Monitoring Tool Usage in Cataract Surgery Videos using NASNet and Boosted Recurrent Neural Networks

Hassan Al Hajj, Mathieu Lamard, Pierre-Henri Conze, Béatrice Cochener, Gwenolé Quellec

***Abstract*— We recently proposed an algorithm for jointly boosting an ensemble of CNNs and an ensemble of RNNs for monitoring tool usage in cataract surgery videos [1]. An average area under the ROC curve of $A_z = 0.9717$ was achieved. However, recent experiments revealed that the CNN models on which our algorithm operated were suboptimal. Therefore, we decided to replace them, as described in this report. The RNN model was retrained accordingly.**

I. CNN Model

Although state-of-the-art CNN architectures were used in our previous study [1], including Inception-V4 and Inception-ResNet-v2 [2], two key elements had not been investigated: transfer learning [3] and batch normalization [4]. Moreover, a promising CNN architecture, namely NASNet [5], was released recently. Therefore, we decided to revise the CNN part of our architecture. Due to the time constraints, a single CNN architecture was used: NASNet, with batch normalization. Precisely, the "NASNet-A Large" model was used: this network processes images of 331×331 pixels and achieved a top-1 accuracy of 82.7% on the ImageNet dataset. The "NASNet-A Large" model, pretrained on ImageNet, was fine-tuned on the CATARACTS training set. For data augmentation purposes, images were randomly rotated, shifted and scaled at each epoch [1]. The root mean square propagation optimizer was used, with mini-batches of 6 images.

II. RNN Model

Once NASNet was trained, an ensemble of RNNs was boosted on top of it, as described previously [1]. Our boosting algorithm selected three RNN architectures, in the following order:

1) LSTM [6] with one cell of 128 units,
2) LSTM with two cells of 256 units,
3) LSTM with one cell of 256 units.

For data augmentation purposes, 30 uniformly subsampled versions of each training video were used to train the RNNs. For compatibility with the trained models, each test video had to be processed similarly: as a result, 30 sets of 21 prediction signals (one signal per tool) were obtained per test video. Prediction signals from the 30 sets were then interleaved to construct a single set of prediction signals for the test video as a whole [1]. Finally, to remove discontinuities caused by this interleaving process, the resulting prediction signals were smoothed by a median filter.

III. Implementation Details

The NASNet implementation and the pre-trained NASNet model originate from the *TensorFlow-Slim image classification model library* [1]. Keras was used for training the RNNs. The same cost function was used for training the CNN and RNN models, namely the sigmoid cross-entropy.

Two videos were used for validation: 'train13' and 'train23'. These videos were used:

- for early stopping CNN or RNN training (performed on the remaining 23 videos),
- to select the video subsampling rate and the window size of the median filter.

All videos in the CATARACTS training set were used to compute the boosting weights.

IV. Future Works

Our experiments on joint CNN+RNN boosting, reported in [1], will be replicated in full using multiple competitive CNNs, including NASNet.

References

[1] H. Al Hajj, M. Lamard, P.-H. Conze, B. Cochener, and G. Quellec, "Monitoring tool usage in cataract surgery videos using boosted convolutional and recurrent neural networks," *arXiv:1710.01559 [cs]*, Oct. 2017.

[2] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the impact of residual connections on learning," in *Proc AAAI*, San Francisco, CA, USA, Feb. 2017, pp. 4278–4284.

[3] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" *arXiv:1411.1792 [cs]*, Nov. 2014.

[4] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv:1502.03167 [cs]*, Feb. 2015.

[5] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," *arXiv:1707.07012 [cs, stat]*, July 2017.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

All authors are with Inserm, UMR 1101, Brest F-29200, France gwenole.quellec@inserm.fr

M. Lamard and B. Cochener are with Univ Bretagne Occidentale, Brest, F-29200 France

P.-H. Conze is with IMT Atlantique, LaTIM UMR 1101, UBL, Brest F-29200 France

B. Cochener is with Service d'Ophtalmologie, CHRU Brest, Brest, F-29200 France

[1] https://github.com/tensorflow/models/tree/master/research/slim