Image Segmentation
Automated ROI Detection Labeling

Author:
Behnaaz Firouzeh

January 22, 2024

Contents

Introduction

Bullets and cartridge cases discharged by a firearm provide ballistic samples containing characteristic marks, offering crucial information about the firearm used. Firearm identification relies on the principles of tool mark identification, a forensic discipline dedicated to determining if a specific tool has left a mark on another object. In the context of ballistics, the firearm serves as a tool, transferring distinctive marks onto bullets and cartridge cases (1, 2, 3, 4, 5, 6, 7). To confirm or rule out the involvement of a specific firearm, experts in ballistic identification depend on observing, recognizing, and comparing the distinctive marks found on ballistic specimens (8). Since the 1980s, researchers have increasingly utilized computer technology to advance the field of firearms identification. The surge in database samples, the labor-intensive nature of expert visual evaluations, and the inherent subjectivity in human opinions have all driven the automation of processes related to acquiring and matching ballistic specimens.

Several automated systems, such as the Integrated Ballistic Identification System (IBIS) and the National Integrated Ballistics Information Network (NIBIN), have been developed to streamline sample imaging, delineate regions of interest (ROI), and offer a numerical assessment of probable matches (2, 9, 10). These platforms harness advancements in image acquisition tools, 3D topography, image-processing techniques, and comparison algorithms to improve the precision and effectiveness of ballistic evaluations (2, 4, 7).

The firing process of a firearm imparts distinct marks on the cartridge case, encompassing the breech-face impression (BF), firing pin impression (FP), and firing pin drag (FPD). These marks, unique to each firearm model, serve as essential tool marks for identification. However, the alignment of these marks may vary among exhibits, necessitating their annotation or masking before computational ballistic analysis. The manual execution of masking or coloring, a preparatory function, is currently a laborious and time-consuming process. Automating this task would represent a substantial improvement (11).

Recognizing the challenges posed by the manual and time-consuming nature of the current masking process in forensic laboratories, this study proposes a deep learning-based model. By leveraging advanced computational techniques, the model aims to automatically detect and mask specific features on cartridge case images. The focus on a 9mm caliber ensures uniformity in primer size for preliminary testing. This innovative approach holds the potential to significantly enhance the efficiency of the forensic workflow, reduce subjectivity, and contribute to the ongoing advancements in firearm identification technology.

Problem Statement

Various studies focusing on ballistic cartridge case samples propose solutions to improve matching algorithms for establishing quantitative similarity scores between tool marks. However, a notable subset of these studies lacks well-defined steps for detecting and selecting the region of interest, leading to practical challenges in forensic analysis (11). The current manual masking process, delegated to forensic experts, introduces subjectivity in human interpretation, resulting in variations in the obtained results. This subjectivity, coupled with the exceptionally time-consuming nature of manual masking, becomes increasingly impractical as sample sizes grow.

In response to these challenges, ongoing research and development efforts are dedicated to devising automated solutions aimed at enhancing the efficiency of forensic analysis in firearm identification. Automation seeks to alleviate the subjectivity and time constraints associated with manual processes, thereby contributing to the overall accuracy and effectiveness of firearm identification methodologies.

Objectives

This project aims to develop an algorithm for the automatic masking of cartridge case images, addressing a multi-class semantic segmentation task in computer vision. Multi-class semantic segmentation involves assigning a class label to each pixel in an image, effectively partitioning it into distinct and meaningful regions.

The proposed automated method utilizes deep learning techniques to segment specific features on cartridge case images, contributing to the advancement of firearm identification systems. The identified features include:

1. **Breech-face impression (BF):** Imprinted on the cartridge case head post-propellant ignition, BF characteristics (shape, position, dimensions, and depth) are firearm model-specific.
2. **Aperture shear (AS):** Marks created by the firing pin aperture edge against the breech face, contributing to firearm identification through unique markings.
3. **Firing pin impression (FP):** Formed as the firing pin strikes the primer just before propellant ignition, the FP impression captures the distinctive negative topography of the firing pin's surface on the cartridge case. Its firearm model-specific characteristics, including shape, position, dimensions, and depth, render it a dependable tool mark for precise firearm identification.
4. **Firing pin drag (FPD):** Resulting from the firing pin dragging across the cartridge case surface during the firing process, the FPD is associated with the movement of the firing pin post-primer strike. The distinctive patterns left by the drag are firearm model-specific.
5. **Direction of the firing pin drag:** Referring to the specific path or orientation of the firing pin drag mark on the cartridge case, this element provides insights into firing pin movement during the firing process, this element contributes to firearm model characteristics.

This work aims to automate the masking process, providing a more efficient and accurate means of identifying these crucial firearm features in forensic investigations.

Materials and Methods

4.1 Overview

Several studies utilize preprocessing steps to segment images and distinguish between different types of marks, but the specific methods are often not clearly defined (12, 13, 14, 15). Some studies mention automatic ROI identification using traditional image-processing methods, such as edge detection (Sobel operator or Canny edge detector), along with segmentation operations employing thresholding techniques to isolate the ROI (2). Tai et al. provide detailed insight into an automated mark selection step, combining various operations to identify the primer region and remove the firing pin impression (16).

While machine learning is increasingly applied in various fields for tasks like classification or segmentation, the utilization of deep learning-based methods to delineate ROI in cartridge case images remains limited. In medical imaging, deep learning architectures, namely CNNs, faster R-CNN, YOLOV3, FCN-DenseNet, U-Net, etc., have demonstrated success in classifying and segmenting various structures. In (11), a modern deep-learning approach is employed to evaluate the positioning of circular delimiters in cartridge case images. This investigation focuses on two types of cartridge case ROI: the breech face (BF) impression and the firing pin (FP) impression. The proposed solution entails optimizing and training U-Net segmentation models using 2D images from 1195 samples of cartridge cases fired by different 9MM firearms. The results indicate that the proposed U-Net model offers a more accurate segmentation of the real shape of FP and BF.

In this report, a U-Net segmentation model is proposed to execute the automated masking of discernible features within 3D images of cartridge cases discharged by 9MM firearms. Subsequent sections will delve into further details regarding the dataset, the chosen model architecture, the training process, and the evaluation metrics employed.

4.2 Database

The Fadul dataset, sourced from the NIST Ballistics Toolmark Research Database—an openly accessible repository located at `https://tsapps.nist.gov/NRBTD`—serves as the training and evaluation dataset for the proposed method. This dataset comprises both 2D and 3D scans of cartridge cases. In recent years, the National Institute of Standards and Technology (NIST) has advocated for the adoption of 3D images due to their resilience against variations in lighting conditions and their traceability to the International System of Units (SI) unit of length (17). The SI system comprises measurement units based on precise standards derived from both invariant constants of nature (observable and measurable with high accuracy) and a physical artifact. Consequently, measurements of cartridge cases using any instrument can be compared to a known standard, enabling the calibration of instruments to ensure precision. The primary focus of this work is on developing methods for handling 3D data.

For the purpose of this project, a dataset consisting of 40 topography images of cartridge cases has been examined. These cartridge cases were fired from 10 consecutively manufactured 9MM Ruger P95 pistol slides, ensuring a representative and diverse collection for analysis. Each image

within the dataset exhibits common characteristics, boasting a uniform diameter of approximately 3.5 mm. The topography contrast in each image is meticulously illuminated by a virtual light source positioned from the left. The dataset is openly available and can be retrieved from the following link: Dataset Download Link.

4.3 Preprocessing

4.3.1 Image Annotation

In this segmentation project, the training and validation processes involve the use of both the original images and their corresponding ground truth masks. The ground truth masks provide pixel-level annotations, outlining the precise boundaries of the regions of interest within the images. During training, each training image is paired with its ground truth mask. The model learns to predict pixel-wise segmentation masks by comparing its predictions to the ground truth masks.

To construct the groundtruth dataset, the LabelMe annotation tool, a widely-adopted tool designed for semantic image segmentation tasks, is utilized. This tool enables the manual annotation of objects within images through the creation of polygons around them. Each image in the dataset was meticulously annotated to represent a cartridge case, focusing on outlining key regions of interest (ROIs), including the breech-face impression (BF), aperture shear (AS), firing pin impression (FP), and firing pin drag (FPD). Notably, no separate annotation was deemed necessary for the direction of firing pin drag. This omission is justified by the ease with which this directional information can be derived from the detected masks for FP and FPD.

The LabelMe tool allows for the convenient export of annotations in machine-readable formats, such as JSON. These annotations play a crucial role as the ground truth dataset for both training and evaluating the deep learning model. The accuracy and comprehensiveness of these annotations are pivotal factors influencing the model's capacity to learn and generalize from the annotated features during the training process.

In a real-world context, the verification of annotations by domain experts is essential to ensure accuracy. However, it is important to note that, within the specific scope of this proof-of-concept project for the project, I manually annotated the images. The annotations were performed with careful attention to detail based on available guidelines and resources. It is acknowledged that there is room for improvement in accuracy by incorporating domain experts' knowledge in future iterations of the project.

4.3.2 Data Partitioning

To evaluate the model's performance robustly on the limited dataset, a 10-fold cross-validation approach is adopted. This technique ensures that the model's generalizability is assessed across different subsets of the dataset. It is important to note that cross-validation is exclusively used for reporting the model's performance.

In addition to cross-validation, the dataset is split into train and test sets using an 90/10 ratio. The test set remains unseen during training, allowing for a fair assessment of the model's performance on new, unseen data.

4.3.3 Data Augmentation

Data augmentation is employed to enhance the model's ability to generalize across diverse scenarios. The augmentation pipeline is applied to both the images and their corresponding ground truth masks in the training and validation sets. This ensures that the augmentation is consistent across the input and target, maintaining the alignment between the original image and its ground truth. The pipeline implemented using the Albumentations library, consists of the following transformations for the training set:

- **Resize:** Images are resized to a fixed dimension of 512 by 512 pixels using the nearest-neighbor interpolation method (cv2.INTER_NEAREST). This standardizes the input size for the model.
- **Horizontal Flip:** Images undergo a horizontal flip with a probability of 50% (p=0.5). This mirrors the images horizontally, introducing variations in object orientation and aiding the model in learning invariant features.
- **Vertical Flip:** Vertical flips are applied with a probability of 25% (p=0.25). This operation introduces additional diversity by flipping images vertically, capturing variations in object positioning.

These augmentations collectively contribute to a more robust training process, enabling the model to better handle variations in object appearance, orientation, and scale. The resizing ensures uniformity in input dimensions, while horizontal and vertical flips simulate different viewpoints, enhancing the model's ability to generalize to unseen data.

***Note:** For the validation and test sets, only resizing is considered.

4.4 Model Architecture

In this project, U-Net deep-learning architecture is proposed for segmentation due to U-Net outstanding performance observed in similar studies, demonstrating excellence in semantic segmentation tasks (11). Devised by Olaf Ronneberger, Philipp Fischer, and Thomas Brox in 2015 (18), U-Net derives its nomenclature from its unique U-shaped design, encompassing a contracting path (encoder) and an expansive path (decoder) interconnected by skip connections. Tailored explicitly for image segmentation tasks, U-Net places emphasis on the precise classification and segmentation of objects within images.

A particularly compelling feature of U-Net is its effectiveness in scenarios characterized by limited datasets. In the context of this study, with a dataset comprising 40 images—a size considered relatively small for deep learning applications—the intrinsic attributes of the U-Net architecture, including skip connections and the integration of pre-trained encoders, prove invaluable. These attributes empower the model to extract meaningful representations, even in the face of limited examples. The strategic use of skip connections within the U-Net architecture plays a pivotal role in preserving fine-grained details throughout both the encoding and decoding phases—an essential consideration for tasks where capturing intricate details within segmented regions is of paramount significance.

Another notable advantage of adopting U-Net is its expedited convergence during training, making it particularly practical for various applications, including rapid prototyping, efficient processing of large datasets, and accessibility for individuals who do not have access to powerful GPU resources. This characteristic facilitates a more efficient experimentation and iteration process during model development.

The encoder-decoder architecture of U-Net contributes significantly to its efficacy. The contracting path (encoder) adeptly captures abstract features, while the expansive path (decoder) employs up-convolutional layers and skip connections to recover spatial details, fostering a comprehensive understanding of both high-level and low-level features.

Furthermore, the integration of pre-trained encoders, such as EfficientNet-B0, imparts a strategic advantage to the model. By leveraging features learned from a large-scale image classification task (ImageNet), this approach furnishes the model with a robust starting point for learning relevant features. This proves particularly advantageous in scenarios characterized by small datasets, where transfer learning from pre-trained models substantially enhances overall performance.

In essence, the decision to employ U-Net for segmentation in this project is anchored in its well-documented successes, particularly in scenarios with limited data, its swift convergence during

training, and its adeptness at preserving intricate details—qualities that collectively position U-Net as a robust and effective choice for achieving the research objectives at hand. The subsequent subsections provide a detailed exploration of the proposed model architecture.

4.4.1 Encoder (EfficientNet-B0)

In a U-Net architecture, the encoder is responsible for capturing hierarchical features from the input image and gradually reducing the spatial resolution through a series of convolutional and pooling layers. The encoder's primary role is to extract abstract and high-level representations of the input, which are then used by the decoder to generate a segmentation map.

The proposed model architecture leverages the EfficientNet-B0 as the encoder component. EfficientNet is a family of convolutional neural network architectures known for their efficiency in terms of computational resources and model performance. EfficientNet-B0 is a good choice in situations where computational resources are limited, and a balance between model size, training time, and performance is essential. It is particularly well-suited for segmentation and transfer learning scenarios on small to medium-sized datasets.

EfficientNet-B0 is composed of stacked blocks of convolutional layers, each tailored to capture features at various abstraction levels with varying numbers of filters, kernel sizes, and other parameters. A notable innovation in EfficientNet-B0 is the integration of depthwise separable convolutions, dividing the standard convolution into depthwise and pointwise convolutions to reduce parameters and computations, enhancing model efficiency. The architecture incorporates down-sampling operations through strided convolutions or pooling layers, facilitating progressive downsampling to capture features at different scales, from low-level details to high-level semantics in deeper layers. The final layers of the encoder produce feature maps representing learned hierarchical features, encompassing spatial patterns, textures, and structures in the input image. EfficientNet-B0's adaptability is harnessed through pre-training on extensive datasets like ImageNet using transfer learning, allowing the model to grasp generic features. In semantic segmentation tasks, the pre-trained encoder can be fine-tuned on a specific segmentation objective with a more focused dataset.

4.4.2 Decoder

The U-Net decoder plays a crucial role in semantic segmentation. Starting with the low-resolution feature map from the encoder's last layer, the decoder uses upsampling techniques, such as transposed convolutions or bilinear interpolation, to recover lost spatial information due to downsampling. Notably, U-Net incorporates skip connections, directly linking encoder feature maps to decoder layers. This preserves fine details, as skip connections allow information to bypass upsampling, enhancing contextual understanding.

In U-Net's decoder, upsampled feature maps are concatenated with corresponding encoder feature maps. This fusion of low-level and high-level features facilitates accurate predictions by considering both local and global context. Additionally, convolutional layers in the decoder refine and learn intricate patterns from the combined feature maps.

4.4.3 Output Layer and Activation Function

The decoder concludes with a final output layer that produces segmentation masks for each class. The number of output channels in the final layer is determined by the number of classes in the segmentation task, with an additional channel reserved for the background class. The softmax activation function is applied to convert raw model outputs into probabilities for each class, facilitating multi-class segmentation.

4.5 Model Training

4.5.1 Loss Function

The training process leverages the Cross Entropy Loss, an effective loss function tailored for multi-class segmentation challenges. This loss function quantifies the dissimilarity between the predicted segmentation masks and the ground truth, providing guidance for the model to refine its understanding of class boundaries accurately. The formula for Cross Entropy Loss is defined as follows:

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C} y_{i,j} \cdot \log(\hat{y}_{i,j}) \tag{4.1}$$

where:

N is the total number of pixels,

C is the number of classes,

$y_{i,j}$ is a binary indicator of whether class j is the correct classification for pixel i,

$\hat{y}_{i,j}$ is the predicted probability of pixel i belonging to class j.

During training, the gradient of the loss with respect to the network parameters is computed, and backpropagation is employed to update the model weights. This iterative process empowers the model to enhance its predictions progressively.

4.5.2 Optimizer

The Adam optimizer is employed for training the model. Adam dynamically adjusts the learning rates for individual parameters and incorporates momentum-like terms, contributing to accelerated convergence during the optimization process. The initial learning rate is specified as 0.001, and default parameters of the Adam optimizer, including betas=(0.9, 0.999) and eps=1e-08, are employed to optimize the model effectively. Additionally, weight decay is applied to the optimization process with a parameter value of 0. This regularization term is proportional to the magnitude of the model parameters, aiding in preventing overfitting by penalizing large weights. The combination of these features in the Adam optimizer helps enhance the model's generalization and training stability.

4.5.3 Evaluation

Dice Score

In image segmentation, the Dice Score, also known as the Dice coefficient, is a metric commonly used to assess the performance of segmentation models. It measures the similarity or overlap between the predicted segmentation and the ground truth segmentation. In the context of our study, the Dice Score is utilized to evaluate the effectiveness of the U-Net segmentation model in capturing the desired features on 3D images of cartridge cases fired by 9MM firearms. The formula for calculating the Dice Score is given by:

$$Dice\,Score = \frac{2 \times |A \cap B|}{|A| + |B|} \tag{4.2}$$

- A represents the set of pixels in the predicted segmentation.
- B represents the set of pixels in the ground truth segmentation.
- $|A \cap B|$ is the number of pixels common to both A and B.

- $|A|$ and $|B|$ are the total number of pixels in A and B, respectively.

The Dice Score ranges from 0 to 1, where 0 indicates no overlap and 1 indicates a perfect match. A higher Dice Score implies better alignment between the predicted and ground truth segmentations.

4.5.4 Training Procedure

The model is trained through two distinct approaches, with their respective results presented in Figure 6.1 and Figure 6.2. In the first approach, an early stopping condition is implemented, triggered if there is no improvement in performance on the validation set for 10 consecutive Epochs (patience = 10). This strategy optimizes training time by halting the process when no further enhancement is observed, ensuring efficient training while maintaining the model's generalization to unseen data. The second approach entails training with a predefined number of Epochs, with the chosen value set to 100 for this study.

A batch size of 3 is utilized during training. The use of small batch sizes can facilitate faster convergence and better generalization, particularly when working with limited computational resources.

The training process involves minimizing the Cross Entropy Loss 4.5.1. Backpropagation is employed to update the model parameters based on the computed loss. This iterative optimization process enables the model to learn and improve its ability to accurately mask the specified features on cartridge case images.

Throughout training, the model's performance is continuously assessed using the Dice Score metric. The training progress is monitored, and Dice Scores for both the validation and test sets for each Epoch are calculated and logged.

Given the small size of the dataset, consisting of only 40 images for U-Net segmentation, the validation set mirrors the training set. However, a nuanced difference exists in the augmentation process. While the training set undergoes various augmentation strategies as explained in 4.3.3, the validation set experiences only resizing of images. This choice is made to address the dataset's limited size, as incorporating additional augmentation strategies during validation could lead to overfitting. Thus, the validation set serves as a representative sample of the overall dataset, allowing for effective model evaluation without introducing unnecessary variability.

After the training process, the best-performing model is identified based on the highest Dice Score. This model is saved and subsequently applied to the test set for further evaluation. The Dice Score 4.5.3 serves as a crucial metric in selecting the model that exhibits the most accurate and consistent masking performance.

By adhering to these training procedures, the algorithm aims to achieve a high level of accuracy and efficiency in automatically masking cartridge case images. Section 6 will present the results and discuss the performance of the trained model.

Implementation

The experiments were conducted using Python 3 on Google Colab. The segmentation model was implemented with the assistance of the `segmentation_models_pytorch` library (19). This open-source library, based on the PyTorch framework, provides a comprehensive set of pre-trained segmentation models, making it an optimal choice for our image segmentation task.

5.1 Instructions for Running the Code

- **Environment Setup:** Consider using Google Colab for an efficient cloud-based environment.
  - Save the folder in Google Drive for easy mounting and code execution. This folder comprises the following components:
    1. **Data:** This folder houses the original Fadul dataset, downloaded from the NIST Ballistics Toolmark Research Database, along with the corresponding annotation files in JSON format.
    2. **dataset:** Contains folders for images, masks, JSON files, and datasets for both the 10-Fold Split and 90/10 Train/Test Split approaches.
    3. **training:** Contains the trained models, logged scores, and masked images generated by the model.
    4. `utils.py`: Houses classes for data loading, training, and model evaluation.
    5. `Model_Training_Evaluation.ipynb`: This notebook produces masked images using the proposed model.
    6. `Model_Output_Masked_Image.ipynb`: This notebook presents figures of the output masked images using the trained model.
  - Change the Google Colab runtime type to utilize the T4 GPU for enhanced performance.
- **Code Execution:**
  - To build the datasets, train and evaluate the model, run the following Jupyter Notebook: `Model_Training_Evaluation.ipynb`. This notebook utilizes the Trainer and Evaluator classes from the 'utils' module and includes code for the following tasks:
    1. Initializations and Installations
    2. Generating Masks from JSON Annotations
    3. Data Partitioning
       3.1 10-Fold Split
       3.2 Train/Test Split
    4. Module Integration: Importing Trainer and Evaluator Classes
    5. Model Training

5.1 Model Training for 10-Fold Split
5.2 Model Training for Train/Test Split Data
6. Model Evaluation
7. Loading and Analyzing Model Training Scores
7.1 Loading Logged Scores
7.2 Visualization: Performance Curves
8. Model Output: Masked Cartridge Case Image

– To visualize the output masked images using the trained model, run the following Jupyter Notebook: `Model_Output_Masked_Image.ipynb:`. This notebook utilizes the Evaluator class from the 'utils' module and includes code for the following tasks:

1. Initializations and Installations
2. Module Integration
3. Model Output: Masked Cartridge Case Image

Results and Discussions

6.1 Assessing Model Generalizability: Cross-Validation

In this section, the results of the implemented U-Net segmentation model is presented, focusing on its performance across different folds of cross-validation. The goal is to assess the model's generalizability and robustness on diverse subsets of the dataset.

To ensure a comprehensive evaluation, k-fold cross-validation is employed, dividing the dataset into k = 10 distinct folds. The model underwent training and validation 10 times, incorporating an early stopping approach. Each fold served as the validation set exactly once, resulting in a robust estimation of the model's performance on previously unseen data.

This section presents the results of each fold individually, highlighting the performance metrics achieved in terms of Dice Score. This detailed analysis allows us to observe variations in performance across different subsets of the dataset. The Dice Scores for each fold (rounded to the Forth decimal place) are presented below:

- **Fold1:** 0.9692
- **Fold2:** 0.9676
- **Fold3:** 0.9680
- **Fold4:** 0.9736
- **Fold5:** 0.9742
- **Fold6:** 0.9688
- **Fold7:** 0.9616
- **Fold8:** 0.9683
- **Fold9:** 0.9706
- **Fold10:** 0.9668

The average Dice Score across the 10 folds is 0.9689. Additionally, the standard deviation of Dice Scores provides insights into the variability of model performance:

- **Average Dice Score:** 0.9689
- **Standard Deviation:** 0.003 (rounded to the third decimal place)

This communicates that the typical variability observed in the Dice Scores across the folds is approximately $\pm$ 0.003 from the average value of 0.9689.

6.2 Model Performance for Automated Masking

This section evaluates the performance of the proposed automated masking model through two training strategies: early stopping and training for 100 Epochs. The assessment includes a comprehensive analysis of Dice Scores and cross-entropy loss across each Epoch for the validation and test sets. In addition to numerical metrics, visual insights into the model's segmentation capabilities are provided by presenting masked images from the test set.

6.2.1 Performance Curves

The figures, specifically Figure 6.1 and Figure 6.2, illustrate the performance curves that outline Dice Scores and cross-entropy loss throughout the training process for both the validation and test sets, corresponding to early stopping and 100 Epochs, respectively. These visualizations highlight the average scores computed across all validation and test sets for all classes, encompassing breech-face impressions, aperture shear, firing pin impressions, and firing pin drag. Furthermore, individual average scores for each class are presented, providing a comprehensive overview of the model's performance on specific segmentation tasks. Table 6.1 presents a comprehensive comparison of the model's performance using two training strategies: Early stopping at Epoch 38 and 100-Epoch at Epoch 98, which demonstrates the best model performance in terms of Dice Score. The corresponding trained models for the mentioned Epochs are saved for masking new, unseen images. The evaluation is conducted based on Dice Scores, measuring the accuracy of the model's segmentation across different classes. The table provides separate metrics for the validation set and the test set, offering insights into how each strategy performs on both seen and unseen data. Additionally, class-specific Dice Scores, including breech-face impressions, aperture shear, firing pin impressions, and firing pin drag, are detailed, providing a nuanced understanding of the model's proficiency in segmenting specific firearm-related features. The Average Score consolidates the overall performance, highlighting the effectiveness of each strategy in achieving accurate and reliable segmentation results.

- **The Early Stopping Strategy:** Demonstrates a consistent rise in Dice Scores until Epoch 38. Subsequently, a temporary decline in Dice Scores occurs over the next 4 Epochs, followed by an upward trend. Regarding the loss, a continuous decrease is observed until Epoch 38, with a subsequent increase for 4 Epochs before returning to a decreasing trend. The fluctuations in Dice Scores and cross-entropy loss after Epoch 38 may stem from potential overfitting, issues with the learning rate, or variability in the training data. The model might have become overly specialized in capturing nuances of the training set, leading to a temporary performance dip on the validation set. Addressing these challenges, such as adjusting the learning rate, employing regularization techniques, or enhancing training data diversity through augmentation, could help mitigate these fluctuations.

  - **Validation Set:**
    - **Overall Performance:** The model demonstrates excellent segmentation performance on the validation set, achieving a high average Dice Score of 0.98. This score reflects the model's ability to accurately delineate regions of interest in the images.
    - **Class-wise Dice Scores:** At Epoch 38, the average Dice Scores for specific classes are detailed in Table 6.1.
  - **Test Set:**
    - **Overall Performance:** Similar to the validation set, the model maintains strong overall performance on the test set, achieving an average Dice Score of 0.98. This suggests that the model generalizes well to unseen data.
    - **Class-wise Dice Scores:** At Epoch 38, the average Dice Scores for individual classes in the test set are outlined in Table 6.1.

These results imply that the model excels in accurately segmenting certain classes, such as breech-face impressions and firing pin impressions, as reflected by high Dice Scores. However, challenges exist in segmenting classes like aperture shear and firing pin drag, where the model's performance is comparatively lower. The fluctuations in Dice Scores across classes suggest that the model may struggle with certain nuances or variations specific to these classes. Further investigation and potential model adjustments may be warranted to improve performance on the more challenging classes, ensuring a more balanced and reliable segmentation across all classes.

- **The 100-Epoch Training Strategy:** Demonstrates the best Dice Score at Epoch 98, with an average Dice Score of 0.99 and a cross-entropy loss of 0.03. This strategy provides better model performance compared to the early stopping strategy explained above.
  - **Validation Set:**
    - **Overall Performance:** The model demonstrates excellent segmentation performance on the validation set, achieving a high average Dice Score of 0.99. This score reflects the model's ability to accurately delineate regions of interest in the images.
    - **Class-wise Dice Scores:** At Epoch 98, the average Dice Scores for specific classes detailed in Table 6.1
  - **Test Set:**
    - **Overall Performance:** Consistent with the validation set, the model sustains robust overall performance on the test set, achieving an average Dice Score of 0.99. This suggests the model's effective generalization to unseen data.
    - **Class-wise Dice Scores:** Class-specific Dice Scores at Epoch 98 for individual classes in the test set are outlined in Table 6.1.

Table 6.1: Comparison of Model Performance using Early Stopping (Epoch 38) and 100-Epoch (Epoch 98) Training Strategies in Terms of Dice Score

	Early Stopping		**100-Epoch**	
	Validation Set	**Test Set**	**Validation Set**	**Test Set**
Breech-face Impressions	0.99	0.98	0.99	0.98
Aperture Shear	0.91	0.78	0.94	0.76
Firing Pin Impressions	0.98	0.98	0.99	0.98
Firing Pin Drag	0.58	0.80	0.67	0.84
Avg. Score	0.98	0.98	0.99	0.99

6.2.2 Masked Image Visualizations: U-Net Model Masks in Action

To visually evaluate the model's segmentation capabilities, Figures 6.3 to 6.6 showcase segmented images from the test set, featuring a comparison between masked images generated using both the early stopping and 100 Epochs training strategies. These visualizations highlight the model's accuracy in masking key regions of interest, including breech-face impressions, aperture shear, firing pin impressions, firing pin drag, and the direction of the firing pin drag. The color-coded segmentation aids in a clear interpretation of the model's proficiency in distinguishing and accurately masking these intricate features.

Notably, a specific area in the image of Fadul 6-1 3DVM, as shown in Figure 6.3, draws attention due to the model's highlighting in the case of aperture shear. This raises questions about its accuracy when compared to the ground truth. This underscores the significance of involving domain experts in the annotation process, particularly for nuanced features, to ensure the accuracy and reliability of the labeled data. Additionally, increasing the volume of data, further preprocessing, and fine-tuning hyperparameters can contribute to improvement. These visualizations provide valuable insights into the model's performance, prompting considerations for further refinement and validation.

Figure 6.1: Performance Curve: Segmentation with Early Stopping - Early Stopping Achieved at Epoch 38

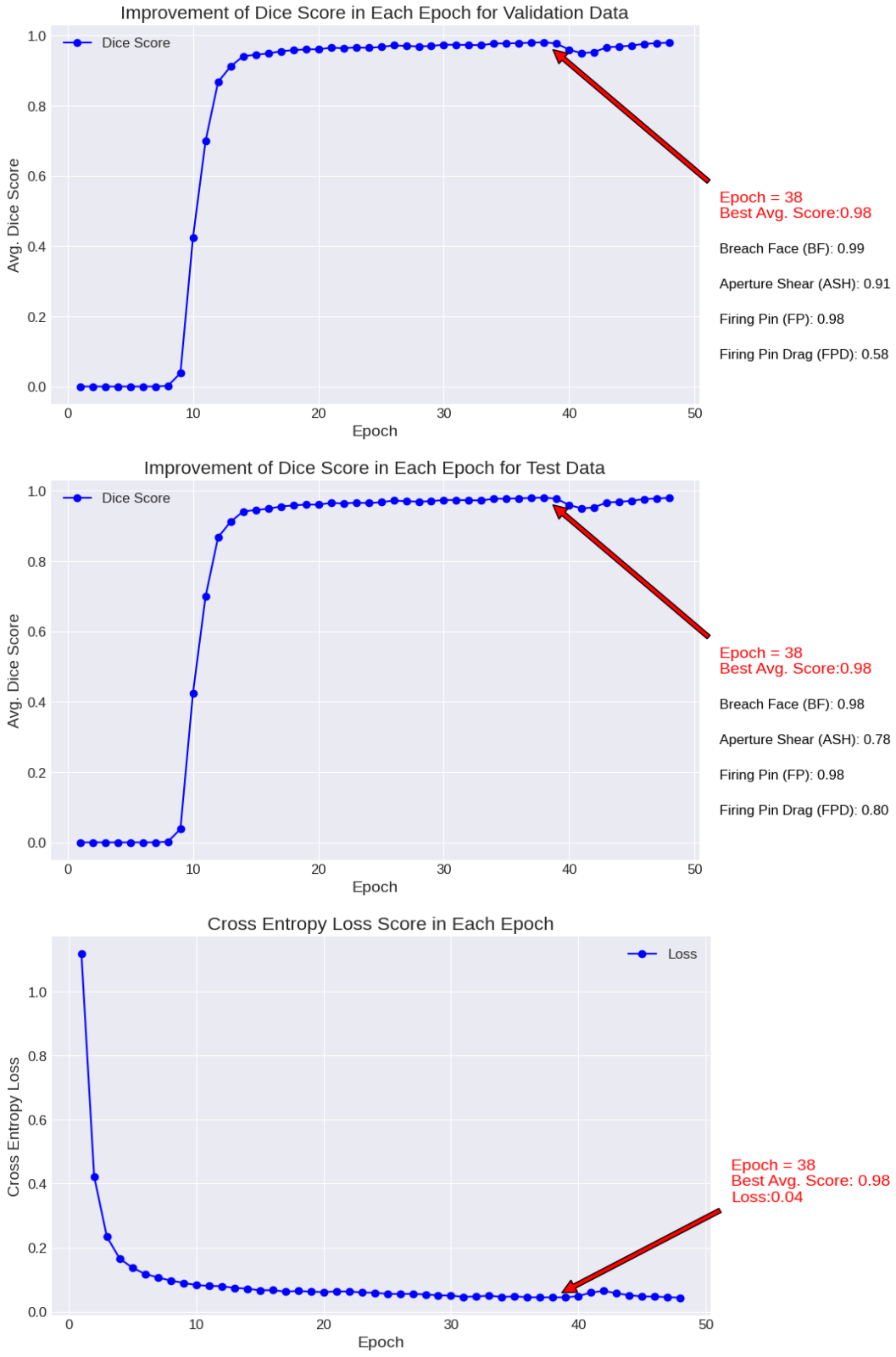


Figure 6.2: Performance Curve: Segmentation with 100 Epochs

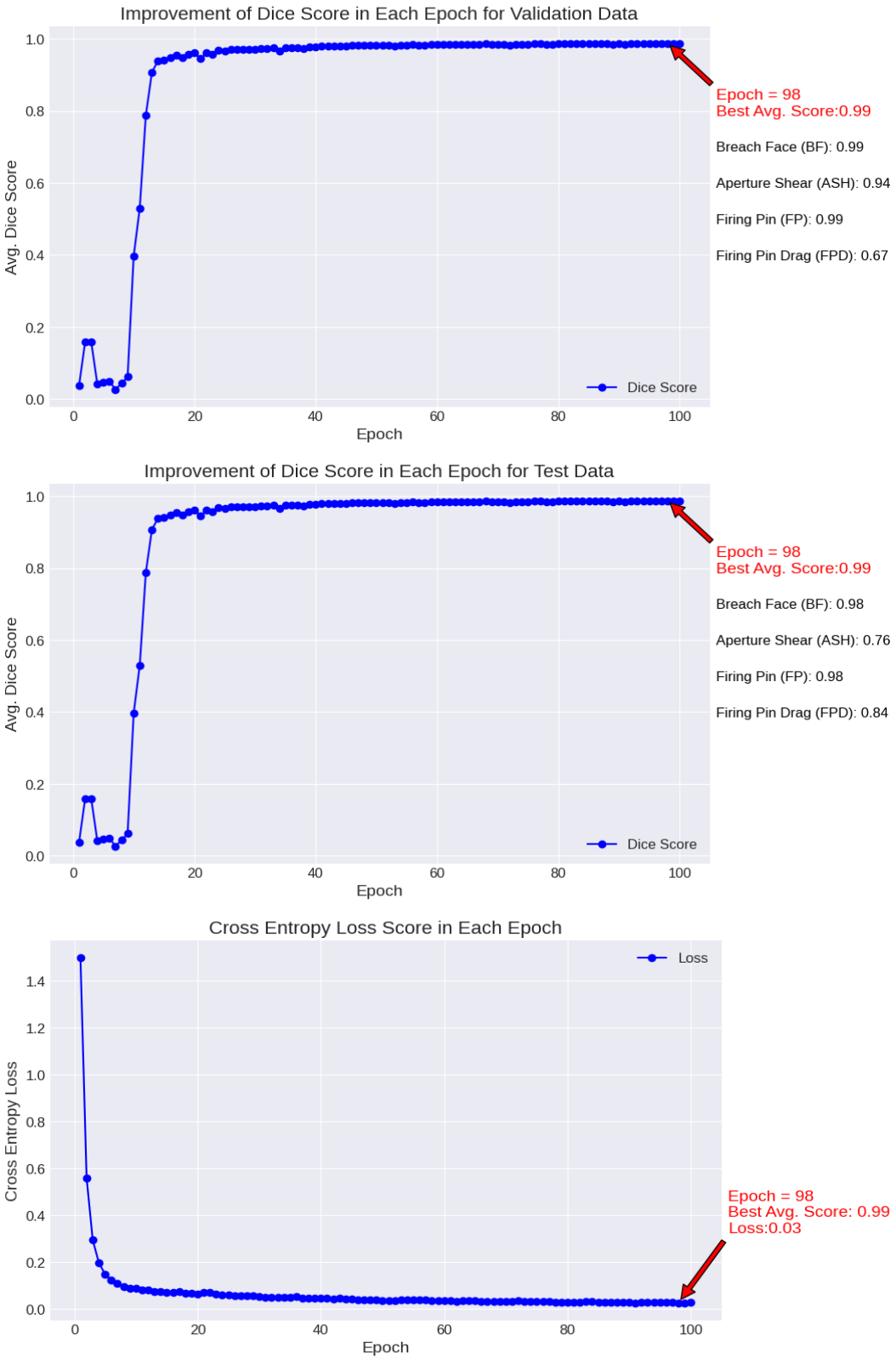
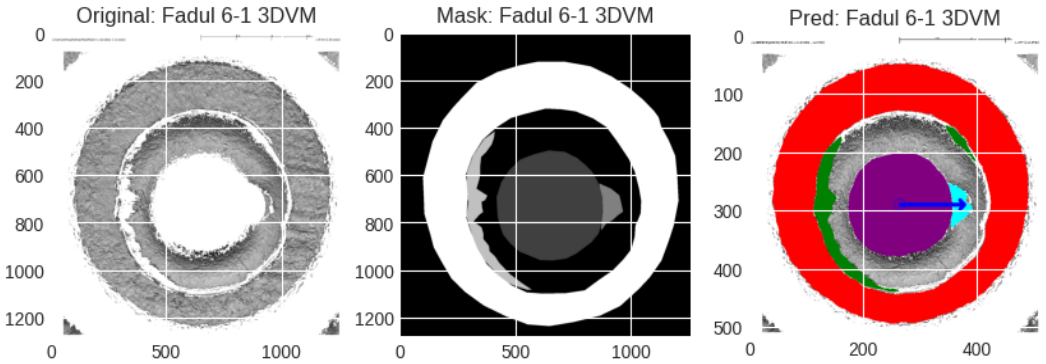


Figure 6.3: Comparison of Test Image Segmentation Results for **Fadul 6-1 3DVM** (breech-face impression in red, aperture shear in green, firing pin impression in purple, firing pin drag in light blue, direction of the firing pin drag indicated by blue arrow)

(a) Test Image Segmentation (Early Stopping): Fadul 6-1 3DVM



(b) Test Image Segmentation (100 Epochs): Fadul 6-1 3DVM

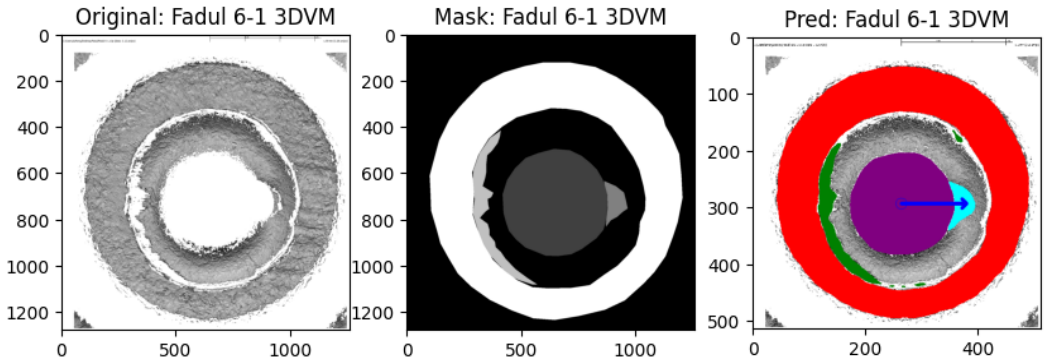
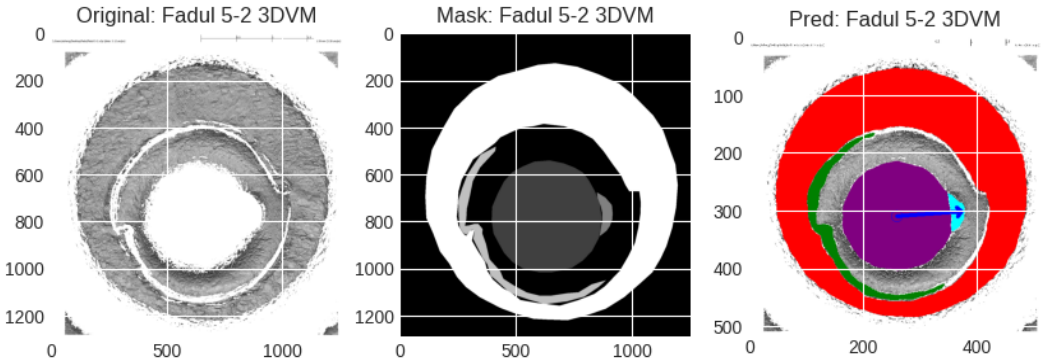


Figure 6.4: Comparison of Test Image Segmentation Results for **Fadul 5-2 3DVM** (breech-face impression in red, aperture shear in green, firing pin impression in purple, firing pin drag in light blue, direction of the firing pin drag indicated by blue arrow)

(a) Test Image Segmentation (Early Stopping): Fadul 5-2 3DVM



(b) Test Image Segmentation (100 Epochs): Fadul 5-2 3DVM

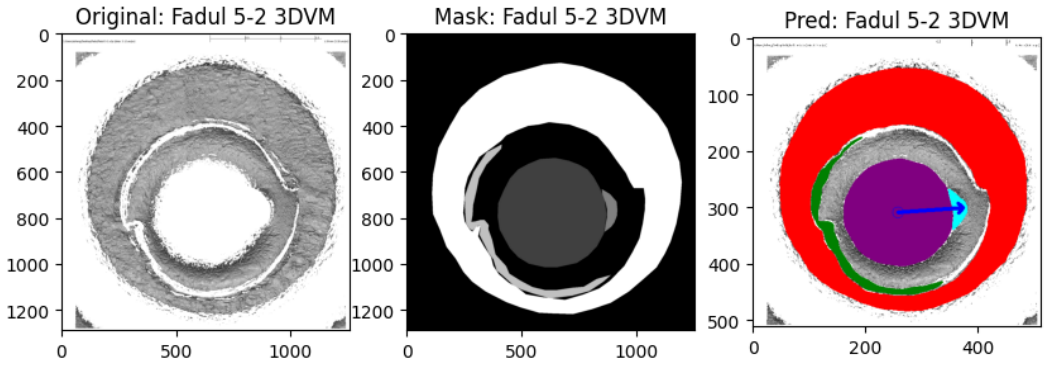
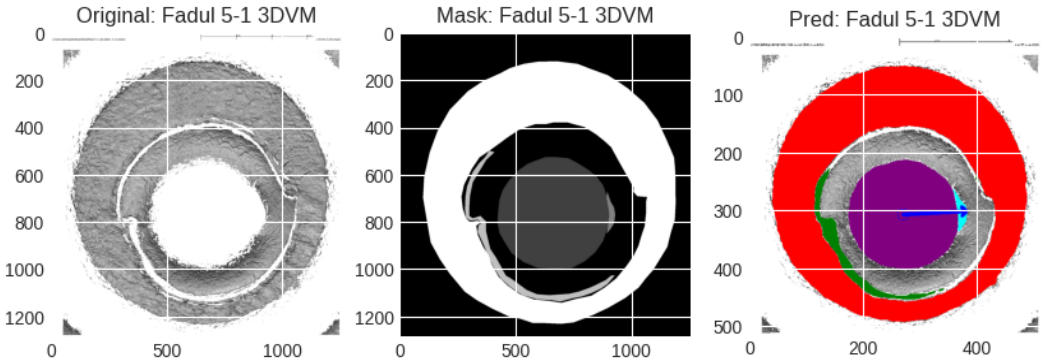


Figure 6.5: Comparison of Test Image Segmentation Results for **Fadul 5-1 3DVM** (breech-face impression in red, aperture shear in green, firing pin impression in purple, firing pin drag in light blue, direction of the firing pin drag indicated by blue arrow)

(a) Test Image Segmentation (Early Stopping): Fadul 5-1 3DVM



(b) Test Image Segmentation (100 Epochs): Fadul 5-1 3DVM

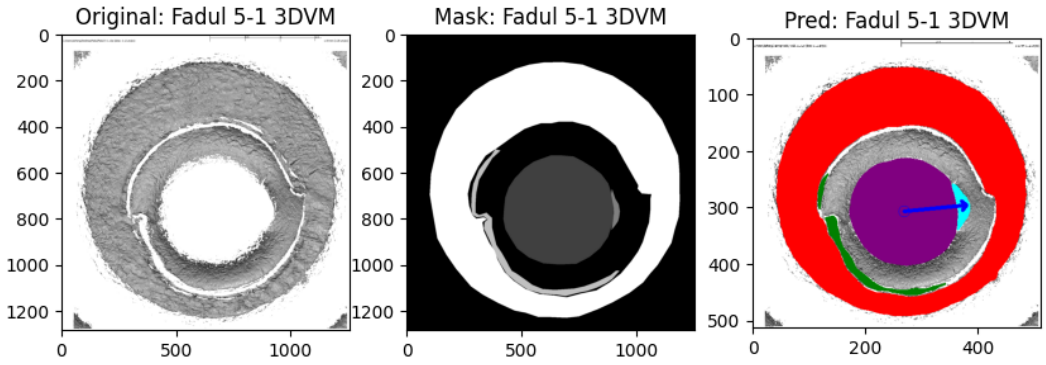
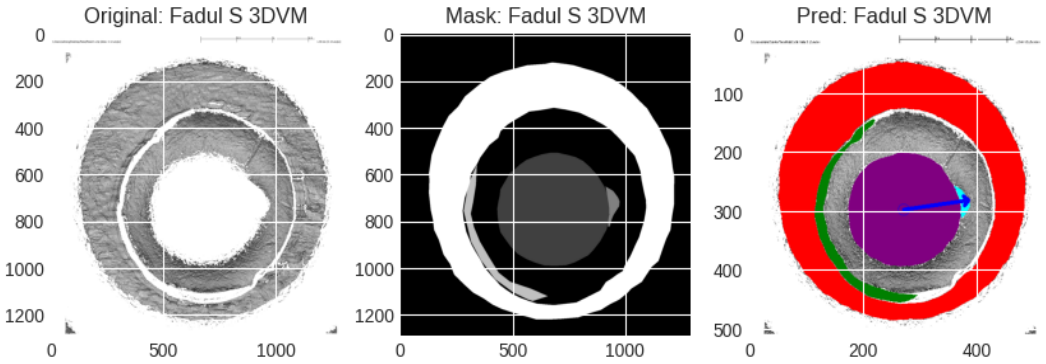
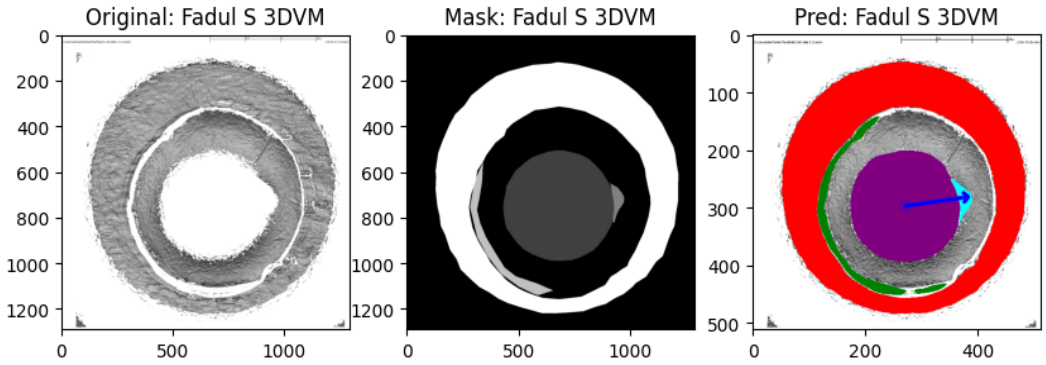


Figure 6.6: Comparison of Test Image Segmentation Results for **Fadul S 3DVM** (breech-face impression in red, aperture shear in green, firing pin impression in purple, firing pin drag in light blue, direction of the firing pin drag indicated by blue arrow)

(a) Test Image Segmentation (Early Stopping): Fadul S 3DVM



(b) Test Image Segmentation (100 Epochs): Fadul S 3DVM



Future Work

To gain deeper insights into the intricacies of the model's performance and foster improvements, a thorough examination of the distribution and characteristics within classes exhibiting lower Dice Scores, such as aperture shear and firing pin drag, is imperative. Identifying challenging or ambiguous instances within these classes is essential to pinpoint specific hurdles that may hinder the model's efficacy.

It is crucial to acknowledge that the current annotations were established as a proof of concept in this project. Seeking approval from domain experts, particularly for challenging classes, is vital to ensure the accuracy and consistency of annotations. Discrepancies or inaccuracies in annotations can have profound repercussions on the model's performance, directly impacting its ability to generalize effectively.

Furthermore, tailoring augmentation techniques to the unique characteristics of each class has the potential to enhance the model's ability to generalize. Exploring adjustments to the U-Net model architecture or hyperparameters offers another avenue for potential improvement. Experimenting with different configurations to identify settings that enhance performance on challenging classes, including exploring learning rate schedules or adaptive learning rate methods to stabilize training and improve convergence, is a valuable endeavor.

Strategically expanding the dataset for classes with lower performance is a proactive approach. Incorporating additional diverse examples can enrich the model's understanding of more robust features. Fine-tuning the model specifically on challenging classes can augment its sensitivity to distinguishing features within those classes. Additionally, investigating the potential benefits of other deep-learning architectures holds promise for enhancing overall segmentation performance.

Conclusion

In conclusion, this project has successfully achieved its primary objectives by developing an algorithm for the automatic masking of cartridge case images, focusing on a multi-class semantic segmentation task in computer vision. The proposed automated method harnesses the power of deep learning techniques, showcasing its capability to segment firearm-specific features within the images. This contribution marks a significant advancement in firearm identification systems.

The firearm-specific features, including breech-face impression, aperture shear, firing pin impression, firing pin drag, and the direction of the firing pin drag, hold paramount importance in forensic investigations. Each feature exhibits firearm model-specific characteristics, making them invaluable tool marks for precise firearm identification.

The automation of the masking process introduced in this work provides an efficient and accurate approach to identifying these critical firearm features within cartridge case images. The algorithm's potential scalability opens avenues for broader applications in processing larger datasets, promising enhanced forensic capabilities in firearm identification through advanced computer vision techniques.

The Fadul dataset, comprising 40 topography images of 9mm caliber cartridge cases from the NIST Ballistics Toolmark Research Database, served as the foundation for model training and evaluation. The LabelMe annotation tool was employed to construct the ground truth dataset. To demonstrate the model's generalizability on this limited dataset, a 10-fold cross-validation approach was adopted. The variability observed in the Dice Scores across folds is approximately $\pm$ 0.003 from the average Dice Score of 0.9689. Additionally, the dataset was split into train and test sets using an 90/10 ratio, with the test set remaining unseen during training for a fair assessment of the model's performance on new, unseen data.

Two distinct training approaches were implemented. The first involved training with an early stopping condition, achieving Early Stopping at Epoch 38 with an average Dice Score of 0.98 and a cross-entropy loss of 0.04 on both the validation and test sets. In the second approach, the model was trained over a predefined number of Epochs (Epoch = 100), achieving the best Dice Score at Epoch 98 with an average Dice Score of 0.99 and a cross-entropy loss of 0.03 on both the validation and test sets. This approach demonstrated superior model performance compared to the early stopping strategy in terms of Dice Score and loss. Finally, segmentation prediction images were presented to visually showcase the model's performance in masking regions of interest.

This proof of concept underscores the promising results of employing a deep-learning segmentation method, particularly in the domain of firearm identification.

References

[1] F. Riva and C. Champod, "Automatic comparison and evaluation of impressions left by a firearm on fired cartridge cases," *Journal of Forensic Sciences*, vol. 59, no. 3, pp. 637–647, 2014. [Online]. Available: https://doi.org/10.1111/1556-4029.12382

[2] G. Gerules, S. K. Bhatia, and D. E. Jackson, "A survey of image processing techniques and statistics for ballistic specimens in forensic science," *Science & Justice*, vol. 53, pp. 236–250, 2013. [Online]. Available: https://doi.org/10.1016/j.scijus.2012.07.002

[3] I. Kara, "Investigation of ballistic evidence through an automatic image analysis and identification system," *Journal of Forensic Sciences*, vol. 61, no. 3, pp. 775–781, 2016. [Online]. Available: https://doi.org/10.1111/1556-4029.13073

[4] X. Zheng, J. Soons, T. V. Vorburger, J. Song, T. Renegar, and R. Thompson, "Applications of surface metrology in firearm identification," *Surface Topography*, vol. 2, no. 1, p. 014012, 2014. [Online]. Available: https://doi.org/10.1088/2051-672X/2/1/014012

[5] K. B. Morris, E. F. Law, R. L. Jefferys, E. C. Dearth, and E. B. Fabyanic, "An evaluation of the discriminating power of an integrated ballistics identification system® heritage™ system with the nist standard cartridge case (standard reference material 2461)," *Forensic Science International*, vol. 280, pp. 188–193, 2017. [Online]. Available: https://doi.org/10.1016/j.forsciint.2017.09.004

[6] P. Pisantanaroj, P. Tanpisuth, P. Sinchavanwat, S. Phasuk, P. Phienphanich, P. Jangtawee *et al.*, "Automated firearm classification from bullet markings using deep learning," *IEEE Access*, vol. 8, pp. 78 236–78 251, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.2989673

[7] M. Kudonu, M. A. AlShamsi, S. Philip, G. Khokhar, P. B. Hari, and N. Singh, "Artificial intelligence: Future of firearm examination," in *Proceedings of the 2022 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2022, pp. 1–5. [Online]. Available: https://doi.org/10.1109/ASET53988.2022.9735105

[8] F. Riva, R. Hermsen, E. Mattijssen, P. Pieper, and C. Champod, "Objective evaluation of subclass characteristics on breech face marks," *Journal of Forensic Sciences*, vol. 62, no. 2, pp. 417–422, 2017. [Online]. Available: https://doi.org/10.1111/1556-4029.13274

[9] R. Sibert, "Drugfire: revolutionizing forensic firearms identification and providing the foundation for a national firearms identification network," *Crime Laboratory Digest*, vol. 21, pp. 63–67, 1994.

[10] National integrated ballistic information network. Tobacco, Firearms and Explosives: ATF Bureau of Alcohol. [Accessed 20 Jun 2022]. [Online]. Available: https://www.atf.gov/firearms/national-integrated-ballistic-information-network-nibin

[11] M.- Le Bouthillier, L. Hrynkiw, A. Beauchamp, L. Duong, and S. Ratté, "Automated detection of regions of interest in cartridge case images using deep learning,"

Journal of Forensic Sciences, vol. 68, pp. 1958–1971, 2023. [Online]. Available: https://doi.org/10.1111/1556-4029.15319

[12] J. Song, W. Chu, M. Tong, and J. Soons, "3d topography measurements on correlation cells — a new approach to forensic ballistics identifications," *Measurement Science and Technology*, vol. 25, p. 064005, 2014. [Online]. Available: https://doi.org/10.1088/0957-0233/25/6/064005

[13] C. Gambino, P. McLaughlin, L. Kuo, F. Kammerman, P. Shenkin, P. Diaczuk, and et al., "Forensic surface metrology: Tool mark evidence," *Scanning*, vol. 33, p. 272–278, 2011. [Online]. Available: https://doi.org/10.1002/sca.20251

[14] F. Riva, E. Mattijssen, R. Hermsen, P. Pieper, W. Kerkhoff, and C. Champod, "Comparison and interpretation of impressed marks left by a firearm on cartridge cases – towards an operational implementation of a likelihood ratio based technique," *Forensic Science International*, vol. 313, p. 110363, 2020. [Online]. Available: https://doi.org/10.1016/j.forsciint.2020.110363

[15] R. Fischer and C. Vielhauer, "Towards automated firearm identification based on high-resolution 3d data: Rotation-invariant features for multiple line-profile measurement of firing pin shapes," in *Proceedings SPIE 9393 – IS&T/SPIE Electronic Imaging*, R. Sitnik and W. Puech, Eds., vol. 9393. Society of Photo-Optical Instrumentation Engineers, 2015, p. 93930Q. [Online]. Available: https://doi.org/10.1117/12.207756720

[16] X. Tai and W. Eddy, "A fully automatic method for comparing cartridge case images," *Journal of Forensic Sciences*, vol. 63, no. 2, p. 440–448, 2018. [Online]. Available: https://doi.org/10.1111/1556-4029.13577

[17] J. Song, W. Chu, T. Vorburger, R. Thompson, T. Renegar, A. Zheng, J. Yen, R. Silver, and M. Ols, "Development of ballistics identification—from image comparison to topography measurement in surface metrology," *Measurement Science and Technology*, vol. 23, no. 5, p. 054010, 2012.

[18] O. Ronneberger, P. Fischer, and T. 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.

[19] P. Yakubovskiy, "Segmentation models pytorch," https://github.com/qubvel/segmentation_models.pytorch, 2022.