Computer vision surgical skill assessment using operative video

Rory Holmes

***Abstract*—This report focuses on assessing surgical skill within colorectal surgery. Although operative reviews have shown to be valuable in competency-based surgical training, manual reviews are too time-consuming to be a routine practice. Colorectal surgery has shown potential for skill assessment due to the correlation between Vascular Pedicle Dissection Time (VPDT) and manually assessed competency scores. This paper proposes a CNN that will be used to extract relevant features from laparoscopic videos combined with a 3D-CNN to quantify the VPDT and used to calculate an automatically produced competency score. The initial outcome of this approach is a feature extractor with 81% accuracy and an F1 score of 76%. While the phase detector underperformed with an accuracy of 29% and an F1 score of 3.9%.**

I. Introduction

A. Background

This is an individual project conducted by Rory Holmes, supervised by James Atlas and is partnered with Isaac Tranter-Entwistle from The Department of Surgery and Critical Care at the University of Otago and Christchurch. The Department of Surgery and Critical Care undertake research aiming to improve patient care and to optimize surgical outcomes. Operative video reviews have shown potential to be a valuable tool in competency-based surgical training, allowing for the better education of future surgeons [4] [5]. Despite this value, manual review of operative videos are time-consuming for skilled surgeons. Given this time intensive nature of evaluating operative skill, it is not a routine practice. However, the introduction of automated techniques for gauging technical skill in surgery provides an assessment of its association with adverse outcomes in patients.

This report focuses on assessing surgical skill within colorectal surgery, specifically a right hemicolectomy and anterior resection, which involve removing a part of the colon and rectum. In colorectal surgery, Vascular Pedicle Dissection Time (VPDT), the time taken to dissect the vascular pedicle, has shown potential for automation due to its correlation with manually assessed Competency Assessment Tools (CAT) [15].

B. Objectives

The objective of this project is to create, implement, and evaluate a phase detection algorithm that quantifies the VPDT measurement in order to address the following research questions:

1) How accurately can relevant surgical information be automatically classified?
2) What is the accuracy of automated VPDT measurements?
3) How closely can a CAT score be approximated from an automated VDPT measurement?

C. Content

In section II, both the related work to automatic surgical skill assessment is discussed, and the best available datasets for feature extraction. Section III discuses the methodology and project management processes utilised in the creation of software artefacts for this project, while Section IV discusses the results of these artefacts. Then section V details the discussion of the results and the evaluation of the strengths and weaknesses of the project. Lastly, Section VI provides a conclusion for this report.

II. Related Work

In the realm of computer vision for surgeries in the colon, previous studies have primarily focused on surgeries such as cholecystectomies due to the availability of the Cholec80 dataset, an annotated dataset containing 80 laparoscopic videos [18] [19]. While some progress has been made towards colorectal surgeries and the potential for automatic skill assessment, a lack of data in this area is hindering progress [11] [17]. The papers that have explored colorectal surgery either stop after obtaining temporal annotations [13] or have explored skill assessment but through other means [12]. This project differs from the related work due to its goal of exploring the automatically produced VPDT and its correlation to the CAT score for evaluating surgical skill.

The recognition of tool presence has shown to be an essential step in surgical workflow analysis [14]. Due to this reason, the relevant surgical information to be automatically classified in this paper will be a binary classification of what tools are present in each frame of the laparoscopic videos.

When searching for annotated datasets of laparoscopic surgery, other research recommended the Cholec80 dataset for tool classification [2] [7]. The use of this dataset was also confirmed with industry partners. The Cholec80 dataset will be used for tool classification due to the correlation in tools between colorectal and cholecystectomy surgeries and the lack of appropriate datasets for colorectal surgery.

Various models have been utilised for transfer learning in the classification of surgical tools within the Cholec80 dataset. Transfer learning is taking a pretrained network and applying it to recognise new categories of images in order to improve model performance and reduce the need for large amounts of data. This paper explores three options of pretrained networks due to their presence in other research: InceptionResnetV2 [7], Resnet50 [10], and VGG16 [9].

Phase detection in laparoscopic surgery has been done through different means. One study posed a CNN architecture, EndoNet, that is trained to carry out phase recognition and tool presence detection simultaneously [20]. However, this underperforms the most common approach where a CNN is used in conjunction with a LSTM to quantify both spatial and temporal information [21] [1]. This approach of using a CNN for feature extraction and an LSTM achieves an overall accuracy of around 90% as opposed to 86% with EndoNet [16]. However, while these solutions work well for cholecystectomy surgery, none of these solutions have been tried on colorectal surgery.

III. Method

This project follows an iterative software process, allowing for incremental improvements with each iteration. The initial scope of the project is to obtain feature extraction of tools which will be used for phase detection in the future of the project to calculate VPDT. This approach was taken as it allows for greater flexibility and improved risk management as problems can be identified early.

Additionally, an iterative development process is effective for receiving and incorporating feedback from industry partners and my supervisor. Feedback has typically been collected on a weekly or fortnightly basis, through emails as well as online and in person meetings, ensuring progress towards identified milestones. This frequent feedback cycle has allowed for improved identification and integration of useful datasets. For instance, the initial cycle for CNN feature extraction utilised the EndoVis'15 dataset, which contained 40 images with tool segmentation masks. Upon communication with industry partners, this dataset was passed up due to its small size being insufficient for training robust models, and directed to the Cholec80 dataset with tool classification annotations instead. Furthermore, frequent communication with my supervisor has provided insight into the strengths and weaknesses of different machine learning architecture and assisted my decision to stay away from a LSTM approach to phase classification.

To track progress and to prevent any loss of work, GitHub has been used in order to track and version source code, utilising a task based branching strategy. The main branch is reserved for completed models after a development cycle, while new branches are created for working with new datasets, and further branching is done for re-engineering or to address significant issues. Furthermore, documentation of progress, future tasks, and issues faced during implementation is stored in GitHub alongside the code.

Jira is used to track tasks and manage project workflows through a Kanban board, allowing for easy prioritization and monitoring of task progress. Each task is assigned with clear deadlines and labels, such as "TODO","In progress," "Done," or "Blocked," which helps maintain visibility on the project's status. Time logging is integrated into Jira, enabling the tracking of hours spent on individual tasks to ensure accurate estimates for project planning. An example of this is displayed in figure 1.

Unit tests have been incorporated throughout the development process to ensure the functionality of individual components and to detect any errors brought about by the iterative process early. Tests are written in parallel with code development, this ensures that each method is tested in isolation, providing confidence in the correctness of low-level components before they are integrated into the broader system. Additionally, testing is integrated into the version control process, with all tests being run on a branch before it is merged into the main branch to prevent the introduction of faulty code.

To ensure the project can be easily picked up and run by others in the future, a comprehensive text file was created that lists all the necessary libraries and their corresponding versions. This guarantees consistency across different environments, ensuring that anyone that may work on this project in the future will have the same setup, reducing the risk of compatibility issues. Additionally, the project includes methods for setting up the required folder structure for training the models, simplifying the onboarding process for any future contributors. This ensures that everything from dependencies to the data organization is clearly defined, allowing for easy replication of the training pipeline.

In order to evaluate the quality of the models made in this project, the metrics utilised are accuracy as well as precision, recall, and F1 Score. These metrics are tracked across epochs during the validation stage of training in order to monitor performance improvements and track for overfitting.

IV. Results

A. Design

The objective of accurately classifying relevant surgical information focuses on developing models capable of identifying key elements within surgical procedures. Due to previous research, the relevant information being focused on is a classification of what surgical tools are within each frame [14], through the use of a Convolutional Neural Network (CNN). The use of a CNN for feature extraction is justified by its proven ability to efficiently capture spatial hierarchies and patterns in image data, making it highly effective for visual tasks like surgical phase detection or tool classification [8].

In order to answer the rest of the objectives, a method for calculating VPDT is necessary. The key to identifying VPDT is detecting when each phase of the vascular pedicle dissection begin and end, this problem is considered a surgical phase detection problem. The colorectal surgery is thus split into four phases: time retraction start, time dissection start, time vessel ligated, and completing dissection. Hence, the surgical phase recognition task is a multi-class classification problem.

In order to solve this problem, various papers were analysed to identify suitable model structures for phase detection. Previous research indicates that a CNN for feature extraction combined with an LSTM model for phase detection is the optimal approach [21] [1]. An LSTM is generally utilized due to its robustness against the vanishing gradient problem, while a CNN is combined with a LSTM and independently trained to avoid convergence to a poor local optima [6] [3].

However, while LSTMs are highly effective for sequence-based tasks, such as phase detection, they may not be the ideal solution in this case due to the limited amount of phase annotations available. LSTMs excel when provided with large datasets to capture long-term dependencies, but the data constraints in this project make it challenging to fully leverage their capabilities. Additionally, handling high-dimensional image data with LSTMs can be complex and computationally intensive, especially when the feature space is large. Instead, a 3D-CNN is better suited for this problem. By stacking video frames into 3D blocks, a 3D-CNN can capture both spatial and short-term temporal features efficiently, allowing for accurate phase detection without requiring vast amounts of phase annotations [22]. The ability to select specific frames for temporal context further enhances the 3D-CNN's capability to capture relevant transitions, making it a more practical and efficient choice for this particular task.

When using a CNN to extract relevant features of the laparoscopic surgery, a relevant and comprehensive dataset is necessary. Cholec80 contains 80 videos performed by 13 surgeons. Video resolutions are 1920 x 1080 pixels and a frame rate of 25 Frames Per Second (fps) with tool presence annotations at 1 fps for binary classification. Each videos' length is varied between 12 minutes and 1 hour 39 minutes. Cholec80 has seven tools annotated, being: specimen bag, bipolar, scissors, clipper, hook, grasper, and irrigator. There are some differences in tools between cholecystectomy and colorectal surgeries, mainly the lack of a specimen bag being used and plastic tools being more common than metal ones. Within Cholec80, a tool is considered present in the image only if over half of the tool is present, thus this project continues with this definition. Additionally, for each annotated frame one binary label is provided per tool, hence the feature extraction CNN is also considered a multi-label classification problem.

The colorectal surgery dataset provided in this project contains 192 laparoscopic videos. Of those videos, 70 do not have proper annotations due to various reasons, such as the video starting too late into the surgery. The annotations in the remaining 122 videos cover tool presence at key moments of vascular pedicle dissection, as well as time stamps for the various phases identified above. All videos have a resolution of 720 x 560 pixels and a frame rate of 25 fps. Each videos' length is varied between 52 minutes and 1 hour 42 minutes.

Due to the lack of comprehensive tool annotations in the colorectal dataset, it was not possible to train a new classification layer to evaluate the competency of the feature extractor developed using the cholecystectomy dataset. However, since this classification layer is removed and only the feature maps are being passed into the 3D-CNN, this does not affect the performance of the phase detector. Thus, the model structure decided for phase detection is a combination of a CNN for feature extraction which is trained to classify tool presence within frames. The classification layer is then removed, and the feature maps are then stacked linearly into 3D blocks with phase annotations corresponding to each frame. The output of this model is a phase classification for each frame within the 3D block.

B. Implementation

1) Feature Extraction: In order to match tool presence annotations, the Cholec80 dataset is initially preprocessed to only retain 1 frame per second. Additionally, all frames need to be reduced to a fixed size to be trained and are thus resized to 256 x 256 pixels and with normalised pixels between 0 and 1. A data generator is made to retrieve these preprocessed frames in conjunction with the annotated labels for training. A generator is used to yield batches of training data in order to limit memory usage during training.

Due to previous research with Cholec80 and tool classification, three pretrained models all pretrained on the 'ImageNet' dataset were evaluated on their performance: VGG16, InceptionResnetV2, and ResNet50. Each model needs to be altered in order to adapt the models to the new classification problem of identifying the presence of the seven surgical tools. This is done by removing the fully connected layer at the end of the pretrained networks, this layer contained the 1000 classes from the ImageNet dataset labels, a new sigmoid layer that is compatible with seven classes for the number of surgical tools is then added in its place. This new layer assigns a probability between 0 and 1 for each of the seven tools being present within each frame tested. Due to the problem being a multiclass classification, the final activation function is a Sigmoid function defined as,

$$S(x) = \frac{1}{1 + e^{-x}}$$

Where e = Euler's number.

After this customisation of the models, each are trained, however, due to time constraints, each model has only been trained on three epochs in order to get an initial assessment of their capability. Furthermore, a low training rate of 0.0001 has been used, as it is expected that the pretrained weights are reasonably accurate and only need to be fine-tuned to the type of classification. Additionally, binary Cross-Entropy is used as a Loss function, this is calculated as:

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i))$$

Where y is the label (1 for the tool being present and 0 for absence) and p(y) is the predicted probability of the class. The performance of each of these models on the tool classification task is measured by both the F1 Score and accuracy on the validation data. The F1 Score is calculated as:

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

With precision and recall being defined as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Figure 2 displays the F1 Scores for each model across the 3 epochs they are trained with. InceptionResnetV2 performs the best under this metric by the final epoch with a score of 66%, ResNet50 is second with 52%, and VGG16 is last with 43%. The classification accuracy of each model can be seen in fig 3. ResNet50 performed the best when evaluated with the accuracy of validation data, with an accuracy of 73% by the third epoch. InceptionResnetV2 had the lowest accuracy by the third epoch with 60% accuracy.

The model used for feature extraction going forward is the InceptionResnetV2 model based on its superior F1 score, despite lower accuracy compared to ResNet50. While ResNet50 achieved higher accuracy of 73% compared to InceptionResNetV2s of 60%, accuracy does not account for the trade-offs between false positives and false negatives. InceptionResNetV2's higher F1 score of 66% reflects its better ability to correctly identify relevant features and minimize classification errors, making it the more reliable choice for extracting meaningful features for this project.

The next step involved freezing all layers of the base model, InceptionResNetV2, except for the classification layer. This was followed by implementing a custom callback to monitor the validation loss during training. When the validation loss stopped decreasing with a patience of two epochs, the callback would unfreeze 25 layers of the 164-layer network and resume training. Freezing the initial layers ensures that they retain the general features learned from the original dataset, preventing them from being negatively impacted by further training on the Cholec80 dataset. By allowing only the final layers to be trained, the model can adapt and learn specific features relevant to the surgical dataset, while maintaining the integrity of the pre-trained layers. This approach balances the transfer of knowledge from the base model with the need for fine-tuning on domain-specific data. This model was trained over 8 epochs, with the validation F1 score of each class shown as in fig 4. The model performed well with the grasper and hook classes while underperforming on classes such as the clippers and scissors. This difference may be due to an imbalance of tool presence within the dataset itself, suggesting that data augmentation of the less prevalent classes may improve their classification. The overall F1 score is 70% and accuracy 77%, showing the improvement of incrementally unfreezing the layers of the network.

2) *Phase Detection:* The preprocessing for each frame of the colorectal dataset was performed the same as with the cholec80 dataset to ensure consistency and compatibility within the training process. Each frame was resized to 256 x 256 pixels, with normalised pixels between 0 and 1. Phase annotations for each frame are stored within an Excel spreadsheet which record the time in seconds marking the transition into each phase, these were used to return a one hot encoding for each frame calculated with the fps of the video. Thus, the annotations for each frame are a list of 0's with length 4, where 1 marks the present phase. The data generator extracts features from the raw frames using InceptionResnetV2, then stacks them into 3D blocks suitable for input to the 3D-CNN, and yields these batches along with their labels for training.

The phase detections model structure is designed as a 3D-CNN that leverages the features extracted from the InceptionResNetV2 architecture. The model begins with an input layer that accepts a linear sequence of 5 frames, where each frame is represented by a 6×6 feature map with 1536 channels outputted from InceptionResnetV2. The architecture includes multiple TimeDistributed convolutional blocks, each comprising Conv2D layers followed by MaxPooling2D and BatchNormalization layers, allowing the model to learn spatial features across the sequence of frames. The TimeDistributed Conv2D layers apply convolutional operations across each frame independently, defined by the equation:

$$H_c(x) = W * x + b$$

Where W represents the filter weights, x is the input frame, and b is the bias term. After convolution, MaxPooling2D layers reduce the spatial dimensions while preserving important features. BatchNormalization layers standardize the outputs of the previous layer to improve training stability and speed, described by:

$$H_{bn}(x) = \gamma \cdot \frac{x - \mu}{\sigma} + \beta$$

Where μ and σ are the mean and variance of the batch, and γ and β are learnable parameters.

Following the convolutional and pooling layers, a Global Average Pooling layer computes the average of each feature map, resulting in a single vector per feature map. Finally, the output layer is also TimeDistributed, producing a softmax activation for multi-class classification across the specified number of classes. The softmax function is defined as:

$$p(y = k|x) = \frac{e^{H(x)_k}}{\sum_{j=1}^{C} e^{H(x)_j}}$$

where C is the total number of classes, and $H(x)_k$ is the output vector for class k. The model is then compiled and trained with categorical cross-entropy loss:

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

Where N is the number of samples, C is the number of classes, y_{ij} is the true label and $\hat{y}_{ij}$ is the predicted probability.

The validation results of the phase detection model, as shown in figure 5, indicate a final accuracy of only 29% after 4 epochs. Despite the high precision of 98%, the recall is extremely low at just 2%, leading to a final F1 score of 3.9%. These metrics highlight the significant imbalance in the model's performance, where it is able to correctly identify very few of the relevant phases. These results indicate that the model is not accurate in detecting the surgical phases and therefore does not perform well in addressing the objective of measuring the accuracy of automated VPDT measurements.

The low F1 score underscores the model's inability to generalize across the dataset effectively, making it unsuitable for reliable phase detection.

V. Discussion & Evaluation

Initially, one of the key challenges faced during the design of the feature extraction pipeline was finding an appropriate dataset. The provided colorectal surgery dataset lacked the necessary tool annotations for accurate tool classification, which complicated the process. To address this, the EndoVis'15 dataset was initially used, it contained 40 2D in-vivo images from four laparoscopic colorectal surgeries, along with annotated masks. The assumption was that annotation masks would provide more accurate feature extraction compared to binary tool presence labels. However, the small dataset led to poor performance, causing delays in the project timeline. Fortunately, the much larger Cholec80 dataset, which includes annotations for cholecystectomy surgeries, was later found, and produced significantly better results for the first objective of evaluating the accuracy of surgical information classification.

Additionally, significant design limitation emerged from the tool differences between colorectal and cholecystectomy surgeries. Since the Cholec80 dataset was designed for cholecystectomy procedures, its tool annotations did not entirely match those required for colorectal surgeries. The tools also differ in the type of materials used, with the cholec80 dataset containing metal tools, contrasting with the plastic tools in the colorectal surgery. This mismatch would limit the feature extractor's effectiveness when applied to colorectal procedures. Although additional tool annotations for the colorectal dataset were later acquired, they were too limited in quantity to effectively retrain a classification layer that could generalize across both surgery types.

One notable success during implementation was the custom callback function used to unfreeze certain layers of the pretrained neural network during fine-tuning. By progressively unfreezing these layers, the model was able to extract more relevant features, improving classification performance. This technique led to significant improvements in both accuracy and F1 score, which supported the effectiveness of the approach. The model's accuracy improved from 72% to 81%, while the F1 score saw an increase from 68% to 76% after implementing this callback. These metrics demonstrated that the callback contributed to better feature extraction when using the Cholec80 dataset. Despite the improvements in F1 score and accuracy for the feature extractor, the performance still fell short of the expected benchmarks necessary to enhance phase detection accuracy. However, time constraints limited the depth of research that could be conducted into design and testing.

Time constraints especially effected the 3D-CNN architecture, which aimed to measure the accuracy of automated VPDT measurements. The resulting model underperformed, showing poor accuracy in phase detection. Specifically, the model exhibited high precision but low recall, suggesting that it was overly conservative in its predictions. This could have resulted from insufficient training time or a lack of fine-tuning for capturing temporal dependencies in the video sequences.

The poor phase detection results could also be attributed to class imbalance within the data, where certain phases are underrepresented in comparison to others. This imbalance makes it difficult for the model to learn meaningful features for the underrepresented phases, leading to skewed or inaccurate predictions. To address this, data augmentation could be employed to balance the dataset. One approach would involve cropping surgical videos to generate more training examples for the phases with less representation. Alternatively, synthetic data could be added through augmentation techniques such as rotation, scaling, or flipping, which would increase the diversity of the underrepresented phases, helping the model generalize better and improving phase detection performance.

Additionally, the choice of frame stacking can significantly impact model performance, and linear sampling was not the only option. Experimenting with non-linear sampling, such as using two frames to represent the context before and after a given frame, could have yielded better results. More research would be needed to determine the optimal sampling method, which might be crucial to optimize the models performance.

In addition to these considerations, further evaluation of the chosen performance metrics is warranted. While accuracy and F1 score are valuable, additional metrics such as precision-recall curves or confusion matrices may provide more insight into the model's performance, especially in terms of understanding the types of errors it makes. This can inform future iterations of the model, allowing for more targeted improvements.

VI. Conclusion

In conclusion, this report outlines the initial steps in developing an automated phase detection algorithm for measuring Vascular Pedicle Dissection Time (VPDT) in colorectal surgeries. This project aims to improve surgical training and identify complications in patient outcomes from the automation of the evaluation of surgical skill. The final objectives for this project of measuring VPDT accuracy, approximating Competency Assessment Tool (CAT) scores, and predicting post-operative complications are yet to be achieved.

However, progress has been made on the initial objectives of classifying relevant surgical information and automatically measuring VPDT. Despite achieving a reasonable F1 score of 70% and an accuracy of 77% for tool classification, challenges persisted in the phase detection model. The final results indicated a low accuracy of 29% with an F1 score of 3.9% for the phase detection task, hindered due to the limited quantity and quality of tool and phase annotations in the colorectal surgery dataset. While the project achieved some success in feature extraction, the phase detection model's performance demonstrated the need for further model refinement. This highlights the importance of well annotated datasets and research into appropriate model architectures.

Looking ahead, several long-term goals have been identified to enhance the outcomes of this project. First, there is a need to acquire a more comprehensive and diverse dataset that includes detailed annotations for both tool presence and surgical phases specific to colorectal procedures. This expansion will significantly improve the model's generalizability and performance. Additionally, it will be essential to investigate alternative phase model architectures and techniques that may better handle the unique challenges of calculating VPDT.

References

[1] Abdulbaki Alshirbaji, T., Jalal, N.A., Docherty, P.D., Neumuth, T., Möller, K.: A deep learning spatial-temporal framework for detecting surgical tools in laparoscopic videos. Biomedical Signal Processing and Control **68**, 102801 (2021). https://doi.org/https://doi.org/10.1016/j.bspc.2021.102801, https://www.sciencedirect.com/science/article/pii/S1746809421003980

[2] Abdulbaki Alshirbaji, T., Jalal, N.A., Möller, K.: Surgical tool classification in laparoscopic videos using convolutional neural network. Current Directions in Biomedical Engineering **4**(1), 407–410 (2018)

[3] Bengio, Y., Simard, P., Frasconi, P.: Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks **5**(2), 157–166 (1994). https://doi.org/10.1109/72.279181

[4] Celentano, V., Smart, N., Cahill, R.A., McGrath, J.S., Gupta, S., Griffith, J.P., Acheson, A.G., Cecil, T.D., Coleman, M.G.: Use of laparoscopic videos amongst surgical trainees in the united kingdom. The Surgeon **17**(6), 334–339 (2019)

[5] Celentano, V., Smart, N., McGrath, J., Cahill, R.A., Spinelli, A., Obermair, A., Hasegawa, H., Lal, P., Almoudaris, A.M., Hitchins, C.R., et al.: Lap-vegas practice guidelines for reporting of educational videos in laparoscopic surgery: a joint trainers and trainees consensus statement. Annals of Surgery **268**(6), 920–926 (2018)

[6] Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Computation **9**(8), 1735–1780 (11 1997). https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735

[7] Jaafari, J., Douzi, S., Douzi, K., Hssina, B.: Towards more efficient cnn-based surgical tools classification using transfer learning. Journal of Big Data **8**(1), 115 (2021)

[8] Jaafari, J., Douzi, S., Douzi, K., Hssina, B.: Towards more efficient cnn-based surgical tools classification using transfer learning. Journal of Big Data **8**(1), 115 (2021)

[9] Jaafari, J., Douzi, S., Douzi, K., Hssina, B.: The impact of ensemble learning on surgical tools classification during laparoscopic cholecystectomy. Journal of Big Data **9**(1), 49 (2022)

[10] Jalal, N.A., Alshirbaji, T.A., Docherty, P.D., Arabian, H., Neumuth, T., Möller, K.: Surgical tool classification & localisation using attention and multi-feature fusion deep learning approach. IFAC-PapersOnLine **56**(2), 5626–5631 (2023)

[11] Kitaguchi, D., Ito, M.: Computer vision in colorectal surgery: Current status and future challenges. Seminars in Colon and Rectal Surgery **35**(1), 101008 (2024). https://doi.org/https://doi.org/10.1016/j.scrs.2024.101008, https://www.sciencedirect.com/science/article/pii/S1043148924000071, technologic Advances in Colon and Rectal Surgery

[12] Kitaguchi, D., Takeshita, N., Matsuzaki, H., Igaki, T., Hasegawa, H., Ito, M.: Development and validation of a 3-dimensional convolutional neural network for automatic surgical skill assessment based on spatiotemporal video analysis. JAMA network open **4**(8), e2120786–e2120786 (2021)

[13] Kitaguchi, D., Takeshita, N., Matsuzaki, H., Oda, T., Watanabe, M., Mori, K., Kobayashi, E., Ito, M.: Automated laparoscopic colorectal surgery workflow recognition using artificial intelligence: Experimental research. International Journal of Surgery **79**, 88–94 (2020). https://doi.org/https://doi.org/10.1016/j.ijsu.2020.05.015, https://www.sciencedirect.com/science/article/pii/S1743919120303988

[14] Kondo, S.: Lapformer: surgical tool detection in laparoscopic surgical video using transformer architecture. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization **9**(3), 302–307 (2021)

[15] Miskovic, D., Ni, M., Wyles, S.M., Kennedy, R.H., Francis, N.K., Parvaiz, A., Cunningham, C., Rockall, T.A., Gudgeon, A.M., Coleman, M.G., et al.: Is competency assessment at the specialist level achievable? a study for the national training programme in laparoscopic colorectal surgery in england. Annals of surgery **257**(3), 476–482 (2013)

[16] Namazi, B., Sankaranarayanan, G., Devarajan, V.: Automatic detection of surgical phases in laparoscopic videos. In: Proceedings on the international conference in artificial intelligence (ICAI). pp. 124–130 (2018)

[17] Quero, G., Mascagni, P., Kolbinger, F.R., Fiorillo, C., De Sio, D., Longo, F., Schena, C.A., Laterza, V., Rosa, F., Menghi, R., et al.: Artificial intelligence in colorectal cancer surgery: present and future perspectives. Cancers **14**(15), 3803 (2022)

[18] Shinozuka, K., Turuda, S., Fujinaga, A., Nakanuma, H., Kawamura, M., Matsunobu, Y., Tanaka, Y., Kamiyama, T., Ebe, K., Endo, Y., et al.: Artificial intelligence software available for medical devices: surgical phase recognition in laparoscopic cholecystectomy. Surgical Endoscopy **36**(10), 7444–7452 (2022)

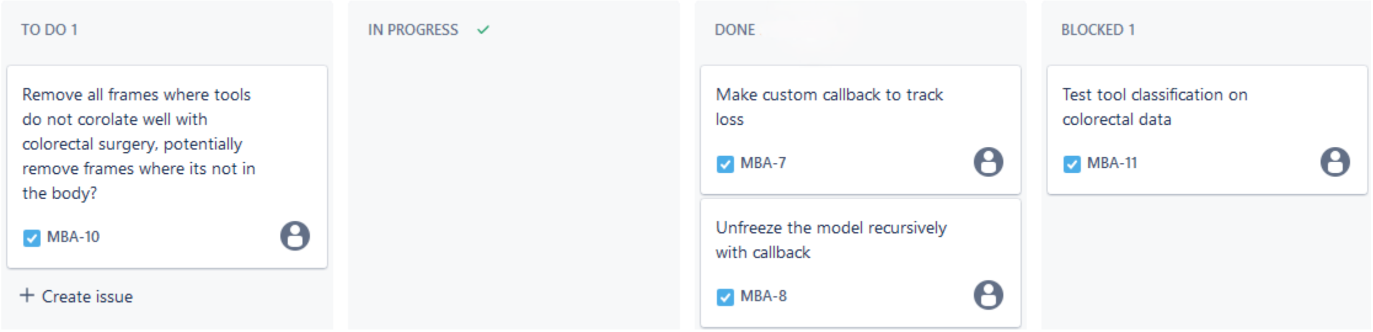
[19] Shinozuka, K., Turuda, S., Fujinaga, A., Nakanuma, H., Kawamura, M., Matsunobu, Y., Tanaka, Y., Kamiyama, T., Ebe, K., Endo, Y., et al.: Artificial intelligence software available for medical devices: surgical phase recognition in laparoscopic cholecystectomy. Surgical Endoscopy **36**(10), 7444–7452 (2022)

[20] Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M., Padoy, N.: Endonet: a deep architecture for recognition tasks on laparoscopic videos. IEEE transactions on medical imaging **36**(1), 86–97 (2016)

[21] Yengera, G., Mutter, D., Marescaux, J., Padoy, N.: Less is more: Surgical phase recognition with less annotations through self-supervised pre-training of cnn-lstm networks. arXiv preprint arXiv:1805.08569 (2018)

[22] Zhang, B., Ghanem, A., Simes, A., Choi, H., Yoo, A.: Surgical workflow recognition with 3dcnn for sleeve gastrectomy. International Journal of Computer Assisted Radiology and Surgery **16**(11), 2029–2036 (2021)

Appendices



TO DO 1

Remove all frames where tools do not corolate well with colorectal surgery, potentially remove frames where its not in the body?

☑ MBA-10

+ Create issue

IN PROGRESS ✓

DONE

Make custom callback to track loss

☑ MBA-7

Unfreeze the model recursively with callback

☑ MBA-8

BLOCKED 1

Test tool classification on colorectal data

☑ MBA-11

Fig. 1: Kanban Board tracking tasks

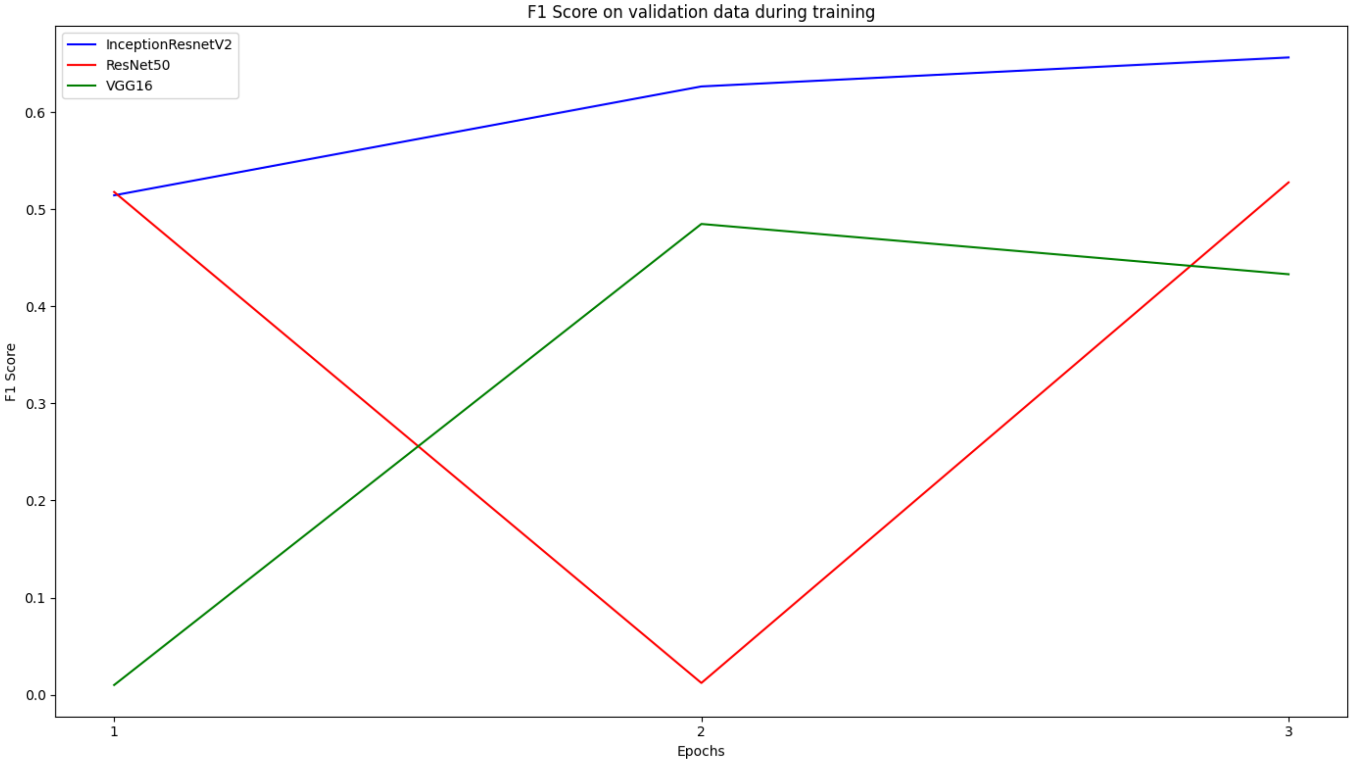


Fig. 2: Comparison of F1 Scores on validation data.

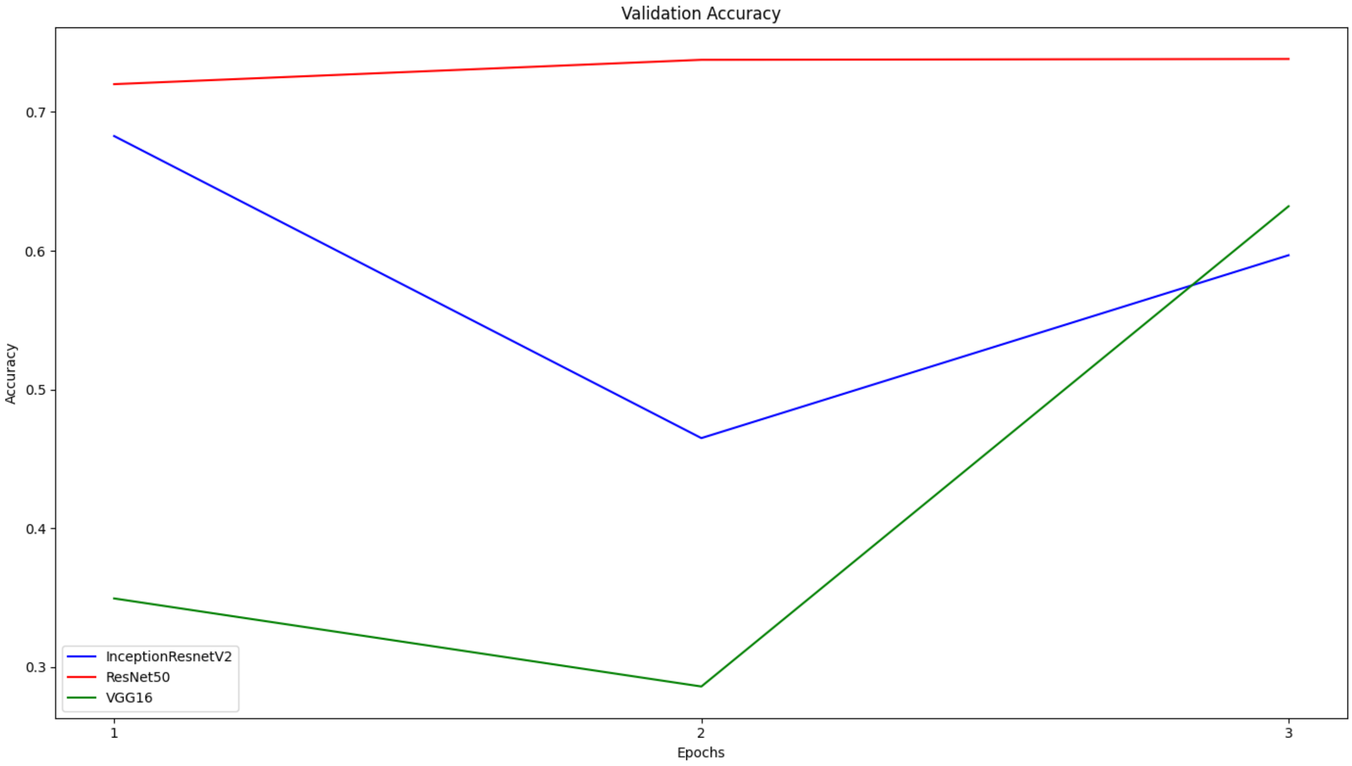


Fig. 3: Comparison of accuracy on validation data.

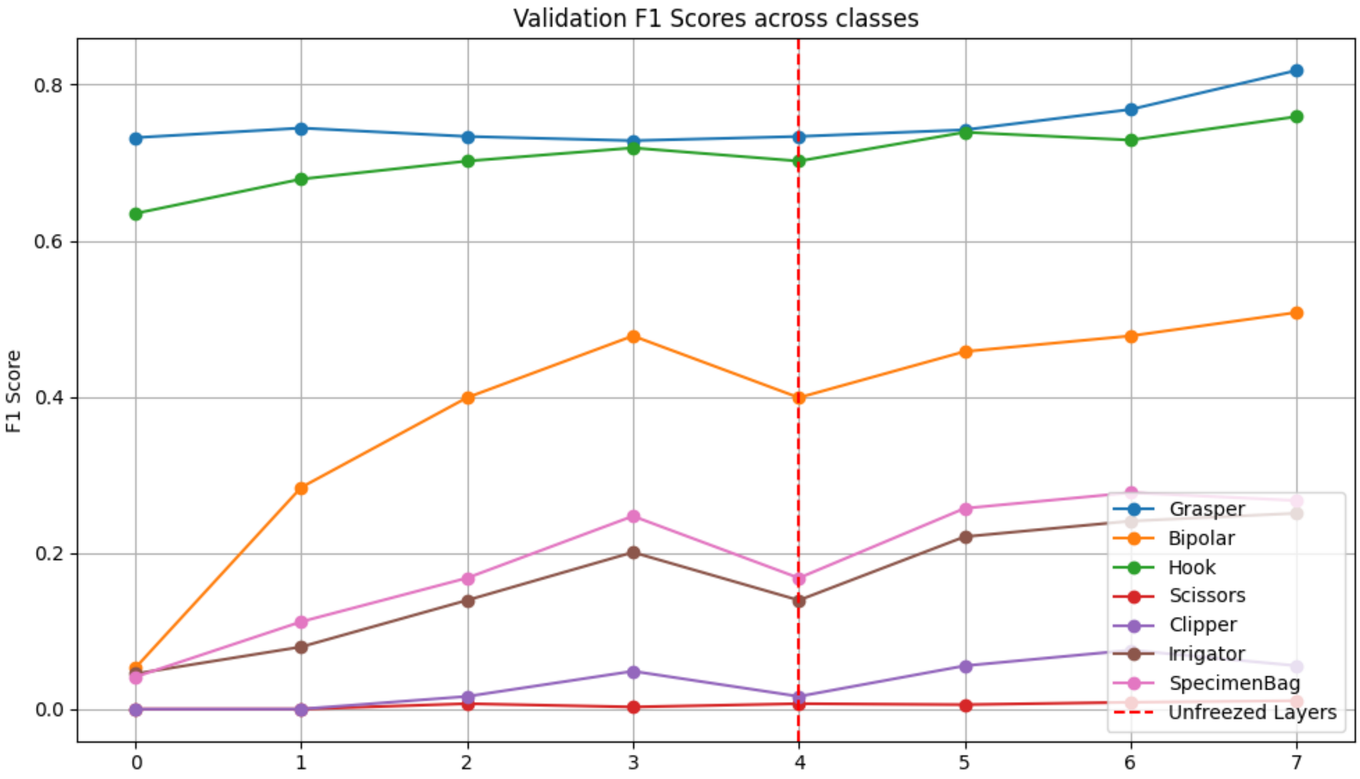


Fig. 4: F1 Score for each tool class

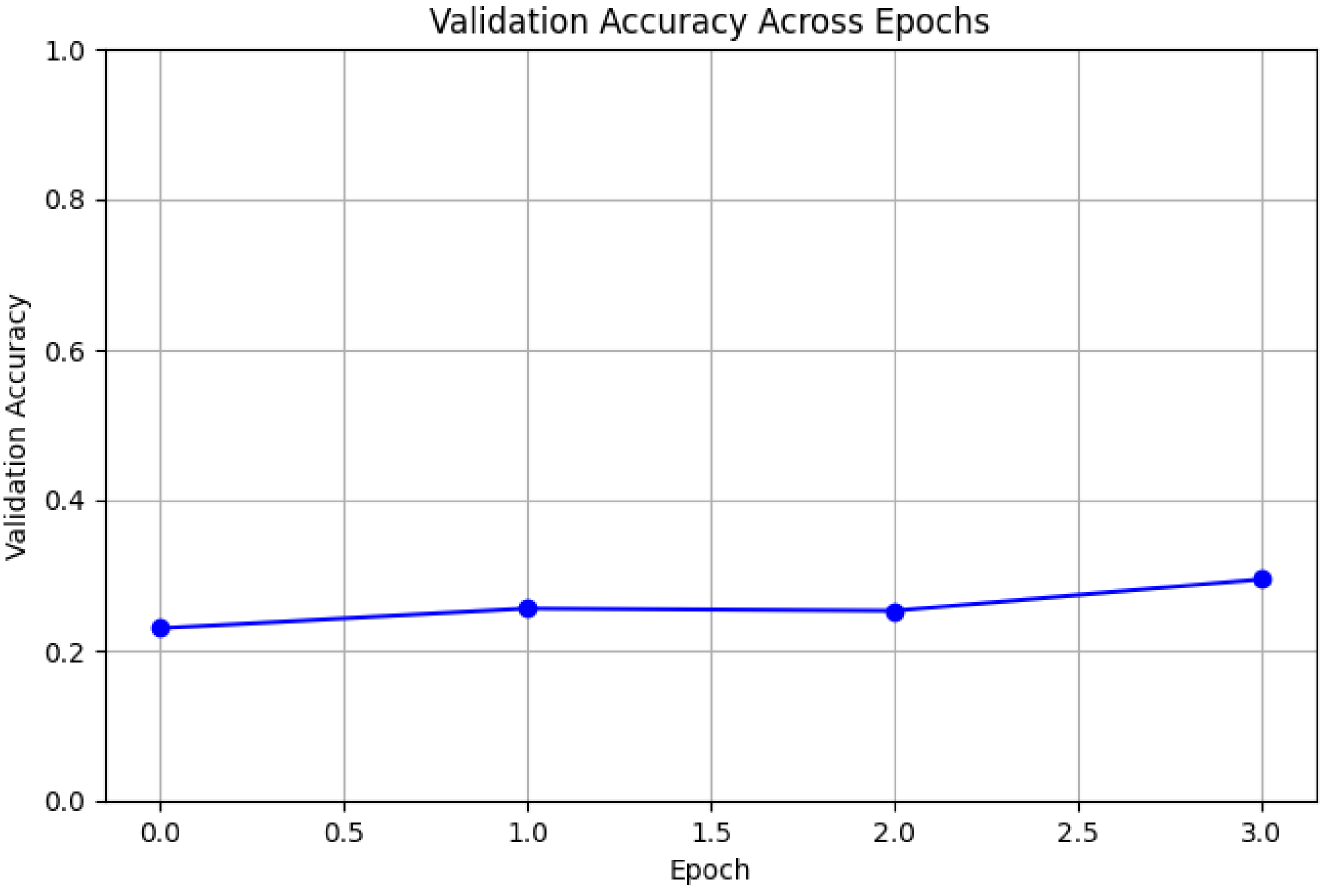


Fig. 5: Phase detection accuracy