ResNet

Isabel Amaya-Rodriguez, Javier Civit-Masot, Francisco Luna-Perejon, Lourdes Duran-Lopez, Alexander Rakhlin, Sergey Nikolenko, Satoshi Kondo, Pablo Laiz, Jordi Vitrià, Santi Seguí, and Patrick Brandao

General Motivation

In this chapter, all groups have used Residual Network (ResNet) (He et al. 2016) as part of different architectures with the purpose of solving the GIANA challenge. In some cases like RTC-ATC group ResNet-50 was used as a layer in Faster Convolutional Neural Network (FCNN) in order to build an automated recognition system to detect the presence of polyps in colonoscopy images.

The main reason to use this network is because ResNet models try to solve the overload of the accuracy which comes from network depth. The accuracy saturation is not due to overfitting or the quantity of layers is because of the named Vanishing Gradient (Hochreiter 1998) this effect try to explain when the network is deep the loss functions in gradients value are near to zero after several chain rule applications. Then weights are not updated and consequently no learning is being performed. To

I. Amaya-Rodriguez (✉)
Vicomtech, Paseo Mikeletegi 57, Donostia-San Sebastian, Spain
e-mail: iamaya@vicomtech.com

J. Civit-Masot · F. Luna-Perejon · L. Duran-Lopez
Robotics and Computer Technology Lab., University of Seville, Seville, Spain

A. Rakhlin · S. Nikolenko
Steklov Institute of Mathematics at St. Petersburg, Russia nab. r. Fontanki, 27, St. Petersburg 191023, Russia

S. Kondo
Konica Minolta, Inc., 1-2 Sakura-machi, Takatsuki, Osaka 569-8503, Japan

P. Laiz · J. Vitrià · S. Seguí
Departament de Matemàtiques i Informàtica de la Universitat de Barcelona, Barcelona, Spain

P. Brandao
Wellcome/EPSRC Centre for Interventional & Surgical Sciences, University College London, London, UK

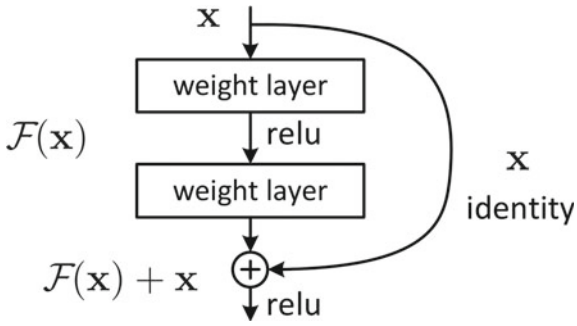

Fig. 1 Residual learning block

solve this problem, Microsoft created a new deep learning concept based on residual leaning which allows gradients to flow between layers.

Introduction to ResNet Architecture

In this section, the basic concepts of ResNet architecture is explained. As it was introduced in the Sect. 12.1 ResNet architecture makes it possible to implement hundreds or even thousands of layers and still achieves compelling performance. Residual Network works substracting features learned from input of that layer.

The main characteristic introduced by ResNet is the identity shortcut connection defined as $F(x) := H(x) - x$ shown in Fig. 1. This shortcut connections X are identity mappings and their outputs are added to the following stacked layers. Then ResNet apply simply stacked identity mappings and the residual of X in H(x) is learned. It solves problems like training error increase when the depth increases too.

In the case of ResNet-12 contains five Residual Blocks as shows Fig. 2. For each two-convolutional layer there is one identity shorcut connection.

Methodologies

RTC-ATC Group

In this section, RTC-ATC group shows an application of Faster R-CNNs (FRCNN) in order to build an automated recognition system to detect the presence of polyps in colonoscopy images presented in GIANA challenge 2019. To realize this goal, they used an implementation of FRCNN with ResNet-50 as Fully Convolutional Network (FCN) architecture. FRCNN builds on the idea of Region Proposals by sharing intermediate features with the classification network. For example, the ResNet takes an

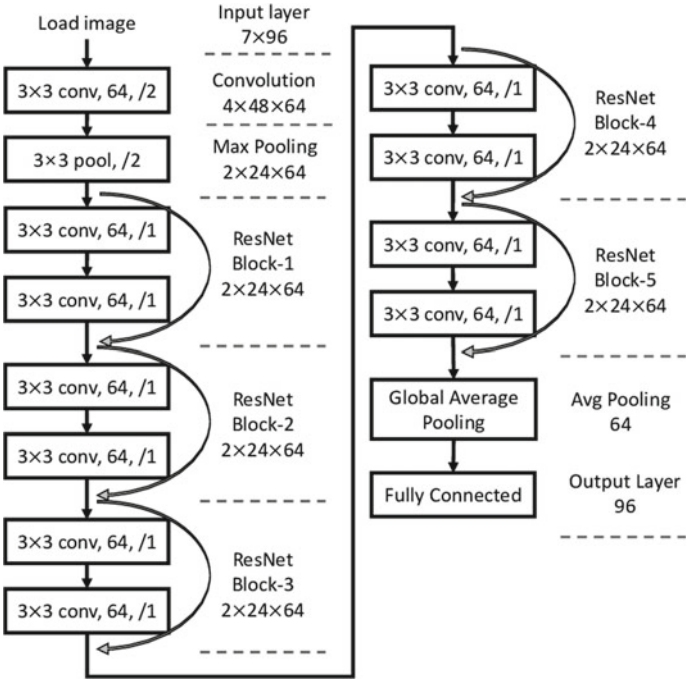

Fig. 2 ResNet-12 architecture

input image and produces a series of transformations before arriving at the prediction. The FRCNN will use the intermediate features of ResNet to aid in region proposal.

Brief Methodology Introduction

Faster R-CNNs have been used for different purposes: face detection (Jiang and Learned-Miller 2017), driver's cell-phone usage and hands on steering wheel detection (Hoang Ngan Le et al. 2016) are some application examples of this algorithm, which has proven to show good results. As it was mentioned in the introduction, a FRCNN was used in this work for the polyp detection task. This algorithm is divided into two modules (Fig. 3):

1. First of all, a deep Fully Convolutional Network (FCN) (Ren et al. 2015) receives the images from the dataset. Then, it extracts feature maps or descriptive characteristics and analyzes them to propose regions of interest. The novel step that this architecture introduced is the way to determine the regions of interest. Region Proposal Network (RPN) is computed base on the output feature map of the previous step. Then, RPN is connected to a convolutional layer with 3×3 filters, 1 padding, 512 output channels. The output is connected to two 1×1 convo-

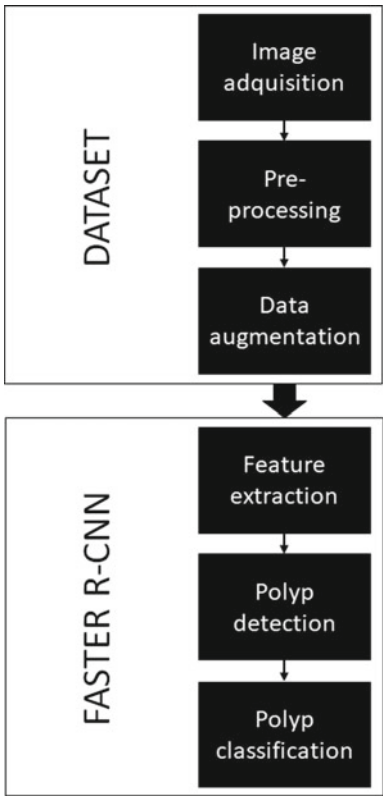

Fig. 3 Block diagram of the implemented approach

lutional layer for classification and box-regression (Note that the classification here is to determine if the box is an object or not).

2. Next, as shows Fig. 4 ROI pooling layer is used for these proposed regions in order to ensure the standard and pre-defined output size. These valid outputs are passed to a fully connected layer as inputs. In our case, by using a neural network that takes advantage of the mathematical operations made in the convolutional layers. In our architecture we have used the ResNet-50 model (He et al. 2016) as FCN. ResNet models try to solve the saturation of the accuracy caused by increasing the network depth (Fig. 5).
3. Finally, the proposed regions that are the input of the second module, called Fast R-CNN detector, composed of two fully connected layers, a regression layer and a classification layer (Ren et al. 2015).

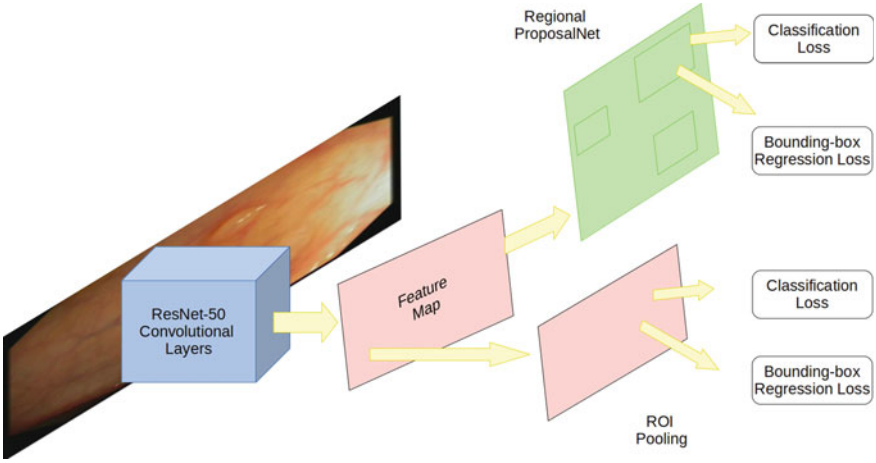

Fig. 4 ResNet-50 in faster convolutional neural network

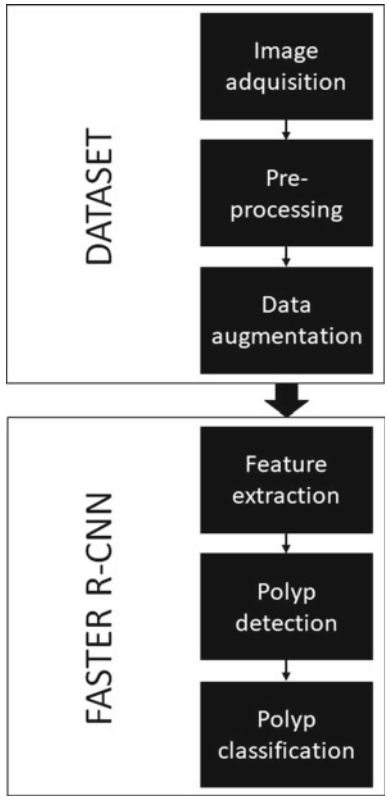

Fig. 5 Block diagram of the implemented approach

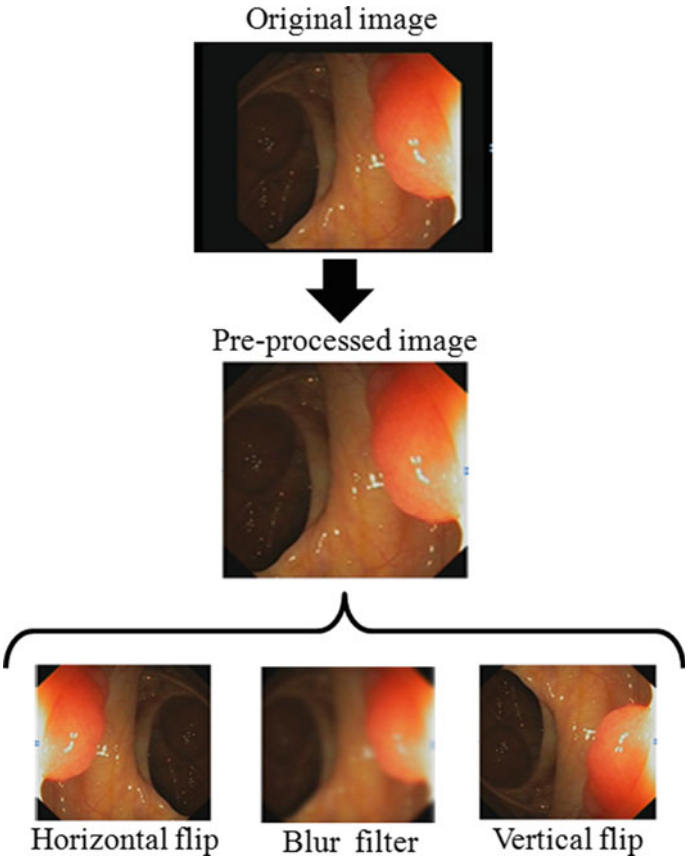

Fig. 6 Processing applied to the original images. First, black edges are removed in a pre-processing step. Then data augmentation is applied, generating three different new images

Architecture and Parameters Tuning

Basic architecture was modified to achieve the better results, aiming to observe the benefit of using different datasets by training the network with background examples and augmented dataset. Firstly, with the aim of reducing the number of false positives, a technique called hard-negative mining was used (Felzenszwalb et al. 2009). It consists of adding negative samples, which means, including examples of images that do not contain polyps in the training step, labeling them as background. The dataset was augmented using a series of transformations so that the model would never train twice the exact same image. For each original preprocessed image, an horizontal flip, a vertical flip, and a blur filter have been applied. Thus, we obtain three new images from each original sample. After this data augmentation step, we obtain a dataset that consists of 47.816 images in total (Fig. 6).

Our model contains several parameters to be defined in order to improve the results training with some invariant parameters as learning rate 10^{-5}, 1000 iterations per epoch, 32 number of Regional Object Interest (ROIs) and the increased image dataset with rotations and flips. Tests were performed every 50 epochs, selecting different confidence thresholds in order to obtain the best results.

Neuromation

In this section, the Neuromation team discusses their model architecture and segmentation uncertainty estimation based on Bayesian approximation.

Network Architecture. The model architecture stems from the Hourglass and U-Net design principles (Ronneberger et al. 2015; Liu et al. 2017). The contracting branch of the model is based on the Resnet-34 encoder where we introduce useful modifications: ELU activations instead of ReLU, reversed order of batch normalization and activation layers (Mishkin et al. 2017), and He normal weight initialization (He et al. 2015). One major difference from the classical U-Net architecture is meant to deal with the limited dataset size characteristic for the GIANA challenge and for medical imaging problems in general. We use two approaches to alleviate the problem of overfitting to limited training data: (1) extreme data augmentation and (2) Spatial 2D Dropout (Tompson et al. 2015) incorporated into the upsampling branch. The upsampling branch is implemented as a Feature Pyramid Network (FPN) (Lin et al. 2016), reconstructing high-level semantic feature maps at 4 scales simultaneously. We implement a Feature Pyramid block as a convolutional layer with 64 activation maps followed by upsampling to the original resolution with upsampling rate of 8, 4, 2, or 1 depending on the feature map depth (see Fig. 7). We concatenate upsampled maps into a single layer of $64 \times 4 = 256$ maps and finalize it with the Spatial 2D Dropout layer. Spatial 2D Dropout acts like a regularizer and prevents co-adaptation of the network weights, but unlike conventional dropout it drops out not individual neurons but entire activation maps. In all experiments, we use dropout rate 0.5, i.e., drop 128 out of 256 activation maps.

Finally, the output of the model is a sigmoid layer that assigns to every pixel a continuous probability from 0 to 1 of being a polyp region.

Loss functions. It is known that the categorical cross entropy (CCE), while convenient for training, does not directly translate into the metric of interest, Jaccard index (Rakhlin et al. 2018; Iglovikov et al. 2017; Rakhlin et al. 2019). Hence, as the loss function we use

$$L(w) = (1 - \alpha)CCE(w) - \alpha J(w), \tag{1}$$

a weighted sum of CCE and the soft Jaccard loss

$$J = \frac{1}{P} \sum_{p=1}^{P} \left(\frac{y_p \hat{y}_p}{y_p + \hat{y}_p - y_p \hat{y}_p} \right), \tag{2}$$

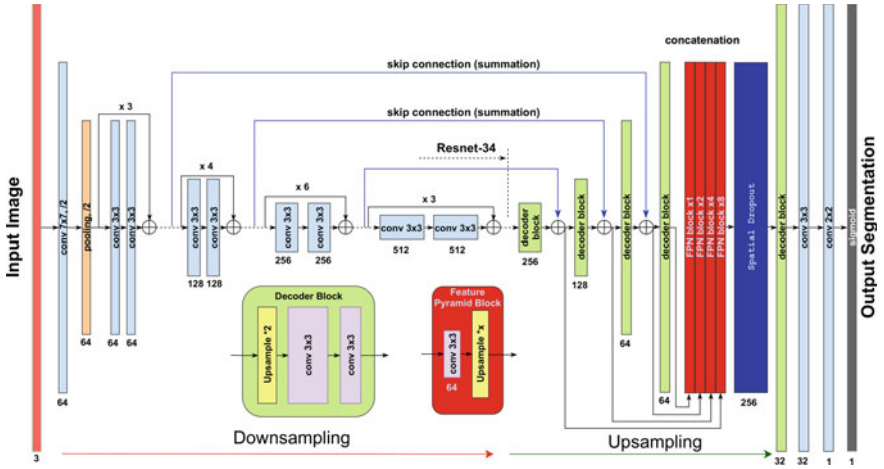

Fig. 7 Neuromation architecture

where y_p is the binary label for pixel p, $\hat{y}_p$ is the predicted probability for p, and P is the number of pixels in the image.

Segmentation uncertainty estimation. In the domain of medical imaging, it is particularly important to tell whether a model is confident about its estimate or not. One distinctive feature of our approach is an innovative application of dropout as a Bayesian approximation, as recently proposed by Gal and Ghahramani (2016), http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html.

Classical deep learning tools do not capture model uncertainty, returning only a point estimate at the output. Using softmax to get probabilities is actually insufficient to obtain model uncertainty (http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html). Bayesian models, on the other hand, offer a framework suitable to reason about model uncertainty, but usually do it with a prohibitive computational cost. Gal et al. show that dropout neural networks are identical—under certain, not too restrictive, assumptions—to variational inference in Gaussian processes. In particular, they demonstrate "that averaging forward passes through the dropout network is equivalent to Monte Carlo integration over a Gaussian process posterior approximation" (http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html).

Traditionally, dropout is considered as model averaging, and it was originally explained that scaling the weights at test time without dropout gives a reasonable approximation to the "average" model (Srivastava et al. 2014). However, for convolutional networks this approximation is not sufficient and can be improved considerably (Gal and Ghahramani 2016).

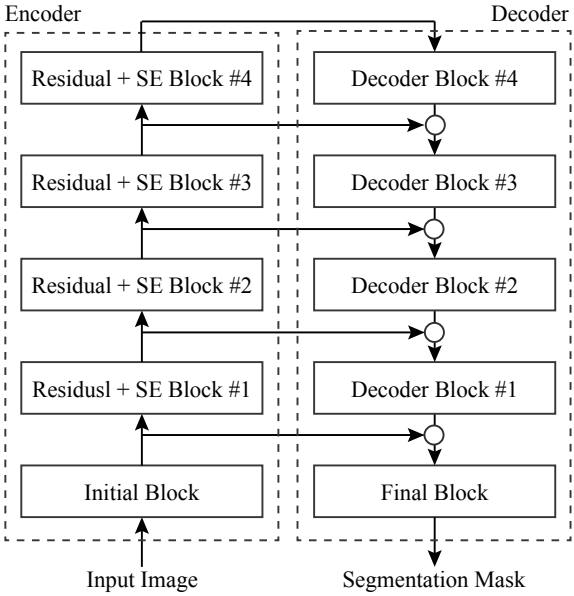

Fig. 8 Basic network architecture

Figure 8 shows the basic structure of our network. We use U-Net (Ronneberger et al. 2015) or Link Net (Chaurasia and Culurciello 2017) type deep neural networks with different encoders from the original U-Net and Link Net. U-Net and Link Net both have an encoder-decoder structure and intermediate feature maps in the encoder are concatenated or summed to intermediate feature maps in the decoder, respectively.

Modification of Base Architecture

Polyp detection, localization and segmentation tasks Our encoder is based on 101 layer ResNeXt (Xie et al. 2017) with Squeeze-and-Excitation blocks (Hu et al. 2018). The decoder is almost same as the original U-Net and Link Net networks except the number of feature maps. We use Link Net type for the polyp detection task, and use U-Net type network for the polyp localization and segmetation tasks.

WCE detection and localization tasks Figure 9 shows the whole structure of our network for the WCE detection and localization tasks. We use Link Net type for the WCE detection and localization tasks. Our encoder is based on 101 layer ResNeXt (Xie et al. 2017) with Squeeze-and-Excitation blocks (Hu et al. 2018). The decoder is almost same as the original U-Net and Link Net networks except the number of feature maps. We add two fully connected layers on top of the last residual block of the encoder ("Residual + SE Block #4" in Fig. 9) and obtain the

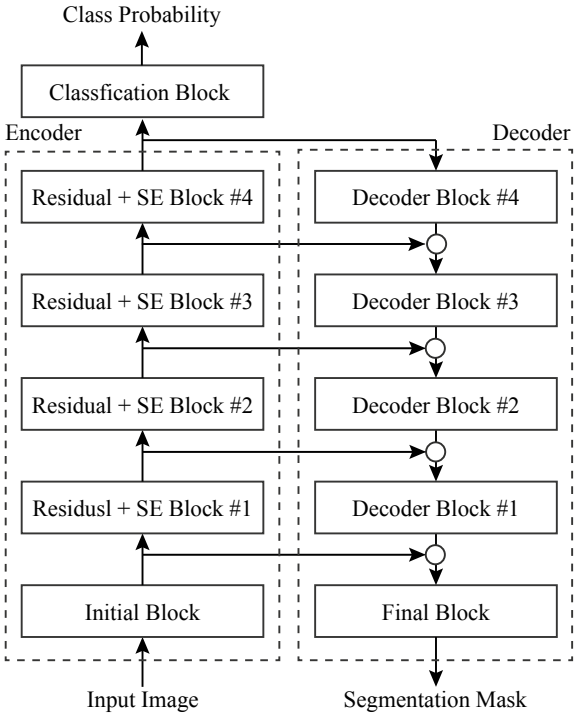

Fig. 9 Network architecture for WCE detection and localization task

classification results. We also obtain the lesion area (segmentation mask) as output of the decoder. The network is trained on the tasks of classification and segmentation simultaneously. The locations of the lesions are obtained by post-processing the segmentation results as described later.

Parameter Tuning to Solve the Task

Polyp detection, localization and segmentation tasks The training procedure is as follows. An input image is resized to 320×320 pixels after the border area is cropped. We use stochastic gradient descent for the optimization. The hyper-parameters in the optimization are that the initial learning rate is 0.1 and the momentum is 0.9. We decay the learning rate with a cosine annealing for each epoch. The mini-batch size is 32 and we run 200 epochs. The loss function is summation of softmax cross entropy loss and dice loss (Milletari et al. 2016). The softmax cross entropy loss is weighted depending on the distance from the contour of the polyp area (Anas et al. 2017). Data augmentation is applied on the fly during the training. We augment using translation, rotation, resizing, flipping, and contrast. We also use mixup (Zhang et al. 2017).

At inference, the final probability map is resized to the original size and thresholded. When probabilities of any pixels are greater than the threshold, we decide there are polyps. Otherwise, we decide there are no polyps.

In the polyp detection task, the threshold value is 0.2 which is decided by using the validation dataset.

In the polyp localization task, the threshold value is 0.4 which is decided by using the validation dataset. When we decide there are polyps, we find the largest area and we use the center of gravity of the largest area as the location of the polyp.

In the polyp segmentation task, the threshold value is 0.3 which is decided by using the validation dataset.

WCE detection and localization tasks The training procedure is as follows. An input image is resized to 320 $\times$ 320 pixels after the border area is cropped. We use stochastic gradient descent for the optimization. The hyper-parameters in the optimization are that the initial learning rate is 0.1 and the momentum is 0.9. We decay the learning rate with a cosine annealing for each epoch. The mini-batch size is 64 and we run 400 epochs. The loss function is summation of the classification loss and the segmentation loss. The classification loss is softmax cross entropy, and the segmentation loss is the summation of pixel-wise softmax cross entropy loss and dice loss (Milletari et al. 2016). Data augmentation is applied on the fly during the training. We augment using translation, rotation, resizing, flipping, and contrast adaptations.

At inference, the classification results are obtained from the output of the fully connected layer on top of the encoder. When the classification result is vascular or inflammatory, we identify the locations of the lesions by using the segmentation result. The segmentation result is obtained from the output of the decoder as a probability map. The probability map is resized to the original image size. Candidates of lesions are regions where the probability is greater than a threshold. If the region size is greater than another threshold, we identify the region as a lesion. The centroids of the detected lesion regions are used as localization results. The threshold values for the probability map and the region size are 0.7 and 50, respectively, which are chosen based on the results on the validation dataset.

Examples of Results (on the Training Sets)

RTC-ATC

Different experiments were carried out to determine if polyps were detected correctly or not. Tests were performed every 50 epochs, selecting different confidence thresholds in order to obtain the best results. Polyps detection performance is reported in Table 1. The results show the robustness of the proposed Faster R-CNN architecture on detecting the polyp position in colonoscopy images with a precision of

Table 1 Polyps detection performance. TP: True Positive, FP: False Positive, TN: True Negative, FN: False Negative

TP	FP	TN	FN	Precision	Recall	Accuracy	Specificity	F1	F2
3533	866	1659	1154	80.31	75.37	71.99	65.70	77.76	76.30

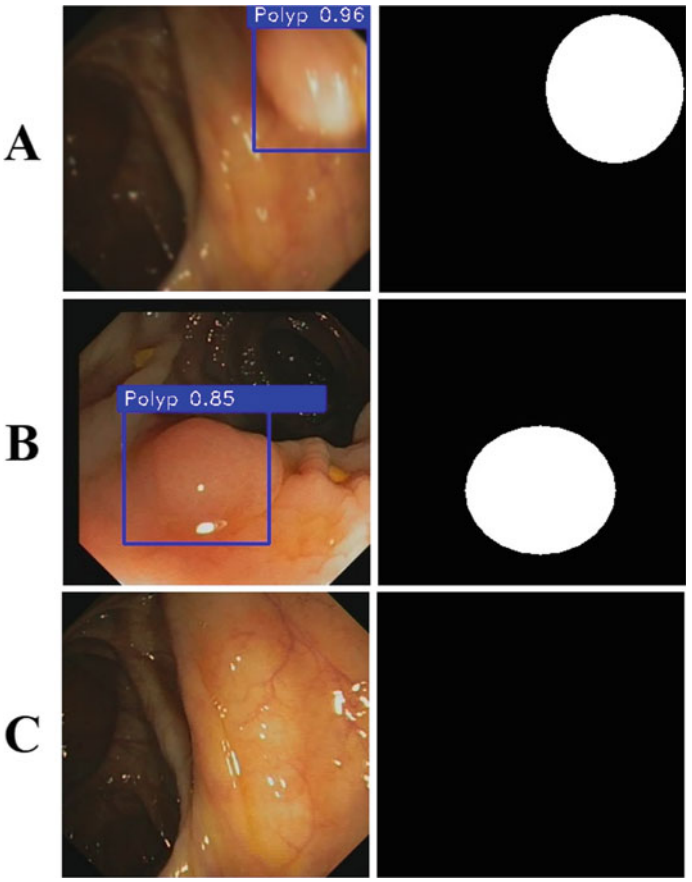

Fig. 10 RTC-ATC Polyp detection results task. Left: polyps detected by Faster R-CNN. Confidence values are represented in blue. Right: their corresponding ground truth. A and B show the performance in case a polyp appears, while C shows the performance in case there is no polyp

80.31%, a recall of 75.37%, an accuracy of 71.99% and a specificity of 65.70%. The minimal threshold was established at 0.80.

In Fig. 10 the results of our recognition system can be seen by showing the precision when detecting polyps inside samples from the dataset, and their corresponding mask images (as a ground truth) indicating where the polyps are located.

Neuromation

A direct application of Gal and Ghahramani (2016), http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html theory gives us tools to model uncertainty out of deep learning networks at almost zero additional cost. To this end, at test time we do not scale the weights, as it would be in the case of classical dropout. Instead, the model keeps dropping out random activation maps, producing multiple predictions for the same input. This output distribution provides more accurate point estimate and makes possible to assess the uncertainty of polyp segmentation.

Figure 11 shows sample segmentation results of our model on validation set samples that we set aside from the training set. It shows, left to right, the original image, ground truth segmentation mask, the model's binary prediction, and, finally, the level of uncertainty estimated by spatial 2D dropout. We see that not only the model shows excellent segmentation results but also assigns reasonable uncertainty values, usually being least certain near the boundaries of a polyp.

Konica Minolta

WCE detection and localization tasks We evaluated our proposed method by using the training data set provided in WCE lesion detection and localization challenge in gastrointestinal image analysis (GIANA). The training data set is composed of 600 images without lesion, i.e., normal, 600 images with a vascular lesion and 600 images with an inflammatory lesion. The evaluation was conducted with cross-validation of the training data set. We divided the training data set into six groups and used four groups for training, one group for validation, and one group for testing. Thus, we had six folds for cross-validation and the performance was evaluated with average values and standard deviations of test data in six folds.

We used some evaluation metrics based on the definition in the WCE lesion detection and localization challenge. For classification, we calculated the following metrics; true positive rate (TPR), false positive rate (FPR), false negative rate (FNR), true negative rate (TNR), and accuracy. With respect to localization, we calculated precision, recall, F1 and F2 for two lesion types, i.e., vascular and inflammatory lesions.

Tables 2 and 3 show the summary of classification and localization performance, respectively. In those tables, the numbers mean "average $\pm$ standard deviation" and the units are percent. The average and standard deviation are calculated for test data of all folds in all lesion types for classification and two lesion types (vascular and inflammatory) for localization.

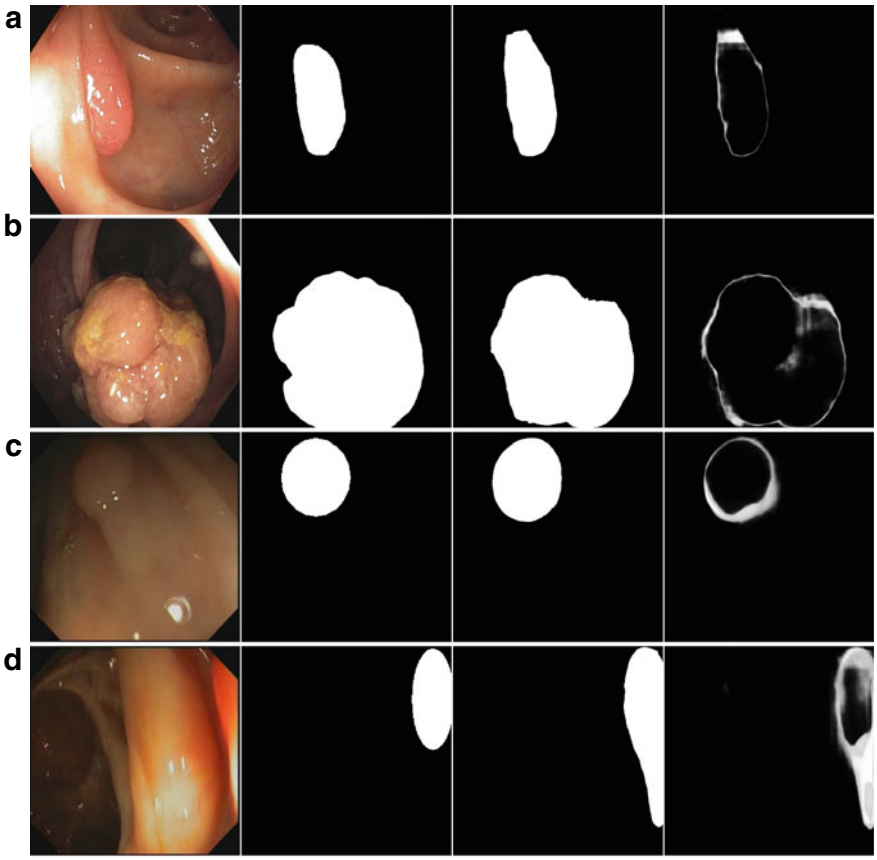

Fig. 11 Neuromation results. Left to right: **a** original image, **b** ground truth, **c** predicted mask, **d** uncertainty of the prediction

Table 2 Results of classification task

TPR	FPR	FNR	TNR	Accuracy
98.67 ± 0.42	0.67 ± 0.21	1.33 ± 0.42	99.33 ± 0.21	99.11 ± 0.28

Table 3 Results of localization task

Precision	Recall	F1	F2
88.76 ± 1.59	76.26 ± 4.05	81.98 ± 2.28	78.44 ± 3.37

References

Anas, E. M. A., Nouranian, S., Mahdavi, S. S., Spadinger, I., Morris, W. J., Salcudean, S. E., Mousavi, P., & Abolmaesumi, P. (2017). Clinical target-volume delineation in prostate brachytherapy using residual neural networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 365–373). Springer.

Chaurasia, A., & Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE Visual Communications and Image Processing (VCIP)* (pp. 1–4). IEEE.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., & Ramanan, D. (2009). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(9), 1627–1645.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML-16)*.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 1026–1034).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).

Hoang Ngan Le, T., Zheng, Y., Zhu, C., Luu, K., & Savvides, M. (2016). Multiple scale faster-rcnn approach to driver's cell-phone usage and hands on steering wheel detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (pp. 46–53).

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *6*(02), 107–116.

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 7132–7141).

Iglovikov, V., Rakhlin, A., Kalinin, A., & Shvets, A. (2017). Pediatric bone age assessment using deep convolutional neural networks. arXiv preprint arXiv:1712.05053.

Jiang, H., & Learned-Miller, E. (2017). Face detection with the faster r-cnn. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)* (pp. 650–657). IEEE.

Lin, T., Dollár, P., Girshick, R. B., He, K., Hariharan, B., & Belongie, S. J. (2016). Feature pyramid networks for object detection. *CoRR*, arXiv:abs/1612.03144.

Liu, Y., Minh Nguyen, D., Deligiannis, N., Ding, W., & Munteanu, A. (2017). Hourglass-shapenetwork based semantic segmentation for high resolution aerial imagery. *Remote Sensing* (vol. 9(6), p. 522).

Milletari, F., Navab, N., & Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)* (pp. 565–571). IEEE.

Mishkin, D., Sergievskiy, N., & Matas, J. (2017). Systematic evaluation of convolution neural network advances on the imagenet. *Computer Vision and Image Understanding*.

Rakhlin, A., Davydow, A., & Nikolenko, S. (2018, June). Land cover classification from satellite imagery with u-net and lovász-softmax loss. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

Rakhlin, A., Tiulpin, A., Shvets, A. A., Kalinin, A. A., Iglovikov, V. I., & Nikolenko, S. (2019). Breast tumor cellularity assessment using deep neural networks. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 234–241). Springer.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 648–656).

"What My Deep Model Doesn't Know...." http://mlg.eng.cam.ac.uk/yarin/blog_3d801aa532c1ce.html.

Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1492–1500).

Zhang, H., Cisse, M., Dauphin, Y. N., & Lopez-Paz, D. (2017). mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412.