DATA VISUALIZATIONS THAT IMPROVE THE SEGMENTATION OF MEDICAL IMAGES

Yingnan Ju

Submitted to the faculty of the Graduate School in partial
fulfillment of the requirements
for the degree
Doctor of Philosophy

in the Department of
Intelligent Systems Engineering,
Indiana University

August 2024

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Doctoral Committee

Katy Börner, Ph.D., Chair

David Crandall, Ph.D.

Minje Kim, Ph.D.

Ariful Azad, Ph.D.

Date of defense — Dec 15, 2023

Acknowledgments

I am extremely grateful to my committee members for their unwavering support and guidance throughout my Ph.D. journey. I would like to extend my deepest gratitude, especially to my advisor, Dr. Katy Börner, for her excellent guidance, mentorship, and support over the past six years. Her influence has been crucial in introducing me to the world of scientific research and has greatly shaped my development as a researcher. I am also wholeheartedly thankful to all my research committee members David, Minje, and Ariful. All the professors provided me with invaluable and professional support.

I deeply appreciate the support from the Department of Intelligent Systems Engineering (ISE) and the Cyberinfrastructure for Network Science Center (CNS) at Indiana University. Additionally, the collaborations with the GE Team and NEEC Team were highlights of my doctoral studies. Their contributions greatly enriched our research, and I am grateful for their collaborative spirit. I am thankful to everyone who has collaborated with me, and particularly to Andi, Yash, Fiona, and Soumya, whose insights and contributions significantly strengthened our collective research efforts.

I am immensely thankful to my partner, Yue, whose constant support and encouragement have been the most important treasures over these years. As another Ph.D. student, her dedication to research and her contribution to my physical and mental well-being have been invaluable throughout this challenging journey for both of us. Last but not least, I am grateful to my mom and family for their unwavering trust in me and for providing me with the freedom to pursue my dreams without pressure. Their understanding and patience have been key to my success.

Yingnan Ju

DATA VISUALIZATIONS THAT IMPROVE THE SEGMENTATION OF MEDICAL IMAGES

As machine learning (ML) models become more complex, their internal logic becomes more difficult to understand and optimize. This poses significant challenges for ML training, optimization, and result interpretation, especially in medical image analysis where reliability and interpretability are critically important. Effective data visualizations can help address these challenges by transforming high-dimensional data into intuitive 2D/3D visualizations that explain complex ML models.

This dissertation examines the utility of interactive, coupled window visualizations in explaining the segmentation of medical images and enhancing the interpretability of deep learning models, such as convolutional neural networks and U-Nets. Two studies were conducted within the Human BioMolecular Atlas Program (HuBMAP): (1) Functional Tissue Unit (FTU) segmentation and (2) Vasculature Common Coordinate Framework (VCCF) computation. Interactive visualizations were designed to aid experienced users in comprehending and optimizing a U-Net model for the FTU study and to enable novice users to accurately interpret complex segmentation outcomes in the VCCF study. Formal user studies were conducted to assess the effectiveness of different visualizations in aiding ML optimization and understanding across diverse user groups. Results demonstrate that well-designed, interactive visualizations integrated into standard ML workflows can make complex models more transparent and their results easier to analyze and utilize. The visualizations successfully facilitate model refinement and provide intuitive explanations for real-world medical imaging applications.

Table of Contents

Chapter One

Introduction

This dissertation explores the potential of data visualization techniques to provide intuitive insights into complex machine learning models used for image segmentation, focusing on applications in medical image analysis projects, such as the Human BioMolecular Atlas Program (HuBMAP). For visualizations, "intuitive" refers to the presentation of data in ways that align with human natural cognitive abilities, making complex patterns and relationships immediately graspable without extensive technical explanation. Such visualizations use common data formats, interactive elements, and the links between them to enhance the interpretability of machine learning algorithms. Despite the recognized benefits of visualization for improving model interpretability, it remains an underutilized tool in standard research workflows. This dissertation aims to demonstrate the value of intuitive visualizations in advancing machine learning research, particularly in the field of medical imaging segmentation, by making the models' decisions more transparent and their results easier to analyze and trust.

In recent years, machine learning models have rapidly increased in complexity, achieving state-of-the-art results. However, these complex models often become opaque "black boxes" that obscure their internal logic. Visualization provides a promising solution by transforming high-dimensional data into intuitive visual representations that offer insights into these models, such as Convolutional Neural Networks (CNNs) and U-Nets.

A Convolutional Neural Network (CNN) is a type of artificial neural network where the mathematical operation known as convolution plays a key role. CNNs are deep learn-

ing models specifically designed for processing data that has a grid-like topology, such as images, inspired by the organizational structure of the animal visual cortex (Fukushima, 1980; Hubel et al., 1968). They are designed to automatically and adaptively learn spatial hierarchies of features, from low-level to high-level, and from simple to complex patterns (Yamashita et al., 2018). CNNs have been widely applied in areas such as image classification (Krizhevsky et al., 2012), image and video recognition (Liang et al., 2015), medical image analysis (Tajbakhsh et al., 2016), and natural language processing (Collobert et al., 2008). Although the concept of CNNs was introduced in the 1980s (Le Cun et al., 1989), they gained significant breakthroughs in the 2000s after the rapid development of powerful graphics processing units (GPUs), enabling their implementation on these GPUs.

U-Net, a specialized architecture for biomedical image segmentation, is another notable advancement in the field of deep learning models. Originating from the CNN family, U-Net is particularly designed to work efficiently with fewer training samples and still produce precise segmentations. Its distinctive architecture features a contracting path to capture context and a symmetric expanding path that enables precise localization (Ronneberger et al., 2015). This design resembles a U-shape, hence the name U-Net.

This dissertation develops and evaluates 'coupled window' visualizations for two medical image segmentation projects within HuBMAP: the Functional Tissue Unit (FTU) project and the Vasculature Common Coordinate Framework (VCCF) project. The FTU project focuses on the segmentation of functional tissue units (an important level between micro-level and macro-level in the HuBMAP project), and the VCCF project uses the body's vasculature as a reference grid to map cells. Image segmentation is crucial for both projects, and they require advanced visualization techniques to help users derive meaningful insights from the data.

Coupled window visualizations integrate various visualization types into a single view, allowing users to establish links or connections between panels and navigate through linked visualizations seamlessly. Two user studies were conducted to assess the effectiveness of these

multi-panel visualizations in enhancing the understanding and optimization of segmentation pipelines. Each study focuses on a group of users: novice users with no prior experience in machine learning, and expert users with substantial experience. The studies aimed to determine how each user group benefits from the coupled window visualizations. For expert users, the focus was on how the visualizations could facilitate the optimization of the machine learning process. In contrast, for novice users, the emphasis was on enhancing their understanding of the segmentation outcomes.

Chapters 2-4 provide a background and literature review. Chapter 2 reviews prior work on visualizing machine learning models, particularly CNNs and U-Nets, and highlights challenges with model interpretability. Chapter 3 introduces core data visualization concepts and presents an overview of the types and frameworks of data visualization that will be applied in subsequent chapters. Chapter 4 specifically focuses on the visualization of CNNs, serving as an introduction and a basis for comparison with the visualization of U-Nets in Chapters 5 and 6.

Chapters 5 and 6 present in detail the application and evaluation of coupled window visualizations for two medical imaging projects. Chapter 5 describes the Functional Tissue Unit (FTU) segmentation pipeline within the HuBMAP project, demonstrating how coupled window visualizations enabled experienced users to better understand the U-Net model and select optimal parameters to improve its performance. Chapter 6 focuses on the Vasculature Common Coordinate Framework (VCCF) developed under HuBMAP, demonstrating the effectiveness of coupled window visualizations in helping even novice users accurately interpret the complex outputs from medical image segmentation models.

This dissertation highlights the significance and potential of visualizations in understanding and optimizing complex machine learning models and techniques. By applying these visualizations, experienced users can analyze and enhance complex models such as CNNs and U-Nets, while novice users can more easily comprehend complicated machine learning results. The user study cases and evaluations included in this dissertation demonstrate the

importance of visualizations in the comprehension of machine learning, particularly within the context of medical imaging applications.

Chapter Two

Related Work

2.1 Visualizations of CNN

With the rapid progress in machine learning, deep learning, and AI, researchers, engineers, and practitioners worldwide have begun applying these technologies to various applications. While it is possible to understand human decision-making processes, it is much more difficult to learn how machine learning models work internally and how they make predictions. To understand and explain how a machine learning model works and identify the reasons for its failures, researchers have employed visualization techniques. These techniques translate machine learning models, especially their complex structures and data, into 2D or 3D visualizations that are comprehensible to humans. This chapter presents a review of papers and websites on visualizing machine learning algorithms, structures, features, and predictions.

2.1.1 Visualizations of CNN Structure

Understanding the structures of Convolutional Neural Networks (CNNs) is important for researchers in the field of deep learning, especially when these models are applied to complex tasks in image recognition and segmentation. Visualization techniques bridge the gap from the abstract, high-dimensional operations of CNNs to interpretable, two-dimensional, or three-dimensional representations. This section explores various strategies and tools developed for visualizing CNN structures, aiming to illustrate how these networks process input data, learn features, and make predictions.

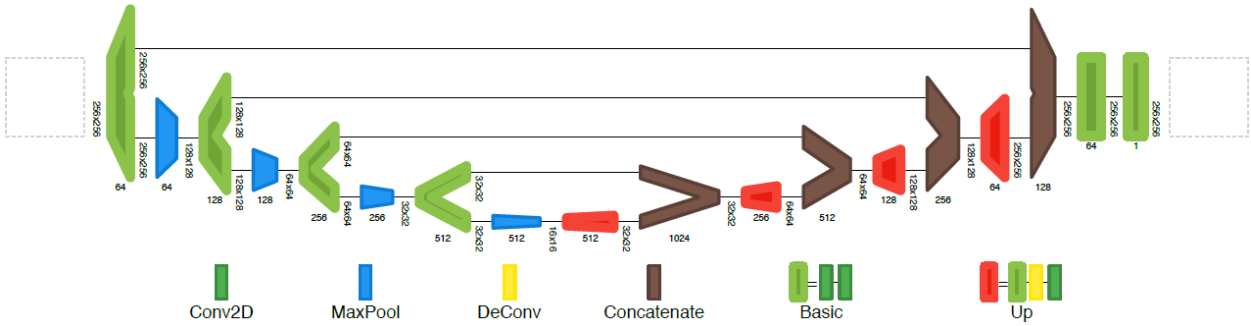


Figure 2.1 The Visualization of Encoder-Decoder CNN Architecture Generated by Net2Vis
Source: Understanding Neural Networks through Deep Visualization (Bäuerle et al., 2019)

Net2Vis: Transforming Deep Convolutional Networks into Publication-Ready Visualizations (Bäuerle et al., 2019)

When reviewing publications that visualize neural networks, the diversity in their presentation is apparent. Most visualizations are handcrafted and thus lack a unified visual grammar, which can lead to ambiguities and misinterpretations. To address this issue, Alex Bäuerle et al. developed Net2Vis, a tool that provides abstract network visualizations while still conveying detailed information about individual layers (Bäuerle et al., 2019). The number of features, as well as the spatial resolution of the tensor, are reflected, and layer types can be identified through colors in the glyph design. Additionally, it offers functionality to group layers and simplifies common layer sequences, reflecting the network's complexity. The system is designed to seamlessly integrate with models created using Keras (Chollet et al., 2015), a popular high-level programming interface for building neural networks. This compatibility ensures that visualizations of the network's structure can be automatically generated directly from the Keras code, providing an efficient and user-friendly experience. Figure 2.1 presents an example of an encoder-decoder architecture visualization created with Net2Vis. Users can export these visualizations directly from Net2Vis or customize them further for use in various publications (Bäuerle et al., 2019).

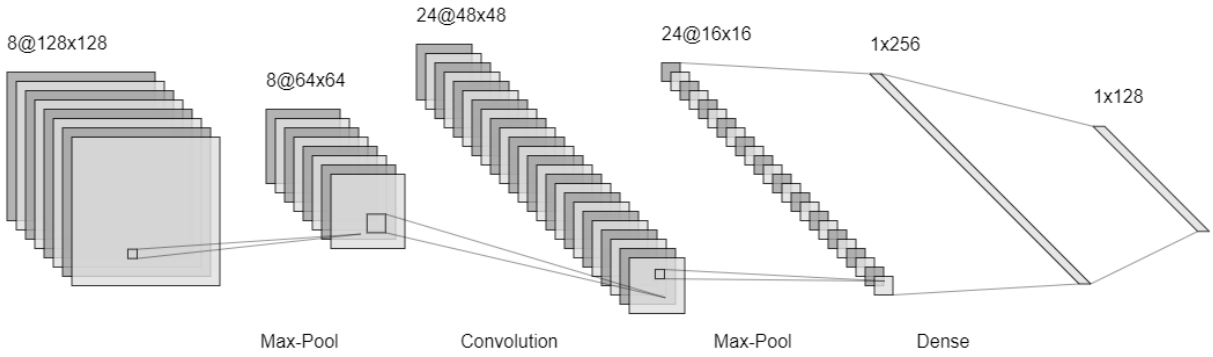


Figure 2.2 The Visualization of LeNet Generated by NN-SVG
Source: Understanding Neural Networks through Deep Visualization (LeNail, 2019)

NN-SVG (LeNail, 2019)

NN-SVG is a tool designed to automate the creation of neural network (NN) architecture diagrams. Rather than crafting these diagrams manually, users can generate figures based on specific parameters. The tool supports the generation of three distinct types of figures: traditional Fully-Connected Neural Network (FCNN) diagrams, Convolutional Neural Network (CNN) diagrams as introduced in the seminal LeNet paper (LeCun et al., 1998), and Deep Neural Network (DNN) diagrams in the style of the influential AlexNet paper (Krizhevsky et al., 2012). The FCNN and CNN diagrams are rendered using the D3 JavaScript library, while the DNN diagrams utilize the Three.js library. NN-SVG offers extensive customization options, allowing users to adjust sizes, colors, and layout parameters to suit their preferences. An example of a LeNet visualization generated by NN-SVG is depicted in Figure 2.2.

2.1.2 Visualizations of CNN Features

Understanding neural networks through deep visualization (Yosinski et al., 2015)

Jason Yosinski and his team have developed two innovative open-source tools aimed at clarifying the operations of deep neural networks (DNNs), particularly focusing on the visualization of learned features at different layers. The first tool provides a live visualization

of the activation patterns across the layers of a trained convolutional network while processing images or videos. This dynamic visualization offers a deeper understanding of the network's functionality, revealing the progression from simple to complex feature representations throughout the layers (Yosinski et al., 2015).

The second tool takes feature visualization a step further by using regularized optimization to produce more interpretable visualizations of a DNN's layers. This method not only provides clearer visual representations but also helps in understanding the contribution of various features to the network's decision-making process. The application of innovative regularization techniques has resulted in visualizations that are both qualitatively clearer and more interpretable, providing a more comprehensive understanding of the features learned by the network (Yosinski et al., 2015).

Together, these tools offer a robust framework for examining and interpreting the features learned by deep neural networks, contributing to enhanced model debugging and refinement. The interactive visualization software allows users to interactively explore the complex hierarchies of features learned by the network, as depicted in Figure 2.3. The lower part of the figure displays a screenshot of the interactive visualization interface, while the upper part shows the optimized and detailed visualizations for selected channels, highlighted in a green box, in comparison to other channels (Yosinski et al., 2015).

ConvNetJS: Deep learning in your browser (2014) (Karpathy, 2014)

ConvNetJS is a JavaScript library designed for training deep learning models, including neural networks, directly within a web browser. Unlike TensorFlow Playground (Smilkov et al., 2017), which primarily provides simplified demonstrations of classification and regression on two-dimensional data and will be discussed in more detail in the following subsection, ConvNetJS offers a range of more practical and complex demonstrations. These include tasks such as classifying digits from the MNIST (Modified National Institute of Standards and Technology) dataset (LeCun et al., 2010), categorizing CIFAR-10 (Krizhevsky et al.,

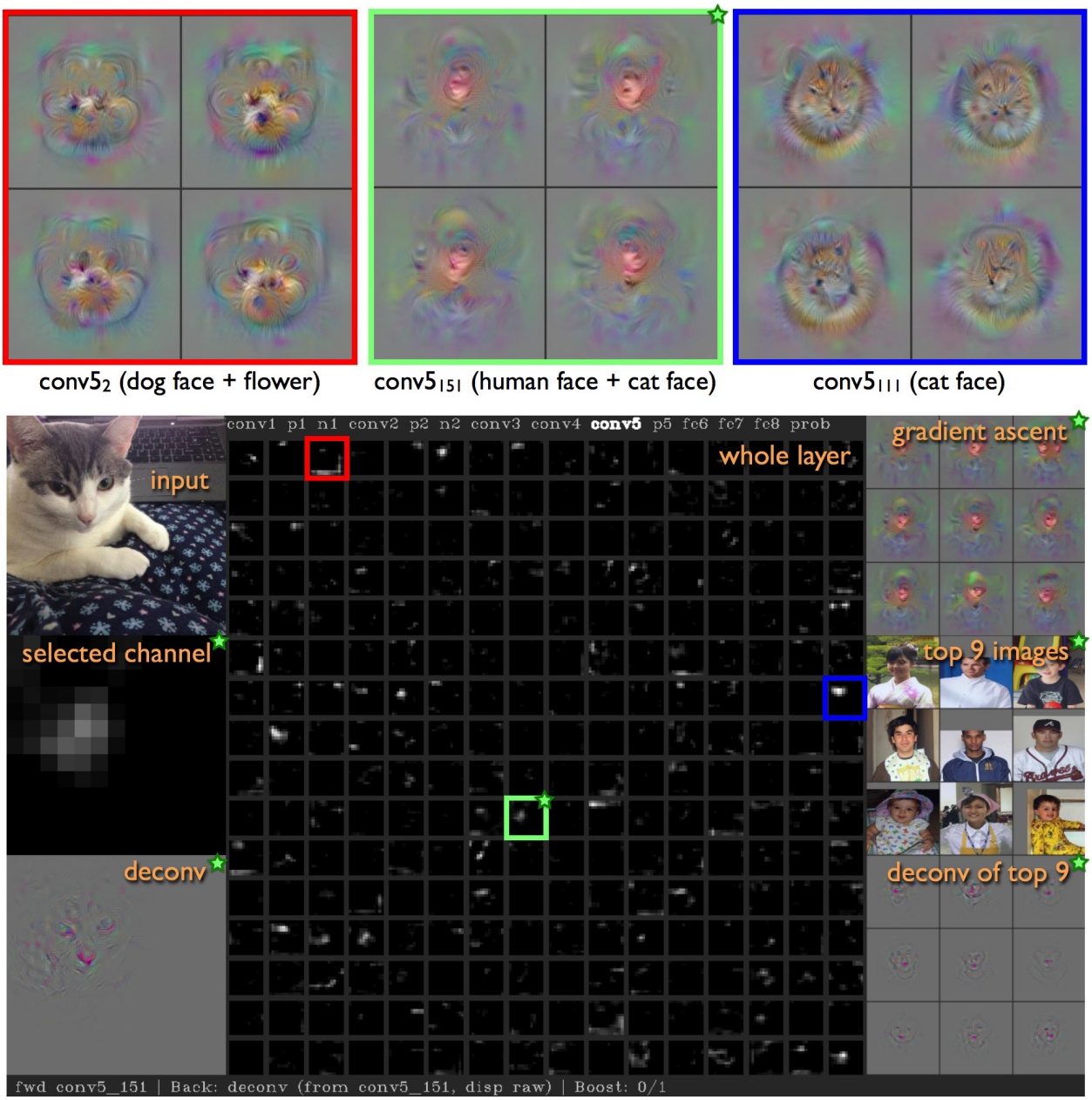


Figure 2.3 Screenshot from the Tools Introduced in "Understanding Neural Networks through Deep Visualization"
Source: Understanding Neural Networks through Deep Visualization (Yosinski et al., 2015)

2009) images using a convolutional neural network, and demonstrating how neural networks can generate images (Karpathy, 2014). Users have the flexibility to adjust various parameters, and the results are dynamically updated to reflect these changes. Figure 2.4 shows ConvNetJS in action, specifically illustrating the MNIST digit classification demo. The left side of the figure visualizes the network's architecture, and the right side presents sample predictions on a test dataset.

2.1.3 Visualizations of CNN Prediction

A Neural Network Playground for TensorFlow (Smilkov et al., 2017)

TensorFlow Playground, also known as Deep Playground, is an interactive web-based visualization tool for neural networks, developed in TypeScript with the use of d3.js. As an open-source tool, it allows users to visualize the processes, performance, and outcomes of deep learning in real-time. TensorFlow Playground primarily focuses on easy demonstrations of deep learning concepts, facilitating the classification and regression of various two-dimensional datasets (Smilkov et al., 2017). It features four distinct types of data, depicted on the left side of Figure 2.5, and allows for the adjustment of the training-to-test data ratio, level of noise, and batch size. Users can also manipulate other key parameters such as the learning rate, activation function, regularization method, and regularization rate. The architecture of the neural network is flexible, with options to change the structural parameters including the number of hidden layers and the number of neurons in each layer. This tool provides a dynamic and visual representation of the deep learning process, enhancing comprehension of how each parameter influences the model's performance and the final outcomes. Figure 2.5 captures a screenshot of TensorFlow Playground at its initial state.

GAN lab: Understanding complex deep generative models using interactive visual experimentation (Kahng et al., 2018)

Beside TensorFlow Playground, there are additional visualization tools like GANLab that

Figure 2.4 Screenshot of ConvNetJS MNIST Demo
https://cs.stanford.edu/people/karpathy/convnetjs/

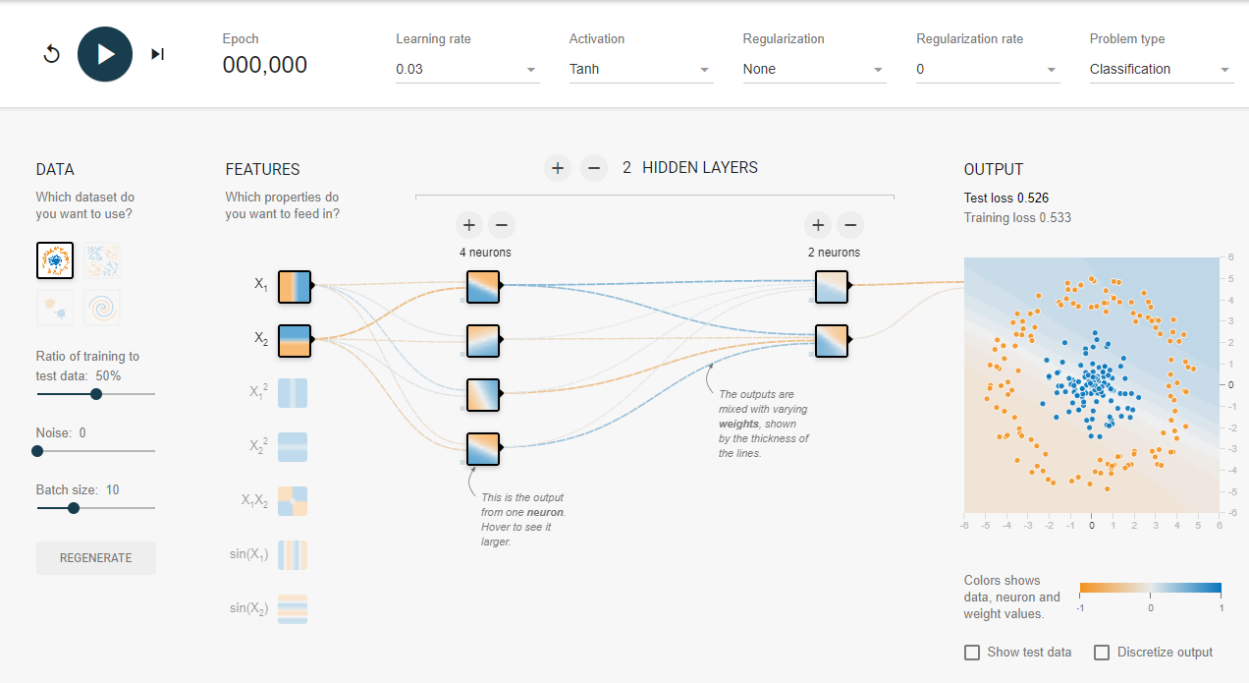


Figure 2.5 Screenshot of "TensorFlow Playground"
Source: http://playground.tensorflow.org/

provide insights into neural network training concepts (Kahng et al., 2018). GANLab stands out for its interactive visualization, which offers a user-friendly interface to demonstrate the learning progression of Generative Adversarial Networks (GANs) (Goodfellow et al., 2020). Instead of generating realistic images or photos, GANLab simplifies the process by focusing on a GAN that learns to model a distribution of points within a two-dimensional space. Although real-world GAN applications are far more complex, GANLab effectively demonstrates the fundamental mechanics of GANs. According to the associated paper, the visualization consists of two main parts: the model overview graph, which outlines the GAN architecture and the outputs of its components, and the layered distribution view, which overlays component visualizations on the model overview graph, allowing for a more straightforward comparison of outputs during model analysis (Kahng et al., 2018). Figure 2.6 captures a screenshot of GANLab at its initial setup.

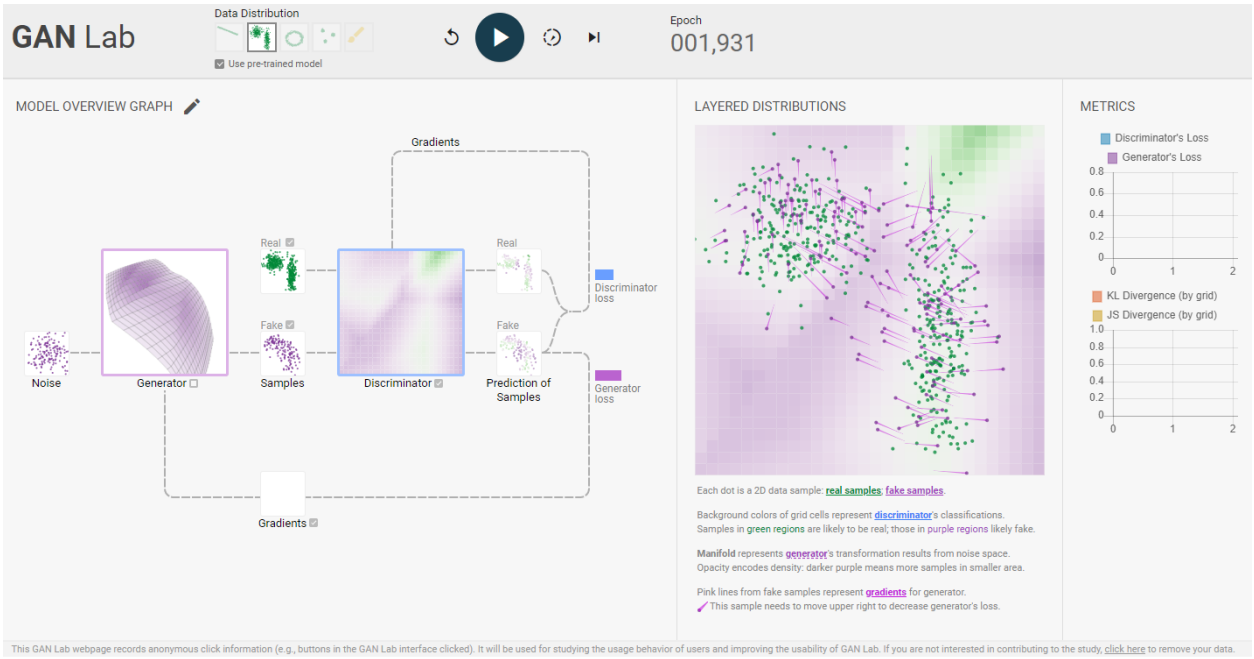


Figure 2.6 Screenshot of GAN Lab
Source: https://poloclub.github.io/GANLab/

Gradient-weighted Class Activation Mapping (Grad-CAM)

Gradient-weighted Class Activation Mapping (Grad-CAM) uses the class-specific gradient information flowing into the final convolutional layer of a CNN to produce a coarse localization map, highlighting the important regions in the image for predicting a given class (Selvaraju et al., 2017). The purpose of Grad-CAM is to increase the transparency of Convolutional Neural Network (CNN)-based models. The method visualizes the regions of the input that are "important" for predictions from these models - also known as visual explanations. Both Grad-CAM and Guided Grad-CAM can provide these visual explanations, which help in better understanding image classification, image captioning, and visual question answering (VQA) models (Antol et al., 2015). One major advantage of Grad-CAM is that it is applicable to a wide variety of widely used CNN model families in real-world applications, including (1) CNNs with fully connected layers (e.g., VGG), (2) CNNs used for structured outputs (e.g., captioning), and (3) CNNs used in tasks with multimodal inputs (e.g., VQA) or reinforcement learning, without requiring any architectural changes or re-training (Sel-

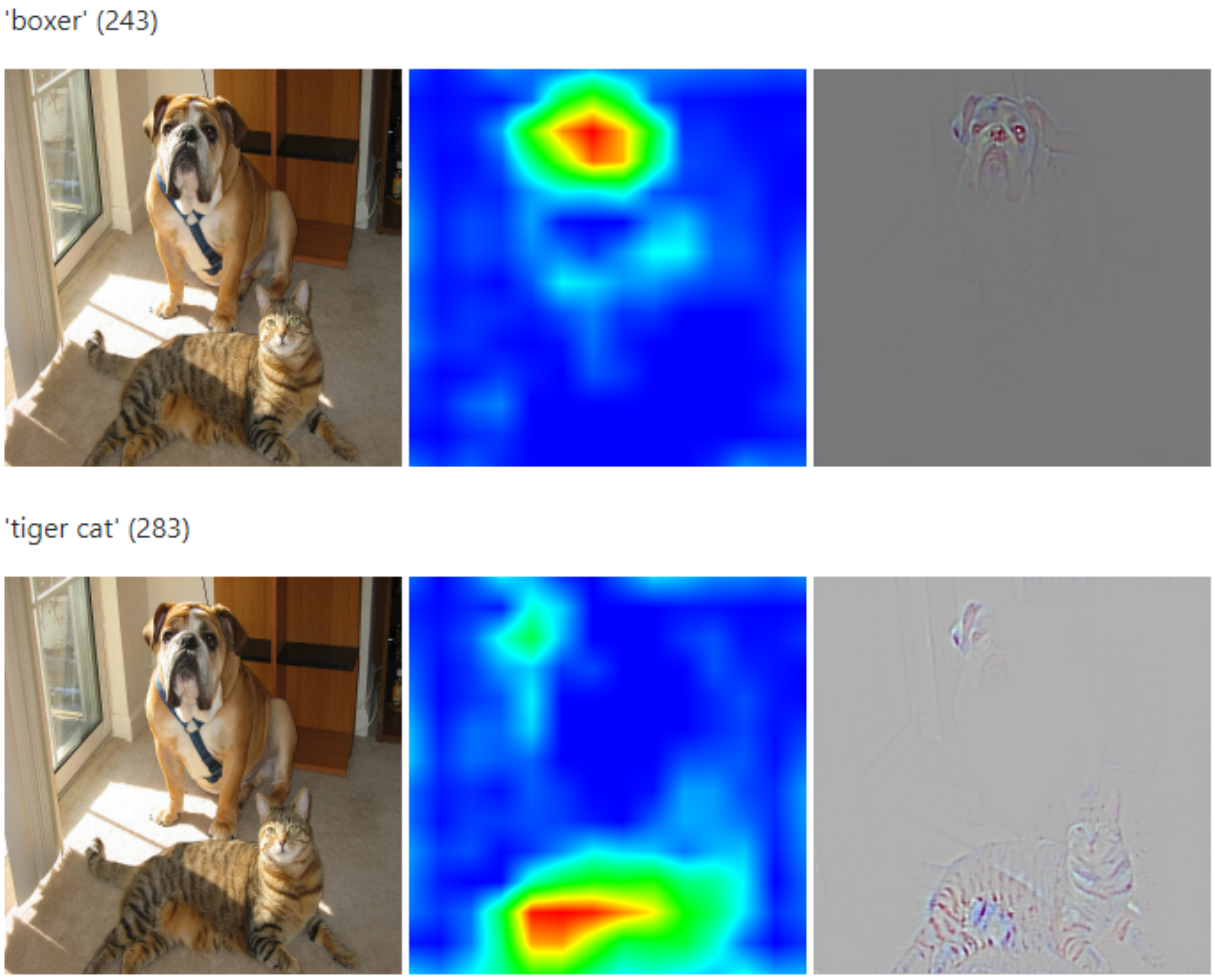


Figure 2.7 An Example of the Input and Output of Grad-CAM: The Grad-CAM and Guided Grad-CAM Images for Dog and Cat Classification

varaju et al., 2017). Figure 2.7 shows an example output of Grad-CAM: the Grad-CAM and Guided Grad-CAM images for dog and cat classification.

Deepmind Starcraft II

Games have long been used as a method to test and evaluate the capabilities of artificial intelligence systems (Vinyals et al., 2019b). AlphaStar, an AI gaming program developed by DeepMind (Vinyals et al., 2019a), has achieved a significant milestone by defeating a top human professional player in the video game StarCraft II.

Figure 2.8 visualizes AlphaStar's perspective during one game against a human opponent.

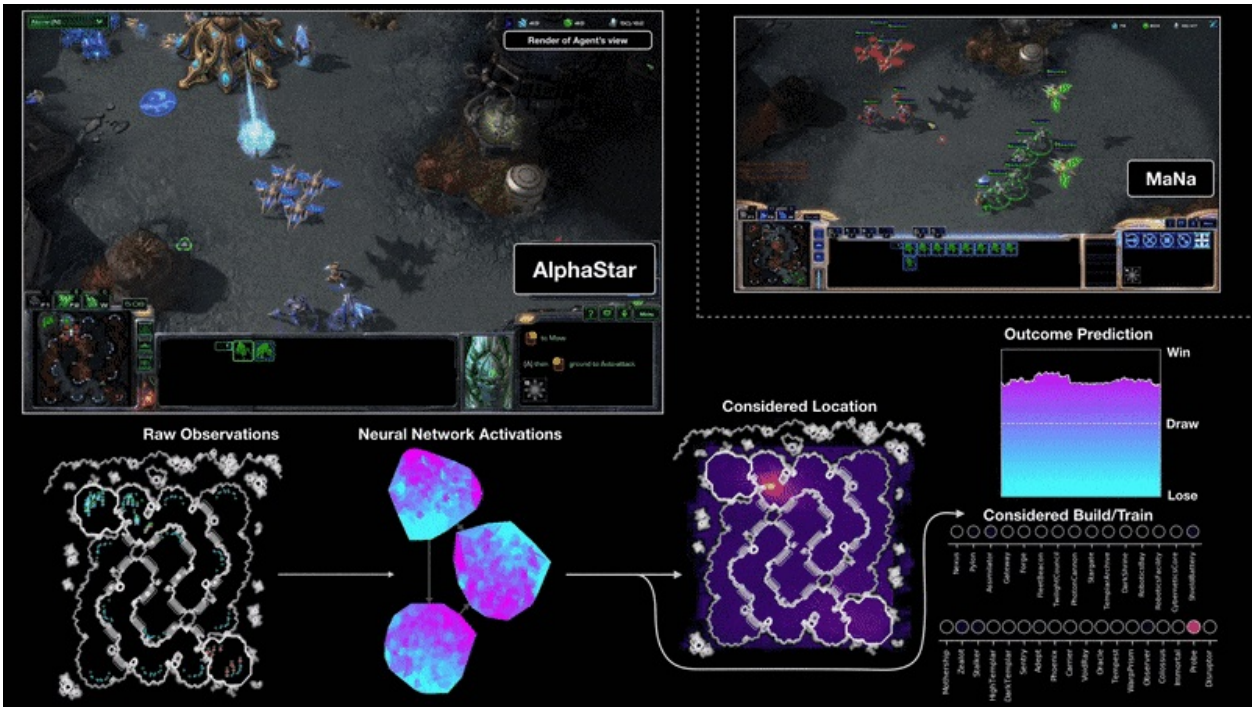


Figure 2.8 Visualization of the Alphastar Program

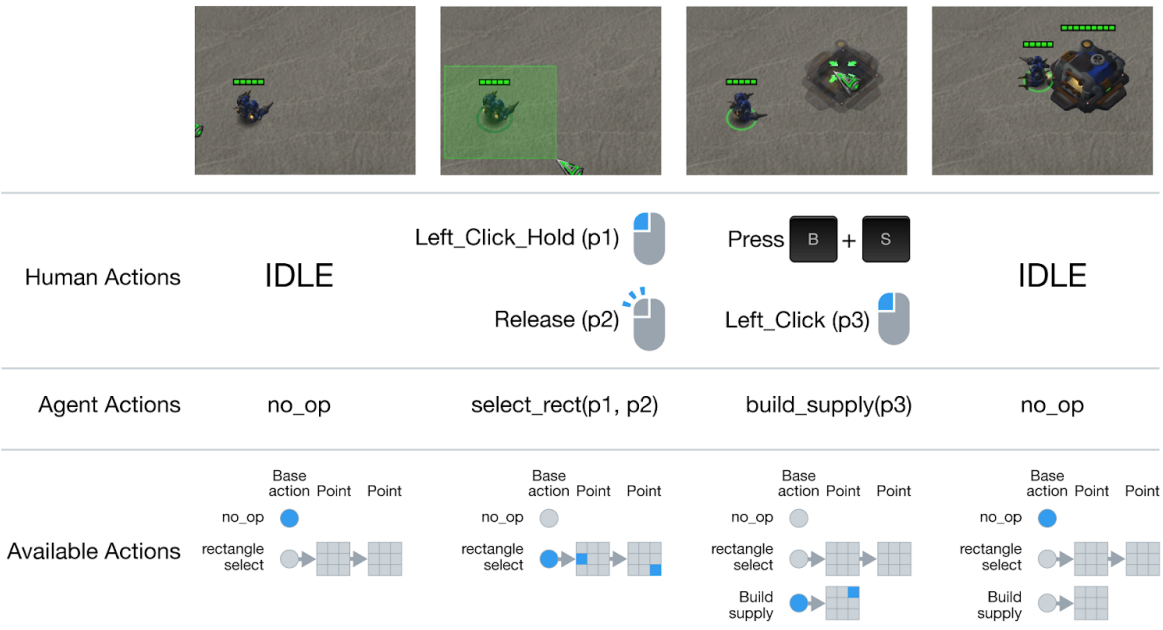


Figure 2.9 Meta-Operation of the Alphastar Program

It shows the raw observation input fed into the neural network, the neural network's internal activations, some of the actions considered by the program, such as where to click and what to build, and the predicted outcome (Vinyals et al., 2019b). Figure 2.9 compares the meta operations between humans and computers, highlighting how visualizations of machine learning can demonstrate differences in perception and decision-making processes between humans and machines.

2.1.4 Visualizations of CNN Evaluation

MLxtend: Providing machine learning and data science utilities and extensions to Pythons scientific computing stack (Raschka, 2018)

MLxtend, created by Sebastian Raschka, is an open-source library that provides machine learning and data science utilities and extensions to Python's scientific computing stack (Raschka, 2018). In addition to offering machine learning algorithms and model evaluation techniques for performance comparison, MLxtend includes visualization functions that allow users to examine the predicted performance across different feature subsets. Figure 2.10 shows decision regions plotted for various classifiers, with each region uniquely colored according to the algorithm's results. This helps users understand the differences between various machine learning methods in terms of performance, processes, and outputs.

Weka: A machine learning workbench (Holmes et al., 1994)

Weka (Waikato Environment for Knowledge Analysis) is a comprehensive workbench that provides a collection of machine learning algorithms and data preprocessing tools (Frank et al., 2009; Hall et al., 2009). Aimed at simplifying the application of machine learning to real-world problems, Weka provides a graphical user interface for data exploration, experiment setup, and data stream processing. It is widely used in both academic and business settings and has a robust and active user community. Although Weka was first implemented over 20

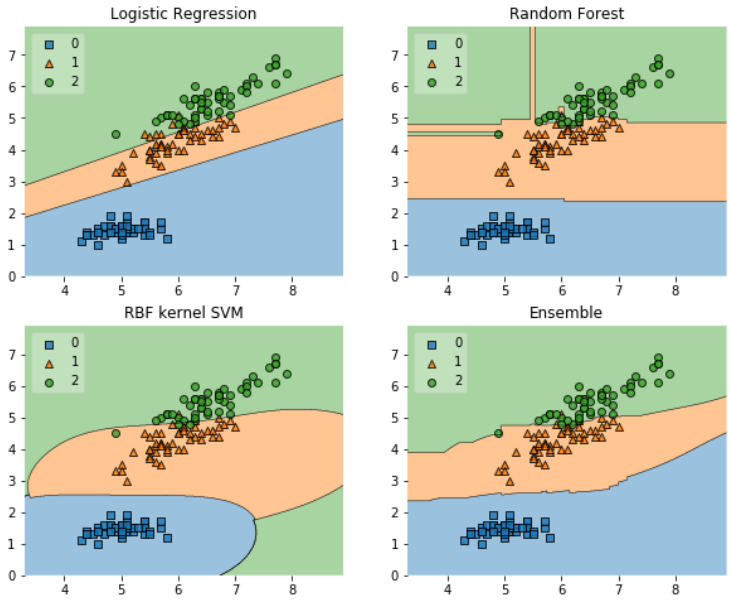


Figure 2.10 Screenshot of MLxtend: The Comparison of the Plotting of Decision Regions between Different Classifiers
Source: http://rasbt.github.io/mlxtend/

years ago, the third paper (Hall et al., 2009) introduced its latest major update, with the software completely rewritten from scratch.

EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers (Talbot et al., 2009)

In this paper, authors from Stanford University and Microsoft Research present an interactive visualization system called EnsembleMatrix, which provides a graphical view of confusion matrices to help users understand the relative merits of various machine learning classifiers (Chang et al., 2011; Friedman et al., 1997). This system helps users understand various machine learning classifiers and build ensemble models interactively. EnsembleMatrix has proven to be effective for important multiclass classification problems, providing a user-friendly interface to address these challenges. Figure 2.12 shows EnsembleMatrix's main interface, with thumbnails of component classifiers' confusion matrices on the right and the ensemble classifier's confusion matrix on the left, customizable via the graphical interface

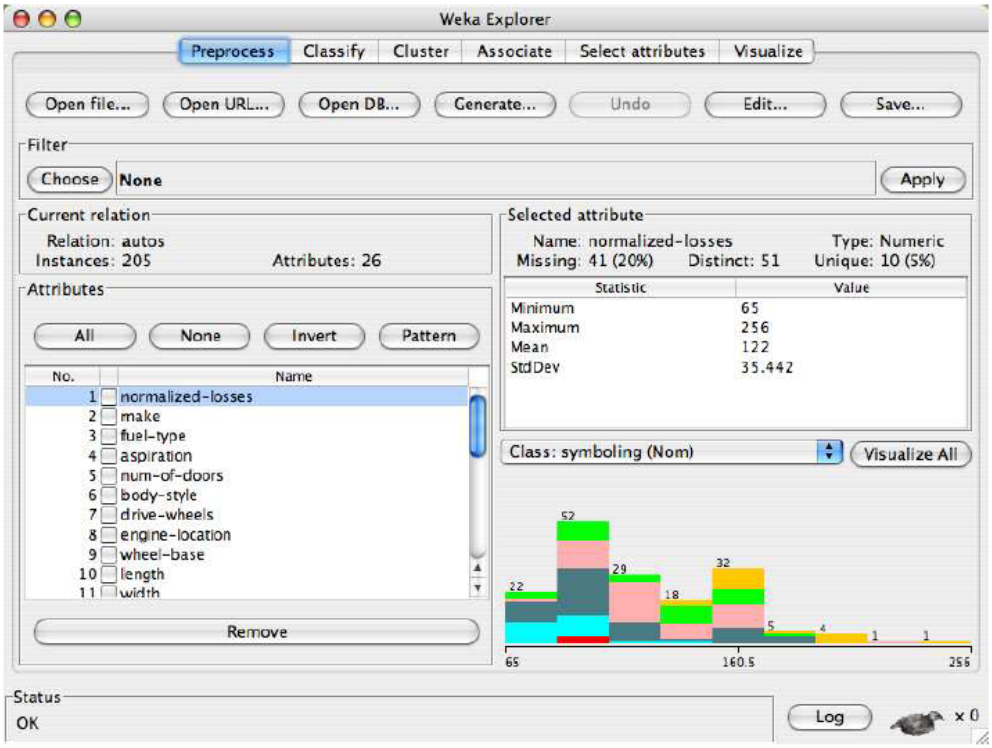


Figure 2.11 Screenshot of Weka Explorer
Source: The WEKA Data Mining Software: An Update (Hall et al., 2009)

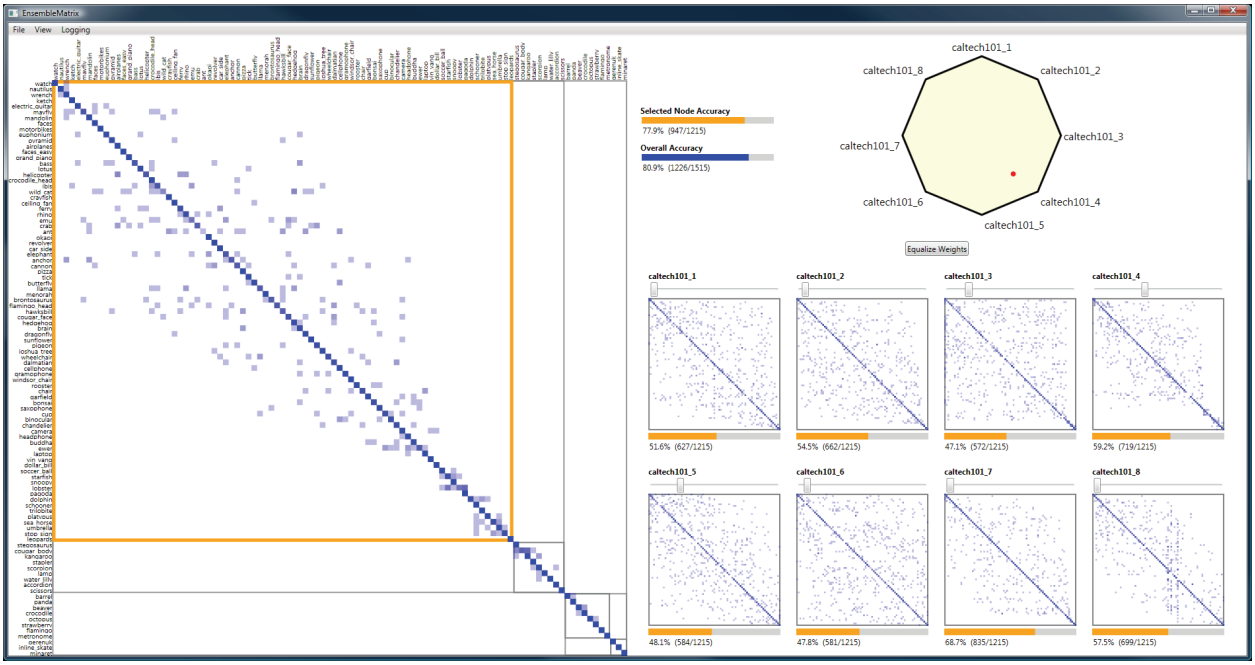


Figure 2.12 Primary View in EnsembleMatrix
Source: EnsembleMatrix: Interactive Visualization to Support Machine Learning with Multiple Classifiers (Talbot et al., 2009)

(Talbot et al., 2009).

2.1.5 Comparison between the Existing Works

Table 2.1 compares the frequency of graphic symbols used in existing works. The graphic symbols and variables are defined in (Börner et al., 2019). The symbols include point, line, area, surface, volume, text, numerals, images, and icons. The variables include position, form, and color. From the table, it is evident that "position" and "color" are utilized in all visualizations. However, the same variable can represent different functions across graphs. For example, in the "Grad-CAM" (R. R. Selvaraju et al., 2016) visualization, "color" depicts the highlighted areas of highest or lowest importance for CNN predictions, but in TensorFlow Playground (Smilkov et al., 2017) and GANLab (Kahng et al., 2018) visualizations, "color" separates different dataset parts and clusters. After "position" and "color", "text" and "area" are the most frequently used symbols. The visualization from DeepMind's StarCraft II com-

bines the highest number of symbols and variables, showcasing diverse types of visualization (Vinyals et al., 2019b).

Related Work	*Point*	*Line*	*Area*	*Surface*	*Volume*	*Text*	*Numerals*	*Images*	*Icons*	*Position*	*Form*	*Color*	**Total**
Net2Vis		X	X			X	X			X		X	*6*
NN-SVG-FCNN	X	X								X		X	*4*
NN-SVG-LeNet		X	X			X	X			X		X	*6*
NN-SVG-AlexNet		X	X		X	X	X			X		X	*7*
*Understading **			X			X		X		X		X	*5*
ConvNetJs-MNIST			X			X	X	X		X		X	*6*
TensorFlow Playground	X	X	X	X		X			X	X		X	*8*
GANLAB	X	X	X	X		X			X	X		X	*8*
Grad-Cam				X				X		X		X	*4*
Deepmind StarCraft II	X	X	X	X		X	X	X	X	X	X	X	*11*
Mlxtend	X		X			X	X			X		X	*6*
Weka			X				X			X		X	*4*
EnsembleMatrix	X		X			X				X		X	*5*

* "Understading" = "Understanding neural networks through deep visualization (Yosinski et al., 2015)"

Table 2.1 Comparison of Frequency of Graphic Symbols and Variables

2.2 Machine Learning and Visualizations in HuBMAP

2.2.1 HuBMAP Project

The goal of the Human BioMolecular Atlas Program (HuBMAP) is to develop an open and global platform to create a comprehensive atlas of human biology at the cellular level (Snyder et al., 2019). HuBMAP researchers utilize the latest molecular and cellular biology technologies to study cellular connections throughout the body (Börner et al., 2021).

HuBMAP dataset and FTU segmentation

The focus of HuBMAP is understanding the intrinsic intra-, inter-, and extra- cellular biomolecular distributions in human tissues. HuBMAP will use fresh, fixed, or frozen healthy human tissue using in situ and dissociative techniques that have high spatial resolution.

HuBMAP's high-resolution, heterogeneous data requires advanced preprocessing and augmentation (Jeddi et al., 2020). It also supports nuanced models that capture tissue interactions and clinically relevant segmentation (Zhou et al., 2018). HuBMAP's diverse tissue datasets have enabled robust segmentation models like U-Net variants (Rivenson et al., 2019), allowing researchers to precisely identify cellular structures and provide deeper insights into human biology (H. Wang et al., 2019). Functional tissue units are the fundamental building blocks of human anatomy, and their segmentation can elucidate tissue microarchitecture and interactions between cells and the microenvironment (Cao et al., 2019).

Vasculature Common Coordinate Framework (VCCF)

HuBMAP is developing a Common Coordinate Framework (CCF) for the healthy human body, which will support the cataloging of individual cells, understanding of functions and relationships between cell types, and modeling of their individual and collective functions (Ghose et al., 2022; G. M. Weber et al., 2020). The Vasculature Common Coordinate Framework (VCCF) standardizes vascular structure analysis, advancing medical image segmentation and visualization (Ghose et al., 2022).

2.2.2 Medical Image Segmentation

The primary goal of semantic segmentation is to categorize various image regions into distinct, meaningful groups accurately (Moorthy et al., 2022). In computer-aided diagnosis, segmentation is important for extracting anatomical structures (Ritter et al., 2011). It involves isolating regions of interest (ROIs) from 3D MRI or CT data with precision (Milletari et al., 2016), facilitating diagnosis, surgery planning, and other medical applications (Litjens et al., 2017). Recent advances in segmentation techniques have significantly improved accuracy and performance in medical image analysis (Moorthy et al., 2022).

To achieve effective segmentation, end-to-end architectures like U-Net have been proposed (Ronneberger et al., 2015). U-Net addresses the limitations of CNNs through its symmetric

design and skip connections (Çiçek et al., 2016). Unlike natural images, medical images often exhibit noise and blurred boundaries, challenging segmentation that relies solely on low-level features (Ronneberger et al., 2015). Details can also be lost when relying only on high-level semantics (Ronneberger et al., 2015). U-Net integrates both through skip connections between low and high-resolution features. This effectiveness of integration has made it a benchmark in medical segmentation, inspiring many improvements and enhancements (Moorthy et al., 2022).

Variants of U-Net in Medical Image Segmentation

The original U-Net established a benchmark in biomedical image segmentation. However, the evolving complexities of medical image segmentation have led to the development of several U-Net variants.

U-Net++ (Zhou et al., 2018) creates nested, dense skip connections, unlike U-Net's singular links. These multi-resolution feature captures benefit the segmentation of intricate details such as vessels or tumor boundaries. Attention U-Net (Oktay et al., 2018) integrates attention gates, weighing features by relevance. This selective focus is particularly useful for tasks like lesion detection, where certain regions are more critical than others. Another notable variant worth mentioning is the Residual U-Net (Alom et al., 2018, 2019), which incorporates residual blocks, inspired by the ResNet architecture (He et al., 2016), into the U-Net design. The vanishing gradient problem occurs during the training of deep networks when the gradients of the loss function decrease exponentially as they are backpropagated through the network, leading to insufficient weight updates in the initial or earlier layers (Pascanu et al., 2013). By incorporating residual blocks, the Residual U-Net provides shortcuts for the gradients to flow through, which helps prevent the gradients from diminishing rapidly and ensures that all layers of the network learn effectively (He et al., 2016). This alleviates the vanishing gradient problem, enabling the training of deeper networks, which is particularly beneficial for segmenting more complex medical images where deeper archi-

tectures are necessary to capture finer details.

In comparison, while the original U-Net provided a robust foundation for medical image segmentation, its variants, with their modifications, address specific challenges more effectively. The choice between them often depends on the particular requirements of the medical image segmentation task at hand.

U-Net and CNNs: Distinctions and Similarities in Image Segmentation

Both Convolutional Neural Networks (CNNs) and U-Net have gained attention in image processing. While they share foundational elements, their design philosophies and applications, especially in the context of medical image segmentation, exhibit significant and notable differences.

CNNs, primarily developed for image classification, consist of layers that systematically capture hierarchical features by reducing spatial dimensions. This architecture comprises a series of convolutional and pooling layers followed by fully connected layers that make final decisions based on the extracted features (Krizhevsky et al., 2012; LeCun et al., 2015; Simonyan et al., 2014). CNNs are highly effective at pattern recognition but are not inherently suited for pixel-level segmentation (Ronneberger et al., 2015).

In contrast, U-Net is optimized for biomedical image segmentation (Çiçek et al., 2016). Its architecture is characterized by a contracting path, capturing context, and a symmetric expanding path for precise localization (Milletari et al., 2016). This design allows U-Net to produce high-resolution segmentation maps, a critical requirement in medical imaging where detailed segmentations are essential (Gibson et al., 2018), such as tumors or blood vessels.

While both architectures use convolutional layers, their objectives are different. CNNs reduce spatial dimensions for classification purposes, whereas U-Net restores these dimensions to produce segmentation maps that match the input size (Long et al., 2015). Furthermore, recent advancements include hybrid models that combine the strengths of both CNNs and U-Net to enhance medical image segmentation (Drozdzal et al., 2016).

Visualizations of U-Net in Medical Image Segmentation

Visualization plays an essential role in interpreting and validating U-Net models in medical imaging, where model transparency is crucial. It is imperative to ensure that models like U-Net are not only accurate but also interpretable.

One of the primary methods is the visualization of intermediate feature maps within the U-Net architecture (Matthew D Zeiler et al., 2014). By examining these maps, researchers can identify what features the network emphasizes at different layers. Earlier layers might focus on textures while deeper layers capture complex structures, such as tumors, based on medical images (Yosinski et al., 2015). Since 2018, attention mechanisms have been integrated into U-Net architectures for medical image segmentation (Oktay et al., 2018). These mechanisms allow the model to concentrate on specific regions of the input image, which is particularly beneficial in medical scenarios where certain areas, like tumors, are important. Visualizing these attention maps can clarify which areas the model is focusing on. Additionally, there are also techniques like Grad-CAM (Selvaraju et al., 2017) that have been adapted for U-Net to show which parts of the input image contribute most significantly to the model's decision. For medical images, this can highlight areas of concern and ensure reliance on relevant features. Techniques such as t-SNE (Van der Maaten et al., 2008) are also used to visualize the embeddings produced by U-Net layers. It can project high-dimensional data into 2D or 3D spaces for researchers to observe the clustering of similar features.

As U-Net's role in medical image segmentation continues to grow and expand, the need for effective visualization techniques becomes increasingly important, providing essential into the model's operations and enhancing interpretability.

Chapter Three

Introduction to Data Visualization

In the information age, the ability to read and create visualizations of data is as important as the ability to read and write text (Börner et al., 2019). The Data Visualization Literacy Framework (DVL-FW) by Börner et al., 2019 categorizes common visualization techniques into insightful types, including tables, charts, graphs, maps, trees, and networks. These techniques help cluster, order, show distributions, make comparisons, and identify trends and relationships. Understanding these core concepts and procedures of the DVL-FW enables us to systematically construct optimal visualizations tailored to specific goals and audiences, which is essential for analyzing complex models like CNNs and U-Nets in later chapters.

Humans can generally infer meaning from pictures more quickly than from text (Williams, 2015). The practice of using images to interpret information has existed for centuries; as long as people have recorded findings and written papers, diagrams, illustrations, charts, and tables have accompanied scientific works (Gore, 2018). Data visualization refers to the methods used to convey statistical data or information by representing it as graphical objects (e.g., dots, lines, or areas) in graphics (Few, 2004) to describe relationships, comparisons, compositions, and distributions. As both an art and a science (Aparicio et al., 2015), effective data visualization should not only communicate clearly but also enhance the viewer's interest and attention (Viegas et al., 2011). Every professional industry field, including all STEM (Science, Technology, Engineering, and Math) disciplines, can benefit from more comprehensible data representations.

The DVL-FW further defines core concepts of data visualization, such as insight needs,

data scales, analyses, graphic symbols, graphic variables, and interactions (Börner et al., 2019). Understanding these elements helps us systematically create optimal visualizations matched to specific goals and audiences. For instance, CNN and U-Net models produce quantitative data with meaningful numeric differences. This leads to the selection of appropriate graphic symbols like points, lines, or shaded areas instead of qualitative symbols like shapes.

The process of applying the DVL framework is critical as well. It involves iterative steps of acquiring and analyzing relevant data, followed by visualization through carefully chosen encodings. This method effectively communicates findings and insights derived from CNNs and U-Nets. The DVL framework distinctively integrates this conceptual knowledge with practical steps for implementing data visualization.

3.1 Types of Visualizations

When considering data visualization, bar graphs or pie charts might come to mind first. However, these are only the tip of the iceberg of data visualization. Visualizations can be categorized into different types based on various scales or criteria, and many researchers have suggested various visualization taxonomies and frameworks over the past five decades (Börner, 2015). Comprehensive and effective data visualizations are designed using one or more data visualization frameworks (DVL-FWs) by Börner et al., 2019. The specific visualization types are discussed subsequently.

3.1.1 Tables

Tables are a simple yet highly effective way to convey data and are widely used in communication, research, and data analysis across various media, including print, handwritten notes, computer software, architectural ornamentation, traffic signs, and more (McNabb, 2015; Zieliski et al., 2005). A table organizes data into rows and columns. As a comprehen-

sive visualization type, tables can present all data points, graphics, and icons in a structured format that is ideal for reporting. Cells in a table may contain proportional symbols or small charts/graphs (Börner, 2015). Figure 3.1 from (Börner et al., 2019) shows various graphic symbols and variables used in a table.

In a table, columns are typically labeled with a title, phrase, or index number, and each row usually represents one data record. Frequently, the first row displays the column names and is referred to as the "header" or "header row". Similarly, the first column often acts as a summary of the index or information element navigating the rest of the table, known as the "header column". The intersection of a row and column is called a cell.

3.1.2 Charts

The term "chart" refers to visualizations without an inherent reference system (Börner et al., 2014). Charts are used to represent tabular numeric data, functions, qualitative structures, and various quantitative information (Yates, 2010). They are supported by many spreadsheet programs and are commonly used in information graphics. Charts often facilitate the interpretation of large data datasets and relationships between data pieces or points (Slutsky, 2014).

There is a wide variety of chart forms in the field of data visualization. The most common and well-known types include the pie chart, the bubble chart, and their variants:

- **Pie Chart**: A pie chart displays data as proportional disks, with percentage values depicted as slices of the pie. A variant of the pie chart is the doughnut chart. The sequence and sizes of the slices are arbitrary; the angles and areas of the slices represent percentages of the total (i.e., all slices should sum to a meaningful total) (Börner, 2015). Figure 3.2 exemplifies a pie chart and a doughnut chart showing values across three years.

- **Bubble Chart**: A bubble chart represents each data record as a randomly positioned

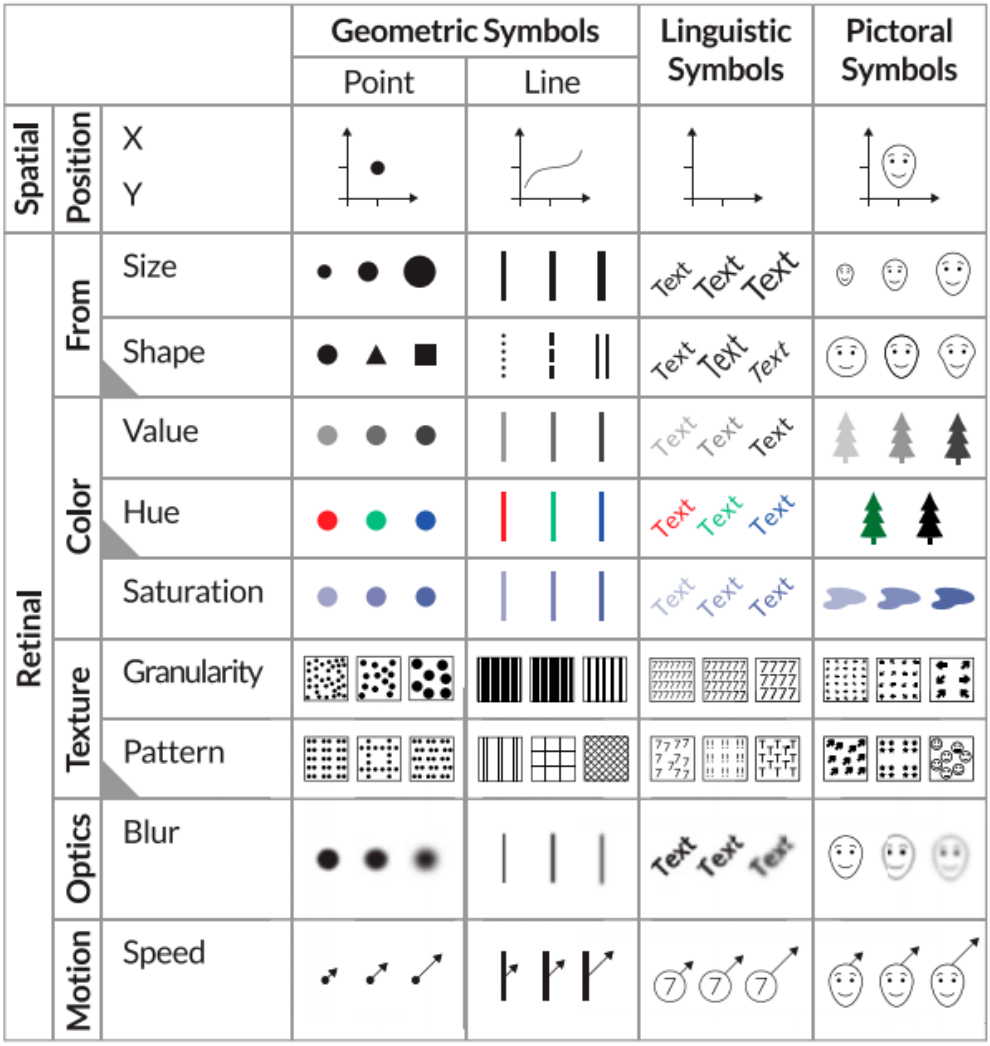


Figure 3.1 Visualization Type: Example of Tables - Graphic Variable Types Versus Graphic Symbol Types
Source: Data Visualization Literacy: Definitions, Conceptual Frameworks, Exercises, and Assessments (Börner et al., 2019)

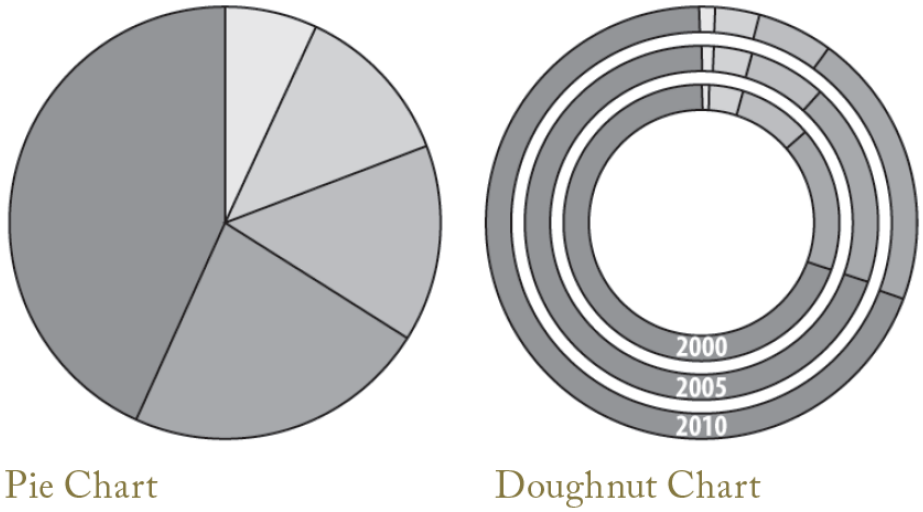


Figure 3.2 Visualization Type: Example of Pie Chart & Doughnut Chart
Source: Atlas of Knowledge: Anyone Can Map (Börner, 2015)

geometric object (Börner, 2015). To optimize the use of space or establish a clear pattern, larger items may be positioned closer to the center of the chart (Börner, 2015). Figure 3.3 exemplifies a bubble chart that provides a perspective on world population.

A word cloud, also known as a tag cloud, is a variant of a bubble chart where words replace geometric objects. Figure 3.4 shows a word cloud of movie titles from the Internet Movie Database (IMDb) (Börner et al., 2014). Words that appear more frequently are larger, with the largest usually in the center (Börner et al., 2014).

3.1.3 Graphs

Graphs are the most commonly used data visualization types, plotting quantitative and/or qualitative variables. Compared to charts, graphs share many similar visualization elements. The key difference is that graphs plot data using a well-defined reference system, whereas charts depict data without a reference system (Börner, 2015). There are numerous types of graphs, including line graphs, bar graphs, stacked line and bar graphs, and scatter plots.

- **Scatter plot**: A scatter plot, also known as an x-y plot or dot, point, or symbol graph,

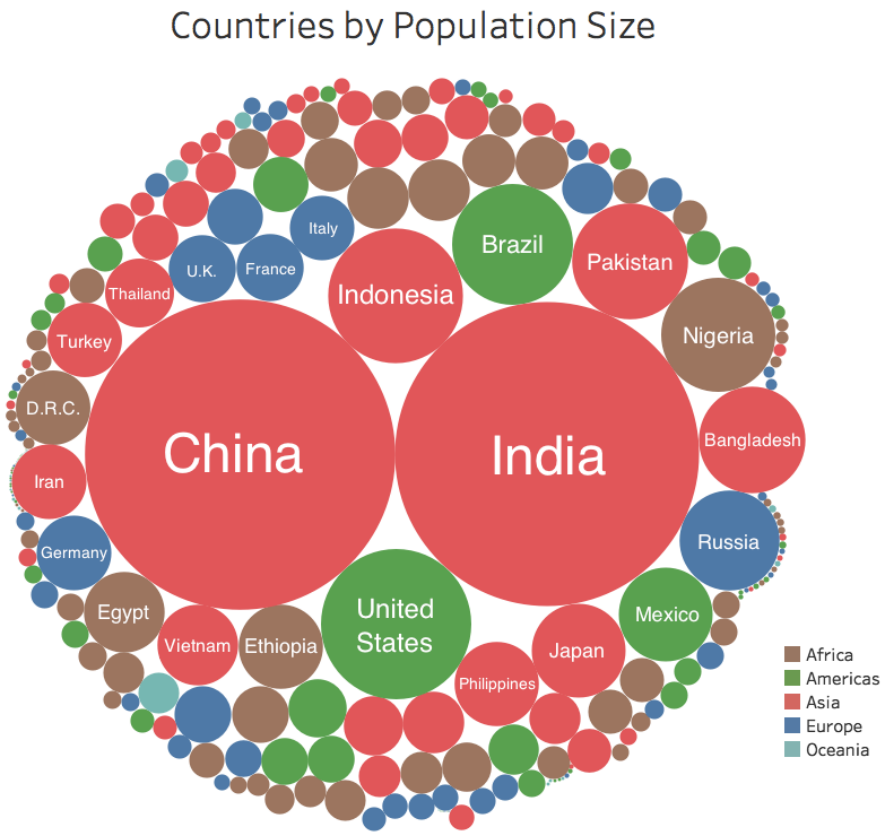


Figure 3.3 Visualization Type: Example of Bubble Chart - Countries by Population Size (2017)
Source: https://www.visualcapitalist.com/population-every-country-bubble/, Title: The Population of Every Country is Represented on this Bubble Chart, Author/Copyright holder: Jeff Desjardins

Figure 3.4 Visualization Type: Example of Word Cloud - Word Cloud of Movie Titles from the Internet Movie Database (IMDb) Created Using Wordle
Source: Visual Insights (Börner et al., 2014)

displays quantitative data records as graphical symbols (e.g., points). Scatter plots are used to investigate correlations between variables or multiple datasets. Typically, the x-axis plots the independent variable, while the y-axis depicts the dependent variable. In other words, the x-axis represents measured entities like time periods or objects, and the y-axis represents the measurement values. Figure 3.5 is an example of a scatter plot showing the correlation between infant mortality rate and total fertility rate among 194 nations in 1997. It indicates that countries with higher infant mortality rates also tend to have higher total fertility rates.

- **Line graph**: A line graph is a two-dimensional scatter plot of ordered observations with points connected in order. Line graphs depict quantitative data as a series of points connected by lines, often to describe changes or trends over time. Figure 3.6 is an example of a line graph showing quantitative trends for n-grams (contiguous sequences of n items from a text sample) appearing in over five million books from 1800 to 2000 (Börner, 2015).

- **Bar graph**: A bar graph visualizes data with rectangular bars proportional in length to the values they represent. Bar graphs allow for graphical comparisons of discrete

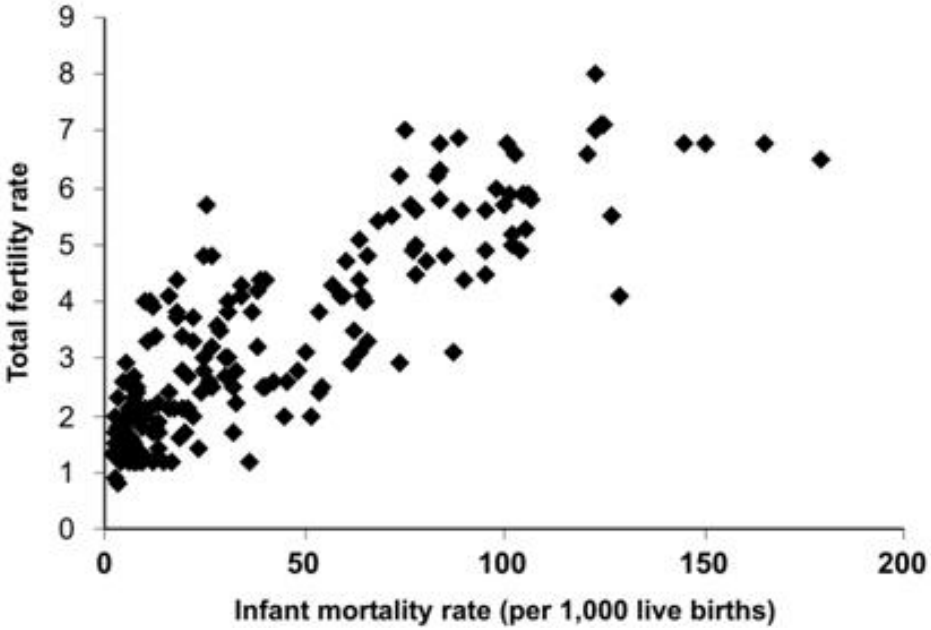


Figure 3.5 Visualization Type: Example of Scatter Plot - Correlation of Infant Mortality Rate and Total Fertility Rate Among 194 Nations, 1997
Source: Population Reference Bureau (PRB, 2004)

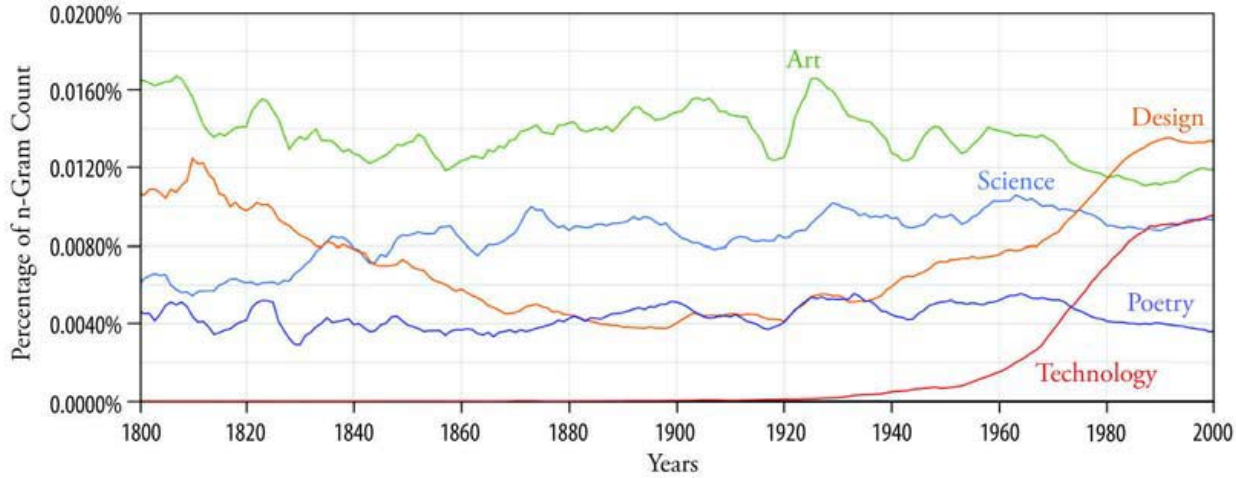


Figure 3.6 Visualization Type: Example of Line Graph
Source: Atlas of Knowledge: Anyone Can Map (Börner, 2015)

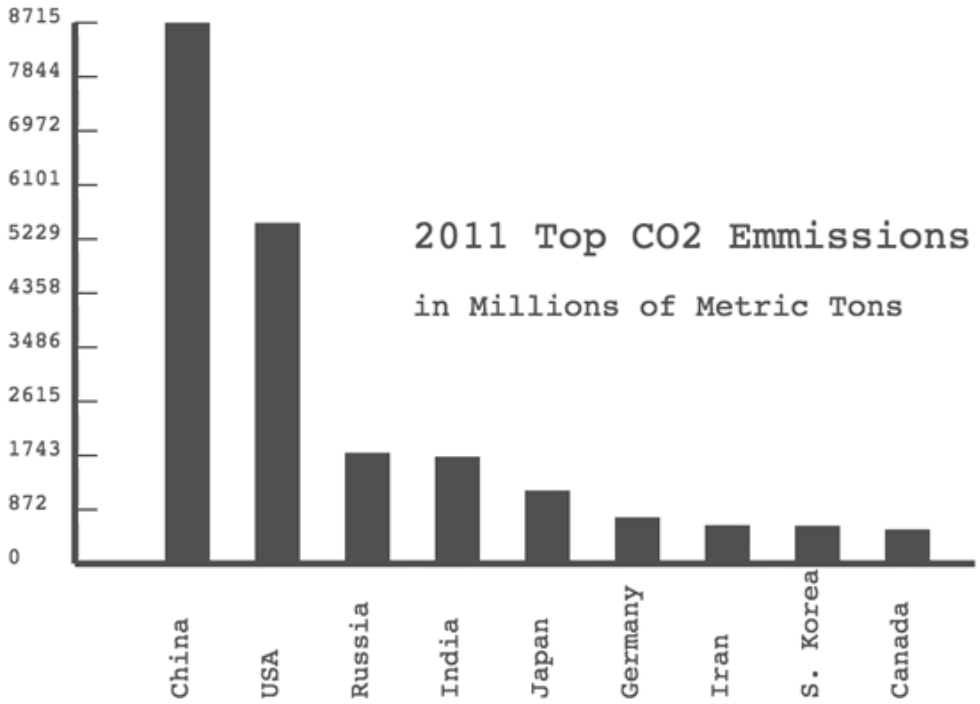


Figure 3.7 Visualization Type: Example of Bar Graph
Source: https://bjc.edc.org/

data categories. Horizontal bar graphs emphasize category similarities, and vertical graphs (also called columns) highlight the values (Ingre et al., 2016). Figure 3.7 shows the top countries that emitted the most carbon dioxide in 2011.

A histogram is a specialized bar graph that displays data aggregated into bins, rather than as individual records. Histograms are widely used to visualize the distributions of quantitative variables, showing their shape, central tendency, range, and variability. Figure 3.8 shows a frequency distribution for response times to tickets submitted to a support system. Each bar covers one hour, and the height represents the count of tickets within each hourly interval (Pearson, 1895).

3.1.4 Geospatial Maps

Geospatial maps utilize a latitude/longitude reference system and include varieties such as world maps, city maps, and topographic maps. Map visualizations often integrate points,

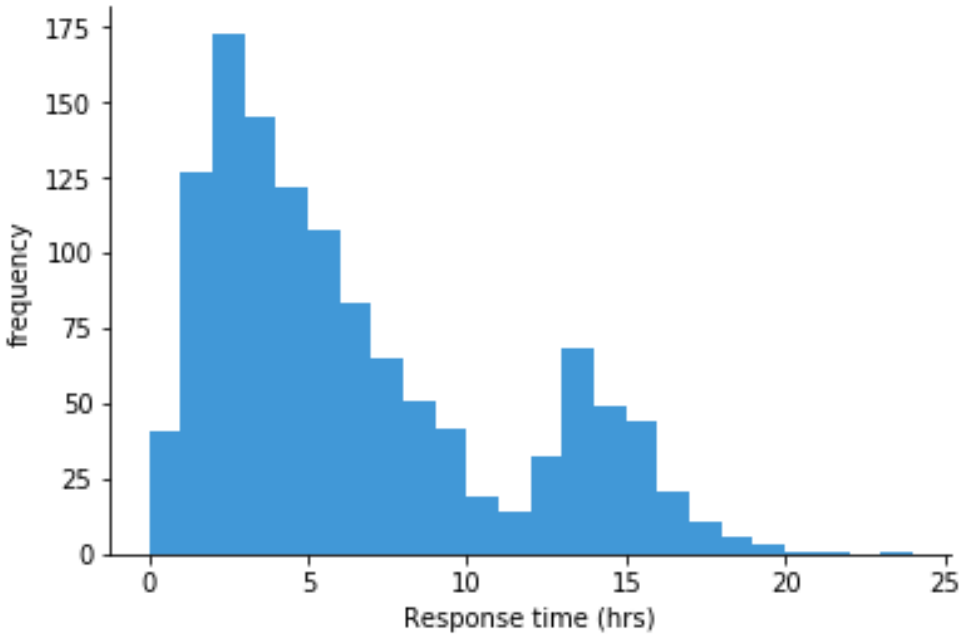


Figure 3.8 Visualization Type: Example of Histogram
Source: https://chartio.com/learn/charts/histogram-complete-guide/

lines, areas, and volumes, and can take forms such as regional maps, heatmaps, numerical maps, point maps, bubble maps, and flow maps. These maps are used to evaluate, view, and analyze geographical data including flows, distributions, and comparisons. Figure 3.9 shows a 2008 U.S. county unemployment choropleth map by Bostock. Darker blues indicate higher unemployment rates, and lighter blues indicate lower rates (Börner et al., 2014).

3.1.5 Trees

Trees visually represent tree data structures, which can be defined recursively as collections of nodes (Knuth, 1997). These structures typically begin with a root node that has no parent. Nodes may be positioned based on their attribute values or relationships, and their size, shape, or color can encode additional quantitative variables. Nodes are connected by links, which represent relationships between them. Tree visualizations are useful in various fields such as family relations, taxonomy, evolution, computer science, mathematics, and business. Figure 3.10 shows a tree that represents an organizational structure.

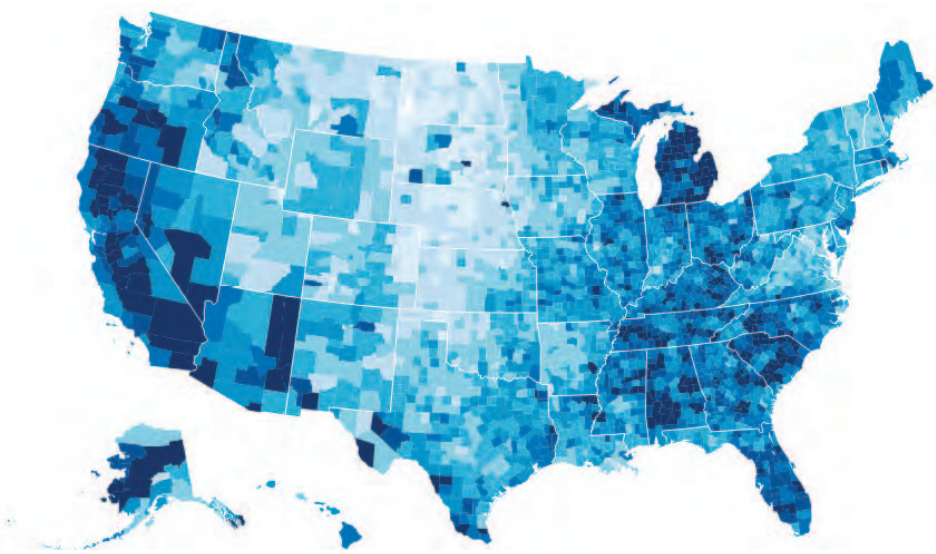


Figure 3.9 Visualization Type: Example of a Geospatial Map - Choropleth Map of U.S. Unemployment Data
Source: Visual Insights (Börner et al., 2014), Rendered by Bostock

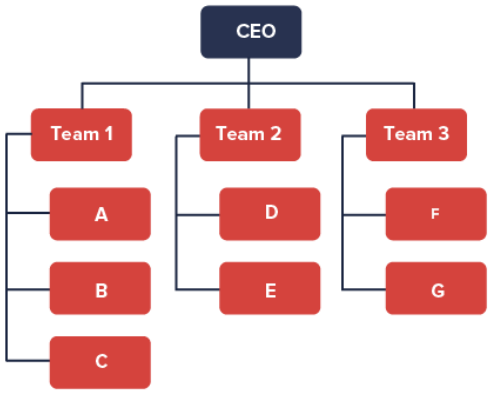


Figure 3.10 Visualization Type: Example of a Tree - Organization Structure
Source: https://datavizproject.com

An alternative tree structure visualization is the treemap, which uses nested rectangles sized according to category quantities (Asahi et al., 1995; Shneiderman et al., 1998, 2001). Each category is represented by a rectangle that contains rectangles for its subcategories. Treemaps provide a compact, space-efficient overview of hierarchical structures.

3.1.6 Networks

Networks, also known as node-link diagrams, use nodes and links to represent connections between entities. These visualizations effectively display relationships such as social networks, corporate structures, and other interconnections. Figure 3.11 shows a network that aims to communicate bursts of activity in a dataset of publications over two decades from the *Proceedings of the National Academy of Sciences* (PNAS) in the United States (Mane et al., 2004).

In network graphs, nodes are typically shown as dots or circles, and links are usually displayed as simple lines connecting the nodes. In some networks, not all of the nodes and links have the same value, and the size, shape, or color of the nodes is used to represent different data values.

3.2 Types of Visualization Deployment

3.2.1 Static Visualizations

Static visualizations are fixed and do not change over time nor allow for interaction. They exist as printed or digital media formats and are effective if they represent all necessary information without needing further interaction from the viewer.

3.2.2 Coupled Windows

Coupled Windows, or multiple panel visualizations, bring multiple visualizations, either static or interactive, together to support comparisons or show evolution over time. Figure

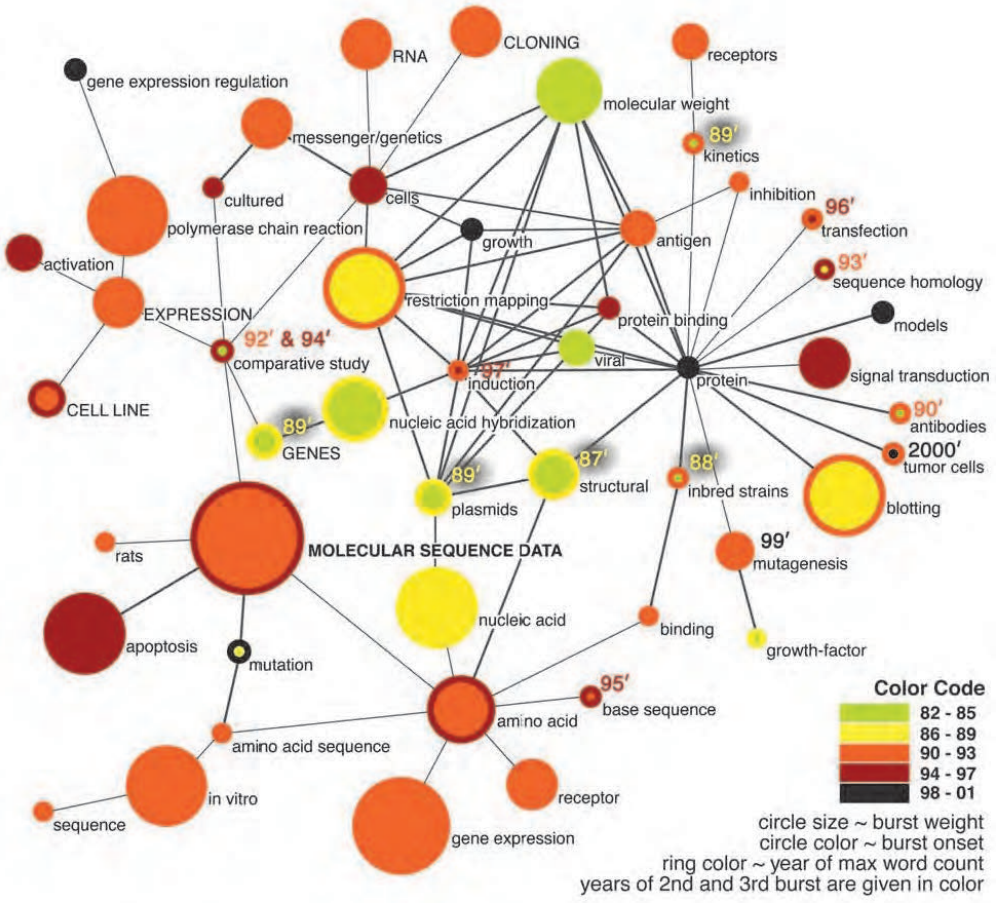


Figure 3.11 Visualization Type: Example of a Network - Mapping Topic Bursts in PNAS Publications
Source: Visual Insights (Börner et al., 2014; Mane et al., 2004)

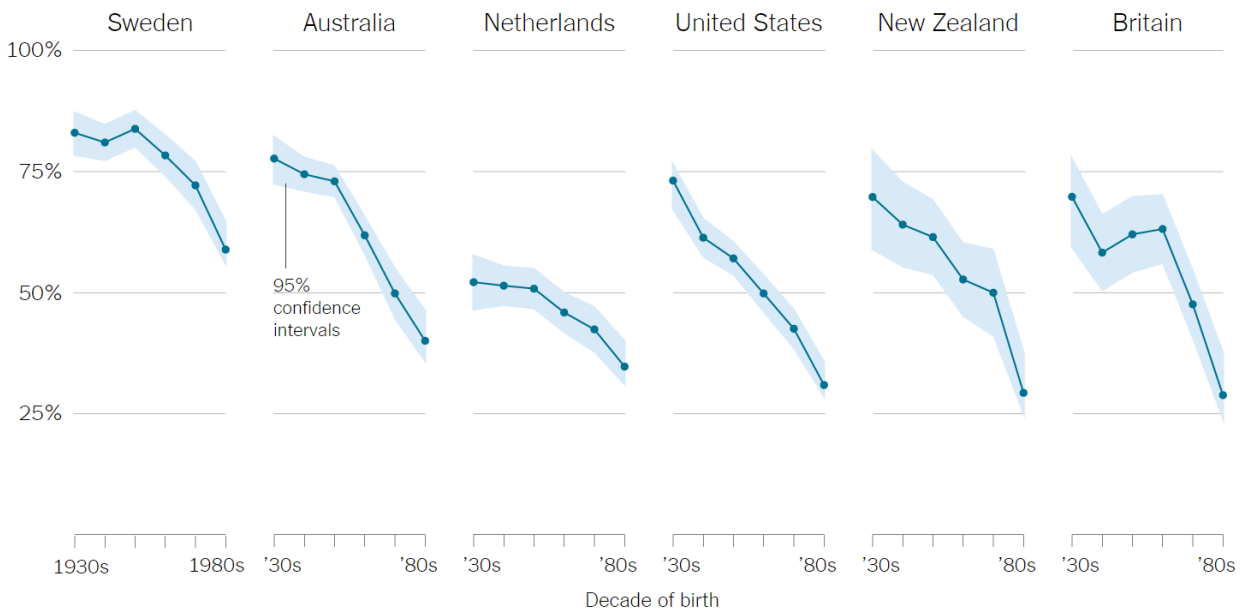


Figure 3.12 Example of Multiple Panels Misualization - The Signs of Democratic Trend
Source: The Signs of Deconsolidation (Foa et al., 2017)

3.12 shows an example that features the comparison of the signs of democratic trends in the United States and many other liberal democracies. The panels in these visualizations can be arranged horizontally, vertically, or in an m*n matrix, effectively expanding the dimension of the visualization in a way that is straightforward for most viewers to understand.

3.2.3 Videos

Video is an electronic medium that depends on the transfer of electronic signals for recording, copying, playback, broadcasting, and displaying moving visual media (Y. Spielmann, 2010). Video infographics, as a type of data visualization, display either static or dynamic data across a series of frames, with or without audio. Unlike static media such as paper, posters, or images, video visualization extends the time dimension by showing a sequence of static visualizations or multiple channels. Video visualization can also have multiple frames of images as animation to show dynamic visualization, e.g., a dynamic visualization might track stock market trends over a month. Although most video visualizations are read-only,

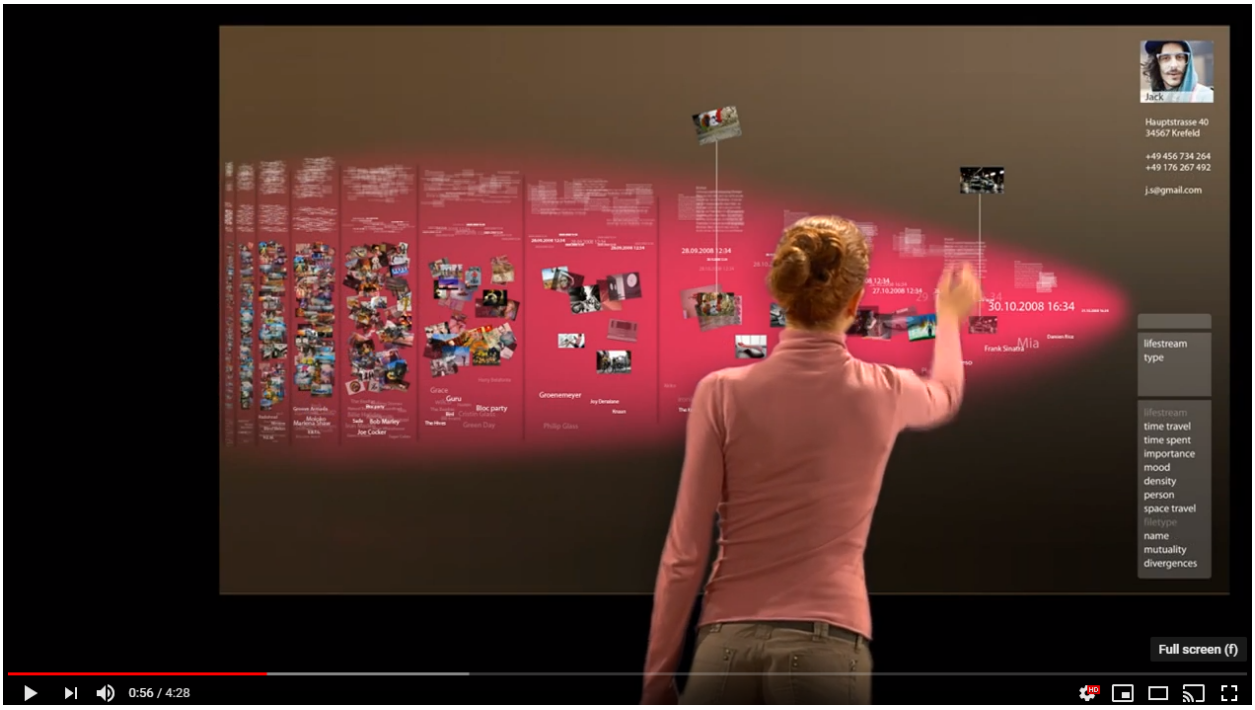


Figure 3.13 Example of Video Visualization
Source: https://www.youtube.com/watch?v=blBDUGveTRo, Video Title: Experientia - LifeStream, Author/Copyright Holder: Experientia

the development of HTML5 (Hjelsvold et al., 2001; Purnamasari et al., 2014) and new video formats like MPEG-4 (Schafer, 1998) allow for the embedding of interactive UI elements (such as buttons, sliders, checkboxes, etc.) for interactive visualization on web platforms.

The primary advantage of video visualization lies in its storytelling capability. Modern storytelling often relies on video, which effectively communicates narratives through dynamic elements like animated charts and numbers. This allows viewers to easily follow data transitions and narratives, enhancing their understanding of the story conveyed by the visualization.

3.2.4 Interactive Visualizations

Interactive visualizations combine visualization with interactivity that encourages the viewer to explore and manipulate the data space (Brodbeck et al., 2009). These interactive visualizations facilitate the display of multidimensional data and promote an intuitive under-

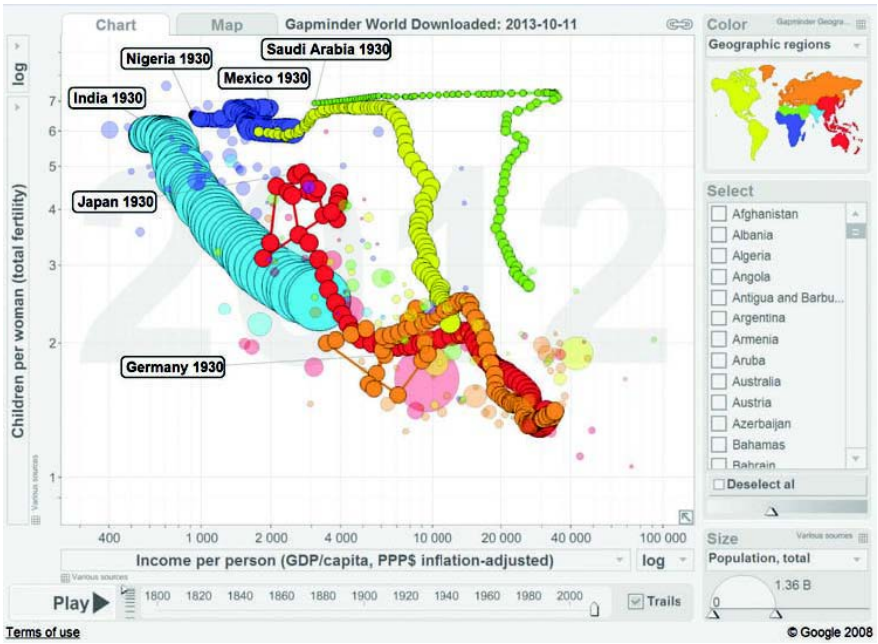


Figure 3.14 Example of Interactive Visualization
Source: Atlas of Knowledge: Anyone Can Map (Börner, 2015)

standing through various visualization styles. They also allow users to perform conventional data exploration activities through interactive charts. As discussed in Section 3.2.3, some video visualizations can have interactive elements. Interactive visualizations require human inputsuch as clicking a button, sliding a slider, or entering data in a fieldand provide real-time feedback that illustrates the relationship between input and output. This interactivity enables viewers to explore, manipulate, and interact with the information through dynamic charts, color changes, and shapes based on queries or interactions. The growing popularity of interactive visualizations in business intelligence is increasing because interactive visualizations provide access to real-time data, making it possible to create dashboards.

3.2.5 Physical Model Visualizations

Physical model visualization uses simple materials, such as "building blocks," to create three-dimensional representations of data (Huron et al., 2014). Unlike two-dimensional diagrams or photographs, physical models like globes provide an undistorted 3D representation of a

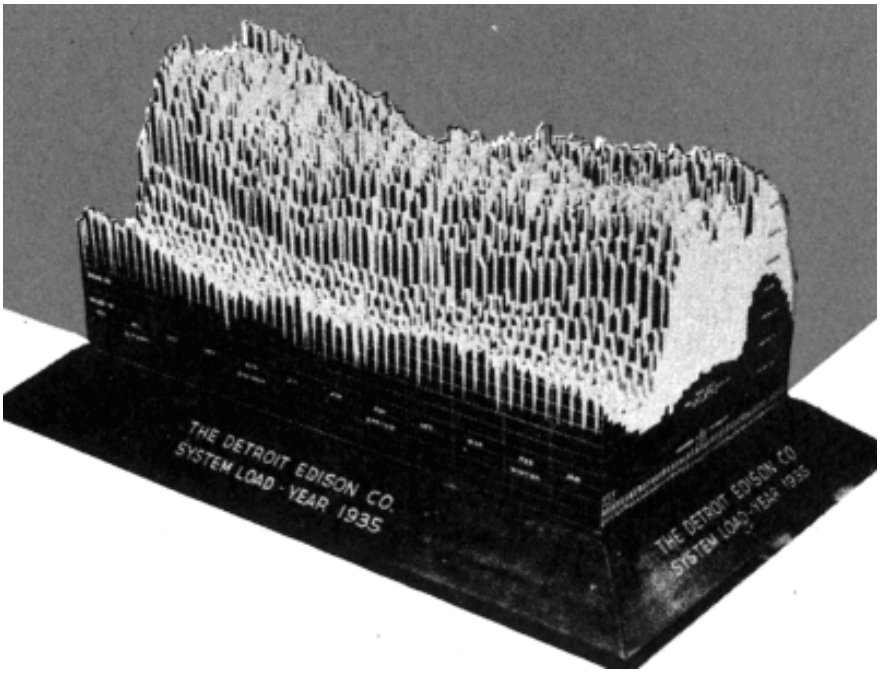


Figure 3.15 Example of Physical Model Visualization
Source: Graphic Presentation (Brinton, 1939)

planet. The range of physical models is extensive, from rescaled physical objects such as architectural models of buildings, to abstract data visualizations using physical models as basic elements, such as Lego blocks to illustrate complex information. Figure 3.15 shows an early 3D physical visualization made by the Detroit Edison Company, which was used to show the anticipated power demands (Brinton, 1939). This large 3D visualization showed electricity consumption in 1935, with each day represented by a slice divided into 30-minute intervals.

3.3 Discussion

The different visualization types introduced in detail in Chapter 3 provide various options for effectively visualizing key insights from Convolutional Neural Networks (CNNs) and U-Nets in subsequent chapters. For instance, tables are useful for describing model parameters precisely. Line and bar graphs effectively compare trends in performance and distributions of model outputs. Heatmaps are excellent for uncovering patterns in feature map activa-

tions. Interactive dashboards provide a flexible analysis tool by combining customizable, coordinated views.

The principles and methods of Data Visualization Literacy (DVL) guide the optimal use of visualizations to understand and share insights about CNNs and U-Nets. For example, recognizing that CNNs learn spatial hierarchies helps determine that layered heatmap grids can communicate findings most clearly and effectively. The DVL's iterative process further refines heatmap design choices and interactivity features to emphasize key discoveries in experiments involving different CNN architectures. Purposefully applying the comprehensive DVL framework helps make the black box of CNNs and U-Nets more transparent through specific visualizations optimized for specific research objectives.

Chapter Four

Visualizations of Convolutional Neural Networks

4.1 Purpose of CNN Visualizations

Nowadays, machine learning systems can outperform humans on some tasks, such as email spam prediction (Androutsopoulos et al., 2000) or human face recognition (Mnih et al., 2013). Sometimes, we find a machine learning system not working correctly or not functioning as humans expect. In certain situations, understanding why a decision was made is less relevant as long as the performance on a test dataset is satisfactory. However, when a model's decision differs from human expectation, comprehending the "how" and "why" will help us understand the issue and optimize the model. Figure 4.1 shows the performance and explainability of different machine learning techniques. The green dots in the figure represent commonly used interpretable models, such as linear regression, logistic regression, and decision trees. From the figure, we can observe that although the red dots, especially deep learning, exhibit top learning performance, they have the poorest explainability. The need for interpretability arises not only from a lack of problem formalization (Doshi-Velez et al., 2017) but also from human curiosity and the interest in correcting predictions or improving accuracy. We are unsatisfied with predictions from a "black box". Even if the prediction is correct, we demand to interpret the model. The current benchmarks for evaluating a machine learning model, like accuracy or loss, indicate when machine learning is functioning correctly, but we are interested in an explainable machine learning model that can provide insights into why it is working or not.

Data visualizations can display quantitative or qualitative data, as described in Chapter

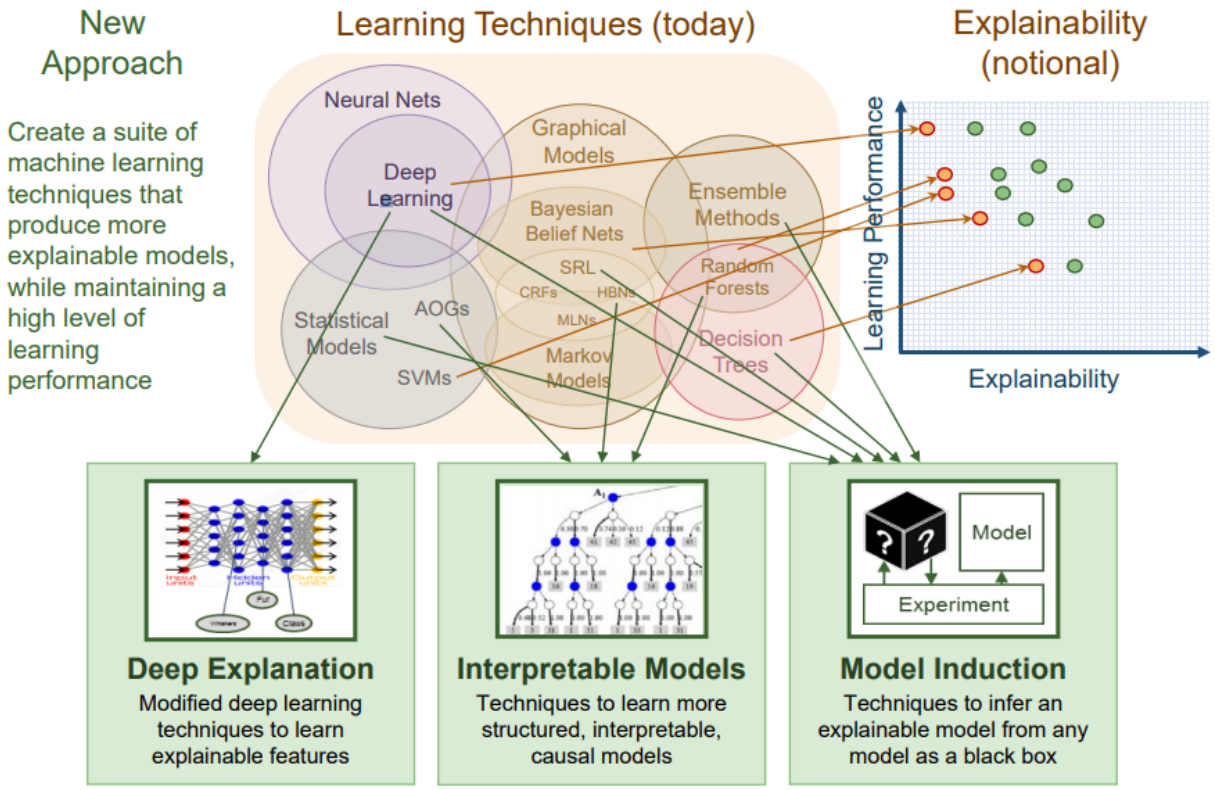


Figure 4.1 Machine Learning: Performance vs. Explainability
Source: Explainable Artificial Intelligence (xai) (Gunning, 2017)

3. The visualization of Convolutional Neural Networks (CNNs) is an underutilized quantitative method to analyze CNNs. We can utilize various visualization methods to cover CNN data, structure, features, learning process, and prediction results. Unlike other machine learning models, CNNs were inspired by the organization of the animal visual cortex (Hubel et al., 1968), and visualizations greatly aid in interpreting CNN's internal features, such as layer activation. Interpretability is related to the human ability to understand the CNN (Qin et al., 2018).

Consider the feature visualization in CNNs as an example. Figure 4.2 shows the analogy between feature extraction in CNNs and information processing in the human visual cortex system. The human visual system processes image features across several visual neuronal areas in a feed-forward and hierarchical approach. When humans recognize a face, the visual neurons with small receptive fields in the V1 (lower visual neuron) area are sensitive to basic visual features (Hubel et al., 1959, 1968; Manassi et al., 2013), such as basic elements of data visualization like edges and lines. As the visual signal is processed by higher-level neuron areas, humans understand more complex features like shapes, objects, and finally, the entire face is detected. Similarly, CNN feature extraction begins in the first several convolution layers (CL) with small features such as edges and colored blobs (Qin et al., 2018). Like the human brain, the last convolution layer shows the most detailed shapes. Therefore, by comparing the functionalities of brain visual neurons' receptive fields to CNN's neurons (Kruger et al., 2012), visualization techniques help depict the CNN and thus improve its interpretability.

The DVL framework provides clear guidance on effectively visualizing CNN insights through deliberate visual encoding choices. The variety of DVL visualization types also offers flexible visualization options for CNNs. Interactive dashboards can combine complementary techniques like loss curves, activation heatmaps, and high-dimensional data plots.

Following the DVL process model is crucial for machine learning models, especially the CNN visualizations discussed in this chapter. The step-by-step workflow ensures proper

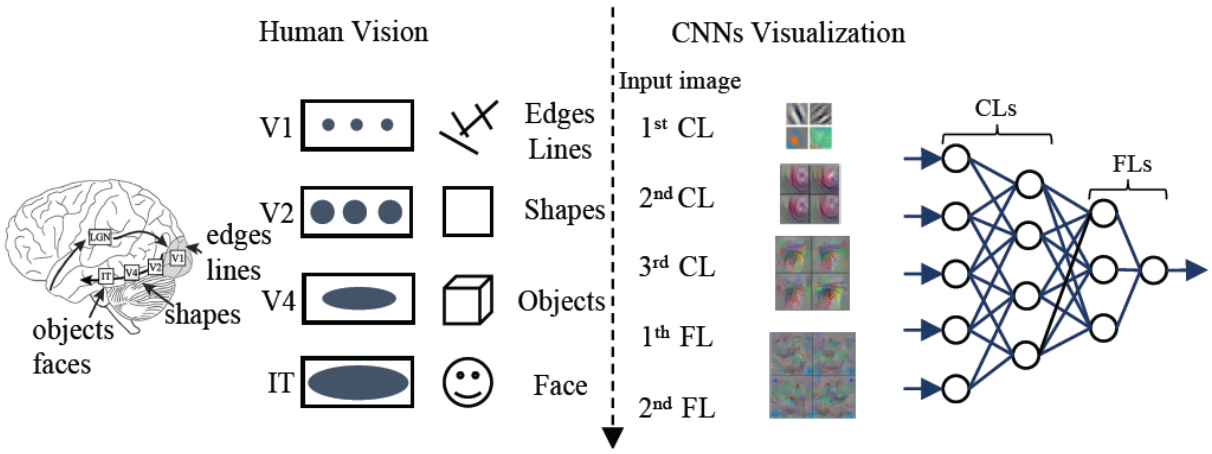


Figure 4.2 Human Vision and CNNs Visualization
Source: How Convolutional Neural Network See the World - A Survey of Convolutional Neural Network Visualization Methods (Qin et al., 2018)

acquisition and examination of model data, then careful visualization design based on different types of readers, from novice users to experienced researchers. This prevents displaying generic visualizations and focuses on conveying specific discoveries and results.

4.2 Types of CNN Visualizations

4.2.1 Visualizations of CNN Datasets

The ability to classify images of CNNs stems from their capacity to learn patterns from large amounts of image data. Understanding the CNN dataset is also a critical problem for comprehending the CNN.

Although it is challenging to find a general method to visualize every dataset, it is possible to identify principles for visualizing specific large-scale datasets consisting of data points ranging from thousands to millions, which may include hundreds or thousands of features. There are two main purposes for visualizing the dataset: one is to show insights from the dataset's metadata, and the other is to help select the most important features of the data before feeding it into the machine learning model.

Typically, the CNN dataset consists of labeled images, and the scales, structure, and

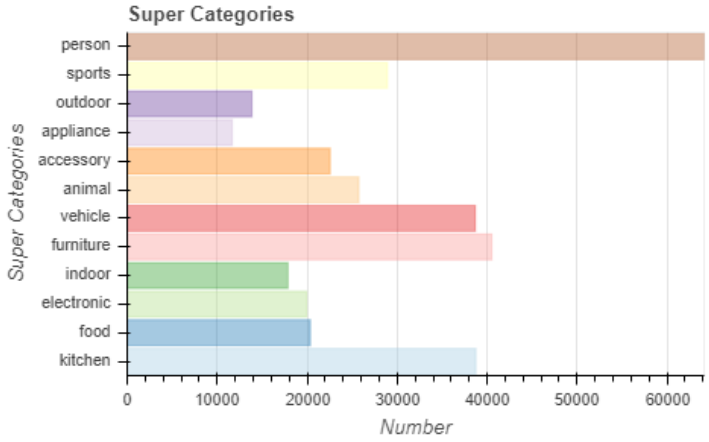
metadata of datasets may vary. For example, Microsoft COCO (T.-Y. Lin et al., 2014) is a publicly accessible large-scale object detection, segmentation, and captioning dataset. Besides the common image category (or class in CNN) labeling, the Microsoft COCO dataset also includes annotations for common object categories, instance spotting, instance segmentation, and caption annotation.

The goal of the COCO dataset is to advance the state-of-the-art in object recognition by placing the question of object recognition in the context of the broader question of scene understanding (T.-Y. Lin et al., 2014). The dataset contains photos of 91 object types that would be easily recognizable by a 4-year-old and contains a total of 2.5 million labeled instances across 328k images (T.-Y. Lin et al., 2014). The distribution of images in 12 supercategories can be found in Figure 4.3 part 2. Figure 4.3 part 1 shows the icons of the 91 object categories in the COCO Explorer.

Figure 4.4 shows a visualization of metadata extracted from the COCO dataset. Figure 4.4 part 1 shows the frequency of two annotations appearing together in one image, where a darker color indicates a higher frequency. For example, Figure 4.4 part 1 shows the frequency of a vase and a book appearing in the same image. Figure 4.3 part 3 shows the distribution of the position of each annotation in the images by calculating the average location of the same object in all images. For instance, it can be seen that the airplane/stop sign usually appears in the center part of the images, and the motorcycle usually appears in the center and the bottom part of the images. The COCO dataset has annotated captions for each image, and Figure 4.4 part 2 shows the statistics of the most frequently used words in the captions and the positions of the words in each sentence. A ridge plot was used to show the distribution of the word positions in the sentences. The circles indicate the frequency of appearance of the most used words in the captions, with larger circles indicating higher frequency.

Part 1

Part 2



Part 3

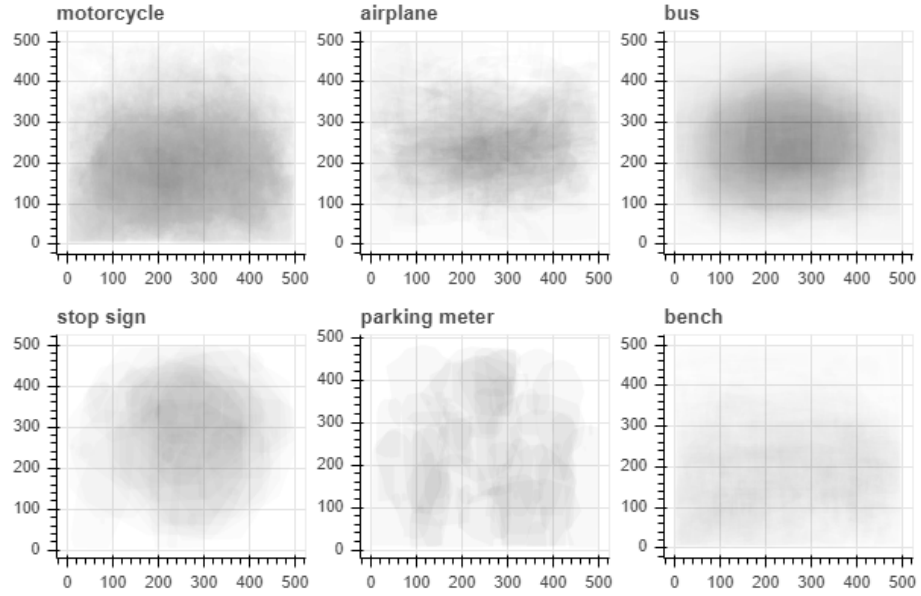


Figure 4.3 Visualizations of CNN Datasets - Metadata

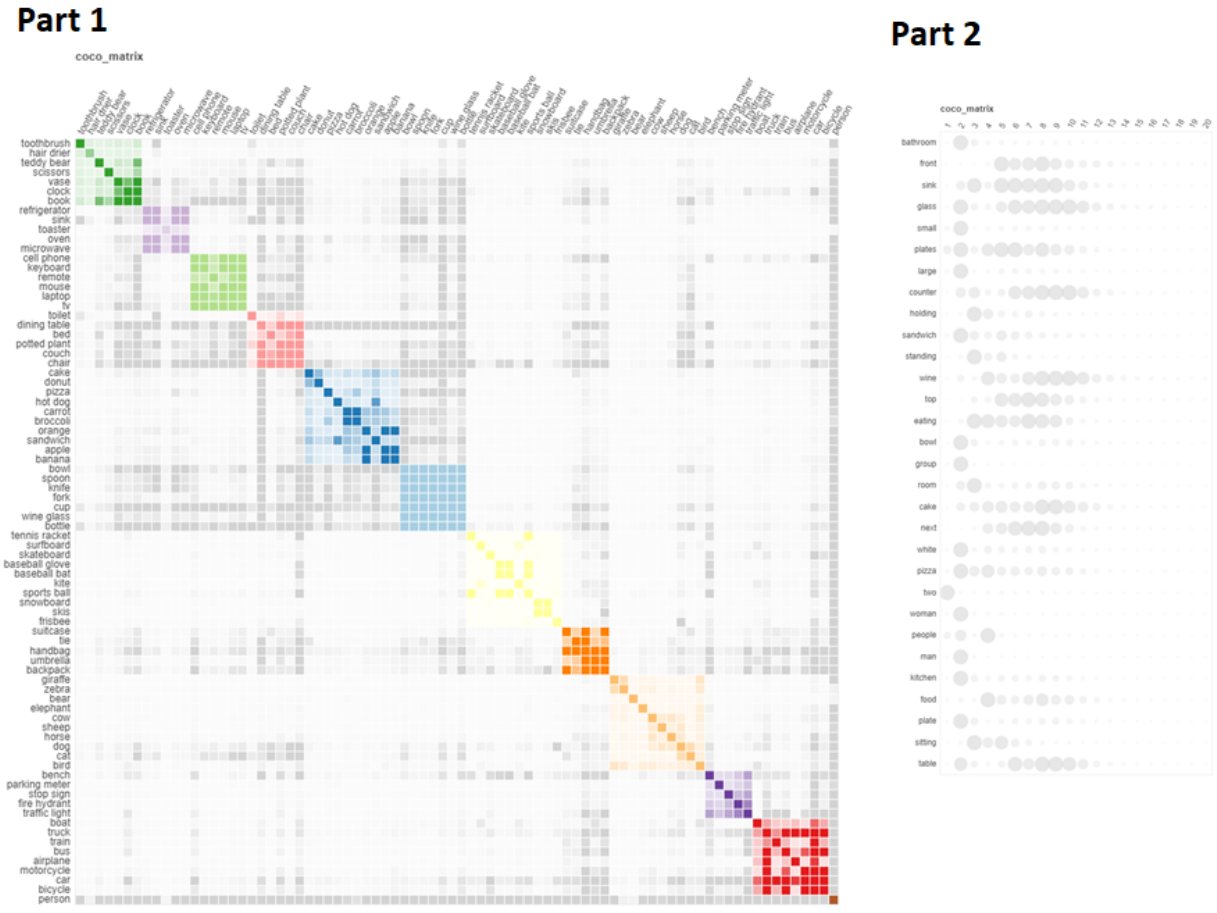


Figure 4.4 Visualizations of CNN Datasets - Matrix

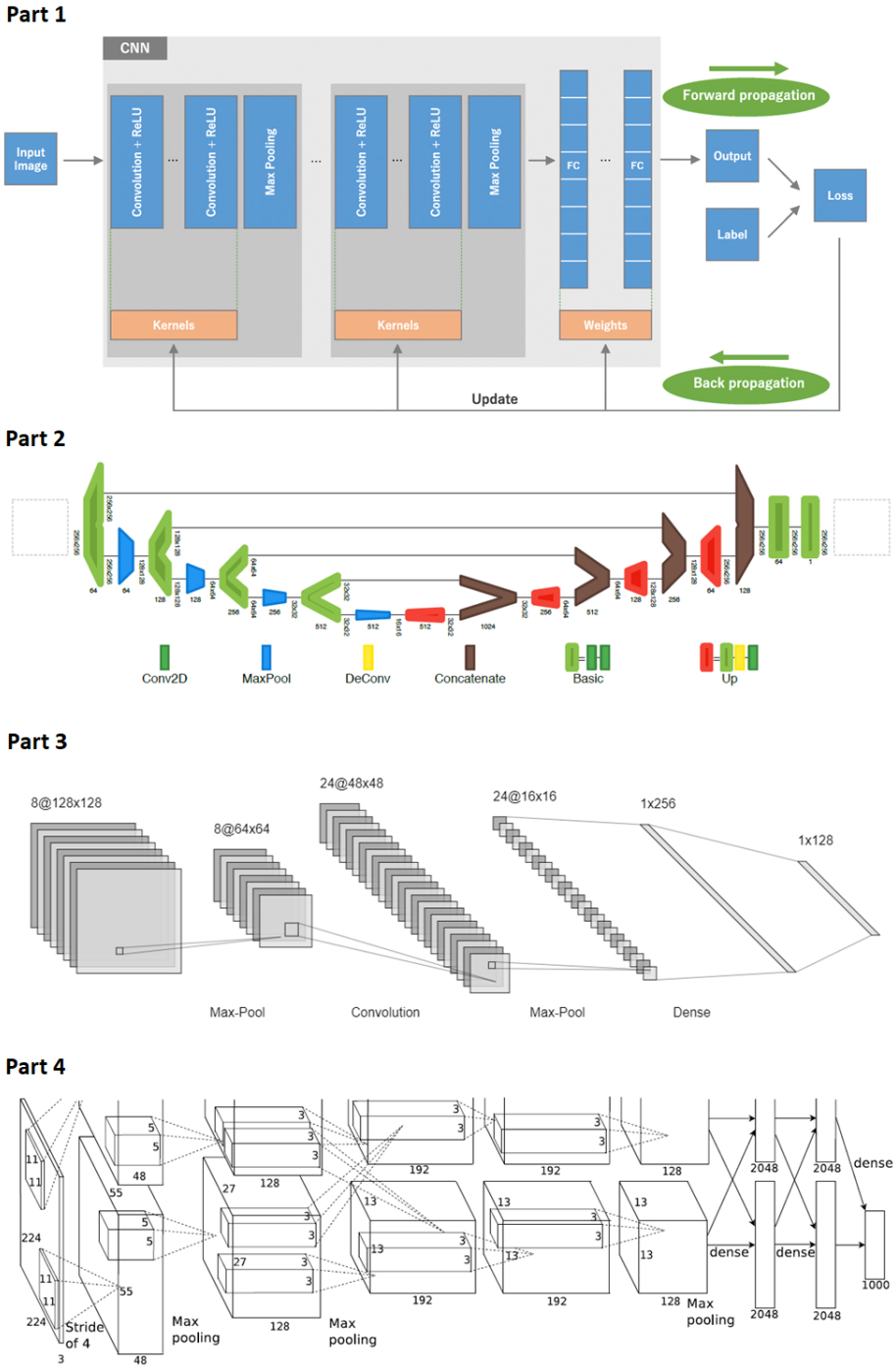


Figure 4.5 Convolutional Neural Network Structure Visualization
Source: Convolutional Neural Networks: an Overview (Yamashita et al., 2018)

4.2.2 Visualizations of CNN Structure

CNN's architecture is formed by a stack of distinct layers. The inputs to a CNN (e.g., images, sentences, etc.) are transformed by these layers into the output (e.g., a class). Figure 4.5 part 1 shows a typical structure of a CNN with images as the input. CNN is a mathematical construct that is typically composed of three types of layers (or building blocks): convolution, pooling, and fully connected layers (Yamashita et al., 2018).

To properly convey the architecture of convolutional neural networks, appropriate visualization techniques are of great importance (Bäuerle et al., 2019). Illustrations of neural Network architectures are often time-consuming to produce, and machine learning researchers frequently find themselves constructing these diagrams from scratch by hand (LeNail, 2019). Various basic graphic variable types can be applied to build the static CNN structure, including point, line, area, surface, volume, and text with different retinal attributes, such as size, shape, rotation, and angle (Börner, 2015). Figures 4.5 parts 2, 3, and 4 show several commonly used visualizations that aim to communicate the global visualization of the structure of CNNs. While there is no established "gold standard" for the visualization of CNN structure from prior work, we can identify some general design patterns. In these figures, convolutional and pooling layers are usually represented by area symbols, and the size is related to the width and height of the layers. The connections between the layers are line symbols, but typically, the network of connections is simplified to a beam of several lines to reduce the complexity of the architecture.

The visualization of Convolutional Neural Networks (CNNs) is typically static, emphasizing the network's overall architecture. An alternative approach to understanding CNN structure involves visualizing the convolutional layer filters, which display the weights of these filters. These weights are most interpretable at the first convolutional layer, where they interact directly with raw pixel data. However, it is also feasible to visualize filter weights deeper in the network to represent CNN's local state. By visualizing the network

layer filters, we can discern the patterns to which each filter is responsive.(Krizhevsky et al., 2012) This process can be achieved by using gradient descent (Ruder, 2016), a first-order iterative optimization algorithm used to find a local minimum of a differentiable function, on a CNN's value to maximize the activation of a particular filter, beginning with a blank input image.

Visualizing the weights is informative because well-trained networks typically show smooth, coherent filters. On the other hand, noisy patterns may indicate a network that has not been sufficiently trained or one with a very low regularization strength, potentially leading to overfitting. Overfitting occurs when a CNN fits the training set too closely, hindering its ability to generalize and make accurate predictions on new data. (Santos et al., 2022)

Figure 4.6 illustrates a typical filter from the first convolutional layer (left) and the second convolutional layer (right) of a trained AlexNet (Krizhevsky et al., 2012). The first-layer weights appear smooth, signifying a well-converged network. The color/grayscale features are clustered because AlexNet contains two separate streams of processing. This architectural feature often results in one stream developing high-frequency grayscale features, while the other focuses on low-frequency color features. Although the second-layer weights are less interpretable, their smoothness and well-defined structure suggest the absence of noisy patterns, indicating a robust training process.

4.2.3 Visualizations of CNN Features

The growing need for neural networks to be interpretable to researchers makes visualizing CNNs increasingly useful and necessary. After visualizing the two-dimensional filters learned by the model, we can also visualize the types of features that the CNN detects. Activation maps help understand what features were detected by the CNN for each given input image (Olah et al., 2017). As mentioned at the beginning of this chapter, the extraction of CNN features begins in the first several convolution layers with small or fine-grained details such as edges and colored blobs, and the last convolution layer shows more general features and

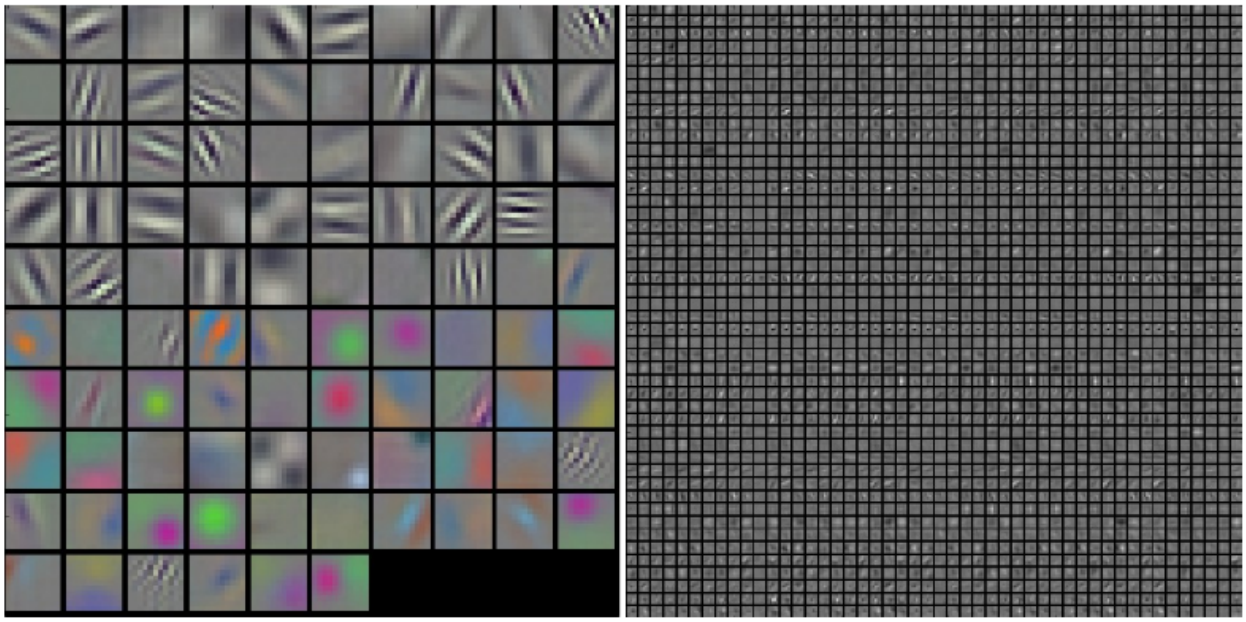


Figure 4.6 Filters on the First and Second Convolution Layer of a Trained AlexNet.
Source: http://cs231n.github.io/understanding-cnn/

the most detailed shapes. (Kruger et al., 2012)

Another approach to visualizing features in CNNs is Activation Maximization (AM) (Erhan et al., 2010). The authors visualized the preferred input patterns for the hidden neurons in the Deep Belief Net (G. E. Hinton et al., 2006) and the Stacked Denoising Auto-Encoder (Vincent et al., 2010) learned from the MNIST digit dataset (LeCun et al., 2010). The learned feature is represented by a synthesized input pattern that can cause the maximal activation of a neuron (Qin et al., 2018). The paper "Understanding Neural Networks Through Deep Visualization" (Yosinski et al., 2015) also uses activation maximization to visualize which channels respond most strongly to the detected object. Figure 4.7 shows a visualization of example features of eight layers of a deep, convolutional neural network by Activation Maximization.

Compared to the feature visualization in CNNs in Figure 4.2, the features in Figure 4.7 are not human-recognizable. Therefore, some methods of incorporating raw image priors into the objective function were introduced to substantially improve the recognizability of

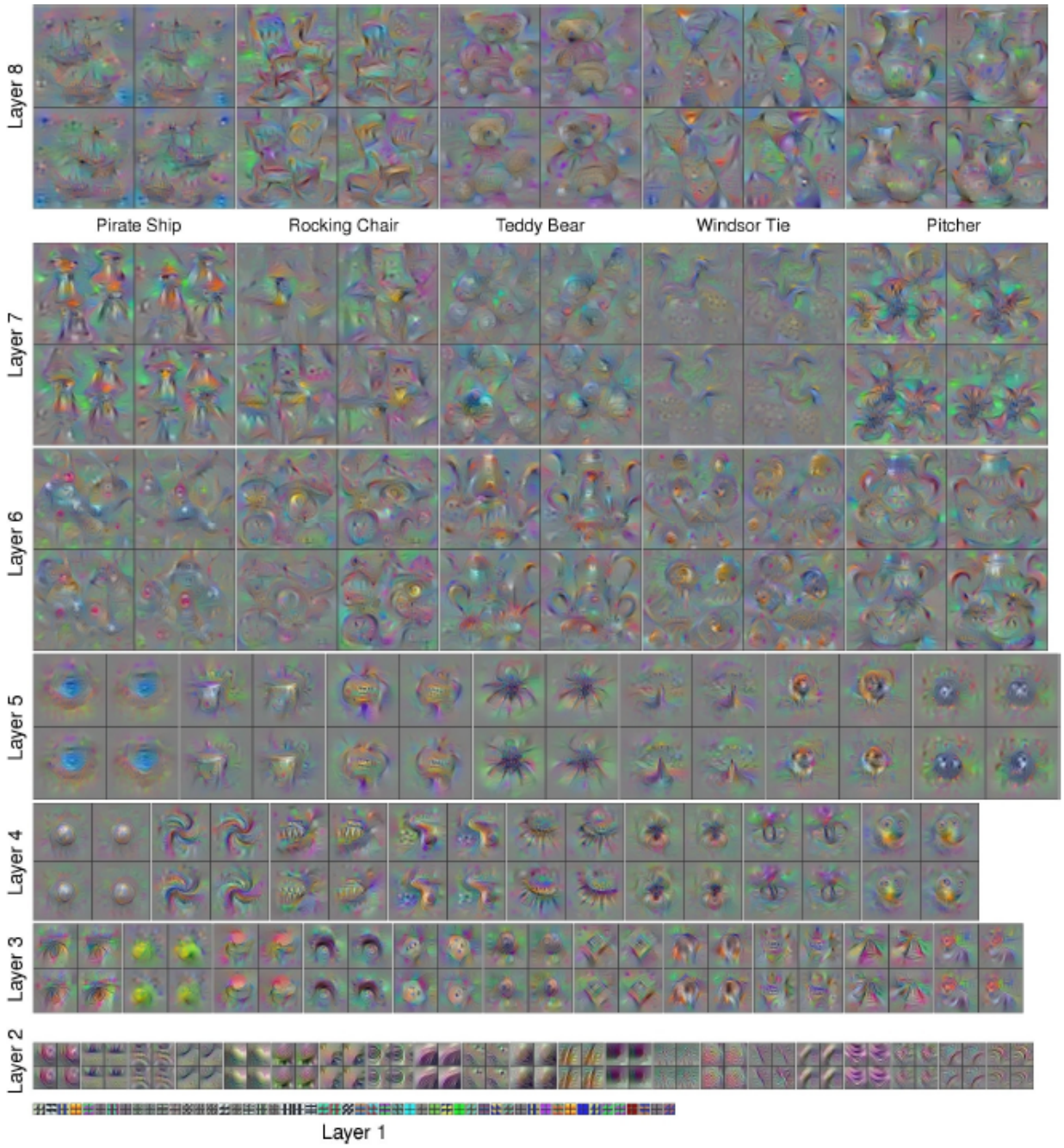


Figure 4.7 Visualization of Example Features of Eight Layers of a Deep, Convolutional Neural Network by Activation Maximization
Source: Understanding Neural Networks Through Deep Visualization (Yosinski et al., 2015)

the features (Mahendran et al., 2016; Nguyen et al., 2016a,b; Olah et al., 2017; Yosinski et al., 2015). Such constraints are often included in the AM formulation as a regularization term $R(x)$, and then the activation function is defined as $x^* = \text{argmax}_x(a(x) - R(x))$. Additionally, different image priors (such as L_2 norm, Gaussian blur, etc.) can be employed in the activation function to produce different feature images. Figure 4.8 shows Activation Maximization results of seven methods which are synthesized to maximize the output neurons (each corresponding to a class) of the CaffeNet image classifier (Krizhevsky et al., 2012), and the categories were selected based on the images available in previous papers (Mahendran et al., 2016; Nguyen et al., 2016a; Simonyan et al., 2013; Wei et al., 2015; Yosinski et al., 2015). From Figure 4.8, we can see that with different constraints, the features can be interpreted as photo-realistic images or images that look like those in the training set, and therefore, the interpretability is improved.

In (Matthew D. Zeiler et al., 2013), a new visualization technology called Deconvnet was proposed that reveals the response relationship between the input and the feature map of any layer in the model. The Deconvnet visualization takes the feature maps obtained by each layer as input, performs deconvolution, and produces the result, which is used to verify and display the feature maps extracted by each layer. To check the activation of a given CNN, they connect the activated feature map to a deconvolution network and then pass it through depooling, deactivation, and deconvolution repeatedly until the input layer. Figure 4.9 shows the top 9 activations in a random subset of feature maps across the validation data, projected down to pixel space using the deconvolutional network approach. Projecting each feature map separately down to pixel space reveals the different structures that excite a given feature map, thus showing its invariance to input deformations (Matthew D. Zeiler et al., 2013). They also show the image patches that correlate with these visualizations. Because visualizations focus primarily on the discriminant structure inside each patch, they exhibit more variance than the visualizations themselves. For example, the patches in layer 5, row 1, and column 2 appear to have little in common, but the visualizations demonstrate

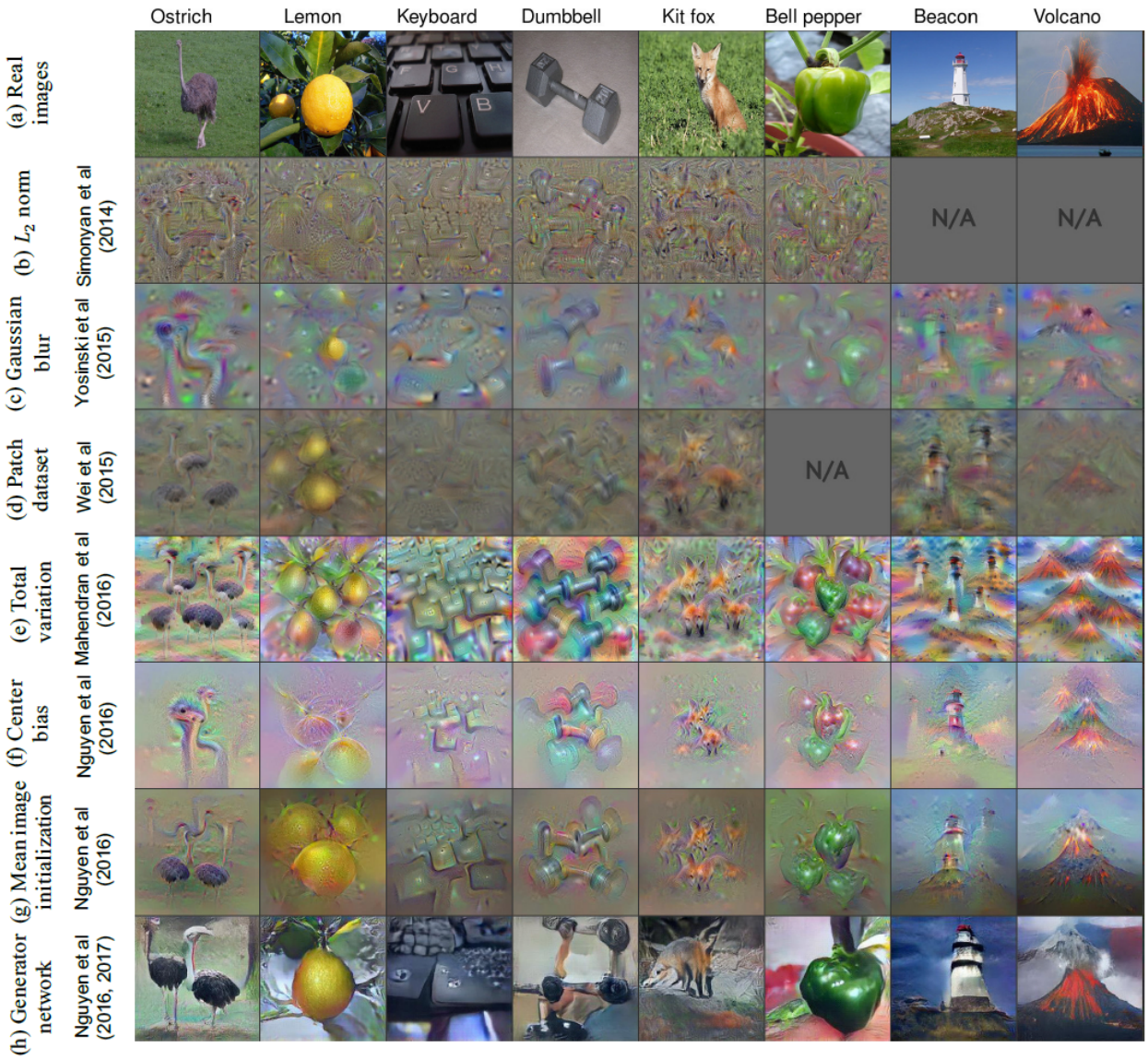


Figure 4.8 Activation Maximization Results of Seven Methods
Source: Understanding Neural Networks via Feature Visualization: A survey (Nguyen et al., 2019)

that this feature map is focused on the grass in the background rather than the foreground objects (Matthew D. Zeiler et al., 2013). The hierarchical nature of the network's features is demonstrated by the projections from each layer. The second layer responds to corners and other edge/color information, the third layer shows more complicated invariance and captures similar textures, the fourth layer shows considerable changes and is more category-specific, and the fifth layer exhibits considerable posture changes. Although the visualizations cannot fully explain the CNN's black box, they show that the features learned by the CNN layered characteristics. The visualization also provides advice on how to improve the CNN model. One of the simplest approaches is to make the network deeper and the characteristics of each layer more unique, so the model can learn more generalized features. The subsequent VGGNet (Simonyan et al., 2014) and ResNet (He et al., 2016) proved this point. VGGNet, known for its simplicity and depth, demonstrated that a network with a greater number of layers could lead to improved performance on challenging image recognition tasks (Minaee et al., 2021; Simonyan et al., 2014). ResNet, which enables the training of even deeper networks, further validated the benefits of depth in CNNs (He et al., 2016; Khan et al., 2022).

4.2.4 Visualizations of CNN Learning Process

For the visualization of machine learning models, especially CNNs, the complexity and scalability of the input must be considered. Complexity is measured in terms of the size of the features, and scalability is often related to complexity. In most CNN models, the sizes of the raw input and the internal features are usually fixed, making complexity a more significant consideration. To compare the complexity of the features, especially those internal features generated during the machine learning process, we can compare how the dimensionality of the features is reduced to 2D or 3D for visualization purposes. This comparison then becomes a discussion on the effectiveness of various dimension reduction techniques, including PCA (Principal Component Analysis) (Ku et al., 1995), t-SNE (T-distributed Stochastic

Figure 4.9 Visualization of Features in a Fully Trained Model
Source: Understanding Neural Networks via Feature Visualization: A survey (Nguyen et al., 2019)

Neighbor Embedding) (Maaten et al., 2008), and UMAP (Uniform Manifold Approximation and Projection) (McInnes et al., 2018).

Two other important factors to evaluate the visualization of features are feature selection and generalizability. In a prior project on emotion recognition, a challenge was encountered with the large volume of features generated by our methods. Techniques such as the Short-time Fourier Transform (STFT), Mel-frequency cepstral coefficients (MFCCs) (Tiwari, 2010), and filter banks (Flandrin et al., 2004) produced matrix-based features. Managing these features, whether by combining them into larger matrices or processing them separately, could become computationally intensive. To address this, we treated these matrices as collections of sub-features. By applying visualization techniques, we were able to evaluate the importance of each sub-feature across various dimensions, identifying the most significant ones for the CNN model. For instance, certain sub-features were indicative of gender differences in speech, as shown in the graphs within Figure 4.10. Through this visualization process, we could simplify the feature set by discarding less relevant dimensions, thus enhancing the efficiency of our CNN in distinguishing between classes without compromising performance.

In addition to feature selection for improving the accuracy of CNNs, generalizability is also a crucial factor to consider. Generalizability refers to the model's capacity to accurately adapt to new, unseen data that share the same or a similar distribution as the dataset used to train the model. It measures the algorithm's effectiveness across a variety of inputs and applications. The structure of the RAVDESS dataset serves as an excellent foundation for exploring the extent to which our approach is generalizable across different speakers and emotions. Specifically, we examine whether our methods maintain their effectiveness across repetitions of sentences, different sentences spoken by the same actor, various actors, different levels of emotion intensity, and the eight distinct emotions represented in the dataset. We proved that our methodology could be able to generalize from one version of an utterance to another, provided that the speaker, sentence, and other factors remain constant. However, it

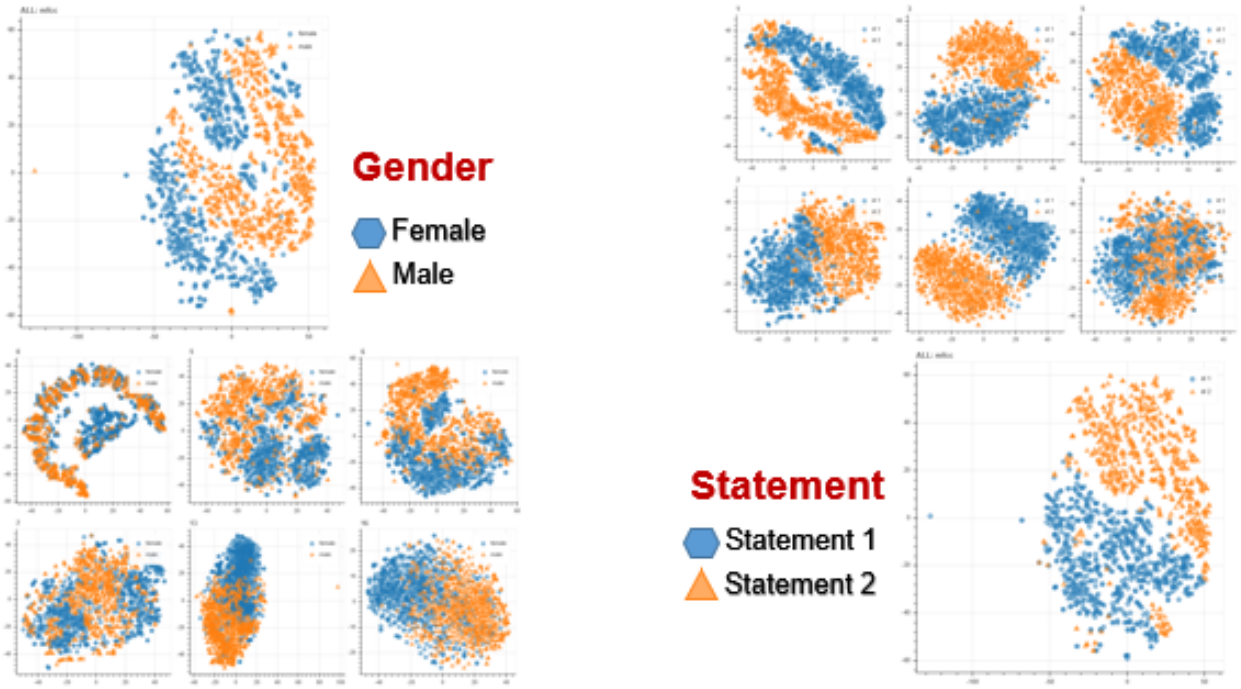


Figure 4.10 The Distributions of Emotion Features (MFCC) between Genders and Statements

remains to be seen whether this generalization holds when considering variations in sentence content (as shown in Figure 4.10). We hypothesize that the challenge of generalization increases when extending it to different actors, emotion intensities, and various emotions, though the degree of this increase is not yet clear.

4.2.5 Visualizations of CNN Results

When using CNNs for image classification, one of the most frequently asked questions is whether the neural network model is truly identifying the location of the object in the image or just using the surrounding context (Matthew D. Zeiler et al., 2013). Class Activation Map (CAM) and Network Dissection visualization are two typical methods to address this issue.

Class Activation Map uses heatmaps to overlay class activation over the input images. This approach replaces the fully connected layers (FL) of the CNN with convolutional layers

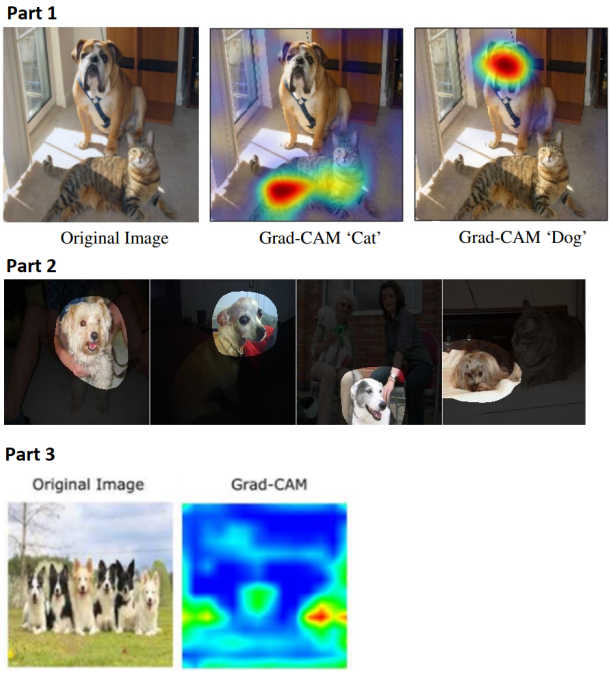


Figure 4.11 Visualization of CNN Results
Source: Grad-CAM: Why Did You Say That? Visual Explanations from Deep Networks via Gradient-based Localization (R. R. Selvaraju et al., 2016)

and global average pooling, thus enabling the generation of class-specific feature maps (M. Lin et al., 2013). Subsequently, the probability can be visualized as a 2-dimensional heatmap of scores associated with a specific output class, indicating the importance of each part of the image for identifying the corresponding output class, as shown in Figure 4.11 part 1.

One limitation of Class Activation Mapping (CAM) is that it requires feature maps that immediately precede softmax layers (a type of neural network output layer that transforms the final feature maps into normalized probabilities) (LeCun et al., 2015). This requirement restricts its applicability to certain CNN architectures that include global average pooling on convolutional maps just before the prediction stage. To overcome this, an alternative method called Grad-CAM (Selvaraju et al., 2017) was introduced, which uses gradient information to extend the CAM approach. Figure 4.11 part 1 illustrates an example of Grad-CAM, showing a comparison between the original image and the visualization of class-discriminative regions. Another drawback of CAM is its requirement for structural modifications and potential

retraining of the original model. This can be impractical if the model is already deployed or if the retraining costs are prohibitive, significantly limiting its usage.

Network Dissection is another visualization method used to interpret the predictions of a CNN (Qin et al., 2018). It provides a general framework for quantifying the interpretability of latent representations within CNNs by evaluating the alignment between individual hidden units and a set of semantic concepts (Bau et al., 2017). Network Dissection directly associates each convolutional neuron with a specific semantic concept, such as color, textures, materials, parts, objects, and scenes by measuring the alignment between the unsampled neuron's activation and the ground truth images with semantic labels (Qin et al., 2018). Therefore, Network Dissection can visualize the types of semantic concepts represented by each convolutional neuron. This process involves the following three steps (Bau et al., 2017).

1. Get images with human-labeled visual concepts
2. Measure the CNN channel activations for these images
3. Quantify the alignment of activations and labeled concepts

The fundamental algorithm of Network Inversion illustrated the correlation between one semantic concept and one individual neuron (Qin et al., 2018). Such a correlation was based on an assumption that each semantic concept can be assigned to a single neuron (Gonzalez-Garcia et al., 2016). The visualization results by using combined neurons are shown in Figure 4.11 part 2.

Despite certain limitations of Grad-CAM and Network Dissection, such as in Figure 4.11 part 3 (Chattopadhay et al., 2018), Grad-CAM's failure to accurately localize objects in images if the image contains multiple occurrences of the same class (Chattopadhay et al., 2018), Class Activation Map and Network Dissection offer unique insights into how neural networks function, particularly in image recognition. Given the complexity of neural networks, visualization is an important step in analyzing and describing them. The visualization of CNN results not only expands our understanding of feature visualization but also enhances the

interpretability of network units. Both Class Activation Map and Network Dissection offer a convenient method to automatically link units to concepts, and that makes them great tools to communicate in a non-technical way how neural networks work. Additionally, these visualizations of CNN results can also be integrated with feature attribution methods, which explain and clarify which pixels were important for the classification.

4.3 Improvement of CNN Visualizations

Accuracy

Besides increasing interpretability, another benefit of visualizing CNNs is the improvement of CNN model accuracy. Predictive accuracy is typically the measure of success for Machine Learning (ML) applications (Bratko, 1997). In some situations, we encounter an interpretability-accuracy trade-off (Ishibuchi et al., 2007), which occurs because as model accuracy increases, so does model complexity, at the cost of interpretability. However, after the stage of "modeling your problem" is complete and when the structure of the model is finalized, the global interpretability of the model provides insight into the relationship between input and output. In other words, an interpreted model can explain why the results are predicted by the current features, and that can help improve the result through feature processing or selection. This improvement can be achieved with or without some domain-specific background knowledge, which is beneficial for the learning system, rather than indiscriminately incorporating all data into the model.

Efficiency

Another advantage of machine learning interpretability is improved efficiency. The trade-off between accuracy and efficiency has been a frequent topic of discussion in the history of machine learning (J. Huang et al., 2017; Xie et al., 2018). In certain cases, a slight reduction in accuracy is acceptable if it leads to a significant increase in speed (Andri et

al., 2017; Rastegari et al., 2016). One alternative approach to enhancing the speed of a machine learning model involves reducing the model size while maintaining the accuracy of predictions. Some studies (Long et al., 2015; Zheng Zhang et al., 2016) start with the well-known VGG 16 network (a 16-layer network developed by the Visual Geometry Group Lab (Simonyan et al., 2014)), remove the topmost layers, and replace them with a different structure to enhance efficiency. Improved interpretability can aid this process by visualizing the fully connected layer to highlight the importance of each connection, thus allowing for the pruning of 'non-essential' connections with minimal accuracy loss. As the input dataset becomes more consistent, prioritizing efficiency becomes crucial since the model's ability to adapt to new data is less of a concern, allowing for a simplified architecture that maintains performance while operating faster.

4.4 Discussion

With recent advancements in technologies such as face and speech recognition, the underlying technologies, such as machine learning and deep learning, have drawn significant attention from engineers and researchers. The cost of constructing and implementing neural networks has never been lower. However, complex deep learning models, on the other hand, remain challenging to train and comprehend. Most engineers concentrate on feature extraction, algorithm selection, parameter tuning, and model verification, making a user-friendly and efficient tool essential for reducing the learning curve. Data visualization has become increasingly popular in recent years as a tool for exploratory data analysis before applying machine learning models. Through visualizations, especially those that are interactive, dynamic, and customizable, people can understand what the models have learned and how they make predictions.

The CNN model has produced excellent results in various applications, particularly in image classification and object detection. However, CNN is often seen as a "black box" model.

It is unclear why CNN performs well and how its accuracy might be enhanced due to the lack of explainability. For example, hidden layers in CNNs do not directly interact with the external world. Therefore, understanding the principles and mechanisms of the hidden layers has become a key objective in deep learning research. These hidden layers play a crucial role in decompressing the input and transforming it into useful knowledge. Visualizing these hidden layers is an effective way to help more people understand the concept of deep learning.

The visualization of CNNs is an underexploited quantitative method to communicate the structure and dynamics of CNNs. Advanced techniques are necessary to allow both researchers and machine learners to see what is learned by deep networks, especially CNNs.

The DVL framework provides guidance for making CNN models more understandable and transparent through various types of visualizations. The diversity of visualization types provides different insights from CNN models. DVL principles also support clear evaluation and potential refinement of accuracy.

Although there are differences in details and specifics, U-Nets share similarities with CNNs: both use convolutional filters to hierarchically extract spatial patterns. Therefore, the benefits of DVL for CNN visualization are likely applicable to U-Nets as well, and DVL's concepts and best practices can further enable intuitive explanations for U-Nets, which will be introduced in the subsequent chapters. For example, combining complementary information like architectural diagrams of the model and prediction overlays can prompt a comprehensive understanding of the results.

Chapter Five

Application of Visualization on FTU Project

5.1 FTU Project Background

The Functional Tissue Units (FTU) project is an integral part of the Human BioMolecular Atlas Program (HuBMAP), aiming to create a comprehensive, open, and computable Human Reference Atlas (HRA) at the cellular level (Godwin et al., 2021; Jain et al., 2023). An FTU is defined as the smallest tissue organization that performs a unique physiological function (Bidanta et al., 2023). It is replicated multiple times within an organ and is pivotal in bridging the gap between the macro-anatomy level of the whole body and the micro-scale of individual cells. The project's primary focus is on capturing the major anatomical structures and cell types of key FTUs, thereby enabling the construction of the HRA.

To utilize and analyze the data captured in the HRA and HuBMAP projects effectively, precise segmentation of FTUs in different types of tissue is essential. Traditional manual segmentation methods are time-consuming and expensive, and they do not scale effectively with the volume of data currently available. Therefore, the development of robust, efficient, and high-performing FTU segmentation models is important. Machine learning algorithms and models provide the tools and methodology for more efficient and accurate FTU segmentation techniques. (Jain et al., 2023)

5.2 FTU Segmentation Methodology

Existing segmentation algorithms often lack accuracy and generalizability. The "Hacking the Kidney" Kaggle competition attracted contributions from over a thousand teams worldwide.

Five Winning algorithms were tested on a large-scale renal glomerulus Periodic acid-Schiff stain dataset and their generalizability was evaluated on a colonic crypts hematoxylin and eosin stain dataset (Jain et al., 2023).

The U-Net (Ronneberger et al., 2015) structure is a key component in the winning algorithms. U-Net is a type of convolutional network that is particularly effective for biomedical image segmentation. It was originally designed for the quick and precise segmentation of images at the pixel level. The U-Net architecture is built on the fully convolutional network and modifies it in a way that allows for more precise localization. The winning teams in the Kaggle competition utilized model architectures based on the U-Net structure, which was combined with different backbones and modules to enhance performance. The key difference between the teams and the original U-Net structure is in the modifications and additional components they integrated into the U-Net structure. (Jain et al., 2023)

For instance, the team "Tom" used a U-Net SeResNext101 architecture, supplemented with a Convolutional Block Attention Module (CBAM), hypercolumns, and deep supervision. The team "Gleb" utilized an ensemble of four 4-fold models, each with an attention decoder. The team "Whats goin on" implemented an ensemble of two sets of 5-fold models using the U-Net architecture with ResNet as backbones, respectively. Additionally, the models from teams "DeepLive.exe" and "Deepflash2" are based on the U-Net structure, with an EfficientNet-B6 and EfficientNet-B2 encoder, respectively. These modifications and enhancements to the U-Net structure have significantly contributed to the performance and accuracy of the FTU segmentation process.

In machine learning, a small dataset can limit the model's ability to generalize to different scenarios, often resulting in overfitting. In such cases, the model may perform well on the training data but not on new, unseen data. Specifically, a model initially trained on a larger dataset, such as the kidney dataset, was adapted to work with the smaller colon dataset using transfer learning. This method leverages knowledge from larger datasets to improve segmentation accuracy on specific, smaller datasets.

5.3 Visualizations of FTU Segmentation

To visualize the segmentation of functional tissue units (FTUs), we developed interactive visualizations using Plotly in Python. These aim to provide an intuitive understanding of both the segmentation model and its results.

The main visualization displays the U-Net architecture used for FTU segmentation. It shows each layer in the encoder and decoder pipelines. We also visualize the output mask, overlaying the predicted FTU segments on an original tissue image. Each FTU has a distinct color, enabling clear visualization of the segmentation.

For each layer of U-Net, we applied dimensionality reduction techniques such as t-SNE (Maaten et al., 2008), UMAP (McInnes et al., 2018), and SVD (Kalman, 1996) to analyze the distribution of sample points. Dimensionality reduction is essential for simplifying multi-dimensional data while retaining its fundamental structure and relationships. This method helps in data visualization and highlights the key features of the data. By using these techniques, the main characteristics of the original data remain in a more understandable format, making it easier for analysis and pattern recognition.

The other visualizations include a mask view visualizing the detailed predictions and a violin plot (Hintze et al., 1998) view showing the accuracy metrics. These visualizations provide both a high-level overview of the U-Net model and a detailed assessment of its performance in FTU segmentation. Different views can be extended to coupled window visualizations that integrate the model architecture, predictions, and accuracy metrics in an integrated interactive view. The subsequent section will discuss user study design to evaluate whether coupled window visualizations enhance the understanding of complex FTU segmentation results compared to individual standalone visualizations.

The methodologies from the Data Visualization Literacy (DVL) paper (Börner et al., 2019) are important in refining the visualization process for U-Net segmentation in the FTU project. The visualization design for the U-Net segmentation reflects key principles and

recommendations from the DVL framework. Specifically, the interactive visualizations use primary types of representations outlined in the DVL typology, such as dimensionality reduction plots to reveal clustering and distributions. The visualization uses positions, colors, sizes, and shapes to map data properties and categories, following best practices for accurate perception based on the literature. Interactivity, aligned with DVL guidelines, allows experienced users to filter, select, and query data points across coupled views of the visualization. Particularly, the use of convolutional layers in the U-Net for the segmentation process gains significant benefits from the application of DVL principles in CNN visualization as introduced in the previous chapters. This helps in creating more intuitive and informative representations of the U-Net model's functionality based on the similarity between CNN and U-Net architectures.

5.4 FTU User Study

This section presents the user study that evaluates the usefulness and user-friendliness of coupled window visualizations in the field of machine learning. As the complexity of machine learning models has increased, it has become challenging for users to improve these models effectively. Visualization tools can help improve the understanding of such models, although their effectiveness has not been thoroughly assessed. This FTU study, therefore, evaluates the usability of coupled window visualizations, particularly among those familiar with machine learning.

In designing the FTU user study, the principles and frameworks from the Data Visualization Literacy (DVL) paper (Börner et al., 2019) played a significant role. The design of the FTU user study reflects principles from the DVL paper to effectively evaluate the utility of coupled window visualizations. The study combines the main concepts from the DVL framework into its hypotheses and research questions. For example, the questions examine whether visual differentiation exists across U-Net layers and views and whether coupled win-

dows aid interpretation and layer optimization. Because of the DVL framework, we were able to construct a user study that not only evaluates the effectiveness of the coupled window visualizations but also assesses how participants interpret and utilize these visualizations in the context of machine learning and FTU segmentation.

The DVL framework guided the development of the user study tasks, ensuring they aligned with key visualization literacy components such as data scales, analysis types, and graphic symbol comprehension. This alignment helped in constructing tasks that effectively measure experienced machine learning users' skills in interpreting complex data from visualizations, particularly in the context of U-Net segmentation. The evaluation compares standalone visualizations and coupled visualizations and evaluates how effectively each type of visualization aids in gaining insights from the data.

5.4.1 Objectives

The primary aim of the FTU user study is to assess the effectiveness of coupled window visualizations in the field of machine learning. Specifically, the study explores how well these visualizations can help experienced users improve their machine learning models. To achieve these objectives, 102 student subjects were recruited for the study, divided equally into control and experimental groups, with each group containing approximately 51 members.

Research Questions:

A. Will the visualization of features of colon FTU look different between the FTU, the edge FTU, and the non-FTU patches in different views? *("FTU" indicates that the patch includes the FTU segmentation mask; "edge FTU" indicates that the patch contains only a very small portion (<10% of the size) of the mask; "non-FTU" indicates that the patch does not contain any part of the mask)*

B. Will the use of coupled window visualization help to effectively and clearly display the differences being discussed to the reader?

C. Can the user select the optimal number of layers for the U-Net to improve machine

learning accuracy, despite the differences between the visualizations of segmentation?

D. Will the utilization of coupled window visualization assist the user in selecting the optimal number of layers for the U-Net to enhance machine learning accuracy?

Hypotheses: A. The visual representation of the functional tissue unit (FTU), the edge FTU, and the non-FTU patches should exhibit different distributions of clustering in either the mask view or any dimensionality reduction view. This differentiation will enable the identification of the origins of the differences in the segmentations, specifically the specific layer of the U-Net.

B. Coupled window visualization should be able to present these differences more clearly and intuitively, and also assist the user in completing tasks based on visualization settings more accurately.

C. An examination of the variations in the visualization of the segmentations can facilitate the determination of the optimal layers for fine-tuning the segmentation U-Net, thus enhancing the accuracy of machine learning.

D. Coupled window visualization should be able to display more information simultaneously in a clearer and more intuitive way and assist the user in more accurately determining how to optimize the freezing of specific layers in the U-Net.

5.4.2 Methodology

Participants

The study recruited 102 student participants for the user study described in this chapter. The sample size estimation was conducted using G*Power, a statistical software commonly utilized for research studies (Faul et al., 2007). To ensure equal representation, participants were evenly distributed between the control and experimental groups. Fifty-one student subjects were assigned to each of the control and experimental groups.

The chosen statistical test for this study was the t-tests (Student, 1908), which is com-

monly used inferential statistical methods to determine the difference between the means of two groups. The effect size was set at 0.5, which indicated a moderate effect size between the means of the two independent groups. The selection of the effect size was essential as it serves as a measure of the magnitude of the relationship between the two variables.

The alpha level was set at 0.05, a standard threshold for determining statistical significance, suggesting a 5% probability that the results could be due to chance. Statistical power, defined as the probability of correctly rejecting the null hypothesis when it is false, was chosen at the conventional value of 0.8 for this study. A statistical power of 0.8 implies that there is an 80% chance that the study can detect a difference between the means of the two groups if one genuinely exists.

The allocation ratio N1/N2 was 1, indicating an equal number of participants in both the control and experimental groups. According to these parameters, a minimum of 51 individuals per group was required to detect a moderate difference in means between the two groups with an 80% probability and a significance level of 0.05.

The study design aligned with ethical considerations and followed strict protocols to ensure the well-being and privacy of all participants. Moreover, the user studies were approved through the Indiana University Institutional Review Board (IRB) application process.

In the FTU study, we aimed to recruit expert users or experienced users to participate in the user studies. Expert users, with considerable machine learning-related experience and a deep understanding of related processes, are expected to view the coupled window visualization to optimize the machine learning process. The user study aimed to assess the effectiveness of the visualization for machine learning expert users or experienced users.

User study pipeline

In the FTU user study, two types of visualizations were employed: simple standalone visualizations and coupled window visualizations. The simple visualizations comprised machine learning predictions using different methods and displaying statistics and distributions ob-

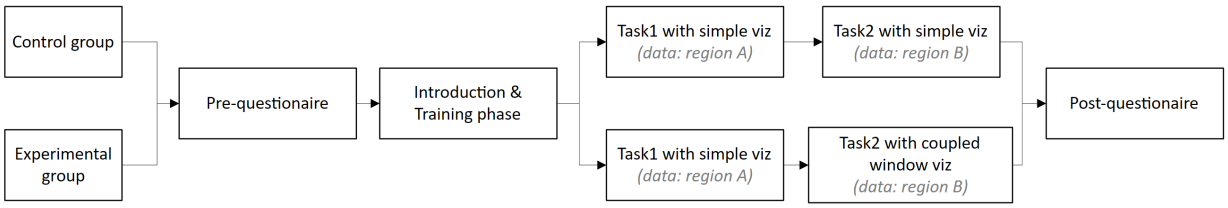


Figure 5.1 User Study Pipeline

tained from these predictions. The coupled window visualizations combined different types of visualizations in a single view, allowing users to link panels and navigate between visualizations based on the connections seamlessly.

The simple visualization included the U-Net structure visualization, mask view, and violin plot. The coupled window visualization combined all three in one view. The U-Net visualization shows the U-Net architecture, including encoder, decoder, upsample pipelines, and final pipeline. The mask view displayed all predicted masks. The violin plot showed the distribution of dice scores obtained from the FTU segmentation.

User studies were conducted to evaluate how participants interact with the visualizations, their reasoning to answer questions based on the visualizations, and how the responses are affected by different types of visualizations. The study involved two tasks (Task 1 and Task 2), and two groups of participants (control and experimental groups), which helped analyze and assess the impact of coupled window visualization on user performance.

To ensure the results from the study reflect the population, demographic information was collected from each participant in both groups (refer to the "User study materials" section in the Appendix Chapter). During the introduction and training phases, participants were guided to become familiar with the task requirements and how it should be completed step by step. Sample questions were provided after every session, with the correct answers included to ensure that they fully understood the instructions.

In Task 1, both the control and experimental groups were presented with simple visualizations and the following relevant questions. For Task 2, the control group continued to

use simple visualization, while the experimental group switched to using coupled window visualization. This setting was designed to evaluate two primary comparisons: (1) To compare the impact of coupled window visualization on the experimental group's performance in Task 2 by comparing it to their earlier performance with simple visualization in Task 1, and (2) To determine the effect of coupled window visualization on users' performance by performing a cross-comparison between the control and experimental groups in both tasks. The combined results from these two comparisons indicated the impact of coupled window visualization on the users' performance.

In addition to the two explicit comparisons outlined above, there were also two implicit comparisons. The first was a comparison for the control group between Task 1 and Task 2 to assess whether they achieved similar performance as the baseline. Since Task 1 and Task 2 involved similar tasks and questions and used simple visualizations, we expected their performance to remain consistent in both tasks. The second implicit comparison was a cross-comparison of Task 1 performance between the control and experimental groups to determine whether they achieved similar performance as the baseline. Both groups were presented with the same simple visualizations in Task 1, and we expected their performance to be similar.

After completing each task, participants from both groups completed post-task questionnaires to help analyze their understanding of the task and their levels of satisfaction. In addition, the experimental group was asked about their familiarity with the coupled window visualization. On the other hand, since the control group was not exposed to the coupled window visualization, they were introduced to the concept of coupled window visualization through text. They were given guidance to compare their experience with the simple visualization they had seen and were asked to provide their insights and suggestions for the improvement of the visualization based on their expectations and the textual description.

Visualization description

The control group was presented with a set of visualizations comprising three distinct types. The first visualization, the structure view (Figure 5.2), displayed data points from each layer of the U-Net machine learning model during the segmentation of functional tissue units (FTUs) in colon samples. This visualization also simultaneously displayed the overall structure view of the U-Net. The second visualization was a violin plot (Figure 5.3) that illustrated the accuracy distribution of all FTU segmentations using the Dice coefficient and other related metrics. The third visualization was the mask view (Figure 5.4) of the segmentation achieved by the U-Net machine learning model. The experimental group was shown a coupled window visualization (Figure 5.5) that integrates all three types of visualizations.

The primary structure view (Figure 5.2), also the main view in the coupled window, showed the input layers of the U-Net, ranging from encoder0 to encoder4, and the output layers from decoder4 to decoder0, including the intermediate layers. Each decoder was paired with its corresponding upsample layer. The final layers, labeled final0 to final2, represented the ultimate output layers of the model. Within each layer, 288 data points were displayed, each representing a segmentation sample from colon data. Every colon sample image was divided into 288 small sections, thus producing 288 data points. These data points were processed and analyzed using dimensionality reduction methods such as SVD and t-SNE. Figure 5.2 below shows the results using the SVD method. The 288 data points were labeled as FTU (blue), non-FTU (red), and FTU edge (white). These labels represented areas with FTU segmentation, areas without FTU segmentation, and edges of areas with minimal FTU segmentation (less than 5% of the segmentation area), respectively. These labels could help visualize the distribution of data points across the U-Net layers. Alongside the visualization for each layer, the Calinski-Harabasz score was provided to assess the uniformity within clusters.

The violin plot (Figure 5.3) provided a different view, highlighting the accuracy of seg-

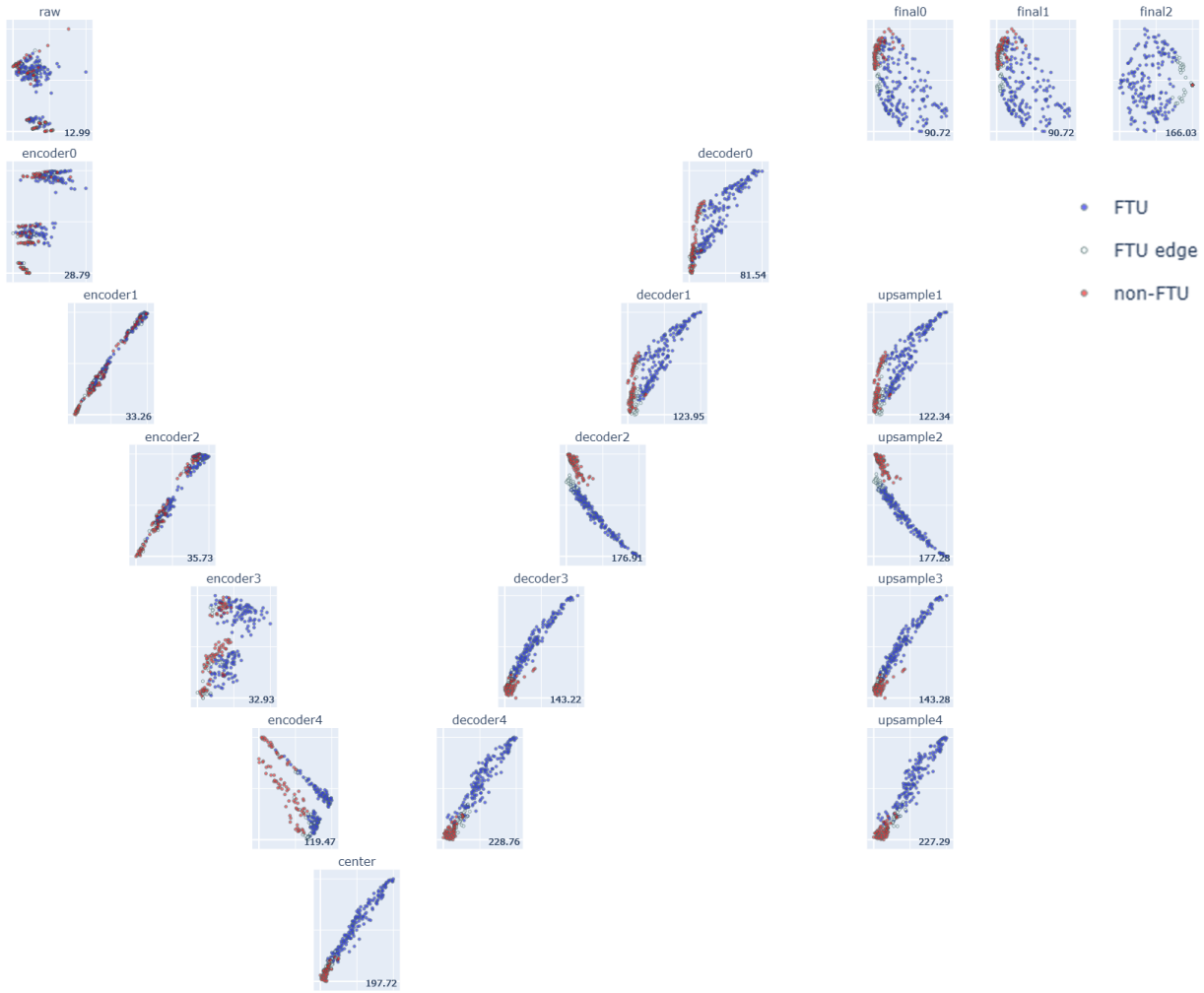


Figure 5.2 Interactive Visualization - U-Net Structure View

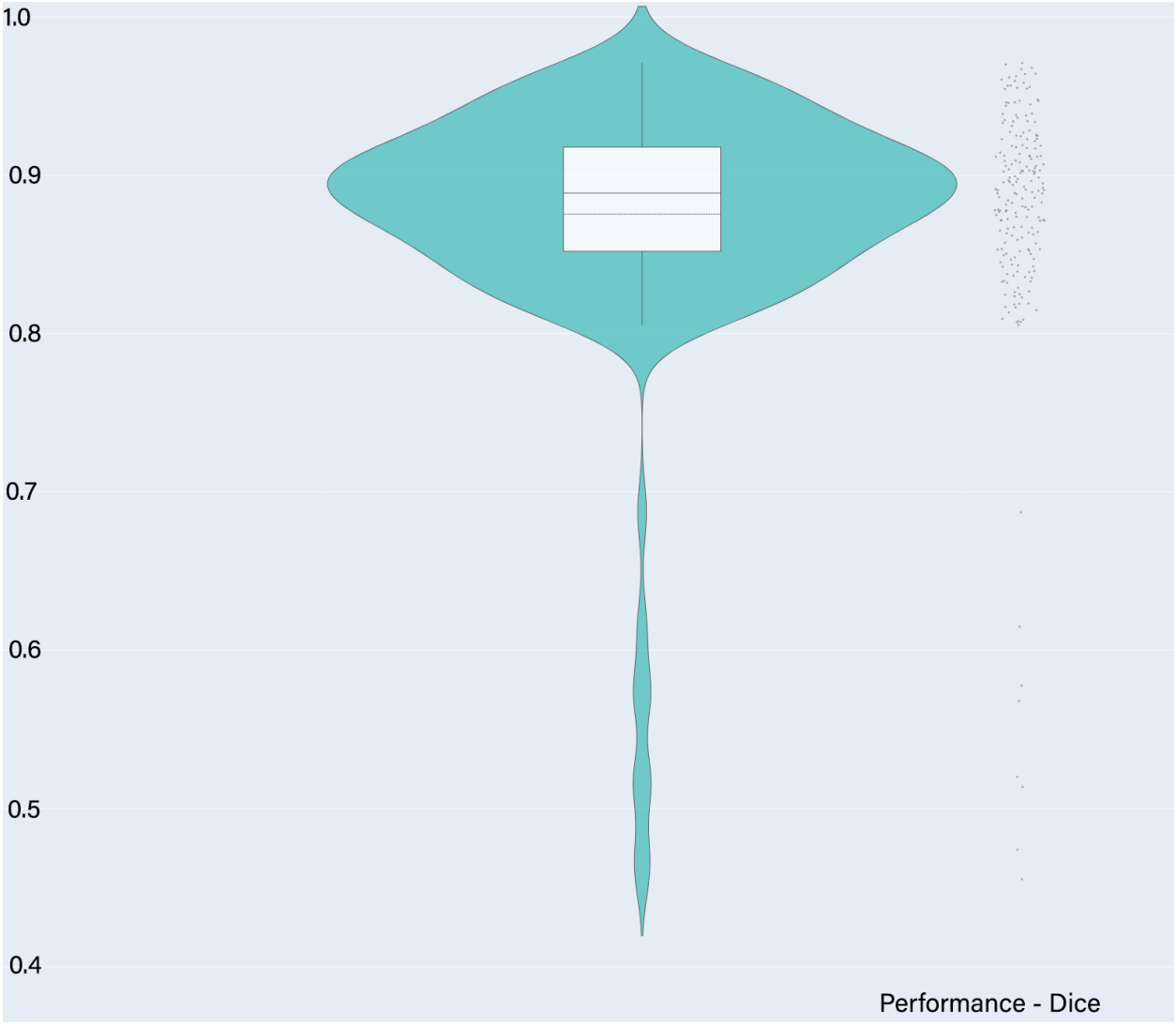


Figure 5.3 Interactive Visualization - Violin Plot

mentation results using the Dice coefficient as the measure. Unlike histograms, the violin plot displayed the distribution of segmentation accuracy for different FTUs. It combined elements of both box plots and density plots, showing the likelihood of data at specific accuracy levels. This format allows for a direct comparison with the actual data points. An interactive feature was included, to enable immediate display of relevant information when a part of the violin plot was hovered over.

The mask view (Figure 5.4) provided an intuitive visualization of the segmentation masks for different segmented sections of each colon sample. Each colon sample image was parti-

tioned into 288 sections, with each section overlapping by 50% on both the x and y axes with its adjacent sections. In this visualization, the background was represented in black, while the segmented areas were in white.

The coupled window visualization (Figure 5.5) contained all the visualizations mentioned above, displaying them side by side in different panels. Beyond the interactions described within each visualization, the coupled window also supported interconnections between the various visualizations, allowing for more than just a simple side-by-side display by linking shared data. For instance, when a data point (representing a particular segmentation region) in the structure view was selected, its corresponding element in the violin plot was highlighted. Simultaneously, the related area in the mask view was also highlighted, making it easily identifiable by users. Selecting another area resulted in the highlighting of new corresponding areas, while previously selected areas reverted to their original state.

5.4.3 Results

In this section, we present the results of the FTU user study. Users in both groups completed tasks by answering questions based on different combinations of visualizations, and their accuracy was recorded for further analysis.

Task analysis

Questions 1 to 5 in the user study were designed to explore the users' ability to discern differences in various visual aspects, ranging from distinguishing mask view differences to understanding U-Net view distribution differences and identifying accuracy differences in violin views. These questions aimed to understand the users' perception of the details presented in different visualizations. The focus was on identifying both the small differences and similarities across various layers and views.

Initially, questions 4 to 5 focused on analyzing users' abilities to notice differences between various elements such as masks, U-Net views, and violin views. A careful review of

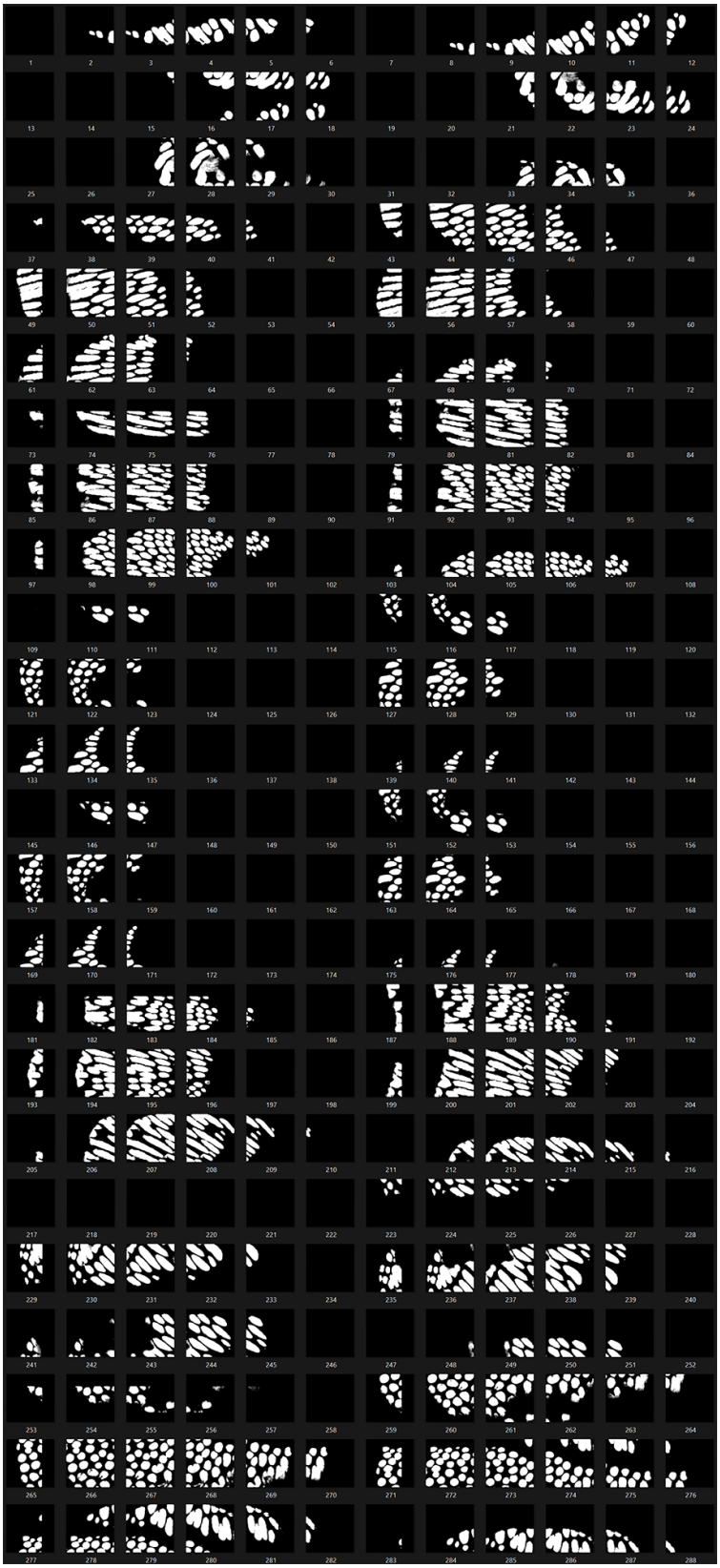


Figure 5.4 Interactive Visualization - Mask Plot

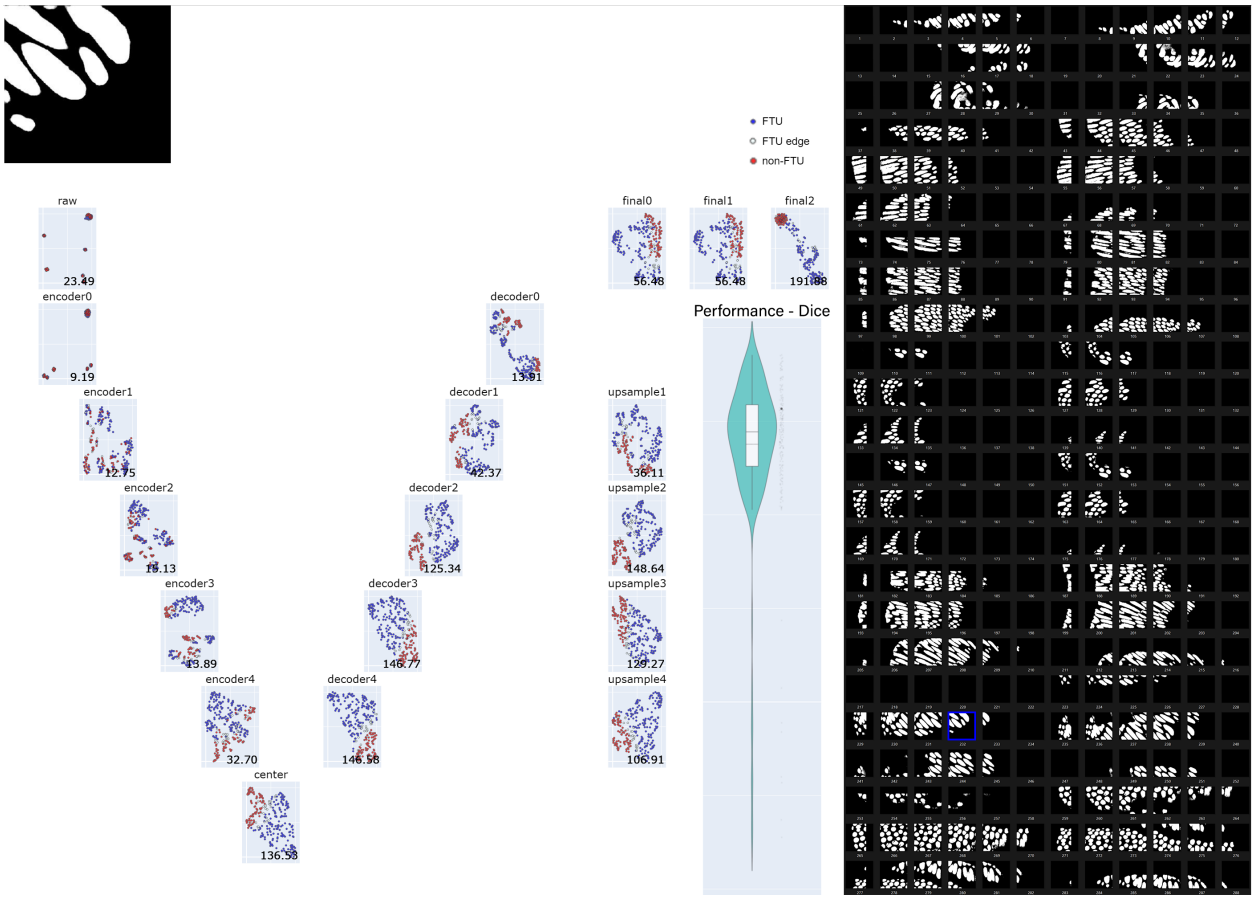


Figure 5.5 Interactive Visualization - Coupled Window Visualization

the responses from Task 1 for both the control and experimental groups showed a shared preference for the option "Somewhat different"; 55% of the control group and 52% of the experimental group selected this choice for question 1. In question 2, 45% of the control group and 48% of the experimental group picked this option. For the other questions, most participants chose the "Somewhat different" option. This preference for "Somewhat different" indicated that participants were able to notice small differences while observing the complicated structures, thereby supporting the hypothesis that both groups began with a similar level of understanding and perception. This data showed a general agreement between the two groups, demonstrating a similar level of understanding, which was essential to note the small differences in the responses to the following Task 2.

In Task 2, the questions remained consistent, aiming to gauge the users' ability to discern differences in various visual aspects presented in the different visualization techniques. The control group continued with the simple visualization method, while the experimental group was introduced to the coupled window visualization. A preliminary analysis of the responses indicates a sustained preference for the "Somewhat different" option in both groups, albeit with a slight variation in the distribution of choices. For instance, in question 1, 53% of the control group and 50% of the experimental group opted for "Somewhat different". This trend was mirrored in the responses to the subsequent questions, with a majority leaning towards the "Somewhat different" option, indicating a consistent approach to discerning differences, albeit with a nuanced understanding of the visualizations presented.

Questions 6 and 7 in the user study were centered around the concept of transfer learning, specifically the transfer of knowledge from a large dataset (kidney) to a smaller one (colon). The questions aimed to gauge the participants' understanding of which layers should be frozen during the transfer learning process and whether the visualizations provided any hints or insights into this decision. Question 6 is: we tried to apply transfer learning to transfer the knowledge learned from the source dataset (kidney, large dataset) to the target dataset (colon, small dataset). In the transfer learning process, will you suggest keeping

	Task 1		Task 2	
	Q6	Q7	Q6	Q7
Control Group	74.51%	52.94%	82.35%	56.86%
Experimental Group	79.25%	54.72%	92.45%	69.81%

Table 5.1 Comparison of Positive Response in Task 1 and Task 2

some layers frozen and only train the rest layers? Question 7 is: From the given information from the visualization, do you think if you could get some hint which layers should be frozen? The results of Questions 6 and 7 in both Task 1 and Task 2 are summarized in Table 5.1 and Figure 5.6. The data in the table shows the percentage of users who chose a positive response to Questions 6 and 7 in Tasks 1 and 2, respectively, while the remaining users selected negative answers. Table 5.1 presents the positive response rates for both the control and experimental groups across Tasks 1 and 2, detailing the results for Questions 6 and 7 in each task. Statistical analysis was done on the objectives of the users, as well as subjective assessments of their performance based on the layer choices.

Table 5.1 presents the positive response rates for both the control and experimental groups across Tasks 1 and 2. The results for Questions 6 and 7 are detailed for each task.

For Task 1, both groups were exposed to simple visualizations. The control group registered a positive response rate of 74.51% for Question 6 and 52.94% for Question 7. On the other hand, the experimental group had slightly higher positive response rates of 79.25% for Question 6 and 54.72% for Question 7. The similarity in performance between the two groups for Task 1 was expected, given that they were exposed to the same type of visualization.

Transitioning to Task 2, where the control group continued with simple visualizations and the experimental group was introduced to the coupled window visualization, a noticeable difference in performance was observed, with 82.35% selecting positive answers for Question 6 and 56.86% for Question 7. In contrast, the experimental group's performance significantly improved with 92.45% of participants selecting positive answers for Q6 and 69.81% for Ques-

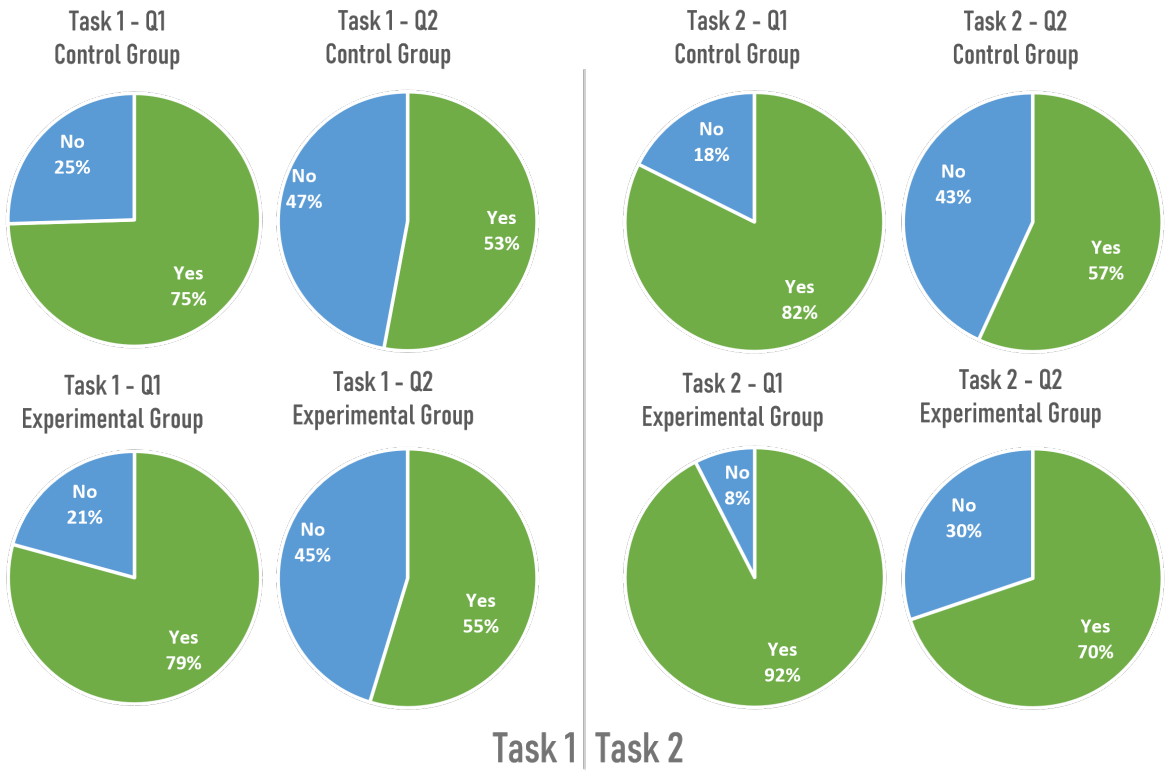


Figure 5.6 Comparison of Positive Response in Task 1 & Task 2

tion 7. Comparing the results of Task 2 between the control and experimental groups, it is clear that the percentage of participants selecting positive answers for Question 7 was significantly higher for the experimental group.

Statistical analysis was used to compare the differences between the control and experimental groups across the two tasks. A t-test showed significant differences in positive response rates for Questions 6 and 7 between the groups in Task 2. For Question 6, the result was $t=3.68$ with $p<0.01$, and for Question 7, it was $t=3.10$ with $p<0.01$. These low p-values indicate that the differences were significant and not random. This suggests that the coupled window visualization effectively helped expert users understand the concept of layer freezing in U-Net.

On the other hand, for Task 1, the differences between the two groups were not significant. For Question 6, the results were $t=1.40$ and $p>0.05$, and for Question 7, they were $t=0.77$ and $p>0.05$. This consistent performance highlights the fair selection of participants as a baseline and that emphasized the differences in Task 2.

Questions 8 to 11 were a question group related to transfer learning, asking which layers participants would suggest freezing in initial experiments. For each question, participants chose from options referring to layers of a neural network model, including "Top layers", "Mid layers", "Bottom layers", and combinations thereof. The choices remained the same for all four questions, with the only difference being the specified type of layer was specified in: encoder, decoder, upsample, or final layers. All questions used the same answer structure, highlighting individual layer groups or combinations.

For this group of questions, the control group's responses remained stable from Task 1 to Task 2, providing an experimental baseline. Their consistent performance aligns with the expectation that performance should be consistent with the same visualizations. Additionally, cross-group comparison for Task 1, both groups had very similar answer distributions, aligned with the hypothesis that their initial performance would be comparable given identical simple visualizations.

In the between-group comparison, the experimental group's responses shifted towards selecting "Bottom layers" in Task 2 compared to Task 1, with 41% selecting it in Task 2 versus 31% in Task 1 for Question 9. Meanwhile, the control group's responses remained relatively stable from Task 1 to Task 2. For Question 10, the experimental group's selection of "Mid layers" increased from 14% to 33%, while the control group's choices changed from minimally from 29% to 27%.

These results demonstrate that the experimental group's selections noticeably changed after introducing coupled window visualizations in Task 2, while the control group's choices remained consistent when repeating simple visualizations. This suggests that the coupled window visualization had a significant impact on the experimental group's responses. The shifts in the experimental group's selections from Task 1 to Task 2 align well with the expected results of the experiment.

Post questionnaire analysis

In the post-questionnaire, both the control and experimental groups were asked two questions about the ease of use and helpfulness of the visualizations in the tasks. The first question was similar for both groups: **"Would you say that the visualizations in the task were easy to use?"**. The control group was asked about simple visualizations, and the experimental group was asked about coupled window visualizations. The second question was different. For the experimental group, the second question asked **"Would you say that compared to the visualizations in Task 1, coupled window visualizations in Task 2 help you more to complete the task?"**. Since the control group had not experienced the coupled window visualization, their second question first described via detailed text what coupled window visualization looks like, and then asked them **"Would you say that compared to the visualizations in Task 1 and 2, the coupled window visualizations will be helpful to complete the task?"**. These questions were designed to assess the effectiveness of the coupled window visualizations and user satisfaction compared to the

		Q1	Q2
Control Group	Agree	62.75%	74.51%
	Neutral	21.57%	21.57%
	Disagree	15.69%	3.92%
Experimental Group	Agree	71.70%	79.25%
	Neutral	16.98%	15.09%
	Disagree	11.32%	5.66%

Table 5.2 Comparison of Post Questions

simple visualizations used in Task 1.

The data in Table 5.2 represent the percentages of users who selected positive, neutral, and negative responses to Questions 1 and 2 in the post-questionnaire. For the first question, "Would you say that the visualizations in the task were easy to use?", the experimental group had a higher proportion of participants who agreed (71.7%) that the coupled window visualizations were easy to use, compared to the control group (62.7%). This suggests that the coupled window visualizations in Task 2 may have been easier to use compared to the visualizations in Task 1 for the experimental group. This result aligns with the higher accuracy and faster completion rates observed in the experimental group, indicating that the ease of use of the coupled window visualizations may have contributed to the higher accuracy and speed of task completion for the experimental group.

For the second question, it is important to note that the question was different for the control and experimental groups. The control group was asked, "Would you say that compared to the visualizations in tasks 1 and 2, the coupled window visualizations will be helpful to complete the task?" while the experimental group was asked, "Would you say that compared to the visualizations in Task 1, coupled window visualizations in Task 2 help you more to complete the task?" The experimental group had a higher proportion of participants who agreed (79.2%) that the coupled window visualizations in Task 2 were more helpful

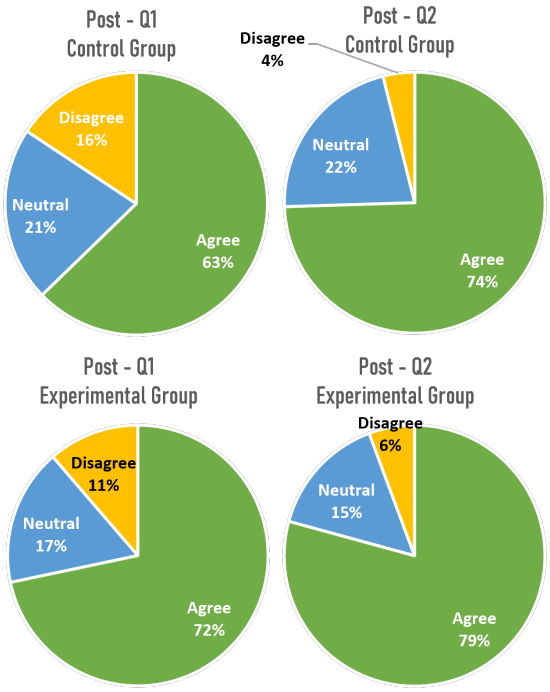


Figure 5.7 Comparison of Post Questions

compared to the visualizations in Task 1, compared to the control group (74.5%) who were asked if the coupled window visualizations would be helpful compared to the visualizations in both tasks 1 and 2. This suggests that the coupled window visualizations in Task 2 were perceived as more helpful compared to the simple visualizations in Task 1, particularly for the experimental group.

Finally, there are three subjective questions that provide feedback on the user's opinion and satisfaction with the different visualizations.

What did you like about the visualizations?

Both control and experimental groups appreciated the interactive and informative nature of the visualizations. The use of colors and clarity in the visuals was praised by both groups. Both groups found it useful and beneficial to have all visualizations in one place, rather than switching tabs to see the differences. The ability to interact with different charts and perform comparative analysis was noted as a positive feature by both groups.

The main difference between the groups was that the experimental group found the controls and legend intuitive and easy to use. The experimental group also appreciated the inclusion of detailed explanations for the visualizations, with several noting the relevance of the visuals to their fields of interest (such as computer vision). The control group gave mixed feedback about the mask visualizations, with some finding them confusing while others appreciated the unique perspective they provided. Finally, the experimental group particularly liked viewing all the visualizations together in a coupled window visualization. Participants highlighted the ease of comparative analysis and the convenience of having all relevant information in one place, which provided a deeper understanding of the data and the model. They found the integrated, coupled window view particularly beneficial for tracking specific data points across different layers and visualizations, enhancing their ability to draw meaningful insights.

What did you not like about the visualizations?

The most commonly mentioned issue among the control group was the complexity of the visualizations, with some participants feeling overwhelmed by the amount of information presented. Some also found the mask visualizations confusing or difficult to understand. The experimental group did not report any major dislikes about the visualizations but suggested some improvements, such as better UI design and clearer explanations of the task at hand. Finally, both groups emphasized the importance of clear instructions to help users effectively navigate the visualizations.

What can be done to improve the visualizations? / What can be done to improve the coupled window visualizations?

Both the control and experimental groups had similar suggestions for improving the visualizations, including the provision of more information and details, greater interactivity, and enhanced legibility and comprehensibility through simplified visuals or added context.

Specifically, the control group recommended adding more graphs, better descriptions for the mask visualization, meta-information, improved visual naming and explanations, more

side-by-side views, instructions, visual examples, and enhanced mask view features. The experimental group suggested more creative and visually appealing UI designs, improved text resolution and orientation, zoom functions, clearer task descriptions, color-coding, graph arrangement, additional visualizations, reduced response times, connected visualizations, context provision, reduced loading times, simplified model metrics, and mobile-friendly designs.

Overall, both groups recommended improving the visualizations by adding more details, context, and interactivity. The control group's suggestions were primarily focused on improvements specific to visualizations, while the experimental group proposed more general improvements to the UI and clearer task explanations. Both groups expressed appreciation for the interactive and informative nature of the visualizations, the use of colors, the clarity of the visuals, and having all the visualizations in one place. The experimental group found the controls and legend intuitive and easy to use, and the detailed explanations of the visualizations. The control group had mixed reactions to the mask visualizations. Suggestions for improvement included better UI design, clearer explanations, including legends or color coding, and providing clear instructions.

The results of this post-study analysis indicated that users perceived the coupled window visualization as helpful in completing the task, which was consistent with the study's objectives. The findings also suggested that the coupled window visualization might be more effective than the simple types of visualizations in improving task performance for expert users.

One notable observation regarding Question 1 was that participants from both groups, on average, rated the visualizations as easy to use. This suggested the importance of designing intuitive visualizations that users can comprehend and interact smoothly, as this can improve task completion, regardless of the types of visualization employed. The experimental group, however, showed a slightly higher agreement rate than the control group for both tasks. This might indicate that the coupled window visualization design, which combined information

from multiple visualizations and required users to make connections between them, was potentially more intuitive for expert users.

Regarding Question 2, the results suggested that users had a positive attitude towards the coupled window visualization, regardless of whether they had experienced it or not. Alternatively, it could suggest that the feedback from the control group was uninfluenced by actually experiencing the coupled window visualization, highlighting the potential benefits of user feedback on visualization designs they have not yet encountered.

A comparison of the responses to Questions Question 1 and 2 between the control and experimental groups suggested that the coupled window visualization designs might improve the understanding of the machine learning model for expert users. The results aligned with expectations, indicating that more complex visualizations could be more effective in communicating multidimensional data to users. Overall, this study's findings suggested the potential utility of coupled window visualization designs in improving experts' performance on complex tasks.

5.5 Layer Freezing Experiment

The results from the user study demonstrated the potential of coupled window visualizations to aid expert users in understanding and optimizing machine learning models, specifically the U-Net, by improving feature visualization, particularly concerning layer freezing. Encouraged by the positive outcomes of this user study, the Layer Freezing Experiment in this section aimed to empirically assess the impact of layer freezing on the performance of U-Net models in the context of medical image segmentation.

This experiment involved training U-Net models by freezing different layers and evaluating segmentation accuracy and computational efficiency. By exploring the effects of layer freezing on U-Net performance, we could provide valuable guidance to machine learning experts and researchers on optimizing U-Net configurations for specific image segmentation

tasks, especially for medical image segmentation tasks whose datasets usually have limited sizes.

Overall, the user study and its positive findings provided a solid foundation for the Layer Freezing Experiment, highlighting the efficiency of coupled window visualizations in improving the optimization of machine learning models.

5.5.1 Methodology

Visualization Setup

The primary objective of this visualization setup was to provide a comprehensive and intuitive representation of U-Net's layers, particularly to identify which layers to freeze. To achieve this, we used the Plotly library to generate interactive visualizations. The data for this visualization was sourced from the embedding information of U-Net.

The visualizations were organized in a grid format, as shown in Figure 5.8, with each cell representing a specific layer of the U-Net. The positioning and layout of these layers were carefully designed to ensure that the visual representation closely reflected the actual architecture of U-Net. For better visualization and dimensionality reduction, techniques such as t-SNE, UMAP, and SVD were employed. The choice of dimensionality reduction techniques could be easily toggled between each other, enhancing flexibility in the visualizations.

We employed a color-coding system to clarify data points, distinguishing between FTU, FTU edge, and non-FTU points. Additionally, we calculated and displayed metrics such as the silhouette score (Rousseeuw, 1987), Calinski-Harabasz score (Caliski et al., 1974), and Davies-Bouldin score (Davies et al., 1979) were computed and annotated on the visualizations. These metrics provided quantitative insights into the clustering quality of the data points for each layer. The silhouette score measures how well samples are classified within their clusters based on intra-cluster tightness and inter-cluster separation (Rousseeuw, 1987). The Calinski-Harabasz score assesses dispersion between and within clusters, with higher

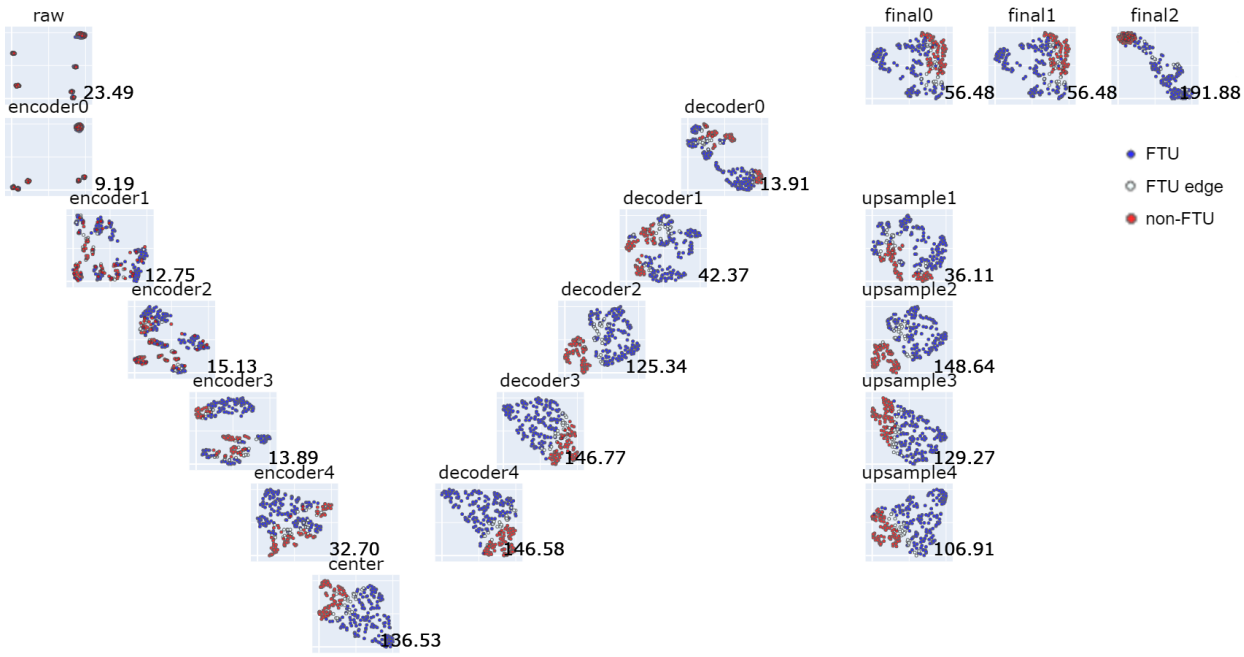


Figure 5.8 Experiment Visualization: U-Net Structure Overview

scores implying better-defined clusters (Caliski et al., 1974). The Davies-Bouldin score evaluates clusters based on scatter within clusters and separation between cluster centroids, with lower scores suggesting better clustering (Davies et al., 1979).

In summary, the visualization setup was designed to provide both qualitative and quantitative insights into the behavior of U-Net's layers, especially when subjected to layer freezing conditions.

U-Net Layer Freezing Experiment Setup

This experiment aimed to determine the impact of freezing different layers within the U-Net architecture on transfer learning performance, transitioning from a kidney to a colon dataset. The goal was to identify which U-Net components most significantly contribute to accuracy when selectively frozen during transfer learning.

The U-Net model contains an encoder path that extracts features and a decoder path that reconstructs the image from these features. The encoder includes convolutional layers paired with max pooling to capture contextual details, where convolutional layers capture

feature representations from the input images, while max pooling reduces spatial dimensions to help the network learn higher-level features (LeCun et al., 1998). The decoder uses transposed convolutions to upsample these features and reconstruct the input image to its original dimensions (Dumoulin et al., 2016).

To explore the effects of freezing, we configured multiple U-Net models by selectively freezing layers in encoders (encoder0, encoder1, etc.), decoders (decoder0, decoder1, etc.), the center layer, deep supervision layers (deep1, deep2, etc.), and the final convolution layer, in various combinations. Freezing these layers prevented updates to their weights during training.

We trained these U-Net configurations on colon tissue image patches extracted from the HuBMAP dataset. The models were trained for 50 epochs using the Adam optimizer (an adaptive learning rate optimization algorithm commonly used for training deep neural networks (Kingma et al., 2014)) with a learning rate of 0.0001 and early stopping based on validation loss. The loss function combined Binary Cross-Entropy (BCE) and Lovasz hinge loss. BCE measures probability error for binary predictions (Zijun Zhang, 2018), while Lovasz hinge loss optimizes mean intersection over union (Berman et al., 2018). We evaluated segmentation accuracy using the Dice similarity coefficient between predicted and ground truth masks. Dice coefficient measures overlap between two samples, with higher values, indicating greater similarity (Sorensen, 1948). All experiments were conducted 10 times, and the average of the results was used to minimize any offsets.

By comparing the validation Dice scores, we identified which layer configurations achieved the highest performance. This revealed which components of the U-Net architecture retained more valuable information from the kidney dataset and were effectively applicable to the colon dataset. Finally, we compared these objective experimental outcomes with subjective user feedback from the visualization-based user study. This analysis provided insights into the correlation between the user study and empirical results from the layer freezing experiments.

5.5.2 Experiment Results

Visualization Insight

The interactive visualizations generated from U-Net layer data provided valuable qualitative insights into the effects of layer freezing. Examining the UMAP and SVD dimensionality reduction views of the encoder, decoder, and other layers showed noticeable differences in the distribution and clustering of FTU, FTU edge, and non-FTU data points.

Specifically, the distribution and separation of these categories varied across layers. Some layers, particularly the bottom encoder and decoder layers, exhibited tightly defined clusters, suggesting effective feature extraction. In contrast, the top encoder layers showed greater diffusion, indicating less clear class discrimination.

Beyond the distribution, the embedded cluster evaluation metrics also differed significantly across layers. Certain layers like the center layer or the bottom layers of upsample layers achieved higher Calinski-Harabasz scores, which indicated effective intra-cluster similarity and inter-cluster differentiation. In contrast, layers such as the top layer of encoder layers showed poorer metric values, implying less-than-ideal clustering.

These visual distinctions provided clues and hints about the relative importance of different layers for U-Net's segmentation capabilities. The tightly clustered layers with high evaluation metric scores seemed to extract discriminative features. Freezing such layers could preserve these valuable features. In contrast, the poorly clustered layers appeared less informative, suggesting that it might be beneficial to continue updating less informative layers during training.

In summary, the interactive visualizations not only offered qualitative insights into the U-Net's layers but also guided decisions regarding layer freezing to optimize performance. The next step will be to validate whether these visual trends translate into improved quantitative outcomes in experimental results.

Experiment Result

The U-Net models underwent transfer learning from the kidney dataset to the colon functional tissue unit dataset, using various layer freezing configurations as described in the methodology. The segmentation performance of the models was evaluated using the Dice similarity coefficient between the predicted masks and ground truth masks on the independent test set. The model with no frozen layers achieved a baseline mean Dice score of 0.875 (std=0.0032) on the test set.

Freezing the bottom encoder layers resulted in an improved Dice score of 0.888 (std=0.0030), compared to the baseline model. This suggests that retaining the pre-trained representations in the bottom encoder layers and early decoder layers enhances segmentation accuracy. In contrast, freezing only the top decoder layers led to a significantly lower Dice score of 0.844 (std=0.0041). Additionally, we experimented with freezing the entire encoding or decoding pipelines. Freezing all encoder layers decreased the Dice score to 0.848 (std=0.0025) compared to the baseline. In contrast, freezing the full decoding pipeline yielded a Dice score of 0.874 (std=0.0033), similar to that of the baseline model. Further freezing the bottom center layer along with the decoder pipeline slightly increased the score to 0.879 (std=0.0032).

In terms of model efficiency, freezing layers reduced the number of trainable parameters. The model with the decoder frozen had the fastest training time of 2.05 minutes per epoch, while the baseline model took 3.15 minutes per epoch.

In summary, the controlled experiments revealed that selectively freezing layers, especially the bottom encoders, could enhance U-Net performance for the colon segmentation task. Next, these objective results will be compared with the subjective opinions from the earlier visualization-based user study.

Comparison

The interactive visualizations provided qualitative insights that aligned well with the quantitative experimental outcomes, validating the utility of coupled window visualizations. Specifically, the visualizations indicated that the bottom encoder layers had tightly clustered classifications, while the top encoder layers were more diffuse. Similar to this trend, the experiment found that freezing bottom encoder layers gave the optimal Dice score, while freezing top encoders decreased performance. This alignment highlights the value of visualization-guided intuition.

Additionally, the FTU user study revealed changes in the experimental group's layer freezing preferences after being exposed to coupled window visualizations. For the encoder layers in Questions 8-11, the experimental group increased its selection of "Bottom layers" from 31% to 41% between Task 1 and Task 2. Correspondingly, the experiment confirmed that freezing bottom encoder layers led to the best segmentation accuracy, further validating the impact of the coupled visualizations.

However, some divergences also emerged between the user study and the results of the experiment. While most users in the user study selected freezing bottom-decoder layers, the second most users selected mid-decoder layers. The experiment found that freezing mid-layers (either decoder, encoder, or both) performed similarly to not freezing at all. This indicated potential limitations of current visualizations based on the embedding information from the model.

Experiment Summary

The Layer Freezing Experiment, along with the user study, demonstrated the potential of layer freezing as an optimization strategy for U-Net models in the following aspects.

Relevance of Visualization: Coupled window visualizations played an important role in guiding participants, especially experts in the machine learning field, towards optimal layer

freezing decisions and approximating the optimal freezing configuration in future experiments. There is a need for intuitive visualization tools in the field of machine learning, especially when dealing with complex models like U-Net.

Optimization through Freezing: The experiment supported the hypothesis that strategic layer freezing not only improves accuracy but also makes the training process computationally efficient and allows the model to adapt to more datasets (e.g., transitioning from the kidney dataset to the colon dataset and potentially to other organ datasets) in transfer learning.

User Study: The success of the Layer Freezing Experiment could largely be attributed to the foundational insights derived from the visualization and user study. The clear preference of the experimental group in the user study for freezing certain layers was reflected in the actual performance benefits observed in the experiment.

5.6 Discussions

The FTU user study was designed to evaluate the potential of coupled window visualization in the domain of machine learning. This section integrates the findings from the study by correlating the research questions, hypotheses, and the results presented.

5.6.1 Visual Differentiation and U-Net Layers

The first research question (A) and its corresponding hypothesis (A) concern the visualization of features of colon FTU. The study's findings confirm that there are distinct visual differences between FTU, FTU edge, and non-FTU patches across various views. Specifically, each layer of the U-Net machine learning model displays 288 data points, representing segmented sections of colon sample images. These data points are categorized into three distinct groups: FTU (blue), non-FTU (red), and FTU edge (white), with the FTU edge meaning regions of minimal FTU segmentation, accounting for less than 5% of the segmen-

tation area. This categorization enables the observation of the distribution of data points from different regions across all U-Net layers. Hypothesis A's assertion that these visual representations should exhibit different clustering distributions, whether in the mask view or any dimensionality reduction view, aligns with these findings. This differentiation aids in pinpointing the specific U-Net layers responsible for segmentation differences.

5.6.2 Effectiveness of Coupled Window Visualizations

The second research question (B) and hypothesis (B) explore the capabilities of coupled window visualization. The study's results provide substantial evidence supporting the efficacy of this visualization technique. When comparing the experimental group (exposed to coupled window visualization) to the control group, a significant improvement was observed in the experimental group's understanding and decision-making capabilities. Specifically, 92.45% of the experimental group participants responded positively to the idea of layer freezing for U-Net optimization in Task 2, compared to 82.35% in the control group. Moreover, 69.81% of the experimental group believed that the coupled window visualization provided hints on which layers should be frozen, notably higher than the 56.86% observed in the control group. Feedback from the experimental group further emphasized the intuitive nature of the controls and legends, the detailed explanations accompanying the visualizations, and the advantage of viewing all visualizations together in a coupled manner. These findings validate Hypothesis B, suggesting that coupled window visualization can present differences more clearly and intuitively, assisting users in tasks based on visualization settings with increased accuracy.

5.6.3 Optimizing U-Net Layers through Visualizations

The third research question (C) and hypothesis (C) focus on the potential of coupled window visualizations to aid users in refining their machine learning models, particularly the U-Net. The results from Questions 6 and 7 in the user study provide evidence for the usefulness of

coupled window visualizations in guiding users toward optimizing the U-Net layers.

Specifically, a significantly higher percentage of the experimental group participants responded positively to the idea of freezing some layers during transfer learning in Task 2 (92.45%) compared to Task 1 (79.25%). This indicates that the coupled window visualization provided insights to guide participants' decisions about layer optimization.

Additionally, in the layer freezing experiment, selectively freezing layers like the bottom encoder layers objectively improved the U-Net's segmentation performance. This aligns with the user study, where coupled window visualizations increased the probability of participants choosing to freeze the bottom encoder layers.

In summary, the noticeable improvement in the experimental group's responses and layer choices in the user study, coupled with the performance benefits observed from selective layer freezing in the experiment, validate Hypothesis C. The results suggest that examining variations in the visualization can help determine optimal U-Net layers to improve machine learning accuracy.

5.6.4 Optimizing Freezing of U-Net Layers

The fourth research question (D) and hypothesis (D) focus on the usefulness of coupled window visualizations in helping users determine which specific layers to freeze in the U-Net.

The user study results provide evidence for this hypothesis. After being introduced to coupled window visualizations in Task 2, the experimental group showed significant improvement in their layer freezing choices compared to Task 1. For instance, 41% chose to freeze bottom encoder layers in Task 2, up from 31% in Task 1.

In the layer freezing experiment, freezing bottom encoder layers led to the highest segmentation accuracy. This alignment between the user study layer choices and experiment performance highlights the impact of coupled window visualizations on guiding optimal layer freezing decisions. Additionally, statistical tests confirmed the differences in responses between the control and experimental groups in Task 2 were significant. Feedback from the

experimental group also emphasized that the coupled window visualization aided their task completion.

In summary, the noticeable shifts in the experimental group's layer freezing choices after using coupled window visualizations, validated by the experiment results, provide evidence for Hypothesis D. The coupled visualizations enabled more accurate optimization of freezing specific U-Net layers.

Chapter Six

Application of visualization on VCCF project

6.1 VCCF Project Background

The Vascular Common Coordinate Framework (VCCF) is a novel approach to mapping all cells in the human body. This concept originated from discussions on cell mapping during a meeting in 2017 meeting (Bethesda, 2017). Among various proposals, the idea of using the vasculature as a coordinate system was further discussed and developed at a 2019 CCF Workshop, specifically concerning kidney cells (HuBMAP, 2019).

The vascular system has several properties that align with previously identified characteristics for a Common Coordinate Framework (CCF) to enable the mapping of all cells in the human body (G. M. Weber et al., 2020). The VCCF does not aim to construct a map of the vasculature itself. Instead, it uses known vascular pathways throughout the body as axes in a coordinate system. This system is designed to describe the positions of anatomical elements, such as cells, within the tissue surrounding the vasculature (G. M. Weber et al., 2020).

6.2 VCCF Methodology

The VCCF methodology uses the vasculature as a new 3D coordinate system to map all cells in the human body comprehensively. It includes sequential steps to ensure the precision of the 3D reconstruction for the comprehensive analysis of interactions between cells and blood vessels. (G. M. Weber et al., 2020)

First, the countless blood vessels are simplified into representative pathways for each

organ and tissue type. At functional units, small serve as common reference points; for example, glomeruli in kidney organs. Once these prototype vessels are mapped, cells can be located based on their proximity to the nearest vessel. In constructing the 3D map, serial tissue sections are analyzed using specific biomarkers, generating a volume of various cell types (Ghose et al., 2022).

The VCCF methodology utilizes the widespread vascular network to map the human body at the cellular level. The vasculature provides a common framework for integrating cell data into a unified reference map. Ongoing research may further refine this system by adding vessel sub-types with different and unique molecular features.

In this VCCF visualization project on skin sample data, which was a collaboration with the GE Research team, 3D serial image sections have been used to merge overlapping cell segmentations (Ghose et al., 2022). The GE team primarily focused on the segmentation of the skin data, and I managed all visualization aspects.

6.3 Visualizations of VCCF

The visualization of VCCF involves creating a 3D volume depicting various immune and endothelial cell types, such as T-killer, T-reg, T-helper cells, and macrophages in skin samples. It includes calculating cell distance metrics and generating interactive visualizations. Taking the skin sample as an example, the cell distance metrics have two analyses: the distance from immune cell nuclei to the edge of the nearest blood vessel (endothelial cell), and the nearest distance between UV damage/proliferation markers and the edge of the skin surface. Interactive visualizations of cell and marker distances can be implemented using various visualization tools or packages, including the Plotly or Bokeh 3D visualization packages.

The results of the distance calculations are displayed and visualized in multiple views. The primary view of the visualization includes an overall 3D projection of all immune cell nuclei, damage/proliferation markers, blood vessels, and their shortest distances (connecting

lines) in tissue space. The visualization of distance links has been optimized by adding invisible links that connect all the existing links into a single long polyline. This reduces the size of the vector data and memory usage, enhancing the responsiveness of online interactions with substantial numbers of nodes or elements (ranging from 20,000 to 50,000) in a web browser.

Several sub-plots can also be added to enhance the visualization. Histograms or violin charts can show statistical information about the distribution of distances between nuclei and blood vessels and between damage/proliferation markers and the skin surface. Another type of visualization includes an immune cell cluster density visualization. This 3D visualization illustrates the distribution of immune cell clusters, with the size and color of the bubbles indicating the relative number of neighboring immune cells in 3D.

After developing the visualization techniques for the VCCF project, we recognized the opportunity to move from single, standalone visualizations to more integrated, coupled window visualizations that combine multiple visualizations and the links or interconnections between them. This change could offer a better and more connected view of the data and help different types of users understand the data better. Therefore, it's important to evaluate the effectiveness and user-friendliness of coupled window visualization, especially in the context of VCCF projects and machine learning. This evaluation and validation will be discussed in the next user study section.

Besides the recommendations on using suitable graphic symbols and variables, as mentioned in the design of visualizations in the previous chapter, the DVL paper (Börner et al., 2019) provides a comprehensive framework to create effective interactive 3D visualizations for better comprehension and interpretation of segmentation results. In the VCCF project, the DVL framework guided our approach to visualize immune cell distributions, and in choosing the coupled 3D view to effectively show the relationships between various cell types.

6.4 VCCF User Study

As the complexity of machine learning models has increased, it has become challenging for users to accurately interpret their results. Visualization tools can help improve the understanding of these models, but their effectiveness has not been thoroughly assessed or evaluated. This section introduces the VCCF user study, which evaluates the usefulness and user-friendliness of coupled window visualizations, specifically applied in the VCCF project based on our prior skin dataset and visualizations. Our goal is to determine whether coupled window visualizations provide a clearer understanding of machine learning results compared to single, standalone visualizations.

The VCCF user study primarily targets participants with little to no experience in machine learning. It aims to help these novice users better understand the results of immune cell segmentation from skin samples using coupled window visualization. The findings from this user study could provide valuable insights into the future application of coupled window visualization in the field of machine learning, particularly for medical image segmentation.

Similar to the FTU user study design in the previous chapter, the DVL framework guided the design of VCCF user studies and tasks to evaluate visualizations, ensuring alignment with key visualization literacy concepts. The most important difference is that VCCF user studies focus on novice users of machine learning. DVL framework guided designing simple and complex question categories to evaluate different aspects of understanding the machine learning result from the visualizations.

6.4.1 Objectives

The main objective of this VCCF user study was to assess the effectiveness of coupled window visualizations in helping novice users understand machine learning predictions and results. To achieve these objectives, we recruited 102 student participants for the study. Both the control and experimental groups comprised an equal number of participants, each group

containing approximately 51 members.

Research Questions

E. Will the use of coupled window visualization effectively help the user in completing simple tasks based on visualization settings more accurately?

F. Will the use of coupled window visualization help the user in completing complex tasks (e.g., performing complicated calculations based on given information) across three panels of visualizations more accurately?

Hypotheses

E. The use of coupled window visualization will effectively and clearly help the user in completing simple tasks faster and more accurately.

F. The use of coupled window visualization will help the user perform complex tasks (e.g., performing complicated calculations based on given information) faster and more accurately across three panels of visualizations.

6.4.2 Methodology

Participants

The participants' sample size estimation was conducted using G.Power, a statistical software commonly utilized for research studies (Faul et al., 2007). To ensure equal representation, there were equal numbers of participants in the control and experimental groups. Therefore, 51 student subjects were assigned to both the control and experimental groups.

The statistical test chosen was the t-test (Student, 1908). As in the prior user study in Chapter 5, the effect size was 0.5, alpha 0.05, and power 0.8. The allocation ratio of 1 indicates equal participant numbers per group. These parameters required a minimum of 51 individuals per group to detect moderate mean differences with an 80% probability and 0.05 significance. Therefore, the 102 total students provided a sufficient sample size to meet the study objectives. The study design aligned with ethical considerations and followed strict

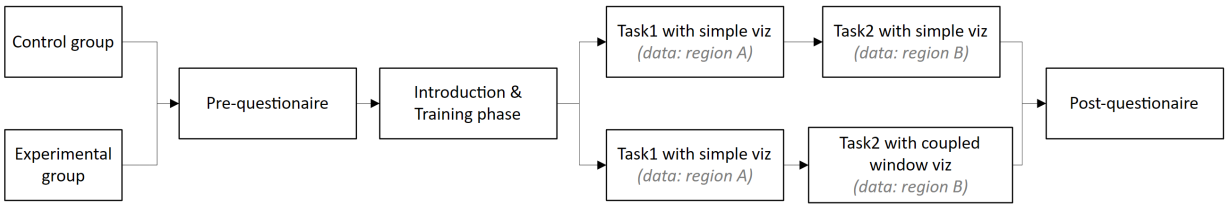


Figure 6.1 User Study Pipeline

protocols to ensure the well-being and privacy of all participants. Moreover, the user studies were approved through the Institutional Review Board (IRB) application process.

In this study, we aimed to recruit novice users to participate in the user studies. The novice users had limited to no prior experience in machine learning. The coupled window visualization was expected to help novice users comprehend the outcomes and results of immune cell segmentation from skin samples. As they lacked technical expertise in data analysis or programming, they required additional guidance and support to fully understand the system complexities.

User study pipeline

In the VCCF user study, two types of visualizations were used: simple standalone visualizations and coupled window visualizations. The simple visualizations comprised machine learning predictions using different visualization methods and displaying statistics and distributions obtained from the predictions. The coupled window visualizations combined multiple different types of visualizations in a single view, enabling users to link the panels and navigate between the various panels of visualizations.

The simple visualization included a 3D view (Figure 6.2), histogram (Figure 6.3), and violin plot (Figure 6.4). The coupled window visualization combined all three visualization types in one view (Figure 6.5). The 3D view visualized immune and endothelial cells and the distances between them. The histogram and violin plot displayed the statistics and distributions of the distances between immune cells and blood vessels.

The pipeline of the user study was similar to the pipeline in the previous chapter. The study consisted of two tasks, Task 1 and Task 2, and featured two groups of participants, the control group and the experimental group. This setup helped to analyze the impact of coupled window visualization on the users' performance.

To ensure that the results from the study reflected the broader population, demographic information was collected from each participant in both groups (refer to the Study Material section in the Appendix). During the introduction and training phases, participants were guided to become familiar with the task requirements and methods to complete the tasks. A sample question was provided after each session, and the correct answer was provided to confirm participants' comprehension of the instructions.

In Task 1, both the control and experimental groups were presented with simple visualizations and the following relevant questions. For Task 2, the control group continued to use simple visualization, while the experimental group switched to using coupled window visualization. This setting was designed to evaluate two primary comparisons: (1) To compare the impact of coupled window visualization on the experimental group's performance in Task 2 by comparing it to their earlier performance with simple visualization in Task 1, and (2) To determine the effect of coupled window visualization on users' performance by performing a cross-comparison between the control and experimental groups in both tasks. The combined results from these two comparisons indicated the impact of coupled window visualization on the users' performance.

In addition to the two explicit comparisons outlined above, there were also two implicit comparisons. The first was a comparison for the control group between Task 1 and Task 2 to assess whether they achieved similar performance as the baseline. Since Task 1 and Task 2 involved similar tasks and questions and used simple visualizations, we expected their performance to remain consistent in both tasks. The second implicit comparison was a cross-comparison of Task 1 performance between the control and experimental groups to determine whether they achieved similar performance as the baseline. Both groups were

presented with the same simple visualizations in Task 1, and we expected their performance to be similar.

After completing each task, participants from both groups completed post-task questionnaires to help analyze their understanding of the task and their levels of satisfaction. In addition, the experimental group was asked about their familiarity with the coupled window visualization. On the other hand, since the control group was not exposed to the coupled window visualization, they were introduced to the concept of coupled window visualization through text. They were given guidance to compare their experience with the simple visualization they had seen and were asked to provide their insights and suggestions for the improvement of the visualization based on their expectations and the textual description.

Visualization description

The participants in the control group were presented with a set of three distinct types of visualizations. The first visualization was a 3D view (Figure 6.2) illustrating the spatial distribution of skin cells, segmented using a U-Net machine learning model. The second visualization was a histogram (Figure 6.3) that showed the distances between skin immune cells and vascular cells. The third one was a violin chart (Figure 6.4) that also illustrated these distances but in different formats. The experimental group, on the other hand, was introduced to a coupled window visualization comprising three panels (Figure 6.5), which included the same visualizations as those provided to the control group but displayed them in a linked, interactive format.

The first visualization (Figure 6.2), a 3D view, also served as the main view within the coupled window. It displayed the segmentation of skin tissue in a specific region through interactive visualization, using 3D space to represent immune and vascular cells segmented via the U-Net model. Distances from each immune cell to the nearest vascular cell were calculated and visualized as 3D line segments. The primary immune cells included macrophages (CD68+), T helper cells (CD3+CD4+), T regulatory cells (CD3+CD4+FOXP3+), and T

killer cells (CD3+ CD8+). They are visualized as 3D spherical shapes in various colors, which were cross-referenced in the legend. Vascular cells were represented by red 3D spherical shapes. Each immune cell was linked to the nearest vascular cell (blood vessel) via a 3D line segment. Similar to the distances between immune cells and vascular cells, the visualization also showed the shortest distances between the skin surface and ultraviolet (UV) damages caused by sunlight. Various types of UV damage, including DDB2 and P53, as well as proliferation marker KI67, were represented as cross-shaped markers in different colors. A color reference was also provided in the legend. Gray 3D points indicated the skin surface, and the shortest distances between UV damage and skin were also represented by 3D line segments. In the subtitles of the visualization, there was a link to the original virtual H&E images. The visualization included interactive features such as zooming, dragging, and rotating operations in the 3D view, and users could enable or disable items in the legend by clicking to show or hide specific elements. When the cursor hovered over a certain immune cell or UV damage, detailed information about this cell or damage would be displayed in real-time, along with the 3D coordinates and the distance to the nearest vasculature or skin surface.

The second view, a histogram (Figure 6.3), statistically showed the distances between skin immune cells and vascular cells. The visualization was mainly divided into two sections. The upper half used a histogram to show distance statistics, and also used a fitted curve to show the nearest distance from different types of immune cells to the vessels. The lower half specifically displayed the distribution of distances for each cell individually. Each small line segment represented the position of a specific immune cell's distance within the overall distribution. The histogram featured three display modes: overlaid, stacked, and grouped, which could be toggled using a dropdown menu at the bottom, making overlapping bars more distinguishable in different modes to enhance visibility. Hovering the mouse over each bar in the histogram displayed the range of the bar and the count of samples. In the lower part that provided specific distribution, hovering the mouse would show the distance of each

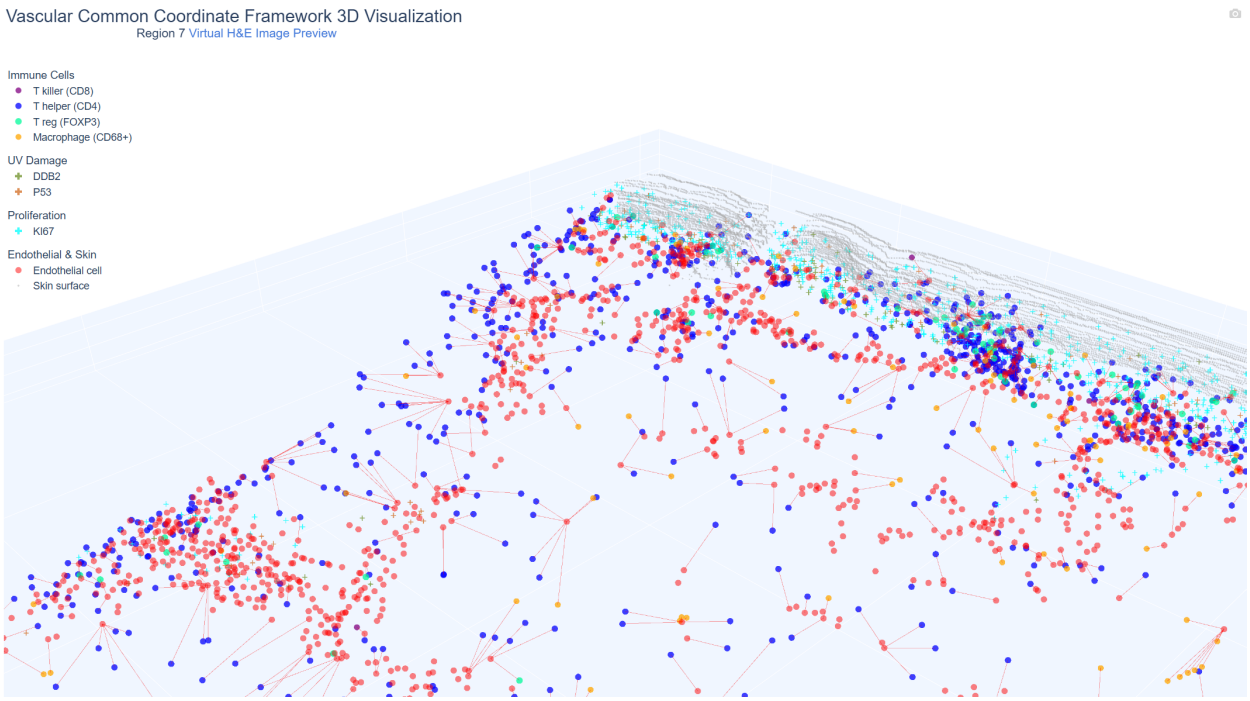


Figure 6.2 Interactive Visualization - 3D View

specific cell.

The third violin chart (Figure 6.4) showed the distance statistics between skin immune cells and vascular cells from another perspective. Unlike the histogram, the violin chart provided a top-down overview of distance distributions across different regions. In the violin chart, the visualization separately showed the distribution of four different immune cells (macrophages, T helper cells, T regulatory cells, and T killer cells) in different skin regions,

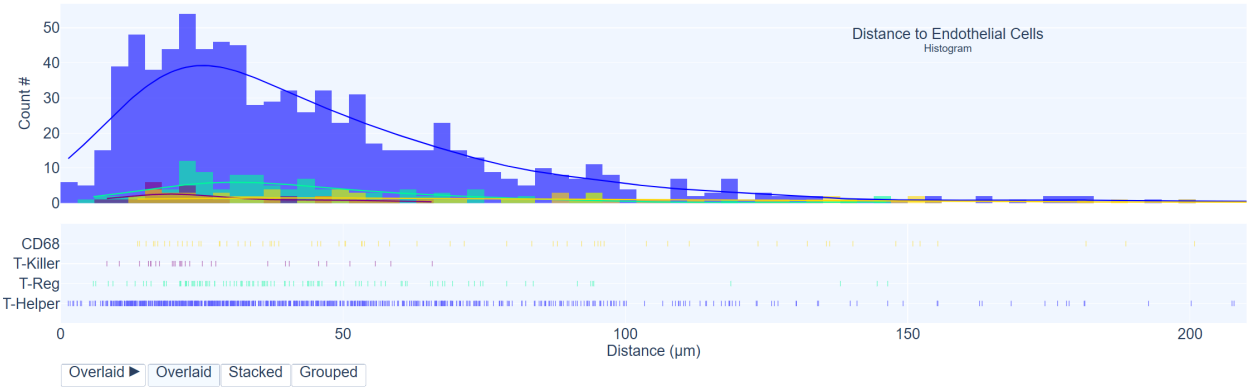


Figure 6.3 Interactive Visualization - Histogram View

also containing the statistics for the sum of all four types of cells (All cells). The outer part of each violin showed the statistical distribution of distances between immune cells and blood vessels, similar to the fitted curve in the histogram but more intended for comparing samples from different regions. These violin charts combined a box plot and a density plot to display the probability density of the data at different values. The interquartile range was represented by the box in the center, and the extended line above/below the box showed the upper (max) and lower (min) adjacent values. The median value of the distance was represented by the line in the middle of the box. When the cursor hovers over each violin chart, all detailed information was displayed in real time.

The coupled window visualization (Figure 6.5) effectively integrated all the described visualizations into a coherent, interactive panel (combining vertically, horizontally, or both). Besides the different interactions described in the three visualizations, the coupled window also supported connections between different types of visualizations. It allowed for more than a simple side-by-side presentation by linking these related data points across different visualizations. Clicking on an immune cell in the 3D view not only highlighted that cell but also the corresponding points in the histogram and the relevant area in the violin plot, enhancing the user's understanding and interaction with the data. If we click on the cell again, all corresponding elements will return to their original state. If we click on another cell, the previously clicked cell would revert to its original state, while the newly clicked cell and corresponding elements in other panel visualizations would be highlighted.

6.4.3 Results

In this section, we present the results of the VCCF user study. Users in both groups completed tasks by answering questions based on different combinations of visualizations, and their accuracy was recorded for further analysis.

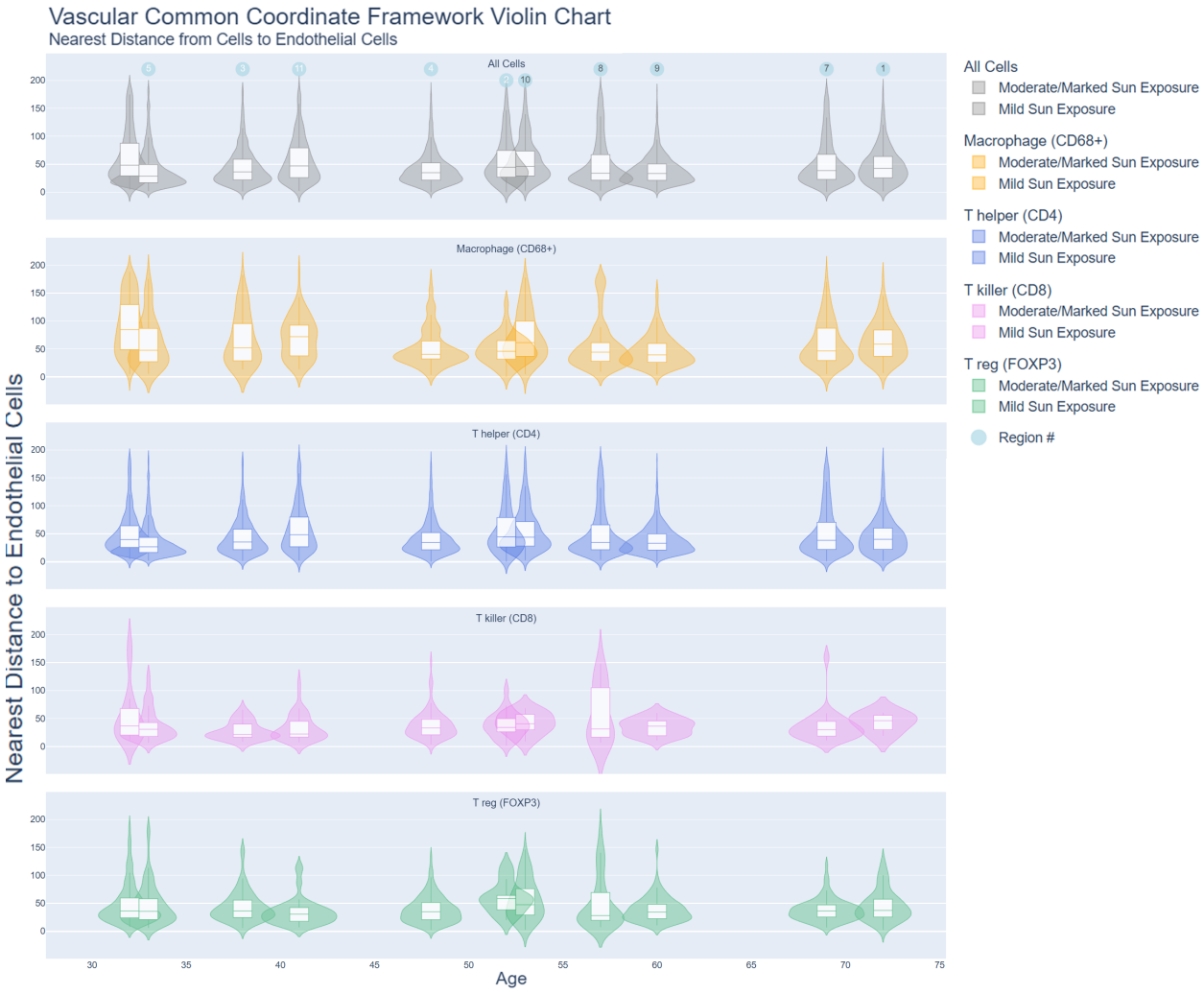


Figure 6.4 Interactive Visualization - Violin Plot

		Q1	Q2	Q3	Q4	Q5	Q6	Q7
Control Group	Task 1	33.33%	58.82%	60.78%	56.86%	50.98%	78.43%	41.18%
	Task 2	50.98%	54.90%	49.02%	47.06%	45.10%	70.59%	37.25%
Experimental Group	Task 1	28.85%	55.77%	63.46%	57.69%	50.00%	76.92%	34.62%
	Task 2	50.00%	57.69%	67.31%	55.77%	57.69%	78.85%	40.38%

Table 6.1 Comparison of Positive Response in Task 1 and Task 2

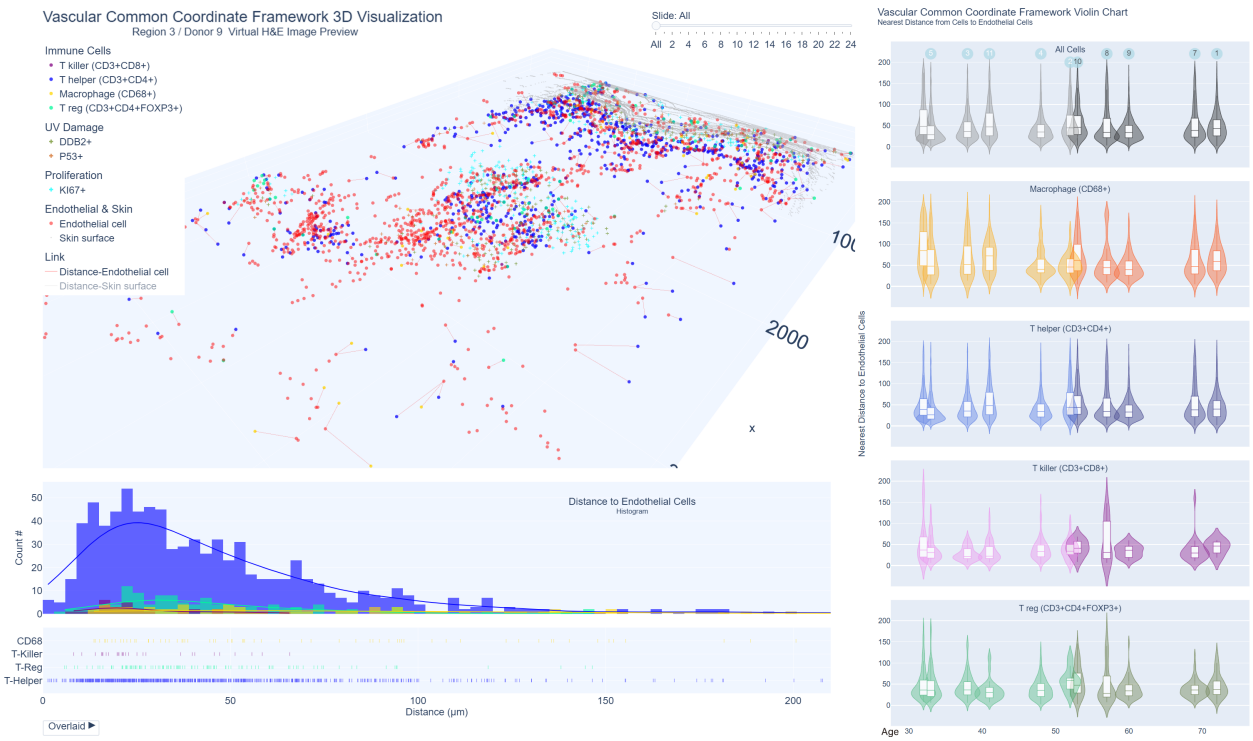


Figure 6.5 Interactive Visualization - Coupled Window Visualization

Task analysis

The VCCF user study results for Questions 1 to 7 are presented in Table 6.1. To better analyze the results, we categorized the questions into two groups: Questions 1 to 4 as a set of simple questions and Questions 5 to 7 as a set of complex questions that required the simultaneous use of multiple visualization types.

For Task 1, the control group achieved an average accuracy of 44.3% on Questions 1 to 4, while the experimental group achieved an average accuracy of 47.8%. This difference was not statistically significant ($t = 1.05$, $p = 0.30$). However, for Questions 5 to 7, the control group achieved an average accuracy of 37.2%, while the experimental group achieved an average accuracy of 45.7%. This difference was statistically significant ($t = -2.36$, $p = 0.024$). This suggests that the coupled window visualization used in Task 2 may have been more effective for Questions 5 to 7 compared to the simple visualization used in Task 1.

For Task 2, the control group achieved an average accuracy of 35.8% for Questions 1 to 4,

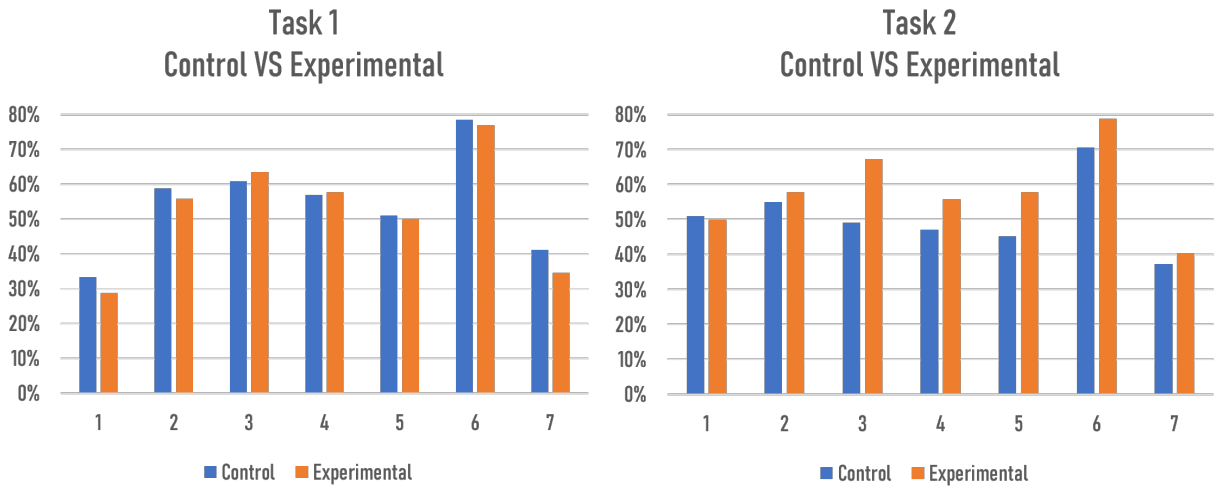


Figure 6.6 Comparison of Positive Response in Task 1 & Task 2

whereas the experimental group achieved 60.4%. This difference was statistically significant ($t = -6.12$, $p < 0.001$). Similarly, for Questions 5 to 7, the control group achieved an average accuracy of 30.4%, and the experimental group achieved 52.2%. This difference was also statistically significant ($t = -4.42$, $p < 0.001$). These results indicate that for both groups of questions, the coupled window visualization in Task 2 was more effective than the simple visualization set used in Task 1 for both groups, particularly for Questions 5 to 7, which required integrating multiple visualizations.

Upon examining the performance of the experimental group across Tasks 1 and 2, it was observed that their performance in Task 2 for Questions 5 to 7 was significantly better than in Task 1 ($t = -3.13$, $p = 0.005$). This suggests that for the experimental group, the coupled window visualization in Task 2 was more effective than the simple visualization used in Task 1. However, no significant performance difference was noted for Questions 1 to 4 between the two tasks ($t = -0.23$, $p = 0.82$).

When comparing the control group's performance in Tasks 1 and 2, there was no significant difference for Questions 1 to 4 ($t = -1.20$, $p = 0.24$). However, in Task 2, the experimental group significantly outperformed the control group for Questions 5 to 7 ($t = -4.36$, $p < 0.001$).

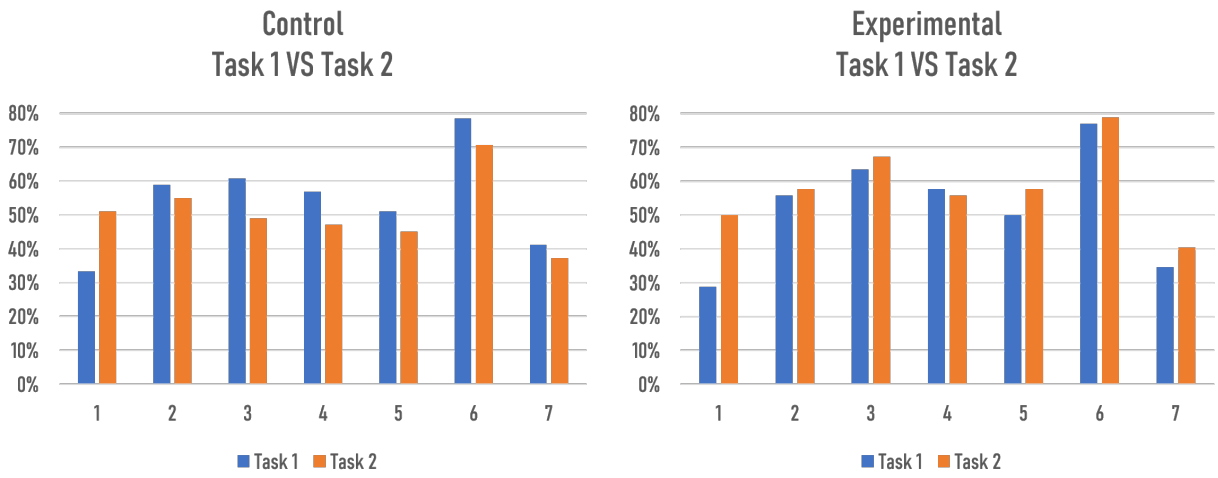


Figure 6.7 Comparison of Positive Response in Task 1 & Task 2

Comparing both groups across Tasks 1 and 2, the experimental group performed significantly better in Task 2 than the control group for both Questions 1 to 4 ($t = -6.28$, $p < 0.001$) and Questions 5 to 7 ($t = -5.86$, $p < 0.001$). This indicates that, for both sets of questions, the coupled window visualization in Task 2 was more effective than the simple visualization used by the control group, especially for Questions 5 to 7.

In summary, the results indicate that the coupled window visualization employed in Task 2 was more effective than the simple visualization used in Task 1 and by the control group, particularly for questions that required information from multiple types of visualizations. However, the type of visualization used in Task 1 did not have a significant impact on performance for Question 1 to Question 4. These findings suggest that coupled window visualizations are preferable for complex questions, and the results highlight the importance of considering the complexity of the task and the type of visualization used when designing visualizations for machine learning. For simpler tasks, a simple visualization may be sufficient, while more complex tasks may benefit from a more informative and comprehensive visualization approach. The use of multiple types of visualizations, such as the coupled window visualization used in Task 2, can also be beneficial for complex tasks that require the integration of information from various sources.

		Q1	Q2
Control Group	Agree	54.90%	64.71%
	Neutral	23.53%	33.33%
	Disagree	21.57%	1.96%
Experimental Group	Agree	62.75%	62.75%
	Neutral	11.76%	23.53%
	Disagree	25.49%	13.73%

Table 6.2 Comparison of Post Questions

Post questionnaire analysis

In the post-questionnaire, both the control group and the experimental group were asked two questions regarding the usability and helpfulness of the visualizations in the tasks. The first question was similar for both groups: **"Would you say that the visualizations in the task were easy to use?"**. The control group was asked about simple visualizations, while the experimental group was asked about coupled window visualizations. The second question was different. For the experimental group, the second question asked, **"Would you say that compared to the visualizations in Task 1, the coupled window visualizations in Task 2 helped you more to complete the task?"**. Since the control group had not experienced the coupled window visualization, the second question first described in detail what coupled window visualization looks like, and then asked, **"Would you say that compared to the visualizations in Tasks 1 and 2, the coupled window visualizations would be helpful to complete the task?"**. These questions were designed to evaluate the effectiveness of the coupled window visualizations and user satisfaction compared to the simple visualizations used in Task 1.

In the control group, 54.9% of participants agreed that the simple visualizations were easy to use, while 64.7% of participants agreed that the coupled window visualizations would help

complete the tasks compared to the visualizations in Tasks 1 and 2. In the experimental group, 62.7% of participants agreed that the coupled window visualizations were easy to use, while 62.7% of participants agreed that the coupled window visualizations in Task 2 were more helpful than the simple visualizations in Task 1.

For Question 1, both the control and experimental groups were mostly positive about the usability of the visualizations, with a higher percentage of agreement in Question 1 in the experimental group (62.7%) compared to the control group (54.9%). This suggests that the coupled window visualization may be more effective in helping novice users complete the task.

In Question 2, the control group was asked about the potential usefulness of the coupled window visualization without experiencing it firsthand, while the experimental group was asked about their perceived usefulness after experiencing it. The results indicated that both groups perceived the coupled window visualization as potentially beneficial, with a slightly higher percentage of agreement in the experimental group (25.5% and 13.7% respectively). This indicates that users recognize the potential benefits of having multiple visualizations side by side and connected, even if they have not experienced it firsthand.

Comparing the results of Questions 1 and 2 between the control and experimental groups, we can see a clear difference in Question 1, with the experimental group reporting a higher percentage of positive responses. This supports the expectation that coupled window visualization would be more effective in aiding understanding compared to simple visualizations. However, there was no significant difference in the responses to Question 2 between the two groups, suggesting that the potential benefits of coupled window visualization were recognized by both groups, even without experiencing it directly.

Overall, the study indicates that coupled window visualization can be an effective tool for helping novice users understand the results of machine learning. Future research should explore how different types of coupled window visualizations can be designed to aid novice users in understanding different types of machine learning results more effectively.

Finally, three subjective questions provided feedback on the user's opinion and satisfaction with the different visualizations

What did you like about the visualizations? In general, both groups appreciated the interactivity and ease of use of the visualizations. They found the representations to be informative and well-organized, with clear graphics and colors. Users liked the ability to interact with the data and examine individual data points closely.

The main difference between the groups was that the experimental group found the coupled window visualization to be convenient and time-saving, allowing them to view all necessary information in one window without having to switch between windows. On the other hand, some users in the control group expressed difficulty in understanding the violin graph.

What did you not like about the visualizations? After analyzing the responses, it appears that the control group had mixed reactions to the visualizations, ranging from "finding them too complex and not clearly communicating information" to "liking almost everything". Some participants found the color selection and controls on the 3D view hard to use, while others complained about the lack of clarity in explaining the visualizations. There were suggestions for improvements such as better clarity and instructions, more intuitive interactions, and highlighting previous values to aid in answering questions.

The experimental group had better reactions. Users from the experimental group thought the instructions were mostly clear, but some participants found it unclear which region to look at in the violin plots during the first task (based on simple visualization).

What can be done to improve the visualizations? / What can be done to improve the coupled window visualizations? After reviewing the answers provided by the control group and experimental group, it is clear that there are some similarities and differences in their feedback regarding the improvement of visualizations and coupled window visualizations.

In summary, both the control group and the experimental group had suggestions for im-

proving the visualizations. The most common suggestions were to simplify the visualizations and improve zooming and panning functions. Specifically, the experimental group, which was asked about improving the coupled window visualizations, had additional suggestions such as showing more relevant information instead of all information and improving label clarity and consistency between plots.

However, there were also some differences in the feedback between the two groups. One notable difference was the experimental group's emphasis on enhancing the interactivity and relationships between visualizations. Users in this group suggested highlighting corresponding points between visualizations or labeling them more clearly. They also suggested separating the visualizations into different rows and adding the option to choose which one to use at a time for simple tasks. This indicates that the experimental group was more concerned with the interactivity and relationships between the different visualizations.

On the other hand, the control group's feedback focused more on the individual visualizations themselves. Feedback in this group suggested changing the colors of the 3D visualization or making the histograms more zoomable. Additionally, some users in this group had trouble understanding the violin charts or finding the specific information they were looking for. This indicates that the control group was more focused on the details and specific aspects of the visualizations. These differences might suggest that the experimental group was more concerned with an integrated overview of the data, while the control group was more concerned with the detailed and specific information within the visualizations.

6.5 Discussions

The discussion section focuses on the analysis of the results and findings from the VCCF user study, particularly focusing on the effectiveness of coupled window visualizations in aiding users, both in understanding the results of VCCF medical image segmentation and in performing complex tasks across multiple panels of visualizations. The results are discussed

based on the research questions and hypotheses presented in the objectives section.

6.5.1 Efficacy in Completing Simple Tasks

The VCCF user study provided substantial evidence supporting the effectiveness of coupled window visualization in the VCCF project, particularly for users with limited experience in machine learning. The experimental group, exposed to this visualization technique, consistently outperformed the control group, which used only simple visualizations. Specifically, in Task 2, the experimental group achieved an average accuracy of 60.4% for simpler questions and 52.2% for more complex ones, while the control group managed only 35.8% and 30.4% respectively. This significant improvement in performance highlights the better clarity and comprehension from the coupled window visualization.

Feedback from participants further reinforced these findings. The experimental group particularly appreciated the convenience and efficiency of the coupled window visualization. They emphasized its capability to present comprehensive information within a single window through multiple panels, without switching between different windows or tabs. This feedback not only focuses on the method's effectiveness but also its user-friendliness.

Additionally, most suggestions from the experimental group focused on improving the interactivity and connectivity between the visualizations. Their goal was to emphasize the overall view of the data. This focus on integrating the visualizations shows that the coupled window design helped users gain a more comprehensive understanding of the topic.

In alignment with Hypothesis E, the findings strongly suggest that coupled window visualizations effectively assist users in completing tasks with greater speed and accuracy based on coupled window visualization settings. This innovative approach demonstrates the potential and importance of visualization techniques in analyzing complex data and results from machine learning.

6.5.2 Efficacy in Completing Complex Tasks

The VCCF user study also explored the utility of coupled window visualizations for complex tasks. The experimental group, with coupled window visualizations, showed notable improvement in their performance for complex questions in Task 2. Their accuracy increased from 45.7% in Task 1 to 52.2% in Task 2 for these questions requiring various visualizations simultaneously. This significant increase in accuracy shows the potential of coupled window visualizations to help users understand and analyze complex data more effectively.

As supporting evidence, a majority (62.7%) of the experimental group participants in the post-task questionnaire agreed that the coupled window visualizations were not only user-friendly but also more beneficial than the simple visualizations provided in Task 1. This feedback is important as it reflects the users' subjective experience and satisfaction with the visualization tool, which is crucial for its adoption and application in further VCCF projects and other potential HuBMAP projects.

Additionally, feedback from the experimental group emphasized the convenience and time-saving aspect of the coupled window visualization. Users particularly mentioned the ability to gain insights from multiple visualizations simultaneously, without having to switch between them. This feedback highlights the real-world benefits of using coupled window visualizations, especially when users need to combine information from different types of sources.

In conclusion, the data strongly supports the claim that coupled window visualizations significantly improve a user's capability to understand the result of machine learning and handle complex tasks across various visualization panels. This not only confirms Hypothesis F but also highlights the potential of coupled window visualizations as a powerful tool in VCCF and HuBMAP projects for data analysis and machine learning.

6.5.3 Conclusions

In summary, the VCCF user study offers strong evidence supporting the effectiveness of coupled window visualizations, both in understanding medical image segmentation results and in completing complex tasks across multiple visualizations. The experimental group, which previewed this innovative visualization method, consistently outperformed the control group. User feedback further highlighted the user-friendliness and efficiency of the coupled window visualization, emphasizing its potential as a valuable tool in machine learning, particularly for medical image segmentation and data analysis. These results not only confirm our initial hypotheses but also emphasize the importance of user-centered visualization techniques in the machine learning field.

Chapter Seven

Conclusions

This dissertation explores the application of data visualization techniques to enhance the interpretability of complex machine learning models for medical image segmentation. Two use cases within the HuBMAP project demonstrate the value of coupled window visualizations in providing intuitive insights into complicated machine learning models such as CNNs and U-Nets.

The Functional Tissue Unit (FTU) study showcased how coupled window visualizations enabled experienced users to better comprehend U-Nets internal representations and select optimal parameters to improve its accuracy and efficiency. The user study and the layer freezing experiment validated that data visualization significantly enhances performance when determining the optimal layers to freeze during transfer learning.

Similarly, the Vasculature Common Coordinate Framework (VCCF) study highlighted the effectiveness of coupled window visualizations in aiding even novice users with no machine learning expertise to accurately interpret the outputs of medical image segmentation models. The user study confirmed the benefits of completing complex analytical tasks across different types of visualizations.

A key contribution of this dissertation is the empirical demonstration of the significance and potential of visualizations for understanding complex machine learning models and techniques. The formal user studies presented in the dissertation make a strong and compelling case for integrating well-designed visualizations into regular research practices.

In conclusion, this dissertation shows that visualizations can simplify the analysis and

improvement of machine learning models like CNNs and U-Nets, making them more intuitive and easier to understand. It presents a strong argument for the essential role of visualization in advancing research and development in machine learning, particularly in applications involving medical imaging. The designs, implementations, and evaluations of the visualizations in this dissertation provide a framework for creating visualizations that reveal insights into complex models. Furthermore, in support of ongoing and future research and development, all data and code associated with this dissertation have been made available on GitHub (Refer to Section B in Appendix).

Appendix A

User Study Material

A.1 Pre-questions

Questions:

Q2.1 Timing

Q2.2 Please answer the following questions below to help us understand your background and experience.

Q2.3 Please indicate your age:

- 18-20
- 21-30
- 31-40
- 41-50
- 51-60
- >60

Q2.4 Please indicate your gender:

- Male
- Female
- Identity not listed above
- I prefer not to answer

Q2.5 Would you say that you are familiar with the following types of data visualizations?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Tables	○	○	○	○	○
Charts, such as pie charts or bubble charts	○	○	○	○	○
Graphs, such as scatter graphs, bar graphs, or line graphs	○	○	○	○	○
Maps	○	○	○	○	○
Tree visualizations	○	○	○	○	○
Networks	○	○	○	○	○

Q2.6 Would you say that you are familiar with the following machine learning model?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Convolutional Neural Network	○	○	○	○	○
UNet	○	○	○	○	○

Q2.7 Are you far or near-sighted, or do you have any other vision impairments?

- Far-sighted
- Near-sighted

- Other: _____
- I do not have any vision impairments.
- I prefer not to answer

Q2.8 Are you color-blind? If so, please specify.

- No
- Yes, specifically: _____
- I prefer not to answer

A.2 Demographics (Base/Universal)

Questions:

Q3.1 What is the highest level of school you have completed or the highest degree you have received?

- Less than high school degree
- High school graduate (high school diploma or equivalent including GED)
- Some college but no degree
- Associate degree in college (2-year)
- Bachelor's degree in college (4-year)
- Master's degree
- Doctoral degree
- Professional degree (JD, MD)

Q3.2 Are you a native English speaker?

- Yes
- No
- Prefer not to answer

A.3 Instructions

FTU User Study Control Group

Questions:

Q4.1 Timing

Q4.2 Introduction

In the following tasks, you will see some visualizations. These visualizations show some information from machine learning features and result of a UNet model. The purpose of UNet is to extract the segmentation of the crypt part (or we call it "FTU" below) of the colon sample. Since our colon dataset is small, we applied transfer learning. We first trained the model on the kidney dataset (a relatively large dataset), and then applied transfer learning to the colon dataset.

Based on the features of each layer in the UNet structure and the predicted mask, we performed the following visualization. Here we will give a detailed introduction to all the visualizations that may be encountered so that you can compete the next tasks without any difficulty.

Q4.3 UNet visualization overview

This is an overview of a visualization that might appear below. This visualization shows a UNet structure that mainly contains encoder pipeline, decoder pipeline, upsample pipeline and the final pipeline.

U-Net is a convolutional neural network that was developed for biomedical image segmentation. The network is based on the fully convolutional network and its architecture

was modified and extended to work with fewer training images and to yield more precise segmentations.

Each small unit shows the feature of a specific layer in the UNet. With a dimensionality reduction (2-dimensional) method applied on the features, the features can be mapped to the 2-dimensional plane. 288 images were fed into the UNet model and therefore the prediction contains 288 masks and each layer has 288 feature points. We will further describe the details of one unit in the next paragraph.

Q4.4 UNet visualization detail

A small portion (extracted from the original view) of the visualization is shown below. On each area there are 288 points, which means the features of 288 inputs. All feature points in the visualization are labeled as three categories here: FTU, non-FTU and FTU edge. FTU here means Functional tissue unit, which is the result we want the model to obtain by segmentation. non-FTU means that this patch does not contain the result we want. FTU edge indicates only a small fraction of FTU is in this patch (less than 5%).

In this unit area, "encoder3" indicates that this graph shows the features extracted from encoder3 layer of the UNet. The distribution of the points shows how all the features are mapped to the 2D plane after the dimensionality reduction calculation. The number in the lower right corner 13.89 indicates the cluster quality of this feature distribution, with higher numbers indicating a better distribution of clusters (easier to distinguish clusters).

Q4.5 Mask view

This is another part of the visualization (mask) that may appear in the task. It shows all 288 predicted masks, corresponding to 288 input images. In some tasks, this kind of overview of the mask will be provided, and in some tasks another kind of single mask view will be provided (e.g., click on the point to show a single mask instead of the tiles view).

Q4.6 Violin Plot

This is another part of the visualization that may appear in the task. We have just described that the purpose of UNet is to do the segmentation of the FTU (crypt on colon

sample). One metric to evaluate the segmentation is dice score. The higher the dice score, the more accurate the results of the segmentation. This visualization visualizes the statistics of the dice score from a macro view. Similar to a histogram, this type of visualization is called a violin plot or violin chart.

A violin plot can have multiple layers. The symmetry of the left and right sides of the violin shows the distribution of the distance of all immune cells to the nearest blood vessel cell. The vertical line represents the values that occur 95% of the time. The next layer inside, or the box part, represents the lower and upper quartile of the distribution (q1 and q3). The first quartile (lower quartile, q1) is equal to the 25th percentile of the data. The second (middle) quartile is the median of the data and is the line in the box. The third quartile, called upper quartile (q3), is equal to the 75th percentile of the data. These mentions are displayed interactively as the mouse moves over each violin plot. The points on the right side show the actual distribution of all dice scores in detail.

Q4.7 Explanation of terms / Glossary

Legend: Legends identify the meaning of various elements in a data visualization and can be used as an alternative to labeling data directly. Legends are commonly used with data visualizations when there is more than one color or line type being used.

Quartile: In statistics, a quartile is a type of quantile which divides the number of data points into four parts, or quarters, of more-or-less equal size.

Image Segmentation: In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple image segments, also known as image regions or image objects (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. For example, given one colon image, one goal of the segmentation is to find the outline or the area of each crypt unit.

Dimension reduction: is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful

properties of the original data. Here the features might be thousands of dimensions but after dimension reduction it will be 2D data and can be visualized.

U-Net: https://en.wikipedia.org/wiki/U-Net

Colon: The colon is also known as the large bowel or large intestine.

Crypt: The intestinal glands in the colon are often referred to as colonic crypts.

Kidney: The kidneys are two reddish-brown bean-shaped organs found in vertebrates.

FTU User Study Experimental Group

Questions:

Q4.1 Timing

Q4.2 Introduction

In the following tasks, you will see some visualizations. These visualizations show some information from machine learning features and result of a UNet model. The purpose of UNet is to extract the segmentation of the crypt part (or we call it "FTU" below) of the colon sample. Since our colon dataset is small, we applied transfer learning. We first trained the model on the kidney dataset (a relatively large dataset), and then applied transfer learning to the colon dataset.

Based on the features of each layer in the UNet structure and the predicted mask, we performed the following visualization. Here we will give a detailed introduction to all the visualizations that may be encountered so that you can compete the next tasks without any difficulty.

Q4.3 UNet visualization overview

This is an overview of a visualization that might appear below. This visualization shows a UNet structure that mainly contains encoder pipeline, decoder pipeline, upsample pipeline and the final pipeline.

U-Net is a convolutional neural network that was developed for biomedical image segmentation. The network is based on the fully convolutional network and its architecture

was modified and extended to work with fewer training images and to yield more precise segmentations.

Each small unit shows the feature of a specific layer in the UNet. With a dimensionality reduction (2-dimensional) method applied on the features, the features can be mapped to the 2-dimensional plane. 288 images were fed into the UNet model and therefore the prediction contains 288 masks and each layer has 288 feature points. We will further describe the details of one unit in the next paragraph.

Q4.4 UNet visualization detail

A small portion (extracted from the original view) of the visualization is shown below. On each area there are 288 points, which means the features of 288 inputs. All feature points in the visualization are labeled as three categories here: FTU, non-FTU and FTU edge. FTU here means Functional tissue unit, which is the result we want the model to obtain by segmentation. non-FTU means that this patch does not contain the result we want. FTU edge indicates only a small fraction of FTU is in this patch (less than 5%).

In this unit area, "encoder3" indicates that this graph shows the features extracted from encoder3 layer of the UNet. The distribution of the points shows how all the features are mapped to the 2D plane after the dimensionality reduction calculation. The number in the lower right corner 13.89 indicates the cluster quality of this feature distribution, with higher numbers indicating a better distribution of clusters (easier to distinguish clusters).

Q4.5 Mask view

This is another part of the visualization (mask) that may appear in the task. It shows all 288 predicted masks, corresponding to 288 input images. In some tasks, this kind of overview of the mask will be provided, and in some tasks another kind of single mask view will be provided (e.g., click on the point to show a single mask instead of the tiles view).

Q4.6 Violin Plot

This is another part of the visualization that may appear in the task. We have just described that the purpose of UNet is to do the segmentation of the FTU (crypt on colon

sample). One metric to evaluate the segmentation is dice score. The higher the dice score, the more accurate the results of the segmentation. This visualization visualizes the statistics of the dice score from a macro view. Similar to a histogram, this type of visualization is called a violin plot or violin chart.

A violin plot can have multiple layers. The symmetry of the left and right sides of the violin shows the distribution of the distance of all immune cells to the nearest blood vessel cell. The vertical line represents the values that occur 95% of the time. The next layer inside, or the box part, represents the lower and upper quartile of the distribution (q1 and q3). The first quartile (lower quartile, q1) is equal to the 25th percentile of the data. The second (middle) quartile is the median of the data and is the line in the box. The third quartile, called upper quartile (q3), is equal to the 75th percentile of the data. These mentions are displayed interactively as the mouse moves over each violin plot. The points on the right side show the actual distribution of all dice scores in detail.

Q4.7 Interaction between coupled window visualization

In addition to the basic operations of each visualization, such as dragging and rotating, there is another type of visualization called coupled window visualization. In this visualization, each page consists of multiple basic visualizations side by side (vertically or horizontally). They are not simply placed side by side but there are also some connections between them. For example, in the visualization below, the Unet view, the mask and the Violin plot are displayed together in a coupled window visualization. They all show a visualization of the features from UNet prediction features, and they share some part of the data. So, we can link these same data so the connection between them can be built, and people can easily jump between visualizations based on the connections between certain data points easily. In the example below, if we click on a certain point of a point (an FTU) in the UNet view, we can see that the point is highlighted (larger size). Also, in other layers, the corresponding points of this patch are also enlarged in size. In the violin plot on the right, the point of this FTU is highlighted so that it can be more easily distinguished. Also, on the top of the

visualization, the corresponding mask is shown. If we click this point again, it will restore the previous status; and if we click another patch, this patch will be restored and another patch will be highlighted.

Since the visualization runs on a web server, there may be delays in the interaction. For example, the results are displayed 0.5 seconds after the click. Therefore, please be patient and wait for about 1 second for each operation for the results to be displayed correctly. Also, due to the dense arrangement of cell dots, each click may cover more than one cell and thus the cell highlighting will not be displayed correctly. If this happens, please move the mouse a bit and click again exactly.

Note: Not all visualizations provided are coupled window visualization.

Q4.8 Visualization reset

In the upper left corner of the visualization window there is a RESET button. Pressing this button at any time will make the visualization return to its original state. If the visualization window reports an error, or fails to respond to any operation, click this button to initialize it. Also, if you open the visualization for the first time or when you exit the visualization, you can press this button to restore the visualization to its original state.

Note: Not all visualizations have this reset button.

Q4.9 Explanation of terms / Glossary

Legend: Legends identify the meaning of various elements in a data visualization and can be used as an alternative to labeling data directly. Legends are commonly used with data visualizations when there is more than one color or line type being used.

Quartile: In statistics, a quartile is a type of quantile which divides the number of data points into four parts, or quarters, of more-or-less equal size.

Image Segmentation: In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple image segments, also known as image regions or image objects (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and

easier to analyze. For example, given one colon image, one goal of the segmentation is to find the outline or the area of each crypt unit.

Dimension reduction: is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data. Here the features might be thousands of dimensions but after dimension reduction it will be 2D data and can be visualized.

U-Net: https://en.wikipedia.org/wiki/U-Net

Colon: The colon is also known as the large bowel or large intestine.

Crypt: The intestinal glands in the colon are often referred to as colonic crypts.

Kidney: The kidneys are two reddish-brown bean-shaped organs found in vertebrates.

VCCF User Study Control Group

Questions:

Q4.1 Timing

Q4.2 Introduction

In the following tasks, you will see some visualizations. These visualizations demonstrate some machine learning predictions using various methods and styles. The image of skin samples is fed into the machine learning, and the output is the segmentation of immune cells from the skin samples. Then, based on the outcome of the machine learning prediction, we created these visualizations to demonstrate various types of view and analysis. Here we will give a detailed introduction to all the visualizations that may be encountered, and give some simple examples, so that you can compete the next tasks without any difficulty.

Q4.3 3D view overview

This is a partial view of a 3D interactive visualization that might appear below. The legend for the visualization is shown in the upper left corner. The legend shows the contents of the different cells, damages, and connections in groups. Here mostly we focus on the immune cells and the endothelial cells (blood vessel cells). Below the topic, it mentions

this is the data from region 10. This part of the visualization is 3D visualization and can be interactive, such as dragging, rotating and other operations. In the next part, we will combine the legend and the detailed view of the visualization to introduce more details.

Q4.4 3D view detail

A small portion (extracted from the top-left corner of the original view) of the 3D view visualization is shown below. Here the legend part is on the left and the view of cells is on the right side. According to the legend on the left, we can recognize that there are red endothelial cells on the right side, which can also be simply understood as blood vessel cells. There are also T-helper cells in blue and CD68 cells in yellow. Both are immune cells. We can also see that between these immune cells and endothelial (blood vessel) cells, there is a line of connection. The lines indicate the distance from each immune cell to the nearest blood vessel cell. Each immune cell has at most one thread connecting it to the nearest blood vessel cell.

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q4.5 Histogram

This is another part of the visualization (Histogram) that may appear in the task. We have just described the distance from each immune cell to the blood vessel cell. If we want to know the statistics or distribution of all distances of different immune cells, we can build a histogram to show the result.

The horizontal x-axis of the graph is the distance from the immune cells to the nearest blood vessel cells and the vertical y-axis is the number of immune cells in each distance partition. The upper half is shown by bar graphs, each containing a fitted curve of the same color.

The second half (the below part) shows the real distribution of all distances in detail, or in simple words, it lists all the distances from left to right in order from smallest to largest. If the mouse hovers over a data point, the specific distance information of this point will be

displayed.

Q4.6 Sample question: from the histogram, which distance partition has the most T-helper cells:

- 0 50 μm
- 51 100 μm
- 100 150 μm
- 150 200 μm
- >200 μm

Q4.7 The answer to the previous sample question is 0-50 μm.

Q4.8 Violin Plot

This is another part of the visualization that may appear in the task. We have just described the distance from each immune cell to the vascular cell. This visualization visualizes the statistics of the distances from a macro view. Similar to a histogram, this type of visualization is called a violin plot or violin chart.

A violin plot can have multiple layers. The symmetry of the left and right sides of the violin shows the distribution of the distance of all immune cells to the nearest blood vessel cell. The vertical line represents the values that occur 95% of the time. The next layer inside, or the box part, represents the lower and upper quartile of the distribution (q1 and q3). The first quartile (lower quartile, q1) is equal to the 25th percentile of the data. The second (middle) quartile is the median of the data and is the line in the box. The third quartile, called upper quartile (q3), is equal to the 75th percentile of the data. These mentions are displayed interactively as the mouse moves over each violin plot. The numbers above the violin are the region ID for each skin tissue, and the legend for each different cell is on the right.

Q4.9 Sample question: what is the median value of distance from the highlighted violin chart (orange) in Region 1 (The sample violin visualization above):

Q4.10 The answer to the previous sample question is 58.64572 μm.

Q4.11 Explanation of terms / Glossary

Legend: Legends identify the meaning of various elements in a data visualization and can be used as an alternative to labeling data directly. Legends are commonly used with data visualizations when there is more than one color or line type being used.

Quartile: In statistics, a quartile is a type of quantile which divides the number of data points into four parts, or quarters, of more-or-less equal size.

Image Segmentation: In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple image segments, also known as image regions or image objects (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. For example, given one colon image, one goal of the segmentation is to find the outline or the area of each crypt unit.

T-reg / T-killer / T-helper / Macrophage (CD68): different kinds of immune cells

Endothelial cell: The main type of cell found in the inside lining of blood vessels. This can be simply understood here as blood vessel cells or vascular cells.

VCCF User Study Experimental Group

Questions:

Q4.1 Timing

Q4.2 Introduction

In the following tasks, you will see some visualizations. These visualizations demonstrate some machine learning predictions using various methods and styles. The image of skin samples is fed into the machine learning, and the output is the segmentation of immune cells from the skin samples. Then, based on the outcome of the machine learning prediction, we

created these visualizations to demonstrate various types of view and analysis. Here we will give a detailed introduction to all the visualizations that may be encountered, and give some simple examples, so that you can compete the next tasks without any difficulty.

Q4.3 3D view overview

This is a partial view of a 3D interactive visualization that might appear below. The legend for the visualization is shown in the upper left corner. The legend shows the contents of the different cells, damages, and connections in groups. Here mostly we focus on the immune cells and the endothelial cells (blood vessel cells). Below the topic, it mentions this is the data from region 10. This part of the visualization is 3D visualization and can be interactive, such as dragging, rotating and other operations. In the next part, we will combine the legend and the detailed view of the visualization to introduce more details.

Q4.4 3D view detail

A small portion (extracted from the top-left corner of the original view) of the 3D view visualization is shown below. Here the legend part is on the left and the view of cells is on the right side. According to the legend on the left, we can recognize that there are red endothelial cells on the right side, which can also be simply understood as blood vessel cells. There are also T-helper cells in blue and CD68 cells in yellow. Both are immune cells. We can also see that between these immune cells and endothelial (blood vessel) cells, there is a line of connection. The lines indicate the distance from each immune cell to the nearest blood vessel cell. Each immune cell has at most one thread connecting it to the nearest blood vessel cell.

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q4.5 Histogram

This is another part of the visualization (Histogram) that may appear in the task. We have just described the distance from each immune cell to the blood vessel cell. If we want to know the statistics or distribution of all distances of different immune cells, we can build

a histogram to show the result.

The horizontal x-axis of the graph is the distance from the immune cells to the nearest blood vessel cells and the vertical y-axis is the number of immune cells in each distance partition. The upper half is shown by bar graphs, each containing a fitted curve of the same color.

The second half (the below part) shows the real distribution of all distances in detail, or in simple words, it lists all the distances from left to right in order from smallest to largest. If the mouse hovers over a data point, the specific distance information of this point will be displayed.

Q4.6 Sample question: from the histogram, which distance partition has the most T-helper cells:

- 0 50 μm
- 51 100 μm
- 100 150 μm
- 150 200 μm
- >200 μm

Q4.7 Sample Answer: The answer to the previous sample question is 0 50 μm.

Q4.8 Violin Plot

This is another part of the visualization that may appear in the task. We have just described the distance from each immune cell to the vascular cell. This visualization visualizes the statistics of the distances from a macro view. Similar to a histogram, this type of visualization is called a violin plot or violin chart.

A violin plot can have multiple layers. The symmetry of the left and right sides of the violin shows the distribution of the distance of all immune cells to the nearest blood vessel

cell. The vertical line represents the values that occur 95% of the time. The next layer inside, or the box part, represents the lower and upper quartile of the distribution (q1 and q3). The first quartile (lower quartile, q1) is equal to the 25th percentile of the data. The second (middle) quartile is the median of the data and is the line in the box. The third quartile, called upper quartile (q3), is equal to the 75th percentile of the data. These mentions are displayed interactively as the mouse moves over each violin plot. The numbers above the violin are the region ID for each skin tissue, and the legend for each different cell is on the right.

Q4.9 Sample question: what is the median value of distance from the highlighted violin chart (orange) in Region 1 (The sample violin visualization above):

Q4.10 The answer to the previous sample question is 58.64572 μm.

Q4.11 Interaction between coupled window visualization

In addition to the basic operations of each visualization, such as dragging and rotating, there is another type of visualization called coupled window visualization. In this visualization, each page consists of multiple basic visualizations side by side (vertically or horizontally). They are not simply placed side by side but there are also some connections between them. For example, in the visualization below, the 3D view, the Histogram and the Violin plot are displayed together in a coupled window visualization. They all show a visualization of a machine learning prediction result, and they share some part of the data. So, we can link these same data so the connection between them can be built, and people can easily jump between visualizations based on the connections between certain data points easily. In the example below, if we click on a certain point of the immune cell (T-helper) in the 3D view, we can see that this cell size becomes larger and is highlighted. Also, in the Histogram below, the corresponding point of this cell is enlarged in size. In the violin plot on the right, the violin plot corresponding to the T-helper type in that region (Region 3) is highlighted so that it can be more easily distinguished. If we click this point again, it will restore the previous status; and if we click another immune cell, this T-helper cell will be restored and

another cell will be highlighted.

Since the visualization runs on a web server, there may be delays in the interaction. For example, the results are displayed 0.5 seconds after the click. Therefore, please be patient and wait for about 1 second for each operation for the results to be displayed correctly. Also, due to the dense arrangement of cell dots, each click may cover more than one cell and thus the cell highlighting will not be displayed correctly. If this happens, please move the mouse a bit and click again exactly.

Note: Not all visualizations provided are coupled window visualization.

Q4.12 Visualization reset

In the upper left corner of the visualization window there is a RESET button. Pressing this button at any time will make the visualization return to its original state. If the visualization window reports an error, or fails to respond to any operation, click this button to initialize it. Also, if you open the visualization for the first time or when you exit the visualization, you can press this button to restore the visualization to its original state.

Note: Not all visualizations have this reset button.

Q4.13 Explanation of terms / Glossary

Legend: Legends identify the meaning of various elements in a data visualization and can be used as an alternative to labeling data directly. Legends are commonly used with data visualizations when there is more than one color or line type being used.

Quartile: In statistics, a quartile is a type of quantile which divides the number of data points into four parts, or quarters, of more-or-less equal size.

Image Segmentation: In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple image segments, also known as image regions or image objects (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. For example, given one colon image, one goal of the segmentation is to find the outline or the area of each crypt unit.

T-reg / T-killer / T-helper / Macrophage (CD68): different kinds of immune cells

Endothelial cell: The main type of cell found in the inside lining of blood vessels. This can be simply understood here as blood vessel cells or vascular cells.

A.4 Task 1

FTU User Study Control Group

Questions:

Q5.1 Timing

Q5.2 Here is the link to the visualizations in task 1. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- Unet view: VISUALIZATION_URL
- Violin plot: VISUALIZATION_URL
- Masks: VISUALIZATION_URL

Q5.3 Timing

Q5.4 In the mask view, the difference between the mask of FTU, non-FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.5 In the UNet view, the difference between the distribution of encoder layers and decoder layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.6 In the UNet view, the difference between the distribution of decoder layers and upsample layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.7 In the UNet view, the difference between the distribution of top layers (e.g., encoder/decoder 0 & 1) and bottom layers (e.g., encoder/decoder 4 and center):

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar

- Mostly similar

Q5.8 In the violin view, the difference between the accuracy of FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.9 Which methods will you choose to improve the ML accuracy?

- Add more data
- Feature engineering (e.g., feature transformation, feature creation)
- Feature selection
- Ensemble different models
- Finetune the hyperparameter
- Model fine tuning
- Other methods

Q5.10 If you choose other methods, please specify here: (if not, please input N/A)

Q5.11 Here, we tried to apply transfer learning to transfer the knowledge learned from the source dataset (kidney, large dataset) to the target dataset (colon, small dataset). In the transfer learning process, will you suggest keeping some layers frozen and only train the rest layers?

- Yes
- No

Q5.12 From the given information from the visualization, do you think if you could get some hint which layers should be frozen?

- Yes
- No

Q5.13 Suppose we decide to freeze some layers in the transfer learning, which layers will you suggest we should try first to freeze in the first several experiments?

For the last row - final layers, the three options are 0 1 & 2 (left to right), instead of top/mid/bottom.

	Top layers (0 & 1)	Mid layer (2 & 3)	Bottom layer (4 and center)
Encoder layers	☐	☐	☐
Decoder layers	☐	☐	☐
Upsample layers	☐	☐	☐
Final layers (three options are 0 1 & 2)	☐	☐	☐

FTU User Study Experimental Group

Questions:

Q5.1 Timing

Q5.2 Here is the link to the visualizations in task 1. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- A. Unet view: VISUALIZATION_URL
- B. Violin plot: VISUALIZATION_URL
- C. Masks: VISUALIZATION_URL

Q5.3 Timing

Q5.4 In the mask view, the difference between the mask of FTU, non-FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.5 In the UNet view, the difference between the distribution of encoder layers and decoder layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.6 In the UNet view, the difference between the distribution of decoder layers and upsample layers:

- Mostly different

- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.7 In the UNet view, the difference between the distribution of top layers (e.g., encoder/decoder 0 & 1) and bottom layers (e.g., encoder/decoder 4 and center):

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.8 In the violin view, the difference between the accuracy of FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q5.9 Which methods will you choose to improve the ML accuracy?

- Add more data

- Feature engineering (e.g., feature transformation, feature creation)
- Feature selection
- Ensemble different models
- Finetune the hyperparameter
- Model fine tuning
- Other methods

Q5.10 If you choose other methods, please specify here: (if not, please input N/A)

Q5.11 Here, we tried to apply transfer learning to transfer the knowledge learned from the source dataset (kidney, large dataset) to the target dataset (colon, small dataset). In the transfer learning process, will you suggest keeping some layers frozen and only train the rest layers?

- Yes
- No

Q5.12 From the given information from the visualization, do you think if you could get some hint which layers should be frozen?

- Yes
- No

Q5.13 Suppose we decide to freeze some layers in the transfer learning, which layers will you suggest we should try first to freeze in the first several experiments?

	Top layers (0 & 1)	Mid layer (2 & 3)	Bottom layer (4 and center)
Encoder layers	☐	☐	☐
Decoder layers	☐	☐	☐
Upsample layers	☐	☐	☐
Final layers (three options are 0 1 & 2)	☐	☐	☐

VCCF User Study Control Group

Questions:

Q5.1 Timing

Q5.2 Here is the link to the visualizations in task 1. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- A. 3D view: VISUALIZATION_URL
- B. Histogram: VISUALIZATION_URL
- C. Violin plot: VISUALIZATION_URL

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q5.3 Timing

Q5.4 Which immune cell types have the shortest average distance from the endothelial cells?

- CD68
- T-Helper

- T-Reg
- T-Killer

Q5.5 Which immune cell types have the longest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q5.6 Which immune cell types have the distances all within 150 μm to the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q5.7 How many T-reg cells have a distance greater than 100 μm to the nearest endothelial cells?

Q5.8 What is the median distance of T-helper cells to the nearest endothelial cells?

Q5.9 What is the upper distribution quartile of the distance from T-killer to the nearest endothelial cells?

Q5.10 Please find a specific endothelial cell (x: 3155.287 / y: 1175.2 / z: 57.2), and there should be one T-helper cell connecting to it. Let's call it T0.

What is the distance from T0 to its nearest endothelial cell (x: 3155.287 / y: 1175.2 / z: 57.2)? *Hint: please refer to the x,y,z axis value in the 3D view*

Q5.11 Continued from the previous question Now we have the T-helper cell T0. Is the distance from T0 to the nearest endothelial cell (the answer to the previous question) greater or less than the median distance of the T-helper in region 7? *Hint: check violin chart*

- Greater
- Equal
- Less

Q5.12 Continued from the previous question Now we have the T-helper cell T0. Another T-helper T1, has the smallest distance greater than T0's distance. What is T1's distance to its nearest endothelial cell? *Hint: check histogram*

VCCF User Study Experimental Group

Questions:

Q5.1 Timing

Q5.2 Here is the link to the visualizations in task 1. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- A. 3D view: VISUALIZATION_URL
- B. Histogram: VISUALIZATION_URL
- C. Violin plot: VISUALIZATION_URL

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q5.3 Timing

Q5.4 Which immune cell types have the shortest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q5.5 Which immune cell types have the longest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q5.6 Which immune cell types have the distances all within 150 μm to the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q5.7 How many T-reg cells have a distance greater than 100 μm to the nearest endothelial cells?

Q5.8 What is the median distance of T-helper cells to the nearest endothelial cells?

Q5.9 What is the upper distribution quartile of the distance from T-killer to the nearest endothelial cells?

Q5.10 Please find a specific endothelial cell (x: 3155.287 / y: 1175.2 / z: 57.2), and there should be one T-helper cell connecting to it. Let's call it T0. What is the distance from T0 to its nearest endothelial cell (x: 3155.287 / y: 1175.2 / z: 57.2)? *Hint: please refer to the x,y,z axis value in the 3D view*

Q5.11 Continued from the previous question Now we have the T-helper cell T0. Is the distance from T0 to the nearest endothelial cell (the answer to the previous question) greater or less than the median distance of the T-helper in region 7? *Hint: check violin chart*

- Greater
- Equal
- Less

Q5.12 Continued from the previous question Now we have the T-helper cell T0. Another T-helper T1, has the smallest distance greater than T0's distance. What is T1's distance to its nearest endothelial cell? *Hint: check histogram*

A.5 Task 2

FTU User Study Control Group

Questions:

Q6.1 Timing

Q6.2 Here is the link to the visualizations in task 2. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- A. Unet view: VISUALIZATION_URL
- B. Violin plot: VISUALIZATION_URL
- C. Masks: VISUALIZATION_URL

Q6.3 Timing

Q6.4 In the mask view, the difference between the mask of FTU, non-FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.5 In the UNet view, the difference between the distribution of encoder layers and decoder layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.6 In the UNet view, the difference between the distribution of decoder layers and upsample layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.7 In the UNet view, the difference between the distribution of top layers (e.g., encoder/decoder 0 & 1) and bottom layers (e.g., encoder/decoder 4 and center):

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.8 In the violin view, the difference between the accuracy of FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar

- Mostly similar

Q6.9 Here, we tried to apply transfer learning to transfer the knowledge learned from the source dataset (kidney, large dataset) to the target dataset (colon, small dataset). In the transfer learning process, will you suggest keeping some layers frozen and only train the rest layers?

- Yes
- No

Q6.10 From the given information from the visualization, do you think if you could get some hint which layers should be frozen?

- Yes
- No

Q6.11 Suppose we decide to freeze some layers in the transfer learning, which layers will you suggest we should try first to freeze in the first several experiments?

	Top layers (0 & 1)	Mid layer (2 & 3)	Bottom layer (4 and center)
Encoder layers	☐	☐	☐
Decoder layers	☐	☐	☐
Upsample layers	☐	☐	☐
Final layers (three options are 0 1 & 2)	☐	☐	☐

FTU User Study Experimental Group

Questions:

Q6.1 Timing

Q6.2 Here is the link to the visualizations in task 2. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- Link to the visualizations: http://bahamut.westus2.cloudapp.azure.com:8752/

Q6.3 Timing

Q6.4 In the mask view, the difference between the mask of FTU, non-FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.5 In the UNet view, the difference between the distribution of encoder layers and decoder layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.6 In the UNet view, the difference between the distribution of decoder layers and upsample layers:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.7 In the UNet view, the difference between the distribution of top layers (e.g., encoder/decoder 0 & 1) and bottom layers (e.g., encoder/decoder 4 and center):

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.8 In the violin view, the difference between the accuracy of FTU and FTU edges is:

- Mostly different
- Somewhat different
- Neutral
- Somewhat similar
- Mostly similar

Q6.9 Here, we tried to apply transfer learning to transfer the knowledge learned from the source dataset (kidney, large dataset) to the target dataset (colon, small dataset). In the transfer learning process, will you suggest keeping some layers frozen and only train the rest layers?

- Yes
- No

Q6.10 From the given information from the visualization, do you think if you could get some hint which layers should be frozen?

- Yes
- No

Q6.11 Suppose we decide to freeze some layers in the transfer learning, which layers will you suggest we should try first to freeze in the first several experiments?

	Top layers (0 & 1)	Mid layer (2 & 3)	Bottom layer (4 and center)
Encoder layers	☐	☐	☐
Decoder layers	☐	☐	☐
Upsample layers	☐	☐	☐
Final layers (three options are 0 1 & 2)	☐	☐	☐

VCCF User Study Control Group

Questions:

Q6.1 Timing

Q6.2 Here is the link to the visualizations in task 2. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- A. 3D view: VISUALIZATION_URL
- B. Histogram: VISUALIZATION_URL
- C. Violin plot: VISUALIZATION_URL

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q6.3 Timing

- First Click
- Last Click
- Page Submit
- Click Count

Q6.4 Which immune cell types have the shortest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.5 Which immune cell types have the longest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.6 Which immune cell types have the distances all within 150 μm to the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.7 How many T-reg cells have a distance greater than 100 μm to the nearest endothelial cells?

Q6.8 What is the median distance of T-reg cells to the nearest endothelial cells?

Q6.9 What is the upper distribution quartile of the distance from T-killer to the nearest endothelial cells?

Q6.10 Please find a specific endothelial cell (x: 3937.7 / y: 1186.9 / z: 57.2), and there should be one T-reg cell connecting to it. Let's call it T0. What is the distance from T0 to its nearest endothelial cell (x: 3937.7 / y: 1186.9 / z: 57.2)? *Hint: please refer to the x,y,z axis value in the 3D view*

Q6.11 Continued from the previous question Now we have the T-reg cell T0. Is the distance from T0 to the nearest endothelial cell (the answer to the previous question) greater or less than the median distance of the T-reg in region 3? *Hint: check violin chart*

- Greater
- Equal
- Less

Q6.12 Continued from the previous question Now we have the T-reg cell T0. Another T-reg T1, has the smallest distance greater than T0's distance. What is T1's distance to its nearest endothelial cell? *Hint: check Histogram*

VCCF User Study Experimental Group

Questions:

Q6.1 Timing

Q6.2 Here is the link to the visualizations in task 2. Please interact with the visualization and answer the following questions. You can keep the visualizations open when you answer the questions and check any information in the visualization.

- Link: http://bahamut.westus2.cloudapp.azure.com:8751/

Note: If not specified, all distance below refer to the distance to the nearest endothelial cell (blood vessel cell, color is red)

Q6.3 Timing

- First Click
- Last Click
- Page Submit
- Click Count

Q6.4 Which immune cell types have the shortest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.5 Which immune cell types have the longest average distance from the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.6 Which immune cell types have the distances all within 150 μm to the endothelial cells?

- CD68
- T-Helper
- T-Reg
- T-Killer

Q6.7 How many T-reg cells have a distance greater than 100 μm to the nearest endothelial cells?

Q6.8 What is the median distance of T-reg cells to the nearest endothelial cells?

Q6.9 What is the upper distribution quartile of the distance from T-killer to the nearest endothelial cells?

Q6.10 Please find a specific endothelial cell (x: 3937.7 / y: 1186.9 / z: 57.2), and there should be one T-reg cell connecting to it. Let's call it T0. What is the distance from T0 to its nearest endothelial cell (x: 3937.7 / y: 1186.9 / z: 57.2)? *Hint: please refer to the x,y,z axis value in the 3D view*

Q6.11 Continued from the previous question Now we have the T-reg cell T0. Is the distance from T0 to the nearest endothelial cell (the answer to the previous question) greater or less than the median distance of the T-reg in region 3? *Hint: check violin chart*

- Greater
- Equal
- Less

Q6.12 Continued from the previous question Now we have the T-reg cell T0. Another T-reg T1, has the smallest distance greater than T0's distance. What is T1's distance to its nearest endothelial cell? *Hint: check Histogram*

A.6 Post-Questions

FTU User Study Control Group

Questions:

Q7.1 Timing

Q7.2 After participating in this experiment, I feel...

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Satisfied with my task performance	○	○	○	○	○
Like I learned something new	○	○	○	○	○
Like I was able to make good use of the visualization above	○	○	○	○	○

Q7.3 Would you say that you liked these parts of the visualization?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
The controls	○	○	○	○	○
The visual design	○	○	○	○	○
The instructions	○	○	○	○	○
The overview	○	○	○	○	○
The details view	○	○	○	○	○
The interactivity	○	○	○	○	○
Overall	○	○	○	○	○

Q7.4 Would you say that you found these parts of the "Task 0 - Instruction" we provided for the visualizations helpful?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Text instructions	○	○	○	○	○
Images	○	○	○	○	○
Sample questions	○	○	○	○	○
Explanation of terms / Glossary	○	○	○	○	○

Q7.5 Would you say that the visualizations in the task were easy to use?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.6 There is another type of visualization called coupled window visualization... Would you say that compared to the visualizations in task 1 and 2, the coupled window visualizations will be helpful to complete the task?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.7 What did you like about the visualizations?

Q7.8 What did you not like about the visualizations?

Q7.9 What can be done to improve the visualizations?

Q7.10 What is your email address? We need this information so we can send you your Amazon.com gift card code.

Q7.11 This is your Response ID:

Please copy and paste it into the field below.

FTU User Study Experimental Group

Questions:

Q7.1 Timing

Q7.2 After participating in this experiment, I feel...

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Satisfied with my task performance	○	○	○	○	○
Like I learned something new	○	○	○	○	○
Like I was able to make good use of the visualization above	○	○	○	○	○

Q7.3 Would you say that you liked these parts of the visualization?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
The controls	○	○	○	○	○
The visual design	○	○	○	○	○
The instructions	○	○	○	○	○
The overview	○	○	○	○	○
The details view	○	○	○	○	○
The interactivity	○	○	○	○	○
Overall	○	○	○	○	○

Q7.4 Would you say that you found these parts of the "Task 0 - Instruction" we provided for the visualizations helpful?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Text instructions	○	○	○	○	○
Images	○	○	○	○	○
Sample questions	○	○	○	○	○
Explanation of terms / Glossary	○	○	○	○	○

Q7.5 Would you say that the coupled window visualizations were easy to use?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.6 Would you say that compared to the visualizations in task 1, coupled window visualizations in task 2 help you more to complete the task?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.7 What did you like about the coupled window visualizations?

Q7.8 What did you not like about the coupled window visualizations?

Q7.9 What can be done to improve the coupled window visualizations?

Q7.10 What is your email address? We need this information so we can send you your Amazon.com gift card code.

Q7.11 This is your Response ID:

Please copy and paste it into the field below.

VCCF User Study Control Group

Questions:

Q7.1 Timing

Q7.2 After participating in this experiment, I feel...

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Satisfied with my task performance	○	○	○	○	○
Like I learned something new	○	○	○	○	○
Like I was able to make good use of the visualization above	○	○	○	○	○

Q7.3 Would you say that you liked these parts of the visualization?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
The controls	○	○	○	○	○
The visual design	○	○	○	○	○
The instructions	○	○	○	○	○
The overview	○	○	○	○	○
The details view	○	○	○	○	○
The interactivity	○	○	○	○	○
Overall	○	○	○	○	○

Q7.4 Would you say that you found these parts of the "Task 0 - Instruction" we provided for the visualizations helpful?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Text instructions	○	○	○	○	○
Images	○	○	○	○	○
Sample questions	○	○	○	○	○
Explanation of terms / Glossary	○	○	○	○	○

Q7.5 Would you say that the visualizations in the task were easy to use?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.6 There is another type of visualization called coupled window visualization...

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.7 What did you like about the visualizations?

Q7.8 What did you not like about the visualizations?

Q7.9 What can be done to improve the visualizations?

Q7.10 What is your email address? We need this information so we can send you your Amazon.com gift card code.

Q7.11 This is your Response ID:

Please copy and paste it into the field below.

VCCF User Study Experimental Group

Questions:

Q7.1 Timing

Q7.2 After participating in this experiment, I feel...

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Satisfied with my task performance	○	○	○	○	○
Like I learned something new	○	○	○	○	○
Like I was able to make good use of the visualization above	○	○	○	○	○

Q7.3 Would you say that you liked these parts of the visualization?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
The controls	○	○	○	○	○
The visual design	○	○	○	○	○
The instructions	○	○	○	○	○
The overview	○	○	○	○	○
The details view	○	○	○	○	○
The interactivity	○	○	○	○	○
Overall	○	○	○	○	○

Q7.4 Would you say that you found these parts of the "Task 0 - Instruction" we provided for the visualizations helpful?

	Strongly disagree	Somewhat disagree	Neither agree nor disagree	Somewhat agree	Strongly agree
Text instructions	○	○	○	○	○
Images	○	○	○	○	○
Sample questions	○	○	○	○	○
Explanation of terms / Glossary	○	○	○	○	○

Q7.5 Would you say that the visualizations in the task were easy to use?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.6 Would you say that compared to the visualizations in task 1, coupled window visualizations in task 2 help you more to complete the task?

- Strongly disagree
- Somewhat disagree
- Neither agree nor disagree
- Somewhat agree
- Strongly agree

Q7.7 What did you like about the coupled window visualizations?

Q7.8 What did you not like about the coupled window visualizations?

Q7.9 What can be done to improve the coupled window visualizations?

Q7.10 What is your email address? We need this information so we can send you your Amazon.com gift card code.

Q7.11 This is your Response ID:

Please copy and paste it into the field below.

Appendix B

Data and Source Code

B.1 Data:

Here are the important links to the open data used in the projects in the dissertation:

VCCF Raw data on HuBMAP Portal:

https://portal.hubmapconsortium.org/browse/collection/34b068d4a926f77fd98b3d968b6c172f

VCCF Processed data:

https://github.com/hubmapconsortium/MATRICS-A

B.2 Source Code:

Here are the important links to the open source code used in the projects in the dissertation:

VCCF Codes:

https://github.com/hubmapconsortium/vccf-visualization-2022

VCCF Companion Website:

https://hubmapconsortium.github.io/vccf-visualization-2022/

List of Tables

List of Figures

References

Alom, Md Zahangir et al. (2018). "Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation". In: *arXiv preprint arXiv:1802.06955*.

Alom, Md Zahangir et al. (2019). "Recurrent residual U-Net for medical image segmentation". In: *Journal of Medical Imaging* 6.1, pp. 014006–014006.

Andri, Renzo et al. (2017). "YodaNN: An architecture for ultralow power binary-weight CNN acceleration". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 37.1, pp. 48–60.

Androutsopoulos, Ion et al. (2000). "Learning to filter spam e-mail: A comparison of a naive bayesian and a memory-based approach". In: *arXiv preprint cs/0009009*.

Antol, Stanislaw et al. (2015). "Vqa: Visual question answering". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433.

Aparicio, Manuela et al. (2015). "Data visualization". In: *Communication design quarterly review* 3.1, pp. 7–11.

Asahi, Toshiyuki et al. (1995). "Using treemaps to visualize the analytic hierarchy process". In: *Information Systems Research* 6.4, pp. 357–375.

Bau, David et al. (2017). "Network Dissection: Quantifying Interpretability of Deep Visual Representations". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3319–3327.

Bäuerle, Alex et al. (2019). "Net2Vis: Transforming Deep Convolutional Networks into Publication-Ready Visualizations". In: *ArXiv* abs/1902.04394.

Berman, Maxim et al. (2018). "The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4413–4421.

Bethesda, M. D. (2017). *Common Coordinate Framework (CCF) Meeting. (2017)*. Tech. rep. NIH Common Fund.

Bidanta, Supriya et al. (2023). "Functional Tissue Units in the Human Reference Atlas". In: *bioRxiv*.

Börner, Katy (2015). *Atlas of knowledge: Anyone can map*. MIT Press.

Börner, Katy et al. (2014). *Visual insights: A practical guide to making sense of data*. MIT Press.

Börner, Katy et al. (2019). "Data visualization literacy: Definitions, conceptual frameworks, exercises, and assessments". In: *Proceedings of the National Academy of Sciences* 116.6, pp. 1857–1864.

Börner, Katy et al. (2021). "Anatomical structures, cell types and biomarkers of the Human Reference Atlas". In: *Nature cell biology* 23.11, pp. 1117–1128.

Bratko, Ivan (1997). "Machine learning: Between accuracy and interpretability". In: *Learning, networks and statistics*. Springer, pp. 163–177.

Brinton, Willard Cope (1939). *Graphic presentation.*.

Brodbeck, Dominique et al. (2009). "Interactive visualization-A survey". In: *Human machine interaction*. Springer, pp. 27–46.

Caliski, Tadeusz et al. (1974). "A dendrite method for cluster analysis". In: *Communications in Statistics-theory and Methods* 3.1, pp. 1–27.

Cao, Junyue et al. (2019). "The single-cell transcriptional landscape of mammalian organogenesis". In: *Nature* 566.7745, pp. 496–502.

Chang, Chih-Chung et al. (2011). "LIBSVM: A library for support vector machines". In: *ACM transactions on intelligent systems and technology (TIST)* 2.3, pp. 1–27.

Chattopadhay, Aditya et al. (2018). "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks". In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, pp. 839–847.

Chollet, François et al. (2015). *Keras*. https://keras.io.

Çiçek, Özgün et al. (2016). "3D U-Net: learning dense volumetric segmentation from sparse annotation". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*. Springer, pp. 424–432.

Collobert, Ronan et al. (2008). "A unified architecture for natural language processing: Deep neural networks with multitask learning". In: *Proceedings of the 25th international conference on Machine learning*. ACM, pp. 160–167.

Davies, David L et al. (1979). "A cluster separation measure". In: *IEEE transactions on pattern analysis and machine intelligence* 2, pp. 224–227.

Doshi-Velez, Finale et al. (2017). "Towards a rigorous science of interpretable machine learning". In: *arXiv preprint arXiv:1702.08608*.

Drozdzal, Michal et al. (2016). "The importance of skip connections in biomedical image segmentation". In: *International Workshop on Deep Learning in Medical Image Analysis, International Workshop on Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*. Springer, pp. 179–187.

Dumoulin, Vincent et al. (2016). "A guide to convolution arithmetic for deep learning". In: *arXiv preprint arXiv:1603.07285*.

Erhan, Dumitru et al. (2010). "Understanding representations learned in deep architectures". In.

Faul, Franz et al. (2007). "G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences". In: *Behavior research methods* 39.2, pp. 175–191.

Few, Stephen (2004). "Eenie, meenie, minie, moe: selecting the right graph for your message". In: *Intelligent Enterprise* 7, pp. 14–35.

Flandrin, Patrick et al. (2004). "Empirical mode decomposition as a filter bank". In: *IEEE signal processing letters* 11.2, pp. 112–114.

Foa, Roberto Stefan et al. (2017). "The signs of deconsolidation". In: *Journal of Democracy* 28.1, pp. 5–15.

Frank, Eibe et al. (2009). "Weka-a machine learning workbench for data mining". In: *Data mining and knowledge discovery handbook*. Springer, pp. 1269–1277.

Friedman, Nir et al. (1997). "Bayesian network classifiers". In: *Machine learning* 29.2-3, pp. 131–163.

Fukushima, Kunihiko (1980). "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". In: *Biological cybernetics* 36.4, pp. 193–202.

Ghose, Soumya et al. (2022). "Human digital twin: automated cell type distance computation and 3D atlas construction in multiplexed skin biopsies". In: *bioRxiv*, pp. 2022–03.

Gibson, Eli et al. (2018). "NiftyNet: a deep-learning platform for medical imaging". In: *Computer methods and programs in biomedicine* 158, pp. 113–122.

Godwin, Leah L et al. (2021). "Robust and generalizable segmentation of human functional tissue units". In: *bioRxiv*, pp. 2021–11.

Gonzalez-Garcia, Abel et al. (2016). "Do Semantic Parts Emerge in Convolutional Neural Networks?" In: *International Journal of Computer Vision* 126, pp. 476–494.

Goodfellow, Ian et al. (2020). "Generative adversarial networks". In: *Communications of the ACM* 63.11, pp. 139–144.

Gore, Sally A (2018). "A Brief History of Data Visualization (and the role of libraries and librarians)". In: *Journal of eScience Librarianship* 7.1, p. 6.

Gunning, David (2017). "Explainable artificial intelligence (xai)". In: *Defense Advanced Research Projects Agency (DARPA), nd Web* 2.

Hall, Mark et al. (2009). "The WEKA data mining software: an update". In: *ACM SIGKDD explorations newsletter* 11.1, pp. 10–18.

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Hinton, Geoffrey E et al. (2006). "A fast learning algorithm for deep belief nets". In: *Neural computation* 18.7, pp. 1527–1554.

Hintze, Jerry L et al. (1998). "Violin plots: a box plot-density trace synergism". In: *The American Statistician* 52.2, pp. 181–184.

Hjelsvold, Rune et al. (2001). "Web-based personalization and management of interactive video". In: *Proceedings of the 10th international conference on World Wide Web*. ACM, pp. 129–139.

Holmes, Geoffrey et al. (1994). "Weka: A machine learning workbench". In.

Huang, Jonathan et al. (2017). "Speed/accuracy trade-offs for modern convolutional object detectors". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311.

Hubel, David H et al. (1959). "Receptive fields of single neurones in the cat's striate cortex". In: *The Journal of physiology* 148.3, pp. 574–591.

— (1968). "Receptive fields and functional architecture of monkey striate cortex". In: *The Journal of physiology* 195.1, pp. 215–243.

HuBMAP (2019). *Common Coordinate Framework Workshop (CCFWS-01). (2019)*. Tech. rep. Indiana University.

Huron, Samuel et al. (2014). "Constructing visual representations: Investigating the use of tangible tokens". In: *IEEE transactions on visualization and computer graphics* 20.12, pp. 2102–2111.

Ingre, David et al. (2016). *Engineering communication: A practical guide to workplace communications for engineers*. Cengage Learning.

Ishibuchi, Hisao et al. (2007). "Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning". In: *International Journal of Approximate Reasoning* 44.1, pp. 4–31.

Jain, Yashvardhan et al. (2023). "Segmentation of human functional tissue units in support of a Human Reference Atlas". In: *Communications Biology* 6.1, p. 717.

Jeddi, Ahmadreza et al. (2020). "Learn2perturb: an end-to-end feature perturbation learning to improve adversarial robustness". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1241–1250.

Kahng, Minsuk et al. (2018). "Gan lab: Understanding complex deep generative models using interactive visual experimentation". In: *IEEE transactions on visualization and computer graphics* 25.1, pp. 1–11.

Kalman, Dan (1996). "A singularly valuable decomposition: the SVD of a matrix". In: *The college mathematics journal* 27.1, pp. 2–23.

Karpathy, Andrej (2014). "Convnetjs: Deep learning in your browser (2014)". In: *URL http://cs. stanford. edu/people/karpathy/convnetjs*.

Khan, Salman et al. (2022). "Transformers in vision: A survey". In: *ACM computing surveys (CSUR)* 54.10s, pp. 1–41.

Kingma, Diederik P et al. (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

Knuth, Donald Ervin (1997). *The art of computer programming*. Vol. 3. Pearson Education.

Krizhevsky, Alex et al. (2009). "Learning multiple layers of features from tiny images". In.

Krizhevsky, Alex et al. (2012). "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*, pp. 1097–1105.

Kruger, Norbert et al. (2012). "Deep hierarchies in the primate visual cortex: What can we learn for computer vision?" In: *IEEE transactions on pattern analysis and machine intelligence* 35.8, pp. 1847–1871.

Ku, Wenfu et al. (1995). "Disturbance detection and isolation by dynamic principal component analysis". In: *Chemometrics and intelligent laboratory systems* 30.1, pp. 179–196.

Le Cun, Yann et al. (1989). "Handwritten digit recognition: Applications of neural network chips and automatic learning". In: *IEEE Communications Magazine* 27.11, pp. 41–46.

LeCun, Yann et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324.

LeCun, Yann et al. (2010). "MNIST handwritten digit database". In: *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist* 2, p. 18.

LeCun, Yann et al. (2015). "Deep learning". In: *nature* 521.7553, pp. 436–444.

LeNail, Alexander (2019). "NN-SVG: Publication-ready neural network architecture schematics". In: *Journal of Open Source Software* 4.33, p. 747.

Liang, Ming et al. (2015). "Recurrent convolutional neural network for object recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3367–3375.

Lin, Min et al. (2013). "Network In Network". In: *CoRR* abs/1312.4400.

Lin, Tsung-Yi et al. (2014). "Microsoft coco: Common objects in context". In: *European conference on computer vision*. Springer, pp. 740–755.

Litjens, Geert et al. (2017). "A survey on deep learning in medical image analysis". In: *Medical image analysis* 42, pp. 60–88.

Long, Jonathan et al. (2015). "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440.

Maaten, Laurens van der et al. (2008). "Visualizing data using t-SNE". In: *Journal of machine learning research* 9.Nov, pp. 2579–2605.

Mahendran, Aravindh et al. (2016). "Visualizing deep convolutional neural networks using natural pre-images". In: *International Journal of Computer Vision* 120.3, pp. 233–255.

Manassi, Mauro et al. (2013). "When crowding of crowding leads to uncrowding". In: *Journal of Vision* 13.13, pp. 10–10.

Mane, Ketan K et al. (2004). "Mapping topics and topic bursts in PNAS". In: *Proceedings of the National Academy of Sciences* 101.suppl 1, pp. 5287–5290.

McInnes, Leland et al. (2018). "Umap: Uniform manifold approximation and projection for dimension reduction". In: *arXiv preprint arXiv:1802.03426*.

McNabb, David E (2015). *Research methods in public administration and nonprofit management*. Routledge.

Milletari, Fausto et al. (2016). "V-net: Fully convolutional neural networks for volumetric medical image segmentation". In: *2016 fourth international conference on 3D vision (3DV)*. Ieee, pp. 565–571.

Minaee, Shervin et al. (2021). "Image segmentation using deep learning: A survey". In: *IEEE transactions on pattern analysis and machine intelligence* 44.7, pp. 3523–3542.

Mnih, Volodymyr et al. (2013). "Playing atari with deep reinforcement learning". In: *arXiv preprint arXiv:1312.5602*.

Moorthy, Jayashree et al. (2022). "A Survey on Medical Image Segmentation Based on Deep Learning Techniques". In: *Big Data and Cognitive Computing* 6.4, p. 117.

Nguyen, Anh et al. (2016a). "Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks". In: *arXiv preprint arXiv:1602.03616*.

Nguyen, Anh et al. (2016b). "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks". In: *Advances in Neural Information Processing Systems*, pp. 3387–3395.

Nguyen, Anh et al. (2019). "Understanding Neural Networks via Feature Visualization: A survey". In: *arXiv preprint arXiv:1904.08939*.

Oktay, Ozan et al. (2018). "Attention u-net: Learning where to look for the pancreas". In: *arXiv preprint arXiv:1804.03999*.

Olah, Chris et al. (2017). "Feature visualization". In: *Distill* 2.11, e7.

Pascanu, Razvan et al. (2013). "On the difficulty of training recurrent neural networks". In: *International conference on machine learning*. Pmlr, pp. 1310–1318.

Pearson, Karl (1895). "X. Contributions to the mathematical theory of evolution.II. Skew variation in homogeneous material". In: *Philosophical Transactions of the Royal Society of London.(A.)* 186, pp. 343–414.

PRB (2004). *2004 World Population Data Sheet*. https://www.prb.org/resources/2004-world-population-data-sheet/.

Purnamasari, Helen Prima Dewi et al. (2014). "Clickable and interactive video system using HTML5". In: *The International Conference on Information Networking 2014 (ICOIN2014)*. IEEE, pp. 232–237.

Qin, Zhuwei et al. (2018). "How convolutional neural network see the world-A survey of convolutional neural network visualization methods". In: *arXiv preprint arXiv:1804.11191*.

Raschka, Sebastian (2018). "MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack." In: *J. Open Source Software* 3.24, p. 638.

Rastegari, Mohammad et al. (2016). "Xnor-net: Imagenet classification using binary convolutional neural networks". In: *European Conference on Computer Vision*. Springer, pp. 525–542.

Ritter, Felix et al. (2011). "Medical image analysis". In: *IEEE pulse* 2.6, pp. 60–70.

Rivenson, Yair et al. (2019). "PhaseStain: the digital staining of label-free quantitative phase microscopy images using deep learning". In: *Light: Science & Applications* 8.1, p. 23.

Ronneberger, Olaf et al. (2015). "U-net: Convolutional networks for biomedical image segmentation". In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI*

2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Springer, pp. 234–241.

Rousseeuw, Peter J (1987). "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis". In: *Journal of computational and applied mathematics* 20, pp. 53–65.

Ruder, Sebastian (2016). "An overview of gradient descent optimization algorithms". In: *arXiv preprint arXiv:1609.04747*.

Santos, Claudio Filipi Gonçalves Dos et al. (2022). "Avoiding overfitting: A survey on regularization methods for convolutional neural networks". In: *ACM Computing Surveys (CSUR)* 54.10s, pp. 1–25.

Schafer, R (1998). "MPEG-4: a multimedia compression standard for interactive applications and services". In: *Electronics & communication engineering journal* 10.6, pp. 253–262.

Selvaraju, Ramprasaath R. et al. (2016). "Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization". In: *ArXiv* abs/1610.02391.

Selvaraju et al. (2017). "Grad-cam: Visual explanations from deep networks via gradient-based localization". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626.

Shneiderman, Ben et al. (1998). "Treemaps for space-constrained visualization of hierarchies". In.

Shneiderman, Ben et al. (2001). "Ordered treemap layouts". In: *IEEE Symposium on Information Visualization, 2001. INFOVIS 2001.* IEEE, pp. 73–78.

Simonyan, Karen et al. (2013). "Deep inside convolutional networks: Visualising image classification models and saliency maps". In: *arXiv preprint arXiv:1312.6034*.

Simonyan, Karen et al. (2014). "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556*.

Slutsky, David J (2014). "The effective use of graphs". In: *Journal of wrist surgery* 3.02, pp. 067–068.

Smilkov, D et al. (2017). "A Neural Network Playground-TensorFlow". In: *URL https://playground. tensorflow. org*.

Snyder, Michael P et al. (2019). "Mapping the human body at cellular resolution–the NIH Common Fund Human BioMolecular Atlas program". In: *arXiv preprint arXiv:1903.07231*.

Sorensen, Thorvald (1948). "A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons". In: *Biologiske skrifter* 5, pp. 1–34.

Spielmann, Yvonne (2010). *Video: the reflexive medium*. The MIT Press.

Student (1908). “The probable error of a mean”. In: *Biometrika* 6.1, pp. 1–25.

Tajbakhsh, Nima et al. (2016). “Convolutional neural networks for medical image analysis: Full training or fine tuning?” In: *IEEE transactions on medical imaging* 35.5, pp. 1299–1312.

Talbot, Justin et al. (2009). “EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, pp. 1283–1292.

Tiwari, Vibha (2010). “MFCC and its Applications in Speaker Recognition”. In: *International Journal on Emerging Technologies* 1.1, pp. 19–22.

Van der Maaten, Laurens et al. (2008). “Visualizing data using t-SNE.” In: *Journal of machine learning research* 9.11.

Viegas, Fernanda et al. (2011). “How to make data look sexy”. In: *CNN. com, April* 19, p. 2011.

Vincent, Pascal et al. (2010). “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion”. In: *Journal of machine learning research* 11.Dec, pp. 3371–3408.

Vinyals, Oriol et al. (2019a). *AlphaStar: Mastering the Real-Time Strategy Game StarCraft II*. https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/.

Vinyals, Oriol et al. (2019b). “AlphaStar: Mastering the real-time strategy game StarCraft II”. In: *DeepMind Blog*.

Wang, Hongda et al. (2019). “Deep learning enables cross-modality super-resolution in fluorescence microscopy”. In: *Nature methods* 16.1, pp. 103–110.

Weber, Griffin M et al. (2020). “Considerations for Using the Vasculature as a Coordinate System to Map All the Cells in the Human Body”. In: *Frontiers in Cardiovascular Medicine* 7, p. 29.

Wei, Donglai et al. (2015). “Understanding intra-class knowledge inside cnn”. In: *arXiv preprint arXiv:1507.02379*.

Williams, Paul (2015). *Visual project management*. Lulu. com.

Xie, Saining et al. (2018). “Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 305–321.

Yamashita, Rikiya et al. (2018). “Convolutional neural networks: an overview and application in radiology”. In: *Insights into imaging* 9.4, pp. 611–629.

Yates, W David (2010). *Safety professional's reference and study guide*. CRC Press.

Yosinski, Jason et al. (2015). "Understanding neural networks through deep visualization". In: *arXiv preprint arXiv:1506.06579*.

Zeiler, Matthew D et al. (2014). "Visualizing and understanding convolutional networks". In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer, pp. 818–833.

— (2013). "Visualizing and Understanding Convolutional Networks". In: *ArXiv* abs/1311.2901.

Zhang, Zheng et al. (2016). "Multi-oriented text detection with fully convolutional networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4159–4167.

Zhang, Zijun (2018). "Improved adam optimizer for deep neural networks". In: *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*. Ieee, pp. 1–2.

Zhou, Zongwei et al. (2018). "Unet++: A nested u-net architecture for medical image segmentation". In: *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, pp. 3–11.

Zieliski, Krzysztof et al. (2005). *Software engineering: Evolution and emerging technologies*. Vol. 130. IOS Press.