Artificial Intelligence 338 (2025) 104240

Contents lists available at ScienceDirect

Artificial Intelligence

journal homepage: www.elsevier.com/locate/artint

Chimeric U-Net – Modifying the standard U-Net towards explainability ☆

Kenrick Schulze [a], Felix Peppert [a], Christof Schütte [b], Vikram Sunkara [a,*]

[a] *Explainable AI for Biology, Zuse Institute Berlin, Takustraße 7, 14195, Berlin, Germany*
[b] *Department Mathematics of Complex Systems, Zuse Institute Berlin, Takustraße 7, 14195, Berlin, Germany*

ARTICLE INFO

Keywords:
U-Net
Invertible decoder
Semantic segmentation
Class embeddings

ABSTRACT

Healthcare guided by semantic segmentation has the potential to improve our quality of life through early and accurate disease detection. Convolutional Neural Networks, especially the U-Net-based architectures, are currently the state-of-the-art learning-based segmentation methods and have given unprecedented performances. However, their decision-making processes are still an active field of research. In order to reliably utilize such methods in healthcare, explainability of how the segmentation was performed is mandated. To date, explainability is studied and applied heavily in classification tasks. In this work, we propose the Chimeric U-Net, a U-Net architecture with an invertible decoder unit, that inherently brings explainability into semantic segmentation tasks. We find that having the restriction of an invertible decoder does not hinder the performance of the segmentation task. However, the invertible decoder helps to disentangle the class information in the latent space embedding and to construct meaningful saliency maps. Furthermore, we found that with a simple k-Nearest-Neighbours classifier, we could predict the Intersection over Union scores of unseen data, demonstrating that the latent space, constructed by the Chimeric U-Net, encodes an interpretable representation of the segmentation quality. Explainability is an emerging field, and in this work, we propose an alternative approach, that is, rather than building tools for explaining a generic architecture, we propose constraints on the architecture which induce explainability. With this approach, we could peer into the architecture to reveal its class correlations and local contextual dependencies, taking an insightful step towards trustworthy and reliable AI. Code to build and utilize the Chimeric U-Net is made available under: https://github.com/kenrickschulze/Chimeric-UNet--Half-invertible-UNet-in-Pytorch

1. Introduction

Deep learning (DL) has become ubiquitous in medical diagnosis. In particular, the sub-field of *semantic segmentation* has the potential to make significant contributions to human well-being through image guided diagnosis of severe diseases [1]. In semantic segmentation tasks, each pixel of an image is tasked to be classified as one of the prescribed set of classes, for example, detecting and localizing cancer in MRI scans of patients [2], [3]. Convolutional Neural Networks (CNNs) are the main work-horse in semantic

☆ This project was funded by the Berlin Institute for the Foundations of Learning and Data (BIFOLD) (Project BIFOLD-BZML 01IS18037H) and the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the Germany's Excellence Strategy – The Berlin Mathematics Research Center MATH+ (EXC-2046/1 project ID: 390685689).

* Corresponding author.
E-mail address: sunkara@zib.de (V. Sunkara).

https://doi.org/10.1016/j.artint.2024.104240
Received 8 November 2022; Received in revised form 6 October 2024; Accepted 18 October 2024

Available online 30 October 2024

segmentation problems, in which features are learnt from data, embracing the ability to find novel features beyond our traditional hand-crafted features [4]. This learning based feature finding leads to CNNs being "black-boxes", as their decision making process lacks transparency and interpretability in many applications [5]. However, in order to prescribe DL aided healthcare to people, trust in predictions and how they were made is imperative [6].

In 2015, Ronneberger et al. introduced a variant on the classical encoder-decoder CNN architecture that included *skip connections* between respective blocks of the encoder-decoder, which they named the *U-Net* [7]. The U-Net was originally designed for the segmentation tasks of biomedical datasets, but also found application in areas like autonomous driving [8], [9]. Many modifications of the U-Net have been proposed, each driven to exploit structures in particular datasets, especially in the field of biomedical imaging [10], [11]. However, these architectures are still black-boxes and post-hoc analyses need to be performed to explain their decision making processes. Typically, these explanations are derived by investigating locally the architecture's gradients. *Explainability* is still in its infancy for the U-Net and is an active field of research. To date, there are only a few options for Explainable AI (XAI) in semantic segmentation tasks [12], [13], [14].

XAI emerged out of the need to explain how and why a classification task reaches its prediction. In mathematical nomenclature, XAI is the study and approximation of the derivative of the DNN with respect to the prediction. The broadest grouping of XAI approaches is differentiated into *global* and *local* explanations. Global methods focus on the average or expected behaviour of the model concerning the provided dataset or by finding prototypical inputs. This is used to detect biases [13], highlight feature importance [15], extract knowledge [5], or capture inter-class relationships [13], [16]. In clinical practice where the model's prediction may alter patient treatment, the decision making process has to be revealed for each prediction. Hence, local explanations are provided by highlighting areas within the input the models decision was based on [5].

Most XAI approaches for CNNs can be divided into two subgroups, namely perturbation-based [17], [18] and backpropagation/gradient-based approaches [15], [19], [20], [21]. In practice, the gradients are computed with respect to the activations of the latent space. We can state the core XAI question in classification tasks as follows:

(XAI α) – *Which region in the source contributed to that particular prediction?*

In gradient-based approaches, this question is answered by studying *saliency maps*, that is the derivative of the prediction w.r.t. the source, which highlights regions in the source, that induce strong derivatives in the prediction. In practice, those maps are, for example, superimposed directly onto the input [19] or collapsed with the target activations beforehand [15].

However, in semantic segmentation, the (XAI α) question needs to also incorporate spatial information into the prediction. To do this, Vinogradova et al. proposed to sum all pixel-wise class scores of interest in the prediction and evaluate the derivative of this scalar with respect to the source, which is currently the most cited XAI approach for semantic segmentation tasks [12].

Inspired by gradient-based XAI analysis, we propose a modification/constraint to incorporate the XAI motivated gradients into architectures. In particular, we present the Chimeric U-Net, which is the standard U-Net with an invertible decoder unit, i.e. a continuous invertible decoder map. This invertibility restriction on the decoder implies that the architecture:

I: has a continuous-differentiable mapping between the target space and the latent space,
II: has a simple mathematical form for the derivatives of the latent space w.r.t. the target space and *vice versa*,
III: has the ability to remove redundant information inside the non-invertible encoder, giving a concise latent representation.

Furthermore, by only considering invertibility to the latent space and not all the way back to the input space, we focus only on the features learned by the architecture, rather than the full feature space of the input.

Exploiting the invertibility of the decoder of the Chimeric U-Nets, in this work, we study a novel XAI question:

(XAI β) – *What is the sensitivity of the latent space w.r.t. the changes in the target prediction?*

Without loss of generality, we can think of the question (XAI α) as a tool for studying the sensitivity of the architecture, and (XAI β) as a tool to study its specificity, we will derive and explore this in the later sections of this paper. In this work, we study explainability by studying the learned embedding of the XAI questions given above. With these embeddings, we aim for a *global* understanding through the internal representation of the dataset in the architectures, and also the *local* understanding through saliency maps of individual data points.

In this work, we first demonstrate how making the decoder invertible does not hinder the network in performing semantic segmentation tasks. Then, we show how using the invertibility of the decoder enables XAI analysis, such as disentangled class information in the latent space embedding, statistical predictions of unseen data, and lastly, constructing saliency maps. The paper is structured as follows: In Section 2 we begin by introducing the Chimeric U-Net and presenting the mathematical framework for the architecture and of the subsequent XAI analysis. In Section 3, we present the numerical experimental setup that was used to understand the properties of the Chimeric U-Net. In Section 4, we present the results of how the Chimeric U-Net compares to other methods on two medical segmentation datasets. We perform XAI analysis on the medical datasets for new insights through the XAI analysis mentioned above. Lastly, in Section 5, we discuss the limitations, impact, and future research questions in regards to the Chimeric U-Net approach.

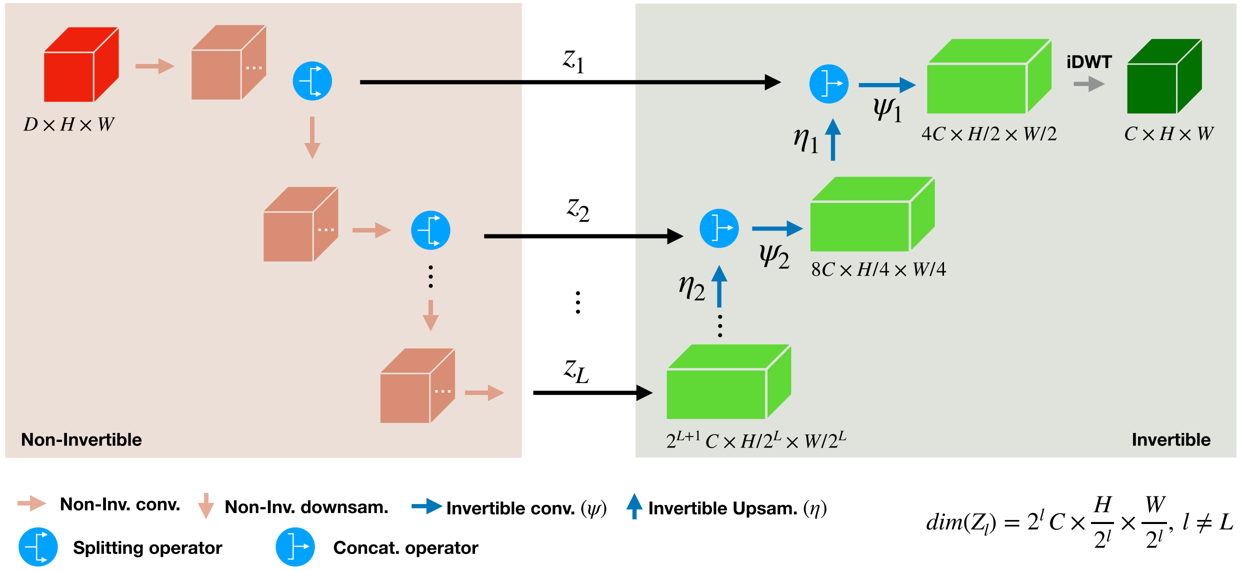
$D \times H \times W$
z_1
z_2
z_L
η_1
ψ_1
$4C \times H/2 \times W/2$
iDWT
$C \times H \times W$
η_2
ψ_2
$8C \times H/4 \times W/4$
$2^{L+1}\, C \times H/2^L \times W/2^L$
Non-Invertible
Invertible
Non-Inv. conv.
Non-Inv. downsam.
Invertible conv. (ψ)
Invertible Upsam. (η)
Splitting operator
Concat. operator
$dim(Z_l) = 2^l\, C \times \frac{H}{2^l} \times \frac{W}{2^l},\ l \neq L$

Fig. 1. The Chimeric U-Net Schematic: non-invertible encoder connected over skip connection to an invertible decoder.

2. Background

2.1. Notation and naming conventions

The following are core notations of this paper: The integer product $H \times W$ denotes the height and width of the images. The integer C denotes the number of class/target channels. The integer D denotes the number of channels in the input signal/image. We use $L \in \mathbb{N}$ for the number of layers, we index each layer by $l \in \{1, \dots, L\}$. The operators of the architecture act on tensors, for which we use for brevity bold notations, e.g. $\mathbf{x}$, for general tensors whose dimensions should be determined in context of its appearance. Furthermore, for a tensor $\mathbf{x}$, we denote the *splitting,* which inherently only acts on the channel dimension of the tensor, as

$$\mathbf{x} := \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \text{ and } \textit{vice versa } x_1 := [\mathbf{x}]_1 \text{ and } x_2 := [\mathbf{x}]_2,$$

for when we *concatenate* two tensors into a single tensor and when we want to extract the first or second split of the tensor, respectively. We propose this tensor like notation, which is notably unconventional, so that later in the article we can describe the entire decoder in a single equation. In regards to naming conventions, we interchangeably use “target” and “segmentation” for easy reading.

2.2. The architecture

We first describe the steps of the Chimeric U-Net (see Fig. 1) in plain language and proceed to reiterate the steps formally using mathematical nomenclature. To begin, the Chimeric U-Net consists of two sub-units connected by skip connections: the *non-invertible encoder* and the *invertible decoder*. The invertibility of the decoder fixes the dimensionality of the skip connections, which is determined by the number of target classes and the dimensions of the image. The encoder is non-invertible and is free to be modelled to the data of interest, with a constraint to contribute the prescribed dimensionality of the skip connections. Given the encoder is specific to the data, we state the specifics of the encoder in the numerical experiments, and from here on only focus on describing the invertible decoder.

Observing the invertible decoder in Fig. 1 (highlighted with green background), we see that the input to a layer in the decoder is the concatenation of the corresponding encoder layer output and the upsampled output of the layer below. The dimensions of these two tensors are the same, therefore, we concatenate them in the channel dimension and then pass it through the operations of that layer. In particular, in these middle layers, the input tensor goes through a sequence of simple *additive coupling layers* from *Normalising Flows* [22]. Those additive coupling layers are in essence transforming the features in a continuous and invertible fashion. After this, the output tensor is sent to the layer above via the *invertible learnable upsampler* [23], which rearranges the pixels of the layer from, for example, $C \times H \times W$ to $C/4 \times 2H \times 2C$, in order to alter the dimensionality of their input. Invertible downsampling filters are the adjoint operator of the exponential of the orthogonal skew-symmetric matrices. In that, rows when reordered into filters and convolving with an appropriate stride act as orthogonal convolutions. Since the operator is orthogonal and the involved dimensions are finite, then its inverse, i.e. the invertible upsampling filter, is just the adjoint operator. Hence, all operations in the decoder of the Chimeric U-Net are inherently invertible. This decoder layer design is that of the *Invertible U-Net* which was introduced by Etmann et al. [23].

The first and the last layers of the decoder are exceptions, that is, after the additive coupling layers of the first layer ($l = 1$) an *inverse discrete wavelet transform* ($\mathtt{iDWT}$) is applied to the output tensor, reducing the dimensionality down to the target dimensions. Lastly, the last layer ($l = L$) does not receive any input from a layer below, therefore, simply passes the skip connection directly through to the layer above via the upsampler.

In summary, the decoder is rearranging-, concatenating-, and continuously deforming the points from the skip connections to the segmentation. This gives us a unique correspondence between the Cartesian product of the skip connections and the segmentation. We now present the steps of the decoder unit with mathematical nomenclature.

We begin by introducing the notation for the sets of interest. We denote the input image (*source*) set by $\mathcal{X} \subset [0,1]^{D\times H\times W}$, containing a stack of D image channels of dimension $H\times W$. We define our target (*segmentation*) set by $\mathcal{Y} \subset \{\mathbf{y} \in [0,1]^{C\times H\times W} : \sum_c \mathbf{y}[c,\cdot,\cdot] = 1\}$, stack of C classes each of shape $H \times W$, where the sum over the classes is equal to 1 for each coordinate in $H \times W$. In addition to the input- and target sets, we have the set of skip connections, which we denote by,

$$\mathcal{Z} \subset \left(\bigoplus_{l=1}^{L-1} \mathbb{R}^{2^l C\times \frac{H}{2^l}\times\frac{W}{2^l}}\right) \bigoplus \mathbb{R}^{2^{L+1}C\times\frac{H}{2^L}\times\frac{W}{2^L}}. \tag{2.1}$$

For $\mathbf{z} \in \mathcal{Z}$ and $l \in \{1,\dots,L\}$, we denote to the l^{th} skip connection by z_l, i.e. $\mathbf{z} := (z_1,\dots,z_L)$.

Using these three sets, $(\mathcal{X},\mathcal{Z},\mathcal{Y})$, the Chimeric U-Net is defined as the mappings:

$$\mathcal{X} \xrightarrow{\Phi} \mathcal{Z} \xrightarrow{\Psi} \mathcal{Y}, \tag{2.2}$$

where Φ is the non-invertible encoder function and Ψ is the invertible decoder function.

We recall, the encoder function, $\Phi : \mathcal{X} \to \mathcal{Z}$, is free to be chosen to exploit the specific features of the dataset to be studied. However, the map $\Psi : \mathcal{Z} \to \mathcal{Y}$ is a fixed nested function, with the evaluation of an element $\mathbf{z} \in \mathcal{Z}$ given by,

$$\Psi(\mathbf{z}) := \mathtt{iDWT}\left(\begin{bmatrix} z_1 \\ \eta_1 \circ \psi_1\left(\begin{bmatrix} z_2 \\ \eta_2 \circ \psi_2\,([\dots]) \end{bmatrix}\right) \end{bmatrix}\right), \tag{2.3}$$

where the last term in the nesting is z_L.

The invertible decoder function Ψ, (2.3), is constituted of three key functions: the inverse discrete wavelet transform, the learnable invertible upsampling operators ($\eta_\bullet$), and lastly, the normalising flow units ($\psi_\bullet$). In particular, for a layer l of the Chimeric U-Net, the learnable invertible upsampling operator is given by the map,

$$\eta_l : \mathbb{R}^{2^{l+2}C\times\frac{H}{2^{l+1}}\times\frac{W}{2^{l+1}}} \longrightarrow \mathbb{R}^{2^l C\times\frac{H}{2^l}\times\frac{W}{2^l}}, \tag{2.4}$$

which preserves the total number of elements in the mapping from the domain to the range [23]. The normalising flow of layer l is given by,

$$\psi_l(\mathbf{x}, m) := f_l^1 \circ f_l^2 \circ \cdots \circ f_l^{m-1} \circ f_l^m(\mathbf{x}), \tag{2.5}$$

a composition of $m \in \mathbb{N} \geq 2$ additive coupling layers of $f_l^\bullet$, given by,

$$f_l^\bullet\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) := \begin{bmatrix} x_2 \\ x_1 + F_{(l,\bullet)} \circledast x_2 \end{bmatrix}. \tag{2.6}$$

For brevity, we omit specifying m in the general representation of ψ in (2.3).

With the definitions above, we can define the inverse of Ψ, mapping points from the segmentation back into the skip connections, as follows: for $\mathbf{y} \in \mathcal{Y}$,

$$\begin{aligned}\Psi^{-1}(\mathbf{y}) := \big(&[\psi_1^{-1}(\mathtt{DWT}(\mathbf{y}))]_1, \\ &[\psi_2^{-1}\circ\eta_1^{-1}([\psi_1^{-1}(\mathtt{DWT}(\mathbf{y}))])]_2)]_1, \\ &[\psi_3^{-1}\circ\eta_2^{-1}([\psi_2^{-1}\circ\eta_1^{-1}([\psi_1^{-1}(\mathtt{DWT}(\mathbf{y}))]]_2)]_2)]_1, \\ &\vdots \\ &\eta_{L-1}^{-1}(\dots)\big),\end{aligned} \tag{2.7}$$

where the inverse of the normalising flows is given by,

$$\psi_l^{-1}(\mathbf{x}, m) := (f_l^m)^{-1} \circ \cdots \circ (f_l^1)^{-1}(\mathbf{x}), \tag{2.8}$$

with

$$(f_l^\bullet)^{-1}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) := \begin{bmatrix} x_2 + F_{(l,\bullet)} \circledast x_1 \\ x_1 \end{bmatrix}. \tag{2.9}$$

Lastly, the inverse of the learnable invertible upsampling function becomes an invertible downsampling function,

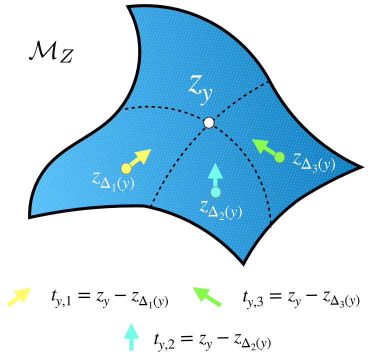
$\mathcal{M}_Z$
z_y
$z_{\Delta_1(y)}$
$z_{\Delta_2(y)}$
$z_{\Delta_3(y)}$
$t_{y,1} = z_y - z_{\Delta_1(y)}$
$t_{y,3} = z_y - z_{\Delta_3(y)}$
$t_{y,2} = z_y - z_{\Delta_2(y)}$

Fig. 2. Cartoon. A sub-manifold $\mathcal{M}_Z$ around the point z_y. Three other points, coloured yellow, green and cyan, are pull-backs of y after perturbation with the function Δ_c to the class $c \in \{1, 2, 3\}$. The arrows are pull-back gradients for y and class c pointing to z_y. Approximate direction to travel to reach z_y from the perturbed state $z_{\Delta_c y}$. (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

$$\eta_l^{-1} : \mathbb{R}^{2^l C \times \frac{H}{2^l} \times \frac{W}{2^l}} \longrightarrow \mathbb{R}^{2^{l+2} C \times \frac{H}{2^{l+1}} \times \frac{W}{2^{l+1}}}. \tag{2.10}$$

Remark 2.1. In practice, we apply a `softmax` operation after the `iDWT` to satisfy the probability constraint of the elements in the target set. This makes Ψ non-invertible on the whole range $\mathcal{Y}$, however, we are strictly interested in the subset of the target set, $\Psi \circ \Phi(\mathcal{X}) \subset \mathcal{Y}$, on which Ψ is invertible, which here on in we denote by $\mathcal{Y}^{\dagger}$. We treat this set as the range of the architecture, and its latent space is the focus of this work.

The invertibility of the decoder is a tool for bringing explainability to the U-Net. With this map, we can explain the sensitivity and the computational reasoning behind the predicted segmentation of the architecture. In the following section, we derive how the invertible decoder can be used for explainability.

2.3. Latent space

The Chimeric U-Net constructs a latent space in its skip connections, we interpret $\mathcal{Z}$ as a multi-resolution latent space of the elements of $\mathcal{X}$. In general, studying latent spaces in imaging based tasks is an active field of research; the main focus is put on constructing meaningful latent representations of the input data for classification- and generative tasks [24]. However, in the particular case of semantic segmentation tasks, the inability to untangle the multiple segmentation classes of the target in the latent space pose a challenge. That is, given the target $\mathbf{y}$ is a multi-class tensor, then each point $\mathbf{z} = \Psi^{-1}(\mathbf{y})$ has entangled information of all the classes and pixels of $\mathbf{y}$; making it hard to naturally partition $\mathcal{Z}$ w.r.t. $\mathcal{Y}$.

An approach to disentangle the classes has been the use of *saliency maps* [19]. In brief, saliency maps are heatmaps that highlight pixels, which contributed most to the prediction score of the target class. So far, they have been predominantly applied in image classification tasks rather than image segmentation. Probably, the most well-known of such gradient-based saliency maps is *Grad-CAM* [15] with all its recent modifications [25], [20], [26]. *Grad-CAM* provides visual explanations for CNN predictions of a convolutional layer (usually the last is chosen) within the network by assigning a relevance score to each pixel. The relevance score is computed by collapsing the activations maps of the target convolutional layer, weighted by the respective gradients. In *Grad-CAM*, each channel is weighted by the same average gradient score, whereas in a more recent modification, namely HiResCAM, the *Hadamard product* of the feature maps with the respective gradients is taken before collapsing them [25]. Saliency maps are an active field of research to explain Neural Network decisions. However, their evaluations, which are generally based on human interpretation [25], [27], [28], [5], are a topic of ongoing scientific debate [29], [30], [31], [32], [5].

We borrow concepts from *Grad-CAM* to help create a class-deconvolved latent space over the skip connection of the Chimeric U-Net. The invertibility and the continuity of the decoder map facilitates the computation of the *pull-back* of points of the target space onto the latent space, but more importantly, we also gain a natural pull-back of the tangents in $\mathcal{Y}^{\dagger}$ to tangents in $\mathcal{Z}$. That is, given an element $\mathbf{y} \in \mathcal{Y}^{\dagger}$ and a class $c \in \{1, \ldots, C\}$, we define an average gradient of the pixels of y being predicted to class c by

$$\mathbf{t}_{\mathbf{y},c} := \sum_i \frac{\partial \Psi^{-1}(\mathbf{y})}{\partial \mathbf{y}_{i,c}} w_{\mathbf{y},i,c}, \tag{2.11}$$

where $i \in \{1, \ldots, H \times W\}$ is an indexing of pixels and $w_{\mathbf{y},i}$ an importance weighting of the i^{th} pixel in $\mathbf{y}$ to class c. If we denote $z_y \in \mathcal{Z}$ as the corresponding element satisfying $\Psi(z_{\mathbf{y}}) = \mathbf{y}$, and $\mathcal{Y}^{\dagger}$ is a manifold, then for each pixel i and class c, the partial derivative $\partial \Psi^{-1}(\mathbf{y})/\partial \mathbf{y}_{i,c} \in \mathbb{R}^{dim(\mathcal{Z})}$ describes the sensitivity of $z_{\mathbf{y}}$ with respect to the i^{th} pixel predicted as class c. Then, we take the weighted average over all pixels in the class of interest to construct an average sensitivity of $z_{\mathbf{y}}$ with respect to class c, (2.11).

Using the *pull-back function*, Ψ^{-1}, we define a new latent space with untangled channels as follows:

$$\mathcal{T}_{\mathcal{X}} := \{\mathbf{t}_{\mathbf{y},c} \,|\, \mathbf{y} \in \Psi \circ \Phi(\mathcal{X}) \text{ and } c \in \{1, \dots, C\}\}. \tag{2.12}$$

Let us consider a simple example to gain intuition into what the elements of the set above can represent. As from before, let $z_{\mathbf{y}}$ be the pull-back of a point $\mathbf{y}$ (see Fig. 2). Let $\Delta_c(\mathbf{y})$ be a small perturbation of $\mathbf{y}$, for example, a decrease in the prediction of class c. We pull-back $\Delta_c(\mathbf{y})$ to $z_{\Delta_c(\mathbf{y})}$ using the pull-back Ψ^{-1}. Then, the difference vector, $z_{\mathbf{y}} - z_{\Delta_c(\mathbf{y})}$, is a coarse first order approximation of $\mathbf{t}_{\mathbf{y},c}$, that is, the vector pointing from $z_{\Delta_c(\mathbf{y})}$ to $z_{\mathbf{y}}$. Such a directional vector can be constructed for each class c, (coloured arrows in Fig. 2). It is not intuitive why a set of such tangent-like vectors, $\mathbf{t}_{\bullet}$, should give a good latent space. Nevertheless, we will show through numerical experimental that $\mathcal{T}_{\mathcal{X}}$ is a meaningful latent space, as it can be used to build saliency, predict functionals, like Intersection over Union (IoU), and more surprisingly, qualify unseen test data.

3. Methods

We performed numerical experiments to understand the Chimeric U-Net, which consists of a non-invertible encoder and an invertible decoder, in comparison to the standard (fully non-invertible) U-Net and the fully invertible U-Net. Furthermore, we study the latent space induced in the Chimeric U-Net and investigate its significance and impact. The following are the sub-modules used to construct the numerical experiments presented in the results.

3.1. Architecture models

We constructed three architectures: the standard U-Net (sU-Net), the invertible U-Net (iU-Net), and the Chimeric U-Net. We modified the architectures of the standard- and the invertible U-Net, such that, the encoder of the sU-Net and the decoder of the iU-Net matched with the encoder and decoder of the Chimeric U-Net, respectively. We present the detailed model representations in Appendix S-VI Fig. S-1.

The architecture of the sU-Net was based on the work by Ronneberger et al. [7]. We modified their architecture in two ways: firstly, we expanded the input dimensionality to match the input channels with the output channels, which was done with a standard 2D convolution; secondly, we passed half the channels across the skip connection and passed the other half down to the successive layer, in contrast to [7], in which all channels were passed to both the skip connections and the layer below. Other than these changes, we used standard 2D convolutions with stride of two, kernel size of 3, and padding of one, to retain the spatial dimensions. Each convolution was followed by a Leaky ReLU with a negative slope of 0.01 and group normalisation [33]. Group normalization was chosen over batch normalization to stabilize the training of the invertible components [23].

The core of the iU-Net was based on the work by Etmann et al. [23]. Two modifications were made to the original architecture proposed, firstly we replaced the first and last non-invertible layer of the encoder and decoder in the architecture, with the `DWT` and `iDWT`, respectively, and secondly, to match the input and output channels, we inflated the input channels and set them to zeros prior to the DWT. Given the convolutions in the iU-Net only act on half the channels, we increased the number of additive coupling layers to four, to keep the total number of parameters in the architectures comparable.

The Chimeric U-Net is the combination of the encoder of the modified sU-Net and the decoder of the modified iU-Net. Once again, to match the overall number of parameters between the architectures, we reduced the number of additive coupling layers in the decoder units to be two. Descriptions of the models are given in Appendix S-VI.

All the U-Nets were build to have a depth of four. The architectures were implemented in `PyTorch` v1.9 [34]. We used the *Cross Entropy Loss* function from the package as the loss function, furthermore, we used the Adam optimiser and trained with a learning rate of 10^{-4}. We chose the model with the smallest loss on the respective validation set. The training was performed on a Nvidia Tesla P100-SXM2 and the subsequent analyses on an Intel Xeon Gold 6246 CPU.

3.2. Datasets

We evaluate the U-Nets on two publicly available *Magnetic Resonance Imaging* (MRI) datasets: The Multimodal Brain Tumor Segmentation Challenge 2017 (BraTS), and the Zuse Institute Berlin's curated Osteoarthritis Initiative dataset 2018 (OAI ZIB).

***BraTS* [35], [36], [37]:** The dataset contains densely annotated *low grade gliomas* (LGG) and *high grade glioblastomas* (HGG). For each patient: a T1 weighted, a post-contrast T1-weighted, a T2-weighted, and a *FLAIR* channel was available. Unlike other *BraTS* challenges, the 2017 edition contains only four target classes, namely: *background*, *edema* (ED), *non-enhancing tumor* (NET), and *enhancing tumor* (ET). The performance analysis was done for all classes, and also performed the analysis on the post-hoc merged class *tumor core*, which is the combination of ET and NET. The segmentation task was performed on 2D slices, consequently, 2D slices (240×240 pxls) of the 3D volumes were extracted. This was done by sampling three different slices from the inner 30% of each MRI scan. We chose a batch size of eight and chose to train for a maximum of 350 epochs. The data was split into five folds after reserving 10% of the data for testing. The splits were performed, such that samples of the same patient were not distributed among different data splits. In regards to data augmentation, flipping, blurring, random brightness, and elastic deformation were applied to the inputs during training (see Appendix S-I). We chose this particular dataset from the collection of *BraTS* datasets, because the chimeric- and invertible architectures did not require further channel adjustment to make the input channels match the number of target classes.

***OAI ZIB* [38], [39]:** The dataset consists of roughly 500 3D MRI scans of human knees. The segmentation task was to predict five target classes, namely: *background* (BG), *femoral bone* (FB), *tibial bone* (TB), *femoral cartilage* (FC), and *tibial cartilage* (TC). As done for the *BraTS* dataset, 2D slices (384×384 pxls) were sampled from the inner 30% of each patients MRI. We chose a batch size of eight and chose to train for a maximum of 125 epochs. As for the BraTS dataset, the data was split into five folds after reserving 10% of the data for testing. During training the inputs were augmented by horizontal flipping, blurring and brightness changes, as mentioned above.

***Tesselation dataset* [40]**: For the analysis of saliency maps we utilized the synthetic *Tesselation dataset* from the *Gap-Filling* problem. We trained a Chimeric U-Net for a maximum of 150 epochs with a batch size of eight. Thousand data samples were generated, each of size 240×240 pxls, and split into a train (80%), validation (10%) and test set (10%). In contrast to the Chimeric U-Nets trained for the *OAI ZIB* and *BraTS* dataset, no data augmentation was applied. The other hyperparameters were set as stated above (Methods 3.1).

3.3. Evaluation metrics

We employed standard computer vision metrics for medical image segmentation, such as class-wise and mean: *Intersection over Union* (IoU), *Recall*, and *Precision*.

3.4. Pull-back gradient approximations

Constructing the latent space $\mathcal{T}_{\mathcal{X}}$ for a dataset is computationally challenging. For this reason, we chose to approximate the elements in $\mathcal{T}_{\mathcal{X}}$ with coarse- and fine resolution approximations. In these computations our set $\mathcal{Y}^{\dagger}$ are the activation prior to applying the `softmax` (inline with Remark 2.1).

Coarse approximation: for $\mathbf{y} \in \mathcal{Y}^{\dagger}$ and class $c \in \{1, \dots, C\}$, our approximation of $\mathbf{t}_{\mathbf{y},c} \in \mathcal{T}_{\mathcal{X}}$ is given by,

$$\bar{\mathbf{t}}_{\mathbf{y},c} := \Psi^{-1}(\mathbf{y}) - \Psi^{-1}(\Delta_c(\mathbf{y})), \tag{3.1}$$

where we set Δ_c to be the operator that decreased the model prediction of class c by 70% (see Appendix Algorithm 3). This is a simple first-order difference approximation, assuming Ψ^{-1} is differentiable everywhere, the error can be bounded above by a scalar of the Lipschitz coefficient. We denote the set of coarse approximation of the set $\mathcal{T}_{\mathcal{X}}$ by $\bar{\mathcal{T}}_{\mathcal{X}}$. This approximation was used to generate the latent space embedding in the numerical experiments.

Fine approximation: We used a fourth-order numerical derivative approximation (5 point stencil) to estimate the partial derivative,

$$\begin{aligned}\frac{\partial \Psi^{-1}(\mathbf{y})}{\partial \mathbf{y}_{i,c}} \approx &\frac{-\Psi^{-1}(\mathbf{y}_{i,c}+2h) + 8\Psi^{-1}(\mathbf{y}_{i,c}+h)}{12h} \\ &+ \frac{-8\Psi^{-1}(\mathbf{y}_{i,c}-h) + \Psi^{-1}(\mathbf{y}_{i,c}+2h)}{12h},\end{aligned} \tag{3.2}$$

with $h = 0.3$ as the step size. Here the approximation error is of $\mathcal{O}(h^4)$. We substituted the approximation of the partial derivative into (2.11) and use the predicted class probability as the weight $w_{\mathbf{y},i,c}$. We denote the set of fine approximations of the elements of $\mathcal{T}_{\mathcal{X}}$ by $\tilde{\mathcal{T}}_{\mathcal{X}}$.

3.5. Vectorizing and embedding the pull-back gradients

Performing the analysis on the pull-back gradients, $\bar{\mathcal{T}}_{\mathcal{X}}$, was not numerically feasible, and furthermore, we had to remove the spatial information, which was still encoded in them. To achieve this, we constructed a collapsing function,

$$\kappa : \mathbb{R}^{dim1,dim2,dim3} \times \mathbb{N} \to \mathbb{R}^{dim1}, \tag{3.3}$$

which takes an integer as input to determine how $dim2$ and $dim3$ are collapsed. The operation of this function is more suited to be presented as pseudo-code, hence, we present it as such in Appendix Algorithm 1. In this work, we chose the integer for collapsing to be the number of pixels which were positively predicted in the target. That is, let $\mathbf{t}_{y,c} \in \bar{\mathcal{T}}_{\mathcal{X}}$, we performed a pull-back of y w.r.t. class c. We define $k_{y,c}$ to be the number of pixels which were predicted (relative highest probability) to be class c in the target y. Then the collapse of $\mathbf{t}_{y,c}$ was given by,

$$\mathbf{g}_{y,c} := \left(\kappa(t_l, \lceil k_{y,c}/4^l \rceil)\right)_{l=1,\dots,L}, \tag{3.4}$$

where t_l is the l^{th} element of $\mathbf{t}_{y,c}$. We refer to $\mathbf{g}_{y,c}$ as the *vectorized activation* of $\mathbf{t}_{y,c}$. We chose to take the average of the top k pixels per channel to collapse, as opposed to averaging the entire channel, to focus our attention on the most excited neurons. We found that most activations were close to the mean excitation corresponding to the encoding of background features. This was performed for all elements of $\bar{\mathcal{T}}_{\mathcal{X}}$ and we chose to visualize only the vectorized activation from only the deepest layer.

We used the t-SNE implementation in Python v3.7 [41] to visualize the vectorized activations. We chose *dim* to be 2, *perplexity* to be 150, *early exaggeration* to be 10, and the *correlation metric*. We normalised our data according to unit norm using ℓ_2 norm before performing the embedding. The embedding was constructed after mixing the gradient pull-backs of $\mathcal{X}_{Train}$ and $\mathcal{X}_{Test}$.

Table 1
Performance on BraTS $\mathcal{X}_{\text{Test}}$ for enhancing tumor (ET) and tumor core (TC).

Method (U-Nets)	IoU (%)		Recall (%)		Precision (%)	
	ET	TC	ET	TC	ET	TC
Standard	69 ± 13	69 ± 7	83 ± 9	87 ± 8	80 ± 11	77 ± 9
Chimeric	65 ± 7	68 ± 4	81 ± 10	86 ± 6	77 ± 10	77 ± 7
Invertible	64 ± 3	63 ± 3	85 ± 6	88 ± 5	73 ± 5	69 ± 6

$(\mu \pm \sigma)$ is such that $[\mu - \sigma, \mu + \sigma]$ is the 95% confidence interval.
Remaining classes are shown in Appendix S-IV TABLE S-1.
Number of trainable parameters:
sU-Net 367k (100%), Chimeric U-Net 273 K (74%), iU-Net 247k (67%).

3.6. KNN classifier

We used a k-nearest neighbours (kNN) implementation given in `scikit-learn` [42]. We set $k = 10$ and constructed a function to map from the vectorized activations of $\mathcal{T}_{\mathcal{X}}$ (as defined in (3.4)) to the IoU scores. The IoU score for an instance $\mathbf{g}_{\mathbf{y},c}$ is constructed by summing the IoU scores of the k nearest neighbours with the same class $\mathbf{g}_{\bullet,c}$ and dividing by k. These calculations were performed layer-wise, and then for the final score, the average score over all layers were taken for robustness.

3.7. Saliency maps

We used the fine gradient approximations (Methods (3.2)) to study saliency maps. In order to obtain the saliency map, the Hadamard product of the elements from the gradient pull-backs, $\tilde{\mathcal{T}}_{\mathcal{X}}$, and pull-backs, $\mathcal{Z}$, were summed, as in HiResCAM [25]. The saliency maps were then sequentially upsampled, the corresponding pseudo-code is given in Appendix Algorithm 2.

4. Results

In this section, we evaluate the qualitative and quantitative performance of the Chimeric U-Net compared to the iU-Net and the sU-Net for the BraTS- and OAI ZIB datasets. Furthermore, we present the approximations of the gradient pull-backs of the Chimeric U-Net for the BraTS- and the Gap Filling datasets, in the context of explainability.

4.1. Combining a non-invertible encoder and an invertible decoder improves performance in comparison to the fully invertible U-Net

We evaluated the performance of the three architectures on BraTS $\mathcal{X}_{Test}$ with five replicates and found that, the non-invertible sU-Net performed on average better with respect to the IoU and Precision scores over all the target classes (see Table 1). However, we also found an overlap of the 95% confidence intervals of the scores between the three architectures, suggesting that the performances were not necessarily significantly different. When we compared the Precision and Recall scores of the architectures for the classes NET and ET, we found that the iU-Net performed the worst. The observed decrease in Precision could be attributed to the lack of reduction of information by the iU-Net's invertible encoder. Overall, we found that the performance scores of the Chimeric U-Net were closer to that of the sU-Net than the iU-Net, even though the number of trainable parameters for the Chimeric U-Net were closer to the iU-Net ($\approx -10\%$) than the sU-Net ($\approx +30\%$).

When we looked further into the shape of the individual predicted segmentations, we again found that the Chimeric U-Net prediction performance was between the sU-Net and the iU-Net. For example, the predictions of the sU-Net showed smoother contours along ED and TC (see segmentations in Fig. 3 (Row 2)). In contrast, the Chimeric U-Net and the iU-Net predictions showed rough boundaries (see segmentations in Fig. 3 (Row 3 and 4)). However, the iU-Net also produced artifacts (false positive predictions) on arbitrary positions on the image (see Fig. 3 (•)) and rougher boundaries than the Chimeric U-Net (see segmentations in Fig. 3 (Column 4, 5 and 7)). Given the Chimeric U-Net and iU-Net have a similar decoder, we expected a closer behaviour, however, we found having a non-invertible encoder in the Chimeric U-Net, aided in reduction of the false positives rate and rough boundaries as observed in the iU-Net. We observed that all U-Nets struggled with detecting ED for low intensity samples (see yellow seg. in Fig. 3 (Column 6)). However, we found the predictions of the Chimeric U-Net to be closer to the sU-Net than the iU-Net (see yellow seg. in Fig. 3 (Column 2-5)). The Chimeric U-Net showed the precisest prediction for ED, whereas the sU-Net and iU-Net over-predicted it (see Fig. 3 (Column 2 and 5)), sometimes at regions where no ED was present (see yellow seg. in Fig. 3 (‡)), suggesting that the Chimeric U-Net is much more robust against intensity fluctuations within its input.

In summary, having a non-invertible encoder aids in performing a better segmentation task and by this the Chimeric U-Net is at least as good as the iU-Net.

As a sanity check, we also trained the three architectures on the OAI ZIB dataset. Here we observed a similar trend in performance between the three architectures as with the BraTS dataset (see Table 2). That is, the sU-Net scores on average were better than that of the Chimeric U-Net and the iU-Net. In particular, we observed that the 95% confidence intervals of the Recall and Precision scores of the three architectures in the FC and TC classes were overlapping. Interestingly, the sU-Net had significantly better IoU scores ($[74\%, 76\%]$) in the FC class to the Chimeric U-Net ($[71\%, 73\%]$), but not to the iU-Net ($[72\%, 74\%]$), again showing that the Chimeric U-Net is not statistically different in the IoU distribution than theiU-Net. Looking closer at the individual predicted segmen-

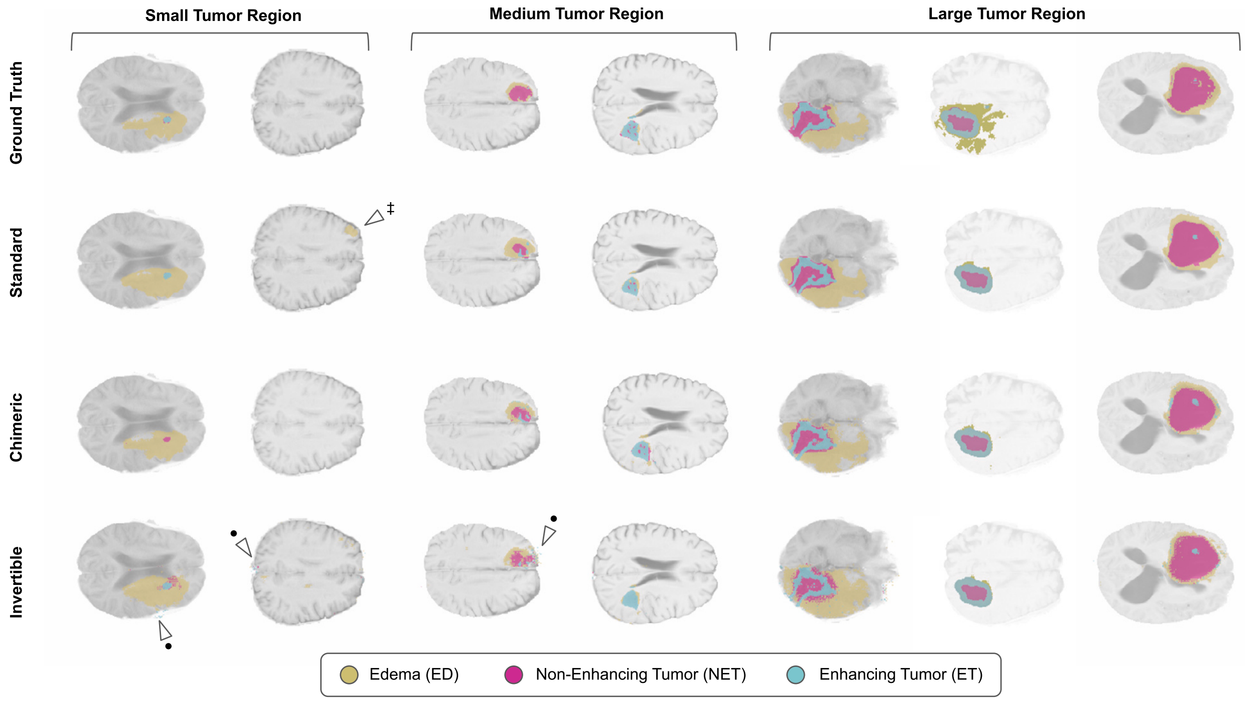

Fig. 3. Segmentation results for samples from BraTS 2017 test dataset over input images (grey scale background) for the classes *edema* (yellow), *enhancing tumor* (blue) and *non-enhancing tumor* (purple). Rows show the ground truth and the models segmentation for standard, chimeric and non-invertible U-Nets. The columns show different samples further grouped by the severity of the tumor. The markers highlight observations discussed in the Results 4.1.

Table 2
Performance on OAI ZIB $\mathcal{X}_{\text{Test}}$ for *femoral cartilage* (FC) and *tibial cartilage* (TC).

Method (U-Nets)	IoU (%)		Recall (%)		Precision (%)	
	FC	TC	FC	TC	FC	TC
Standard	75 ± 1	61 ± 1	85 ± 3	74 ± 6	87 ± 3	78 ± 5
Chimeric	72 ± 1	58 ± 2	82 ± 3	70 ± 3	86 ± 1	78 ± 4
Invertible	73 ± 1	56 ± 3	84 ± 1	66 ± 5	84 ± 1	79 ± 3

$(\mu \pm \sigma)$ is such that $[\mu - \sigma, \mu + \sigma]$ is the 95% confidence interval.
Remaining classes are shown in Appendix S-IV TABLE S-2.
Number of trainable parameters:
sU-Net 574k (100%), Chimeric U-Net 427 K (74%), iU-Net 386k (67%).

tations on the OAI ZIB test set, we observed the sU-Net to produce slightly smoother boundaries (best visible in Fig. 4 (Column 2)). Besides that, the overall differences between the architectures were subtle. The iU-Net was found to produce the best segmentations for FC, whereas the other architectures mostly over predicted it (see pink seg. in Fig. 4 (★)). For TC, we observed it to be vice versa, i.e. the iU-Net over predicted TC. This can be seen for the second sample of Fig. 4, for which only the ChimericU-Net and sU-Net predicted correctly the absence of TC (see green seg. in Fig. 4 (‡)). Furthermore the Chimeric U-Net often showed more precise predictions for TC (see green overlay in Fig. 4 (■)). Lastly, for the sU-Net we noticed sometimes false predictions in bright spots of the images (see orange seg. in Fig. 4 (•)).

In summary, the Chimeric U-Net is on par –at times better– with the iU-Net for segmentation tasks over the BraTS and ZIB OAI datasets. The non-invertible encoder is needed to increase precision, through the removal of redundant information. Combining a non-invertible encoder with an invertible decoder, as in the Chimeric U-Net, gave results closer to the purely non-invertible setting. Nevertheless, we observed a minor performance drop of the Chimeric U-Net to the sU-Net, but the trade-off for this is in the explainable properties, which comes from the invertible decoder of the Chimeric U-Net, which we demonstrate next.

4.2. *The vectorized activations contain a meaningful structure w.r.t. the target classes*

We used the coarse grained derivative approximation (Eq. (3.1)) to compute the pull-back gradients of the individual segmentations (see Methods 3.4 - 3.5) of the BraTS dataset and then used the t-SNE embedding to visualize their corresponding vectorized activations (see Fig. 5).

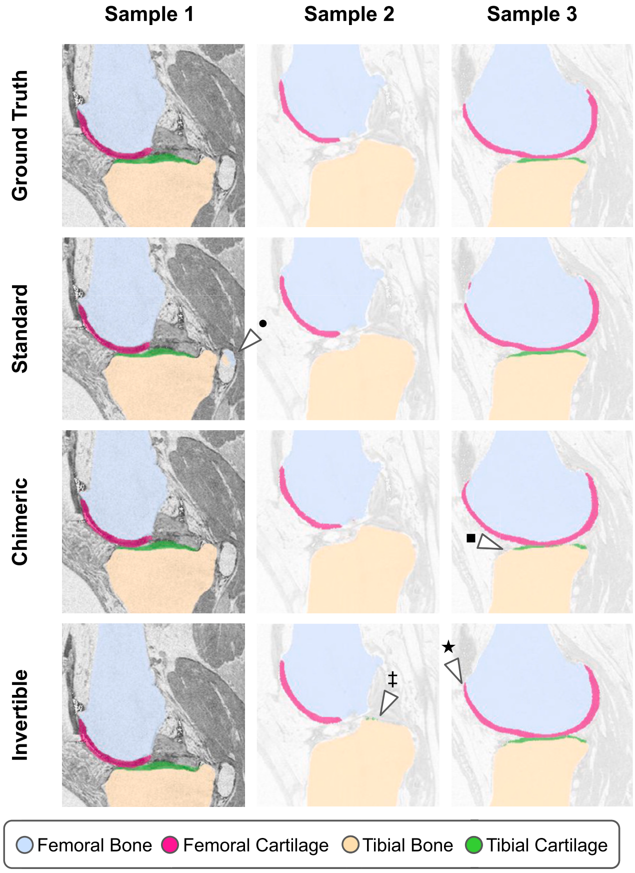

Fig. 4. Segmentation results for samples from OAI ZIB test dataset over input images (grey scale background) for the classes *femoral bone* (blue), *tibial bone* (orange), *femoral cartilage* (pink) and *tibial cartilage* (green). Rows show the ground truth and the models segmentation for standard, chimeric and non-invertible U-Nets. The markers highlight observations discussed in the Results 4.1.

In the embedding of the train dataset ($\mathcal{X}_{Train}$), we observed that the three target classes: NET (purple), ET (blue), and ED (yellow), clustered distinctly from each other, with slight overlap of the two tumor types, i.e. NET and ET (see Fig. 5 (■)). Furthermore, we observed in the embedding, that the ED points had a small sub-cluster closer to the other classes (see Fig. 5 (★)), suggesting that segmentation of ED corresponding to these points was perhaps different to the remaining ED points.

To understand the meaning of the spatial distances between the vectorized activations, we overlayed the IoU scores of the respective class prediction on the t-SNE visualisation (see Fig. 6). Surprisingly, we saw that the smaller ED cluster points and the overlapping NET and ET points showed low IoU scores (see Fig. 6 (★)). Additionally, instances of images from the mixed regions of NET and ET were falsely predicted as the other class (see Fig. 6 (‡)), as expected. Lastly, we saw that there were regions within each class cluster, where all the points in the spatial proximity had similar high scores (e.g. Fig. 6 (•)). Recall, that we normalized the pull-back gradients by the size of the class positives (see (3.4) to reduce the influence of varying sizes of predictions.

In summary, we found that the vectorized activations of the target class predictions cluster in a meaningful way, driven by the class membership and the prediction performance.

4.3. The vectorized activations form a meaningful latent space to infer the accuracy of unseen data

We investigated if the IoU score relationship to the spatial proximity also translated to unseen data ($\mathcal{X}_{Test}$). For this reason, we employed a simple kNN-classifier (see Methods 3.6) to estimate the IoU score corresponding to vectorized activations based on the local proximity to the vectorized activations from the training data. When overlayed onto the embedding of the vectorized activations of the test dataset the evaluated IoU score and the kNN estimated IoU scores, we found that they were visually similar (see Fig. 7a–b). Furthermore, plotting the estimated verses the evaluated IoU scores, we observed that the estimated IoU scores were a weak upper bound to the evaluated scores, and that the two became positively correlated for higher score values (Fig. 7 c).

Given the pull-backs were made for all classes predicted by the Chimeric U-Net, naturally, the *false positives* (FPs) were also pulled back. Hence, we observed 58 vectorized activations which had positive estimated IoU, but scored a value of zero, under this metric (Fig. 7 c (x-axis)). Even though these were FPs, the kNN estimate gave only roughly 5% of the vectorized activations a score of above

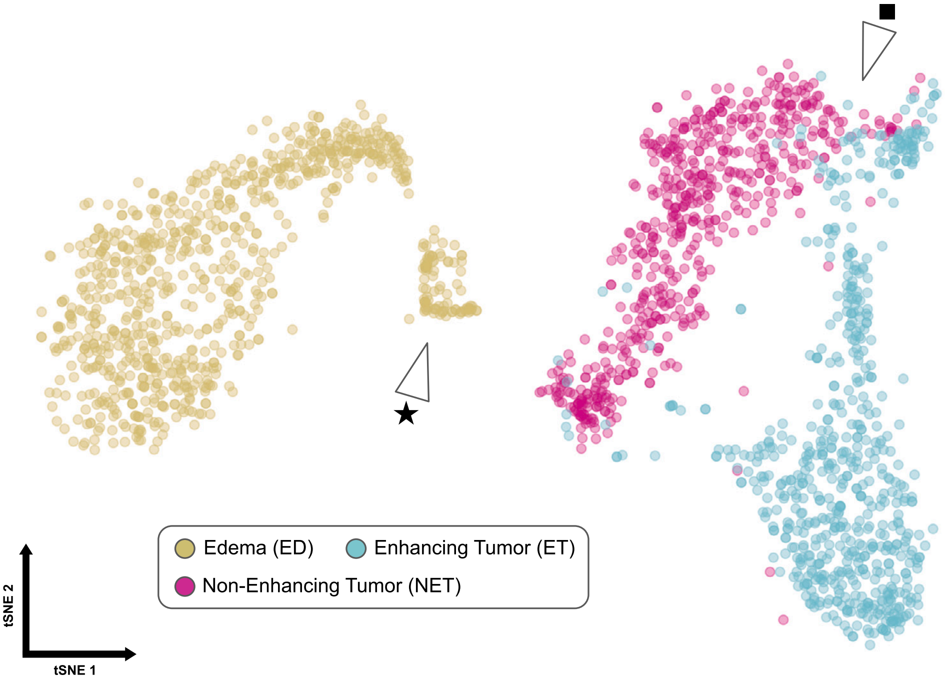

Fig. 5. Visualization of t-SNE embedding of vectorized activations for BraTS $\mathcal{X}_{Train}$ performed by Chimeric U-Net. Colours display target classes, namely *ED* (yellow), *ET* (blue) and *NET* (purple). Each point represents a single pull-back for one class of target sample. The markers highlight observations discussed in the Results 4.3.

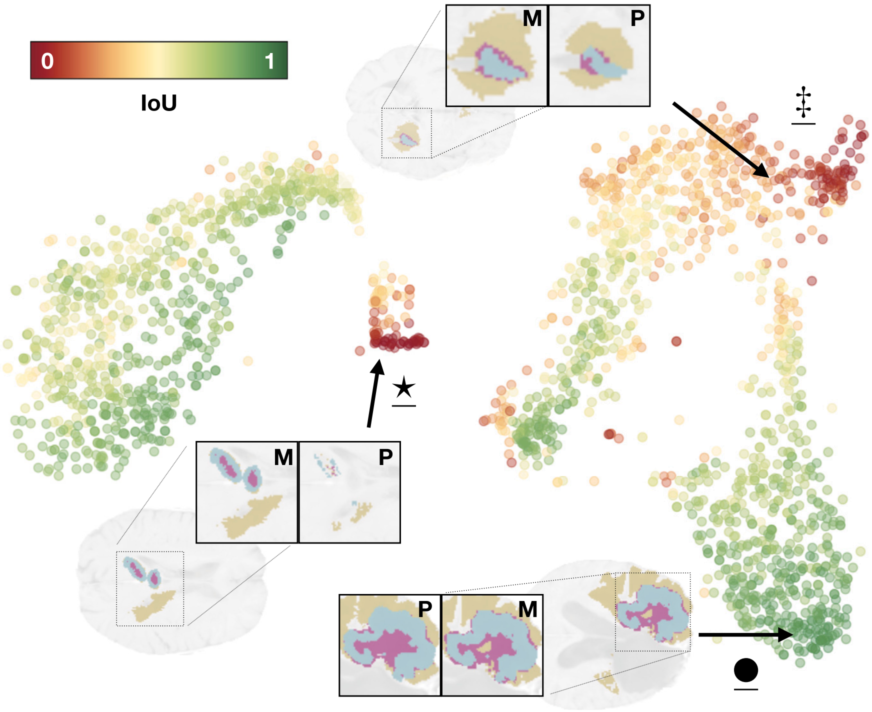

Fig. 6. Visualization of t-SNE embedding of the vectorized activations for BraTS $\mathcal{X}_{Train}$ coloured by evaluated per class IoU, performed by the Chimeric U-Net. The brain scans depict exemplary samples from different positions within the embedding, overlayed with their respective model's prediction (**P**) and the ground truth mask (**M**). The markers highlight observations discussed in the Results 4.3.

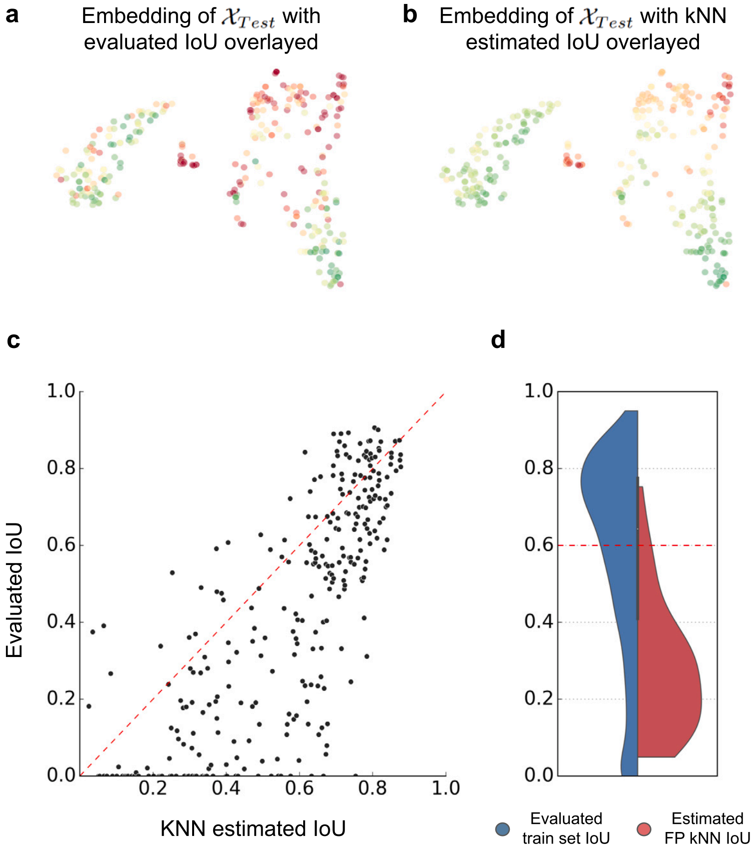

Fig. 7. Top: Visualization of t-SNE embedding of vectorized activations for BraTS $\mathcal{X}_{Test}$ performed by the Chimeric U-Net. Colours display (a) the evaluated IoU scores (b) and the IoU scores estimated with the kNN-classifier. Bottom: Comparison of evaluated and estimated IoU. (c) Overall fit of kNN-classifier-estimate versus evaluated IoU for $\mathcal{X}_{Test}$ and (d) contrast of estimates of the kNN-classifier for false positive predictions to the evaluated IoU scores for $\mathcal{X}_{Train}$.

60% (see Fig. 7 d). In addition, the majority of the FPs had a low estimated score of roughly 20%. When considering the evaluated IoU scores for $\mathcal{X}_{Train}$, it can be seen clearly that those low scores correctly represented poor prediction performance.

In summary, the clustering of the vectorized activations of the training dataset encodes how confident the architecture is in its class predictions. More importantly, the vectorized activations of the training data can be used to give an estimate on the true accuracy of the prediction of unseen data.

4.4. The vectorized activations of the iU-Net differ from that of the cU-Net

Given that both the cU-Net and iU-Net employ a similar non-invertible decoder, we wanted to investigate if the iU-Net exhibited a similar clustering pattern in the vectorized activations. We used the iU-Net decoder to conduct an analysis that was similar to that presented in Section 4.2 done for the cU-Net. The vectorized activations of the target classes; Edema (ED), Enhancing Tumor (ET), and Non-Enhancing Tumor (NET) of the iU-Net decoder clustered distinctly w.r.t. the classes (see Fig. 8 (top row: middle)). Unlike in the cU-Net case, we did not observe any overlap between the clusters corresponding to ET or NET. Hence, the interactions between the classes was not observed, meaning that each class is predicted by a distinct set of activations in the iU-Net.

Next, as done in Section 4.3, we superimposed the IoU over the t-SNE embedding of the vectorized activations to see if the clusters as well spatially encoded the IoU as observed in the cU-Net. We found that the IoU of the classes ET and ED could be delineated easily. Furthermore, we observed that within their own clusters the points corresponding to images mapping to a high IoU prediction were opposite to points corresponding to a low IoU prediction (see Fig. 8 bottom Row (middle)). However, the class of Non-Enhancing Tumor (NET) showed no delineation between the points corresponding to high and low IoU scoring images (see Fig. 8 bottom Row (⋆)), the IoU was spatially mixed. Hence, even though the decoders are the same for the iU-Net and the cU-Net, the differences in the encoders do contribute to different latent representation structures.

The pull-backs are an inherent property of the invertible decoders, by extension, as are the class delineated embeddings of the latent representations. For consistency, we computed the Grad-CAM based vectorized activations as described in SEG-Grad-CAM [12]. We found that the class ED separated from NET and ET, but no separation between NET and ET was observed (see Fig. 8 top Row (right)). We overlayed the IoU onto the embedding and did not observe a clear meaningful relationship between the spatial distribution of points and their corresponding predicted IoU. We also computed the SEG-Grad-CAM based vectorized activations on the Chimeric

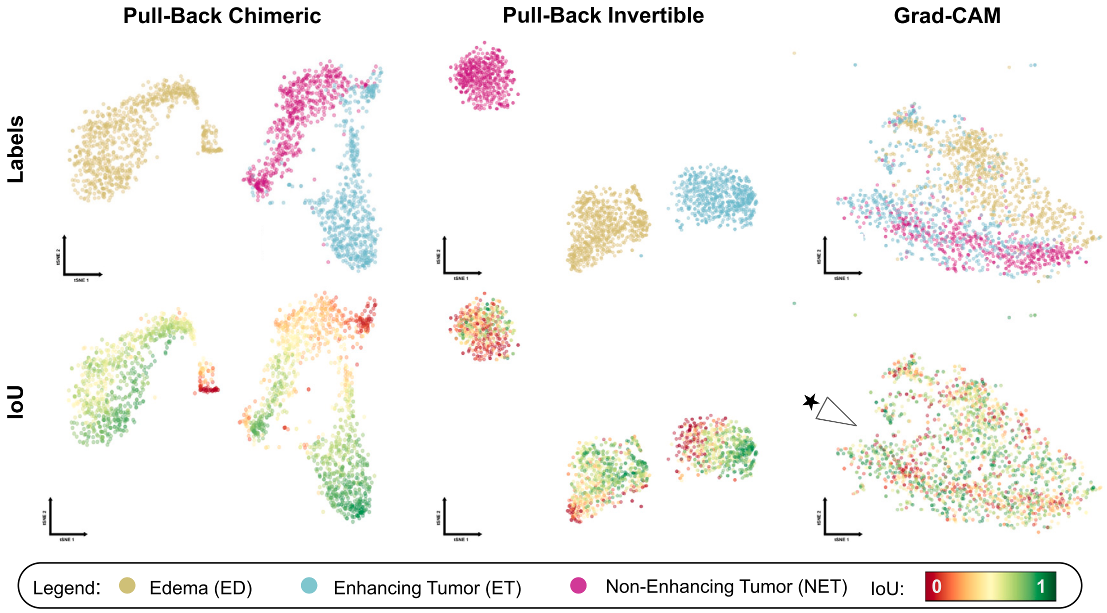

Fig. 8. Visualization of t-SNE embedding of vectorized activations for BraTS $\mathcal{X}_{Train}$ performed by sU-Net, iU-Net, and Chimeric U-Net. Top: Colours display target classes, namely *ED* (yellow), *ET* (blue) and *NET* (purple). Bottom: Colours display evaluated per class IoU.

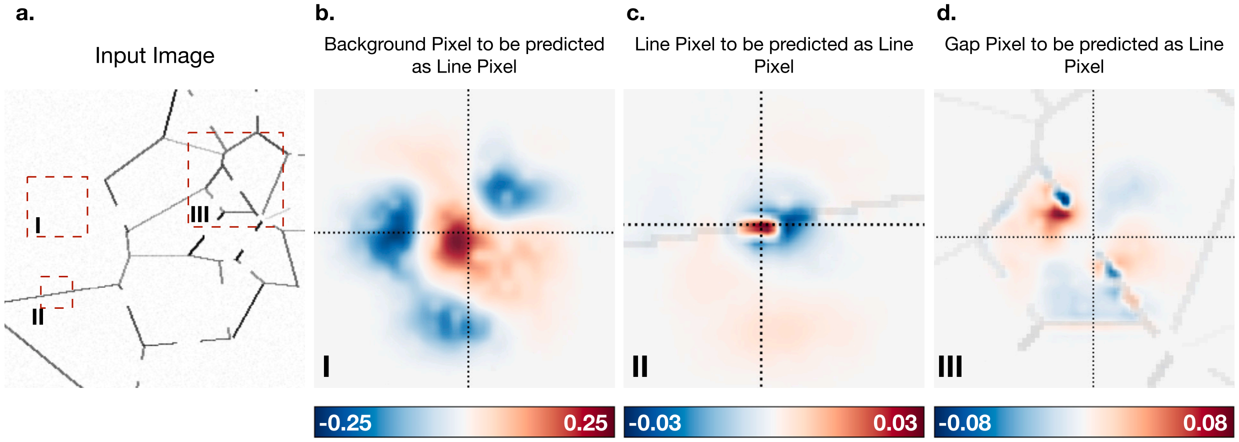
a.
Input Image
I
II
III
b.
Background Pixel to be predicted as Line Pixel
I
-0.25
0.25
c.
Line Pixel to be predicted as Line Pixel
II
-0.03
0.03
d.
Gap Pixel to be predicted as Line Pixel
III
-0.08
0.08

Fig. 9. Saliency maps for different pixels (indicated by dashed cross) of sample from gap dataset. Only for pixels that had to be filled in order to fill a gap relied only on context information. The colouring shows the positive (red) and negative (blue) shaping the saliency maps.

U-Net and found that the classes separated similarly to that of the pull-back activations, however, when we overlayed the IoU, we did not observe any clear spatial delineation of the IoU within the clusters w.r.t. the target classes (see Appendix Fig. 7).

4.5. Pull-back gradients from the invertible decoder can be used to compute saliency maps of segmentations

Saliency maps are the state-of-the-art approach for explainability in classification tasks. We investigated if the decoder in the Chimeric U-Net could be used to also generate saliency maps of the pixel predictions in the segmentation task. We found that both, the BraTS and OAI ZIB, dataset sets produced trivial saliency maps, suggesting that the segmentation task was driven by texture rather than by context (Appendix Figs. 8-13). Hence, to detect context decisions made by the Chimeric U-Net, we utilized the synthetic *tessellation dataset* from the *Gap-Filling* problem. In this task, the architecture had to segment lines in the input image, but additionally also had to fill the gaps between incomplete lines, for which the gap pixels are indistinguishable from background pixels (Methods 3.1).

The Chimeric U-Net performed well on the *Gap-Filling* problem, as lines were correctly segmented with an IoU score of 86%. We choose a representative source image from the test set to be segmented. On this image, we picked three points of interest: a background pixel, a line pixel, and a gap pixel (see Fig. 9 a (dashed boxes)). For these pixels, we computed their respective saliency maps, that is, highlighting the regions in the source image which aided in these pixels to be classified as a line pixel (Methods 3.6).

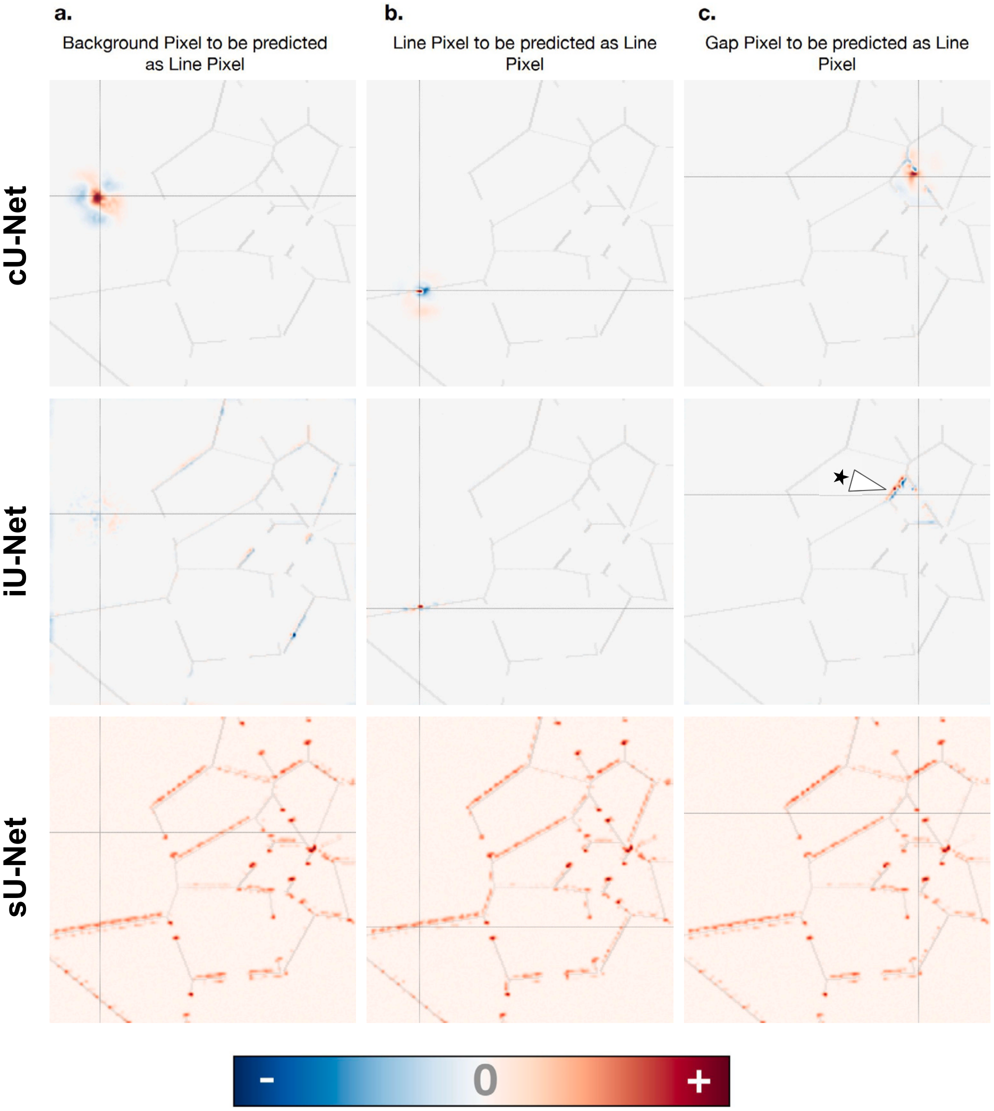

Fig. 10. Saliency maps for different pixels (indicated by dashed cross) of a sample from the tesselation dataset for the sU-Net, iU-Net, andChimeric U-Net. The saliency maps of the iU-Net and Chimeric U-Net were constructed with pull-backs, whereas for the sU-Net SEG-Grad-CAM was utilized. The colouring shows the positive (red) and negative (blue) attributions of the saliency maps.

Firstly, for the background pixel, we saw that the highest saliency was in proximity of the pixel itself and the signal diffused outwards, mimicking the receptive field (see Fig. 9 b). Secondly, for the line pixel, we saw a strong saliency signal on the line segment, and some weaker signal radiating away from the line (see Fig. 9 b). Lastly, for the gap pixel, we saw that the saliency was highest at the ends of the lines at either end of the gap, with nearly no signal at the pixel, showing that the prediction is explained by the context of the edges on either side of the gap.

We also computed the saliency maps for the outputs of the iU-Net and the sU-Net. The saliency maps of the iU-Net were computed similar to the saliency maps of the cU-Net (see Methods (3.1)), and for the saliency maps of the sU-Net we followed the adaption of Grad-CAM for segmentation tasks [12]. We found the explanations of the iU-Net to be comparable to that of the cU-Net (see Fig. 10 Row 1 -2). However, for the case of the gap pixel, we found the iU-Net to highlight not only the edges at the end of the explained gap segment (as seen for cU-Net), but also on line pixels in its proximity unrelated to the gap position (see Fig. 10 Row 2 (★)).

On the other hand, the saliency maps for the sU-Net computed with SEG-Grad-CAM showed strong differences to saliency maps derived from pull-backs (see Fig. 10 Row 1-2 vs 3). Those explanations showed strongest highlighting on edges of the line enclosing the gap, suggesting the architecture has a clear understanding of the two ends of the gap (see Fig. 10). However, the explanations for the single gap pixel also highlighted every other line edge enclosing a gap position unrelated to the position of the gap pixel to be explained. Furthermore, when explaining a background pixel or a line pixel the attribution shape did not significantly change. It is interesting to note that for the three pixels we chose, when we calculated the derivative with respect to predicting this pixel as a line

(Grad-CAM saliency), the saliency highlighted all pixels that appeared to be a line. That is, changes in the intensity of any line pixel has an impact on the individual pixel being classified as line. This is different to the pull-backs of the invertible which more locally highlight the locals changes which would impact the pixel being classified to be class line.

In summary, the pull-back gradients from the invertible decoder can be easily used to construct saliency maps. For the context based gap-filling problem, we observed that the cU-Net and the iU-Net pull-backs highlighted the local contextual region. In contrast, the SEG-Grad-CAM explanations of the sU-Net highlighted a more global context.

5. Discussion

We proposed the Chimeric U-Net, a novel explainable CNN architecture which combines a non-invertible encoder and an invertible decoder to successfully solve segmentation tasks with built-in explainability. By taking this combination, we inherit the explainability analysis from the iU-Net and the feature extraction of the sU-Net encoder.

The discussion points below are structured to match the points presented in the results, that is, we discuss the impact and limitations of: the technical aspects of the architecture, the class disentangled latent space embedding, and lastly, the pull-back gradients based saliency maps.

We showed that the Chimeric U-Net performed en part with the purely invertible U-Net on two biomedical image segmentation tasks, suggesting that the non-invertible encoder has little negative impact on the performance. In contrast, the Chimeric U-Net performed slightly worse compared to the sU-Net, suggesting that the invertible decoder does have some negative effect on performance. Here, we proposed a minimal architecture to demonstrate the concept of using an invertible decoder, since technical and practical aspects of architecture analysis were not direct focus of this work. We speculate that the decrease in performance shown in this work is primarily due to the inverse discrete wavelet transform (`iDWT`) that we introduced to obtain the total invertibility of the iU-Net. Given the `iDWT` has no trainable parameters, this enforces the Chimeric U-Net to learn the segmentation in the steps prior to its application. We performed post hoc experiments, in which the `iDWT` was exchanged with non-invertible standard convolutions, and observed that this enhanced the segmentation quality, at the cost of the model's explainability (results not shown). Naturally, higher order basis functions for the `iDWT` can be chosen, and also, it can be exchanged with other invertible neural functions. In practice, this should be tailored to the dataset of interest. Given the flexibility of the encoder in the Chimeric U-Net, the optimal choice of the architecture's encoder and its influence on the information bottleneck over the skip connections is still to be explored. Another technical advantage we did not exploit in this work is the memory consumption savings given by the invertible components as presented by Etmann et al. This would be a natural future direction of research to construct bigger Chimeric U-Net architectures.

Constructing an embedding that can estimate the quality of the data for which no ground truth is available has tremendous impact in fields like healthcare, in which large amounts of data are generated [43]. As an example, we can envisage that the embedding could be used to tag samples which would need re-imaging due to artefacts and noise. Furthermore, in rural regions, where radiologist's visits are scarce, the embedding could help prioritize the images which are out of confidence of the architecture, for diagnosis. Furthermore, with the establishment of laws such as the EU AI: Act, patients can demand local saliency and global embedding as evidence of rightful diagnostics and practices. What is broadly referred/interpreted as *the right to explainability*. The use of the derivative of the pull-backs with respect to the classes (pull-back gradients) was a fruitful idea for constructing a class-disentangled latent space embedding. We could see that in the skip connections of the Chimeric U-Net architecture, different classes were passed over different activations, and over the same activations when there was a confusion in delineating the classes. This feature splitting was anticipated given the bottleneck caused by the non-invertible decoder, and since clustering of the latent space with respect to the classes was already previously observed in classical classification tasks. However, what came as a surprise was how the distribution of points inside the clusters were correlated with the IoU, indicating that there is a way to estimate the performance of unseen data through the latent space. Such observations were previously made for binary segmentation tasks [44], and now we could also observe this in a multi-class setting. This could naturally be used to check for generalisability of the architecture. For example, if the architecture was trained on a particular genealogical demographic and then was applied to another demographic, then we could check where the prediction of the two groups are located in the latent space and interpret hidden biases. A line of questioning which we didn't consider in this work was to investigate if the pull-back gradients could detect subclasses within a poorly labelled class. For example, if we had "enhancing tumor" and "non-enhancing tumor" labelled as a meta class "tumor", then we would want the pull-back gradients of the class "tumor" to be split into at least two clusters, where one cluster represented patients who had more enhancing tumor over non-enhancing, and vice versa. Another question we only explored in part was how the correlations between the classes form over the training period. By using a simple correlation measure, we can investigate which classes are correlated within the Chimeric U-Net architecture. We could further apply tools such as *Testing with Concept Activation Vectors* [45] with pull-back gradients to understand which classes contribute to specific concepts. This knowledge can be used for better training or model debugging, for example, by increasing specificity of the class labels, assessing data quality and normalization, and monitoring the training progress of the network [46], [47]. Hence, studying the pull-back gradients and their effectiveness to explain the overall behaviour of the architecture is a natural future research direction.

Historically, saliency maps were highlighted regions on an image where a viewer was focusing in visual experiments. This concept was translated into modern XAI, and since has been a major tool and topic of much ongoing debate. The latter stems from the lack of ground truth and quantitative metrics, thus making correctness and faithfulness of saliency methods contentious [29], [30], [48], [31], [32], [5]. However, saliency maps are important for human interpretation, hence, we wanted to investigate if the question (XAI β) could also produce a saliency map. To circumvent the criticism around validation of the saliency maps, we chose to use the gap-filling problem, where a mathematical proof was already given on where the saliency of a classification of a line segment should lay in the source image. We could visually verify that the saliency maps made from (XAI β) did highlight the expected regions in the input image, suggesting that (XAI β) could also construct meaningful contextual saliency maps. The saliency maps in this work were constructed for only the positive class predictions, however, a future direction would also be to consider the pull-back gradients of the false negative predictions and investigate their saliency maps. Lastly, recalling from the introduction, we do not give a comparison between (XAI α) and (XAI β) in this work. Despite the proliferation of many XAI methods over the past years, only a few focus on their evaluation. This is rooted to the absence of ground truths, making the comparison of (XAI α) and (XAI β) challenging. But with the emergence of metrics and ground truth for saliency maps, a clear future direction would be to investigate and categorize the various saliency maps arising from XAI questions.

In conclusion, the Chimeric U-Net arises from the principle of enforcing explainability into the architecture, rather than studying explainability of a general architecture. Through the invertibility of the decoder we could inherently produce both global- and local explainability through class embeddings and saliency maps, respectively. The motivation for this work was to bring explainability to the segmentation task as this is under studied in the field of XAI. However, if other DL-based tasks have a similar encoder-decoder compartmental structure, applying the Chimeric principle, that is, making the decoder invertible, would make the pull-back gradients automatically accessible. For example, the classification linear layers of VGG/Inception could be replaced by Normalizing flows [49], we could compute the pull-back directly to encodings. However, in practice, invertibility is a strong requirement for general DL-based tasks. We believe that the Chimeric U-Net architecture is a step in the right direction towards growing confidence, reliability, and trust in Deep Learning approaches in healthcare.

CRediT authorship contribution statement

Kenrick Schulze: Writing – review & editing, Writing – original draft, Validation, Methodology. **Felix Peppert:** Writing – review & editing, Software, Methodology. **Christof Schütte:** Writing – review & editing, Writing – original draft, Supervision, Funding acquisition, Conceptualization. **Vikram Sunkara:** Writing – review & editing, Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.artint.2024.104240.

Data availability

The authors do not have permission to share data.

References

[1] S. Asgari Taghanaki, K. Abhishek, J.P. Cohen, J. Cohen-Adad, G. Hamarneh, Deep semantic segmentation of natural and medical images: a review, Artif. Intell. Rev. 54 (1) (2021) 137–178.

[2] S. Mehralivand, D. Yang, S.A. Harmon, D. Xu, Z. Xu, H. Roth, S. Masoudi, D. Kesani, N. Lay, M.J. Merino, et al., Deep learning-based artificial intelligence for prostate cancer detection at biparametric mri, Abdom. Radiol. 47 (4) (2022) 1425–1434.

[3] P.K. Mallick, S.H. Ryu, S.K. Satapathy, S. Mishra, G.N. Nguyen, P. Tiwari, Brain mri image classification for cancer detection using deep wavelet autoencoder-based deep neural network, IEEE Access 7 (2019) 46278–46287.

[4] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[5] W. Samek, G. Montavon, S. Lapuschkin, C.J. Anders, K.-R. Müller, Explaining deep neural networks and beyond: a review of methods and applications, Proc. IEEE 109 (3) (2021) 247–278.

[6] B. Goodman, S. Flaxman, European Union regulations on algorithmic decision-making and a “right to explanation”, AI Mag. 38 (3) (2017) 50–57.

[7] O. Ronneberger, P. Fischer, T. Brox, U-net: convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.

[8] S. Ghosh, A. Pal, S. Jaiswal, K. Santosh, N. Das, M. Nasipuri, Segfast-v2: semantic image segmentation with less parameters in deep learning for autonomous driving, Int. J. Mach. Learn. Cybern. 10 (11) (2019) 3145–3154.

[9] J. Haspiel, N. Du, J. Meyerson, L.P. Robert Jr, D. Tilbury, X.J. Yang, A.K. Pradhan, Explanations and expectations: trust building in automated vehicles, in: Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, 2018, pp. 119–120.

[10] N. Siddique, S. Paheding, C.P. Elkin, V. Devabhaktuni, U-net and its variants for medical image segmentation: a review of theory and applications, IEEE Access (2021).

[11] G. Du, X. Cao, J. Liang, X. Chen, Y. Zhan, Medical image segmentation based on u-net: a review, J. Imaging Sci. Technol. 64 (2) (2020) 20508.

[12] K. Vinogradova, A. Dibrov, E.W. Myers, Towards interpretable semantic segmentation via gradient-weighted class activation mapping, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 13943–13944.

[13] L. Hoyer, M. Munoz, P. Katiyar, A. Khoreva, V. Fischer, Grid saliency for context explanations of semantic segmentation, Adv. Neural Inf. Process. Syst. 32 (2019).

[14] T. Koker, F. Mireshghallah, T. Titcombe, G. Kaissis, U-noise: learnable noise masks for interpretable image segmentation, in: 2021 IEEE International Conference on Image Processing (ICIP), IEEE, 2021, pp. 394–398.

[15] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.

[16] A. Inglis, A. Parnell, C.B. Hurley, Visualizing variable importance and variable interaction effects in machine learning models, J. Comput. Graph. Stat. 31 (3) (2022) 766–778.

[17] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, Adv. Neural Inf. Process. Syst. 30 (2017).

[18] M.T. Ribeiro, S. Singh, C. Guestrin, "why should I trust you?" explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144.

[19] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: visualising image classification models and saliency maps, arXiv preprint arXiv: 1312.6034, 2013.

[20] D. Smilkov, N. Thorat, B. Kim, F. Viégas, M. Wattenberg, Smoothgrad: removing noise by adding noise, arXiv preprint, arXiv:1706.03825, 2017.

[21] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, W. Samek, On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation, PLoS ONE 10 (7) (2015) e0130140, https://doi.org/10.1371/journal.pone.0130140.

[22] D. Rezende, S. Mohamed, Variational inference with normalizing flows, in: International Conference on Machine Learning, PMLR, 2015, pp. 1530–1538.

[23] C. Etmann, R. Ke, C.-B. Schönlieb, iunets: fully invertible u-nets with learnable up- and downsampling, arXiv preprint arXiv:2005.05220, 2020.

[24] G. Arvanitidis, S. Hauberg, B. Schölkopf, Geometrically enriched latent spaces, arXiv preprint arXiv:2008.00565, 2020.

[25] R.L. Draelos, L. Carin, Hirescam: faithful location representation in visual attention for explainable 3d medical image classification, arXiv preprint arXiv:2011.08891, 2020.

[26] A. Chattopadhay, A. Sarkar, P. Howlader, V.N. Balasubramanian, Grad-cam++: generalized gradient-based visual explanations for deep convolutional networks, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018, pp. 839–847.

[27] P.-J. Kindermans, K.T. Schütt, M. Alber, K.-R. Müller, D. Erhan, B. Kim, S. Dähne, Learning how to explain neural networks: patternnet and patternattribution, arXiv preprint arXiv:1705.05598, 2017.

[28] A. Chattopadhay, A. Sarkar, P. Howlader, V.N. Balasubramanian, Grad-Cam++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks, 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Mar 2018.

[29] L. Sixt, M. Granz, T. Landgraf, When explanations Lie: why many modified bp attributions fail, in: International Conference on Machine Learning, PMLR, 2020, pp. 9046–9057.

[30] A. Ghorbani, A. Abid, J. Zou, Interpretation of neural networks is fragile, Proc. AAAI Conf. Artif. Intell. 33 (2019) 3681–3688.

[31] R. Tomsett, D. Harborne, S. Chakraborty, P. Gurram, A. Preece, Sanity checks for saliency metrics, Proc. AAAI Conf. Artif. Intell. 34 (2020) 6021–6029.

[32] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K.T. Schütt, S. Dähne, D. Erhan, B. Kim, The (un) reliability of saliency methods, in: Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer, 2019, pp. 267–280.

[33] Y. Wu, K. He, Group normalization, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3–19.

[34] 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, S. Chintala, Pytorch: an imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, vol. 32, Curran Associates, Inc., 2019, pp. 8024–8035, http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[35] B.H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al., The multimodal brain tumor image segmentation benchmark (brats), IEEE Trans. Med. Imaging 34 (10) (2014) 1993–2024.

[36] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J.S. Kirby, J.B. Freymann, K. Farahani, C. Davatzikos, Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features, Sci. Data 4 (1) (2017) 1–13.

[37] S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R.T. Shinohara, C. Berger, S.M. Ha, M. Rozycki, et al., Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge, arXiv preprint arXiv:1811.02629, 2018.

[38] F. Ambellan, A. Tack, M. Ehlke, S. Zachow, Automated segmentation of knee bone and cartilage combining statistical shape knowledge and convolutional neural networks: data from the osteoarthritis initiative, Med. Image Anal. 52 (2) (2019) 109–118, https://doi.org/10.1016/j.media.2018.11.009.

[39] The osteoarthritis initiative database, https://nda.nih.gov/oai/. (Accessed 29 March 2022).

[40] F. Peppert, M. von Kleist, C. Schütte, V. Sunkara, On the sufficient condition for solving the gap-filling problem using deep convolutional neural networks, IEEE Trans. Neural Netw. Learn. Syst. (2021).

[41] M. Lutz, Programming Python, O'Reilly Media, Inc., 2001.

[42] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[43] S. Dash, S.K. Shakyawar, M. Sharma, S. Kaushik, Big data in healthcare: management, analysis and future prospects, J. Big Data 6 (1) (2019) 1–25.

[44] A. Janik, K. Sankaran, A. Ortiz, Interpreting black-box semantic segmentation models in remote sensing applications, in: D. Archambault, I. Nabney, J. Peltonen (Eds.), Machine Learning Methods in Visualisation for Big Data, The Eurographics Association, 2019.

[45] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, et al., Interpretability beyond feature attribution: quantitative testing with concept activation vectors (tcav), in: International Conference on Machine Learning, PMLR, 2018, pp. 2668–2677.

[46] J. Schneider, M. Vlachos, Reflective-net: learning from explanations, arXiv preprint arXiv:2011.13986, 2020.

[47] J. Adebayo, M. Muelly, I. Liccardi, B. Kim, Debugging tests for model explanations, arXiv preprint arXiv:2011.05429, 2020.

[48] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, B. Kim, Sanity checks for saliency maps, Adv. Neural Inf. Process. Syst. 31 (2018).

[49] J.P. Agnelli, M. Cadeiras, E.G. Tabak, C.V. Turner, E. Vanden-Eijnden, Clustering and classification through normalizing flows in feature space, Multiscale Model. Simul. 8 (5) (2010) 1784–1802.