Tool detection and classification in cataract surgery using weakly supervised convolutional neural networks

Adrian Galdran*†, Pedro Costa* and Aurélio Campilho*
*INESC TEC, Porto
†Corresponding Author: adrian.galdran@inesctec.pt

I. Introduction

In this work we approach the problem of surgical tool detection in two steps. First we propose a Multiple-Instance-Learning technique for the task of tool detection understood as the binary problem of deciding whether a frame contains a tool or not. This first model M_1 is trained in an independent way, and afterwards a second model M_2 is trained to classify the kind of tool present in each of the frames for which M_1 declares the presence of a tool.

Since the main novelty lies in the MIL based technique behind M_1, we describe it in more detail. The second model M_2 is just a Resnet34 deep Convolutional Neural Network [1] trained for the 21-class classification problem.

II. MIL-based Surgical Tool Detection

We start by formalizing M_1 as a graphical model view and then we show how to apply it to the tool detection problem.

A. Graphical Model

In this work, we propose a method to train an instance classifier with bag labels only (*i.e.* training a patch classifier using only image labels). For that, the instances' labels y_i are treated as latent variables which are then combined by a pooling function f parameterized by w to infer the bag's label $Y = f(y_1, ..., y_N; w)$. Therefore, it is important to carefully design the pooling function in order to properly encode the relationship between the instances and the bag labels.

The graphical model depiction of our approach is shown in Figure 1. We define X as a random variable representing the set of input instances, θ as the parameters of the instance classifier and $P(y_i|X_i, \theta)$ as a Bernoulli distribution although, in principle, it could follow any other distribution such as a categorical distribution. The choice of this distribution is tied with the problem to solve and influences the design of the subsequent pooling function f. We focus on problems where each input instance has a binary label. Moreover, Y and (X, θ) are conditionally independent given y, meaning that the label of the bag is completely determined by the labels of the instances and w. This allows us to write the likelihood of Y as $P(Y|y_1, ..., y_N; w)$. The goal is, then, to find the parameters (θ, w) that maximize the likelihood:

$$\theta, w = \arg\max_{\theta, w} P(Y|y_1, ..., y_N; w) \tag{1}$$

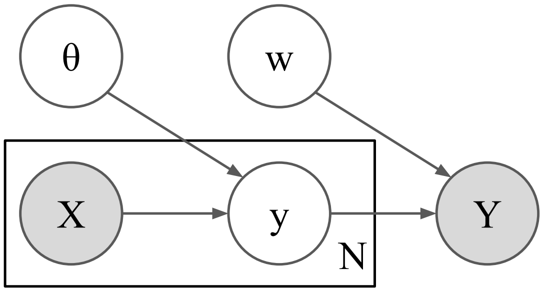


Fig. 1. **Proposed graphical model.** Only the N input instances X and the bag label Y are observed. Each instance has an associated label y that depends on parameters θ. The instance's labels y are combined by means of a pooling function parameterized by w to produce Y.

There are some design choices that need to be considered:

Choice #1 How to define the input instances X? We could use wavelets, patches or any other feature extraction method.

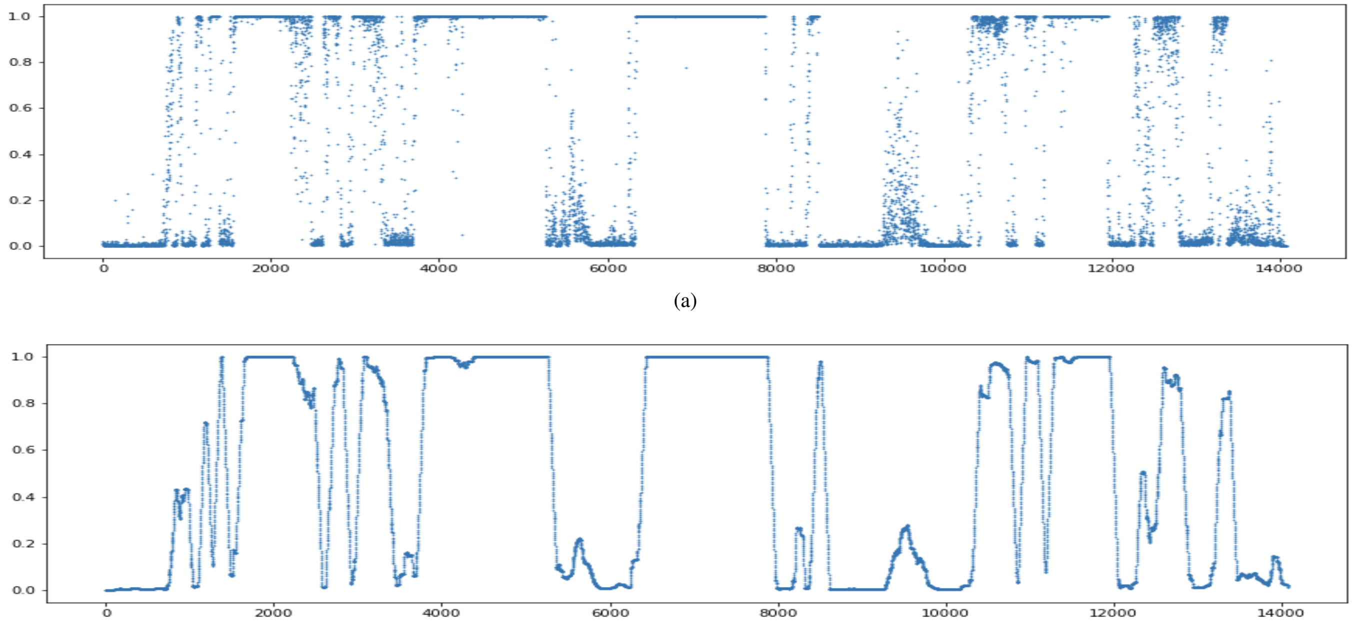
Choice #2 What learning algorithm should be used to model $P(y_i|x_i, \theta)$? Some choices include SVM and logistic regression.

Choice #3 What pooling function f should be used? For instance, Sum Pooling, Max Pooling or Average Pooling could be used.

Since these choices depend on the problem to solve, we decided to test this model on the problem of tool detection in cataract surgery. For that we chose to *#1* use patches of the input eye fundus image as instances X; *#2* use a Convolutional Neural Network (CNN) to perform the patch classification; and *#3* use the max as the pooling function f.

B. Instance Feature Learning and Classification

Choices *#1* and *#2* are related. CNNs have been used with great success for image classification problems. These models are able to extract features from raw data often achieving superior results compared to feature engineering approaches. However, as CNNs can easily learn irrelevant features, they usually require large amounts of data to avoid overfitting. To minimize this issue, some works use CNNs to classify patches of the images making it impossible for the network to correlate two pixels that are far away from each other in the image.



(a)

(b)

Fig. 2. (a) Raw tool presence predictions in one of the test videos (b) Tool presence predictions after applying a rolling trimmed mean filtering step.

Moreover, it has been shown that Convolutional Neural Networks pre-trained on a larger dataset and then finetuned on another smaller dataset yields better results than training the same network on the smaller dataset alone. In this work, we used the Inception V3 network trained on the Imagenet dataset .

Instead of extracting patches from the image and using the same CNN model on different patches, we use the Inception network to perform patch classification given the full input image. However, this network was trained to perform image classification and not patch classification. In order to overcome this issue, we discarded the deeper layers of the Inception V3 network, as the receptive field of each layer grows as the network gets deeper. For instance, the receptive field of a 3×3 Convolution Layer is, indeed, 3×3, while two 3×3 Convolution Layers have a 5×5 receptive field. Therefore, by discarding deeper layers of the network, we are reducing the receptive field of the output layer.

More concretely, we kept the pre-trained layers until the mixed1 layer and added two 1×1 Convolutional layers, one with 1024 units followed by a LeakyReLU activation function and another one with a single unit followed by the sigmoid activation function to obtain the label of each patch y_i.

After computing the patch labels, we need to combine them to produce the image label. We build on the realization that this problem follows the *Standard MIL Assumption*, which states that negative examples only contain negative instances while positive examples contain at least one positive instance. In this application this means that there will be at least one patch where a tool is visible in a positive image whereas, in an image without a tool, no patch will contain a visible tool. Problems that follow the *Standard MIL Assumption* can be modeled with the max-pooling function [2], [3].

III. CNN-based Surgical Tool Classification

The output of the above model was a frame-wise prediction in $[0, 1]$ with an estimate of the probability of each frame containing a tool. We regularized these probabilities by means of a rolling trimmed mean with a window size of 100 frames, see Fig. 2.

Afterwards, the training set provided by the organization was reduced to those frames that contained a tool and a resnet34 deep CNN was trained to make a decision about which tool was present inside a given frame. This approach is standard in the image classification literature and won't be described in detail, the reader can refer to [1]. The training dynamics consisted on a random train/validation split of the availabe data in a proportion of $80\% - -20\%$, after selecting only odd frames for training. We resized the input frames to 128×128 resolution, and training until we detect overfitting. At this point, we increased the resolution of the images to 256×256, and trained for 7 extra epochs.

The resulting model was applied to generate tool predictions on every frame of the test videos, and the frame-wise product of these predictions with the result of the tool presence detector described in the previous section was computed.

IV. Relationship to previous submissions

In this case, we sub-sampled the training by selecting odd frames instead of randomly selecting 25% of the available data, which may leave some of the minority tool classes outside the training set. The trimmed mean filter in Fig. 2 was also tried in an attempt to better regularize the tool presence predictions generated by the Multiple-Instance Learning model described in section II.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[2] S. Andrews, I. Tsochantaridis, and T. Hofmann, “Support vector machines for multiple-instance learning,” *Advances in neural information processing systems*, pp. 577–584, 2003.

[3] P. Costa and A. Campilho, “Convolutional bag of words for diabetic retinopathy detection from eye fundus images,” *IPSJ Transactions on Computer Vision and Applications*, vol. 9, no. 1, p. 10, 2017.