Chapter 5

Salient Object Detection using YOLOv2 and Faster R-CNN

This chapter describes the deep learning-based models developed for salient object detection and colon tumor localization. The main contributions are: development of a hybrid model for SOD; providing user-choice to improvise result; using shrunk box co-ordinates for SOD; analysing behaviour of augmented images for polyp detection. The models are introduced in Section 5.1. The brief introduction of state-of-the-art methods is described in Section 5.2. The description of the proposed models is given in Section 5.3. Section 5.4 gives the result generated by the proposed models and their analysis. Section 5.5 concludes the chapter.

5.1 Background

Salient object detection is the process of locating prominent objects in an image. In this field, deep learning methods are providing outstanding results. One way of finding salient objects is to first obtain a bounding box for the prominent object in the image. Then, use the bounding box to form the actual shape of the salient object. In this work, the proposed model finds the object bounding box using the YOLOv2 network. Next, boundary correction is applied to the bounding box predicted by the deep network. In the third step, the proposed model segments the image using a set of Gabor filters. Then the matching segment from the first level boundary correction is selected. On the matching segment, the proposed model applies second level boundary correction. Usually, in salient object detection, the end-user plays no role in choosing the prominent object. In this work, the user is provided with a choice to improvise on the salient object detected at the first level. If the user is not satisfied with first level boundary correction, he/she can choose for second-level boundary correction. The method provides a benefit over existing methods as most of the saliency map results are static, and simple deep learning methods have blurred edges. By using this procedure, neat object edges are obtained. The algorithm is tested on three datasets against two state-of-the-art methods. The algorithm is evaluated based on F-Measure.

Colorectal cancer is a cancer of the colon/rectum. Automated polyp localization in colon endoscopy images helps minimize human errors in localizing polyps. Recently, deep learning has been used for localizing polyps. In the proposed method, Faster

R-CNN on Resnet 50 network is used to form a tight bounding box around the polyp. The bounding box is then used as input for the lazy snapping technique to determine polyps correctly.

Three input variants - RGB images, Histogram equalized images, and Luminance images - are fed to the network. The output obtained from each variant is combined to form the final result. For this work, the proposed model uses the CVC-ClinicalDB database, which has 612 images with 672 polyp instances. Thirteen different combinations for obtaining the result are studied, and the best among them are identified. The result is evaluated for all combinations and against a state-of-the-art method in terms of precision, recall, and F-measure. The proposed model achieves a precision of 80.51% and a recall value of 80.33%. It is concluded that with a variety of inputs being fed to deep network, their combination can achieve far better results in detecting polyps than using only a single type of input.

5.2 Literature of YOLOv2 and Faster R-CNN

This section explains the theoretical background of the proposed models. The proposed models use Gabor filters, lazy snapping, and YOLOv2 [198] network for locating salient objects. Gabor filters and lazy snapping have been discussed in Section 4.3.3 and Section 4.3.6, respectively. A brief discussion of YOLOv2 network, Faster R-CNN and Resnet 101 is given below.

YOLOv2 designed by Redmon and Farhadi can detect 9000 object categories. It

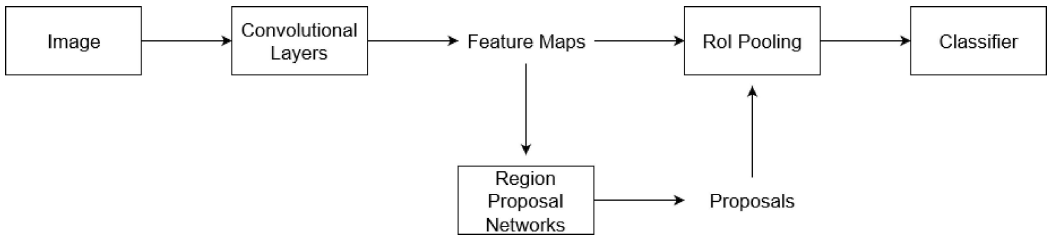

FIGURE 5.1: Working of Faster R-CNN.

employs a multi-scale method of training and uses a hierarchical perspective for categorizing objects. YOLOv2 adds batch normalization for all convolutional layers of YOLO [57] to refine convergence. YOLOv2 trains the network on images of size 448×448 instead of 224×224, thus providing high-resolution training. YOLO predicts coordinates of bounding boxes, whereas YOLOv2 predicts offsets, which simplifies the learning. For predicting offsets, YOLO uses handpicked priors, YOLOv2 instead relies on clustering to select priors. The location of anchor boxes depends upon the location of the grid cell. The feature map produced is of size 26×26. For faster execution, network model Darknet-19 is built. It has 19 convolutional layers and five max-pooling layers. The above features make the YOLOv2 network better, stronger and faster.

Faster R-CNN [199] tries to improve the region proposal network, which was earlier an obstruction for detection networks. It uses convolutional features in full resolution to detect object proposals. The working of Faster R-CNN is shown in FIGURE 5.1. The network is built on the concept of convolutional layers that can be shared. The region proposal network outputs a set of rectangular object bounding boxes with an objectness score. The anchor boxes predicted by the network are translation invariant. Also, they are of multiple scales and aspect ratios. For labeling an

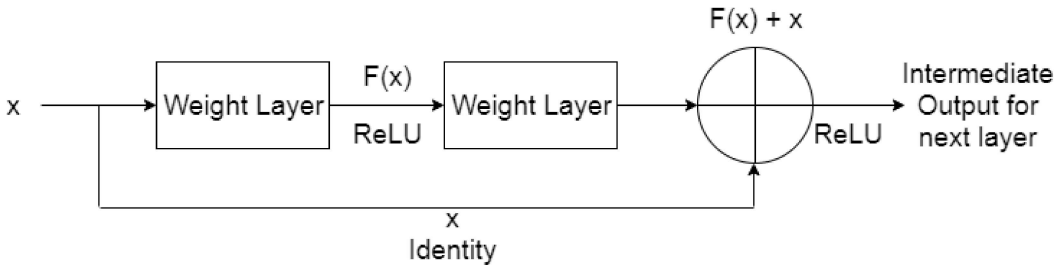

FIGURE 5.2: The residual learning framework used in Resnet 101.

anchor box as a positive/negative intersection of union with ground truth boxes is used. The box with maximum IoU or with IoU greater than 0.7 is used. Using the RPN, region proposal generation is well organized and correct. By sharing convolutional layers, the region proposal generation incurs no cost.

Resnet 101 is used to train very deep networks using residual functions. Together with the ease of optimization, these networks produce results with very high accuracy because of the depth of the networks. In the residual learning framework, the authors try to optimize a residual of the original function rather than optimizing the original function itself. The residual learning framework is shown in FIGURE 5.2. The residual mapping is implemented using "short connections". By using identity mappings, the training error is kept in control.

Following are some state-of-the-art methods against which the proposed models have been compared.

Wu *et al.* [39] (RBS) in their experiment use graph with superpixels as nodes. They generate a foreground as well as a background map. The saliency information from these two maps are added as seeds to the graph.

Zhou *et al.* [33] (FT) in their experiment use several prior maps. Then they use

fuzzy theory to rank the object proposals generated for an image. Fusing information from the prior maps and object proposals, the final salient object is obtained. The authors in [148] have used conditional GANs to produce synthetic images and added these synthetic images to the original database.

5.3 Methods and Models

This section describes the methodology adopted for the implementation of the proposed model. Section 5.3.1 describes the user interactive deep learning-based model for salient object detection. Section 5.3.2 describes a colon tumor detection model based on deep learning.

5.3.1 User Interactive Salient Object Detection using Yolov2, Lazy Snapping and Gabor Filters

This section describes how the selection of the network is made, how the training is performed, and how the salient object is selected.

5.3.1.1 Selection of pre-trained deep network

The YOLOv2 [198] network was tested with many base networks: VGG19 [60], VGG16, Resnet101 [58], Resnet50, Resnet18, Mobilenet [200], Googlenet [59]. After seeing the performance result, YOLOv2 with base network as Resnet101 was

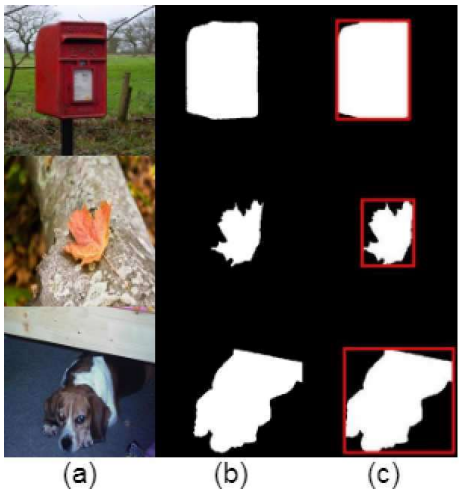

FIGURE 5.3: (a) Original Image (b) Ground Truth (c) Bounding Box for the salient object shown in red.

selected.

5.3.1.2 Training

The following steps are involved for training the YOLOv2 network.

1. **Generating box co-ordinates for ground truth of databases**: The rectangular box co-ordinates of the salient object are calculated from the ground truth of the databases. An example of a bounding box for ground truth is shown in FIGURE 5.3.

2. **Estimating anchor boxes for training the YOLOv2 network**: Anchor boxes are used for the training of the YOLOv2 network to capture the scale

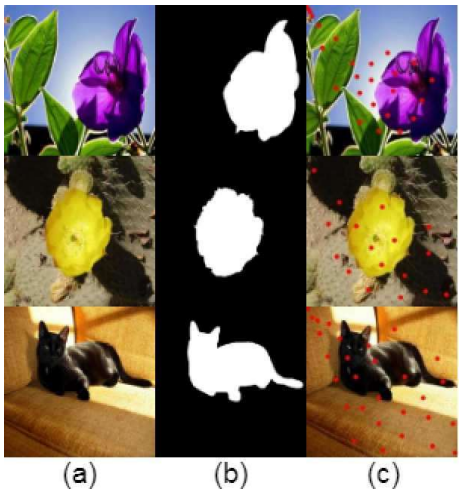

FIGURE 5.4: (a) Original Image (b) Ground Truth (c) The red dots show the center of anchor boxes for the image.

and size aspects of salient objects in the database. The number of anchor boxes is a hyper-parameter that must be tuned to obtain effective performance. The best method to select the number of anchor boxes is to check how much the anchor boxes overlap with the boxes in the training data. This measure is called the mean IoU. Higher mean IoU shows the better distribution of anchor boxes. FIGURE 5.4 shows the center of anchor boxes for a few images.

3. **Training the Yolov2 network**: The YOLOv2 network is trained for each database using Resnet 101 as the base network. Features are extracted from the 22^{nd} ReLU unit in the network.

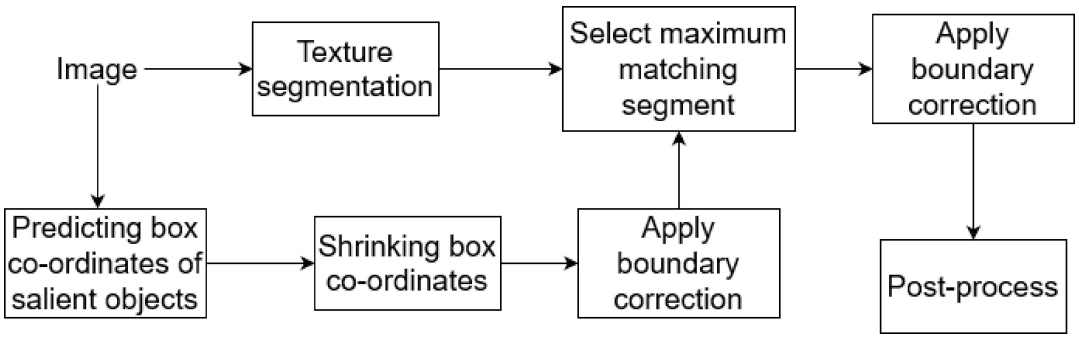

FIGURE 5.5: Block diagram of the proposed deep learning-based model.

5.3.1.3 Procedure of locating salient object

The block diagram of the procedure is shown in FIGURE 5.5. The steps are explained below.

1. **Using the trained network, predicting box co-ordinates for salient object**: In this step, first, the input image resized to 224 × 224. Then it is passed through the trained detector. The detector predicts the box co-ordinates of salient object in the image. The predicted bounding box is shown in yellow in FIGURE 5.6 (b).

2. **Shrinking box co-ordinates such that they represent central half of original box co-ordinates**: The predicted box co-ordinates for the salient object are generally superfluous. This is clear from the images in FIGURE 5.6 (b), where a large portion of the sky is covered for images 1 and 2, and the grass area is covered in image 2. This may lead to errors in the boundary correction algorithm. To overcome this problem, the box co-ordinates are shrunken to cover half the original area at the center of the original box co-ordinates. The

center part contains the salient object most of the time. This improves the performance of the boundary correction algorithm. The shrunken bounding boxes are shown in blue in FIGURE 5.6 (c).

3. **First level boundary correction**: First, non-local mean filtering is applied to the image. It removes noise from the image and retains the sharpness of strong edges. Then, for boundary correction, the proposed model uses the lazy snapping [191] method. The shrunken box co-ordinates are used as foreground object, and the area beyond the original box co-ordinates are marked as background. This result is named as INT and is shown in FIGURE 5.6 (d).

4. **Texture segmentation**: The texture segmentation is done by a set of Gabor filters of particular frequency and orientation. The result of the Gabor filters is post-processed using Gaussian smoothing. K-means clustering is used to produce the final result. This result is named as GAB. The result of the texture segmentation is shown in FIGURE 5.6 (e).

5. **Segment Selection**: After the texture segmentation of the image, the proposed model selects the segment which matches maximum to the result obtained from first level boundary correction. INT is used to select the maximum matching segment in GAB. The selected segment is named as SEG. This step improves the performance of second-level boundary correction. The maximum matching segment is shown in FIGURE 5.6 (f).

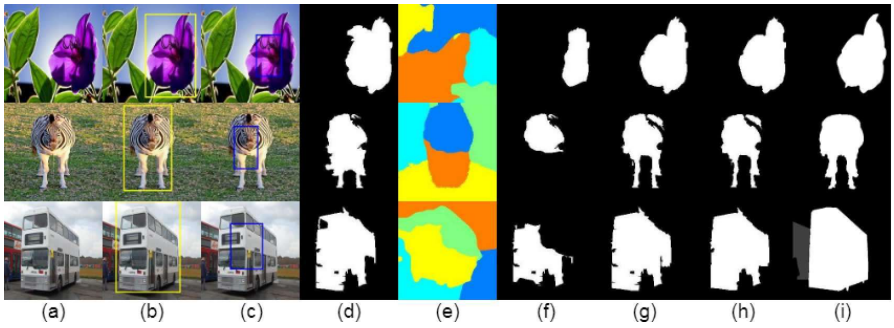

FIGURE 5.6: (a) Original Image (b) Predicted bounding box shown in yellow (c) Shrunken bounding box shown in blue (d) Result of first level boundary correction (e) Texture Segmentation (f) Matching Segment (g) Second Level Boundary Correction (h) Final result after morphological operation (i) Ground Truth.

6. **Second level boundary correction**: The second boundary correction is also done with lazy snapping. This time the proposed model uses the maximum matching segment, SEG, as foreground. The area beyond the original bounding boxes is used as background. The output of second-level boundary correction is shown in FIGURE 5.6 (g).

7. **Morphological operations:** The output of second-level boundary correction is corrected by smoothening of edges using dilation and erosion and filling of holes. FIGURE 5.6 (h) shows the post-processed output.

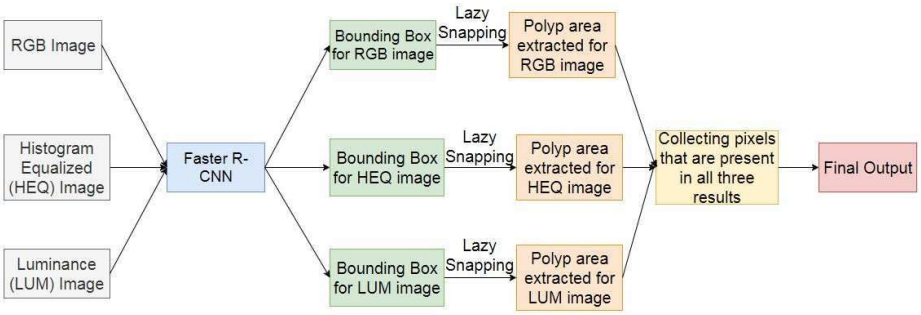

FIGURE 5.7: Block diagram of proposed method for colon tumor detection.

5.3.2 Colon Tumour Localization using Three Input Variants to Faster R-CNN and Lazy Snapping

In this section, the methodology used for developing the proposed model is discussed. Subsection 5.3.2.1 displays the block diagram of the proposed model. In subsection, 5.3.2.2 application of Faster R-CNN for object detection is explained. Subsection 5.3.2.3 describes the types of inputs that are provided to the network. The lazy snapping method is discussed in subsection 5.3.2.4.

5.3.2.1 Block Diagram

The block diagram of the proposed model is given in FIGURE 5.7.

5.3.2.2 Object Detection on Faster R-CNN using Resnet50

Faster R-CNNs [199] have proved to be very useful as they employ the region proposal network, which was earlier a bottleneck for R-CNN and Fast-RCNN. They

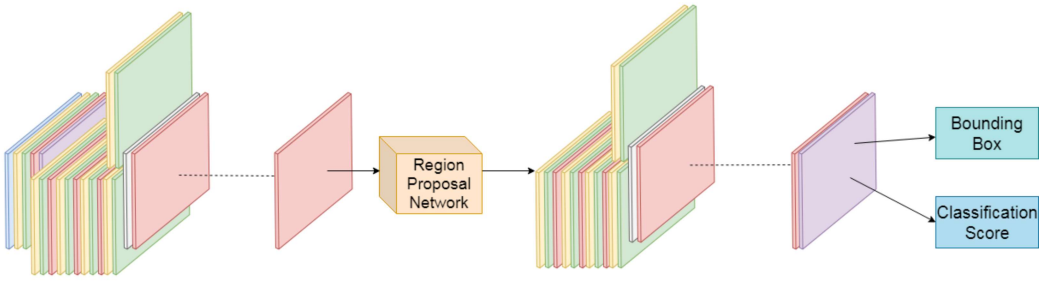

FIGURE 5.8: Block diagram of the network.

engage what is called as the "attention" mechanism, which makes the network look for objects the proposed model needs to find in the image. This property helps the network find the polyps in the colonoscopy images.

In [199], authors mention that Faster R-CNN has better results on Resnet Networks than VGG. For this reason, in this model, Faster R-CNN built on Resnet 50 [58] is used. There are 188 layers and 205 connections in the network. Resnet-50 network provided in MATLAB R2019a is pre-trained on a subset of ImageNet dataset for 1000 class categories.

A block diagram of the network is shown in FIGURE 5.8. The regional proposal network is enclosed between other layers of the network. Two network outputs are obtained: first is the location of the bounding box, and the other is the classification score. The yellow blocks represent 2-D Convolutions, green blocks represent Batch Normalization, and the red blocks are the ReLU activation unit. Purple blocks depict Pooling.

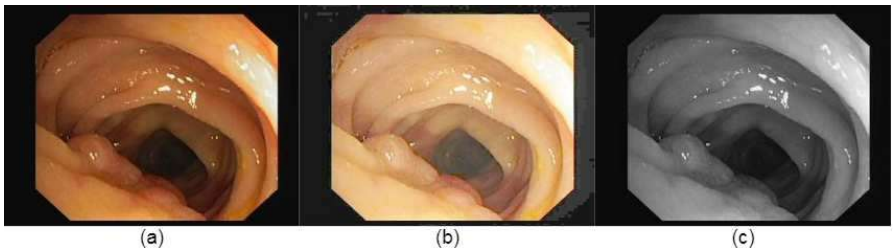

FIGURE 5.9: Input variants to Faster R-CNN. (a) RGB Image (b) Histogram Equalized Image (c) Luminance Image.

5.3.2.3 Input Variants

Three types of inputs are fed into the Faster R-CNN.

1. **RGB images**: RGB color images are fed into the network. The network extracts features from all the three channels of the image.

2. **Histogram Equalized Images**: Histogram equalization adjusts the image intensity to enhance the contrast of the images. For this, all the images are first histogram equalized and then fed to the network.

3. **Luminance**: Luminance is the Y component of YIQ color space defined by the National Television Systems Committee. It is the measure of light reflected from the surface. For this, all images are first converted to the YIQ color space. The Y component is then extracted and fed to the network.

FIGURE 5.9 shows the input variants of the polyp image that are provided to the network.

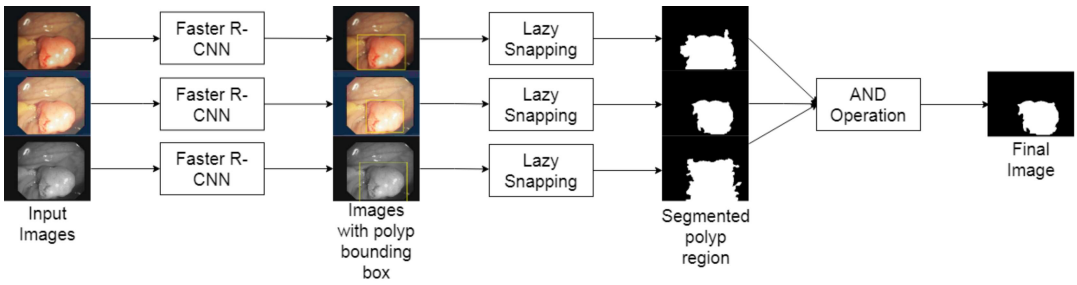

FIGURE 5.10: An example of the proposed model.

5.3.2.4 Lazy Snapping

Lazy Snapping [191] is a combination of graph-cut and over-segmentation. It easily works its way through weak boundaries and edges with low-contrast. The authors claim that it works better than the state-of-the-art technique Magnetic Lasso.

In this work, after the network provides the bounding boxes, lazy snapping is used to detect polyp boundary to separate it from the colon wall.

An example of the working of the proposed model is shown in FIGURE 5.10.

5.4 Results

The results of the proposed model are discussed in the following sections. Section 5.4.1 describes results obtained for user-interactive model of salient object detection. Section 5.4.2 shows the results obtained by using deep learning for colon tumor localization.

5.4.1 Result of YOLOv2 networks for general images

This section discusses the result obtained at each step and compares the final output with state-of-the-art algorithms. An analysis of failure results is done to show the limitations of the proposed model.

5.4.1.1 Parameter Selection

1. **Mean IoU:** To have a better distribution of anchor boxes, the mean IoU threshold was set to 0.8. For ASD we have used 26 anchor boxes. For ECCSD and PASCAL-S we have used 17 and 30 anchor boxes respectively.

2. **Gabor Filters:** Gabor filters are used with orientation ranging from 0° to 150° in steps of 30°. The wavelength ranges from $\frac{4}{\sqrt{2}}$ to the length of the hypotenuse of the input image. These parameters are obtained from the result of the work done by Jain and Farrokhnia [189].

3. **No. of segments:** The texture segmentation of images into 5 clusters give comparatively better result than 3 clusters. We achieve an improvement of 0.7% in weighted F-Measure 0.8% in average precision.

5.4.1.2 Ablation Study

By using the shrunken bounding box instead of original ones, the proposed model achieves an improvement of 19% in the weighted F-measure.

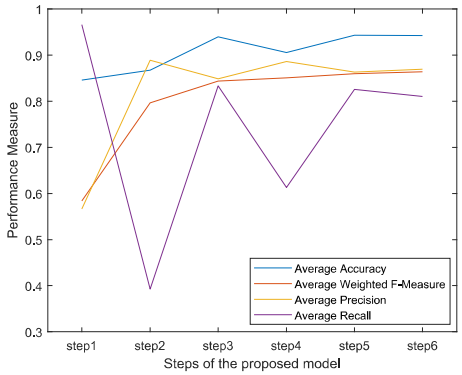

FIGURE 5.11: Step-wise performance measure for ASD dataset. **Step 1**: Original box Co-ordinates **Step 2**: Shrunken box co-ordinates **Step 3**: First-level boundary correction **Step 4**: Maximum matching segment **Step 5**: Second-level boundary correction **Step 6**: Final output after morphological processing.

The proposed model gives an average accuracy of 94.24% for the ASD dataset, 87.17% for the ECCSD dataset, and 86.94% for the PASCAL-S dataset. The average recall is 81.05% for ASD, 62.49% for ECCSD, and 68.34% for PASCAL-S. The mean precision is 86.95% for ASD, 79.77% for ECSSD, and 76.15% for PASCAL-S. A study of the performance improvement achieved in each step can be done by the graph shown in FIGURE 5.11, FIGURE 5.12, and FIGURE 5.13. Full coverage of salient objects by bounding box is the reason for high recall in step 1. High precision is obtained in step 2 as the area covered by shrunken bounding boxes belong to the salient region in most of the images. The reason for high precision in step 4 is also the same. The maximum matching segment belongs to the salient area.

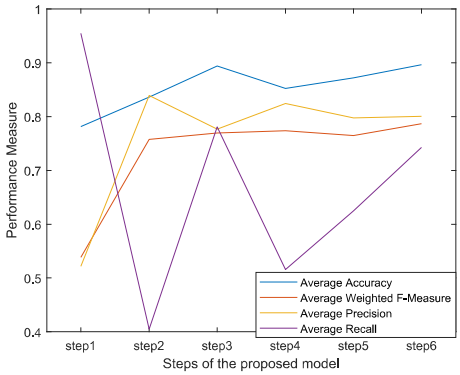

FIGURE 5.12: Step-wise performance measure for ECSSD dataset. **Step 1**: Original box Co-ordinates **Step 2**: Shrunken box co-ordinates **Step 3**: First-level boundary correction **Step 4**: Maximum matching segment **Step 5**: Second-level boundary correction **Step 6**: Final output after morphological processing.

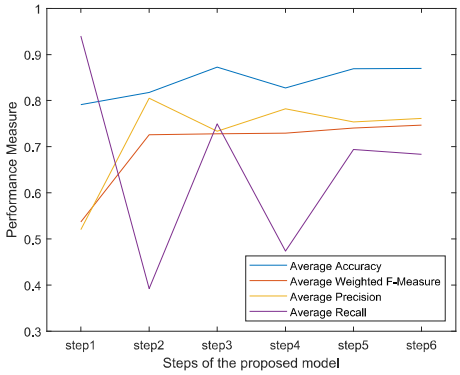

FIGURE 5.13: Step-wise performance measure for PASCAL-S dataset. **Step 1**: Original box Co-ordinates **Step 2**: Shrunken box co-ordinates **Step 3**: First-level boundary correction **Step 4**: Maximum matching segment **Step 5**: Second-level boundary correction **Step 6**: Final output after morphological processing.

5.4.1.3 Comparison with state-of-the-art methods

A comparison of the respective weighted F-Measure of the algorithms is given in TABLE 5.1. For the ECSSD database, the proposed algorithm is better than RBS [39] by a margin of 7.68% and by 22.15% when compared to FT [33]. For PASCAL-S, the proposed model has significant improvement - 27.35% for RBS [39] and 35.45%

TABLE 5.1: Comparison of the proposed algorithm to state-of-the-art algorithm on the basis of Weighted F-Measure

Methods \ Datasets	ASD	ECSSD	PASCAL-S
Proposed	0.86	0.7904	0.745
RBS [39]	0.912	0.734	0.585
FT [33]	0.85	0.65	0.55

for FT [33].

5.4.1.4 Failure Results

The reasons for the failure of the proposed model are analyzed below.

1. **Wrong object detection**: If the object detection by the deep network is far from the actual salient object, the consequent steps will start on the wrong foot. Hence, the final output is affected.

2. **Wrong boundary correction**: The failure of a lazy snapping algorithm due to heterogeneous objects and objects with blurry edges might sometime result in wrong boundary correction. Hence, the proposed model obtains the wrong result.

5.4.2 Result of YOLOv2 network for medical images

This section gives the experimental details and results of the implementation of the proposed model. Subsection 5.4.2.1 provides the environmental information in which the experiment is carried out. Subsection 5.4.2.2 explains the various

combinations that are tried to generate the final result. Subsection 5.4.2.3 lists the evaluation parameters for the proposed model. Subsection 5.4.2.4 explains why Faster R-CNN was the choice for polyp detection. Subsection 5.4.2.5 justifies the choice of input variants. In subsection 5.4.2.6, an ablation study is conducted in which experiments are done to show that the result obtained from bagging pixels from an individual input variant is better than the individual results. In subsection 5.4.2.7, a comparison of all the combinations proposed in section 5.4.2.2 is made, and the best among them are identified.

5.4.2.1 Environment Setup

The proposed model was implemented on MATLAB R2019a on a Windows 7 desktop with Intel(R) Xeon(R) CPU-2630 v3 @ 2.40Ghz with 32 GB memory. The system also has NVIDIA Version 425.31 Quadro K620 GPU with 2 GB dedicated memory and 384 CUDA Cores. The compute capability of the GPU device is 5.0.

5.4.2.2 Exploring Combinations of output from a network trained using three input variants

For deciding which combination will provide the best result, all combinations were tried. Final results were obtained from testing the following combinations:

1. Individual results obtained by performing Lazy Snapping on bounding box obtained from

(a) RGB images (RGB_LS)

(b) Histogram equalized images (HEQ_LS)

(c) Luminance images (LUM_LS)

2. Combining by AND operation: Only pixels occurring in all the bounding boxes results are combined, and then Lazy Snapping is performed.

   (a) RGB image and Histogram equalized image (HR_AND)

   (b) RGB image and Luminance image (LR_AND)

   (c) Histogram Equalized Image and Luminance image (HL_AND)

   (d) RGB image, Histogram equalized image and Luminance image (HLR_AND)

3. Combining by OR operation: Pixels occurring in either of the bounding boxes results are combined, and then Lazy Snapping is performed.

   (a) RGB image and Histogram equalized image (HR_OR)

   (b) RGB image and Luminance image (LR_OR)

   (c) Histogram equalized image and Luminance image (HL_OR)

   (d) RGB image, Histogram equalized image and Luminance image (HLR_OR)

4. Lazy Snapping is performed on individual results, and then final output is obtained by selecting pixels occurring in all results. (ALL)

5. Lazy Snapping is performed on individual results, and then final output is obtained by selecting pixels which are present in at least two results. (MAX)

So in total, 13 combinations are tested and evaluated.

5.4.2.3 Evaluation Parameters

The proposed model is evaluated based on precision and recall.

5.4.2.4 Choice of Deep Network

Three alternatives for Deep Network were selected: YOLOv2 [198], Fast R-CNN [55] and Faster R-CNN [199]. Among these three, Faster R-CNN gave the best average precision in detecting polyp bounding boxes. The results are shown in FIGURE 5.14. The evaluation was done for RGB images.

5.4.2.5 Choice of Input Variants

The choice of input variants was decided based on the average precision they obtained for the finding the polyp bounding box. FIGURE 5.15 shows the average precision found for each of the types of images.

The choice was made against edge intensity images, gradient maps, components of various color models like HSV, LAB, and YCbCr. The Discrete Cosine Transform and Discrete Fourier Transform of images were also used as input to Faster R-CNN, but satisfactory results were not obtained.

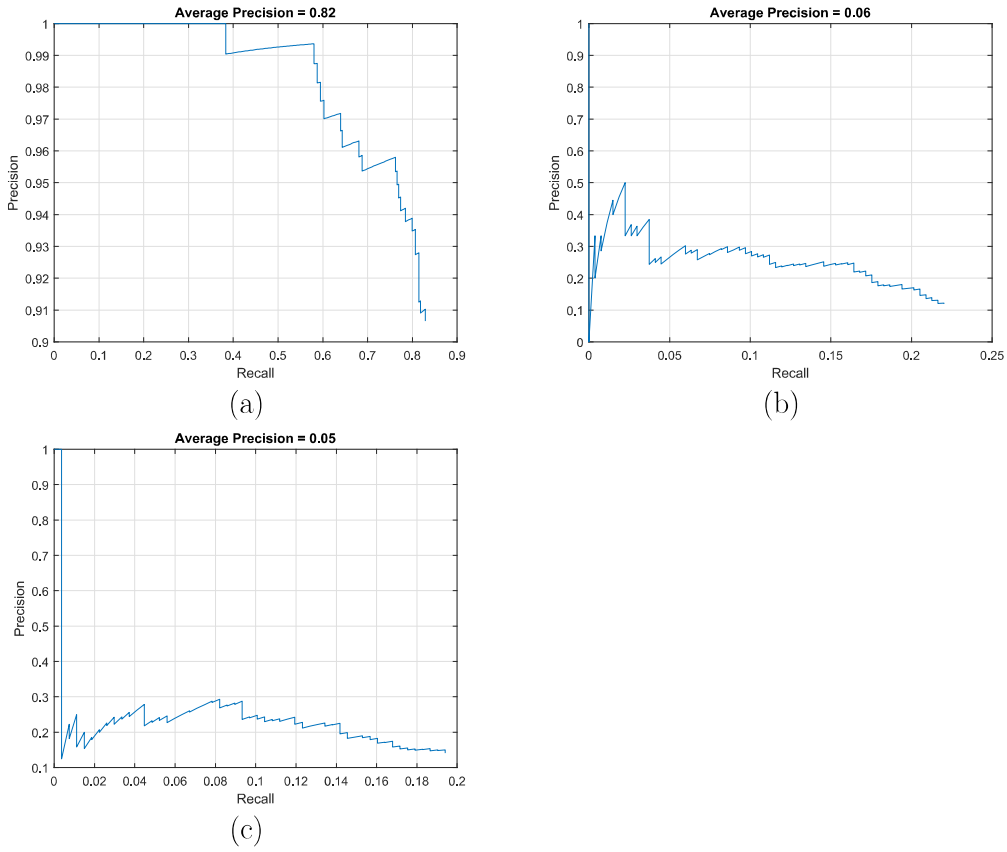

FIGURE 5.14: Average precision for each type of network. (a) Faster R-CNN has average precision 82% (b) Fast R-CNN gives an average precision of 6% (c) YOLOv2 has an average precision of 5%.

5.4.2.6 Ablation Study Results

For ablation study, the result acquired from individual lazy snapping of bounding boxes obtained from RGB images, Histogram equalized images, and Luminance images are compared with the result acquired from collecting pixels which are present in all three individual segmentation results. The results are shown in FIGURE 5.16. It can be seen that none of the individual results are as close to the ground truth as the last column result.

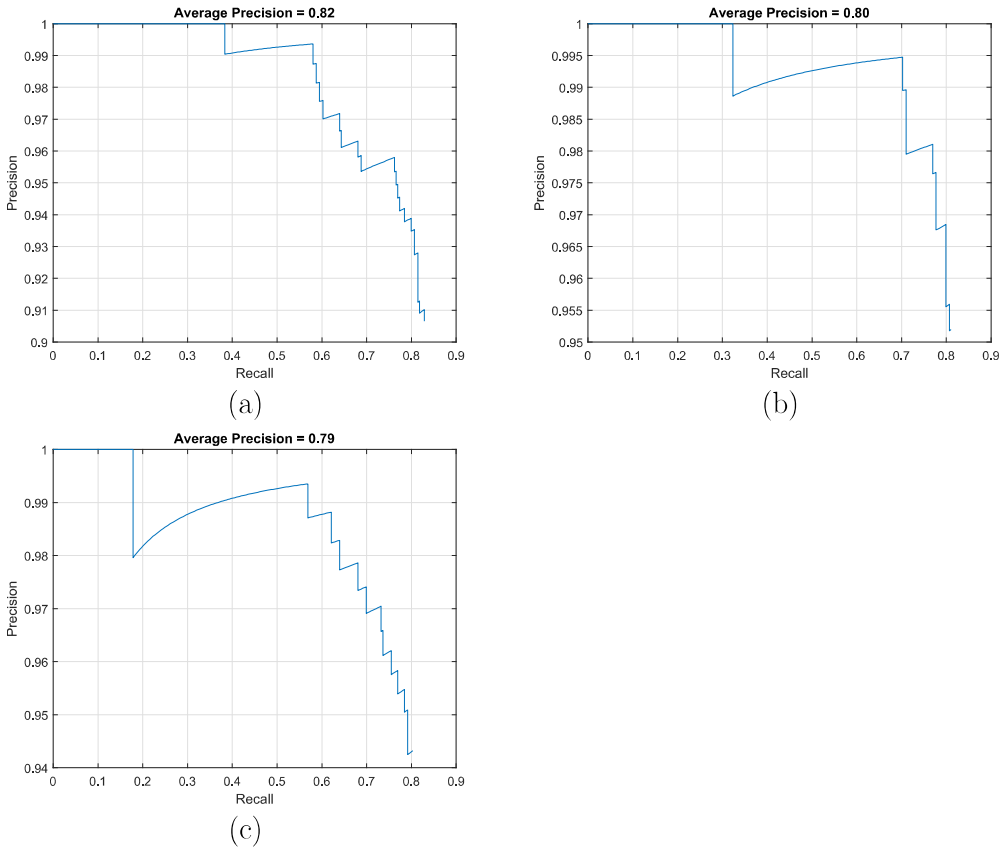

FIGURE 5.15: Average precision for each type of image in detecting the polyp bounding box. (a) RGB has average precision 82% (b) Histogram equalized images give an average precision of 80% (c) Luminace images have an average precision of 79%.

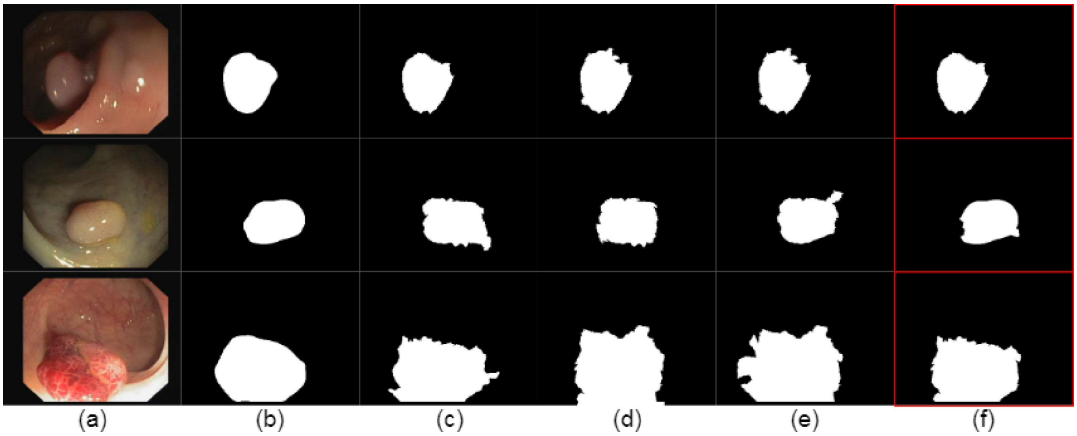

FIGURE 5.16: Results obtained from ablation study. (a) Original Image (b) Ground Truth (c) HEQ_LS (d) LUM_LS (e) RGB_LS (f) ALL.

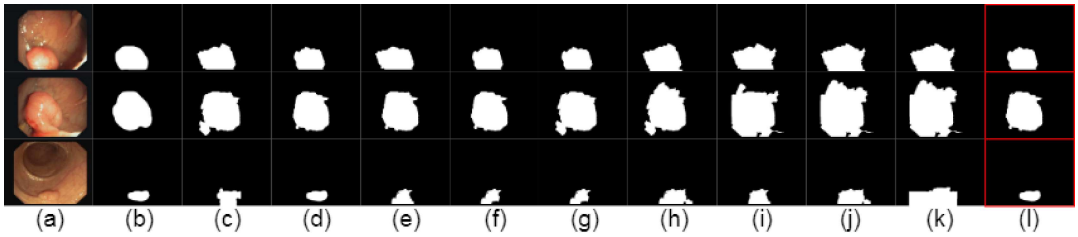

FIGURE 5.17: Results obtained from combination study. (a) Original Image (b) Ground Truth (c) MAX (d) HLR_AND (e) LR_AND (f) HL_AND (g) HR_AND (h) HL_OR (i) HR_OR (j) LR_OR (k) HLR_OR (l) ALL.

5.4.2.7 Combination Study Results

A comparison of the results obtained from the various combination is done to identify, which gives the best result. FIGURE 5.17 gives a visual depiction of the results obtained. ALL gives the best result. HLR_AND also gives comparable results. It is noticeable that all the combinations of OR perform worse than combinations of AND. This can also be deduced by the PR-curve and F-measure plots shown for all the combinations in FIGURE 5.18. ALL performs better than any other combination. The top three best-performing algorithms are ALL, HLR_AND, and HR_AND. The worst performing combinations are HR_OR, LR_OR, and HLR_OR.

5.4.2.8 Discussions

This section presents a discussion of the proposed model. It compares the result of this paper to the one provided by using conditional GANs. Failure cases are also analyzed.

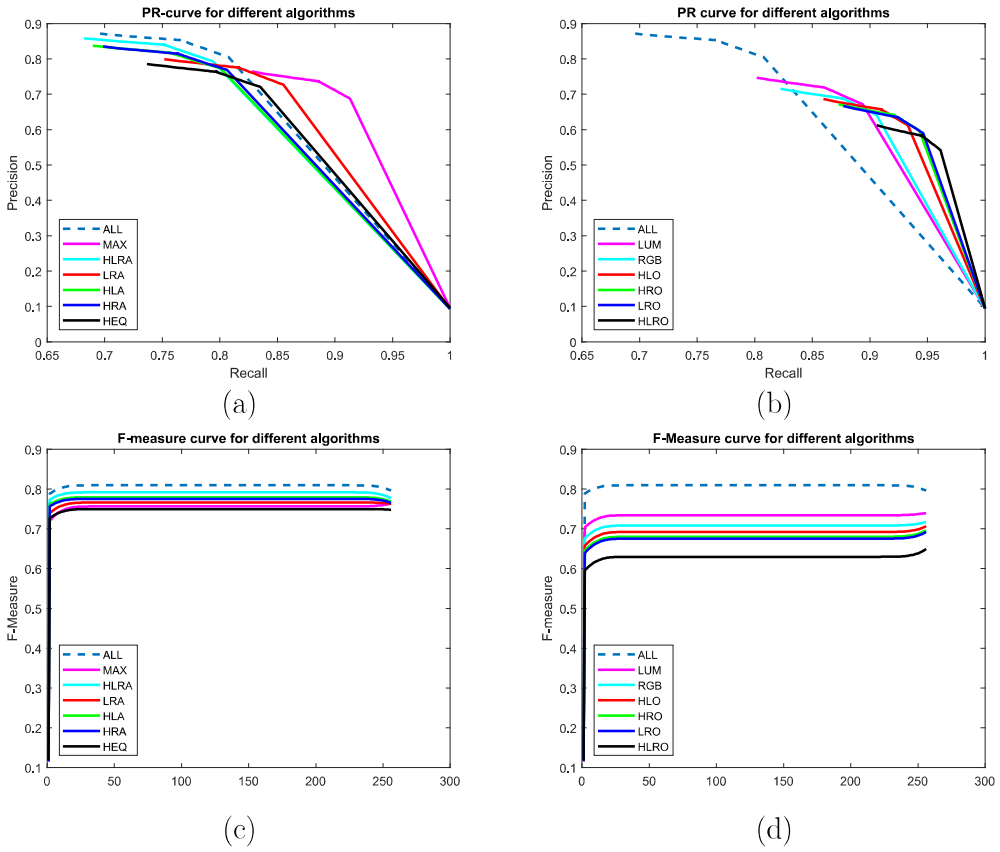

FIGURE 5.18: Performance comparison of various combinations based on PR-curves and F-Measure curves. (a) and (b) show PR-curves (c) and (d) show F-measure curves.

1. **Comparison with state-of-the-art algorithm:** The work of this paper is compared to the results provided by [148]. They have achieved a precision and recall of 85.3% and 68.1%, respectively. It is clear from this result that their algorithm could not cover the entire polyp region, which is a considerable disadvantage. The proposed algorithm shows a precision and recall value of 80.51% and 80.33%, which is marginally higher than [148].

   The output of deep neural network employed in this work takes an approximate of 5.77 seconds, and a lazy snapping procedure takes an approximate

of 0.65 seconds per image. Since three networks are used, and lazy snapping is performed three times in the proposed model, it takes approximately 20 seconds to localize a colon tumor in the image.

2. **Analysis of Failure Cases** There are certain cases in which the proposed model fails to detect the polyp area. The proposed model depends on the bounding box found by the deep network for RGB, HEQ, and LUM images. If, in any case, one of them detects a wrong bounding box, the result will be affected. Some examples are shown in FIGURE 5.19.

   In the first row, it can be seen that HEQ detects two bounding boxes, which leads to the wrong output. In the second case, the HEQ bounding box does not fully cover the polyp region, thus giving incorrect output. In the last row, multiple bounding boxes are provided by the network for RGB image.

   Thus, the final output depends largely on the correct bounding box provided by each of the images. In further work, methods could be found to make the result a little more independent of each input, and other input variants can be tested, which provide more accurate bounding boxes.

5.5 Conclusion

The proposed model was a hybrid approach in two aspects. Firstly, deep learning and conventional methods were used to generate the salient object. Secondly, the proposed model involved a machine as well as user choice in the selection of the

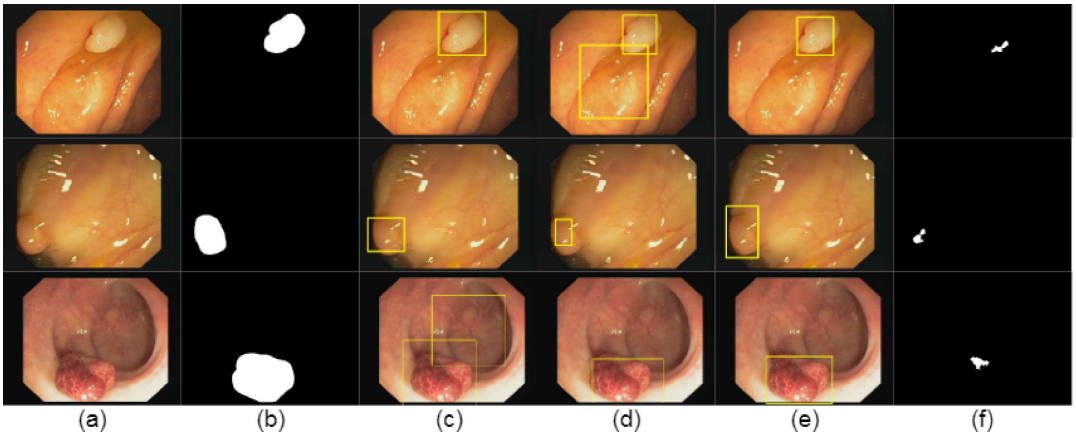

FIGURE 5.19: Examples of failure cases of proposed model (a) Original Image (b) Ground Truth (c) RGB bounding box (d) HEQ bounding box (e) LUM bounding box (f) Final Output.

salient object. This work was a first step in the field of salient object detection, where the proposed model integrated user opinions in building the salient object. Currently, the proposed model has been evaluated only on three datasets. In the future, more varied and complex datasets will be used for evaluation. The proposed model still does not stand firm in front of pure deep learning models. Efforts are to be made in this direction. Also, user opinion was static in this model. The progress of work towards a more dynamic interaction of the user is kept in mind. No evaluation metric is still defined to calculate user-friendliness. Development of a metric for covering this aspect is also of interest.

In the second proposed model, an attempt was made to solve the polyp detection problem in colonoscopy videos. For this, three input variants - RGB images, histogram equalized images, and luminance images - were fed to the deep network, which resulted in polyp bounding boxes. The results were then segmented using

lazy snapping. For combining the results from three bounding boxes, multiple combinations were explored. It was found that the best results were achieved when the proposed model collected only those pixels that occur in all of the segmented results provided by each input variant. The result was compared to a state-of-the-art algorithm in terms of precision and recall. The proposed model performed marginally higher. Failure cases were analyzed, and few possibilities for further improvement of work were given.