Volcanoes and Poets: Ink Detection on the Herculaneum Papyri using U-Net Deep Neural Networks

TaeHyung Kwon, Andrew Jensen, and Ting Yu Chang
Department of Civil and Environmental Engineering
Stanford University
kwon1579@stanford.edu ajensen5@stanford.edu tingyuc@stanford.edu

Abstract

The eruption of Mt. Vesuvius in 79 C.E. devastated the ancient Roman towns of Pompeii and Herculaneum, but it also buried a remarkable collection of well-preserved artifacts, including ancient papyrus scrolls, beneath layers of volcanic mud and ash. Reading these fragile carbonized scrolls without unrolling them is a challenging task, like reading a burned book without opening it. In this paper, we outline our contributions towards finding a working model for the Vesuvius Challenge of reading the Herculaneum Papyri using high-resolution X-ray scans and deep learning techniques. Specifically, we focus on the ink detection aspect of the problem. We explore the application of U-Net deep neural networks, including the basic U-Net, Residual U-Net, Attention U-Net, and Residual Attention U-Net, to perform semantic segmentation of the scrolls. We compare the performance of different U-Net architectures and loss functions and optimizing algorithms. Our experiments demonstrate the relative effectiveness of these models for the problem and provide insights into their suitability for this dense segmentation task.

1 Introduction

The Vesuvius Challenge aims to unlock the secrets of a significant historical treasure hidden within the buried ruins of Pompeii and Herculaneum. Following the eruption of Mt. Vesuvius in 79 C.E., a library containing numerous ancient papyrus scrolls was buried beneath layers of volcanic ash and mud. These scrolls, carbonized by the intense heat, are extremely brittle and cannot be unrolled without destroying them. To address this challenge, our research focuses on training a deep convolutional neural network (DCNN) to detect ink residues within high-resolution X-ray scans of the Herculaneum Papyri. Successful detection of ink residues to the extent that the texts could be read again for the first time in thousands of years is a critical step towards solving the larger problem, which if unlocked would more than double the number of available ancient texts currently available for study. The adopted methodology exploits three-dimensional X-ray scans of scroll fragments as algorithmic input to generate prediction masks for ink residue detection. Despite the complexities presented by the carbon-based ink's low contrast against the background in X-ray scans, there is considerable optimism that the deployment of DCNNs may enable us to unravel the secrets veiled within the scrolls.

CS230: Deep Learning, Spring 2023, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2 Related work

The study focuses on enhancing the ink detection process of carbonized scrolls. For a similar problem involving the 1500-year old En-Gedi scroll Seales et al. [2] pinoneered the technique of virtual unwrapping using X-ray scans. Yet, for low contrast ink files, researchers are exploring semantic segmentation, a computer vision technique, using DCNNs. There are challenges in training these deep neural models, such as the vanishing gradient problem. Different solutions, like the usage of modern activation functions and alterations in U-Net architectures, are being pursued to enhance image segmentation accuracy.

One solution which Z. Alom et al. [**Alom**] have proposed a deep residual model that employs identity mapping, resulting in efficient usage of network parameters and better training and testing performance. Similarly, Abraham et al. [1] introduced a solution that combines the Attention U-Net architecture with the Tversky loss function, demonstrating superior performance due to focused feature selection and training. The current study, inspired by these developments, proposes a "Residual Attention U-Net Model" [4] that combines the strength of residual and attention models, using residual blocks to solve the vanishing gradient issue and attention units to focus on relevant neuron activations, improving network generalization. Future sections will further evaluate the training quality of these architectures.

3 Dataset and Features

AttenitThe dataset used in the current study consists of 3D X-ray scans, infrared images, and manually labeled ink masks for three fragments of papyrus, presented as 10,000 '.tif' files with volumes at both 54eV and 88eV scan energy levels. The scans of these smaller papyrus fragments have been prepared to enable the training of ink detection models for use on the full scrolls. Within the dataset these scans are post-processed into "surface volumes" which are also stored as '.tif' files and are then aligned with infrared photos and binary ink masks which serve as the ground truth for ink locations during training.

4 Methods

4.1 Overview

In this research study, we employed four distinct variations of the U-Net architecture: the basic U-Net, Residual U-Net, Attention U-Net, and Residual-Attention U-Net. By employing a range of architectures, we aimed to gain insights into the relative performance and suitability of specific U-Net architectures and loss functions for segmentation task at hand.

4.2 U-Net Model

The basic U-Net architecture, as depicted in Figure 5, comprises a contracting path (left side) and an expansive path (right side). The contracting path follows a standard convolutional network structure, consisting of repeated 3x3 convolutions (unpadded convolutions) with ReLU activation, followed by 2x2 max pooling for downsampling. Max pooling is commonly used in U-Net architectures as a computational cost saving measure. The number of feature channels, or filters, is doubled at each downsampling step.[7]

The expansive path in the network involves upsampling the feature map, reducing the number of feature channels with a 2x2 convolution, and concatenating it with the cropped feature map from the contracting path. Two 3x3 convolutions with ReLU activation are then applied, followed by a 1x1 convolution to map the features to the desired number of classes. Overall, the network consists of 23 convolutional layers. Due to its superior performance relative to the other models we tested we have selected this model as our primary choice.

Residual U-Net is a semantic segmentation neural network which combines strengths of both U-Net and a residual neural network. This combination bring us two benefits. The first is that the residual unit will ease training of the network, secondly the skip connections within a residual unit and between low levels and high levels of the network will facilitate information propagation without

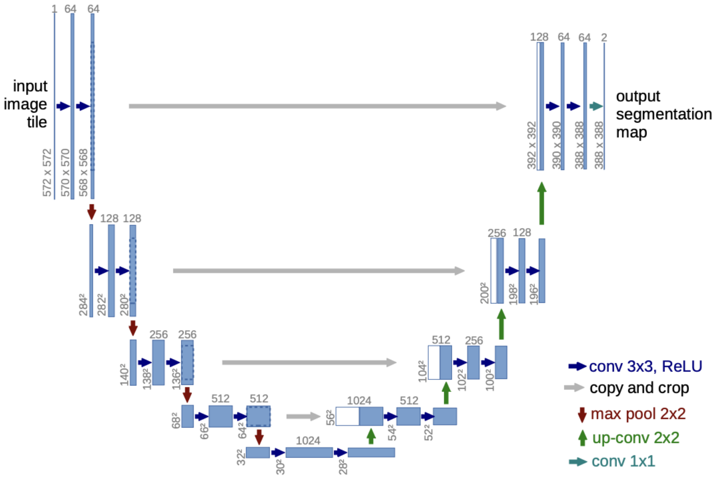

Figure 1: U-Net Architecture

degradation, making it possible to design a neural network with much fewer parameters which can still achieve comparable ever better performance on semantic segmentation tasks.[10]

Attention U-Net is a U-Net variant that incorporates attention mechanisms to focus on important features during segmentation. It uses attention blocks in the skip connections to selectively highlight relevant information. By learning spatial relationships and applying attention weights, Attention U-Net enhances segmentation accuracy by emphasizing informative regions and suppressing noise.[9]

Residual Attention U-Net (RAUNet) adopts an encoder-decoder architecture to get high-resolution masks. The architecture helps reduce the model size and improve inference speed. In the decoder, a new attention module augmented attention module(AAM) is designed to fuse multi-level features and capture global context. Furthermore, transposed convolution is used to carry out upsampling for acquiring refined edges.[6]

4.3 Loss Function

We apply various loss functions on these U-Net architectures. Specifically, we evaluated the effectiveness of binary crossentropy, Dice loss, Tversky loss and IoU loss.

For binary classification, a classification task with two classes — 0 and 1, we have binary crossentropy defined as L in Eq. (1).

The Dice loss defined in Eq. (2) can be used to solve the limitation of cross entropy loss which the loss is calculated as the average of per-pixel loss, and the per-pixel loss is calculated discretely, without knowing whether its adjacent pixels are boundaries or not. [6] It evaluates the similarity between the prediction and the ground truth, which is not affected by the ratio of foreground pixels to background pixels.

Tversky loss sets different weights to false negative (FN) and false positive (FP), which is different from dice loss using the equal weights for FN and FP. To define the Tversky loss function we use the following formula given in Eq. (3). [8]

IoU loss (also called Jaccard loss) is similar to Dice loss and is also used to directly optimize the segmentation metric. Intersection over Union (IoU)-based losses can be unified as Eq. (4). [9]

4.4 Optimization Algorithms

In Machine Learning, an optimization algorithm uses a loss functions to iteratively improve the output prediction of the model. This experiment allowed us to quantitatively measure the performance of different optimizer algorithms previously discussed. Using the quantitative measures of validation accuracy and IoU at the end of training, this study has observed the performance of five different commonly used activation functions: Stochastic Gradient Descent (SGD), Adaptive Moment Estimation (ADAM), Root Mean Square Propagation (RMSProp), Adaptive Gradient Algorithm (AdaGrad) and Adadelta.

5 Experiments, Results, & Discussion

We systematically assessed the U-Net architecture's performance using varied loss functions across distinct modules. The most effective loss function for each module is determined at the end of the model training process. We then compare the results of each optimized U-Net architecture optimized using different loss functions. All code was implemented via the TensorFlow framework.[5]

Performance evaluation relies on two metrics: accuracy and Intersection over Union (IoU). Accuracy gauges how frequently predicted outputs match true values, while IoU, a common metric in image segmentation, evaluates the model's precision in identifying overlapping areas between predicted and actual segmentation masks. The combined use of these metrics affords reliable insights into our model's segmentation prowess [3]. For an individual class, the IoU metric follows per Eq. (5).

Two sets of .tif files were used as test datasets. The training dataset of the three fragments were then split using a 73.5:26.5 ratio into train and validation datasets. The training data is a set of 128 '.tif' images, with sizes of 4000x3094 and 4000x2563 pixels respectively.

5.1 U-Net Loss Functions

Our findings indicate that the U-Net architecture achieves optimal performance when trained using the binary cross entropy loss function. On the other hand, the Residual U-Net variant exhibits improved performance when trained with the Tversky loss function. Similarly, the Attention U-Net variant demonstrates enhanced performance when the IoU loss function is employed. Lastly, the Attention Residual U-Net variant exhibits superior performance when trained using the binary cross entropy loss function. Based on our experimental observations, selecting the training data within the range of layer 15 to layer 34 yielded the best performance in our study. We found that initiating the training process from either the surface layer or the bottom layer resulted in a loss of important features crucial for achieving optimal performance. Also, our observation suggest that training the model using data from layers 15 to 34 yields the best performance. Avoiding either surface of the papyrus is apparently important to preserve key features for the model. Setting the epochs to 15 allows the model to learn from the data, ideally without overfitting, while having 1500 steps per epoch ensures diverse and representative training samples are processed during each stage of training.

Figure 2 shows the qualitative results of the different model architectures. Among the four different models evaluated, the Basic U-Net architecture demonstrated superior performance, warranting a focused discussion on its evaluation using the area under receiver operator characteristic curve (AUC) and area under precision-recall curve (AUPRC) curves, as illustrated in Figure 3.

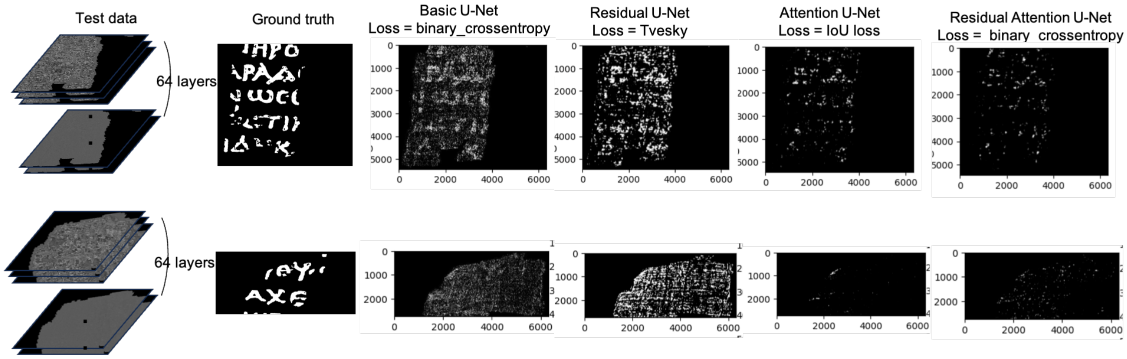

Figure 2: Qualitative results of different U-Net models.

The primary metrics curves in Figure 3 reveal important insights. There is a noticeable gap between the training and validation data. While the training accuracy improves gradually, the validation accuracy fluctuates towards the end. The IoU metric remains relatively stable throughout. The training loss consistently decreases, while the validation loss initially decreases and then fluctuates. These observations provide valuable information about the model's behavior and performance during training and validation.

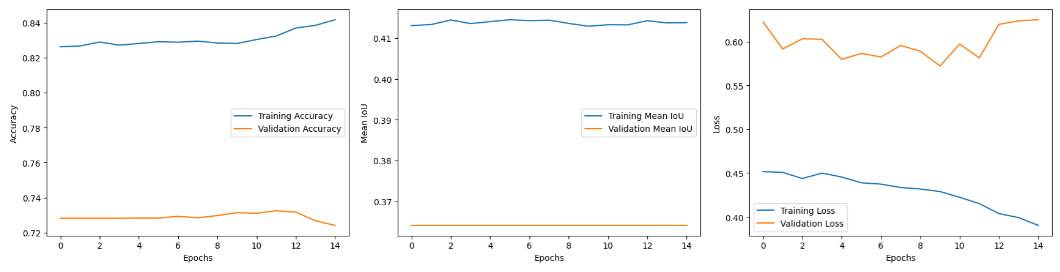

Figure 3: Primary metrics curves of U-Net model.

5.2 Optimizer Algorithms

The performance of prevalent optimization algorithms was quantitatively assessed in this investigation, employing two consistent metrics - validation accuracy and IoU. Emulating prior results, the experiment uniformly applies the binary cross-entropy loss function across all five optimizers under consideration. Empirical findings indicate that the RMSProp optimizer emerged with superior performance in terms of accuracy (74.5%) and IoU (0.42). Remarkably, SGD and Adagrad algorithms shared the runner-up position for accuracy, whereas the ADAM optimizer achieved the highest performance after RMSProp for delineating bounding box overlap with the ground truth bounding box.

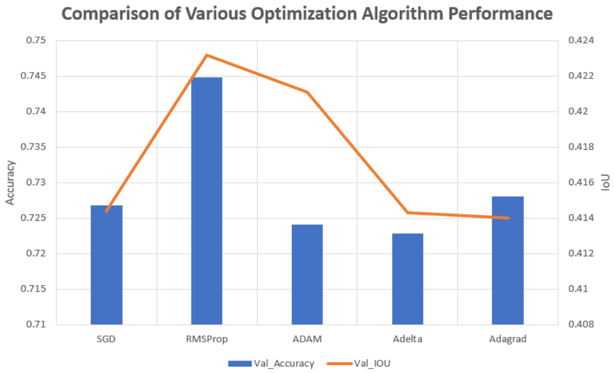

Figure 4: The barplot effectively illustrates the superior performance of the RMSProp optimizer, outperforming other algorithms with a validation accuracy of 74.5% and IoU score of 0.42

Influencing the efficiency of various optimization algorithms, the dataset's properties are pivotal. Herein, surface volumes derived from the 3D X-ray scans of scrolls yield intricate, multi-dimensional data encompassing not only ink presence but also various ink attributes such as density, interaction with papyrus, and thickness. The superior performance of RMSProp in this context can be conjectured to stem from its inherent capability to address noisy, non-convex, and high-dimensional issues.

RMSProp modulates each weight's learning rate individually, leveraging the recent magnitudes of its gradients. This implies its proficiency in handling online and non-stationary problems, potentially equipping it better to handle this dataset's complex features. The sensitivity of RMSProp to gradient oscillations may enable more effective navigation through the data's complex structures compared to ADAM. Despite being a robust optimizer, ADAM amalgamates RMSProp and Momentum benefits. However, in this scenario, Momentum's facet of accelerating gradient vectors in consistent directions, contributing to rapid convergence, might not be advantageous due to the intricate dataset structure. This might induce overshooting in the high-dimensional space and potentially elucidate why RMSProp surpasses ADAM in this context.

6 Conclusion & Future Work

This study embarked on an extensive evaluation of U-Net architecture variations in the context of our 3D ink detection dense segmentation task. The influence of various loss functions and optimization algorithms on model performance were carefully assessed. Empirical results indicate that the basic U-Net architecture trained with a binary crossentropy loss function exhibited superior performance, providing a promising foundation for future work in this area. Further experimentation and optimization led to the discovery of optimal configurations of training data, epoch, and steps-per-epoch settings that yielded optimal performance across the variants of the U-Net models.

The study revealed the importance of aligning the loss functions to the unique characteristics of the U-Net variations, with the binary crossentropy and Tversky loss functions being particularly effective. As for optimization algorithms, RMSProp emerged as the most effective optimizer, demonstrating superior performance in terms of validation accuracy and IoU score. We believe that RMSProp outperformed other optimizers due to its individual weight modulation, adaptive learning rate, and ability to handle noisy, non-convex, and high-dimensional problems, which we believe are all potential issues with our dataset.

Our research provides insight that contributes to the ongoing discourse on image segmentation methods and their application in real-world scenarios. We believe that our systematic approach and rigorous evaluation can serve as a solid foundation for future research in this field, fostering further development of more efficient and robust segmentation methods for 3D imaging.

Future work will involve the utilization of currently experimented loss functions and optimization algorithms with additional modifications to U-Net architecture that might further enhance the performance of U-Net variants. Additionally, given access to more compute resources further experiments on different ranges of layers and configurations of epochs and steps-per-epoch could be pursued to help the model learn the complex patterns within the data more fully.

7 Contributions

Daniel worked on researching, implementing, and testing new candidate model architectures in addition to report writing and video editing; Andrew worked on the problem statement and background, literature review, model parameter testing and ideation, report writing and proofreading; and Ting-Yu worked to set up the notebook data pipeline, performed a literature review, found and set up a usable cloud computing service, tuned model hyperparameters and try different published model architectures, and helped with report writing.

8 Equations and Additional Figures

$$L = -\sum_{i=1}^{2} t_i log(p_i) = -[t_1 log(p_1) + t_2 log(p_2)] = -[tlog(p) + (1-t)log(1-p)] \tag{1}$$

where t_i is the truth value taking a value 0 or 1 and p_i is the Softmax probability for the i^{th} class. Since we have two class 1 and 0, we can have $t_1 = 1$, $t_2 = 0$ $p_1 = p$ and $p_2 = 1 - p$.

$$D = \frac{2\sum_i^w \sum_j^h p_{ij} g_{ij}}{\sum_i^w \sum_j^h p_{ij} + \sum_i^w \sum_j^h g_{ij}} \tag{2}$$

where w, h represent the width and the height of the predictions, p represents the prediction, g represents the ground truth.

$$T(\alpha, \beta) = \frac{\sum_{i=1} N p_{0i} g_{0i}}{\sum_{i=1} N p_{0i} g_{0i} + \alpha \sum_{i=1} N p_{0i} g_{1i} + \beta \sum_{i=1} N p_{1i} g_{0i}} \tag{3}$$

where in the output of the softmax layer, the p_{0i} is the probability of voxel i to contain ink and p_{1i} is the probability of voxel i does not contain ink. Also, for g_{0i} and g_{1i} a value of 1 indicates an ink voxel and 0 a non-ink voxel.

$$L(B, B^{gt}) = 1 - \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} + R(B, B^{gt}) \tag{4}$$

where B and B_{gt} are the predicted box and the target box. The penalty term $R(B,\, B_{gt})$ is designed for the complementary benefit to the original IoU cost.

$$IoU = \frac{True_{pos}}{True_{pos} + False_{pos} + False_{neg}} \tag{5}$$

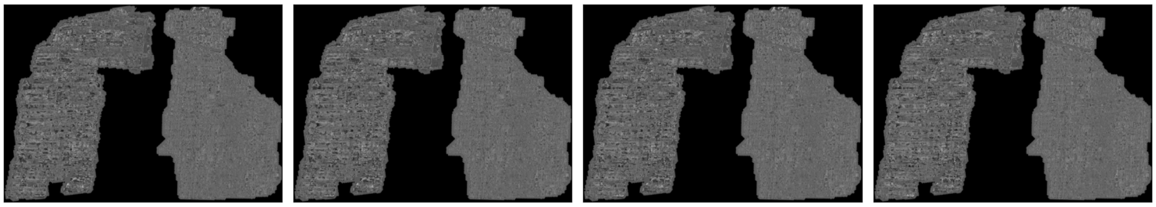

Figure 5: Example Xray scans as the training dataset.

References

[1] Nabila Abraham and Naimul Mefraz Khan. “A Novel Focal Tversky loss function with improved Attention U-Net for lesion segmentation”. In: *IEEE International Symposium on Biomedical Imaging (ISBI)* 5 (2019).

[2] William Brent Seales et al. “From damage to discovery via virtual unwrapping: Reading the scroll from En-Gedi.” In: *Sci. Adv.2,e1601247(2016).DOI:10.1126/sciadv.1601247* (2016).

[3] François Chollet et al. *Keras*. `https://keras.io`. 2015.

[4] Md Imran Hosen and Md Baharul Islam. “Masked Face Inpainting Through Residual Attention UNet”. In: *Innovations in Intelligent Systems and Applications Conference* (2022).

[5] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[6] Zhen-Liang Ni et al. “Raunet: Residual attention u-net for semantic segmentation of cataract surgical instruments”. In: *Neural Information Processing: 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12–15, 2019, Proceedings, Part II*. Springer. 2019, pp. 139–149.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “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. 2015, pp. 234–241.

[8] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. “Tversky loss function for image segmentation using 3D fully convolutional deep networks”. In: *Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 10, 2017, Proceedings 8*. Springer. 2017, pp. 379–387.

[9] Yi-Fan Zhang et al. “Focal and efficient IOU loss for accurate bounding box regression”. In: *Neurocomputing* 506 (2022), pp. 146–157.

[10] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. “Road extraction by deep residual u-net”. In: *IEEE Geoscience and Remote Sensing Letters* 15.5 (2018), pp. 749–753.