

Automatic Open Suturing Skills Assessment with Model Fusion Techniques

Tiago Jesus^{1,2,3}[0000-0003-1437-5439], André Ferreira^{1,4,5}[0000-0002-9332-0091],
and Victor Alves^[0000-0003-1819-7051]¹

¹ Center Algoritmi / LASI, University of Minho, Braga, 4710-057, Portugal

² Life and Health Sciences Research Institute (ICVS), School of Medicine, University of Minho, Braga, Portugal

³ ICVS/3B's – PT Government Associate Laboratory, Braga, Guimarães, Portugal

⁴ Institute for AI in Medicine (IKIM), University Medicine Essen, Girardetstraße 2, Essen, 45131, Germany

⁵ Computer Algorithms for Medicine Laboratory, Graz, Austria
`{id8970}@alunos.uminho.pt`

Abstract. Use of YOLO for feature extraction and a multilayer perceptron network for classification of surgeons' skills in open suturing.

Keywords: Surgery · Skill Evaluation · YOLO · MLP · Classification

1 Introduction

For surgeons to improve their surgical abilities the existence of feedback (assessment) capable of precisely assessing the effect and accuracy of each surgery is required. Open surgery has a higher degree of freedom, which makes it even harder to evaluate when compared to minimally invasive surgery. Therefore, in this challenge machine-learning-based solutions are sought in order to make this evaluation automatic and objective, helping surgeons to improve their skills and consequently improve patient outcome [2].

Our solution uses YOLO [6] to efficiently extract the features from each video frame, and a multilayer perceptron (MLP) [5] network to classify these features into the correct label. A video is composed of several images per second, also known as frames per second (FPS). A video of 5 minutes with 30 FPS contains 9000 images, which makes video processing very slow and computationally heavy. Therefore, our solution aims to reduce the requirements needed to create this pipeline.

We developed our solution without taking into account the criteria for rating surgeons and without talking to any of the evaluators. We have tried to solve the problem by looking at the videos only, which makes our solution more general and applicable to other types of evaluation by video. As it uses a pre-trained network and a light MLP, this pipeline is very fast to reproduce.

2 Methods

All experiments were performed in a machine with GPU NVIDIA RTX A6000, 48GB of VRAM, AMD EPYC 7702P 64-Core CPU, and 256GB of RAM. The solution was developed using MONAI and PyTorch and is publicly available in Github.

2.1 Dataset

The data set consists of videos of medical and dental students and residents practising suturing in open surgery. The videos are recorded using a perspective *bird's-eye-view*, with a length of approximately 5 minutes. Each video is manually rated by three individual raters for both global rating score (GRS) and objective structured assessment of technical skill (OSATS). A total of 330 videos with the respective ground truth are available for training. 80% of the dataset is used for training and the remaining for validation [2].

Pre-processing: Each video was cropped to a duration of 5 minutes, and the FPS were reduced from 30 to 1 in order to process each video completely at once. We tested two distinct pre-trained YOLO networks for feature extraction. For the first, each image was resized to 640x640 using bilinear interpolation to fit the pre-trained YOLOv8 network and normalised with mean 0 and standard deviation 1 using the torchvision package. For the second, all pre-processing was the same but resized to 320x320.

2.2 Evaluation metrics

Two metrics are used to evaluate the performance of the solutions: F1-score (Dice Similarity Coefficient) [1] and Expected Cost (EC) [3].

2.3 Pipeline

Two distinct technologies are used to create the final classifications, also known as model fusion strategy [4] as shown in Figure 1:

YOLOv8: YOLO (You Only Look Once) is a very efficient and fast object detection algorithm composed of convolutions. It was originally developed to create bounding boxes and for classification, however, newer versions (such as version 8) can be used for image classification and instance segmentation as well. Due to its inference speed, this network is suitable for video processing, making feature extraction very fast, when compared with other convolutional networks available.

MLP: Multilayer perceptron is a feedforward artificial neural network consisting of fully connected neurons with a non-linear activation function. These networks are often used for classification and regression problems. Due to the

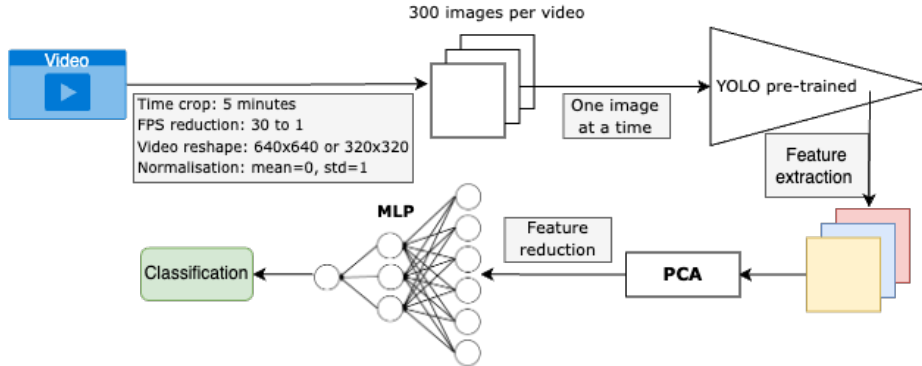


Fig. 1. Inference pipeline

complexity of the data and computational limitations, these networks are preferred over convolutional networks as they require fewer resources to perform well.

Principal component analysis (PCA) was used to reduce the number of features for the MLP network in order to reduce the computational burden for both training and inference.

Although we only show the solution for Task 1, the same solution could also be used for Task 2, as they have similar objectives, even though the second task is more complex.

3 Results and Discussion

Task 1: We tested two distinct loss functions to train each MLP: mean squared error (MSE), and cross-entropy (CE). Each loss function requires a distinct MLP: For the MSE, the MLP network (called MPL_{mse}) consists of an input layer with a size of 90000, a hidden layer with 9000 channels, an activation layer ReLU, and an output layer with the shape 1. When CE is used, the MLP network (called MPL_{ce}) the output layers changes to 4, and a Softmax is added after the last layer.

For all MLP training: the batch size was set to 10, and the shuffling was set to True. Adam was the optimiser used, with a learning rate of $1e-5$. The training was performed during 200 epochs. The best model was saved considering the best F1-score computed in evaluation set. Pre-processing, feature extraction and PCA took around 12 hours. The MLP architecture training took 1 hour. It takes around 20 seconds for one inference.

Experiment 01: The features of each frame of each video are extracted using a pre-trained YOLOv8⁶. Our pipeline for the feature extraction was based on

⁶ <https://github.com/ultralytics/assets/releases/download/v8.2.0/yolov8n.pt>

this repository⁷. These features are then used to train an MLP network, which has to classify the GRS for task 1.

The features have the shape (300x384x20x20) where 300 represents the 300 frames per video, 384 denotes the number of channels/filter, and 20x20 the height x width of the features. Therefore, the input of the MLP is $300 \times 384 \times 20 \times 20 = 46.080.000$. To reduce the number of features, PCA was used, reducing the number of features used in the MLP to 90000.

Experiment 02: Similarly to Experiment 01, a pre-trained YOLOv5 pre-trained YOLOv5⁸ is used to extract features from the last layer (before the prediction layer). PCA is used to reduce the number of features of each video to 90000 and these are used to train the MLP network. The MLP networks are the same as explained in Experiment 01. Both loss functions are also tested in this experiment.

Table 3 contains the F1-score and EC of each experiment. As can be seen, the experiment with best results is the Experiment 02 with CE as loss function. These results were expected as the YOLO network used to extract the features for this experiment was trained for hand tracking in sign language. Therefore, the features are related to tracking the position of the hand and the movements performed by the surgeon. We believe that a YOLO trained for joint recognition and hand position would provide even stronger results. Also, due to computational limitations, we were only able to use 1 FPS, losing 29 FPS of information.

Table 1. Results

Experiment name	Loss function	F1-score	EC
Experiment 01	MSE	0.410	0.267
	CE	0.294	0.306
Experiment 02	MSE	0.402	0.244
	CE	0.472	0.228

4 Authors Statement

T.J., A.F. and V.A. conceived the study. T.J. purified the dataset. T.J., A.F. and V.A. contributed to establishing and refining the prediction model. V.A. supervised the analysis. T.J. and A.F. wrote the manuscript. V.A. proofread the manuscript.

Acknowledgments. Tiago Jesus thanks the Fundação para a Ciência e Tecnologia (FCT) Portugal for the grant 2021.05068.BD. André Ferreira thanks the same institution for the grant 2022.11928.BD. This work was supported by FCT within the R&D Units Project Scope: UIDB/00319/2020.

⁷ <https://github.com/Alaawehbe12/Feature-extraction-YOLOv8>

⁸ https://github.com/SegwayWarrior/Gesture_Recognition_opencv_yolov5/tree/master/sign_lang_detection

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Dice similarity coefficient (dsc). <https://metrics-reloaded.dkfz.de/metric?id=dsc>, accessed: 15-09-2024
2. Endoscopic vision challenge 2024 (endovis24): Structured description of the challenge design. <https://opencas.dkfz.de/endovis/wp-content/uploads/2024/05/39-Endoscopic-Vision-Challenge-2024.pdf>, accessed: 15-09-2024
3. Expected cost. https://metrics-reloaded.dkfz.de/metric?id=expected_cost, accessed: 15-09-2024
4. Li, W., Peng, Y., Zhang, M., Ding, L., Hu, H., Shen, L.: Deep model fusion: A survey. arXiv preprint arXiv:2309.15698 (2023)
5. Popescu, M.C., Balas, V.E., Perescu-Popescu, L., Mastorakis, N.: Multilayer perceptron and neural networks. WSEAS Transactions on Circuits and Systems **8**(7), 579–588 (2009)
6. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)