A DEEP LEARNING-ENABLED AUTOMATIC SEGMENTATION SYSTEM FOR SURGICAL ENDOSCOPY

By

Zachary Andrew Stoebner

Thesis
Submitted to the Faculty of the
Graduate School of Vanderbilt University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE
in

Computer Science

August 12, 2022

Nashville, Tennessee

Approved:

Ipek Oguz, PhD

Yuankai Huo, PhD

Nicholas Kavoussi, MD

ACKNOWLEDGMENTS

Thank you to my advisor, Prof. Ipek Oguz, for being the best research mentor that I could ask for and guiding me through every step of this project.

Thank you to Dr. Nicholas Kavoussi, for all his invaluable help, mentorship, and close collaboration on the project.

Thank you to David Lu, for being an incredible friend and collaborating with me on the early stages of the project.

Thank you to Prof. Yuankai Huo for being the second reader of this thesis.

Thank you to my friends and family, for supporting me in my endeavors.

Thank you to the patients and urologists at Vanderbilt University Medical Center who consented to their data being used in this project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

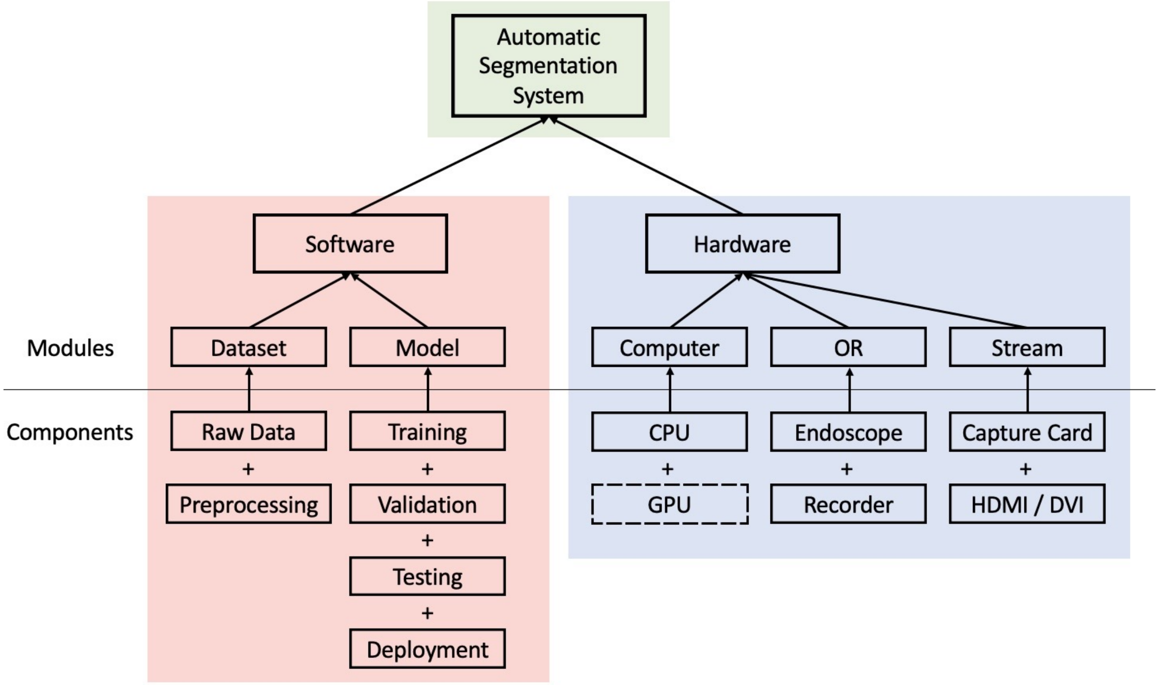


Figure 1.1: Modules and components that comprise the automatic segmentation system. The dashed outline indicates optional components, i.e., a GPU is useful but does not prohibit the system from executing if it is absent.

The advent of deep learning has bestowed many boons to many disciplines, namely computer vision and medical imaging. Deep networks can learn robust functions over many domains, yielding impressive performance in complex applications. For example, deep learning has revolutionized image recognition tasks, automating, scaling, and expediting classification and segmentation of images. Both computer vision and medical imaging have greatly benefited from this revolution; convolutional networks are particularly useful in these areas to detect localized features and patterns in the image [1]. One consideration in both application areas is the representation and dimensionality of the input data. Standard 2D images can generally be learned much more easily than modalities in higher dimensions, such as volumetric images or videos. For the latter, additional processing, such as 2D slicing, and architectural expansions, such as attention mechanisms, are employed to fully learn the desired function and accomplish the visual task to a satisfying degree [2, 3]. Examples of deep learning's impact in traditional computer vision are rife, ranging from facial recognition to super-resolution. In medical imaging, research on the deep learning-based registration and segmentation of

MRI and other imaging modalities has grown tremendously in recent years [4, 5].

In surgical medicine, many specialties have yet to benefit from deep learning technology mainly due to the sparsity of available datasets for training, leading to slow adoption[6]. However, as these technologies develop, many specialties are beginning to investigate applications of deep learning tools to augment clinical practice. Due to the prevalence of minimally invasive surgical techniques in urology, the field is well-positioned for capitalizing on image segmentation to automatically detect kidney stones in endoscopic video feeds [7, 8]. Previously, these techniques have demonstrated accurate prediction of stone composition from CT imaging or direct visual analysis [9, 10]. Likewise, computer vision methods could be leveraged during endoscopic stone surgery to automatically track stones and stone fragments during treatment, and image segmentation can be used as a surgical aid and the ability to process real-time feeds can motivate future growth towards an automated image-guided surgery system.

In this project, we explore automated annotation of kidney stones from endoscopic video feeds with supervised deep learning-based image segmentation methods. Our method focuses on semantic segmentation (i.e., the pixel-wise detection of class belonging), instead of instance segmentation (i.e., the pixel-wise identification of each class instance). We aim to establish the feasibility of real-time annotation of kidney stone video feeds in this project, with the long-term goal of building an image-guided surgical system for clinicians and eventually robotic agents. To meet these goals, we built a novel dataset from surgical endoscopic video feeds and investigated models and techniques to generate accurate segmentations. We explored three baseline models for this purpose: U-Net [11], U-Net++ [12], and DenseNet [13, 14]. The system integrates both hardware (endoscope, capture card, computer, cables) and software (dataset, network, training & testing, deployment) modules. Figure 1.1 summarizes the these modules and components.

This thesis is organized as follows:

- Chapter 2 outlines the problem and motivates this work.
- Chapter 3 discusses related work and its applications to this research.
- Chapter 4 details the framework of the automatic segmentation system and how it was built.
- Chapter 5 presents the model selection process for the system's visual component.
- Chapter 6 recounts the deployment of the model in practice.
- Chapter 7 draws conclusions and proposes future directions for this work.

Chapters 4 & 5 are derived from our publication on a deep learning-based segmentation model for automatic kidney stone annotation at SPIE Medical Imaging: Image Processing 2022 [15]. The comparative analysis in chapter 6 is based on a clinical publication submitted to the Journal of Endourology [16].

CHAPTER 2

Problem Statement

2.1 Use Cases

The intended application of this automatic segmentation system is endoscopic surgery to visually assist surgeons in identifying their targets. During these minimally invasive procedures, physicians maneuver an endoscope through cavities to treat a target identified through prior visual diagnostics, which are guided entirely by video reference streamed from the camera at the tip of the endoscope. However, video quality is not high-definition since streams max out at 20 FPS; fiberoptic scopes are the cheaper option with lower resolution and digital scopes are becoming the dominant modality but are more expensive to use [17]. Figure 2.1 shows an example of a digital and a fiberoptic frame to demonstrate the visibility differences.

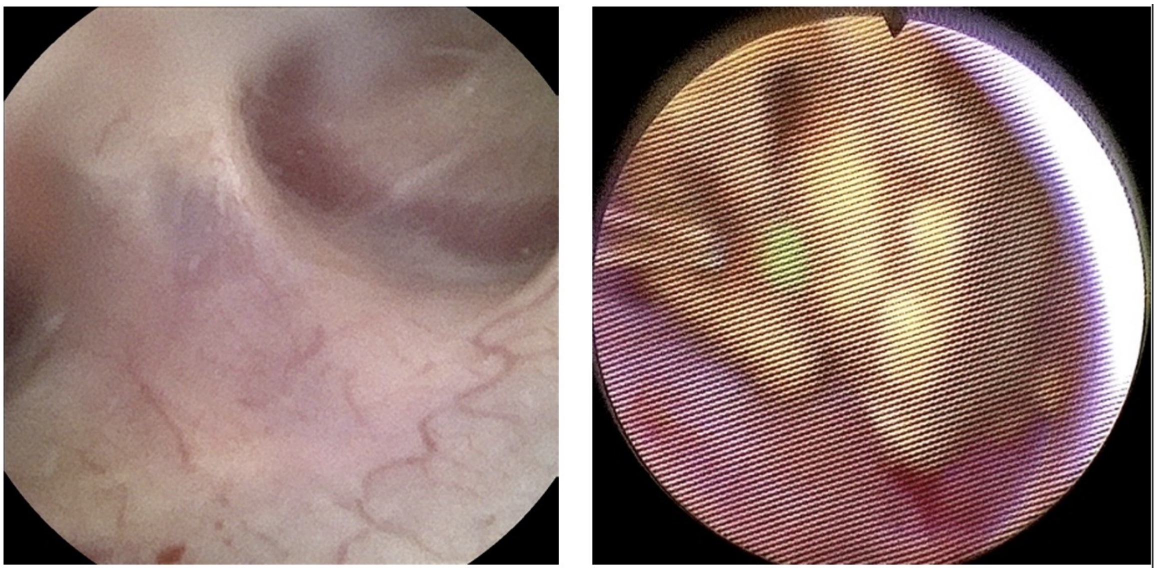


Figure 2.1: Example of a digital (left) and fiberoptic (right) frame. Visibility with a digital scope is much clearer than that with a fiberoptic scope. The digital frame displays blur at the top due to saline treatment.

Additionally, targets are visually obscured by other factors, i.e., saline and tissue, and typically fragment or pulverize during treatment. These obstacles have historically challenged physicians since target identification and tracking rely on visual acuity and memory. Hence, the system aims to provide an assistive interface to augment physicians' vision during the procedure to better identify targets through the endoscopic video feed.

One surgical endoscopic procedure in particular that would benefit from such a system is ureteroscopic kidney stone removal. Serving as the case study in this thesis, the vision models are trained for kidney stone segmentation in digital feeds, which are typically encountered in ureteroscopic scenarios. Using a high-performing segmentation model trained in a supervised fashion, the system takes in a video stream from the endoscope's recording device, used by physicians to save review their procedures after the fact, and outputs a three panel view of the raw feed, the model's segmentation of the frame, and the model's raw output probabilities represented as a heatmap.

2.2 Specification & Usage

An automatic segmentation that will eventually see use during real operating procedures should:

- Segment targets at a high fidelity and pace.
- Display the raw footage next to annotated footage with the predicted segmentation.
- Accept incoming video feed from standard (DVI/HDMI) display output ports.
- Support mobile deployment for any operating room (OR) without unnecessary hardware specialization.

Under these specifications, the system can provide utility to surgeons and be deployed in any OR that outputs endoscopic video feeds accessible at a DVI or HDMI port.

Given its modularity, the system is highly portable between different video segmentation contexts. The dataset, the trained model, and the endoscope are the only components specific to endoscopy; combining a different dataset, model and camera could extend this system to other application areas. The system simplifies to a series of four steps that extend to any deployment scenario:

1. Build and preprocess a dataset.
2. Perform a hyperparameter search on different architectures to identify high performers.
3. Train, validate, and test a high-performing segmentation model.
4. Deploy the system on a video feed using the trained high performer.

CHAPTER 3

Related Work

3.1 Segmentation

Image segmentation, i.e., the identification of the location and boundary of objects within an image, has historically been performed manually by experts or automatically with conventional techniques, such as masking, erosion, & dilation, histogram & Otsu thresholding, multi-label atlasing, and deformable models [18, 19, 20]. Compared to grayscale images, color images convey much more information and often have unclear homogeneous regions that confound conventional automatic segmentation methods, posing a challenge for modern applications to adopt conventional segmentation [21]. In recent years, automatic segmentation has proven useful in myriad applications, from medical image analysis to robotic perception to self-driving cars. With increased applicability, extensive research is underway to develop more efficient, higher fidelity image segmentation techniques, particularly those that use deep learning [22].

3.2 Deep Learning

In computer vision, deep learning has proven extremely capable at learning abstract functions over images. Simple feed-forward networks can classify images treated as a flattened vector, yet convolutional networks have become the standard for deep learning-enabled 2D image processing due to improved performance from including neighborhood information around each pixel [23]. With this increased complexity, large convolutional networks are subject to the vanishing gradient problem so residual connections are introduced to stabilize and improve backpropagation throughout large models [11, 24]. Additionally, recent developments in self-attention and transformers have been translated from natural language processing to computer vision en masse and seen promising results in image and video segmentation [25, 26].

3.3 Generative Learning for Image and Video Segmentation

From a deep learning perspective, segmentation is an abstraction of classification in which each pixel is classified. However, feed-forward classification typically does not preserve the input dimension, rather reducing the output dimension to the number of classes to which any one sample can belong. To generalize traditional deep learning-based classification to the segmentation task, generative learning has been used to automatically produce high-fidelity synthetic annotations of images. In particular, the architecture that has proved especially useful to the task of image segmentation is the encoder-decoder network [27]. These networks learn to encode the image into a low-dimensional latent vector and then decode the latent vector to recreate the image with different attributes or, in the case of segmentation, recreate a mask of the image encoded with the class of each pixel. Example architectures that changed the landscape of image segmentation include SegNet [28] for scene understanding, U-Net [11] for biomedical image segmentation, and DeepLabv3 [29] for multi-scale segmentation.

SegNet contributed max unpooling layers to mirror VGG-16's encoder-like architecture [30] to generate segmentation masks. DeepLabv3 pioneered atrous convolution to capture multi-scale context in images. Of particular interest, U-Net demonstrated surprisingly robust segmentation of biomedical images and many recent medical image segmentation models are variants of this architecture. U-Net has a typical deep encoder-decoder structure but complements a simple feed-forward network with skip connections from each layer of the encoder to each corresponding layer of the decoder, mitigating the vanishing gradient problem. Variants of U-Net, such as U-Net++ [12], propose improvements to extend the characteristic U-Net architecture to compose deeper, yet more robust models.

3.4 Automation in Surgery

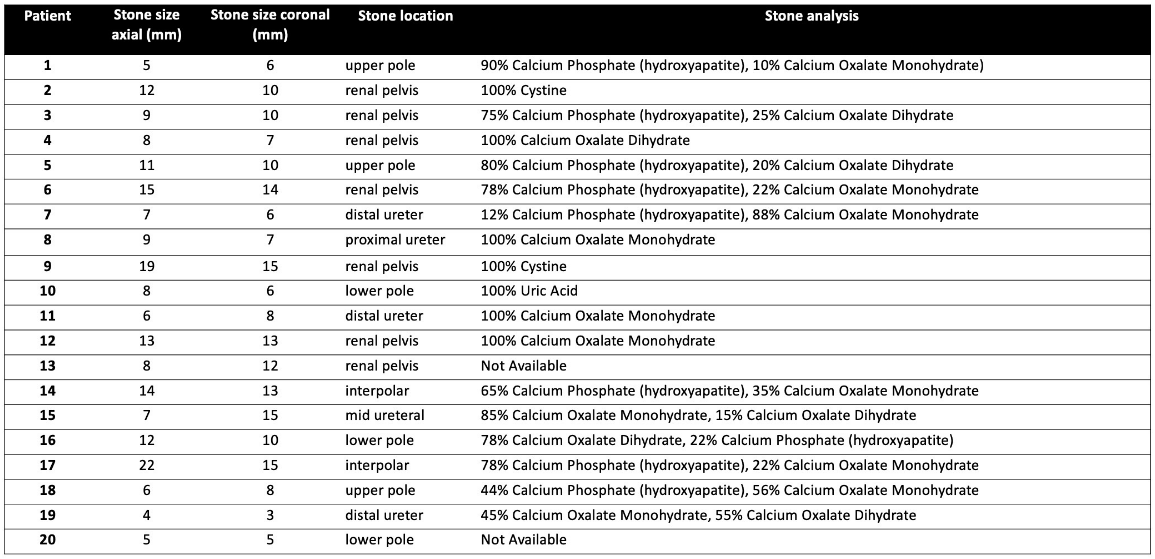
Automation in surgery with modern computing has yet to see widespread adoption but is nonetheless a popular area of research in engineering for surgery. Robotic surgical automation is mainly investigated in minimally invasive surgeries that benefit from image-guided navigation, where surgical instrumentation can more easily be localized relative to anatomy while also reducing intraoperative risk to the patient in the case of malfunction [31]. For example, automatic segmentation of preoperative CT imaging both assists in diagnosis as well as targeting for the robotic surgeon [9, 10]. Often, these systems do not fully replace surgeons but rather augment the visual information available to the surgeon in real time, serving as early testing stages inching towards fully autonomous visually-guided robotic surgery [32].

CHAPTER 4

System Framework

4.1 Dataset & Preprocessing

After approval from the Institutional Review Board, Dr. Kavoussi and colleagues at Vanderbilt University Medical Center (VUMC) obtained video files from 20 patients who had undergone ureteroscopy and holmium laser lithotripsy for kidney stone disease by two surgeons from January to June 2021. All patients were over eighteen years of age and had radiographic evidence of renal stones pre-operatively. Pressure bag irrigation was used for all cases with or without a ureteral access sheath based on surgeon discretion. A digital ureteroscope (Karl Storz Flex Xc) was used for each case. Preoperative imaging characteristics of all stones, as well as their postoperative compositions, were recorded. Table 4.1 details examples of preoperative imaging characteristics and postoperative compositions for 20 patients' stones in the dataset; patients may have multiple stones that appear in the videos and these are only a subset of the full dataset.



Patient	Stone size axial (mm)	Stone size coronal (mm)	Stone location	Stone analysis
1	5	6	upper pole	90% Calcium Phosphate (hydroxyapatite), 10% Calcium Oxalate Monohydrate)
2	12	10	renal pelvis	100% Cystine
3	9	10	renal pelvis	75% Calcium Phosphate (hydroxyapatite), 25% Calcium Oxalate Dihydrate
4	8	7	renal pelvis	100% Calcium Oxalate Dihydrate
5	11	10	upper pole	80% Calcium Phosphate (hydroxyapatite), 20% Calcium Oxalate Dihydrate
6	15	14	renal pelvis	78% Calcium Phosphate (hydroxyapatite), 22% Calcium Oxalate Monohydrate
7	7	6	distal ureter	12% Calcium Phosphate (hydroxyapatite), 88% Calcium Oxalate Monohydrate
8	9	7	proximal ureter	100% Calcium Oxalate Monohydrate
9	19	15	renal pelvis	100% Cystine
10	8	6	lower pole	100% Uric Acid
11	6	8	distal ureter	100% Calcium Oxalate Monohydrate
12	13	13	renal pelvis	100% Calcium Oxalate Monohydrate
13	8	12	renal pelvis	Not Available
14	14	13	interpolar	65% Calcium Phosphate (hydroxyapatite), 35% Calcium Oxalate Monohydrate
15	7	15	mid ureteral	85% Calcium Oxalate Monohydrate, 15% Calcium Oxalate Dihydrate
16	12	10	lower pole	78% Calcium Oxalate Dihydrate, 22% Calcium Phosphate (hydroxyapatite)
17	22	15	interpolar	78% Calcium Phosphate (hydroxyapatite), 22% Calcium Oxalate Monohydrate
18	6	8	upper pole	44% Calcium Phosphate (hydroxyapatite), 56% Calcium Oxalate Monohydrate
19	4	3	distal ureter	45% Calcium Oxalate Monohydrate, 55% Calcium Oxalate Dihydrate
20	5	5	lower pole	Not Available

Table 4.1: Preoperative imaging characteristics and postoperative compositions for 20 patients' stones. A variety of stone types were present in the dataset with calcium oxalate stones as the predominant type. Patients may have multiple stones that appear in the video and these are only a subset of the full dataset.

We visually assured the quality of videos, controlling for factors that obstructed frames for the entire video that were not related to treatment (i.e. text overlay). After quality assuring the raw inputs by visual inspection, we extracted 20 frames per second [FPS] from the videos. We cropped each image by first converting to grayscale and using Otsu thresholding [33] to separate the background from the foreground, then finding the contours using OpenCV and selecting the closed contour with the greatest area to identify a bounding box for the image. Compared to manual cropping and non-parallelized cropping algorithms, this method is extremely efficient, capable of processing videos at their playback speed. Out of a total of 29 videos, two were removed due to text overlaid on the scope's video after capture by the surgeon, which thwarted clean cropping of these videos.

After cropping, the frames were manually annotated for kidney stones to generate the ground truths for the examples in the dataset, both for training and evaluation purposes. To annotate the cropped images, we employed MakeSense.ai [34], a web application that supports polyline annotations and saves the data in JSON format. To pair example images with their annotated ground truths in the dataset, we used OpenCV [35] to convert the polygons into binary images. These pairs of images were used for supervised training of the models. Figure 4.1 displays an example of a cropped image and its manual annotation.

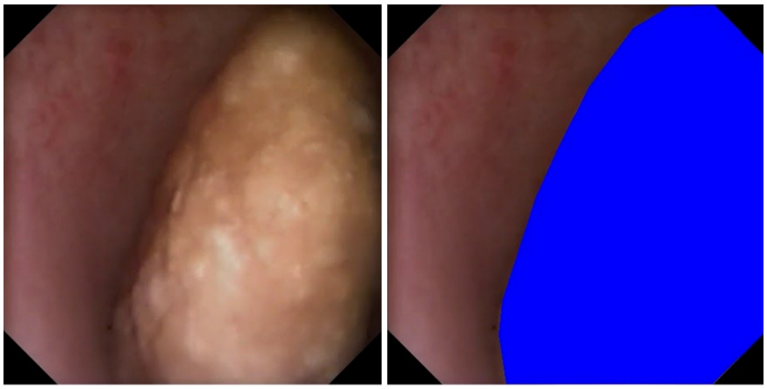


Figure 4.1: Example of a cropped image and its manual annotation. The images were annotated using MakeSense.ai and the vertices of the polygon drawings were saved in JSON files. After parsing the JSON files, the polygons were then converted into binary images and displayed as an overlay on original images using OpenCV to visualize the annotation.

4.2 Architectures

We investigated three architectures, U-Net [24], U-Net++ [12], and DenseNet [13], to establish baselines of current state-of-the-art image segmentation models for the kidney stone segmentation task. The U-Net baseline was set to a depth of 5 and composed of ResNet34 blocks . Similar architecture parameters were applied to our U-Net++ model; however, it is important to note that U-Net++ contains numerous subnetworks that complicate the model but also make it more robust. Hence, U-Net++ has significantly more parameters than a typical U-Net. The third type of model that we looked at was a DenseNet, which implements the U-Net architecture but with DenseBlocks [14] instead of ResNet blocks. The variant that we focused on was DenseNet67. Each variant had a depth of 5 with different blocks sizes, growth rates, and bottleneck sizes. Figure **??** summarizes the baseline architectures used.

In addition to the baseline architectures, we experimented with improvements known to augment the performance of image and video segmentation, namely neural matting and attention. Ideally, these improvements reduce spurious segmentation predictions away from the main body of the object of interest.

4.2.1 Neural Matte

We attempted to improve upon the high performer's architecture with neural matting [36]. A neural matte is a deep architecture – ranging from an additional autoencoder to a few DenseNet layers [37] – applied after a base segmentation model. In essence, the neural matte is an adaptive regularizer that learns foreground and background for the mask prediction. Starting small, we experimented with a neural matte composed of a single DenseBlock, trying a few values for the number of layers and growth rate. The aim of the neural matte was to improve the model's performance by smoothing the output.

4.2.2 Attention

For high-dimensional data such as images, the complexity in the input domain often hinders model performance, especially when convolutional operations only preserve local information at each feature. For structured inputs represented in high dimensions, distant parts of the image may correlate with each other; exploiting these correlations is crucial to improving predictive outcomes for these domains. Modeling human attention, i.e., learning what parts of the input to consider more in a prediction, simplifies the problem because the model does not need to pay equal attention to all features of the input in all channels, just a smaller subset [25]. To learn these long-range dependencies, we investigated U-Net and U-Net++ with spatial and channel 'squeeze & excitation' blocks [26]. Another application of attention involves learning temporal dependencies between subsequent frames, which is left to future work.

4.3 Hardware Integration

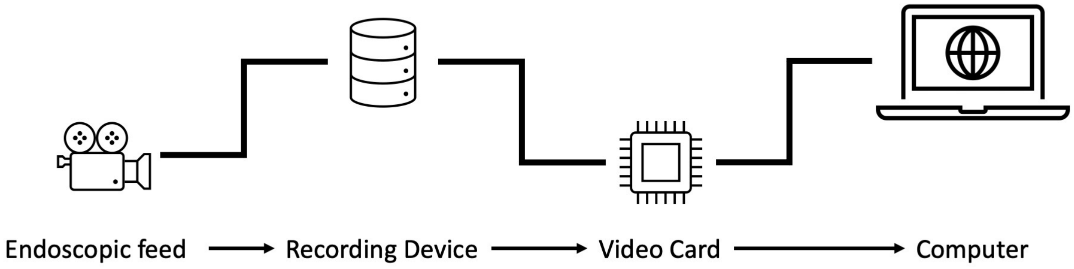


Figure 4.2: Hardware setup for the live system and the flow of information from the endoscopic feed to the computer running the predictive model.

To deploy the high performing model in the field, we devised a straightforward hardware setup that promotes the portability of the model. A computer and video capture card can be set up in any OR with a recording system. The HDMI / DVI output from the recording system are fed to the capture card which is then accessed as a video stream with extractable frames that are passed to the model. Figure 4.2 shows the hardware setup and flow of information between the hardware.

CHAPTER 5

Model Selection

5.1 Setup

We heavily employed PyTorch [1] and Comet.ml [2] to carry out our model training and analysis. PyTorch comprised the deep learning framework for our models and Comet.ml was used to extract relevant metrics and images throughout the training and testing. Comet.ml then logged all of the information to a web browser workspace, aggregating all of our experiments and collected data and plotting the live curves of the model during training. Data points were collected at each step in the training process where $total\ steps = \lceil \frac{dataset\ length}{batch\ size} \rceil * epochs$. All training and testing was performed on the ACCRE computing cluster [3].

5.1.1 Training, Validation, and Testing

Training was a standard iterative pipeline with forward propagation and loss computation, followed by backpropagation with frequent validation checks. Initially, the training / validation and testing sets were split randomly in an approximately 80 /20 split, respectively; at the beginning of each training, the validation set was determined randomly where the percentage to split is a hyperparameter (10-20%, i.e., 2-4 videos from the 22 left for training). Out of 27 videos, 22 videos were thus utilized in training and 676 frames from these videos were annotated. The remaining five videos comprise the test set with a total of 73 annotated frames; the hold-out test set contains many examples of common challenges in the data, i.e., motion blur, debris fragmentation, foreign objects, and saline injection. The model does not update its parameters on the validation data as no loss is computed for the validation set. Rather, it is used in a similar fashion as the test set to generate scores and segmentation masks throughout training to better monitor and understand the model's performance. Figure 5.1 outlines the model workflow.

We also compared the results of vanilla and pretrained U-Net and U-Net++ models to confirm whether pretraining is helpful for this task. Semi-supervised pretraining was conducted on ImageNet [38] to obtain weights for the ResNet components that comprise U-Net and U-Net++ [39]. To confirm consistency in performance, we have trained multiple copies of the best hyperparameter configuration found for each architecture. In total, we have saved 8 high-performing models for U-Net++, 6 for U-Net, and 3 for DenseNet. On top of that, we have archived the many more models acquired from the grid search on each architecture.

[1] https://pytorch.org

[2] https://www.comet.ml

[3] https://www.vanderbilt.edu/accre/

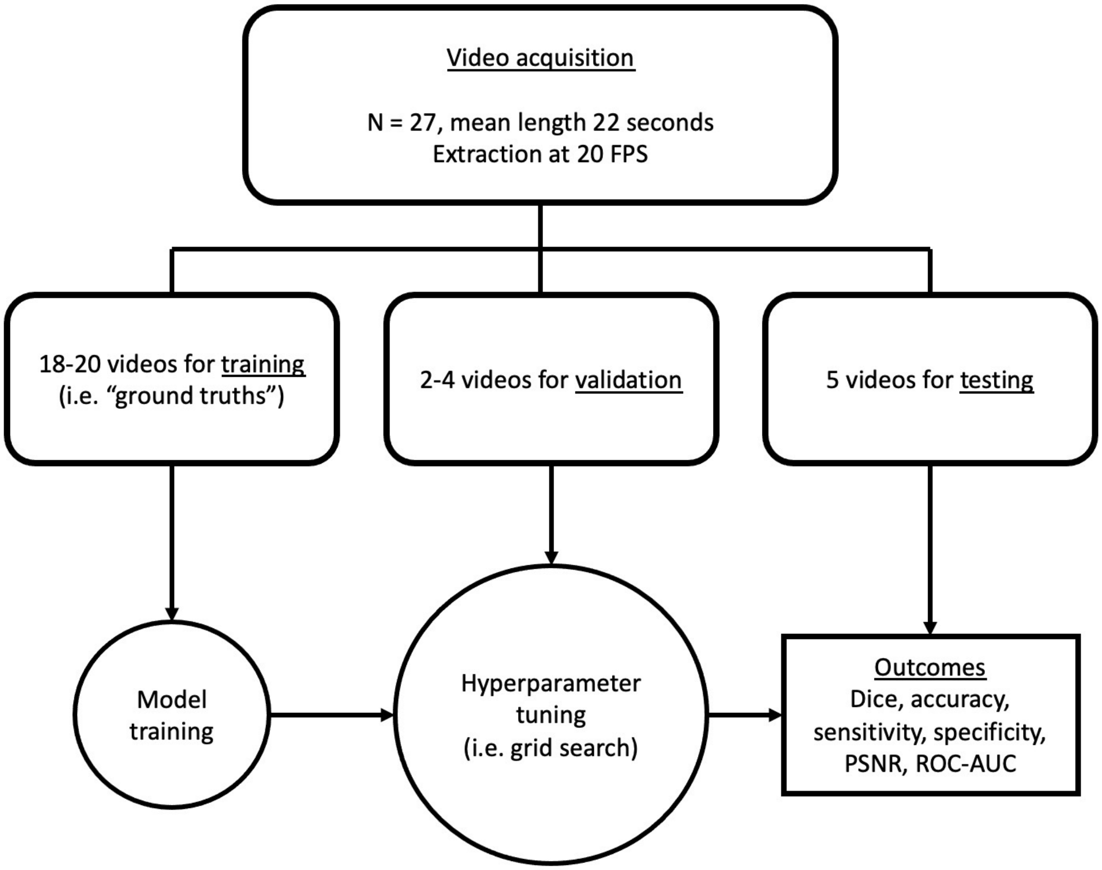


Figure 5.1: The model workflow. At the completion of this pipeline, the models were compared by their outcomes to select the highest performer.

5.2 Evaluation

To robustly identify the best models and their best hyperparameters, we conducted a standard grid search for each model and plotted their training and loss curves. To compare the accuracies of the different architectures throughout training, testing, and searching, we computed the Sorensen-Dice coefficients [40]. At the image level, the Dice score is:

$$Dice(P,G) = \frac{2|P \cap G|}{|P| + |G|}, \text{ where P is the predicted set of pixels and G is the target set of pixels.}$$

We chose the Dice score as our primary measure of performance because it is well-defined for balanced, binary-classed datasets and many deep learning methods for semantic segmentation also measure the Dice score. Other statistics commonly applied to image segmentation include the pixelwise accuracy, intersection over the union (IoU), peak signal-to-noise ratio (PSNR), receiver operating characteristic (ROC), and area under the curve (AUC). These metrics are supplementary and also yield the rates for true positives, true negatives, false positives, and false negatives. For our loss function, we used binary cross-entropy (BCE) [41].

5.3 Results

5.3.1 Pretraining

As expected, we confirmed that pretraining our U-Net models on ImageNet sped training up by approximately 80%. In essence, the pretrained models have a head start and yield higher accuracies early in training and converge to their maximal accuracies shortly thereafter. The training curve comparison between a vanilla U-Net and ImageNet-pretrained U-Net and U-Net++ are displayed in Figure 5.2. Our DenseNet models were not pretrained; regardless, DenseNet executed much more slowly on a GPU than the other two architectures, often taking hours to train while the U-Net and U-Net++ took minutes to train.

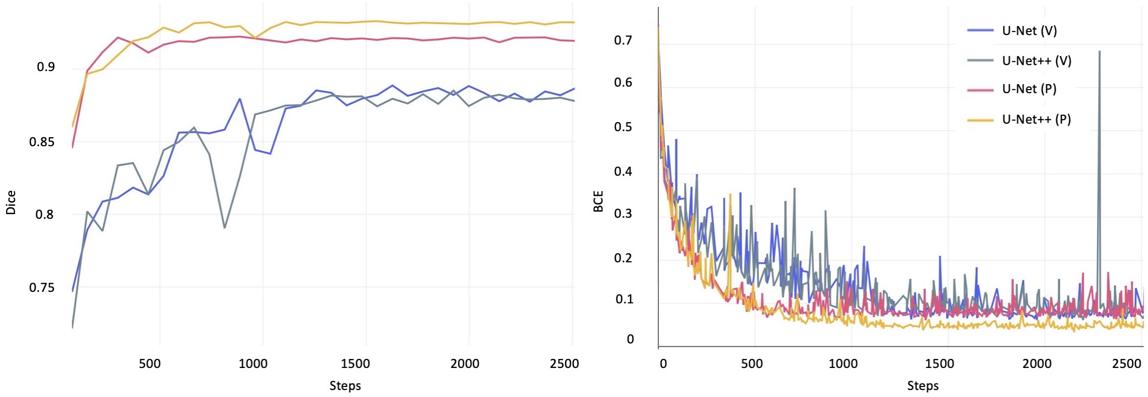


Figure 5.2: A comparison of vanilla (V) and pretrained (P) model training accuracies. Vanilla U-Net and U-Net++ start at a lower accuracy of approximately 0.75 Dice and gradually converge to a Dice score of approximately 0.88. ImageNet-pretrained U-Net and U-Net++ start at approximately 0.85 Dice, the convergence of the vanilla versions, and converge to about 0.91 and 0.92 Dice, respectively.

5.3.2 Training the Baseline Models

Quantitative results for our three baseline models are shown in Figure 5.3, with the set of hyperparameters that led to the best accuracy for each architecture. We observed that the ImageNet-pretrained U-Net++, with a batch size of 8 and learning rate of 5e-5, performed the best on average with the highest Dice scores reaching above 0.91, followed by the ImageNet-pretrained U-Net, with a batch size of 4 and a learning rate of 1e-4, ranging in Dice score between 0.8-0.9, and finally the ImageNet-pretrained DenseNet67, with a batch size of 1 and learning rate of 1e-4, which capped out at a Dice score of approximately 0.7. Performance often converged within a ± 0.1 Dice of the mean score of a certain architecture. Additionally, we observed that our models were highly sensitive to the learning rate and normalization. Low learning rates and no normalization achieved the best performance. Table 5.1 summarizes the maximum validation statistics, in parentheses, for each baseline model.

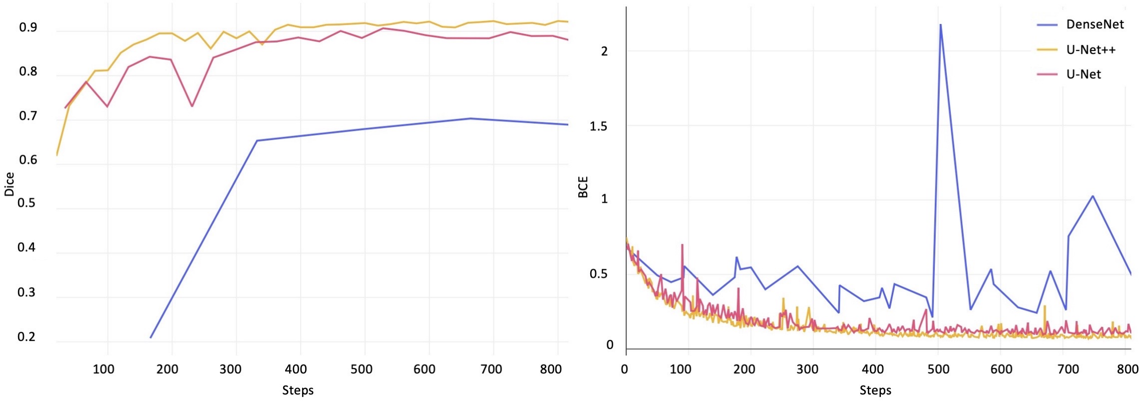


Figure 5.3: A comparison of our best models' Dice score (left) and BCE loss (right) from training. Note that the y-axis is scaled to the range of values. DenseNet had the highest and most variant BCE loss, yet its outlier values are relatively low compared to those in other binary classification tasks as BCE does not have an upper bound. U-Net and U-Net++ are pretrained while DenseNet is trained from scratch.

With these findings, we determined that the ImageNet-pretrained U-Net++ is the best model for the kidney stone segmentation task, so further experimentation will continue in this direction.

5.3.3 Real-time Video Feed and Qualitative Results

We developed a script for processing video feeds and making predictions using our trained models. The script provided sequential frames to the model and ran slightly faster than the duration of each video being processed, which is better than 30 FPS. Videos were separated and reconstructed using OpenCV's video processing pipeline. Together with the model input and the predicted annotation, we reconstructed the video, containing a model input, its predicted annotation, and the corresponding probability map for side-by-side comparison, from the frames sequentially passed through the model. For testing, ground truth images were also available and included in the comparison image. Figure 5.4 shows an example of a frame used for side-by-side comparison with its ground truth, prediction, and probability shown as a heat map overlay. The GPU hardware used was an NVIDIA RTX 2080 Ti.

5.3.4 Hold-Out Performance and Generalizability

To evaluate the performance of our model on challenging unseen data, we held out a test set comprised of data with many challenging examples that are common in realistic scenarios, including motion blur, debris fragmentation, foreign objects, and saline injection. Table 5.1 summarizes the test performances of each baseline model, with the highest validation performances in parentheses.

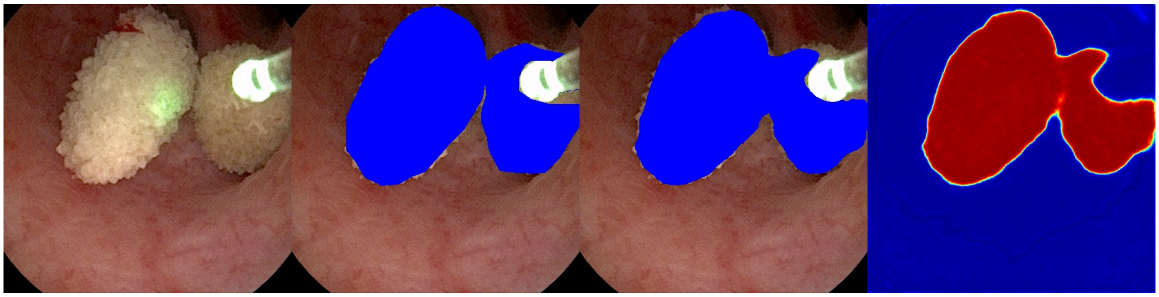


Figure 5.4: Sample frame from the side-by-side video reconstruction of the input, ground truth, automated prediction with U-Net++, and heat map (left to right). The heat map is the raw probability output per pixel whereas the predicted segmentation is the pixels with probabilities ≥ 0.5. The model was able to compute this output at 30 FPS.

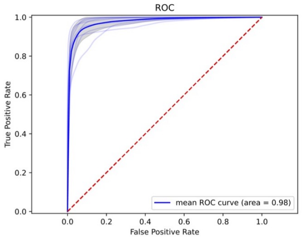
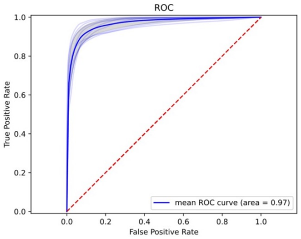
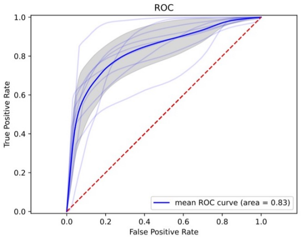
	U-Net++	U-Net	DenseNet
Dice	**0.83** (0.92)	**0.83** (0.91)	0.59 (0.73)
PixAcc	**0.92** (0.95)	0.91 (0.96)	0.75 (0.87)
IoU	**0.81** (0.86)	0.76 (0.86)	0.51 (0.64)
PSNR	**61** (62)	60 (62)	55 (58)
AUC	**0.98** (0.99)	0.97 (0.99)	0.83 (0.91)
ROC			

Table 5.1: Summary of the statistics gathered for each baseline model. Non-parenthetical values are the average scores from all frames in the test set. The reported value in parentheses is the maximum value recorded from each baseline's validation during training. The highest performances between models for each metric are denoted in bold. U-Net++ claimed the highest scores in each metric on the test set. U-Net had the same test Dice score as U-Net++ but lower scores in all other metrics. DenseNet had relatively poor performance for all metrics. Only the ROC curves from test set performances are included.

Additionally, we have unlabeled videos in our dataset as mock up examples of realistic videos that a deployed model would see. Shown in Figure 5.5 for a representative unlabeled video in our dataset, we created segmentation predictions for these videos using our video processing script described above as a proof-of-concept for the visual overlay that we intend to deploy in operating rooms where our model will receive real-time and "in-step" video frames from the endoscopic hardware.

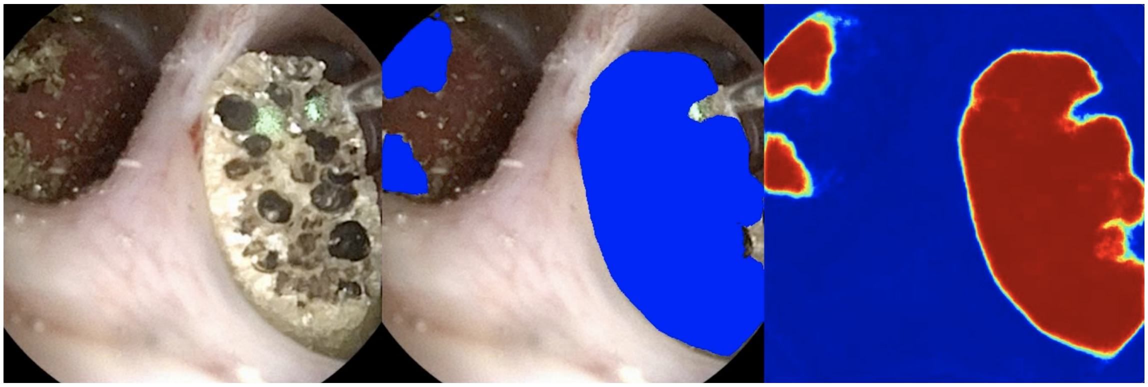


Figure 5.5: Sample output frame for a representative unlabeled input video from our dataset with our best U-Net++ model. This video has no ground truth annotation at the time of publication. Debris collects in the left duct and is still segmented by the model.

5.3.5 Neural Matting

Beyond prior models, we also experimented with a neural matte concatenated onto our best baseline U-Net++ model. To not over-complicate the model, we applied a small DenseBlock as a neural matte to our model with the aim of improving our best model's predictions. A DenseBlock that was too large or too small tended to garble the signal from the base model, resulting in low Dice scores. However, a DenseBlock with 4 dense layers and a growth rate of 4 maintained the signal and consistently yielded 0.85 Dice during validation. Figure 5.6 shows the results of reruns with neural matting appended to the model.

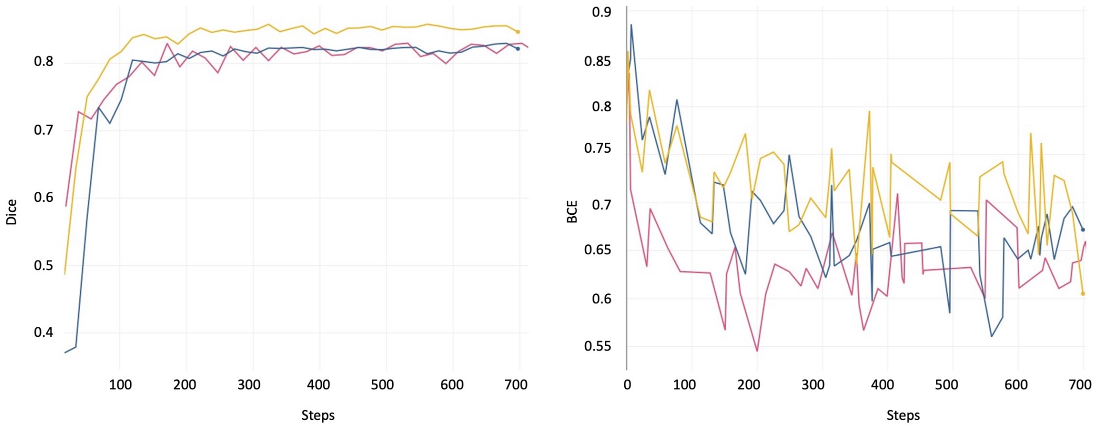


Figure 5.6: Three reruns of training on U-Net++ with neural matting with comparing validation Dice scores (left) and BCE losses (right). Although the performance is decent, the loss is much higher with matting.

5.3.6 Attention

Adding SCSE attention modules in the decoders of the U-Net and U-Net++ models resulted in nearly equivalent performance with their baselines. U-Net++ still slightly outperformed U-Net in generalizability on the test set. Table 5.2 summarizes the performance results of SCSE attention-enhanced U-Net and U-Net++.

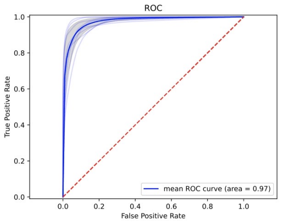
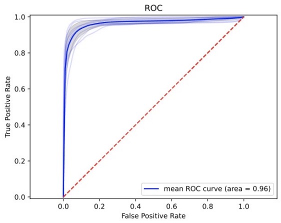
	U-Net++	**U-Net**
Dice	**0.83** (0.93)	0.82 (0.93)
PixAcc	**0.92** (0.96)	**0.92** (0.97)
IoU	**0.80** (0.90)	0.79 (0.92)
PSNR	**61** (64)	60 (64)
AUC	**0.97** (0.99)	0.96 (0.99)
ROC		

Table 5.2: Summary of performances with SCSE attention in the U-Net and U-Net++ decoders. Non-parenthetical values are the average scores from all frames in the test set. The reported value in parentheses is the maximum value recorded from each baseline's validation during training. The highest performances between models for each metric are denoted in bold. Performances were comparable to those without attention modules and U-Net++ still generally outperformed U-Net on the test set. Only the ROC curves from test set performances are included.

5.4 Summary & Interpretation

With relatively few training examples, our U-Net++ and U-Net models achieve an accuracy greater than 90%. At this performance level, our models can be interpreted by surgeons and used to assist them visually to identify kidney stones in the video feed. With a larger dataset and additional improvements to the U-Net and U-Net++ architectures, we expect that the model will perform better on a wider range of scenarios that occur in realistic endoscopic surgeries.

Although empirically good segmentations are predicted by the matted high performer, the score outcomes are comparably worse. Hence, we opted to abandon neural matting to explore alternative improvements. Additionally, exclusion of a neural matte keeps the model simple and lightweight for deployment since Dense-Blocks execute much more slowly than ResBlocks. SCSE attention modules in the decoder did not greatly improve performance for U-Net and U-Net++ and were comparable to the baseline model performances. Although attention is a promising line of work and temporal attention may improve whole video segmentation in future work, these modules may add bloat to the model in a deployment scenario, without arguably better outcomes.

The sensitivity of the models to the learning rate and the slightly variable performances of the same models in random restarts suggest that the cost landscape for this task may require multiple restart attempts for the same model to achieve the best optimization. Given the similarity in performance between U-Net and U-Net++, U-Net may achieve a higher Dice score on some restarts, or with a larger dataset. Although the output videos are real-time, they are not “in-step” with the input video, which is passed through the model first and each frame is added to the OpenCV video object to generate the video reconstruction at the end.

Although the training set still had many examples of motion blur, debris fragmentation, foreign objects, and saline injection, the test set videos had empirically more examples of these kinds. In these challenging situations, novice surgeons could greatly benefit from a tool that visually assists identification of stones. Our high-performing models still score >0.8 Dice on this data and, in video reconstructions of unlabeled video.

With its current performance, our high-performing model could be used to automatically annotate new kidney stone videos to contribute to dataset expansion, alongside rigorous quality assurance. With improved performance, a visual robotic system employing our model could ideally become an end-to-end automatic solution for urological endoscopic surgeries in the long term.

5.4.1 Clinical Relevance

Our innovative approach to training a supervised model for tracking kidney stones and integrating this information in the surgical display during endoscopic stone surgery could enhance a surgeon’s ability to diagnose and treat kidneys stones. Due to the limited field of view, visibility during endoscopic stone surgery can be impacted by blood and debris which decreases stone free rates leading to recurrence events [42]. Our system could potentially mitigate these visibility issues and improve stone treatment by leveraging these computer vision techniques. Similar applications of deep learning algorithms have shown potential in augmenting surgical technique and safety in robotic and laparoscopic surgeries [43]. Furthermore, our system is potentially generalizable since future researchers developing automated tracking and video segmentation systems could use our basic approach for other endoscopic surgeries.

CHAPTER 6

Deployment

6.1 Setup

6.1.1 Comparative Analysis

To evaluate the practicality of our model in real operating scenarios, we performed a comparative analysis of our high-performing U-Net++ model across three videos each of fiberoptic, dusting, and fragmentation and compared the segmentations to those of two endourologists. Three separate videos of stone fragmentation (at 0.8J and 8Hz) and dusting (0.3J and 30Hz) were processed by our high-performing U-Net++ model. Similarly, three separate videos of ureteroscopy using a fiberoptic scope were processed by the same model. Each video was taken and processed at 30 FPS and then the video was reconstructed with the initial input, predicted annotation of stone, and a corresponding heatmap.

6.1.2 Live Deployment

To deploy our high-performing model for surgical use in operating rooms, we extended our video processing script to receive "in-step" frame-by-frame video input from a video capture card connected via DVI/HDMI to the endoscopic hardware. After processing the frame, the side-by-side input, prediction, and heatmap frames, as in Figure 5.5, then displays to a monitor to visually assist the surgeon.

To test our system live, we debugged and practiced deployment during multiple procedures in multiple ORs at VUMC. Using the hardware-integrated system, we stationed a 2018 MacBook Pro at the endoscopic recording device and processed frames during operations on consenting patients. Figure 6.1 portrays photographic evidence of OR deployment and real-time frame processing.

6.2 Evaluation

For the comparative analysis, we computed the average raw pixel accuracy, sensitivity, specificity, and ROC-AUC curves, based on the true positive rates from the pixel accuracy, across the three videos in each subset. Since the same videos were annotated by two endourologists, we also computed Cohen's kappa score to add to the comparison to measure inter-rater reliability both between physicians and between physicians and our model [44].

For the live system proof-of-concept, empirical prediction quality and rate of segmentation were observed to verify model deployability. Since no frames could be manually annotated prior to surgery, we did not compute any statistics since no expert ground truth annotations were made.

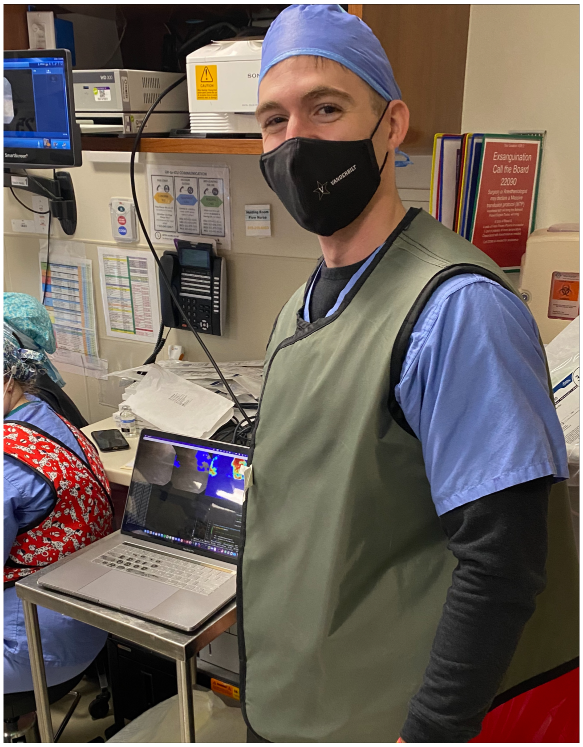


Figure 6.1: Photographic evidence of deployment in the OR where the model is running on a MacBook, reading the feed from the recording device, and processing frames in real time.

6.3 Results

6.3.1 Digital vs. Fiberoptic

Stone segmentation was more accurate with digital scope, compared to fiberoptic scope with a higher pixel accuracy (0.92 vs. 0.87), sensitivity (0.94 vs. 0.64), and ROC-AUC (0.98 vs. 0.93). On the other hand, fiberoptic segmentation had slightly higher specificity than digital segmentation (0.91 vs. 0.92). Table 6.1 summarizes the comparison of model performance metrics on digital and fiberoptic videos. Figure 6.2 shows example an example segmentation for a digital and fiberoptic frame.

6.3.2 Fragmentation vs. Dusting

Additionally, our model performed similarly during stone fragmentation compared to stone dusting. The model performed better sensitivity (0.52 vs. 0.41) and ROC-AUC (0.87 vs. 0.77) for fragmentation whereas model performance on dusting had higher pixel accuracy (0.73 vs. 0.80) and specificity (0.96 vs. 0.97). Table 6.2 summarizes the comparison of model performance metrics on examples of fragmentation and dusting treatments. Figure 6.3 shows example an example segmentation on frames of fragmentation and dusting.

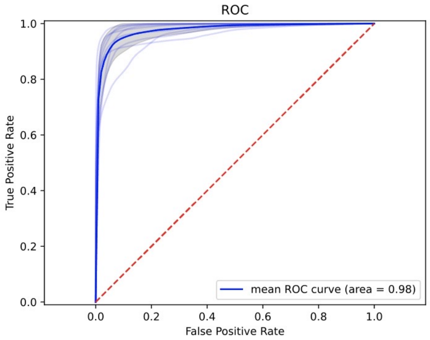
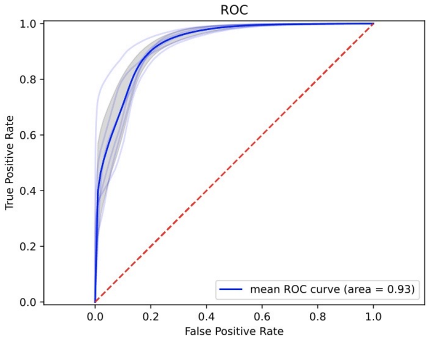
	Digital scope identification	Fiberoptic Scope Identification
Accuracy	**0.92**	0.87
Sensitivity	**0.94**	0.64
Specificity	0.91	**0.92**
AUC	**0.98**	0.93
ROC		

Table 6.1: Comparison of the model performance on digital and fiberoptic videos, after training only on digital videos. As expected, the model performed better on digital frames and still achieved decent performance on fiberoptic frames.

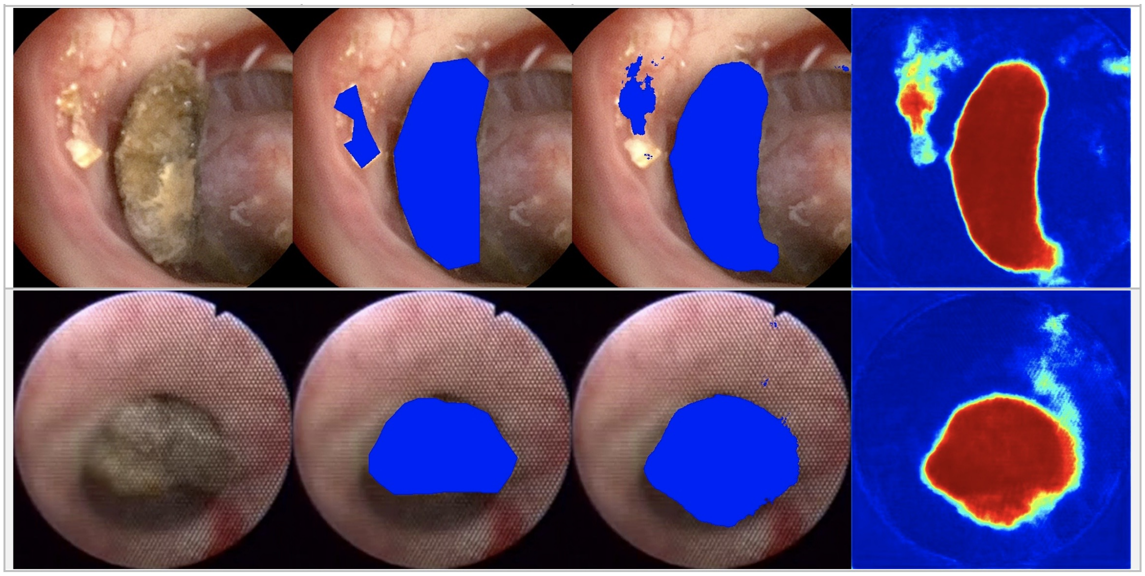


Figure 6.2: Comparison of the model's segmentation on digital (top) and fiberoptic (bottom) examples. The panels show the input, ground truth, automated prediction with U-Net++, and heat map (left to right).

6.3.3 Kappa Agreement

The Cohen's kappa agreement was 0.8 for the comparative analysis. Figure 6.4 summarizes the binary classification results for the comparative analysis in confusion matrices which were then used compute the kappa score.

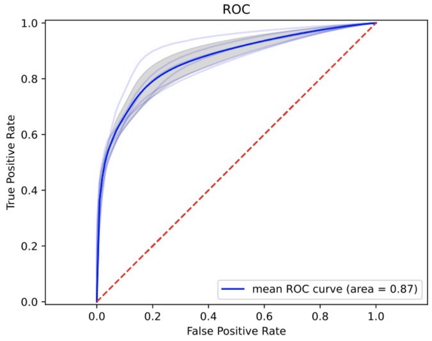
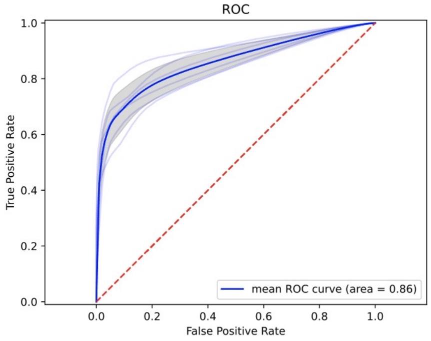
	Stone Fragmentation	Stone Dusting
Accuracy	0.73	**0.80**
Sensitivity	**0.52**	0.41
Specificity	0.96	**0.97**
AUC	**0.87**	0.77
ROC		

Table 6.2: Comparison of the model performance on fragmentation and dusting treatment examples. Performance in these two scenarios trades off with the model performing better on fragmentation in terms of sensitivity and ROC-AUC whereas it scores higher on Dice and specificity on dusting.

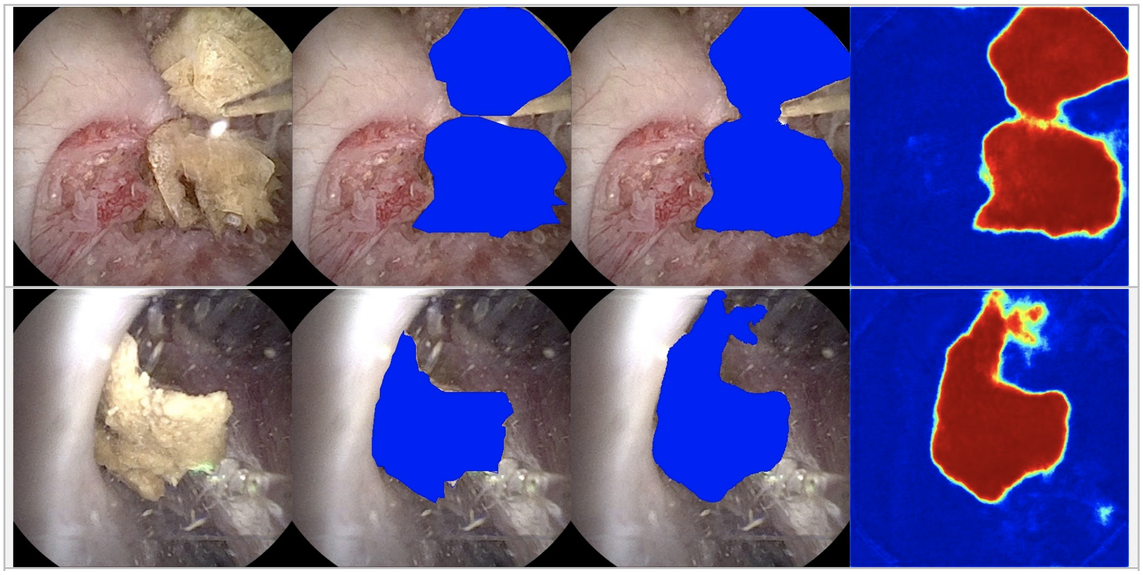
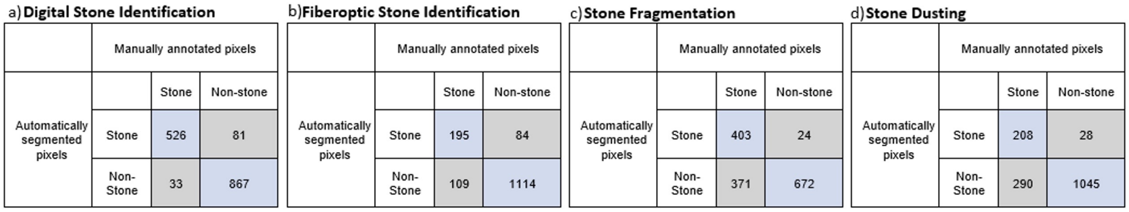


Figure 6.3: Comparison of the model's segmentation on fragmentation (top) and dusting (bottom) examples. The panels show the input, ground truth, automated prediction with U-Net++, and heat map (left to right). Empirically, predicted segmentations appear better for fragmentation.

6.3.4 Live System

The live stream was captured at 30 FPS for 1800 seconds for a total of 53970 frames. 3171 of these frames were segmented by the model resulting in 1.76 FPS, due to the low processing power of the computer. Figure 6.5 shows various unlabeled prediction examples from the live system.



a) **Digital Stone Identification**

	Manually annotated pixels		
		Stone	Non-stone
Automatically segmented pixels	Stone	526	81
	Non-Stone	33	867

b) **Fiberoptic Stone Identification**

	Manually annotated pixels		
		Stone	Non-stone
Automatically segmented pixels	Stone	195	84
	Non-Stone	109	1114

c) **Stone Fragmentation**

	Manually annotated pixels		
		Stone	Non-stone
Automatically segmented pixels	Stone	403	24
	Non-Stone	371	672

d) **Stone Dusting**

	Manually annotated pixels		
		Stone	Non-stone
Automatically segmented pixels	Stone	208	28
	Non-Stone	290	1045

Figure 6.4: Confusion matrices for digital, fiberoptic, fragmentation, and dusting videos (left to right). These binary classification results were then used to compute the kappa score (0.8).

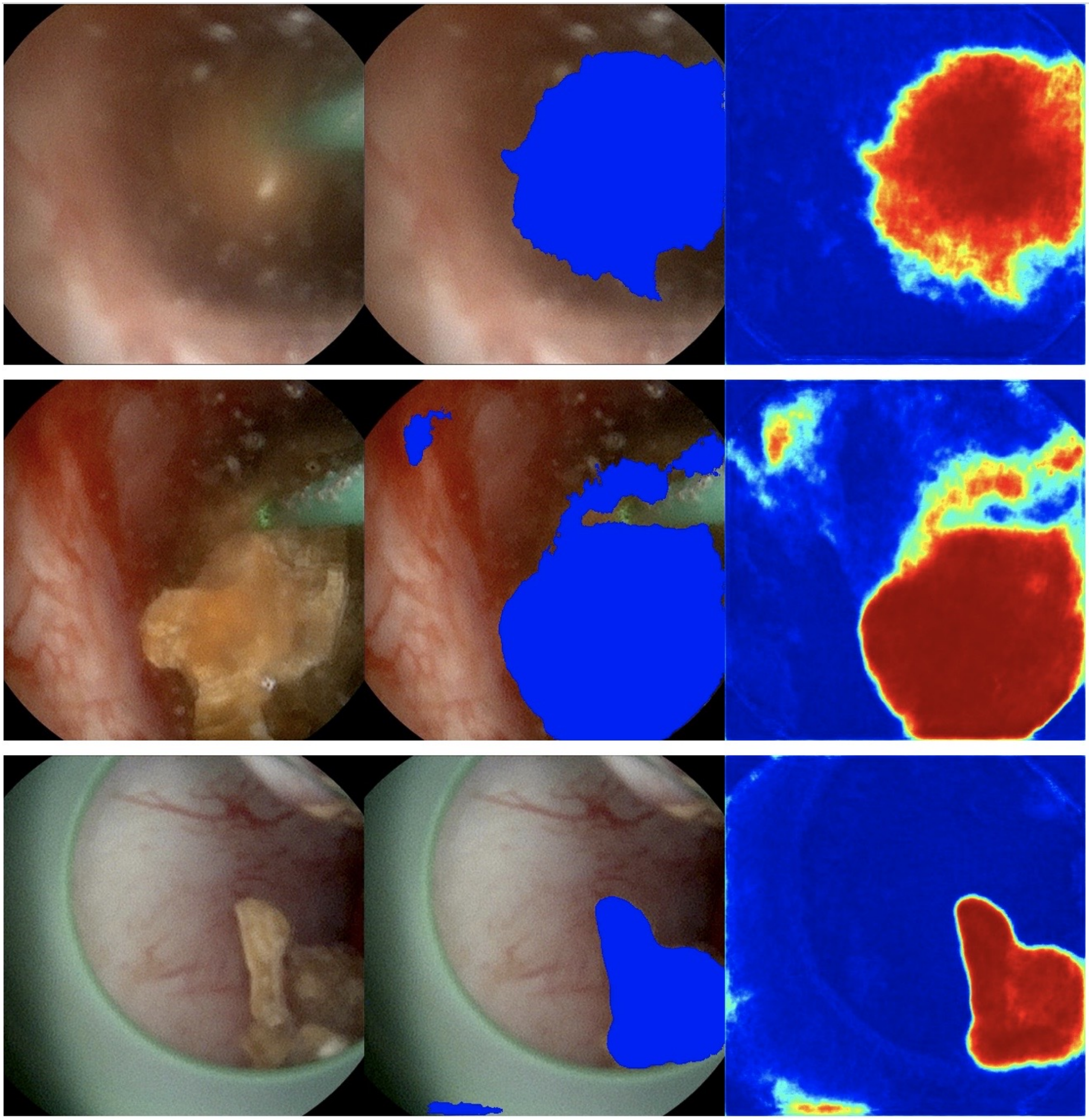


Figure 6.5: Example unlabeled predictions from live deployment of the system showing heavy obfuscation from saline (top), saline and laser treatment (middle), and fragmentation (bottom).

6.4 Summary & Interpretation

6.4.1 Comparative Analysis

The model's performances in the digital vs. fiberoptic comparison were expected. Since the model trained on digital scope videos, it follows that the model is most prepared to make predictions on that modality compared to the new one. Interestingly, the model's performance on fiberoptic videos were not empirically poor, suggesting that the model can handle this modality and assist surgeons in its current state. Moving forward, continued training on fiberoptic examples could improve performance on this modality. However, there is the possibility that the model could undergo "forgetting" of the digital videos on which it originally trained [45], resulting in mediocre performance for both modalities. If this is the case, then training a similar high-performing U-Net++ model on a dataset of fiberoptic examples to specifically address these types of procedures would alleviate the issue. Hence, for either digital or fiberoptic cases, the specific model could easily be loaded and result in high-quality segmentations, granting our system coverage for the two primary endoscopic modalities.

Fragmentation and dusting results were also expected and performance between the two was approximately the same. Compared to common digital examples without any treatment occurring, model performance was slightly reduced for these scenarios with more false positives due to more complicated imagery. Model performance could be improved for fragmentation and dusting with continued training on an expanded dataset with more examples of these treatments.

Overall, the kappa score suggests substantial agreement between model and expert annotations. This agreement implies that the model promisingly annotates frames very closely to what an expert's annotation would be, which bodes well for future clinical adoption of the model in a live scenario.

6.4.2 Live Deployment

As a proof-of-concept, the live system performed well in situ. We anticipate that specialized hardware will be needed for optimal performance to streamline clinical adoption. Particularly, conventional CPU hardware is not optimized for fast matrix computation and are known to be slow for deep learning applications compared to GPU hardware [46]. In the future, a mobile GPU-enabled workstation will be used to process frames at a realistic and faster rate. Potentially, a laptop with an Apple M1 processor may also be sufficient to process frames at an acceptably high rate. Further development of this deployed system will allow us to investigate additional goals, such as monocular depth prediction which might prove critical in fully robotic automation in the future [47].

CHAPTER 7

Conclusion

In this thesis, we present an exploratory analysis of automated kidney stone segmentation from ureteroscopic videos using supervised learning models. Our deep learning models achieve promising performance in this novel application domain, which further credits the utility of deep learning in image and video segmentation. With pretraining on ImageNet, we found that U-Net++ is the best-performing model for the task, followed closely by U-Net, while DenseNet performed the worst.

After selecting a high-performing model, we performed a comparative analysis of the model's performance on digital vs. fiberoptic feeds and fragmentation vs. dusting treatments relative to expert annotation. The findings suggest substantial agreement between model and expert segmentations, which is promising for widespread clinical adoption of the live system. In addition to practical analysis, we also deployed the model in a live system in real ORs during real procedures, demonstrating our proof-of-concept to guide future work.

7.1 Future Directions

We will also investigate the application of temporal models for our system. Incorporating information from previous frames might allow for more consistent prediction between subsequent frames. In addition, such models account for memory of data from previous frames without the overhead of additional input dimensions suffered by our current fully convolutional models. For this task, we will incorporate temporal attention modules into our U-Net and U-Net++ architectures, which has been shown to increase performance in video segmentation [48].

In practice, the problem domain also requires the segmentation of kidney stones after they have been surgically broken down into smaller pieces. As seen in Figures 5.5 & 6.3, our model already segments fragmented stones and clumps of dusting debris; however, our future goal is finer granularity and improved model performance via instance segmentation. Further development will include expansion of another section of the dataset where, as a surgeon breaks stones apart, the model will learn to label debris [49].

Additionally, we plan to incorporate multi-class segmentation to also identify, for example, healthy vs. unhealthy tissue. Since the task performs well on stone segmentation, we hypothesize that we can utilize the same underlying architectures to adapt to multi-class segmentation and that the model will perform similarly well, even with relatively few manually annotated examples [50].

All in all, our future research will be directed towards an automated visual control system for, ultimately, fully robotic surgery. Leveraging the temporal nature of the input domain to generate as high-quality segmentations as possible and expanding the system's capacity for useful segmentation are critical for physicians to interpret the resulting information efficaciously in real time. With these capabilities, such a system could be widely adopted in surgical practice and improve patient outcomes after surgical endoscopy.

References

[1] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, 2017.

[2] A. Ioannidou, E. Chatzilari, S. Nikolopoulos, and I. Kompatsiaris, "Deep learning advances in computer vision with 3d data: A survey," *ACM computing surveys (CSUR)*, vol. 50, no. 2, pp. 1–38, 2017.

[3] X. He, B. J. Guo, Y. Lei, T. Wang, T. Liu, W. J. Curran, L. J. Zhang, and X. Yang, "Automatic epicardial fat segmentation in cardiac ct imaging using 3d deep attention u-net," in *Medical Imaging 2020: Image Processing*, vol. 11313, pp. 589–595, SPIE, 2020.

[4] M. Kim, J. Yun, Y. Cho, K. Shin, R. Jang, H.-j. Bae, and N. Kim, "Deep learning in medical imaging," *Neurospine*, vol. 16, no. 4, p. 657, 2019.

[5] A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on mri," *Zeitschrift für Medizinische Physik*, vol. 29, no. 2, pp. 102–127, 2019.

[6] S. Safi, T. Thiessen, and K. J. Schmailzl, "Acceptance and Resistance of New Digital Technologies in Medicine: Qualitative Study," *JMIR Res Protoc*, vol. 7, p. e11072, Dec 2018.

[7] A. Nithya, A. Appathurai, N. Venkatadri, D. R. Ramji, and C. Anna Palagan, "Kidney disease detection and segmentation using artificial neural network and multi-kernel k-means clustering for ultrasound images," *Measurement: Journal of the International Measurement Confederation*, vol. 149, p. 106952, jan 2020.

[8] K. Viswanath and R. Gunasundari, "Design and analysis performance of kidney stone detection from ultrasound image by level set segmentation and ANN classification," in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, sep 2014.

[9] K. M. Black, H. Law, A. Aldoukhi, J. Deng, and K. R. Ghani, "Deep learning computer vision algorithm for detecting kidney stone composition," *BJU international*, vol. 125, no. 6, pp. 920–924, 2020.

[10] N. Große Hokamp, S. Lennartz, J. Salem, D. Pinto dos Santos, A. Heidenreich, D. Maintz, and S. Haneder, "Dose independent characterization of renal stones by means of dual energy computed tomography and machine learning: an ex-vivo study," *European Radiology*, vol. 30, no. 3, pp. 1397–1404, 2020.

[11] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9351, (Cham), pp. 234–241, Springer International Publishing, 2015.

[12] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, J. Liang, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: Redesigning Skip Connections to Exploit Multiscale Features in Image Segmentation," *IEEE Transactions on Medical Imaging*, vol. 39, no. 6, pp. 1856–1867, 2020.

[13] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation," *IEEE CVPR Workshops*, pp. 1175–1183, 2017.

[14] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *IEEE CVPR*, vol. 2017-Janua, pp. 2261–2269, 2017.

[15] Z. A. Stoebner, D. Lu, S. H. Hong, N. L. Kavoussi, and I. Oguz, "Segmentation of kidney stones in endoscopic video feeds," in *Medical Imaging 2022: Image Processing*, vol. 12032, pp. 900–908, SPIE, 2022.

[16] S. A. Setia, Z. A. Stoebner, C. Floyd, D. Lu, I. Oguz, and N. L. Kavoussi, "Computer vision enabled segmentation of kidney stones during ureteroscopy and laser lithotripsy," *Journal of Endourology [submitted]*, 2022.

[17] P. Aslan, R. L. Kuo, K. Hazel, R. K. Babayan, and G. M. Preminger, "Advances in digital imaging during endoscopic surgery," *Journal of endourology*, vol. 13, no. 4, pp. 251–255, 1999.

[18] C. H. Bindu and K. S. Prasad, "An efficient medical image segmentation using conventional otsu method," *International Journal of Advanced Science and Technology*, vol. 38, no. 1, pp. 67–74, 2012.

[19] J. E. Iglesias and M. R. Sabuncu, "Multi-atlas segmentation of biomedical images: a survey," *Medical image analysis*, vol. 24, no. 1, pp. 205–219, 2015.

[20] C. Xu, D. L. Pham, and J. L. Prince, "Image segmentation using deformable models," *Handbook of medical imaging*, vol. 2, no. 20, p. 0, 2000.

[21] W. Skarbek, A. Koschan, T. Bericht, Z. Veroffentlichung, and P. D. Klette, "Colour image segmentation - a survey," 1994.

[22] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8828, no. c, pp. 1–20, 2021.

[23] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 international conference on engineering and technology (ICET)*, pp. 1–6, Ieee, 2017.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *IEEE CVPR*, vol. 2016-Decem, pp. 770–778, 2016.

[25] M.-H. Guo, T.-X. Xu, J.-J. Liu, Z.-N. Liu, P.-T. Jiang, T.-J. Mu, S.-H. Zhang, R. R. Martin, M.-M. Cheng, and S.-M. Hu, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, pp. 1–38, 2022.

[26] A. G. Roy, N. Navab, and C. Wachinger, "Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks," *IEEE transactions on medical imaging*, vol. 38, no. 2, pp. 540–549, 2018.

[27] N. M. Zaitoun and M. J. Aqel, "Survey on Image Segmentation Techniques," in *Procedia Computer Science*, vol. 65, pp. 797–806, Elsevier, jan 2015.

[28] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 2481–2495, dec 2017.

[29] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," jun 2017.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, International Conference on Learning Representations, ICLR, sep 2015.

[31] D. Manzey, G. Strauss, C. Trantakis, T. Lueth, S. Röttger, J. Bahner-Heyne, A. Dietz, and J. Meixensberger, "Automation in surgery: a systematic approach," *Surg Technol Int*, vol. 18, pp. 37–45, 2009.

[32] A. Pandya, L. A. Reisner, B. King, N. Lucas, A. Composto, M. Klein, and R. D. Ellis, "A review of camera viewpoint automation in robotic and laparoscopic surgery," *Robotics*, vol. 3, no. 3, pp. 310–329, 2014.

[33] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[34] P. Skalski, "Make Sense." https://github.com/SkalskiP/make-sense/, 2019.

[35] Itseez, "Open source computer vision library." https://github.com/itseez/opencv, 2015.

[36] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," *IEEE CVPR*, vol. 2017-Janua, no. 1, pp. 311–320, 2017.

[37] A. Tkachenka, G. Karpiak, A. Vakunov, Y. Kartynnik, A. Ablavatski, V. Bazarevsky, and S. Pisarchyk, "Real-time hair segmentation and recoloring on mobile GPUs," jul 2019.

[38] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[39] I. Z. Yalniz, H. Jégou, K. Chen, M. Paluri, and D. Mahajan, "Billion-scale semi-supervised learning for image classification," 2019.

[40] T. Sørensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons"," *Kongelige Danske Videnskabernes Selskab*, vol. 5, no. 4, p. 1–34, 1948.

[41] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," *Advances in neural information processing systems*, vol. 31, 2018.

[42] V. Iremashvili, S. Li, K. L. Penniston, S. L. Best, S. P. Hedican, and S. Y. Nakada, "Role of Residual Fragments on the Risk of Repeat Surgery after Flexible Ureteroscopy and Laser Lithotripsy: Single Center Study," *J Urol*, vol. 201, pp. 358–363, 02 2019.

[43] T. M. Ward, P. Mascagni, Y. Ban, G. Rosman, N. Padoy, O. Meireles, and D. A. Hashimoto, "Computer vision in surgery," *Surgery*, vol. 169, no. 5, pp. 1253–1256, 2021.

[44] S. Sun, "Meta-analysis of cohen's kappa," *Health Services and Outcomes Research Methodology*, vol. 11, no. 3, pp. 145–163, 2011.

[45] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[46] Y. E. Wang, G.-Y. Wei, and D. Brooks, "Benchmarking tpu, gpu, and cpu platforms for deep learning," *arXiv preprint arXiv:1907.10701*, 2019.

[47] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2002–2011, 2018.

[48] G.-P. Ji, Y.-C. Chou, D.-P. Fan, G. Chen, H. Fu, D. Jha, and L. Shao, "Progressively normalized self-attention network for video polyp segmentation," *arXiv preprint arXiv:2105.08468*, 2021.

[49] Y. Zhou, O. F. Onder, Q. Dou, E. Tsougenis, H. Chen, and P.-A. Heng, "Cia-net: Robust nuclei instance segmentation with contour-aware information aggregation," in *International Conference on Information Processing in Medical Imaging*, pp. 682–693, Springer, 2019.

[50] B. Kayalibay, G. Jensen, and P. van der Smagt, "Cnn-based segmentation of medical imaging data," *arXiv preprint arXiv:1701.03056*, 2017.