YOLO as a Region Proposal Network for Diagnosing Breast Cancer

Ananya Bal
School of Computer Sc. & Engg.
Vellore Institute of Technology, Vellore
Tamil Nadu - 632014, India.
Email: ananyabal3098@gmail.com

Meenakshi Das
Department of Computer Sc. & Engg.
IIIT Delhi
New Delhi - 110020, India
Email: meenakshidas3646@gmail.com

Shashank Mouli Satapathy*
School of Computer Sc. & Engg.
Vellore Institute of Technology, Vellore
Tamil Nadu - 632014, India.
Email: shashankamouli@gmail.com

2021 Grace Hopper Celebration India (GHCI) | 978-1-6654-1284-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/GHCI50508.2021.9513988

***Abstract*—Cytological images of various types are increasingly being classified with the use of neural networks. But deep learning-based image classification systems are heavily reliant on manually sampled RoI (Region of Interest) patches. A lot of time and effort are required to extract RoI patches from whole slide images or larger images that are too complex to be processed by neural networks. A region proposal network (RPN) is an efficient way to automate the extraction of RoIs. In this study, we have proposed the use of the YOLOv3 network as an RPN to suggest RoIs in images from fine needle aspiration cytology of breast tissue. Patches from the suggested RoIs are fed into a Convolutional Neural Network (CNN) for the classification of benign and malignant lesions and ultimately, the diagnosis of Ductal Carcinoma in breast. The YOLO+CNN model yields a highly satisfactory classification accuracy of 95.73%, 100% specificity, 92.4% sensitivity and a precision score of 1.**

***Index Terms*—Breast Cancer, Computer Aided Diagnosis, Deep Learning, RoI Detection and Classification, YOLO**

I. Introduction

In India, Breast Cancer has the highest incidence among all cancers in women. According to the Globocan 2018 database, there were more than 1,60,000 incidences of breast cancer in India in 2018 [1]. Ductal carcinoma is malignancy (uncontrolled cell division) originating from the cells that line the milk ducts in breasts. More than 80% of all breast cancer cases are Ductal Carcinomas [2]. The National Cancer Registry Programme assesses that breast cancer incidence in India rises by 15-20% every decade [3]. For every two women diagnosed with breast cancer, only one survives [4]. This is due to lack of awareness and screening, and also, burdened diagnostic systems. In such a scenario, devising Computer-Automated Diagnosis (CAD) solutions is the need of the hour.

Currently, in the Indian healthcare industry, the most widely used method do diagnose breast cancer (Ductal Carcinoma) is Fine Needle Aspiration Cytology (FNAC). In this process, a narrow-gauge needle is used to collect a sample of a tissue lesion from the breast for microscopic examination. The aspirated sample is fixed onto a slide, stained and viewed under a microscope by a cytopathologist to make a diagnosis. FNAC is minimally invasive and enables faster diagnosis. In the realm of CAD, Deep Learning is increasingly being used to classify cytology images from FNAC lesions. Convolutional Neural Networks (CNN), a class of deep learning models, yield state-of-the-art results for image classification problems and are being adopted for digital diagnosis from medical images.

While CNNs perform well for medical image classification and are relatively convenient to build, they require a lot of data preparation and image processing. Data Scientists often need to collect, enhance, crop and augment images to ensure CNNs learn from relevant data. Relevant data in images is referred to as RoI (Region of Interest). Extracting RoIs from images is the most time-consuming step in medical image processing and even today, is mostly done manually. Automating RoI extraction with a Region Proposal Network (RPN) is key to saving time and effort. Our objective was to integrate an RPN with a lightweight CNN for a fully automated CAD system for Ductal Carcinoma. First, we present related work in section II and then give some background about the techniques being used in our work in section III. In section IV, we propose a framework that uses the YOLOv3 network and a CNN as an RPN and a classifier respectively. This is followed by results in section V. Sections VI and VII outline the conclusions, and discuss our findings and alternate solutions for future work.

II. Prior Work

Al-masni et al. [5] trained a YOLO-based CAD model that detected masses in mammograms and classified their types into benign and malignant. Their results show that the proposed system detected mass locations in the breast with an accuracy of 96.33% and distinguished between benign and malignant lesions with an accuracy of 85.52%. They expanded on their work in [6] where they proposed pre-processing mammograms to extract features before using YOLO for mass detection. The detected masses were classified with a Fully-Connected Neural Network. Their proposed CAD system yielded 99.7% classification accuracy. Mugahed A. et al. [7] proposed an integrated CAD system for lesion detection and classification from entire mammograms. First, a deep learning YOLO detector was used and evaluated for breast lesion detection. Then, three deep learning classifiers, a regular feedforward CNN, ResNet-50, and InceptionResNet-V2, were used for classification. Mugahed A. et al. [8] have utilized YOLO, FrCN and Alexnet for RoI detection, segmentation and classification respectively, achieving 95% accuracy.

978-1-6654-1284-121$31.00 ©2021 IEEE

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.

Gao et al. [9] proposed an approach to detect Squamous Cell Carcinomas from Oesophageal Endoscopic Videos using mask-RCNN and YOLOv3. They have used a Non-Maxima Technique which ensured the detection of a single bounding box. The bounding box region is classified into three classes. YOLOv3 gave a detection accuracy of 85% and a classification accuracy of 74%. Ding et al. [10] proposed a novel approach for lesion localization in gastroscopic videos. They detected upper gastrointestinal disease with a Tinier-YOLO algorithm and improved the modeling performance by integrating lesion RoI segmentation strategy into the YOLOv3 algorithm. Their results exhibit superior performance in mean Average Precision (mAP) and Intersection over Union (IOU).

Spanhol et al. [11] used deep learning on histopathological images from the BreakHis public dataset. They inferred that CNN was better than other ML models. They also investigated the combination of different CNNs using simple fusion rules, attaining some improvement in performance. Saikia et al. [12] used CNNs for the diagnosis of cell samples from FNAC images and tested VGG16 and other architectures for a comparative study. The experimental results showed that GoogLeNet-V3 achieved 96.25% accuracy. Miselis et al. [13] proposed new techniques built on neural networks by contrasting with five other models. They concentrated on the issues of CAD systems using FNAC images. The experimental outcomes showed that Inception-V3 with 92% accuracy, and 0.971 AOC is the preferred technique. Vesal et al. [14] fine-tuned the Inception-V3 and ResNet-50 networks and used weights from the ImageNet competition to classify images of the BACH 2018 challenge. They achieved 97.08% accuracy for Inception-V3 and 96.66% accuracy for ResNet-50.

III. Background

This section will introduce two major techniques in deep learning that we have leveraged - CNNs and YOLO.

A. CNNs

CNNs are neural networks which utilize the convolution function and are well-suited to process image data. They are able to learn low-level geometric features in images like edges and spots and associate them with high-level features such as objects. This has made them valuable for tasks like object detection, image classification, facial recognition, neural style transfer etc. They read images as the input and pass them through various layers where each layer, takes an input, does some processing and gives an output. Most CNNs have an architecture consisting of these main layers:

1) *Convolution Layer:* This layer is the building block of CNNs. It helps extract features by transforming an input tensor (uniform multi-dimensional array) into a feature map. This is done by convolving filters with identically-sized windows from the input (dot product of two matrices). A filter is a small matrix, with height and width smaller than the image to be convolved. This filter slides across the height and width of the image, and the dot product of the filter and the image is computed at every spatial position. The size of the output depends on the number of filters, the dimensions of the filter, the type of padding applied to the input as well as the stride (number of units by which the filter window slides). Another important component of the convolution layer is the Activation Function used to determine the output from the neurons. This is used to increase the non-linearity of the output. The use of activation functions helps CNNs learn complex classifier functions. The commonly used activation functions are Rectified Linear Unit (ReLU), Sigmoid and Tanh.
2) *Pooling Layer:* This layer is responsible for condensing features from the input and speeding up computations. Pooling layers often follow convolution layers. In pooling, a small window slides across the height and width of the input, storing only one value per spatial position for the output. This value can either be the highest value in the window (Max Pooling) or the average of all values in the window (Average Pooling). Max Pooling is the more commonly used pooling technique as it makes the existing features in the input more robust.
3) *Fully-Connected Layer:* CNNs use fully-connected layers as the last few layers to capture high-level features. All neurons are interconnected in these layers. A flatten function converts the output from convolution and pooling layers into a one-dimensional vector which is passed to the fully connected layers. The fully-connected layers classify images by learning relationships between high-level features and how they belong to separate classes.

B. YOLO

The YOLO architecture was designed keeping in mind object detection in video frames. It is a unified network that can identify and classify objects at one go [15] and is faster than other networks used for object detection. Unlike other object detection networks that look at an image in pieces, YOLO looks at the entire image and learns context, hence the name - You Only Look Once. Object detection is considered as a regression problem in YOLO architecture.

In the YOLO framework, an image is divided into an NxN grid and each grid cell is responsible for detecting bounding boxes for objects whose centres lie within the cell. Each grid cell predicts a fixed number of bounding boxes for an object by resizing a predetermined number of boxes (Anchors) of different aspect ratios. After this, the confidence scores for the boxes are calculated (Equation 1 and 2.). Confidence scores (or confidence values in the case of RoI detection) indicate the model's confidence in the accuracy of the box's fit to an object. Duplicate boxes or boxes having high overlap are discarded by Non-maximal Suppression (NMS) [15].

$$IOU = \frac{Area\ in\ intersection}{Area\ in\ union} \tag{1}$$

$$ConfidenceValue\ (cv)\ =\ Probability(Object) \times IOU \tag{2}$$

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.

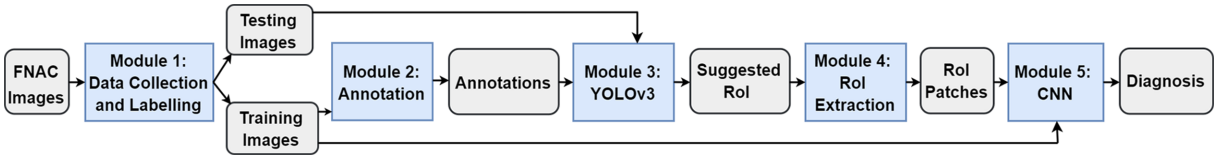

Fig. 1. Proposed Framework

A CNN learns the parameters associated with bounding boxes, namely, the coordinates of the centre of the box relative to the grid cell, height, width and the confidence value (x, y, h, w, cv). The first few layers of the CNN extract features from the images while the fully-connected layers at the end predict bounding box coordinates. The loss function used for training the CNN is a modified sum squared error function which shows the inspiration that the model takes from regression where squared error losses (Root Mean Squared error and Mean Squared Error) are optimized for best performance.

It is to be noted that YOLO has previously been employed for RoI detection in radiology i.e. mammograms and CT scans, and video frames but not to FNAC cytological images. When it comes to breast cancer diagnosis, mammography is a screening process whereas, FNAC is a conclusive diagnostic test. Therefore, a study on cytological images is crucial.

There are certain challenges associated with cytological images. They contain clusters of cells and do not have clearly demarcated objects. The RoIs are cell clusters which are not evenly shaped and do not have uniform boundaries. They vary a lot in size and shape. But perhaps, the biggest challenge with cytological images is the presence of multiple RoIs per image, unlike in the case of most radiological images. Hence, multiple overlapping bounding-boxes are required to cover all RoIs. Therefore, identifying RoIs in these images is a complex problem and required many modifications in the YOLO network. As in studies [7] and [8], we test YOLO as a region proposal network only. Hence, we have used it for RoI detection solely and not to classify the suggested RoIs.

IV. Proposed Framework

The workflow of our proposed framework is seen in Figure 1 and is divided into five modules which are as follows:

A. Data Collection and Labeling

Pathologists use microscopes to view slides fixed with stained tissue obtained from FNAC. The microscopic lesions can be captured by a camera to produce digital images. For this study, FNAC images of breast lesions with Giemsa stain were captured in pathology labs at East Point College of Medical Sciences and Research Centre, Bangalore, India and Maharaja Krushna Chandra Gajapati Medical College, Brahmapur, Odisha, India following all ethical protocols.

Benign FNAC samples collected from 40 patients and malignant FNAC samples (Ductal Carcinoma NOS) collected from 30 patients were used. These samples were observed at 400x magnification by certified cytopathologists using a CH-20i Olympus Binocular Microscope. The diagnosis for each sample has been confirmed by histopathology.

On an average, 1 to 2 images were taken per patient's sample. This produced 79 images of benign lesions and 56 images of malignant lesions. Roughly 80% of these images were used for training (64 benign images and 44 malignant images) the YOLOv3 network after annotation. The remaining images were used for testing (15 benign images and 12 malignant images). For training the CNN, 4 to 8 RoI patches of size 256x256 were manually extracted per image depending on the cellular content. This resulted in 389 benign patches and 300 malignant patches. See Table I. Nearly 75% of these RoI patches were placed under the training set and the remaining were placed under the validation set.

TABLE I
Data Distribution

Category	Benign	Malignant
Slide Images	79	56
Images for training YOLO	64	44
Images for testing YOLO	15	12
RoI patches for training CNN	389	300

B. Annotation

The training images need to be annotated to define ground truth for the model. Annotation is the process of identifying an object in an image with a box and tagging the object with a label. This was done with the help of the Microsoft Visual Object Tagging Tool (VoTT) software. This process saves the x and y coordinates of the annotated RoIs.

There is no prescribed way to annotate cell clusters. Many cell clusters are large and occupy a large part of the image. This type of cluster was not annotated with single large boxes. Instead, multiple small but tight boxes were used to cover the entire cell cluster as the YOLO network needed to learn to identify boxes which tightly fit the cells. It should be noted that annotations are different from the 256×256 training patches extracted for the CNN as the former is used to detect RoIs while the latter is used to classify RoIs.

C. YOLOv3 Network

The YOLOv3 network is an improved version of YOLO which uses the DarkNet53 as the feature extractor [16]. This is followed by the detector module which has multiple 1x1 and 3x3 convolution layers that detect bounding boxes at

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.

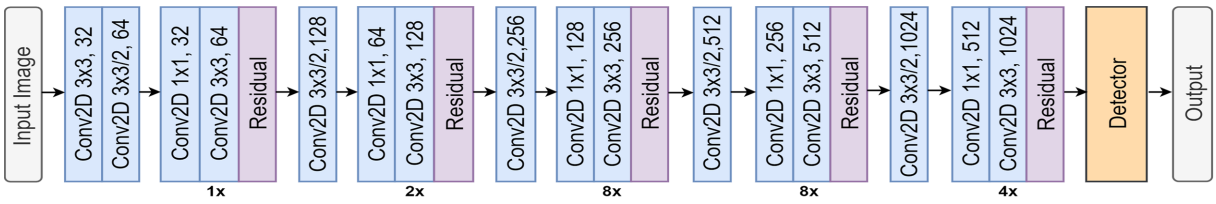

Fig. 2. DarkNet53 Feature Extractor and Detector in YOLOv3

three scales from the outputs of the last three residual blocks. Darknet53 has 53 layers - 52 convolution layers and one fully-connected layer. These layers use the Leaky ReLU activation function as shown in Equation 3 [17] to avoid dying neurons. The layers are interspersed with residual layers which contain residual blocks that feed the output from one layer into the next layer and into a few more consequent layers directly [18]. The architecture can be seen in Figure 2. This network reads annotations from the training images and learns to predict the x and y coordinates of bounding boxes for RoIs. The network's output is a list of suggested RoIs enclosed within x and y coordinates.

We leveraged transfer learning by using pre-trained weights from the ImageNet dataset for the DarkNet53 network. The training continued for 50 epochs with an initial learning rate of 0.0001 which was gradually reduced. We implemented the network with Keras and Tensorflow in Python and trained it on an Intel CoreTM i5-6200U CPU.

$$Leaky\ ReLU\ activation\ function: \quad f(x) = \begin{cases} x & if\ x\ >\ 0; \\ 0.1x & otherwise \end{cases} \tag{3}$$

After fine-tuning hyper-parameters, we settled on a batch size of 12 for the network and used the RMSprop optimizer [19] instead of the default optimizer in YOLOv3 - Adam [20]. In addition to this, in our implementation, we have used 9 anchor boxes. 15% of the training data was used for validation.

The key change in our implementation was the choice of the confidence score threshold value. Traditionally, a higher value (0.5-0.95) is used to retain the best-fitting bounding boxes that have confidence scores above the threshold. However, given the nature of our images and RoIs, the threshold was lowered to 0.1 to retain a few poorly fit bounding boxes. This ensures that RoIs are detected in all images and we get a sufficient number of RoIs for diagnosis by classification.

D. RoI Patch Extraction

The YOLOv3 network performed reasonably well on the test images suggesting a total of 164 bounding boxes for RoIs in 27 test images. However, these bounding boxes, and therefore RoIs, varied in size and shape. To use a CNN for classification, all RoI patches need to be of a uniform size (in our case, 256×256 pts). One option to overcome this hurdle is to resize all RoIs to the required dimensions. However, resizing alters the shape of cells especially if the aspect ratio of the RoI is not maintained. The RoIs vary in shape and so resizing them all to a single fixed size would stretch the cells in the patches. Since the distinction between benign and malignant cells lies in their shape, it is not wise to resize the RoIs.

Therefore, through a Python script, we have selected a 256×256 patch from the centre of each RoI suggested by YOLOv3 instead of resizing the RoIs. Refer to Figure 4 for visualization. This was a logical approach as we know that the required RoI, with the greatest number of cells, is usually at the centre of the bounding box. There were a few suggested RoIs which were slightly smaller in size than 256×256 pts. We did not hesitate to enlarge these images by resizing to 256×256 pts as their respective lengths and widths were nearly equal and were only slightly less than 256. Enlarging these RoIs did not extensively alter the shapes of the cells seen in them.

E. Classification with CNN

The CNN uses 256×256 patches taken from the training images and predicts class labels (benign and malignant) for all the RoI patches from the previous module.

We have constructed a simple lightweight CNN with 11 layers for classification in Python with the use of Keras and Tensorflow libraries. The network has four convolutional blocks followed by three fully-connected layers. The convolutional blocks each have two convolution layers and a max-pooling function. The numbers of filters in the convolution layers of the blocks are 32, 64, 128 and 256 respectively. The first two fully-connected layers have 256 nodes each and the last fully-connected layer has a single node. The CNN is visualized in Figure 3.

All layers except the last fully-connected layer use the ReLU activation function. All convolution layers use a 3×3 convolution window, a stride of 1, and the same convolution method, where the size of the input is preserved by using zero padding at the borders. The output dimensions of each convolution layer can be calculated using Equations 4 and 5. We have used the ReLU activation function (see Equation 6) for these layers as it is computationally more efficient and can accelerate convergence. The pooling layers downsize the input with a 2×2 window for two-dimensional max-pooling where only the highest value from each window is retained. Dimensions of the output from the max-pooling layer are calculated as per Equation 7.

The first two fully-connected layers are each followed a dropout of 0.5 to prevent overfitting and then batch normalization. Batch normalization [21] is used to enhance the network's

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.

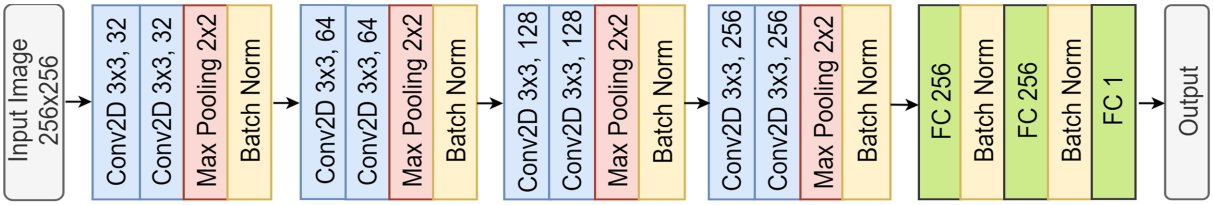

Fig. 3. CNN Architecture

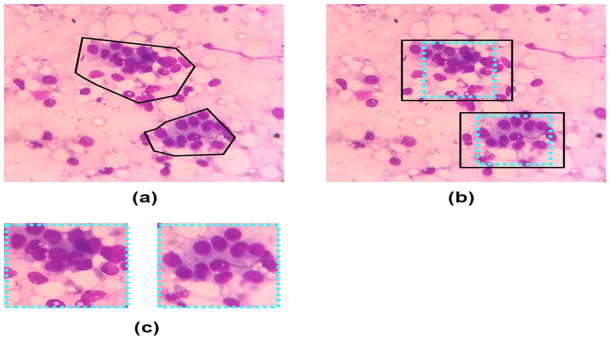

Fig. 4. (a) Ground Truth, (b) Predicted RoIs, (c) Extracted Patches

speed and stability in training. The last fully-connected layer uses the sigmoid activation function, as ours is a binary classification problem.

The CNN is trained with a learning rate of 0.0001 and the RMSProp optimizer [19]. All hyperparameters were fine-tuned and adjusted to give the best performance. Figure 3 shows the structure of the final CNN. The working of our network can be explained with Equations 4, 5, 6 and 7.

If P denotes the padding, F_w the width of the filter, F_h the height of the filter, S the stride value and N_c is the number of channels (3 in the case of RGB images), then:

$$Width\ post\ convolution\ (W) = 1 + \left(\frac{Input\ width + (2 * P)F_w}{S}\right) \tag{4}$$

$$Height\ post\ convolution\ (H) = 1 + \left(\frac{Input\ height + (2 * P)F_h}{S}\right) \tag{5}$$

$$ReLU\ (Rectified\ Linear\ Unit)\ activation\ function: f(x) = \begin{cases} 0\ if\ x\ <\ 0 \\ x\ if\ x\ \geq 0 \end{cases} \tag{6}$$

$$Dimension\ of\ Max-pooling\ Output = (W/2, H/2, N_c) \tag{7}$$

The CNN was trained for 50 epochs using NVIDIA Tesla P100 GPU support from Kaggle. The test set consisting of RoI patches was passed to the compiled model to produce class probabilities (0-1). The probabilities were thresholded at 0.5 to obtain class labels 0 (benign) and 1 (malignant). With another vector of the actual class labels, a confusion matrix was generated and used to calculate evaluation parameters.

V. Results

The YOLOv3 network identified 164 RoIs in 27 test images from which 256x256 patches were used for classification. The CNN correctly classified patches from 157 RoIs.

Evaluation Metrics: For the YOLOv3 RoI detector, evaluation is based on IoU and confidence scores (Refer Equations 1 and 2). The Average Precision recorded was 29.3%. 164 RoIs had a confidence score > 0.1 and an IoU with the ground truth ≥ 0.5. Among the 27 test images, our implementation of YOLOv3 was unable to detect RoIs for one image. This indicates the complexity of cytology images and that YOLOv3 was 96% effective in suggesting RoIs in cytological images.

For classification by the CNN, the evaluation is based on parameters derived from the confusion matrix. Refer Table III. In diagnostic tests, the presence of the disease is a positive outcome and vice versa. Therefore, a truly malignant patch classified as malignant is a True Positive (TP) and a truly benign patch classified as benign is a True Negative (TN). Similarly, a truly benign patch classified as malignant is a False Positive (FP) and a truly malignant patch classified as benign is a False Negative (FN). The evaluation parameters and their values are given in Table II.

VI. Conclusion

Except for one image, RoIs were detected by YOLOv3 in all test images. Further improvements to the network and the addition of more data to the pipeline will certainly improve bounding box detection, enabling the use of a higher confidence threshold in box selection.

Overall, our framework performed very well in detection and classification of RoIs in cytology images from breast FNAC. The diagnostic accuracy of our framework is 95.73%. The high sensitivity and specificity values of 0.924 and 1 indicate that the method is good enough for industry applications. We conclude that the YOLOv3 network is an effective Region Proposal Network for cytological images, but there is still scope for improvement. The combination of YOLOv3 and CNN fully automates the diagnosis of Ductal Carcinoma with good classification accuracy.

VII. Discussion and Future Work

The YOLOv3 failed to detect bounding-boxes in one image. This can be resolved by lowering the confidence threshold further. But since a lower threshold can predict bounding boxes for the background instead of objects, it is preferable to make improvements to the network instead of lowering the threshold.

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.

TABLE II
EVALUATION PARAMETER VALUES

Metric	Description	Formula	Value
Accuracy	Proportion of correctly classified cases	$\frac{(TP + TN)}{Total\ Cases}$	0.9573
Precision	Proportion of cases relevant to the diagnostic test (i.e. true positive cases out of all cases diagnosed positive)	$\frac{TP}{(TP + FP)}$	1
Recall (or Sensitivity)	True positive rate (i.e. the proportion of true positives out of all cases diagnosed positive)	$\frac{TP}{(TP + FN)}$	0.9239
Specificity	Proportion of actual negatives out of all cases diagnosed negative	$\frac{TN}{(FP + TN)}$	1
False Positive Rate	Rate of misclassified negatives	$\frac{FP}{(FP + TN)}$	0
False Negative Rate	Rate of misclassified positives	$\frac{FN}{(FN + TP)}$	0.0761

TABLE III
CONFUSION MATRIX

		Predicted Labels	
	Total = 164	Benign	Malignant
True Labels	Benign	72	0
	Malignant	7	85

While YOLO performs well, there are other architectures like Single Shot Detector (SSD) and Faster-RCNN that can be used for region proposal as well. In the future, a comparative analysis can be done on the efficacy of each of these networks for cytology images. Also, for the CNN classifier, a way to extract a minimum number of even-sized patches to cover all the information suggested in the RoIs needs to be devised to lower the loss of data from RoIs in the current method.

REFERENCES

[1] Cancer Today: International Agency for research on Cancer, "Iarc world cancer report 2020," https://www.iccp-portal.org/sites/default/files/resources/IARC-World-Cancer-Report-2020.pdf, 2018, last Accessed: 20-2-2020.

[2] Breastcancer.org, "Invasive ductal carcinoma: Diagnosis, treatment, and more," https://www.breastcancer.org/symptoms/types/idc, 2019.

[3] National Centre for Disease Informatics and Research, "Ncpr three-year report of population based cancer registries 2012-2014," https://ncdirindia.org/NCRP/ALL_NCRP_REPORTS/PBCR_REPORT_2012_2014/ALL_CONTENT/PDF_Printed_Version/Chapter10_Printed.pdf, 2020.

[4] National Cancer Institute (NCI-AIIMS, "Cancer statistics — drupal," http://nciindia.aiims.edu/en/cancer-statistics, 2020.

[5] M. A. Al-masni, M. A. Al-antari, J. Park, G. Gi, T.-Y. Kim, P. Rivera, E. Valarezo, S.-M. Han, and T.-S. Kim, "Detection and classification of the breast abnormalities in digital mammograms via regional convolutional neural network," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2017, pp. 1230–1233.

[6] M. A. Al-Masni, M. A. Al-Antari, J.-M. Park, G. Gi, T.-Y. Kim, P. Rivera, E. Valarezo, M.-T. Choi, S.-M. Han, and T.-S. Kim, "Simultaneous detection and classification of breast masses in digital mammograms via a deep learning yolo-based cad system," *Computer methods and programs in biomedicine*, vol. 157, pp. 85–94, 2018.

[7] M. A. Al-antari and T.-S. Kim, "Evaluation of deep learning detection and classification towards computer-aided diagnosis of breast lesions in digital x-ray mammograms," *Computer Methods and Programs in Biomedicine*, p. 105584, 2020.

[8] M. A. Al-Antari, M. A. Al-Masni, M.-T. Choi, S.-M. Han, and T.-S. Kim, "A fully integrated computer-aided diagnosis system for digital x-ray mammograms via deep learning detection, segmentation, and classification," *International journal of medical informatics*, vol. 117, pp. 44–54, 2018.

[9] X. Gao, B. Braden, S. Taylor, and W. Pang, "Towards real-time detection of squamous pre-cancers from oesophageal endoscopic videos," in *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE, 2019, pp. 1606–1612.

[10] S. Ding, L. Li, Z. Li, H. Wang, and Y. Zhang, "Smart electronic gastroscope system using a cloud–edge collaborative framework," *Future Generation Computer Systems*, vol. 100, pp. 395–407, 2019.

[11] F. A. Spanhol, L. S. Oliveira, C. Petitjean, and L. Heutte, "Breast cancer histopathological image classification using convolutional neural networks," in *2016 international joint conference on neural networks (IJCNN)*. IEEE, 2016, pp. 2560–2567.

[12] A. R. Saikia, K. Bora, L. B. Mahanta, and A. K. Das, "Comparative assessment of cnn architectures for classification of breast fnac images," *Tissue and Cell*, vol. 57, pp. 8–14, 2019.

[13] B. Miselis, T. Fevens, A. Krzyżak, M. Kowal, and R. Monczak, "Deep neural networks for breast cancer diagnosis: Fine needle biopsy scenario," in *Polish Conference on Biocybernetics and Biomedical Engineering*. Springer, 2019, pp. 131–142.

[14] S. Vesal, N. Ravikumar, A. Davari, S. Ellmann, and A. Maier, "Classification of breast cancer histology images using transfer learning," in *International conference image analysis and recognition*. Springer, 2018, pp. 812–819.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[16] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[17] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1, 2013, p. 3.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[19] G. Hinton, N. Srivastava, and K. Swersky, "Coursera: Neural networks for machine learning: Lecture 6(a)—overview of mini-batch gradient descent," https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf, 2014.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

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

Authorized licensed use limited to: Tsinghua University. Downloaded on August 31,2021 at 20:20:40 UTC from IEEE Xplore. Restrictions apply.