

Simple U-net Based Synthetic Polyp Image Generation: Polyp to Negative and Negative to Polyp

Hemin Ali Qadir^a, Ilangko Balasingham^{a,b}, Younghak Shin^{c,*}

^aThe Intervention Centre, Oslo University Hospital (OUS), Sognsvannsveien 20, 0372 Oslo, Norway

^bDepartment of Electronic Systems, Norwegian University of Science and Technology (NTNU), Høgskoleringen 1, 7491 Trondheim, Norway

^cDepartment of Computer Engineering, Mokpo National University, Jeollanam-do, Mu-an-gun, Cheonggye-myeon, Yeongsan-ro, 1666 KR, South Korea

ARTICLE INFO

Article history:

Keywords: Colonoscopy, polyp detection, image synthesis, convolutional neural network, generative adversarial networks

ABSTRACT

Synthetic polyp generation is a good alternative to overcome the privacy problem of medical data and the lack of various polyp samples. In this study, we propose a deep learning-based polyp image generation framework that generates synthetic polyp images that are similar to real ones. We suggest a framework that converts a given polyp image into a negative image (image without a polyp) using a simple conditional GAN architecture and then converts the negative image into a new-looking polyp image using the same network. In addition, by using the controllable polyp masks, polyps with various characteristics can be generated from one input condition. The generated polyp images can be used directly as training images for polyp detection and segmentation without additional labeling. To quantitatively assess the quality of generated synthetic polyps, we use public polyp image and video datasets combined with the generated synthetic images to examine the performance improvement of several detection and segmentation models. Experimental results show that we obtain performance gains when the generated polyp images are added to the training set.

© 2024 Elsevier B. V. All rights reserved.

1. Introduction

Colorectal cancer (CRC) is the third most frequent cause of cancer mortality and the second leading cause of death in men and women globally [1]. Polyps are precursors to colorectal cancer, and resection of polyps can effectively prevent the development of colorectal cancer [2]. Colonoscopy is an important standard tool for colon screening globally. It has the advantage of being able to visualize and remove lesions simultaneously during screening. However, the effectiveness of a colonoscopy depends on the skills and expertise of the endoscopy specialist. Therefore, the polyp detection rate varies widely, and the polyp miss rate is known to be between 6 and 27% in clinical trials [3].

Over the past two decades, many researchers have been working on automatic polyp detection using efficient computer-aided detection systems. Early studies attempted to detect polyp by extracting features such as color, shape, and texture of polyp [4–6]. Recently, attempts have been made to automatically detect and segment polyps using deep learning models which improved performance over traditional feature extraction-based methods [7–12].

Most deep learning-based studies use several public polyp datasets [7, 9–12] for research purposes and focus on developing deep learning model architectures with proper pre- and post-processing methods to improve polyp detection and segmentation performance. One of the difficulties in polyp detection and segmentation is the diversity of their features. Polyps vary widely in color, shape, size, and texture. However, widely used polyp datasets only have dozens of unique polyps used for deep learning-based detection and segmentation model training

*Corresponding author: email: shinyh0919@gmail.com

[13–15]. Because in medical fields, abnormal data is inherently more difficult to collect than normal data. To obtain the location information required for polyp detection and segmentation, manual annotations by colonoscopy experts are essential and are usually costly and logistically difficult.

In order to overcome the limitation of availability of large polyp image data, many deep learning-based studies are actively using transfer learning and image augmentation techniques [5, 9]. However, to effectively train a deep learning model with a large number of learnable parameters, it is necessary to increase the diversity of polyp appearances within the training dataset.

Recently, a GAN (generative adversarial network)-based image generation approach has also been applied to polyp data to overcome insufficient training datasets [16, 17]. In [16], GAN based domain adaptation approach has been applied to the Endoscopy dataset. They focused on adapting real data to synthetic-like data to remove patient-specific details from real images. This is the difference from our study, which aims to generate new synthetic polyp images to improve polyp detection and segmentation performance.

In [18], to generate a synthetic polyp, a polyp mask is first generated using a GAN framework, and the polyp part is generated when the polyp mask is given using the conditional GAN approach. Then, the polyp-free part is found in the real polyp image and the generated polyp is synthesized. The difference from the proposed approach is that two GAN frameworks should be used for polyp generation. In addition, a separate algorithm for each image is required to find the polyp-free part, and another separate algorithm is used to naturally synthesize the generated polyp part into the existing image. On the other hand, the proposed method uses a single conditional GAN framework to convert a polyp image into a negative image and convert the negative image back to a polyp image. Therefore, there is no need to use a separate algorithm to find a polyp-free part or perform natural synthesis.

In [17], a conditional GAN was used to generate synthetic polyps. The authors proposed to use a conditional mask to control the position, size, and shape of generated polyps and showed improved polyps detection performance when the generated polyps were added as additional training data. However, negative images must be used to generate the polyp image, and additional processing (i.e., edge filtering) was required to create the input condition image. Furthermore, it is not possible to generate polyps with various characteristics from the same input condition. From a performance verification point of view, the performance improvement was weakly evaluated, i.e., only one deep learning model was evaluated for the polyp detection task.

In this paper, to overcome the shortcomings of the previous polyp image generation work, we conduct a study that generates synthetic polyp images similar to real ones using only a given polyp dataset without the use of additional datasets and processes. We propose a framework that uses the same GAN architectures to convert the polyp images into a negative image and the negative image back into a novel polyp image. In addition, we attempt to generate various polyp features by in-

troducing a controllable polyp mask with one input condition image. The experimental results show performance improvement when synthetic polyp images generated by the proposed method are used for training various deep learning models for object detection and segmentation.

The remainder of this paper is organized as follows. In Section 2, the proposed polyp to negative and negative to polyp image generation framework including network architecture and preparation of input conditioned images are introduced. We also explain the polyp detection and segmentation models used for the qualitative evaluation of generated polyp images. In Section 3, experimental datasets used in this study are described. In Section 4, experimental results and discussions are described. Finally, we conclude this study in Section 5.

2. Methods

Figure 1 shows the concept of the proposed synthetic polyp generation procedure. Using a given polyp image and its corresponding label mask, we first transform the polyp part to the negative part. Then, we transform the generated negative image into a new synthetic polyp image using a controllable input condition mask. The controllable mask is used to force the generator network to produce polyps with different characteristics depending on the color of the polyp mask in the input condition image (see Section 2.3). For both transformations, we use the same network architectures. We explain each method in the following subsections in detail.

2.1. Network Architectures

Figure 2 shows the structure of the generator and discriminator networks used for the conditional GAN framework in this study. Recent GAN studies have focused on proposing new network architectures to obtain task-specific high performance image generation [17], image inpainting [19] and image detection and segmentation [12, 20]. On the other hand, we adopt a simple U-net based generator architecture used in [21] to make it universally work for two different tasks (e.g., converting from polyp to negative and from negative to polyp). The difference from the paper [21] is that a multi-patch discriminator is used in the discriminator and WGAN-GP loss [22] is adopted as the GAN loss function. Detailed explanations are provided in the following paragraphs.

As shown in Figure 2, for the input image size of 256×256 , 4×4 convolution with stride 2 is applied in each encoding layer, so that the feature map size is reduced to $1/2$ and then progresses to the final 1×1 feature map. In each subsequent decoding layer, transposed convolution is applied to increase the feature map size by 2 and a final 256×256 image is generated. In the encoder part of the generator, we use Convolution-BatchNorm-leakyReLU module in each layer while in the decoder part, we use transposed Convolution-BatchNorm-ReLU module. Dropout is used in the first 3 layers of the decoder part with a dropout rate of 50%.

In the case of the discriminator, a patch-based discriminator which is widely used for image-to-image transformation tasks is applied [17, 21]. Additionally, to reflect the diversity of polyp

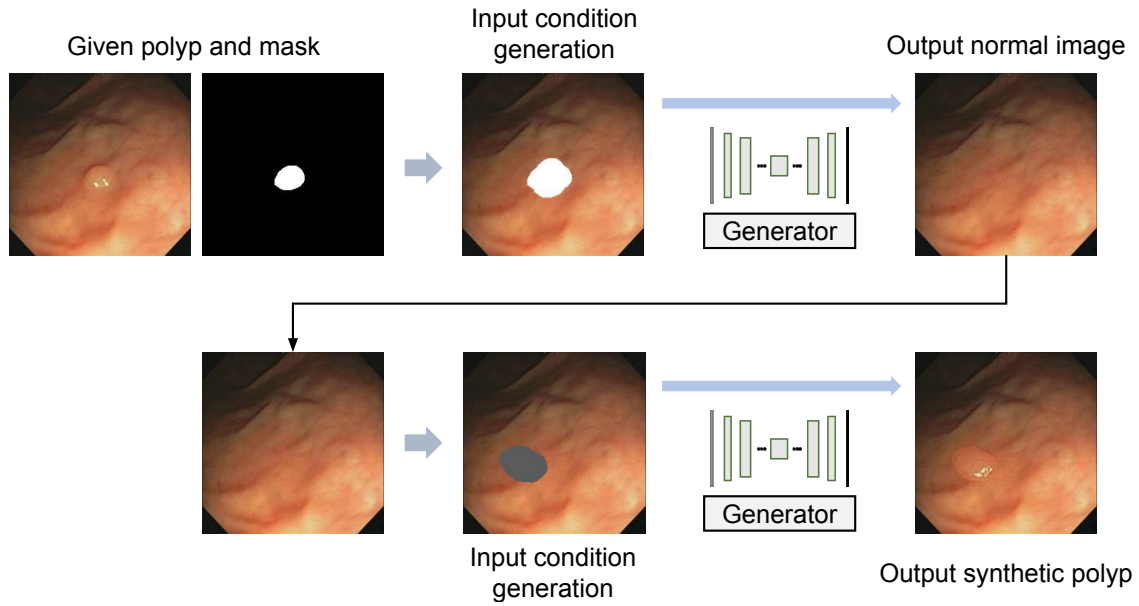


Fig. 1: Proposed conditional GANs based polyp image generation framework.

sizes well, a multi-patch discriminator was used that utilizes two output patches at the same time as shown in Figure 2. In the discriminator, the Convolution-BatchNorm-leakyReLU module was applied to the input image in the same way as the encoding layer of the generator, and the final feature map size is 16×16 as shown in Figure 2. For the multi-patch discriminator, GAN loss for 64×64 and 16×16 feature maps is added.

Our training loss is comprised of a GAN term and a reconstruction (Reconst) term which are commonly used in the existing methods [17, 21]:

$$\arg \min_G \max_D L_{GAN}(G, D) + \lambda L_{Reconst}(G), \quad (1)$$

where G refers to the generator network, D refers to the discriminator network, and λ is a parameter to control the balance between two different loss terms. In the case of the GAN loss term, we use the WGAN-GP loss [22], which is known to improve training stability and has been recently used in many GAN studies [22, 23]. For reconstruction loss, we use the L1 loss between the generated image and the original target im-

age [21]. L1 loss is needed to model low-frequency structure (blurry results) because the GAN loss can only model high-frequency structure and give much sharper results. The framework can introduce virtual artifacts when the GAN loss is used alone [21]. When both losses are used together, they lead the framework to generate realistic images to fool D .

2.2. Polyp Image to Negative Image Generation

To generate a new polyp image, we first train our conditional GAN model to transform a given polyp image to a new negative image. To do this, we need to prepare a pair of training images, i.e., input conditioned image and output target image as shown in Figure 1.

Each row in Figure 3 is an example of a pair of images used for training polyp to negative image transformation. An input condition image (shown on the left side in Figure 3) is created from combining a polyp training image and a generated mask. The original polyp image (shown on the right side in Figure 3) is used as an output target image. Thus, the proposed conditional GAN trains an image inpainting task that targets generating the

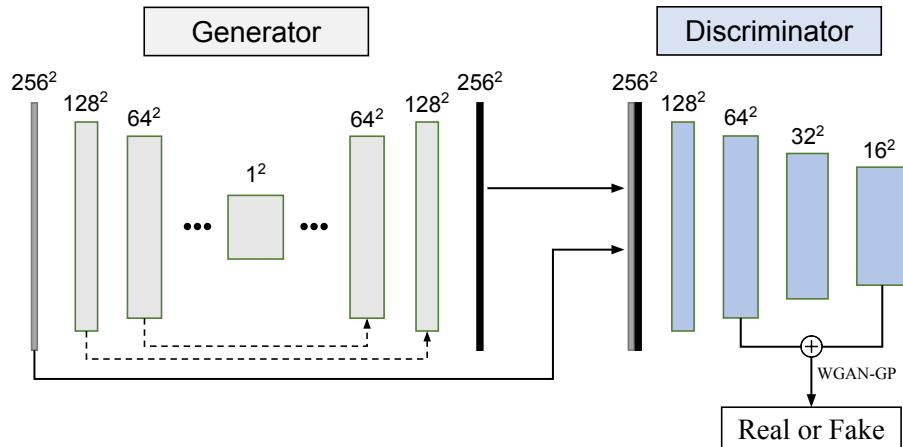


Fig. 2: Proposed U-Net based generator and discriminator architectures. The same architecture is used for both polyp to negative and negative to polyp tasks.

negative parts in the right images by using the masks in the left images in Figure 3 as its condition.

To create the masks shown in the left images in Figure 3, we apply basic image augmentation such as rotation, scaling, position translation, and perspective transform with randomly selected parameters to the original polyp masks in the training dataset. Since the network will be trained to reconstruct the negative image portion only, the generated masks are randomly placed in the input condition images, avoiding overlapping with the polyp portion.

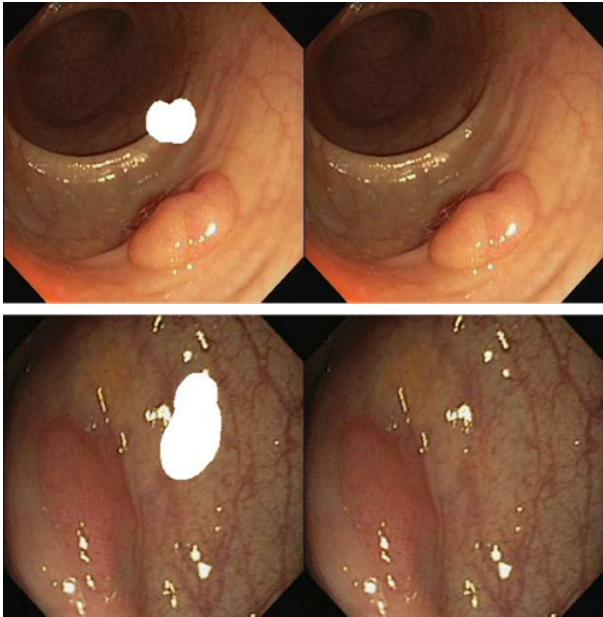


Fig. 3: Training image pairs for polyp to negative image generation. Left: input condition image, Right: output target image.

In the inference phase, the goal is to transform the polyp region in a given polyp image to a negative region. The polyp image is combined with its corresponding polyp mask to create the input condition image. However, if we use the original polyp mask, shadows or connections around the polyp will remain as shown in Figure 4(b). This shadow will appear as an artifact and prevent generating a realistic negative image. To avoid this issue, we dilate the original polyp mask by 10 pixels (as shown in Figure 4(c),) as the final mask for the input condition in the inference.

2.3. Negative Image to Polyp Image Generation

Next, we prepare training data for the final objective, the negative image to polyp image transformation task. Figure 5 shows two pairs of training images used for the negative image to polyp image transformation. The input condition image is generated from combining the polyp images with their corresponding polyp masks provided with the datasets. Figure 5 shows two pairs of training images: the input condition image is shown on the left side, and the output target shown on the right is the polyp image. We aim to train the generator network to regenerate the same form of polyp in the given mask region.

In the previous image to image translation method [21], there is a shortcoming in that the same image is always generated for

the same input condition. Also, in [17], there is a shortcoming in that the characteristics of the generated polyps for various input conditions are very similar. The reason is that a single binary polyp mask was used for various polyps. To overcome this problem, we try to use a controllable input condition. It is intended to generate a new polyp by adjusting the mask values even if the the shape of the mask in the input condition is the same shape (see the left side in Figure 5).

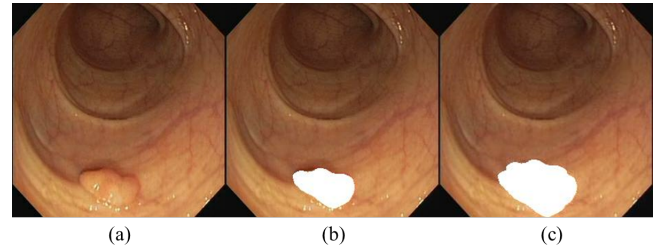


Fig. 4: Example of input conditioned image for negative image generation in inference. (a) original polyp image (b) original polyp image combined with original polyp mask (c) original polyp image combined with extended polyp mask.

The dataset of polyp images used for training comprises 34 different unique polyps (see the information of CVC-ClinicDB dataset in section 3). We allocate 34 different values evenly distributed between 0 and 255 rather than traditional binary masks for the polyp regions in the training images. Therefore, the value of the polyp masks varies depending on the polyp in the given image as shown on the left side of Figure 5, meaning a different gray-scale value is assigned for each unique polyp in the dataset.

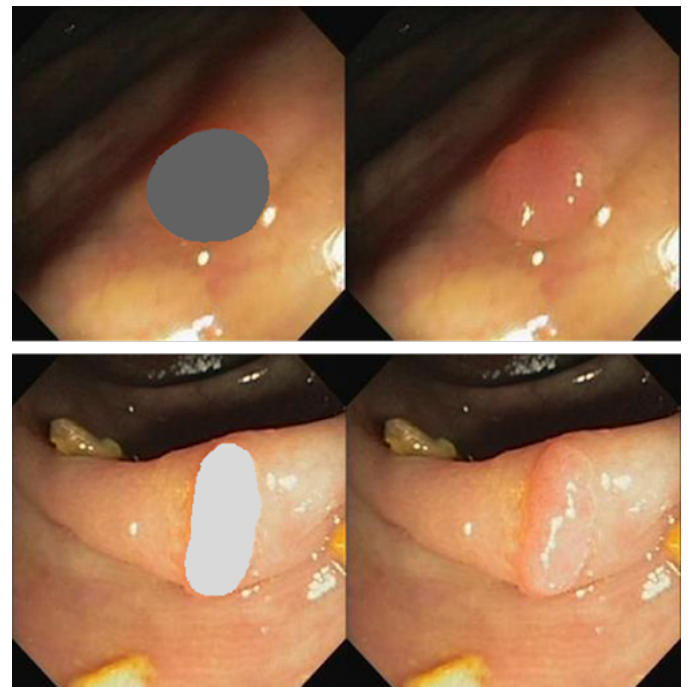


Fig. 5: Training image pairs for negative to polyps image generation. Left: input condition images, Right: output target images (original polyp images).

By controlling the value of the polyp masks in the input condition images, we can generate a polyp with different features and characteristics from the same input condition of the same

shape at the inference phase, see Section 4 for detailed generation results. When inferring the final polyp generation, the input condition image is produced by combining a label mask, which is created from a randomly generated shape and selected value, with a negative image, which is obtained by the polyp to negative image transformation model.

2.4. Polyp Detection and Segmentation Models

For quantitative evaluation of the generated polyp images, we assess the performance of several polyp detection and segmentation models when trained with and without generated synthetic polyps. In more detail, performance evaluation is performed on several detection and segmentation models trained only with the data given as the original training set and the models trained with the generated polyp images added to the original training set.

For polyp detection, we select three well-known detection models: Faster R-CNN ResNet101, Faster R-CNN Inception Resnet v2 [24] and R-FCN ResNet101 [25]. Faster R-CNN utilizes a pre-trained deep CNN as a feature extractor and uses the last convolution layer as a region proposal network (RPN) [24]. Then, the classification and regression are performed on the RoIs (Region of Interest) proposed by the RPN. We use two deep CNN models i.e. ResNet101 and Inception Resnet v2, as the feature extractors pre-trained on Microsoft's (MS) Common Objects in Context (COCO) datasets.

R-FCN has been proposed to effectively overcome the slow training and inference speeds occurring during RoI-specific processing of the existing Faster R-CNN [25]. R-FCN implements the position-sensitive score maps technique that can include information about the location of objects while sharing the operation of RoIs extracted from RPN using a fully convolutional layer. Resnet101 pre-trained on MS COCO dataset is used as the feature extractor of R-FCN.

We also use three segmentation models to qualitatively evaluate the generated synthetic polyps when used as additional training data. All three models are based on encoder-decoder architecture of the U-Net family: TerausNet-16 [20, 26], AlbuNet-34 [20], and MDeNetplus [12]. U-Net is a widely used segmentation model in the field of medical imaging where training data is limited [27]. A U-Net architecture consists of an encoder-decoder structure with skip connections. The skip connections enable feature combinations of the encoder layers and decoder layers, which allows for sophisticated localization. The TerausNet-16 model uses ImageNet pre-trained VGG-16 for the encoder network, while the AlbuNet-34 uses ImageNet pre-trained ResNet-34 as an encoder and improves the method of skip-connections from U-Net [20]. MDeNetplus has not only skip connections from the encoder layers to decoder layers, but also feedback connections. The feedback connections sum the activation maps of similar layers of different decoders.

2.5. Model Training

We train the GAN network from scratch and the Adam optimizer with a momentum of 0.5 at a learning rate of 0.0002. Before training, the input images of 256×256 are resized to

312×312 and then randomly cropped back to 256×256 for applying random jittering [21]. For other parameters in model training such as learning rate, batch size, and weight for reconstruction loss, we refer to the values which were used in the image to image translation work [21].

For training of detection models, i.e., Faster R-CNN ResNet101 and Inception Resnet v2 and R-FCN ResNet101, we use the Tensorflow object detection API [28]. The stochastic gradient descent (SGD) method with a momentum of 0.9 is used as an optimizer. In each iteration of the RPN training, 256 training samples are randomly selected from each training image and the ratio between positive ('polyp') and negative ('background') samples is 1:1. For all other parameters such as learning rate, non-maximum suppression (NMS), and the maximum number of proposals, we use the default values which were used in the original Faster R-CNN work [24]. Each segmentation model is trained with an Adam optimizer with a learning rate of 0.0001 and batch size of 5 for a different number of epochs to achieve the maximum performance for each model. During training, the training dataset was randomly split into training and validation subsets using 5-fold cross-validation. The 5-fold cross-validation method was performed to choose the best hyper-parameters for the models (e.g. the learning rate relative to the batch size, best epoch, etc) and thus avoid overfitting. The size of the images was changed to 512×512 . Finally, we use Jaccard loss combined with a pixel-wise binary cross-entropy to optimize the parameters of the models

3. Experimental Datasets

We use three publicly available polyp datasets, CVC-ClinicDB [13], ETIS-LARIB [14] and CVC-ClinicVideoDB [15]. The CVC-ClinicDB image dataset is used for the whole procedure of the polyp image generation task, i.e., model training for negative and polyp image generation. It is also utilized for training detection and segmentation models as a baseline original image set which is compared to those of the original plus generated synthetic image set. The CVC-ClinicVideoDB and ETIS-LARIB image dataset are used for detection and segmentation model tests, respectively.

The CVC-ClinicDB image dataset consists of 612 polyp image frames with corresponding 612 binary polyp mask images. The mask images were annotated by skilled clinicians. The 612 polyp image frames have a pixel resolution of 388×284 pixels in SD (standard definition). These images are extracted from 31 colonoscopy videos which contain 34 unique polyps. The CVC-ClinicVideoDB dataset contains 18 SD (384×288 pixels) videos. In this dataset, a total of 10025 frames out of 11954 frames contain a polyp. For each image in the video database, an approximate circle-shaped binary polyp mask is given which makes the dataset suitable for the evaluation of performance improvement the polyp detection models and benchmark our results with the existing studies. For the evaluation of detection models, we use the whole 11954 frames as a test set.

The ETIS-LARIB dataset comprises 196 still images extracted from 34 colonoscopy videos. The images have an HD (high definition) 1255×966 pixels resolution. In this dataset, 44

different polyps are presented in 196 images. A binary polyp segmentation mask annotated by an experienced clinician is provided for each image. This dataset is suitable for the evaluation of performance improvement of the segmentation models because the polyp masks are polygon shapes drawn around the polyp boundaries.

4. Results and Discussion

4.1. Generated Negative Images

In this work, we first transform the given polyp image into a negative image and then create a new polyp in the generated negative image using the polyp condition. In both image translation tasks, i.e., polyp to negative and negative to polyp, we use the same proposed conditional GAN architectures.

Figure 6 shows some results of the generated negative images from polyp to negative translation task. In each column of Figure 6, a different generated negative image is represented in (c) which is created from images presented in (b) as the input condition image and (a) is the corresponding original polyp image. Thus, from this task, we transform polyp image (a) to negative image (c). As we can see in Figure 6, the generated negative images not only clearly remove the corresponding polyp in the polyp portion given by input condition (b), but also naturally harmonize the removed and surrounding parts. We can see that it produces natural negative images even for the third and fifth columns, which are relatively large polyps.

Figure 7 is an example of generated negative images with large texture changes in the part of the binary condition mask. As we can see in the generated negative images (c) of Figure 7, our trained model adaptively creates a realistic negative colon image by applying the shape and texture of the surrounding

parts of the polyp mask. As a result, we can see that capillaries or folds of colon in the generated negative images in the blue circle part of (c).

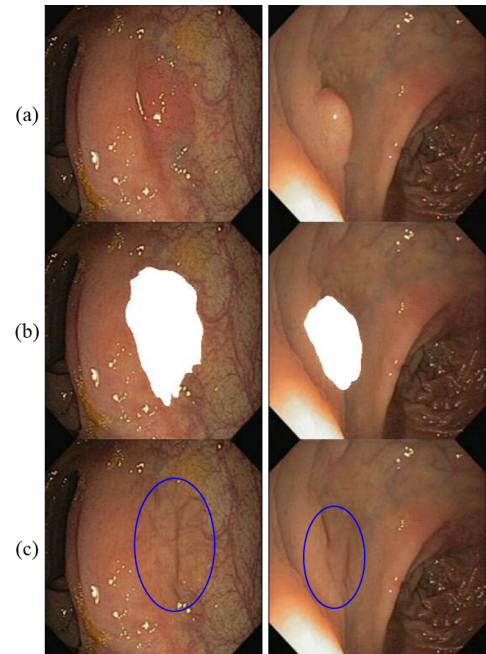


Fig. 7: Example of images with large texture changes in the corresponding polyp mask part among the generated negative images. (a), (b) and (c) represents original polyp image, input condition and generated negative image.

4.2. Generated Polyp Images

Our main goal is to create new polyp-looking images. For this purpose, the negative to polyp translation is performed using the previously mentioned method. Figure 8 shows several examples of negative to polyp translation and each column

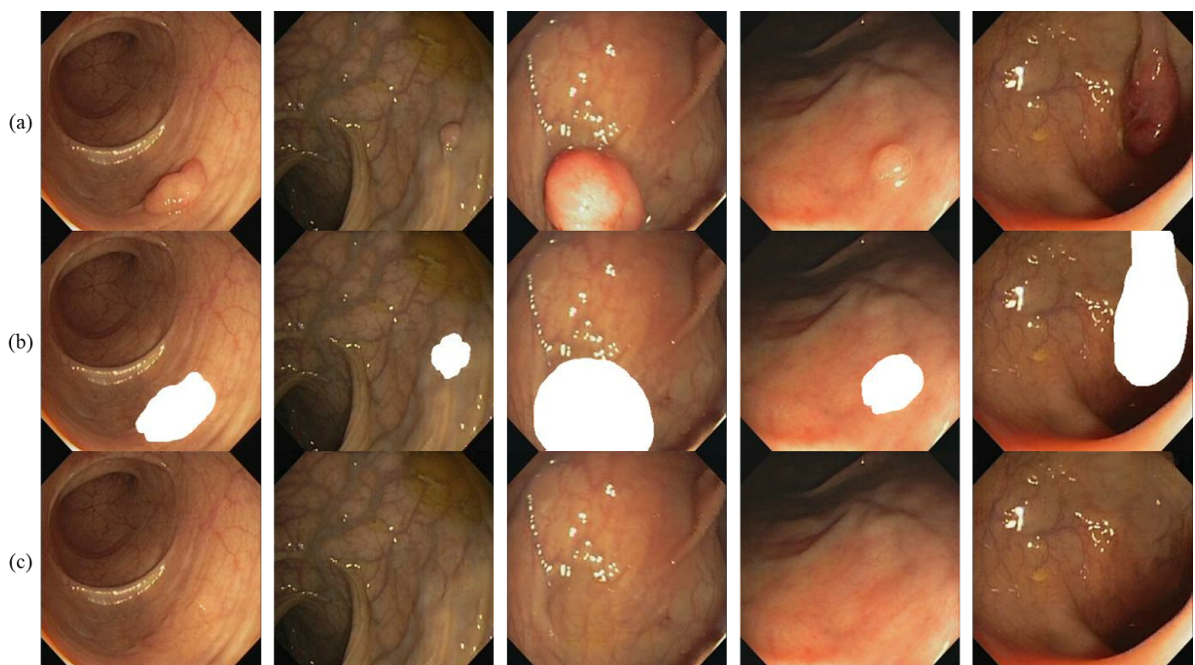


Fig. 6: Some results of the polyp to negative translation. Each column represents different image generation result. (a) is polyp image used for preparing the corresponding input conditioned image (b). (c) is generated negative image from the corresponding input condition (b).

shows a different image result. In this experiment, we use the generated negative image (a) for the preparation of the input condition image (b). Then, our trained model, i.e., negative to polyp model generated output polyp image (c) using the (b) as the input condition.

As mentioned in Section 2.3, in the input condition image (b) of Figure 8, the value of the polyp mask is randomly selected among 34 different pixel values representing different gray-scale colors. The resulting images in Figure 8 (c) show that the position and shape of the generated polyp are controlled by the polyp mask of (b). In addition, it can be seen that various gray-scale colors and textures are generated rather than similar characteristics, which was a challenge in the existing synthetic polyp generation study [17]. This is because different pixel values are used as input conditions for the polyp mask part.

To analyze this in detail, we have generated synthetic polyp images by varying the pixel value of the polyp mask part in the same input condition image in Figure 9. Each row in Figure 9 is an example of a polyp image generation result when only the pixel value of the polyp mask part varies for the same input condition. In other words, each column is generated from different input conditions with different polyp mask values.

As it can be seen from the results of each row in Figure 9, even with the same shape of polyp mask and background image, it is possible to create polyps with various characteristics in terms of color or texture by controlling the pixel value of the polyp mask. If we look at each column in Figure 9, we can confirm that the model learns to generate synthetic polyps depending on the polyp mask value rather than the shape or the background image. Because even if different background images and shapes are used, the model can generate polyps with similar characteristics. These two experiments confirm that diverse image generation using the same input, a limitation of the existing image to image transform study [17, 21], can easily be

achieved by adjusting the mask pixel value of the input condition rather than changing the network structure or training loss. This methodology makes it possible to control the generation of synthetic polyps with characteristics that are difficult for the model to detect, or polyps with clinically malignant characteristics that should not be overlooked.

Furthermore, we measured the inference speed of the generator network after being trained within the Conditional GAN framework used in this paper. As a result, the average inference time was measured to be 51.33 msec on an NVIDIA RTX 2080Ti GPU.

4.3. Evaluation of Polyp Detection Performance

The advantage of synthetic polyp generation based on the proposed input condition mask is that the generated polyp image can be used directly as a training image for polyp detection or segmentation task without mask annotations. This is significant because it can reduce the manual labeling cost of experts in mask annotation of medical data.

For evaluation of generated polyp images, we first assess polyp detection performance when generated polyp images are used as additional training samples. As polyp detection models, the Faster R-CNN ResNet101, Faster R-CNN Inception ResNet v2, and R-FCN ResNet101 models mentioned in section 2.4 are used. To define evaluation metrics for polyp detection, it is important to specify and compute the following medical terminologies: true positive (TP), false positive (FP), false negative (FN), and true negative (TN) where:

TP = detection output within the polyp ground truth.

FP = any detection output outside the polyp ground truth.

FN = polyp not detected for positive (with polyp) image.

TN = no detection output for negative (without polyp) image.

If there is more than one detection output, only one TP is counted per polyp. Based on the above parameters, we de-

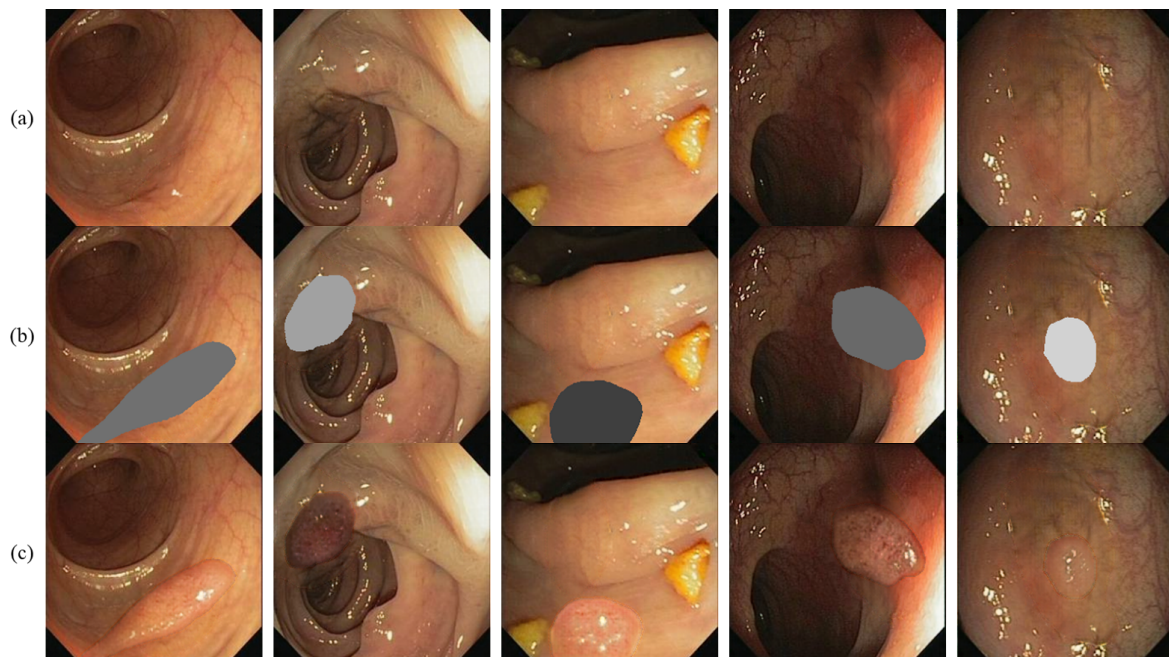


Fig. 8: Some results of the negative to polyp translation. Each column represents a different image generation result. (a) is generated negative image from polyp to negative translation. (b) and (c) is the input conditioned image and the corresponding generated polyp image.

fine three performance evaluation metrics, precision (Pre), recall (Rec) and, f1-score (f1):

$$\begin{aligned} Pre &= \frac{TP}{TP + FP} \times 100, \quad Rec = \frac{TP}{TP + FN} \times 100, \\ f1 &= \frac{2 \times Rec \times Pre}{Rec + Pre} \times 100, \end{aligned} \quad (2)$$

For evaluation of polyp detection performance, the CVC-ClinicVideoDB dataset (see Section 3 for details) is used as the test dataset.

Table 1 lists the polyp detection performance for three different models. The original columns show the performance when the models are trained with The CVC-ClinicDB image dataset (612 polyp images), which is the originally given training dataset. The paper [17] columns show the detection results when the models are trained on the original training dataset plus additional 372 synthetic polyp images generated by the method proposed in paper [17]. The combined columns show the performance when the models are trained by adding 350 polyp images generated by the proposed method in this paper to the original training dataset.

Table 1: Comparison of polyp detection performance for the three detection models using the original training set and combined training set by generated polyp image. Number of samples is: Original (612); [17] (612+372); Combined (612+350)

Models	Faster R-CNN ResNet101						
	TP	TN	FP	FN	Rec	Pre	F1
Original	6047	1431	1513	3978	60.32	79.99	68.76
[17]	5370	1603	1049	4655	53.57	83.66	65.31
Combined	6263	1508	991	3762	62.47	86.34	72.49

Models	Faster R-CNN Inception ResNet v2						
	TP	TN	FP	FN	Rec	Pre	F1
Original	6011	1496	1333	4014	60	81.9	69.22
[17]	6831	1399	1177	3194	68.1	85.3	75.74
Combined	7056	1351	1212	2969	70.38	85.34	77.14

Models	R-FCN ResNet101						
	TP	TN	FP	FN	Rec	Pre	F1
Original	5762	1304	2062	4263	57.48	73.65	64.56
[17]	5554	1653	809	4471	55.4	87.29	67.78
Combined	6555	1596	1032	3470	65.38	86.39	74.43

The results in Table 1 show that the use of generated images (Combined) demonstrates better polyp detection performance for all three models than the use of the original dataset only. For the three models, both recall and precision increased. This means that polyps missed by the model trained with the existing training dataset are improved when the combined dataset is used. For example, when training using the combined set based on the Faster R-CNN Inception ResNet v2 model, 1045 additional polyps are detected compared to the polyps detected by the original set training. In addition, when the polyp images generated by the proposed method are used for training, improved performance is demonstrated in all three models based on f1-score compared to using the polyp image generated by the existing method [17].

In [18], the same dataset as of this study was used for the evaluation of polyp detection performance, i.e., the detection

model was trained on the synthetic polyp generated by their proposed method together with the CVC-ClinicDB training data. They evaluated the detection performance of Faster R-CNN ResNet50 on the CVC-ClinicVideoDB. For comparison, we measure the detection performance for the same model using a dataset generated by our proposed method (i.e., Combined). Table 3 summarizes the detection performance of Faster R-CNN ResNet50 for both methods and shows that our method obtains better results in terms of Precision and F1-score while their model obtained better Recall.

Table 2: Comparison of polyp detection performance of Faster R-CNN ResNet50 between our results and the results shown in [18]

Models	Faster R-CNN ResNet50						
	TP	TN	FP	FN	Rec	Pre	F1
[18]	7378	-	6313	2647	73.5	53.8	62.1
Ours	6597	1692	859	3428	65.8	88	75.3

4.4. Evaluation of Polyp Segmentation Performance

To quantitatively evaluate the effectiveness of the generated synthetic polyps in improving the performance of the segmentation models, we apply commonly used evaluation metrics: Jaccard index (J) also known as intersection over union (IoU) and Dice similarity score (D) as follows:

$$\begin{aligned} J(A,B) &= \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}, \\ Dice(A,B) &= \frac{2|A \cap B|}{|A| + |B|}, \end{aligned} \quad (3)$$

where A refers to a generated output by the models and B is the corresponding ground-truth of A .

Table 3 demonstrates performance evaluation of the three segmentation models where two scenarios in terms of training data, i.e., original and original plus 350 synthetic polyp images (combined training), are used to train the models. As it can be seen from the table, the performance of the three segmentation models improves when the generated synthetic polyps are added to the original training data as follows: The performance of TernaNet-16 improved by 6.44% Jaccard and 9.66% Dice, and AlbuNet-34 by 4.11% Jaccard and 5.87% Dice, and MDeNetplus by 5.27% Jaccard, 6.47% Dice.

Table 3: Comparison of polyp segmentation performance for three different models using original training set and combined training set by generated polyp image.

Models	TernaNet-16		AlbuNet-34		MDeNetplus	
	Jaccard	Dice	Jaccard	Dice	Jaccard	Dice
Original	35.47	43.18	46.75	56.98	45.77	56.53
[17]	37.86	47.57	50.48	60.31	50.49	60.77
Combined	41.91	52.84	50.86	62.85	51.04	63

In Table 2, we use the same segmentation models to compare the improvement capability of the synthetic polyps generated by this work and the synthetic polyps generated by the method presented in [17]. For all three models, the synthetic polyps

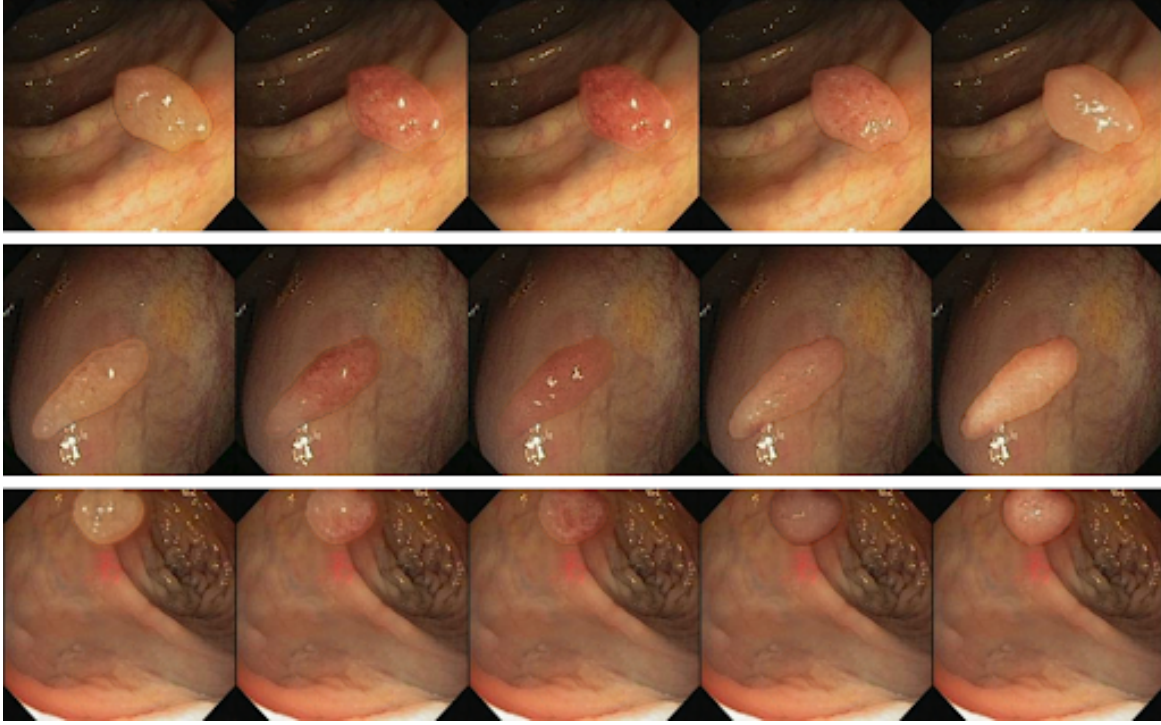


Fig. 9: Example of generated polyps. The images in each row are generated when only the pixel value of the polyp mask is changed under the same input condition..

of the current study show a better result improvement than the synthetic polyps of [17] study. This might be because we are capable of controlling the generation of synthetic polyps with various shapes and features from the same input condition image. As a result, with the proposed method, polyps with various characteristics can be generated. This is a difference from [17] in which only the shape of the polyp can be controlled.

4.5. Limitations and Future Work

One natural question might be how performance changes as the number of generated polyps increases. In Figure 10, the detection performance of the RFCN ResNet101 model is evaluated when different numbers of synthetic polyp images are added to the original training dataset. As it can be seen that the model starts gradually improving its performance when it is exposed to additional synthetic polyp images. However, the model reaches a saturation performance improvement after a certain point even if more synthetic polyp images are added to the training set. This is due to the limitation of this method that is unable to introduce new unseen features. This method only manipulates the existing features in the training dataset used to train the GAN model and tries to reuse the same set of features to generate new-looking synthetic polyps. In other words, the proposed framework cannot improve data distribution and is unable to add new features to the training data, it only leads to an image-level transformation and feature-level manipulation. This is a common limitation of GAN-based image generation models.

For future work, studies that generate polyps with new characteristics not included in the training data will be needed. Performance can further be improved if characteristics of two/multiple polyps can be fused into a single synthetic polyp.

This could be done by fusing obtained abstract features from different polyps or by manipulating the features by adding random noise. Both approaches could be performed in the latent space. Alternatively, we can apply a method such as CAN (Creative Adversarial Networks) [29] that adjusts the loss term of GAN to generate a creative image by maximizing deviation from established styles and minimizing deviation from the distribution of several datasets.

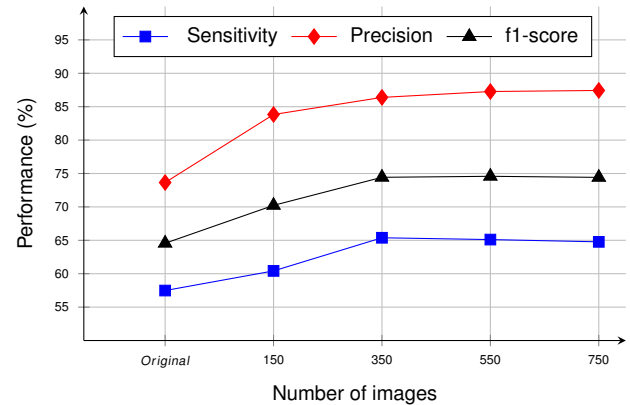


Fig. 10: Performance improvement with different number of synthetic polyp images added to the training data

5. Conclusion

It is expensive to acquire images of various types of polyps that can be used for training deep learning based automatic polyp detection and segmentation models. We proposed a synthetic polyp image generation framework based on the conditional GAN architectures. In the proposed framework, we converted a given polyp image into a negative image and then the

negative image back into a new-looking synthetic polyp image using the same networks for both tasks. In the previous polyp generation studies, a single value was assigned to the polyp mask in the condition input image. This led the models to face difficulties to generate various polyps with different characteristics. In this study, we attempted to overcome this shortcoming. We developed a framework to generate various polyps with different features by controlling the value of the polyp masks in the input condition images. The experimental results showed that the proposed framework could generate polyps with various characteristics similar to the real ones. In addition, it was confirmed that the polyp detection and segmentation performance could be improved when the generated synthetic polyp images were used to train several detection and segmentation models.

Acknowledgements

This research was supported by the Research Funds of Mokpo National University in 2020. The authors would like to thank Jacob Bergsland at Oslo University Hospital for his valuable comments.

References

- [1] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, A. Jemal, Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries, *CA: a cancer journal for clinicians* 68 (6) (2018) 394–424.
- [2] A. G. Zauber, S. J. Winawer, M. J. O'Brien, I. Lansdorp-Vogelaar, M. van Ballegoijen, B. F. Hankey, W. Shi, J. H. Bond, M. Schapiro, J. F. Panish, et al., Colonoscopic polypectomy and long-term prevention of colorectal-cancer deaths, *N Engl J Med* 366 (2012) 687–696.
- [3] S. B. Ahn, D. S. Han, J. H. Bae, T. J. Byun, J. P. Kim, C. S. Eun, The miss rate for colorectal adenoma determined by quality-adjusted, back-to-back colonoscopies, *Gut and liver* 6 (1) (2012) 64.
- [4] S. A. Karkanis, D. K. Iakovidis, D. E. Maroulis, D. A. Karras, M. Tzivras, Computer-aided tumor detection in endoscopic video using color wavelet features, *IEEE transactions on information technology in biomedicine* 7 (3) (2003) 141–152.
- [5] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, J. Liang, Convolutional neural networks for medical image analysis: Full training or fine tuning?, *IEEE transactions on medical imaging* 35 (5) (2016) 1299–1312.
- [6] Y. Shin, I. Balasingham, Automatic polyp frame screening using patch based combined feature and dictionary learning, *Computerized Medical Imaging and Graphics* 69 (2018) 33–42.
- [7] J. Bernal, N. Tajbakhsh, F. J. Sanchez, B. J. Matuszewski, H. Chen, L. Yu, Q. Angermann, O. Romain, B. Rustad, I. Balasingham, et al., Comparative validation of polyp detection methods in video colonoscopy: results from the miccai 2015 endoscopic vision challenge, *IEEE transactions on medical imaging* 36 (6) (2017) 1231–1249.
- [8] L. Yu, H. Chen, Q. Dou, J. Qin, P. A. Heng, Integrating online and offline three-dimensional deep learning for automated polyp detection in colonoscopy videos, *IEEE journal of biomedical and health informatics* 21 (1) (2016) 65–75.
- [9] Y. Shin, H. A. Qadir, L. Aabakken, J. Bergsland, I. Balasingham, Automatic colon polyp detection using region based deep cnn and post learning approaches, *IEEE Access* 6 (2018) 40950–40962.
- [10] H. A. Qadir, I. Balasingham, J. Solhusvik, J. Bergsland, L. Aabakken, Y. Shin, Improving automatic polyp detection using cnn by exploiting temporal dependency in colonoscopy video, *IEEE journal of biomedical and health informatics* 24 (1) (2019) 180–193.
- [11] H. A. Qadir, Y. Shin, J. Solhusvik, J. Bergsland, L. Aabakken, I. Balasingham, Polyp detection and segmentation using mask r-cnn: Does a deeper feature extractor cnn always perform better?, in: 2019 13th International Symposium on Medical Information and Communication Technology (ISMICT), IEEE, 2019, pp. 1–6.
- [12] H. A. Qadir, Y. Shin, J. Solhusvik, J. Bergsland, L. Aabakken, I. Balasingham, Toward real-time polyp detection using fully cnns for 2d gaussian shapes prediction, *Medical Image Analysis* 68 (2021) 101897.
- [13] J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodríguez, F. Vilariño, Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians, *Computerized Medical Imaging and Graphics* 43 (2015) 99–111.
- [14] J. Silva, A. Histace, O. Romain, X. Dray, B. Granado, Toward embedded detection of polyps in wce images for early diagnosis of colorectal cancer, *International journal of computer assisted radiology and surgery* 9 (2) (2014) 283–293.
- [15] Q. Angermann, J. Bernal, C. Sánchez-Montes, M. Hammami, G. Fernández-Esparrach, X. Dray, O. Romain, F. J. Sánchez, A. Histace, Towards real-time polyp detection in colonoscopy videos: Adapting still frame-based methodologies for video sequences analysis, in: *Computer Assisted and Robotic Endoscopy and Clinical Image-Based Procedures*, Springer, 2017, pp. 29–41.
- [16] F. Mahmood, R. Chen, N. J. Durr, Unsupervised reverse domain adaptation for synthetic medical images via adversarial training, *IEEE transactions on medical imaging* 37 (12) (2018) 2572–2581.
- [17] Y. Shin, H. A. Qadir, I. Balasingham, Abnormal colon polyp image synthesis using conditional adversarial networks for improved detection performance, *IEEE Access* 6 (2018) 56007–56017.
- [18] V. de Almeida Thomaz, C. A. Sierra-Franco, A. B. Raposo, Training data enhancements for improving colonic polyp detection using deep convolutional neural networks, *Artificial Intelligence in Medicine* 111 (2021) 101988.
- [19] L. Yuan, C. Ruan, H. Hu, D. Chen, Image inpainting based on patch-gans, *IEEE Access* 7 (2019) 46411–46421.
- [20] A. A. Shvets, V. I. Iglovikov, A. Rakhlin, A. A. Kalinin, Angiodysplasia detection and localization using deep convolutional neural networks, in: 2018 17th IEEE international conference on machine learning and applications (ICMLA), IEEE, 2018, pp. 612–617.
- [21] P. Isola, J.-Y. Zhu, T. Zhou, A. A. Efros, Image-to-image translation with conditional adversarial networks, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [22] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A. Courville, Improved training of wasserstein gans. in *advances in neural information processing systems*.
- [23] T. R. Shaham, T. Dekel, T. Michaeli, Singan: Learning a generative model from a single natural image, in: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4570–4580.
- [24] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems* 28 (2015) 91–99.
- [25] J. Dai, Y. Li, K. He, J. Sun, R-fcn: Object detection via region-based fully convolutional networks, in: *Advances in neural information processing systems*, 2016, pp. 379–387.
- [26] V. Iglovikov, A. Shvets, Ternaunet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation, *arXiv preprint arXiv:1801.05746*.
- [27] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.
- [28] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, et al., Speed/accuracy trade-offs for modern convolutional object detectors, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7310–7311.
- [29] A. Elgammal, B. Liu, M. Elhoseiny, M. Mazzone, Can: Creative adversarial networks, generating "art" by learning about styles and deviating from style norms, *arXiv preprint arXiv:1706.07068*.