, the results looked slightly different: Since implementation details were often lacking in the publications, it was hard to retrace what actually was done. This resulted in a wide range of received metric values, especially for FID and IS. The reported IS for different implementations were the least comparable results, evident for the original MUNIT implementation [48] which reported a much lower score than to be expected from the visual quality of the model results. For medical datasets such as those of Rivoir et al. [2], KID and FID scores from the paired setting delivered results with comparable scores. In the unpaired setting however, the FID was again much larger and also the KID was significantly larger, judging by the reported small error from 5 calculations. Notably, results from literature rarely computed standard deviations and hence reported arbitrary amounts of decimal places.

Source	PSNR (↑)	SSIM (↑)	$\text{FID}_{50.000}$ (↓)	KID (↓)	IS (↑)
This Work/ Top Paired	29.3 ± 3.2	0.83 ± 0.06	12.9	0.0025 ± 0.0006	6.5 ± 7.1
This Work/ Top Unpaired	-	-	71	0.0224 ± 0.0006	3.8 ± 4.4
[80]	32.05/ 27.58	0.9019/ 0.7620	-	-	-
[81]	30.09/ 25.87	0.907/ 0.784	-	-	-
[2]	-	-	26.8	0.0114	-
[75]	-	-	15.71	0.00288	-
[89]	-	-	-	-	63.702 ± 7.869
[48]	-	-	-	-	1.050

Table 11: Displayed results of Ledig et al. [80] and Yang et al. [81] depend on the dataset they tested on, highest and lowest scores are presented respectively. Scores for Rivoir et al. [2] are from different models and the best model values for the individual scores are reported. Note that the FID for Rivoir et al. [2] is reported on 10.000 images instead of the usual 50.000. For StyleGAN2 [75], values are computed for the medical BreCaHAD histopathological breast cancer dataset. From Barratt et al. [89], the IS for Inception v3 on ImageNet is reported. For MUNIT [48], the average IS is reported. All values are displayed with the same amount of digits as in the original publications.

6. Downstream Task: Image Segmentation

One possible application for synthetic HSI patches is providing data for image analyses tasks such as organ segmentation. Seidlitz and Sellner et al. [4] explored the usage of HSI to improve semantic segmentation for different amounts of training data and different input data modalities such as pixel inputs, HSI patch inputs and full hyperspectral images. For their proposed approach and an artificially limited dataset, the presented image generation pipeline provides in this section additional input data to answer, *'whether the generated image patches can improve a downstream organ segmenation task'*.

Experimental Design

This experiment evaluated the generated HSI patches with a downstream semantic organ segmentation task. As a baseline, real image patches were collected and used to train the image segmentation network. The training was repeated for a mix of real and reconstructed or post-processed image patches as well as for only the generated patches.

The contribution of the results was two-fold: At first, the downstream task served as an additional image quality measure, when the performance of a model trained on synthetic patches was assessed on real data. Second, the evaluation of a downstream task trained on generated HSI patches allowed to answer the research question, *whether the generated hyperspectral image patches improve a downstream organ segmentation task* in a limited data setting. Interpretation of the received results was further made possible by comparison to the patch model and inter-rater variability, reported in Seidlitz and Sellner et al. [4]. Results tested in a non-limited data setting as well as full metric reports for the pipeline training can be found in *Artificially Limited Data Evaluation* section in the appendix.

Method Details

Once more, the dataloader with augmentations was used to crop 64 × 64 × 100 image patches from a limited HSI training set, which contained only 39 images from 3 pigs instead of 236 images from 12 pigs for the full training set. For each trained network of the image generation pipeline, 8192 real HSI patches and the corresponding reconstructed or postprocessed patches were collected. Gathering synthetic samples generated from real HSI patches was necessary, since corresponding segmentation maps were needed for downstream task training. Synthesis of 'new' samples by reconstruction from artificially generated latent space vectors was in this context not possible, due to a missing connection of the generated patches and the underlying segmentation maps. Noisy latent space sampling as described in *Embedding Analysis* was not performed, to better be able to evaluate the quality and realism difference between training on generated samples and testing on real image patches.

The collected real and generated HSI patches were used to train the 64 × 64 semantic segmentation network of Seidlitz and Sellner et al. [4] in three ways: Either only generated patches or only original HSI patches were used for training; or the segmentation network was trained on a combination of one batch of real HSI patches and one batch of (124 + 124) generated patches. To make these results more comparable, training for purely real HSI patches as well as training for purely generated HSI patches combined two batches, such that the batch size of 248 was equal across the different training datasets. As in the original paper, the segmentation network was trained for 100 epochs of 500 patches per epoch and the same shift, scale and rotation as well as flip augmentations were used.

Results

The results for the different datasets were visualized with the ChallengeR toolkit [119] and are displayed in Figure 31 and Figure 33. The metric value evaluated on the test set was the dice score for the predicted segmentation maps of the trained network and the correct surgeon annotations. The individual data points in the plot represent results for images from one specific pig in the test set.

Two features are striking in the boxplot of Figure 31: The dice score spread between individual pigs was large and training on real image patches consistently outperformed mixed training or training on only generated data from the first stage or second stage of the image generation pipeline. Surprisingly, the three different networks of the image generation pipeline performed very similar in terms of the dice score reported on the test dataset.
To allocate the obtained results from limited patch results within achievable results, results from Seidlitz and Sellner et al. [4] are stated in the following: Training with data from only 3 pigs achieved an average dice score of around 0.79, which is similar to the achieved score reported in Figure 31 on real HSI patches. Training with the full training dataset achieved a dice score of 0.89 with a standard deviation of 0.04, on par with inter-rater variability with the same dice score of 0.89 and a standard deviation of 0.07.

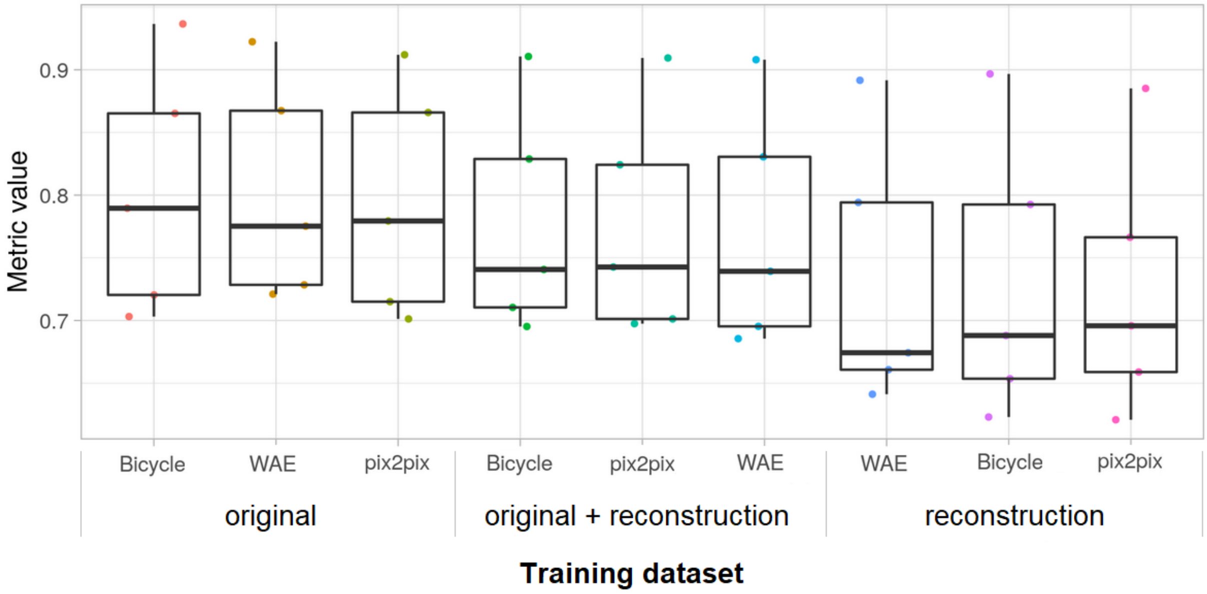
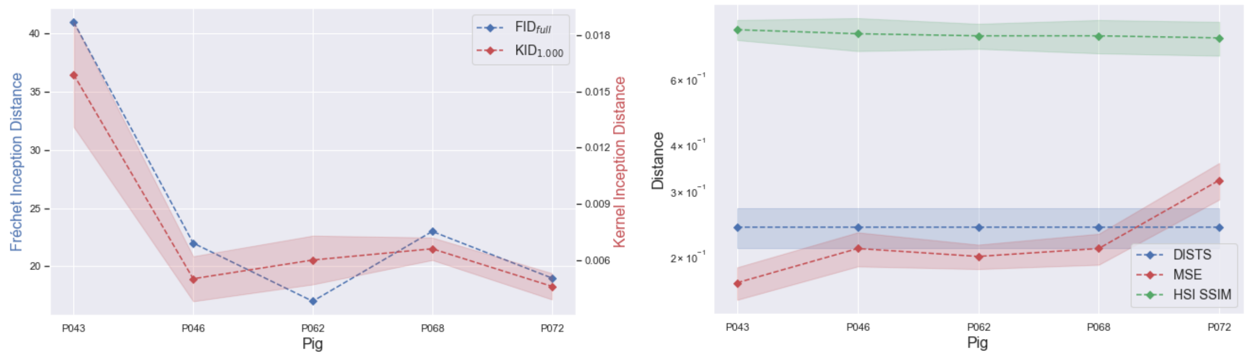


Figure 31: Boxplot of dice score results for data of individual pigs from the test dataset. Dots indicate individual results on test pigs and boxplots median and quartiles. 'original' denotes results obtained from real HSI patches, 'original + reconstruction' mixed training reconstruction training on only generated data. Each of these fractions is subdivided into the three network types, where the originals only deviate in the randomly cropped patches. Real HSI training for all three aggregated sets of patches consistently outperforms mixed and synthetic data training. Deviations in dice score between the different pigs are large but constant across different datasets [4].

Figure 32 presents metric results of overall 50.000 postprocessed patches from Bicycle GAN with underlying real HSI patches, retraced to the individual pigs. The distribution for the patch amount of the individual pigs reflects the underlying image amount inequality and was: P043: 4894 patches (from 23 images), P046: 6488 patches (from 16 images), P062: 19655 patches (from 59 images), P068: 5619 patches (from 23 images), P072: 13344 patches (from 47 images).



(a) Feature-based FID and KID metric results.

(b) Ambiguous classic metric results.

Figure 32: Overall 50.000 Bicycle GAN patches from the test dataset were evaluated. KID results are computed for 1.000 samples, all other results for all available patches for one unique pig. MSE results are scaled by a factor of 100 and its error divided by a factor of 10, to visualize them with the other classic IQA metrics. Lines connect the same metrics across different test pigs.

KID reported scores from 50 randomly selected subsets of size 1.000 within the pig subset. All other metrics were computed on the full patch amount of the individual pig subset.

While the feature-based metrics FID and KID showed larger differences between distinctive pigs in Figure 32, this ranking was not free from ambiguity and the MSE contradicted their ranking. Other classic IQA metrics were indecisive and overall no clear correlation between metric scores and individual pig results in Figure 31 and Figure 33 was visible.

Figure 33 depicts the ranking stability of the achieved dice scores on the individual pigs as spaghetti plot, with the ranking of the individual datasets for all five pigs in the bottom. The differences between training with real image patches and training on purely generated samples are thus illustrated in a more pig-centered manner.

With the same data as in Figure 31, the differences in dice score for the individual pigs were well visible and remained comparable across the different training datasets. Overall, the dice scores dropped more for pigs with already lower dice scores. The rankings depict a stable difference between purely real data for training, mixed data and purely synthetic data; however, the individual network rankings surprisingly showed no clear preference for the individual pigs.

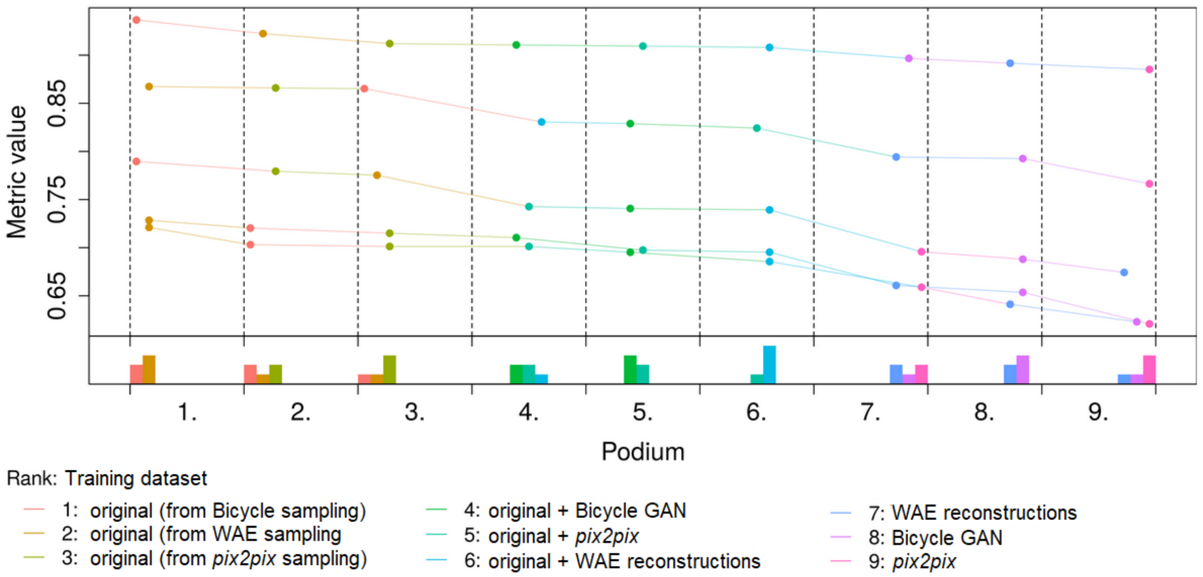


Figure 33: Spaghetti plot on top presents the dice score results of images from individual pigs, each unique pig connected with a line. Podium plots in the bottom show, how many times training on the respective dataset resulted in a specific rank in dice score for the five different pigs. 'original' HSI patches only deviate in randomly cropped image regions across the three networks. Dice score spread stays constant and it decreases more for already low scores [4].

Part V.
Discussion and Conclusion

The main objective of the present work was the investigation of realistic hyperspectral tissue patch generation. The experiments section therefore explored visual results, similarity of real and generated spectra, generalization performance as well as textural quality. Additionally, the generated HSI patches were used as an extension of an artificially limited dataset to study the effect on test results of an image segmentation task.

Overall Conclusion The proposed image generation pipeline generalizes beyond the training dataset and returns spectrally consistent HSI patches by adapting deep learning frameworks from (medical) RGB literature to HSI with 100 wavelengths. Remarkably, spectral consistency is achieved without dedicated architectures or loss terms that focus on spectral properties. Further, the image generation pipeline performes well for different cameras with unique spectral responses. Together with obtained realistic and to SOTA comparable textural results, this presents a first technique to synthesize physiologically correct, hyperspectral tissue patches.
A limitation of the current pipeline is its inability to generate organ segmentation maps or other physiological ground truth along with the HSI patches, hindering its application in a downstream task, since the labels also have to be of high diversity to create a benefit. The image generation pipeline is thus so far not able to generate additional labelled synthetic data which aids data sparsity.

Following the structure of the research questions, RGB results and visual improvement possibilities are discussed first in the *Imaging Effect Analysis* subsection. Different image pipeline stages are compared among each other and against the original WAE implementation [67] in the *Spectral Features* subsection, before anatomical correlations and latent space neighbourhoods are reviewed in the *Embedding Analysis* subsection. The *Texture Analysis* subsection discusses quantitative textural results in context with SOTA results and practices. Finally, a discussion of the *Downstream Task* examines the usability of generated hyperspectral data in (medical) use-cases.

Imaging Effect Analysis

For globally well-illuminated scenes, the image WAE reconstructions were blurry but overall correct and postprocessing could accurately recover the underlying real image patch. Accuracy here refers to accordance of general shape and colour outline, but also to specular highlights, shadows and tissue-specific texture. Visually good results were received for both image reconstructions as well as random decoded samples from the learned WAE latent space in Figure 34. Presented skin and spleen patches depicted the correct tissue-specific textures in Figure 15. Therefore, a weaker form of spectral correctness is with the colour correctness given, which together with correct textures and shadows, that affect 'three-dimensionality' of the scene, hence generate realistic tissue patches.

An overall deficiency across the different networks - no matter whether WAE, Bicycle GAN or *pix2pix* - is the inaccuracy of finer patterns. While the postprocessing GANs were able to recover many details that were blurred out by the WAE, the synthesized finer structures are often not authentic: For both camera-specific patterns as well as physiological structures such as vessel-meshes, attempts of mimicking these structures were seen. These attempts yielded for both Bicycle GAN and *pix2pix* approach edgy (camera pattern) or colour-wise noisy (vein pattern) recovered structures, which were distinguishable from the original physiological pattern by the human eye. When looking at the RGB patches in more detail, it should be noted that distinct organ borders, which e.g. are marked by shadows, are respected by both the vein colour jitter as well as the camera pattern, which mostly is present over background cloth. In light of recent research [120], the edgy reproduction of camera patterns as well as the inaccuracy of vein-meshes can be attributed to unsuccessful data augmentation policies which altered the orientation of the patterns in each HSI patch, thus hindering learning.

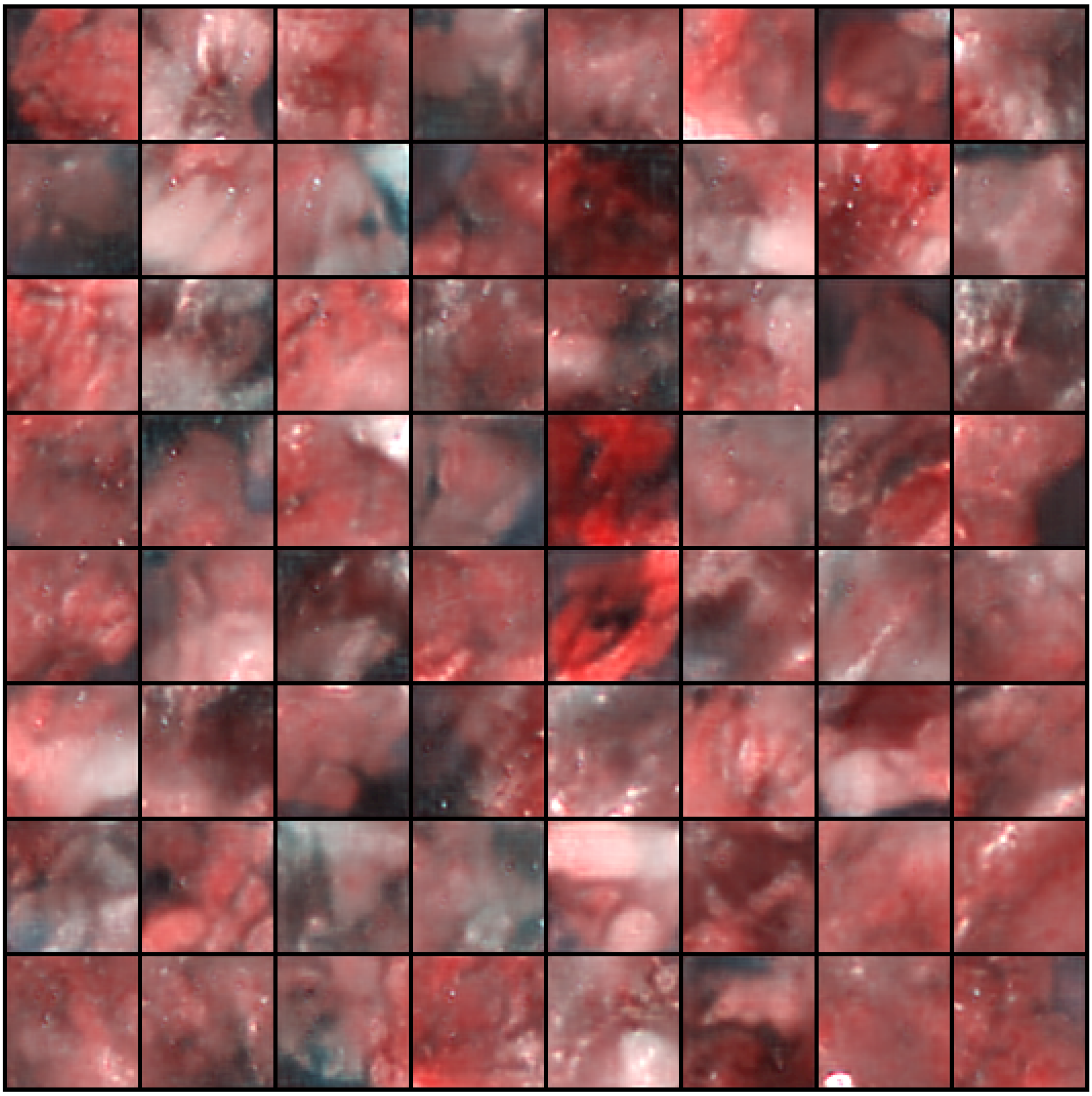


Figure 34: 64 low-discrepancy Sobol latent space samples, decoded with the WAE and postprocessed with Bicycle GAN. Results have realistic colour and structure contents as well as high diversity. The for Bicycle GAN typical, additional specular highlights are visible in the individual patches.

Two further, specific limitations were observed for synthesis of rare and complex structures such as heart and surgical instruments. Complex structures and movement during recording as for the heart observed, are in the postprocessing not accurately recovered. This can be attributed to the blurriness of the WAE results and the augmentations, that wash out finer structures for the vessel and camera patterns. The qualitative appearance of instruments is of mixed quality, as cases with good scene illumination were able to generate samples with realistic metallic shine, with sharp instrument borders and correct reflections on the instruments. When generating an instrument in a darker scene, a more blurry reconstruction was observed and subsequent, the postprocessing also did not recover clear borders and had oscillating colour properties.

Conclusions Oscillating colour properties on instruments and similar effects of edgy checkerboard patterns as for vessel and camera patterns are visible for *pix2pix* approaches in RGB literature [18], which therefore lead to the conclusion, that improvement in the framework is required to effectively overcome presented visual deficiencies. In the context of rarer organs, also an enlightening observation of relatedness of specular highlights and Bicycle GANs style code was made: When postprocessing results within the present work with Bicycle GAN, sometimes additional, across the image scattered, specular highlights appear. Specific qualitative evaluation of the eight-dimensional Bicycle GAN style space has shown that these scattered specular highlights are a property of the style code and that the style code hardly affects other image properties. This makes disentangling of imaging effects into 'layers' like camera pattern, vessel mesh and organ shapes desirable for more physiologically correct generation in ill-posed situations of bad illumination. Disentangling of different layers would also lead to clearer structure and more constraints for input data, which would aid the *pix2pix* in this work and in RGB literature [18].

To introduce the terms of work on data augmentation [120] for later usage, the obtained diversity of image patches was good, while the visual affinity (closeness) to the given data could be increased for the deficient cases such as bad illumination or intricate patterns, by means of incorporating more elaborate data structures like surface normal maps and layered information with different frameworks.

Spectral Features

The qualitative single organ PCA visualization showed satisfactory results, where reconstructions and postprocessed results lay close to the PCA embedding of organ weighted median spectra from real HSI patches. While no specific bias for closeness of reconstruction or postprocessing to the real embedding as well as no directional bias into more populated areas was observed, results in less dense regions tended to lie further apart. While this is to be expected, as samples from close to the border of the manifold should be rarer, it also implies that overall reconstruction and postprocessing have a lower affinity for median spectra of rarer cases, which coincides with previous qualitative observations.

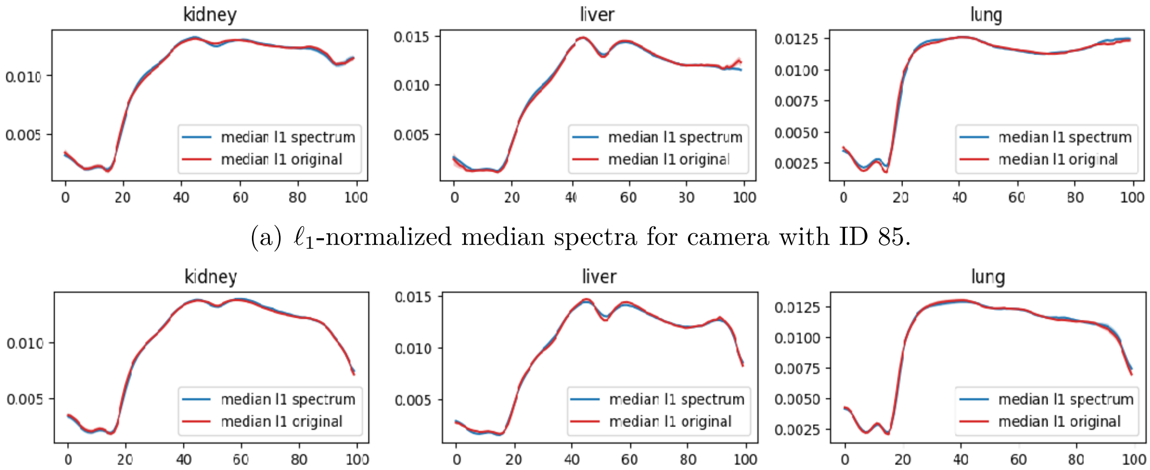
For the overall image patch PCA, similar effects were visible: The low-discrepancy samples, which served the purpose to cover the latent space evenly, showed a significant difference to the density manifold of 400 random HSI patch median spectra. This can imply two things: At first, the sample amounts could be too low to accurately compare the densities, or an actual difference is visible in PCA densities.

PCA Conclusions With observed higher KID distances from the texture analysis, this leads to the conclusion that the PCA densities show an underlying difference in data distribution.

However, this is not necessarily bad, as the low-discrepancy samples from the WAE latent space showed results of high diversity in Figure 34 and should do so to not only be restricted to HSI patches from the test dataset. The difference is thus attributable to a comparison of different data distributions, which has to lead to a different median spectrum distribution. This also leads to the conclusion, that adding in physiological boundaries via e.g. rejection sampling might become necessary to incorporate, when using automated sample generation as data source e.g. for organ-specific applications.

Quantitative median spectra differences of real and generated HSI patches showed good spectral consistency for latent dimension sizes ranging from 128 to 1024. Results for a larger or equal latent space size to 128 are visually not distinguishable in terms of general correctness and also the standard deviations of the median spectra are close to the underlying HSI patch data. For a too small amount of latent dimensions, the reconstructions suffer from missing content information while too large dimensionality adds complexity and thus decreases the overall results. The different behaviour of the two divergences for the largest latent dimension of 2048 is noticeable and can be attributed to the sensitivity of the divergences themselves to different aspects: Since the KL divergence weighs differences with the respective original distribution's value, difference in regions of higher spectral intensity weigh heavier. For the EMD, the score stays small even for large values of the true distribution, if the differences between the distributions are dispersed and thus the error is sometimes positive and sometimes negative, such that the differences do not have to be transported far. This explanation corresponds well to the observation, that the visualized qualitative organ-weighted median spectra for the largest latent dimension alternated with smaller, non-systematic deviations around the true median spectrum. It should be noted, that the deviations in the presented qualitative results do not seem to be an effect of the organ size in the respective HSI patch.
The quantitative median results of the postprocessed patches were for both divergences slightly worse than for the WAE results with latent dimensionality of 512. Looking at the qualitative results, postprocessed organ weighted median spectra are either very close to or in some wavelength range a good portion from the original median spectrum apart. Figure 26 columns six and seven show such failcase examples for Bicycle GAN and *pix2pix* approach respectively. Since specular highlights as additionally introduced by Bicycle GAN are spatially and not spectrally localized, they are ignored by the median and can not explain such behaviour or otherwise would be noticeable in the overall spectrum. An explanation for this behaviour can be provided by the texture change, incorporated through the postprocessing. As the postprocessing GANs try to mostly incorporate structures such as vessels or the camera pattern, the organ-wise medians change through shift in specific colour ranges, which therefore introduces a tradeoff between spectral and textural consistency.
The quantitative median spectra for both WAE and postprocessing GANs were troubled by large errors, which were still present when the sample size was increased to 5.000. While this might at first sight seem like a non-negligible amount of outliers, the more detailed evaluations of the downstream task showed, that the different pigs caused the diversity in quality. The observed error is hence large due to inherent different results for specific pigs and their different image as well as the content amount, which can not be reduced by further increasing sample size.

Remarkably, spectral consistency is obtained without specifically incorporating it in architecture - via three-dimensional convolutions which hindered performance- or training - in a dedicated loss regularizer term. Spectral consistency goes as far as being able to learn different camera types which have specific filter responses, shown in Figure 35. This result is not presented more prominently due to the small amount of data from different cameras which thus did not allow independent testing.



(a) ℓ_1-normalized median spectra for camera with ID 85.

(b) ℓ_1-normalized median spectra for camera with ID 98.

Figure 35: Mean of aggregated median spectra of three organs for cameras with different filters in top and bottom row. The reconstructions were obtained for 32 × 32 × 100 patches from initial WAE results on the HSI masks dataset, different camera responses in the NIR (band 90-100) well-visible.

Putting the latent space dimension of the HSI adapted WAE into perspective with the original RGB implementation of the WAE by Tolstikhin et al. [67] on CelebA dataset with a resolution of 64 × 64 × 3, the latent space dimensionality is with 64 much smaller than found necessary for HSI implementation. An increase in dimensions can be justified in two ways:
First of all, while the spatial resolution is the same, the spectral resolution increases from 3 to 100 by a factor of approximately 33. If this increase of features should be recovered accurately, while assuming a linear relation between image dimensionality and latent space for encoding of content, a larger latent space dimensionality of around 1000 is recovered naturally.
Less clear in scaling implication but of similar importance is the kind of dataset encountered: While CelebA contains faces, placed centrally in the image, the overall image structure of the HSI dataset is much richer while of similar content detail. Thus, an increase in latent space dimensionality would also be justified from this qualitative point of view.

Conclusions The quantitative spectral consistency results together with further results from literature [78] thus allow to conclude, that deep learning frameworks can be extended to hyperspectral data without special requirements beyond incorporating higher feature amounts in the involved architectures.

Embedding Analysis

Visualization of the embedding space by means of UMAP revealed an explainable latent space structure. Close organs in the UMAP were also close in an anatomical sense which is positive, since they are likely to appear in one image patch and hence justifying the expectation of closeness in latent space. Example cases for such a closeness were jejunum, colon and stomach or gallbladder, liver and lung. Thin grouping and overall sparse structure, with the same organs spread over several larger regions of the visualization, can be attributed to the small chosen nearest neighbour parameter of the UMAP, which hence partially gives rise to the two separate clusters. These two clusters of the UMAP split the encodings into a more unordered cluster around lower body jejunum samples and a second more orderly cluster of other internal organs. From this visualization, good generalization can be expected due to the meaningful cluster structure which embeds similar content close.

Results from the quantitative embedding metrics were remarkable, since random Gaussian samples in high dimensions are likely to be orthogonal [121]. In this light, it was surprising, that the cosine similarity values of image patch embedding and encoding of the reconstructed patches did not worsen with increasing latent dimensionality. This also speaks in favour of the embedding quality. Assessing the EMD, the encoding distances improved until a latent dimension of 1024 and afterwards increased massively, meaning that real HSI patch and reconstruction were percepted very differently. This likely results from the unfavourable trade-off of a higher latent dimensionality against more intricate feature encodings. For both distances large errors were observed, which again can be attributed to differing embedding quality among the pigs.
The quantitative results for the reembedded postprocessed patches were slightly worse than for the WAE results. This is another outstanding result, since this shows that the spectral consistency is implicitly incorporated in the embedding and more 'important' or easier to recover than the sharpness of the postprocessed patches.

The qualitative analysis comprises interpolation between latent space samples and exploration of latent space neighbourhoods. Both interpolation types delivered shape and colour-wise meaningful, smoothly transitioning results, also for the postprocessing GANs. The difference between linear and spherical interpolation was surprisingly small, as the linear interpolation was expected to leave the spherical latent space manifold when connecting two points in latent space with a straight line. However, the similarity of the different interpolation types is obtained, when the encodings do not lie on opposite sides of the sphere, but within 45° to 90°, where random latent samples occur [121].
Warping was observed to be important when trying to sample noisy versions of real HSI patch embeddings, which allowed to explore the neighbourhood on an image patch basis. If warping is not used, colour-wise wrong or shape-wise weird samples are observed, as to be expected from a latent space vector which does not lie on the latent manifold. Noisy latent vector samples which were renormalized (warped), show meaningful content manipulation, translation and rotation.

Conclusions All presented results were calculated, reconstructed or compared to data from the previously unseen test dataset. The global content structure and visual similarity thus allow to conclude that generalization beyond the training dataset was achieved and the proposed image generation pipeline was capable of learning meaningful features.

Texture Analysis

The initial dataset analysis provided a baseline for later results and at the same time displayed first issues with the reliability of some feature-based metrics: Images from the training (12 pigs), validation (3 pigs) and test (5 pigs) dataset, which were taken with the same camera and in an overall similar way, displayed for FID comparisons larger differences, unaffected for different sample amounts. The KID results were consistent across the datasets for both sample amounts but still all results were significantly different from a score of zero and hence provided a benchmark of achievable scores.

Evaluation of paired, generated data delivered much better values for FID and KID; however, the results were still significantly different from real HSI patches in terms of standard deviations. It is important to note that the FID can improve in the paired data setting, since the calculated preactivations then correspond to one another. For the displayed KID results the situation was different, as the computed preactivations were randomly shuffled and the pairing was thus revoked. The KID scores for the patches randomly generated from the WAE latent space agree within errors to those of inter-dataset comparison. The FID scores are much larger but are due to previously seen unreliability not discussed further.
The feature-based MI and Inception score showed better scores for real HSI patches, although the large errors for both scores question, whether they provide an in this case meaningful metric. This problem is not an issue of the trained Inception network or the score itself, but rather with the task given to the underlying Inception network: Contrary to many common image synthesis datasets, the HSI datasets with semantic labels do not contain unique objects in the patch, but rather a distribution of organ percentages, which is to be learned. For the MI and thus also the IS, calculated from the Inception network's probability outputs, this naturally allows both low and high scores and only depends on whether an HSI patch with rather unique organ content or with mixed organ content is assessed. Both IS and MI are hence no useful metric for this task and dataset.
While feature-based metrics preferred the postprocessed samples, the 'classic' metrics MSE, PSNR and SSIM clearly favoured WAE generated patches. This is the case, since the improved visual texture of the postprocessing GANs increased the MSE, which decreased the PSNR. This also manifests in the SSIM, which relies on mean and variance calculations. Mean and variance calculations in the SSIM also led to worse RGB SSIM results compared to SSIM computed for HSI data, which is not an effect of the HSI domain itself as the SSIM is calculated spectrum-wise and meaned afterwards. The worse RGB SSIM results from the RGB conversion, which takes less than half of all wavelength values into account (530 - 725 nm) and applies scaling, clipping and a gamma correction. Latter transformations increase the mean difference, which outweighs possible variance or covariance reductions, as these are only squared effects of values smaller than one, in Equation 30.
For the handcrafted feature or image statistic metrics DISTS and BRISQUE, the postprocessing models provided significantly better values. This is to be expected, since both metrics claim to coincide better with human judgement than classic IQA metrics and the depicted visualizations improved for the postprocessing GANs. The BRISQUE score for random WAE samples further decreased to a value close to the real images. As the generated images are of comparable visual quality, this also questions the suitedness of natural image statistics for application in a medical context. One reason for the counterintuitive BRISQUE scores as well as the problems with intermediate BRISQUE parameters for Bicycle GAN might be the patch size, since the original paper presents results on images of $256 \times 256 \times 3$. The results in this work were calculated with the to RGB converted HSI patches which thus have 16 times less MSCN statistics values, likely also explaining the error in Bicycle GAN result computation.

Comparisons with results from literature show no large deviations, as already visible from the qualitative results. When comparing PSNR with mostly superresolution tasks [80, 81], achieved results lay in between results obtained from literature, which depended on the dataset and varied on an error scale of 6 dB - a factor of four in MSE. Also, hyperspectral SSIM results were

comparable to the results from the same works of literature [80, 81].
Especially when comparing to results obtained from medical datasets [2, 75], the KID score is comparable for the Sobol-sample reconstructions and better for the paired generated patches. FID and IS are not directly discussed due to their unreliability, which also can be seen for the MUNIT results [48] with a low IS, even though MUNIT achieved remarkable visual results.

Conclusions Presented IQA metrics evaluated textural realism of the generated HSI patches to be comparable to state of the art work. At the same time, qualitative visualizations showed missing or inaccurate finer structures such as vessels or camera-specific patterns. Therefore, two conclusions can be drawn:
Similar frameworks applied the RGB data suffer from similar problems, implying that further investigations should focus on trying out different modalities rather than improving single models or architectures. This especially aims at more recent deep learning approaches such as neural rendering [2] and transformers which utilize a learned vocabulary [122]. Both promise high quality and visually consisted results and could incorporate several 'layers' of one image for better results, as mentioned in *Imaging Effect Analysis*.
Second, IQA metrics need to be handled and compared with care. Said metrics were often reported sloppily in literature but themself require attention and more interpretation, as they often crucially depend on features of (retrained) neural networks and underlying dataset.

Downstream Task

Training a downstream segmentation task with an artificially shrunken, real data training set, which was enlarged by generated reconstructions or postprocessed patches, did not improve test dice score results. As to be expected, the test results when training with purely reconstructed or postprocessed data were even worse. When using the vocabulary of Gontijo-Lopes et al. [120] this is to be expected due to a combination of two reasons: The generated HSI data exhibited e.g. visual differences from real HSI patches, which means that the affinity was low. At the same time, the generated patch content was restricted to already existing HSI patches which hence does not increase diversity. According to their [120] work, this kind of dataset augmentation empirically does not improve generated (test) results. The restriction to existing patches was required, since segmentation maps were needed to train the segmentation network and the pipeline contained no way to synthesize these label maps.
Using the downstream task as realism measure, intricate results obtained from the spaghetti plots in Figure 33 were, that the dice score decreased different for the different pigs. For the pig with the highest dice score, it decreases more slowly, while for the other four pigs it dropped by a larger margin. The also observed large spread in individual pig results once more depicts the difference between pigs, which was already observed as cause for large errors of the quantitative spectral and embedding results. When trying to associate the individual differences with observed differences for IQA metrics, no clear correlation was seen, since rankings by the different IQA metrics were ambiguous. Therefore, no clear answer can be given, on which specific aspect of the generated results within the proposed pipeline has to be improved for better affinity, especially as visually dissimilar WAE and postprocessing results returned similar segmentation performance on the test dataset.

Conclusions The conclusion from the conducted experiment is that a data bottleneck for this case of a labelling task should be tackled by a task-specific model, which also alternates the ground truth labels as e.g. proposed for registration and segmentation in literature [123, 124].

Outlook

Regarding the implemented model, additional tuning and investigation could be undertaken, specifically for more sophisticated loss terms [2] and further examining and tuning the latent space as the variance parameter of the WAE was kept default. With the comprehensive results, obtained from the different experiments; however, the expected return of this is rather small as issues appeared for ill-posed cases with e.g. bad illumination. Therefore trials with new frameworks should be preferred.

Such frameworks comprise the StyleGAN2 with adaptive discriminator augmentations (ADA) which is designed especially for working with sparse data. A HSI adapted version was already implemented but not reported, as it requires further tuning of hyperparameters to not result in mode collapse. While such GAN-based approaches are supposed to have lower diversity than the here implemented WAE, they are expected to increase the affinity, which could aid in deficient cases. However, this approach does not incorporate label maps or further ground truth properties and higher affinity with the current pipeline could be obtained by e.g. rejection sampling.
The MUNIT framework would be an example for a new framework, which can alter segmentation maps to e.g. provide segmentation tasks with high diversity synthetic data. A similar implementation for simultaneous generation of label maps and HSI patches was tested initially, but the quality of segmentation maps was found to be insufficient. While newer works in data augmentation [120, 123] see less of a problem in this, MUNIT only incorporates one additional feature, namely label maps, and is not explicitly designed to disentangle different 'layers' of the image which could further be leveraged to improve generation performance.

Therefore, model-based deep-learning [16], neural rendering approaches [2] or generative radiance fields [125] are the probably best next implementations to test, since they allow incorporation of much more (physiological) details, at the cost of having to provide this data. As the used datasets of the first two publications [2, 16] are publicly available, HSI image generation could be approached in a similar unpaired, rendering manner with already existing model data. The neural rendering approach would also allow for incorporation of further conditioning input parameters such as tissue oxygenation maps to bridge the gap to physiological parameter learning, which further gives the possibility to provide downstream physiological parameter extraction tasks with ground truth label information.

Bibliography

[1] N. T. Clancy, G. Jones, L. Maier-Hein, D. S. Elson, and D. Stoyanov, "Surgical spectral imaging," *Medical Image Analysis*, vol. 63, p. 101699, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841520300645

[2] D. Rivoir, M. Pfeiffer, R. Docea, F. Kolbinger, C. Riediger, J. Weitz, and S. Speidel, "Long-term temporally consistent unpaired video translation from simulated surgical 3d data," 2021.

[3] S. Wirkert, A. Vemuri, H. Kenngott, S. Moccia, M. Götz, B. Mayer, K. Maier-Hein, D. Elson, and L. Maier-Hein, "Physiological parameter estimation from multispectral images unleashed," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2017*, 09 2017, pp. 134–141.

[4] S. Seidlitz, J. Sellner, J. Odenthal, B. Özdemir, A. Studier-Fischer, S. Knödler, L. Ayala, T. Adler, H. G. Kenngott, M. Tizabi, M. Wagner, F. Nickel, B. P. Müller-Stich, and L. Maier-Hein, "Robust deep learning-based semantic organ segmentation in hyperspectral images," 2021.

[5] D. Nepogodiev, J. Martin, B. Biccard, A. Makupe, A. Bhangu, D. Nepogodiev, J. Martin, B. Biccard, A. Makupe, A. Ademuyiwa, A. O. Adisa, M.-L. Aguilera, S. Chakrabortee, J. E. Fitzgerald, D. Ghosh, J. C. Glasbey, E. M. Harrison, J. A. Ingabire, H. Salem, M. C. Lapitan, I. Lawani, D. Lissauer, L. Magill, R. Moore, D. C. Osei-Bordom, T. D. Pinkney, A. U. Qureshi, A. Ramos-De la Medina, S. Rayne, S. Sundar, S. Tabiri, A. Verjee, R. Yepez, O. J. Garden, R. Lilford, P. Brocklehurst, D. G. Morton, and A. Bhangu, "Global burden of postoperative death," *The Lancet*, vol. 393, no. 10170, p. 401, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140673618331398

[6] G. H. Jacobs, "Primate photopigments and primate color vision," *Proceedings of the National Academy of Sciences*, vol. 93, no. 2, pp. 577–581, 1996. [Online]. Available: https://www.pnas.org/content/93/2/577

[7] I. V. Meglinski and S. J. Matcher, "Quantitative assessment of skin layers absorption and skin reflectance spectra simulation in the visible and near-infrared spectral regions," *Physiological Measurement*, vol. 23, no. 4, pp. 741–753, oct 2002. [Online]. Available: https://doi.org/10.1088/0967-3334/23/4/312

[8] L. V. Wang and H.-i. Wu, *Biomedical optics.* Hoboken, NJ: Wiley-Interscience, 2007.

[9] A. Holmer, J. Marotz, P. Wahl, M. Dau, and P. W. Kämmerer, "Hyperspectral imaging in perfusion and wound diagnostics – methods and algorithms for the determination of tissue parameters," *Biomedical Engineering / Biomedizinische Technik*, vol. 63, no. 5, pp. 547–556, 2018. [Online]. Available: https://doi.org/10.1515/bmt-2017-0155

[10] M. Van Gemert, S. Jacques, H. Sterenborg, and W. Star, "Skin optics," *IEEE Transactions on Biomedical Engineering*, vol. 36, no. 12, pp. 1146–1154, 1989.

[11] E. J. M. Baltussen, E. N. D. Kok, S. G. B. de Koning, J. Sanders, A. G. J. Aalbers, N. F. M. Kok, G. L. Beets, C. C. Flohil, S. C. Bruin, K. F. D. Kuhlmann, H. J. C. M. Sterenborg, and T. J. M. Ruers, "Hyperspectral imaging for tissue classification, a way toward smart laparoscopic colorectal surgery," *Journal of Biomedical Optics*, vol. 24, no. 1, pp. 1 – 9, 2019. [Online]. Available: https://doi.org/10.1117/1.JBO.24.1.016002

[12] E. Torti, G. Florimbi, F. Castelli, S. Ortega, H. Fabelo, G. M. Callicó, M. Marrero-Martin, and F. Leporati, "Parallel k-means clustering for brain cancer detection using hyperspectral images," *Electronics*, vol. 7, no. 11, 2018. [Online]. Available: https://www.mdpi.com/2079-9292/7/11/283

[13] Y. Khouj, J. Dawson, J. Coad, and L. Vona-Davis, "Hyperspectral imaging and k-means classification for histologic evaluation of ductal carcinoma in situ," *Frontiers in Oncology*, vol. 8, p. 17, 2018. [Online]. Available: https://www.frontiersin.org/article/10.3389/fonc.2018.00017

[14] S. J. Wirkert, H. Kenngott, B. Mayer, P. Mietkowski, M. Wagner, P. Sauer, N. T. Clancy, D. S. Elson, and L. Maier-Hein, "Robust near real-time estimation of physiological parameters from megapixel multispectral images with inverse monte carlo and random forest regression," *International Journal of Computer Assisted Radiology and Surgery*, vol. 11, no. 6, pp. 909–917, Jun 2016. [Online]. Available: https://doi.org/10.1007/s11548-016-1376-5

[15] J. T. Alander, I. Kaartinen, A. Laakso, T. Pätilä, T. Spillmann, V. V. Tuchin, M. Venermo, and P. Välisuo, "A review of indocyanine green fluorescent imaging in surgery," *International journal of biomedical imaging*, vol. 2012, pp. 940 585–940 585, 2012. [Online]. Available: https://doi.org/10.1155/2012/940585

[16] M. Pfeiffer, I. Funke, M. R. Robu, S. Bodenstedt, L. Strenger, S. Engelhardt, T. Roß, M. J. Clarkson, K. Gurusamy, B. R. Davidson, L. Maier-Hein, C. Riediger, T. Welsch, J. Weitz, and S. Speidel, "Generating large labeled data sets for laparoscopic image processing tasks using unpaired image-to-image translation," 2019.

[17] R. J. Chen, M. Y. Lu, T. Y. Chen, D. F. K. Williamson, and F. Mahmood, "Synthetic data in machine learning for medicine and healthcare," *Nature Biomedical Engineering*, vol. 5, no. 6, pp. 493–497, Jun 2021. [Online]. Available: https://doi.org/10.1038/s41551-021-00751-8

[18] A. Marzullo, S. Moccia, M. Catellani, F. Calimeri, and E. D. Momi, "Towards realistic laparoscopic image generation using image-domain translation," *Computer Methods and Programs in Biomedicine*, vol. 200, p. 105834, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260720316679

[19] G. Lu and B. Fei, "Medical hyperspectral imaging: a review," *Journal of Biomedical Optics*, vol. 19, no. 1, pp. 1 – 24, 2014. [Online]. Available: https://doi.org/10.1117/1.JBO.19.1.010901

[20] T. J. Farrell, B. C. Wilson, and M. S. Patterson, "The use of a neural network to determine tissue optical properties from spatially resolved diffuse reflectance

measurements," *Physics in Medicine and Biology*, vol. 37, no. 12, pp. 2281–2286, dec 1992. [Online]. Available: https://doi.org/10.1088/0031-9155/37/12/009

[21] S. Prahl, "Tabulated molar extinction coefficient for hemoglobin in water, compiled by S. Prahl," https://omlc.org/spectra/hemoglobin/summary.html, OMLC, retrieved on 20. Oct. 2021.

[22] G. M. Hale and M. R. Querry, "Optical constants of water in the 200-nm to 200-μm wavelength region," *Appl. Opt.*, vol. 12, no. 3, pp. 555–563, Mar 1973. [Online]. Available: http://www.osapublishing.org/ao/abstract.cfm?URI=ao-12-3-555

[23] S. L. Jacques and D. J. McAuliffe, "The melanosome: Threshold temperature for explosive vaporization and internal absorption coefficient during pulsed laser irradiation," *Photochemistry and Photobiology*, vol. 53, no. 6, pp. 769–775, 1991. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1751-1097.1991.tb09891.x

[24] S. Jacques, "Generic tissue optical properties," https://omlc.org/news/feb15/generic_optics/index.html, OMLC, retrieved on 20. Oct. 2021.

[25] S. A. Prahl, "A Monte Carlo model of light propagation in tissue," in *Dosimetry of Laser Radiation in Medicine and Biology*, G. J. Mueller, D. H. Sliney, and R. F. Potter, Eds., vol. 10305, International Society for Optics and Photonics. SPIE, 1989, pp. 105 – 114. [Online]. Available: https://doi.org/10.1117/12.2283590

[26] G. A. Reider, *Nonlinear Optics and Acousto-Optics.* Cham: Springer International Publishing, 2016, ch. 8, pp. 351–412. [Online]. Available: https://doi.org/10.1007/978-3-319-26076-1_8

[27] V. Kavvadias, G. Epitropou, N. Georgiou, F. Grozou, M. Paschopoulos, and C. Balas, "A novel endoscopic spectral imaging platform integrating k-means clustering for early and non-invasive diagnosis of endometrial pathology," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 4442–4445.

[28] K. J. Zuzak, R. P. Francis, E. F. Wehner, M. Litorja, J. A. Cadeddu, and E. H. Livingston, "Active dlp hyperspectral illumination: A noninvasive, in vivo, system characterization visualizing tissue oxygenation at near video rates," *Analytical Chemistry*, vol. 83, no. 19, pp. 7424–7430, Oct 2011. [Online]. Available: https://doi.org/10.1021/ac201467v

[29] B. Neumann, "Bildsensoren," *Bildverarbeitung für Einsteiger: Programmbeispiele mit Mathcad*, pp. 339–340, 2005.

[30] B. E. Bayer, "Color imaging array," Jul. 20 1976, uS Patent 3,971,065.

[31] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning.* MIT press, 2016.

[32] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[33] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0893608089900208

[34] P. Mehta, M. Bukov, C.-H. Wang, A. G. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, "A high-bias, low-variance introduction to machine learning for physicists," *Physics Reports*, vol. 810, pp. 1–124, 2019, a high-bias, low-variance introduction to Machine Learning for physicists. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0370157319300766

[35] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning.* PMLR, 2015, pp. 448–456.

[36] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *CoRR*, vol. abs/1607.08022, 2016. [Online]. Available: http://arxiv.org/abs/1607.08022

[37] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," *CoRR*, vol. abs/1710.05941, 2017. [Online]. Available: http://arxiv.org/abs/1710.05941

[38] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," 2018.

[39] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," 2018.

[40] M. A. Nielsen, *Neural networks and deep learning.* Determination press San Francisco, CA, 2015, vol. 25.

[41] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct 1986. [Online]. Available: https://doi.org/10.1038/323533a0

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[43] L. Wang, S. L. Jacques, and L. Zheng, "Mcml—monte carlo modeling of light transport in multi-layered tissues," *Computer Methods and Programs in Biomedicine*, vol. 47, no. 2, pp. 131–146, 1995. [Online]. Available: https://www.sciencedirect.com/science/article/pii/016926079501640F

[44] G. Jones, N. T. Clancy, Y. Helo, S. Arridge, D. S. Elson, and D. Stoyanov, "Bayesian estimation of intrinsic tissue oxygenation and perfusion from rgb images," *IEEE Transactions on Medical Imaging*, vol. 36, no. 7, pp. 1491–1501, 2017.

[45] M. L. De Jode, "Monte carlo simulations of light distributions in an embedded tumour model: Studies of selectivity in photodynamic therapy," *Lasers in Medical Science*, vol. 15, no. 1, pp. 49–56, Jan 2000. [Online]. Available: https://doi.org/10.1007/s101030050047

[46] L. A. Ayala, S. J. Wirkert, J. Gröhl, M. A. Herrera, A. Hernandez-Aguilera, A. Vemuri, E. Santos, and L. Maier-Hein, "Live monitoring of haemodynamic changes with multispectral image analysis," in *OR 2.0 Context-Aware Operating Theaters and Machine*

Learning in Clinical Neuroimaging, L. Zhou, D. Sarikaya, S. M. Kia, S. Speidel, A. Malpani, D. Hashimoto, M. Habes, T. Löfstedt, K. Ritter, and H. Wang, Eds. Cham: Springer International Publishing, 2019, pp. 38–46.

[47] V. Periyasamy and M. Pramanik, "Advances in monte carlo simulation for light propagation in tissue," *IEEE reviews in biomedical engineering*, vol. 10, pp. 122–135, 2017.

[48] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," 2018.

[49] J. Thies, M. Zollhöfer, and M. Nießner, "Deferred neural rendering: Image synthesis using neural textures," 2019.

[50] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[51] S. Moccia, S. Wirkert, H. Kenngott, A. Vemuri, M. Apitz, B. Mayer, E. De Momi, L. Mattos, and L. Maier-Hein, "Uncertainty-aware organ classification for surgical data science applications in laparoscopy," *IEEE Transactions on Biomedical Engineering*, vol. PP, 06 2017.

[52] M. Schellenberg, J. Gröhl, K. Dreher, N. Holzwarth, M. D. Tizabi, A. Seitel, and L. Maier-Hein, "Data-driven generation of plausible tissue geometries for realistic photoacoustic image synthesis," 2021.

[53] A. J. Plassard, L. T. Davis, A. T. Newton, S. M. Resnick, B. A. Landman, and C. Bermudez, "Learning implicit brain mri manifolds with deep learning," *Medical Imaging 2018: Image Processing*, Mar 2018. [Online]. Available: http://dx.doi.org/10.1117/12.2293515

[54] M. Maspero, M. H. F. Savenije, A. M. Dinkla, P. R. Seevinck, M. P. W. Intven, I. M. Jurgenliemk-Schulz, L. G. W. Kerkmeijer, and C. A. T. van den Berg, "Dose evaluation of fast synthetic-CT generation using a generative adversarial network for general pelvis MR-only radiotherapy," *Phys Med Biol*, vol. 63, no. 18, p. 185001, 09 2018.

[55] J. M. Wolterink, A. M. Dinkla, M. H. F. Savenije, P. R. Seevinck, C. A. T. van den Berg, and I. Išgum, "Deep mr to ct synthesis using unpaired data," in *Simulation and Synthesis in Medical Imaging*, S. A. Tsaftaris, A. Gooya, A. F. Frangi, and J. L. Prince, Eds. Cham: Springer International Publishing, 2017, pp. 14–23.

[56] 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)," 2017.

[57] Y. Pan, M. Liu, C. Lian, T. Zhou, Y. Xia, and D. Shen, "Synthesizing missing pet from mri with cycle-consistent generative adversarial networks for alzheimer's disease diagnosis," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, A. Frangi, J. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, Eds., vol. 11072. Cham: Springer, 2018, pp. 455–463.

[58] P. Welander, S. Karlsson, and A. Eklund, "Generative adversarial networks for image-to-image translation on multi-contrast mr images - a comparison of cyclegan and unit," 2018.

[59] X. Yi, E. Walia, and P. Babyn, "Generative adversarial network in medical imaging: A review," *Medical Image Analysis*, vol. 58, p. 101552, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841518308430

[60] X. Yi, E. Walia, and P. Babyn, "Unsupervised and semi-supervised learning with categorical generative adversarial networks assisted by wasserstein distance for dermoscopy image classification," *arXiv preprint arXiv:1804.03700*, 2018.

[61] Z. Qin, Z. Liu, P. Zhu, and Y. Xue, "A gan-based image synthesis method for skin lesion classification," *Computer Methods and Programs in Biomedicine*, vol. 195, p. 105568, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260720302418

[62] J. Lin, N. T. Clancy, J. Qi, Y. Hu, T. Tatla, D. Stoyanov, L. Maier-Hein, and D. S. Elson, "Dual-modality endoscopic probe for tissue surface shape reconstruction and hyperspectral imaging enabled by deep neural networks," *Medical Image Analysis*, vol. 48, pp. 162–176, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841518303736

[63] Q. Li, J. Lin, N. T. Clancy, and D. S. Elson, "Estimation of tissue oxygen saturation from rgb images and sparse hyperspectral signals based on conditional generative adversarial network," *International Journal of Computer Assisted Radiology and Surgery*, vol. 14, no. 6, p. 987–995, Mar 2019. [Online]. Available: http://dx.doi.org/10.1007/s11548-019-01940-2

[64] A. Tewari, O. Fried, J. Thies, V. Sitzmann, S. Lombardi, K. Sunkavalli, R. Martin-Brualla, T. Simon, J. Saragih, M. Nießner, R. Pandey, S. Fanello, G. Wetzstein, J.-Y. Zhu, C. Theobalt, M. Agrawala, E. Shechtman, D. B. Goldman, and M. Zollhöfer, "State of the art on neural rendering," 2020.

[65] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.

[66] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.

[67] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf, "Wasserstein auto-encoders," 2019.

[68] L. Ardizzone, C. Lüth, J. Kruse, C. Rother, and U. Köthe, "Guided image generation with conditional invertible neural networks," 2019.

[69] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2016.

[70] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[72] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

[73] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, "Visualizing the loss landscape of neural nets," 2018.

[74] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," 2019.

[75] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," *CoRR*, vol. abs/2006.06676, 2020. [Online]. Available: https://arxiv.org/abs/2006.06676

[76] I. Bello, W. Fedus, X. Du, E. D. Cubuk, A. Srinivas, T. Lin, J. Shlens, and B. Zoph, "Revisiting resnets: Improved training and scaling strategies," *CoRR*, vol. abs/2103.07579, 2021. [Online]. Available: https://arxiv.org/abs/2103.07579

[77] L. Liebel and M. Körner, "Single-image super resolution for multispectral remote sensing data using convolutional neural networks," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLI-B3, pp. 883–890, 2016. [Online]. Available: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLI-B3/883/2016/

[78] O. Sidorov and J. Y. Hardeberg, "Deep hyperspectral prior: Single-image denoising, inpainting, super-resolution," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 3844–3851.

[79] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[80] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," 2017.

[81] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," *CoRR*, vol. abs/2006.04139, 2020. [Online]. Available: https://arxiv.org/abs/2006.04139

[82] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[83] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, "Image quality assessment: Unifying structure and texture similarity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1–1, 2020. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2020.3045810

[84] M. J. Chong and D. A. Forsyth, “Effectively unbiased FID and inception score and where to find them,” *CoRR*, vol. abs/1911.07023, 2019. [Online]. Available: http://arxiv.org/abs/1911.07023

[85] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” 2018.

[86] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, “Demystifying mmd gans,” 2021.

[87] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” 2015.

[88] C. Ma, C.-Y. Yang, X. Yang, and M.-H. Yang, “Learning a no-reference quality metric for single-image super-resolution,” *Computer Vision and Image Understanding*, vol. 158, pp. 1–16, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S107731421630203X

[89] S. Barratt and R. Sharma, “A note on the inception score,” 2018.

[90] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans,” 2016.

[91] J. P. Cohen, M. Luck, and S. Honari, “Distribution matching losses can hallucinate features in medical image translation,” *CoRR*, vol. abs/1805.08841, 2018. [Online]. Available: http://arxiv.org/abs/1805.08841

[92] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, “Toward multimodal image-to-image translation,” 2018.

[93] A. N. Bashkatov, E. A. Genina, V. I. Kochubey, and V. V. Tuchin, “Optical properties of human skin, subcutaneous and mucous tissues in the wavelength range from 400 to 2000 nm,” *Journal of Physics D: Applied Physics*, vol. 38, no. 15, pp. 2543–2555, jul 2005. [Online]. Available: https://doi.org/10.1088/0022-3727/38/15/004

[94] A. Kulcke, A. Holmer, P. Wahl, F. Siemers, T. Wild, and G. Daeschlein, “A compact hyperspectral camera for measurement of perfusion parameters in medicine,” *Biomedical Engineering / Biomedizinische Technik*, vol. 63, no. 5, pp. 519–527, 2018. [Online]. Available: https://doi.org/10.1515/bmt-2017-0145

[95] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, “Albumentations: Fast and flexible image augmentations,” *Information*, vol. 11, no. 2, 2020. [Online]. Available: https://www.mdpi.com/2078-2489/11/2/125

[96] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” 2017.

[97] W. Gangbo and R. J. McCann, “The geometry of optimal transportation,” *Acta Mathematica*, vol. 177, no. 2, pp. 113 – 161, 1996. [Online]. Available: https://doi.org/10.1007/BF02392620

[98] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, “A kernel two-sample test,” *Journal of Machine Learning Research*, vol. 13, no. 25, pp. 723–773, 2012. [Online]. Available: http://jmlr.org/papers/v13/gretton12a.html

[99] R. Wightman, “PyTorch Image Models,” https://github.com/rwightman/pytorch-image-models, 2019.

[100] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, “Loss functions for image restoration with neural networks,” *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, 2017.

[101] S. Reddi, A. Ramdas, B. Poczos, A. Singh, and L. Wasserman, “On the High Dimensional Power of a Linear-Time Two Sample Test under Mean-shift Alternatives,” in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Lebanon and S. V. N. Vishwanathan, Eds., vol. 38. San Diego, California, USA: PMLR, 09–12 May 2015, pp. 772–780. [Online]. Available: https://proceedings.mlr.press/v38/reddi15.html

[102] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, “The marginal value of adaptive gradient methods in machine learning,” 2018.

[103] E. Schönfeld, B. Schiele, and A. Khoreva, “A u-net based discriminator for generative adversarial networks,” 2021.

[104] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, “High-resolution image synthesis and semantic manipulation with conditional gans,” 2018.

[105] X. Hou, K. Sun, L. Shen, and G. Qiu, “Improving variational autoencoder with deep feature consistent and generative adversarial training,” *Neurocomputing*, vol. 341, p. 183–194, May 2019. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2019.03.013

[106] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[107] M.-Y. Liu, T. Breuel, and J. Kautz, “Unsupervised image-to-image translation networks,” 2018.

[108] S. Kastryulin, D. Zakirov, and D. Prokopenko, “PyTorch Image Quality: Metrics and measure for image quality assessment,” 2019, open-source software available at https://github.com/photosynthesis-team/piq. [Online]. Available: https://github.com/photosynthesis-team/piq

[109] S. Zhao, J. Cui, Y. Sheng, Y. Dong, X. Liang, E. I. Chang, and Y. Xu, “Large scale image completion via co-modulated generative adversarial networks,” 2021.

[110] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” 2015.

[111] R. Liaw, E. Liang, R. Nishihara, P. Moritz, J. E. Gonzalez, and I. Stoica, “Tune: A research platform for distributed model selection and training,” *arXiv preprint arXiv:1807.05118*, 2018.

[112] I. Loshchilov and F. Hutter, “Sgdr: Stochastic gradient descent with warm restarts,” 2017.

[113] H. Niederreiter, “Low-discrepancy and low-dispersion sequences,” *Journal of Number Theory*, vol. 30, no. 1, pp. 51–70, 1988. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0022314X8890025X

[114] G. E. P. Box and M. E. Muller, “A Note on the Generation of Random Normal Deviates,” *The Annals of Mathematical Statistics*, vol. 29, no. 2, pp. 610 – 611, 1958. [Online]. Available: https://doi.org/10.1214/aoms/1177706645

[115] G. Ökten and A. Göncü, “Generating low-discrepancy sequences from the normal distribution: Box–muller or inverse transform?” *Mathematical and Computer Modelling*, vol. 53, no. 5, pp. 1268–1281, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0895717710005935

[116] L. McInnes, J. Healy, and J. Melville, “Umap: Uniform manifold approximation and projection for dimension reduction,” 2020.

[117] K. Shoemake, “Animating rotation with quaternion curves,” in *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, 1985, pp. 245–254.

[118] M. Marchesi, “Megapixel size image creation using generative adversarial networks,” 2017.

[119] M. Wiesenfarth, A. Reinke, B. A. Landman, M. Eisenmann, L. A. Saiz, M. J. Cardoso, L. Maier-Hein, and A. Kopp-Schneider, “Methods and open-source toolkit for analyzing and visualizing challenge results,” *Scientific Reports*, vol. 11, no. 1, p. 2369, Jan 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-82017-6

[120] R. Gontijo-Lopes, S. J. Smullin, E. D. Cubuk, and E. Dyer, “Affinity and diversity: Quantifying mechanisms of data augmentation,” 2020.

[121] R. Vershynin, *High-dimensional probability: An introduction with applications in data science.* Cambridge University Press, 2018, vol. 47.

[122] P. Esser, R. Rombach, and B. Ommer, “Taming transformers for high-resolution image synthesis,” 2021.

[123] M. Hoffmann, B. Billot, D. N. Greve, J. E. Iglesias, B. Fischl, and A. V. Dalca, “Synthmorph: learning contrast-invariant registration without acquired images,” *IEEE Transactions on Medical Imaging*, 2021.

[124] A. Zhao, G. Balakrishnan, F. Durand, J. V. Guttag, and A. V. Dalca, “Data augmentation using learned transformations for one-shot medical image segmentation,” 2019.

[125] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger, “Graf: Generative radiance fields for 3d-aware image synthesis,” 2021.

Part VI.
Appendix

1. Additional Training Results and Validation Plots

This section displays additional exhaustive WAE training data, shows results for different WAE decoder architectures and does a small ablation of the *pix2pix* network regarding the VGG loss and specific discriminator architecture.

Training Result: WAE Exhaustive Training

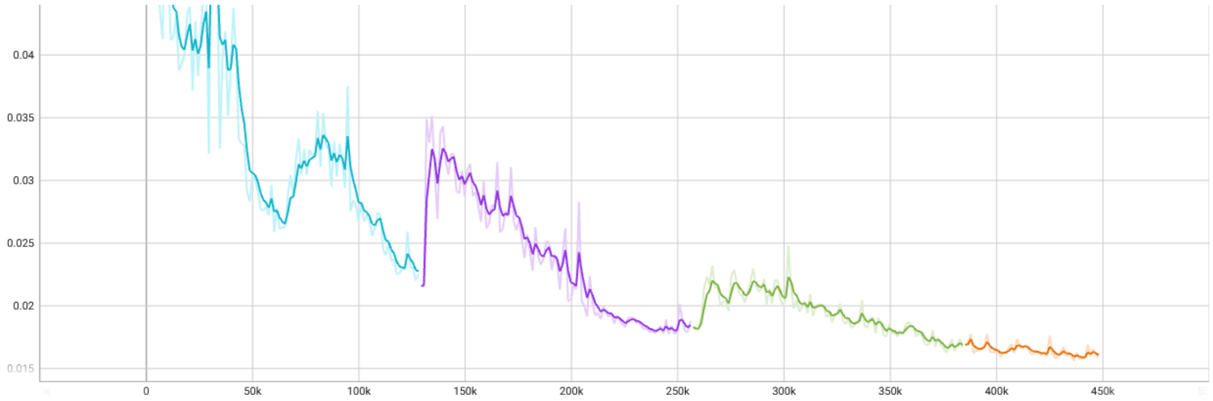
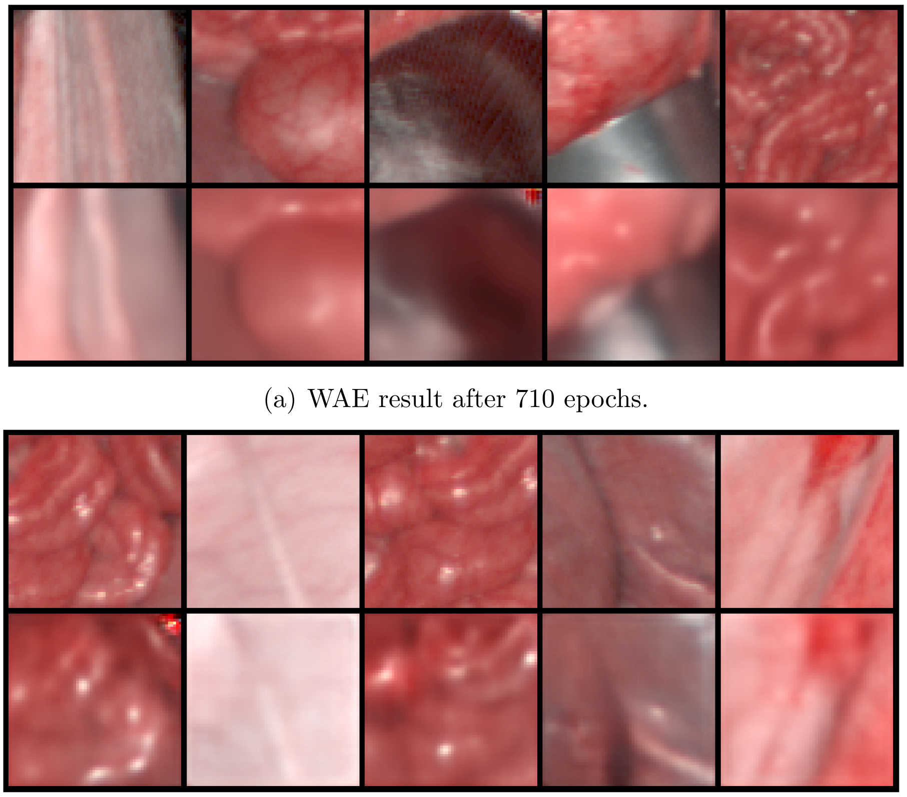


Figure 36: Validation loss of the WAE with latent dimensionality 512 after 500 epochs (light blue), 1000 epochs (purple), 1500 epochs (green) and 1750 epochs (orange). Bumps in validation loss come from the cosine learning rate scheduler. All presented WAE models are like the blue one trained for 500 epochs, visual results of higher epoch results following in the next figure.

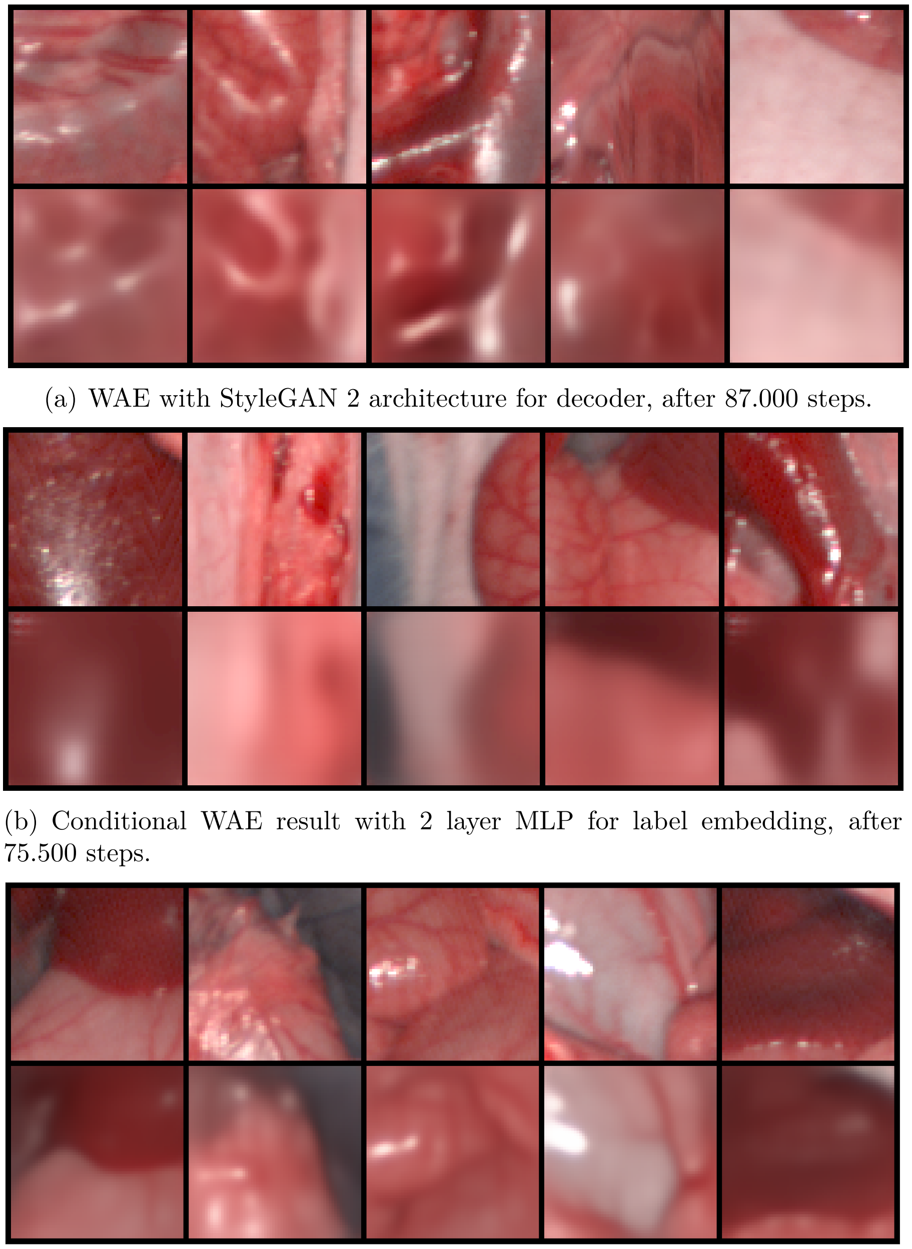


(a) WAE result after 710 epochs.

(b) WAE result after 1299 epochs.

Figure 37: Visual results slightly improve for longer training, however at the cost of visible droplet artefacts in the top right of some image patches. Finer structures are mostly still not visible. Real, to RGB converted patches in the top row, WAE reconstructions in the bottom row.

Training Result: WAE Different Decoder Results



(a) WAE with StyleGAN 2 architecture for decoder, after 87.000 steps.

(b) Conditional WAE result with 2 layer MLP for label embedding, after 75.500 steps.

(c) WAE trained on RGB data, after 128.000 steps.

Figure 38: Real, to RGB converted patches in the top row, WAE reconstructions in the bottom row. A different decoder architecture such as the famous StyleGAN 2 [75] does not provide better visual results within the WAE framework. The conditional approach was initially used on the $32 \times 32 \times 100$ beneath the unconditional approach, however the worse visual results on $64 \times 64 \times 100$ HSI patches led to the conditional approach being omitted completely. When training the WAE on RGB data, the visual results do not improve, showing both suitability of the framework for HSI data as well as the blurriness problem being inherent to the WAE rather than an effect of high spectral dimension and RGB conversion. This also reinforces the claim, that the deep learning models are applicable across different spectral data domains without loss in quality.

Training Result: VGG Ablation and Discriminator Ablation

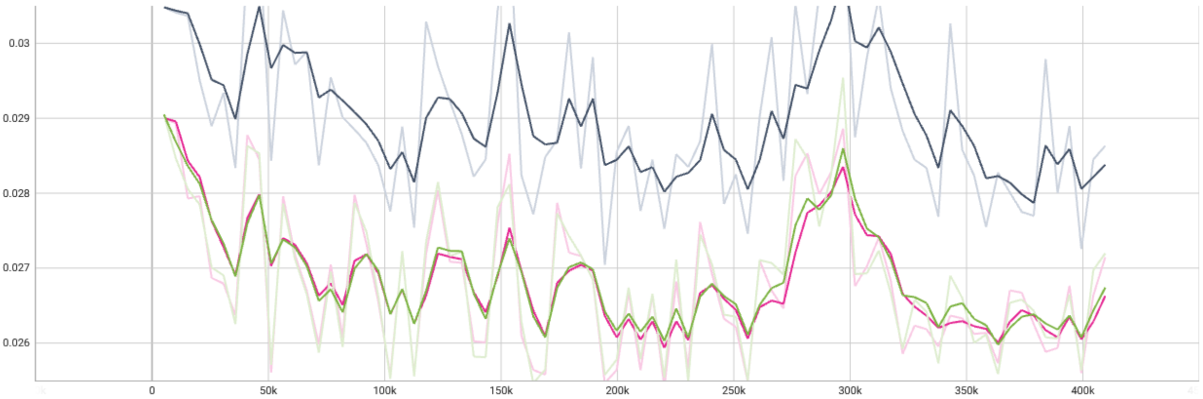
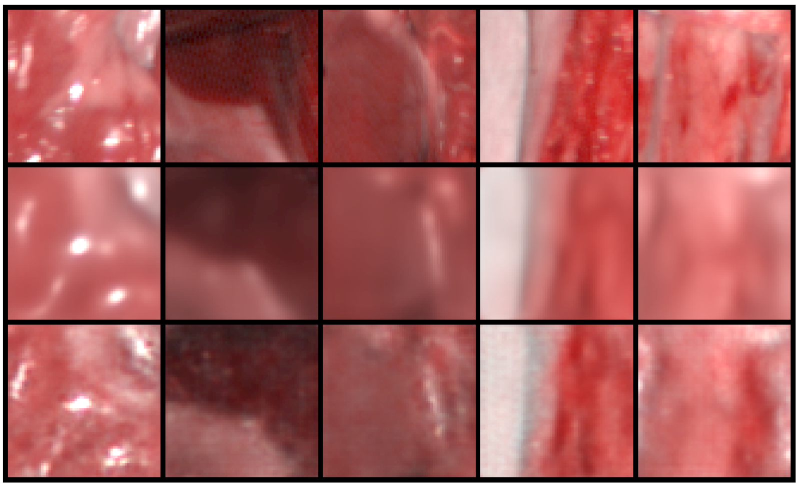
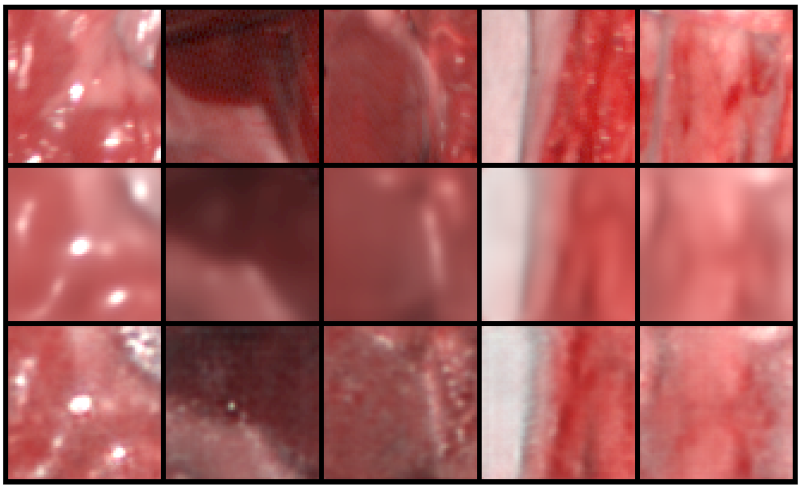


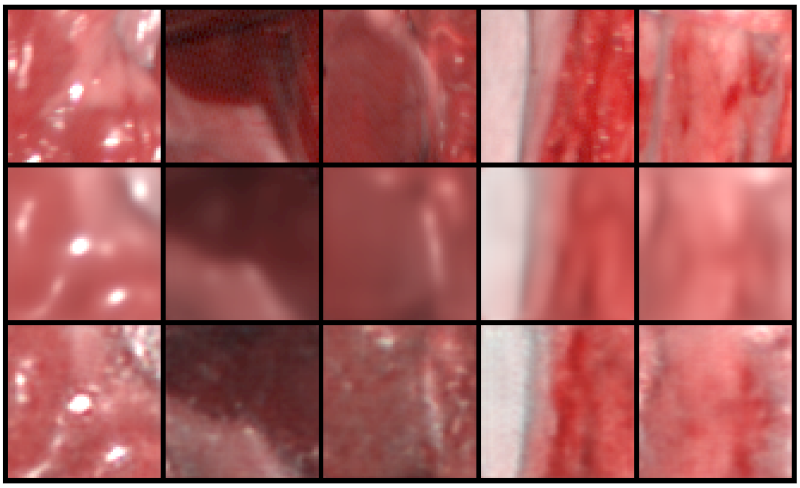
Figure 39: Validation loss of the different *pix2pix* approaches. Full setup as used in this work in dark blue, version with standard PatchGAN discriminator without concatenated inputs in green and version without result concatenation and without VGG loss term in pink. Concatenating results in the discriminator seems to provide additional regularization (higher loss), while the VGG loss with its small weight does not seem to affect the results at all. All versions trained for 400 epochs, visual results of the different versions in the next figure.



(a) *pix2pix* full setup.



(b) *pix2pix* without data concatenation for discriminator.



(c) *pix2pix* without data concatenation and without VGG loss.

Figure 40: Visual results of the network ablation after full training of 400 epochs. Real (top), reconstructed (middle) and *pix2pix* (bottom) patches in above's figure from the validation dataset. Results from the *pix2pix* full setup with concatenated input seems to be less noisy.

2. Additional Experiment Results

Additional plots of visual results, a spectral comparison to Monte Carlo samples, additional embedding latent space manipulations and more textural metric table data is given, including metric scores on training and validation set.

Imaging Effect Analysis: Additions

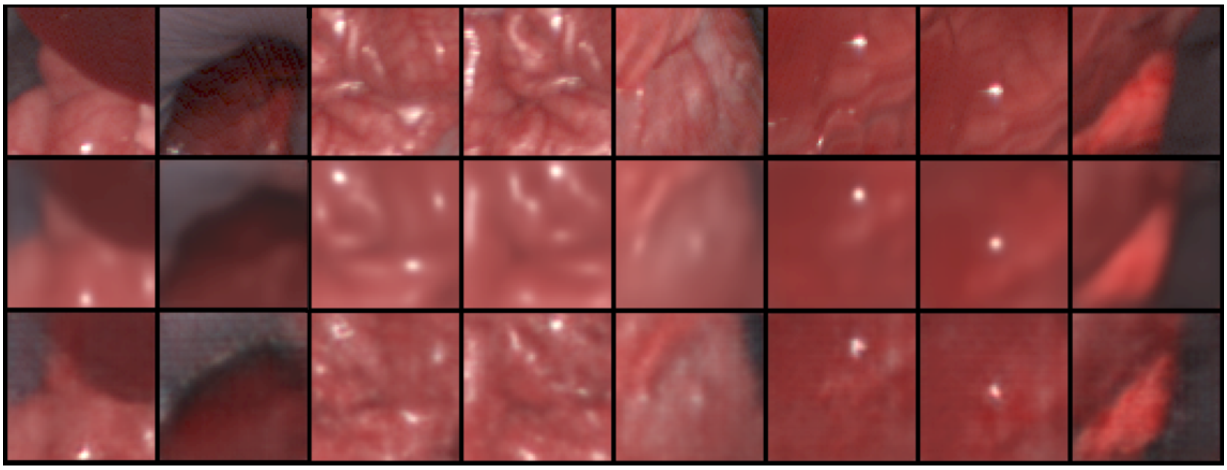


Figure 41: Example *pix2pix* postprocessing cases with similar issues as for Bicycle GAN: No meaningful vessel-mesh is generated, intricate details from some organs are missing and the camera-specific structure becomes a checkerboardish, edgy pattern.

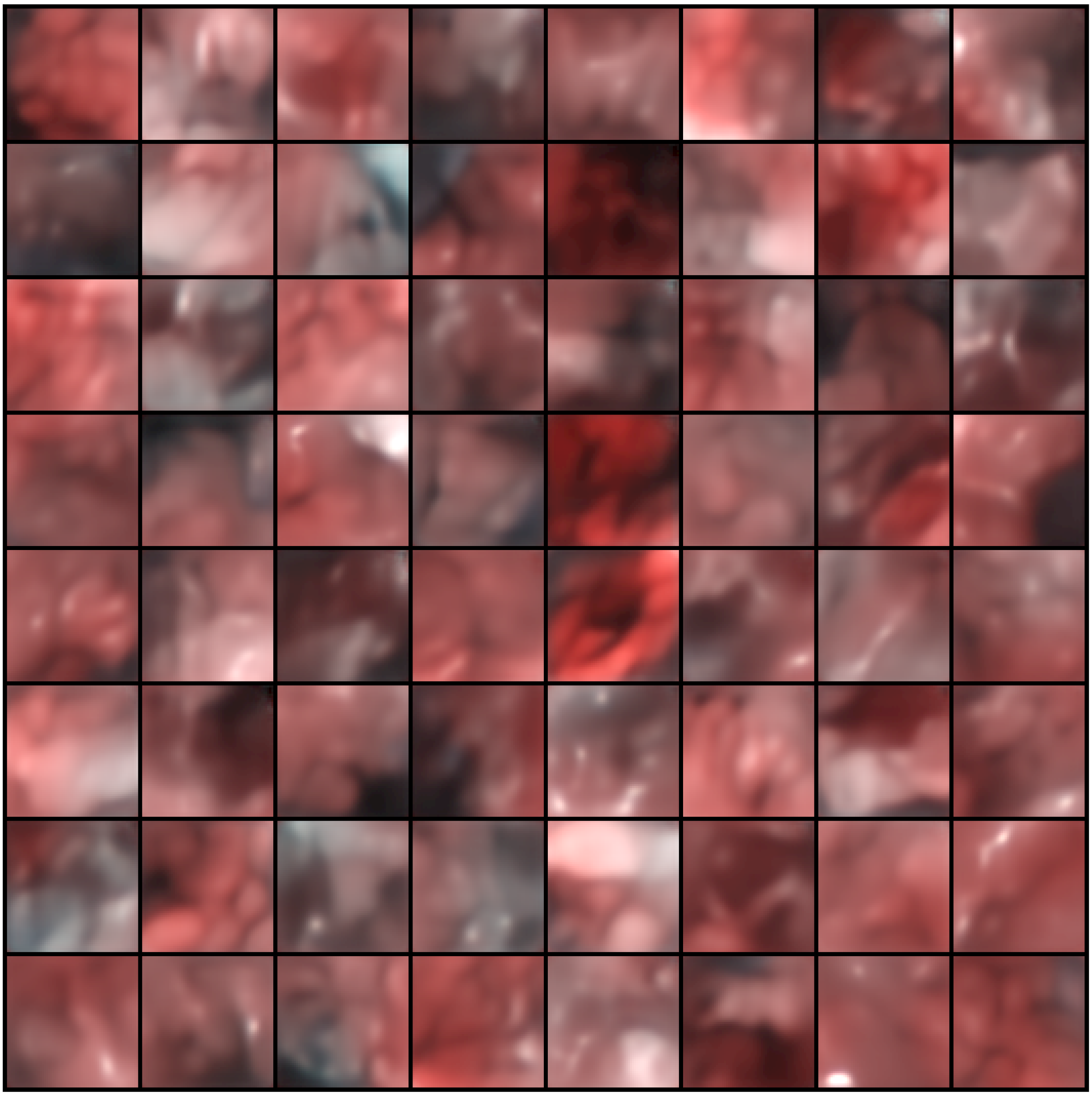


Figure 42: From 64 low-discrepancy Sobol samples decoded WAE results. Typical blurriness with high content diversity can be seen, sometimes the colouring becomes slightly unnatural (white/ blueish/ greenish).

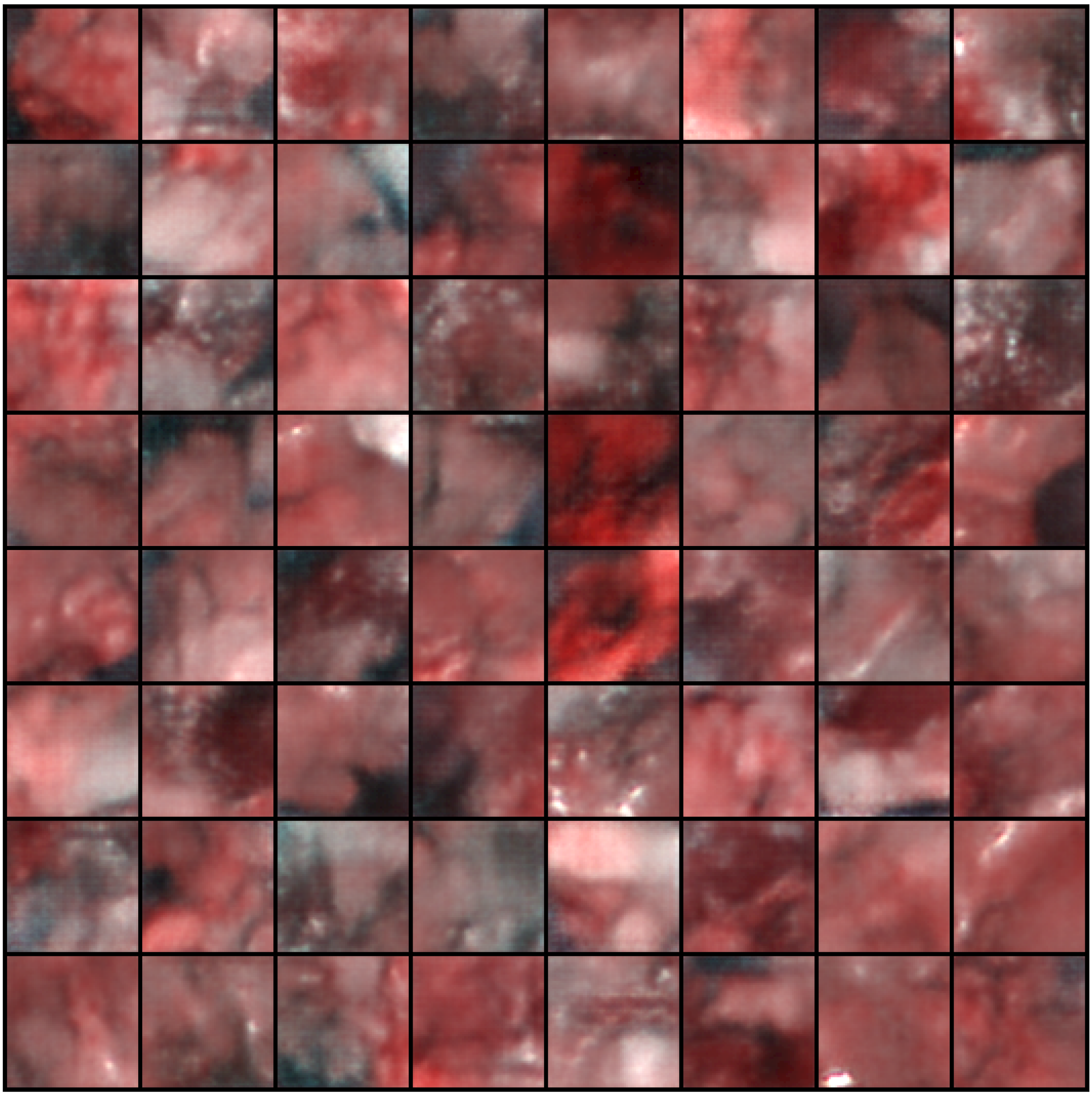


Figure 43: From 64 low-discrepancy Sobol samples decoded WAE results, postprocessed with the *pix2pix* approach. Results have mostly improved visually, however typical edgy patterns can be seen in some parts of the image patches.

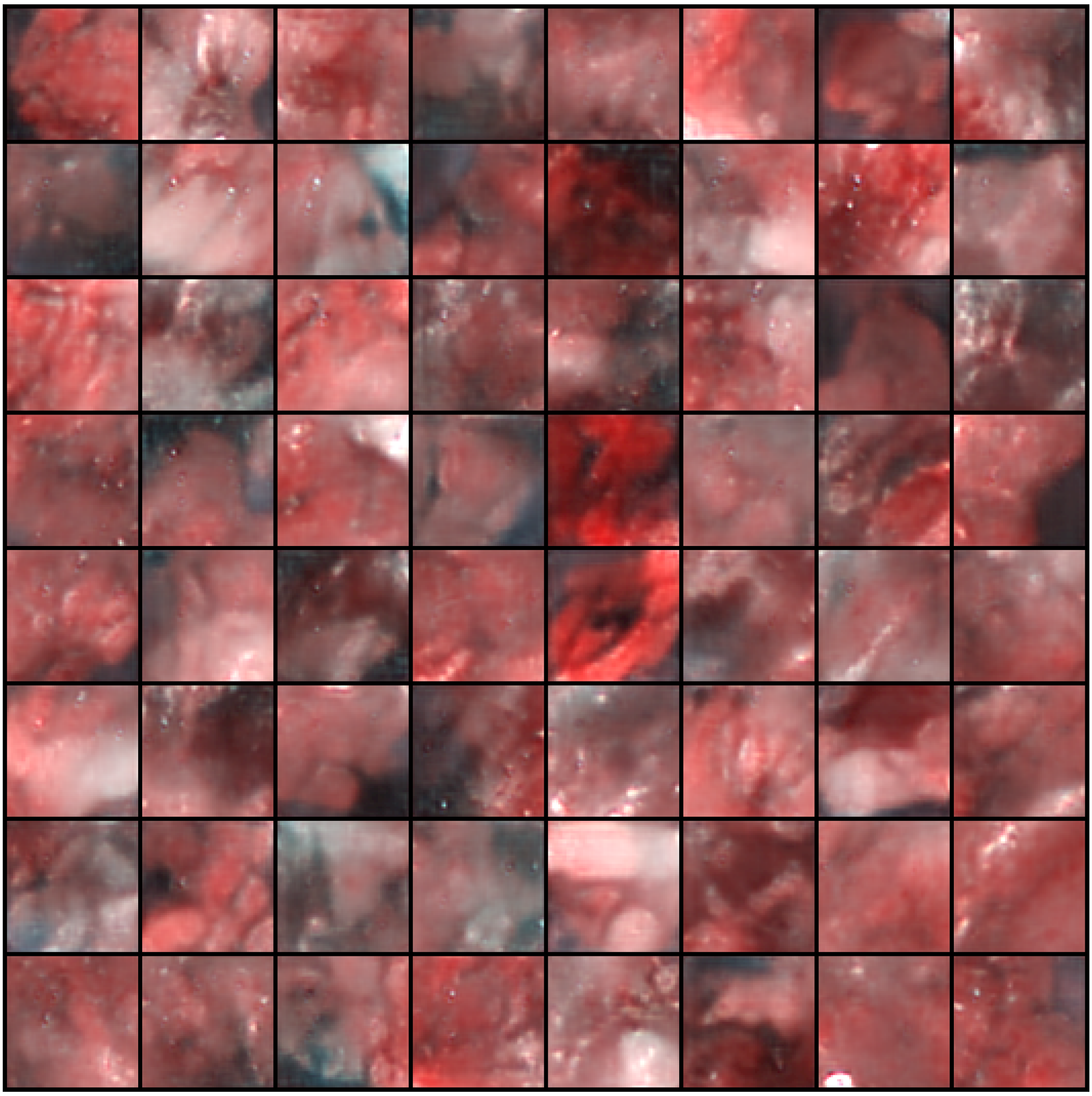


Figure 44: From 64 low-discrepancy Sobol samples decoded WAE results, postprocessed with Bicycle GAN. Results have much more realistic visual structure, shadows and illumination but also typical additional specular highlights can be seen.

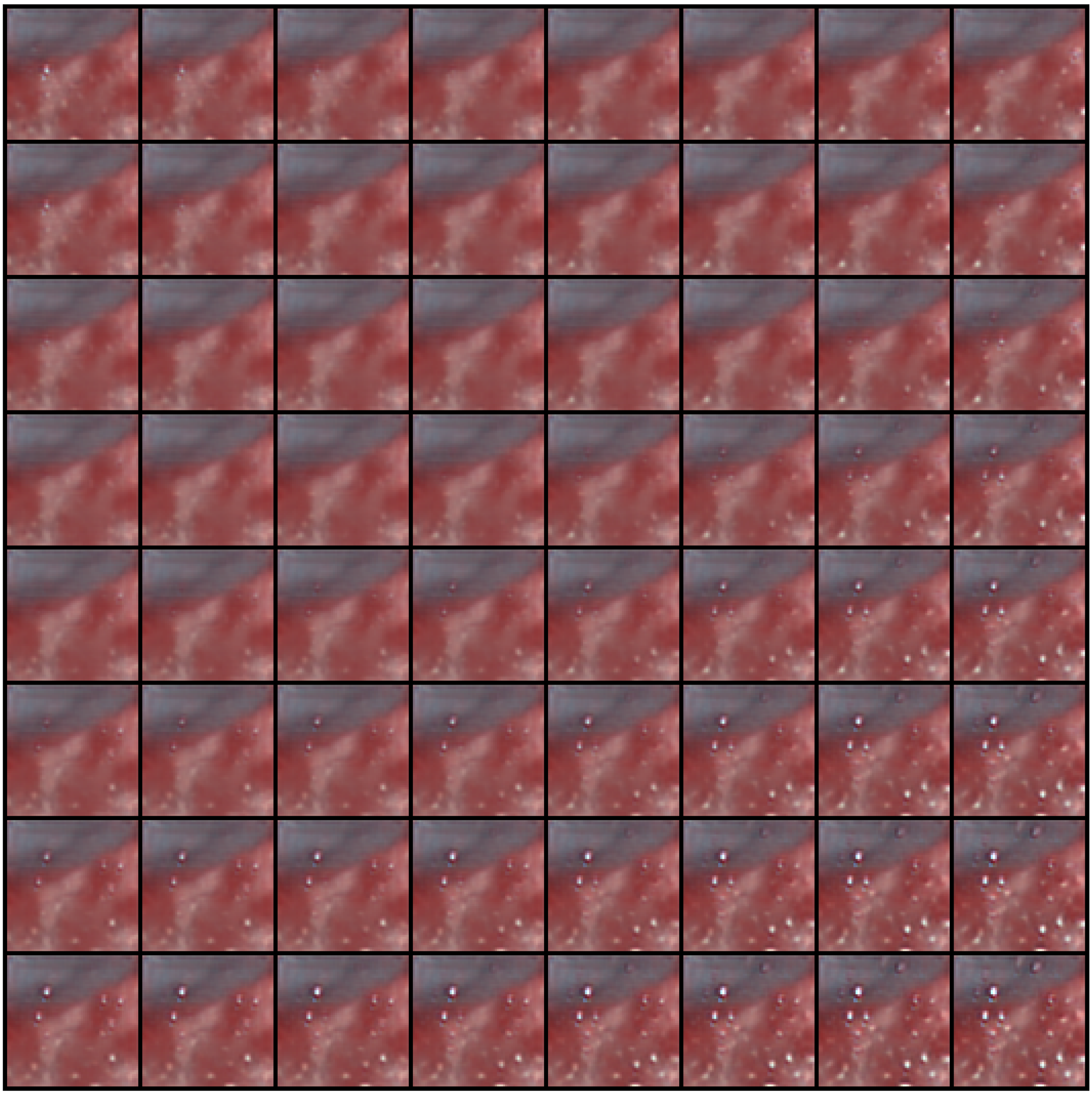


Figure 45: Multimodal interpolation in Bicycle GAN style space, between style images in top left, top right and bottom left. Style differences mostly affect additional specular highlights.

Spectral Features: Additions

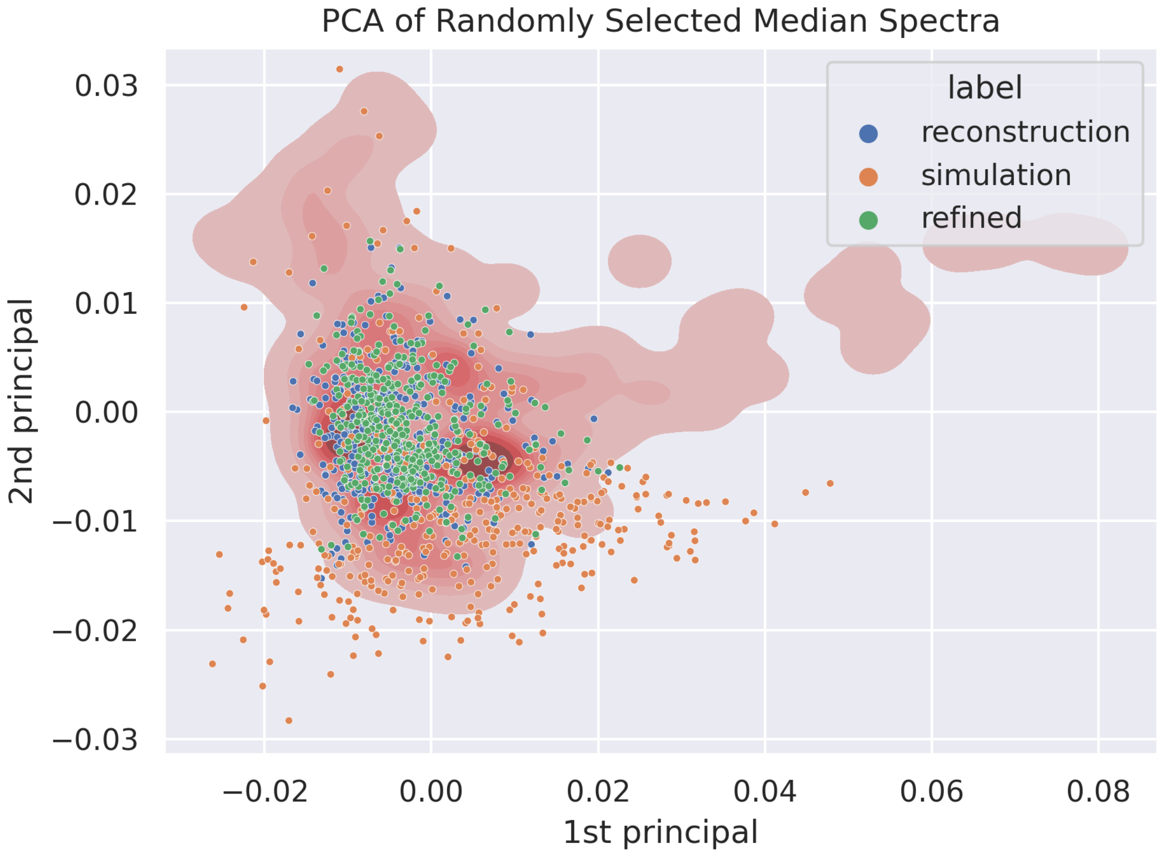


Figure 46: Kernel density plot of the first two PCA components of median spectra, stemming from 400 HSI patches from the test dataset. Explained variance from the first two components of the real HSI patch data is 87.95%. The displaying threshold for the kernel density plot with Gaussian kernel with $\sigma = 0.5$ was chosen to be 0.01. WAE reconstructions are displayed in blue and Bicycle GAN postprocessed PCA median embeddings in green. Additionally, random median spectra, selected from a Monte-Carlo database are displayed [3], showing higher median spectrum diversity but similarly low affinity to the displayed real data manifold.

Embedding Analysis: Additions

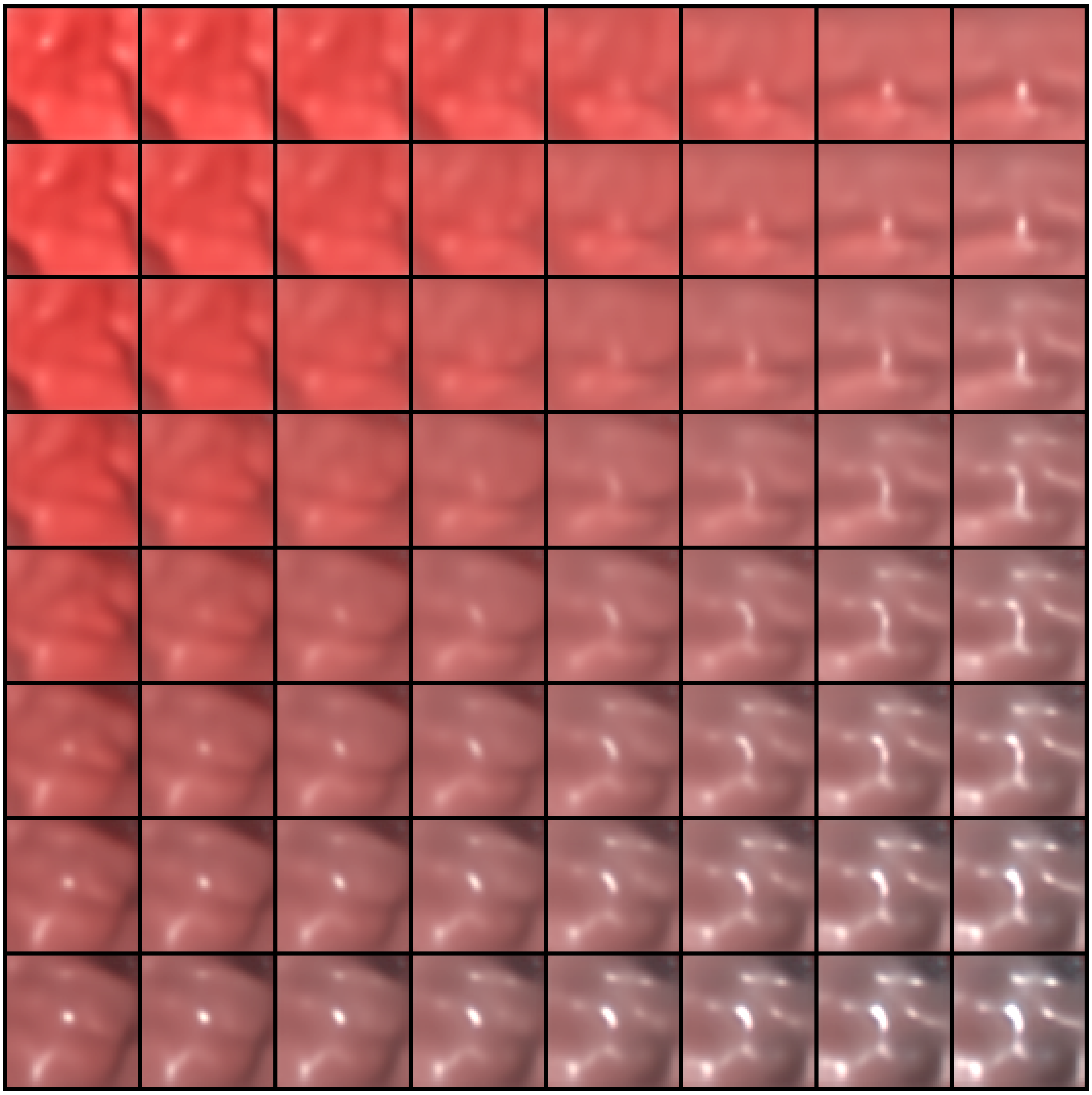


Figure 47: Linear WAE interpolation for latent dimension of 512 between image reconstruction patch in the top left, top right and bottom left. Transitions are smooth but results especially in the middle look very blurry.

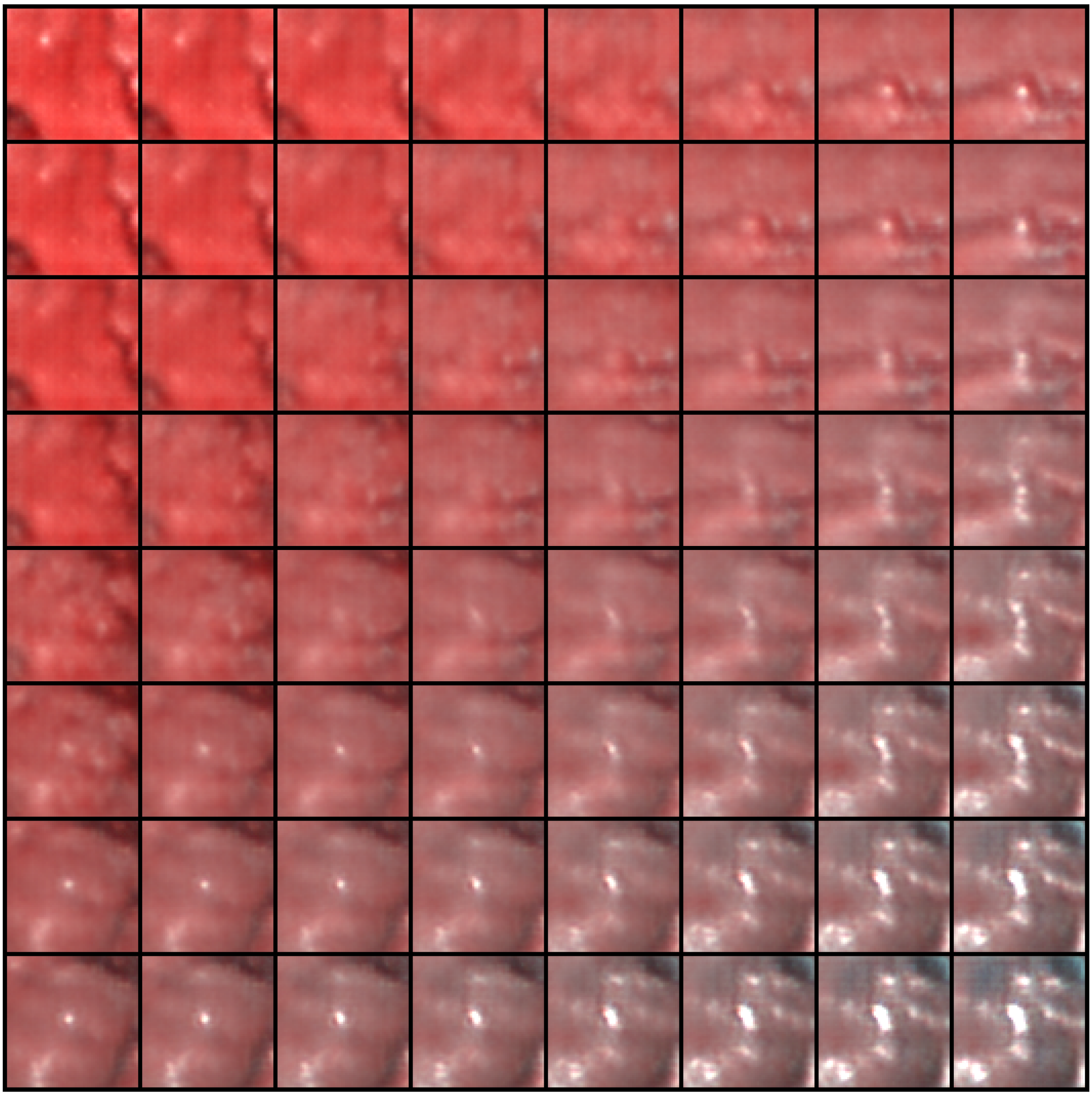


Figure 48: Linear interpolation, postprocessed with the *pix2pix* approach, between image reconstruction patch in the top left, top right and bottom left. Results in the bottom right become more unnatural and edgy patterns are observable.

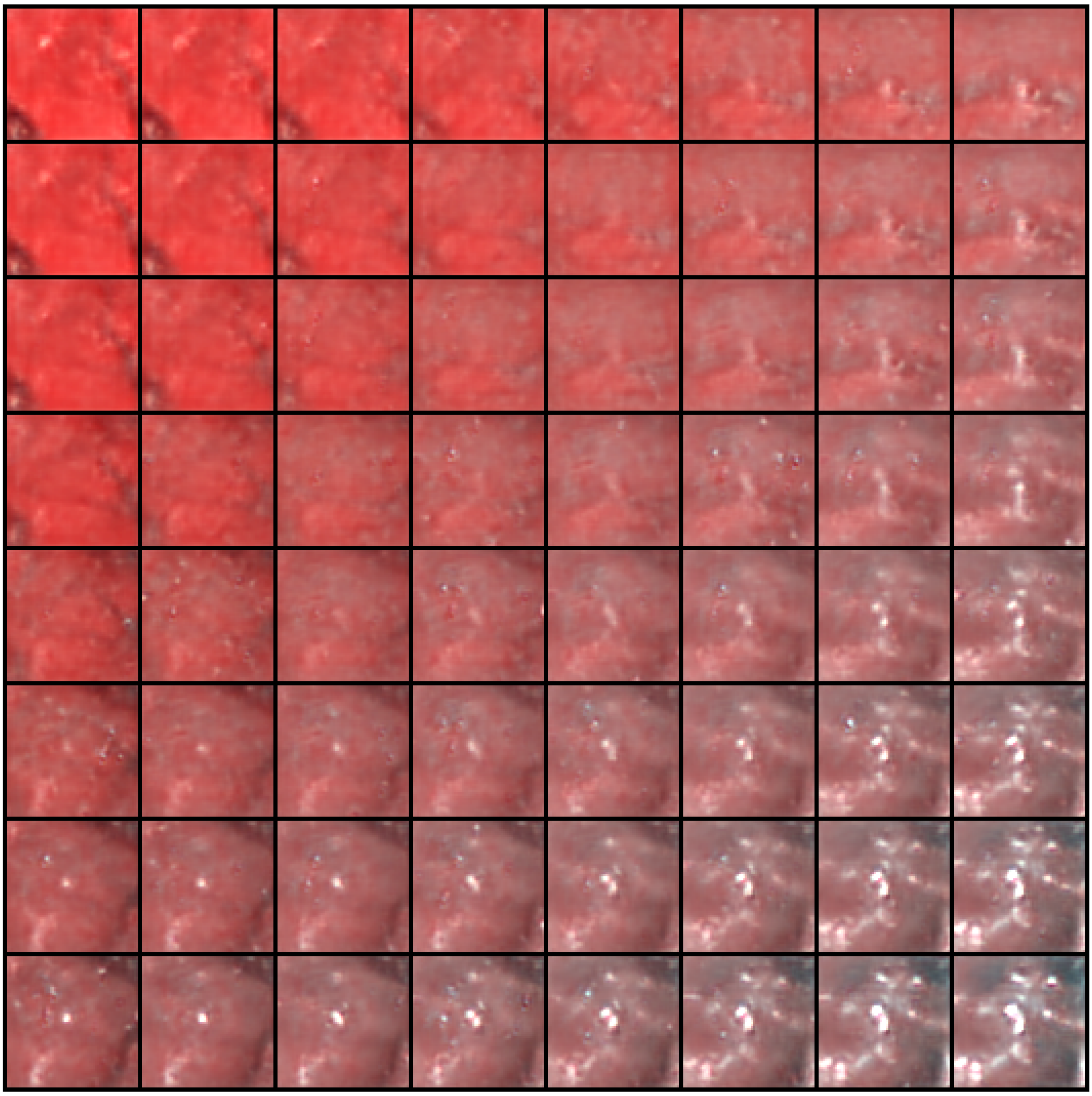


Figure 49: Linear interpolation, postprocessed with Bicycle GAN, between image reconstruction patch in the top left, top right and bottom left. Results in the bottom right become more unnatural and typical additional specular highlights are observable.

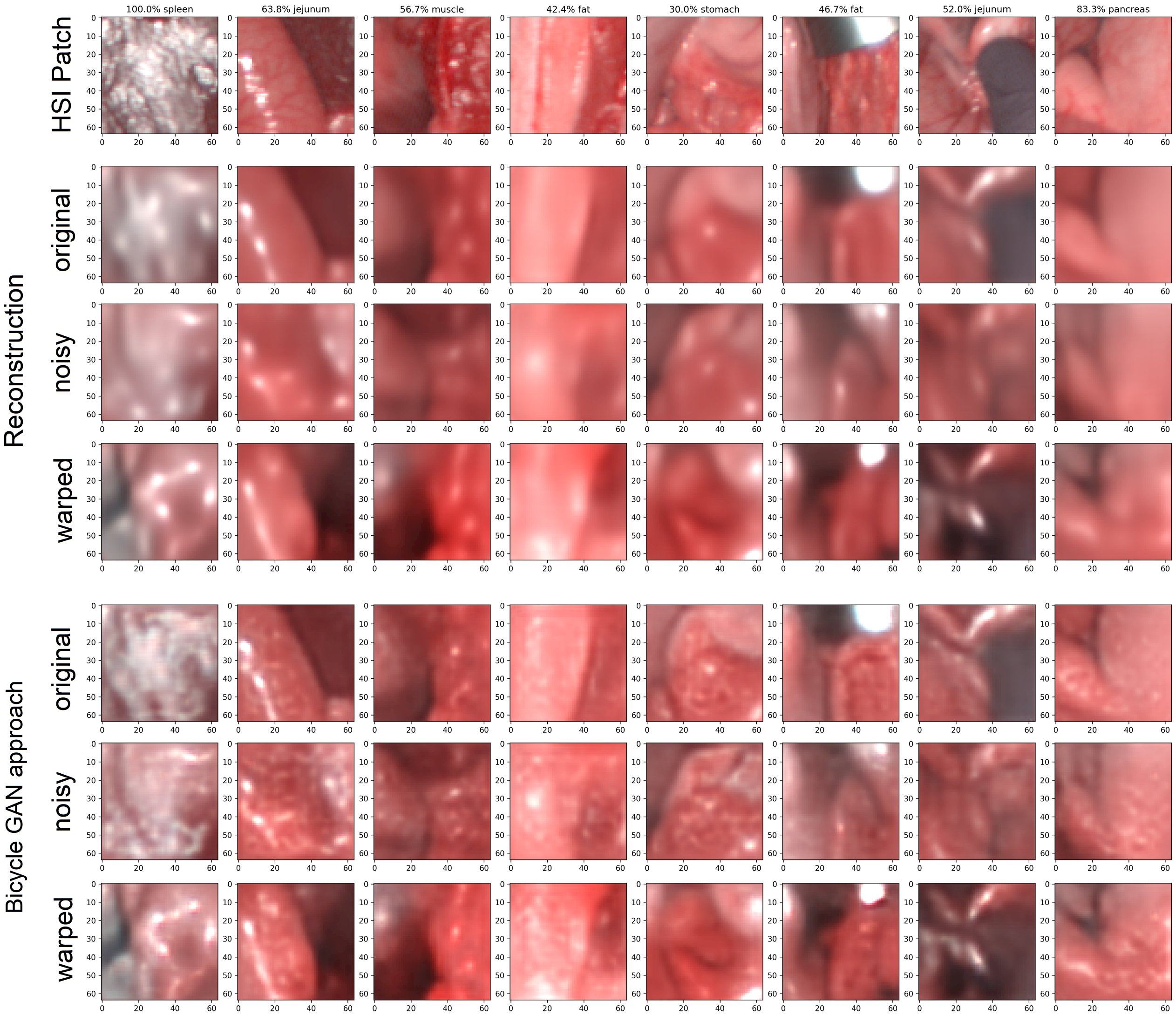


Figure 50: Results split into RGB visualization of HSI patch from the test dataset, three kinds of reconstructions and three kinds of postprocessings. Top to bottom: Real image patches, WAE reconstruction, noisy WAE reconstruction, warped noisy WAE reconstruction, reconstruction post-processed, noisy reconstruction post-processed and warped noisy reconstruction post-processed. Post-processing is done with Bicycle GAN. Except for results from the second last column, warping improves the outcome, while results for both with and without warping look physiologically plausible. Unwarped results however exhibit more oscillating colour properties. With the warped results, the environment of latent space vectors shows minor changes in shape or texture.

Texture Analysis: Additions

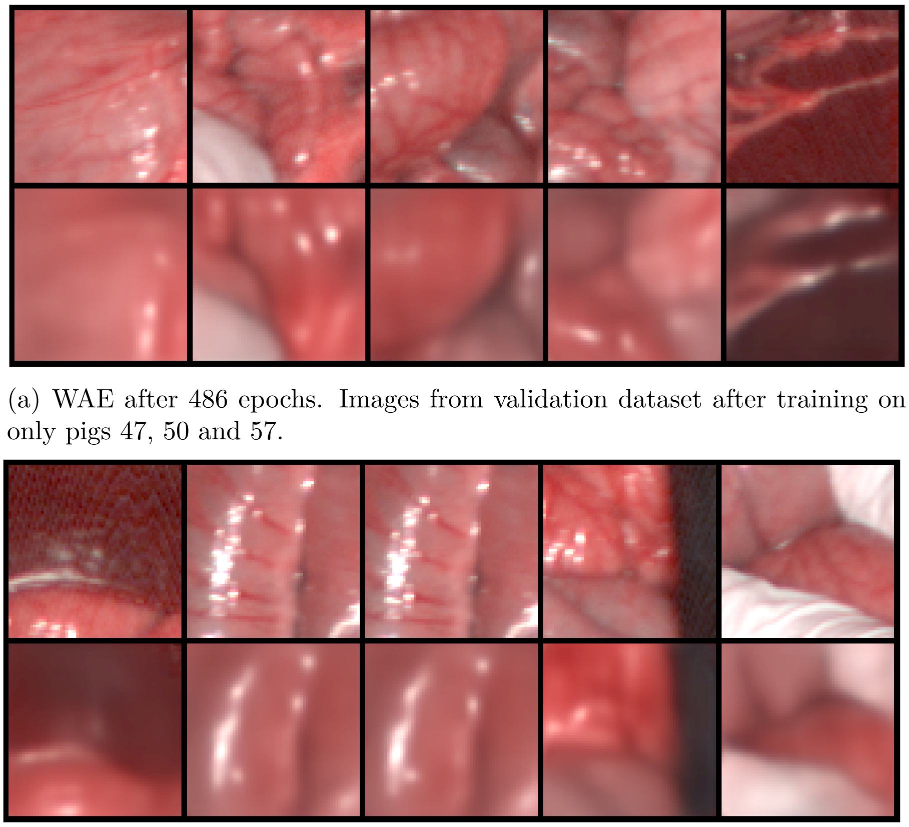
IQA metric	WAE	pix2pix	Bicycle GAN
MSE (↓)	**0.0015 ± 0.0013**	0.0021 ± 0.0018	0.0021 ± 0.0018
PSNR (↑)	**29.5 ± 3.2**	27.9 ± 3.2	27.8 ± 3.0
SSIM RGB (↑)	**0.75 ± 0.07**	0.69 ± 0.08	0.66 ± 0.08
SSIM HSI (↑)	**0.83 ± 0.06**	0.78 ± 0.07	0.76 ± 0.07
DISTS (↓)	0.28 ± 0.03	**0.24 ± 0.03**	**0.24 ± 0.03**
$\text{FID}_{10.000}$ (↓)	171 ± 8	56 ± 6	**26.3 ± 2.6**
$\text{FID}_{50.000}$ (↓)	170	54	**24.8**
$\text{KID}_{1.000}$ (↓)	0.097 ± 0.014	0.025 ± 0.009	**0.010 ± 0.004**
$\text{KID}_{10.000}$ (↓)	0.0931 ± 0.0029	0.0246 ± 0.0023	**0.0105 ± 0.0006**
MI real (↑)	1.8 ± 0.8	1.8 ± 0.8	1.8 ± 0.8
MI synth. (↑)	1.6 ± 0.6	1.7 ± 0.7	1.6 ± 0.6
IS real (↑)	8.9 ± 15	9.0 ± 13.8	9.0 ± 14.2
IS synth. (↑)	5.4 ± 4.7	7.3 ± 10.3	7.3 ± 14.2
BRISQUE real (↓)	41 ± 10	42 ± 10	41 ± 10
BRISQUE synth. (↓)	64 ± 7	**42 ± 9**	44 ± 10

Table 12: IQA metric results for reconstructed and postprocessed HSI patches. Arrows mark the direction of improving scores. Real underlying HSI patches of these results are from the training dataset. Largest difference to the results on the test dataset for FID and IS results, otherwise results similar within error margin or even equal.

IQA metric	WAE	pix2pix	Bicycle GAN
MSE (↓)	**0.0014 ± 0.0011**	0.0020 ± 0.0015	0.0020 ± 0.0015
PSNR (↑)	**29.6 ± 2.9**	27.9 ± 2.9	27.9 ± 2.7
SSIM RGB (↑)	**0.75 ± 0.07**	0.68 ± 0.08	0.66 ± 0.09
SSIM HSI (↑)	**0.83 ± 0.05**	0.78 ± 0.07	0.76 ± 0.07
DISTS (↓)	0.28 ± 0.03	0.25 ± 0.03	**0.24 ± 0.03**
$\text{FID}_{10.000}$ (↓)	109 ± 6	26.3 ± 1.8	**13.9 ± 0.5**
$\text{FID}_{50.000}$ (↓)	108	25.4	**12.9**
$\text{KID}_{1.000}$ (↓)	0.056 ± 0.004	0.0068 ± 0.0016	**0.0021 ± 0.0007**
$\text{KID}_{10.000}$ (↓)	0.0559 ± 0.0009	0.0067 ± 0.0006	**0.0022 ± 0.0002**
MI real (↑)	1.7 ± 0.7	1.7 ± 0.6	1.7 ± 0.7
MI synth. (↑)	1.5 ± 0.6	1.6 ± 0.6	1.5 ± 0.6
IS real (↑)	7.2 ± 10.3	7.1 ± 9.1	7.2 ± 10.7
IS synth. (↑)	5.5 ± 4.7	6.2 ± 8.2	6.0 ± 7.3
BRISQUE real (↓)	40 ± 10	40 ± 10	41 ± 10
BRISQUE synth. (↓)	64 ± 7	**43 ± 9**	44 ± 10

Table 13: IQA metric results for reconstructed and postprocessed HSI patches. Arrows mark the direction of improving scores. Real underlying HSI patches of these results are from the validation dataset. Largest difference to the results on the test dataset for FID and IS results, otherwise results similar within error margin or even equal.

3. Artificially Limited Data Evaluation



(a) WAE after 486 epochs. Images from validation dataset after training on only pigs 47, 50 and 57.

(b) WAE after 499 epochs. Images from validation dataset after training on only pigs 47, 50 and 57.

Figure 51: Even on limited data, the visual results and the validation loss behaviour are similar, depicting the WAE as a stable and diverse training framework even in cases of sparse data.

IQA metric	WAE	pix2pix	Bicycle GAN
Pig 43	4655 samples	5106 samples	4894 samples
MSE (↓)	**0.0012 ± 0.0013**	0.0017 ± 0.0015	0.0017 ± 0.0017
PSNR (↑)	**30.4 ± 3.2**	28.6 ± 2.9	28.9 ± 3.1
SSIM RGB (↑)	**0.76 ± 0.07**	0.71 ± 0.06	0.70± 0.06
SSIM HSI (↑)	**0.86 ± 0.04**	0.81 ± 0.05	0.81 ± 0.05
DISTS (↓)	0.27 ± 0.03	0.25 ± 0.03	**0.24 ± 0.03**
FID_{full} (↓)	133	73	**41**
$\text{KID}_{1.000}$ (↓)	0.073 ± 0.005	0.029 ± 0.004	**0.0159 ± 0.0028**
Pig 46	8365 samples	7445 samples	6488 samples
MSE (↓)	**0.0012 ± 0.0013**	0.0018 ± 0.0017	0.0021 ± 0.0022
PSNR (↑)	**30.4 ± 3.5**	28.6 ± 3.2	28.4 ± 3.6
SSIM RGB (↑)	**0.78 ± 0.07**	0.71 ± 0.08	0.70± 0.09
SSIM HSI (↑)	**0.84 ± 0.04**	0.79 ± 0.07	0.79 ± 0.08
DISTS (↓)	0.26 ± 0.03	0.25 ± 0.03	**0.24 ± 0.03**
FID_{full} (↓)	110	32	**22**
$\text{KID}_{1.000}$ (↓)	0.052 ± 0.006	0.0104 ± 0.0032	**0.0050 ± 0.0012**
Pig 62	18211 samples	19133 samples	19655 samples
MSE (↓)	**0.0013 ± 0.0012**	0.0019 ± 0.0015	0.0020 ± 0.0015
PSNR (↑)	**29.9 ± 3.0**	28.1 ± 2.8	28.0 ± 2.8
SSIM RGB (↑)	**0.76 ± 0.07**	0.70 ± 0.07	0.68± 0.07
SSIM HSI (↑)	**0.84 ± 0.05**	0.79 ± 0.06	0.78 ± 0.06
DISTS (↓)	0.27 ± 0.03	**0.24 ± 0.03**	**0.24 ± 0.03**
FID_{full} (↓)	54	**16**	17
$\text{KID}_{1.000}$ (↓)	0.0252 ± 0.0024	**0.0031 ± 0.0012**	0.0060 ± 0.0013
Pig 68	5036 samples	5465 samples	5619 samples
MSE (↓)	**0.0014 ± 0.0014**	0.0020 ± 0.0018	0.0021 ± 0.0020
PSNR (↑)	**29.8 ± 3.0**	28.2 ± 2.9	27.9 ± 3.0
SSIM RGB (↑)	**0.77 ± 0.06**	0.70 ± 0.07	0.69± 0.08
SSIM HSI (↑)	**0.84 ± 0.05**	0.79 ± 0.06	0.79 ± 0.08
DISTS (↓)	0.27 ± 0.03	**0.24 ± 0.03**	**0.24 ± 0.03**
FID_{full} (↓)	82	29	**23**
$\text{KID}_{1.000}$ (↓)	0.0388 ± 0.0021	0.0076 ± 0.0012	**0.0066 ± 0.0006**
Pig 72	13733 samples	12851 samples	13344 samples
MSE (↓)	**0.0018 ± 0.0019**	0.0026 ± 0.0025	0.0032 ± 0.0036
PSNR (↑)	**28.6 ± 3.2**	27.0 ± 3.2	26.6 ± 3.5
SSIM RGB (↑)	**0.76 ± 0.07**	0.68 ± 0.07	0.67± 0.08
SSIM HSI (↑)	**0.83 ± 0.06**	0.78 ± 0.07	0.77 ± 0.08
DISTS (↓)	0.27 ± 0.03	**0.24 ± 0.03**	**0.24 ± 0.03**
FID_{full} (↓)	74	29	**19**
$\text{KID}_{1.000}$ (↓)	0.032 ± 0.003	0.0066 ± 0.0017	**0.0046 ± 0.0007**

Table 14: IQA metric results for reconstructed and postprocessed HSI patches. Arrows mark the direction of improving scores. Real underlying HSI patches of these results are from the test dataset. MI and IS are due to their unreliability not reported. FID metrics are computed on the full sample amounts of each pig. The same ordering of best values as for non-limited data can be observed. The results vary among the different pigs, as suggested by Figure 33.

Erklärung

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 09.11.2021 ....