IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 27, 2021, accepted September 26, 2021, date of publication September 29, 2021, date of current version October 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3116265

Training on Polar Image Transformations Improves Biomedical Image Segmentation

MARIN BENČEVIĆ[1], IRENA GALIĆ[1], MARIJA HABIJAN[1], AND DANILO BABIN[2]
[1]Faculty of Electrical Engineering, Computer Science and Information Technology, J. J. Strossmayer University of Osijek, 31000 Osijek, Croatia
[2]imec-TELIN-IPI, Faculty of Engineering and Architecture, Ghent University, 9000 Ghent, Belgium

Corresponding author: Marin Benčević (marin.bencevic@ferit.hr)

This work was supported in part by the Croatian Science Foundation under Project UIP-2017-05-4968.

ABSTRACT A key step in medical image-based diagnosis is image segmentation. A common use case for medical image segmentation is the identification of single structures of an elliptical shape. Most organs like the heart and kidneys fall into this category, as well as skin lesions, polyps, and other types of abnormalities. Neural networks have dramatically improved medical image segmentation results, but still require large amounts of training data and long training times to converge. In this paper, we propose a general way to improve neural network segmentation performance and data efficiency on medical imaging segmentation tasks where the goal is to segment a single roughly elliptically distributed object. We propose training a neural network on polar transformations of the original dataset, such that the polar origin for the transformation is the center point of the object. This results in a reduction of dimensionality as well as a separation of segmentation and localization tasks, allowing the network to more easily converge. Additionally, we propose two different approaches to obtaining an optimal polar origin: (1) estimation via a segmentation trained on non-polar images and (2) estimation via a model trained to predict the optimal origin. We evaluate our method on the tasks of liver, polyp, skin lesion, and epicardial adipose tissue segmentation. We show that our method produces state-of-the-art results for lesion, liver, and polyp segmentation and performs better than most common neural network architectures for biomedical image segmentation. Additionally, when used as a pre-processing step, our method generally improves data efficiency across datasets and neural network architectures.

INDEX TERMS Convolutional neural network, medical image processing, medical image segmentation, semantic segmentation.

I. INTRODUCTION

Image segmentation is the task of delineating diagnostically important anatomical structures on medical images. Segmentation is a necessary step in most computer-aided diagnosis use cases, and a pre-processing step for many other medical tasks like disease risk estimation, classification, etc. A common use case for medical segmentation is identifying single structures with a roughly elliptical shape or distribution, like most organs, skin lesions, polyps, cardiac adipose tissues, and similar structures and abnormalities.

Neural networks have achieved state-of-the-art results in many medical image segmentation tasks, however, they often require large amounts of annotated training images, which are time-consuming and costly to obtain. In this paper, we propose a general way to improve neural network segmentation data efficiency and performance on medical imaging segmentation tasks where the goal is to segment roughly elliptically distributed objects.

We propose and explore ways to train neural networks for biomedical image segmentation on polar transformations of images. The polar transformation transforms an image from Cartesian coordinates into a new coordinate system where the two axes are the rotation around an origin and radius from that origin. When the regions to be segmented are elliptical in shape or distribution, this transformation results in a reduction of dimensionality, allowing convergence in fewer epochs and good performance even in models with a low number of parameters.

Experimentally, we observed that selecting a correct polar origin is one of the key parameters that determine segmentation performance. Therefore, we propose two

The associate editor coordinating the review of this manuscript and approving it for publication was Zhan-Li Sun.

different approaches of selecting an optimal polar origin: (1) estimation via a segmentation neural network trained on non-polar images and (2) estimation via a neural network trained to predict heatmaps. Our method is evaluated on the tasks of polyp segmentation, liver segmentation, skin lesion segmentation, and epicardial adipose tissue (EAT) segmentation. The proposed methods can be used as a pre-processing step for existing neural network architectures, so we evaluate the methods using common neural network architectures for medical image segmentation including U-Net [1], U-Net++ [2] with a ResNet [3] encoder, and DeepLabV3+ [4] with a ResNet [3] encoder.

Evaluation of our approach as a pre-processing step shows that it improves segmentation performance across different datasets and neural network architectures while making the networks more robust to small dataset sample sizes. All used code for this paper is available at github.com/marinbenc/medical-polar-training.

A. RELATED WORK

1) COMBINING POLAR COORDINATES AND NEURAL NETWORKS

Several image segmentation methods were proposed that utilize polar coordinates. Liu *et al.* [5] proposed an approach they call Cartesian-polar dual-domain network (DDNet) to perform optic disc and cup segmentation in retinal fundus images. The neural network contains two encoding branches, one for a Cartesian input image and another for the polar transformation of the same input image. The predictions are fused into a single feature vector which is then decoded into a final segmentation. Salehinejad *et al.* [6] used the polar transformation as a way to augment training data by transforming each input image into multiple polar images at various polar origins, thus increasing the number of training data. Kim *et al.* (2020) [7] proposed a convolutional neural network layer for images in polar coordinates to achieve rotational invariance. Their cylindrical convolution layer uses cylindrically sliding windows to perform a convolution. Kim *et al.* [8] proposed a user-guided segmentation method where an expert selects the point used as the polar origin. The transformed image is then segmented using a convolutional neural network (CNN). Esteves *et al.* [9] proposed a polar transformer network for image classification. Note that ''transformer network'' here refers to spatial transformer networks [10] and not attention-based networks commonly called transformers. The network consist of a polar origin predictor and a neural network that predicts a heatmap. The centroid of the heatmap is then used as the origin for a polar transformation of the input image. The polar image is classified using a CNN. This approach is most similar to our proposed method, however, their approach focuses on image classification, not segmentation. Additionally, our approach differs in the ways the ground truth data is prepared, the used neural network architectures, as well as other details.

2) BIOMEDICAL IMAGE SEGMENTATION

One of the most used neural network architectures for biomedical image segmentation is U-Net [1], an encoder-decoder based architecture where intermediate feature maps of the encoder are concatenated with the appropriate feature maps of the decoder, allowing the network to simultaneously learn context and precise localization. Multiple modifications of the U-Net architectures were proposed. Zhou *et al.* [2] proposed a nested U-Net architecture called U-Net++, where the encoder and decoder are connected via dense convolutional blocks instead of simple concatenation. Jha *et al.* [11] proposed an architecture called Double-U-Net based on two U-Nets stacked together, where the first one uses a VGG encoder pre-trained on the ImageNet dataset. The output of the first U-Net is used as input, together with the input image, for the second U-Net. Additionally, the output of the first U-Net is concatenated together with the output of the second U-Net to produce the final segmentation. They achieve state-of-the-art results for lesion segmentation. Azad *et al.* [12] proposed a U-Net-based architecture where the decoder was modified by adding bi-directional convolutional LSTM and squeeze-and-excitation layers [13]. Tomar *et al.* [14] proposed a general network for medical image segmentation, validated on seven biomedical image datasets. Their method uses an encoder-decoder architecture with squeeze-and-excitation residual blocks and recurrent learning. The model's output at each epoch is stored and used as an input to the next epoch, iteratively improving the output while reducing training time. Ibtehaz and Rahman [15] proposed MultiResUNet, an improvement of U-Net wherein U-Net's convolutional blocks are replaced with blocks that use differently-sized convolutional kernels in parallel. Additionally, they added convolutional blocks to U-Net's skip connections.

There are various proposed approaches for polyp segmentation from colonoscopy images that use deep learning. Fan *et al.* [16] proposed a parallel reverse attention network for polyp segmentation. Their method works by first using a parallel partial decoder which decodes feature input maps into a global semantic map of the image. This map is then refined by a series of recurrent reverse attention layers. Fang *et al.* [17] used a network with one encoder and two mutually constrained decoders, one for predicting areas and another for predicting boundaries. The network then aggregates the features. The authors train the network using a boundary-sensitive loss function. Huang *et al.* [18] proposed an encoder-decoder neural network which uses a HarDNet-based [19] encoder and a cascaded partial decoder, with three branches are connected to the encoder, and their features are densely aggregated to produce the final output. Each branch uses proposed neural network layers called receptive field blocks.

For liver segmentation, Valanarasu *et al.* [20] proposed KiU-Net. Their network consists of two branches. The first branch is an overcomplete convolutional network where the input image is projected into a higher-dimensional space,

forcing the network to learn fine details and accurate edges. The other branch is a regular U-Net network. The two branches are then fused to produce a final segmentation.

For EAT segmentation, Zhang *et al.* [21] proposed an approach using two successive U-Net networks. The first network performs a segmentation of the pericardium, a protective layer of connective tissue that encloses EAT. The output segmentation is refined using morphological operators and then used as a mask for the input to the second U-Net, which is trained to segment EAT for the pericardium region. Commandeur *et al.* [22] proposed training two convolutional neural networks. The first network determines the heart limits and segments adipose tissues. The output of the first network is used to sample the input to the second neural network which delineates the pericardium. They also use a polar transformation to transform the input of the second network.

While there are proposed methods which combine the polar transformation with neural networks, most of them solve classification tasks. Some medical image segmentation methods use the polar transformation as a preprocessing step, however the way they obtain the origin of the polar transformation is usually based on heuristics. To our knowledge there is currently no work that explores using the polar transformation with a dynamic polar origin as a preprocessing step for semantic segmentation in a variety of medical image datasets.

II. METHODOLOGY

All of the proposed methods rely on training a neural network model to segment polar images. To train on polar images, the input images need to be transformed using a polar origin which is near the center of the segmented object. The correct origin is not known ahead of time, so a prerequisite for predictions on polar images is a method to determine the correct polar origin. We propose and evaluate two different methods for automatically obtaining the polar origin: (1) estimation via a segmentation trained on non-polar images and (2) training a center-point predictor which predicts heatmaps from input images. This section describes these methods, as well methods to train the final segmentation model on the polar images.

A. POLAR TRANSFORMATIONS AND RATIONALE

Images are most commonly viewed in Cartesian coordinates, where the pixels are arranged along the x- and y-axes. The polar coordinate system has two axes: (1) the radial coordinate ρ, which is the distance of a point from the origin of the polar transformation; and (2) the angular coordinate ϕ, which is the angle between the point and the reference direction. In other words, the x-axis of the polar image represents the distance from an origin, while the y-axis represents the rotation around the origin. This makes polar coordinates invariant to rotation.

Our intuition is that polar transformations can be especially beneficial to segmenting images where an elliptical border must be found on the image. Consider a contrived example of predicting a circular decision boundary on a single-channel image with a linear model. A circular decision boundary must be modeled by a function of at least four dimensions. When transformed to polar coordinates, a perfect circle in Cartesian coordinates becomes a straight line, as shown in Fig. 1. This linear decision boundary can be modeled with a simpler linear function in two dimensions. The image in polar coordinates would require a less complex model to predict a border. It is possible that, even for more complex examples, the polar transform of an image of a roughly elliptical object reduces the required segmentation model complexity, as shown visually in Fig. 2. Furthermore, by transforming an image to polar coordinates using a polar origin that is the center of the object, we fix the location and standardize border distances in each training example. The model can then learn the distance of the border from the origin at each angle around the origin, without having to learn to localize the object.

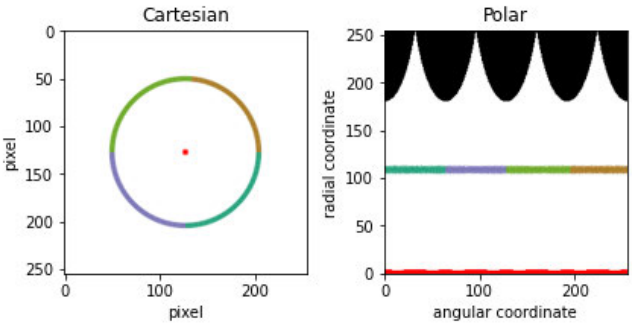


FIGURE 1. An example of a polar transformation.

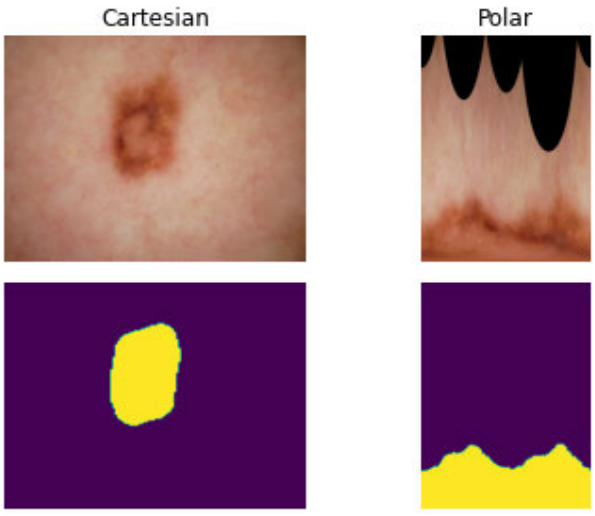


FIGURE 2. An example image and label from the lesion dataset and their corresponding polar transformation.

To obtain a polar transformation, the angle and magnitude of each pixel (x, y) of the original image are calculated using (1):

$$\begin{aligned} magnitude(x, y) &= \sqrt{x^2 + y^2}, \\ angle(x, y) &= atan2(y, x) \cdot \frac{180}{\pi} \end{aligned} \tag{1}$$

where *atan*2 is the 2-argument arctangent function.

Given a polar origin (c_x, c_y) of a Cartesian image $I(x, y)$ of resolution $H \times W$, we obtain each point (ρ, ϕ) of the polar transformation $I'(\rho, \phi)$ using (2).

$$\rho = \frac{H}{2\Pi} \cdot angle(x - c_x, y - c_y)$$
$$\phi = \frac{W}{\sqrt{(W/2)^2 + (H/2)^2}} \cdot magnitude(x - c_x, y - c_y) \quad (2)$$

B. TRAINING A NETWORK ON POLAR IMAGES

In each of our approaches, the final segmentation is done using a neural network trained on polar transformations of the input images. In the rest of this paper, we refer to this network as the polar network. In all of the described approaches, the polar transformation is not part of the network architecture itself, but happens as a preprocessing step for the polar network. To transform each input image, the polar origin is determined as the center of mass of the ground truth label for that image. The center of mass of an image $I(x, y)$ is calculated by first calculating the spatial image moments matrix M, where the entry of the matrix at row i and column j is calculated using (3).

$$M_{ij} = \sum_{x,y} I(x, y) \cdot x^i \cdot y^j \quad (3)$$

The center of mass (c_x, c_y) of the image can then be calculated using (4).

$$c_x = M_{10}/M_{00}$$
$$c_y = M_{01}/M_{00} \quad (4)$$

Finally, to increase the model's robustness to suboptimal center point predictions, we augment the calculated center for training images [9]. Each training image has a 30% chance of varying the center's x and y coordinates by a random value in the range $(-S \cdot 0.05, S \cdot 0.05)$, where S is the smallest resolution of the image, i.e. $S = min(width, height)$.

C. CENTERPOINT PREDICTION

Once the polar network is trained, inference can be done by transforming an input image to polar coordinates. The polar network requires choosing a center that is close to the center of mass of the segmented object. Because a future input image is unlabeled, the correct center needs to be inferred from the image. We propose two ways to accomplish this, described in this section.

1) TRAINING THE SAME NEURAL NETWORK ON CARTESIAN AND POLAR IMAGES

Our first approach is training the same neural network on cartesian and polar images. A summary of this approach is presented in Fig. 3. With this approach, the inference is done by first feeding the original Cartesian input images into a neural network used for segmentation. We refer to this network as the Cartesian network. For an input image, the polar origin is calculated as the center of mass of the Cartesian network's prediction for that input image, using (4). This polar origin is used to transform the original input image to polar coordinates, and the transformed image is fed to the polar network. The output of the polar network is transformed back to Cartesian coordinates to obtain a final segmentation. We assume identical architectures for both the Cartesian and polar networks. This makes applying this framework to existing architectures very straightforward, as it does not require designing new neural network architectures or specific hyperparameter optimization, and allows for using transfer learning to initialize the networks.

2) TRAINING A CENTERPOINT PREDICTOR

In the second approach for determining the optimal polar origin, we train a model specifically tasked with predicting the correct polar origin for each input image, which is then used to transform the input image. The approach is shown in Fig. 4. We do this by training a neural network based on the stacked hourglass architecture [23] first used for human pose estimation. Instead of training a regressor network to predict key points in an image, the stacked hourglass architecture uses a series of stacked encoder-decoder networks, where the output of each stack is a heatmap centered on the key point to be predicted. The output of each stack is fed as input into the next stack, allowing successive refinement of the heatmap prediction. During training, the loss of each stack's output is averaged to produce the final loss, allowing deep supervision. The final prediction heatmap is the output of the last stack in the network. To predict the center point, we use 8 stacked hourglass blocks, which we empirically determined as the value providing the best results. The network receives images in Cartesian coordinates and predicts a heatmap of the image.

The ground truth heatmaps were generated by calculating the center of mass of each ground truth label image using (4). We then create the heatmap as an image with a 2D gaussian with the mean on the center of mass on the image and a standard deviation of 8 pixels for all datasets except the liver, and 16 for the liver. Example heatmaps are shown in Fig. 5. The optimal value for the standard deviation was determined empirically on the validation datasets. We found that the optimal value of the standard deviation is proportional to the size of the object.

Additionally, during training, we use augmentation to increase the number of training inputs. In particular, during training each input example the following random augmentations are applied:

- A 50% chance of a horizontal flip.
- A 30% chance of a random combination of shifting up to 6.5% of the image dimensions, scaling up to 10% and rotating up to 45°.
- A 30% chance for a grid distortion, details of which are described in [24].

The center-point predictor outputs 8 separate heatmaps [23]. We calculate the predicted center as the coordinates of the pixel with the largest intensity in the heatmap predicted by the final layer of the model. This predicted center is then used to transform the input image to polar coordinates, and the

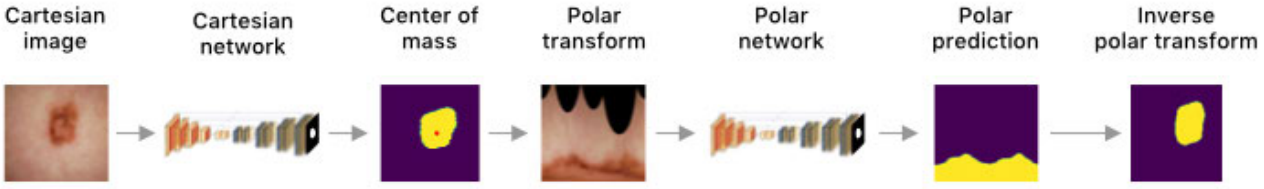


FIGURE 3. A diagram of the approach of predicting polar origins from a Cartesian network. The first network performs an initial segmentation, which is then used to extract a polar origin for the polar transformation. The method does not rely on any specific neural network architecture. The Polar and Cartesian network can be any neural network which takes an input image and produces a binary segmentation mask as output. The red point shows the extracted polar origin. The Polar network is trained on polar image transformations. The polar transformation is not part of the network itself, but happens as a preprocessing step for the Polar network.

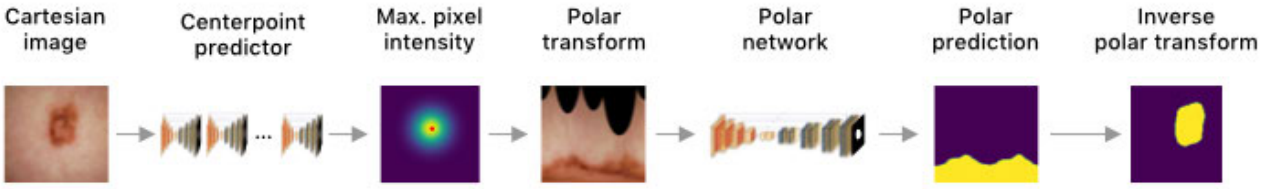


FIGURE 4. A diagram of the approach of using a centerpoint prediction network. The first network can be any neural network which predicts a heatmap from an input image, which is then used to extract polar origin, shown as a red point. The Polar network can be any semantic segmentation neural network which produces a binary mask output from an input image. The Polar network is trained on polar image transformations. The polar transformation is not part of the network itself, but happens as a preprocessing step for the Polar network.

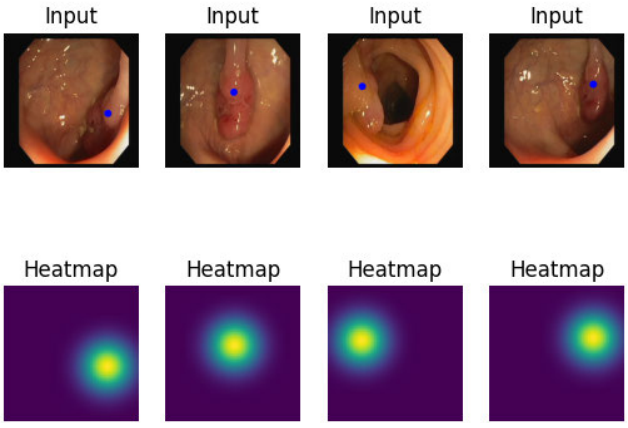


FIGURE 5. Examples of heatmaps generated from the ground truth data. The heatmap is a gaussian centered on the center of mass of the ground-truth label, shown as a blue point on the input images.

transformed image is fed into the polar network to perform the segmentation. Finally, the segmentation label is transformed back to Cartesian coordinates.

III. EXPERIMENTS

To validate the generality of our approach, we trained a variety of neural network architectures on multiple medical imaging datasets. In particular, we trained three different neural network architectures: U-Net [1], U-Net++ [2] with a ResNet encoder and DeepLabV3+ [4] with a ResNet encoder. Notably, each dataset we use presents a problem wherein almost all examples a single roughly elliptical object needs to be segmented. For each dataset and network architecture combination, we train a Cartesian and polar network, and we then perform four different experiments:

1) testing the Cartesian network using Cartesian images
2) testing the polar network using the ground-truth polar origin
3) testing the polar network using polar origins obtained from predictions of the Cartesian network, as outlined in II-C1
4) testing the polar network using polar origins from the center-point predictor, as outlined in II-C2.

A. DATASETS DESCRIPTION

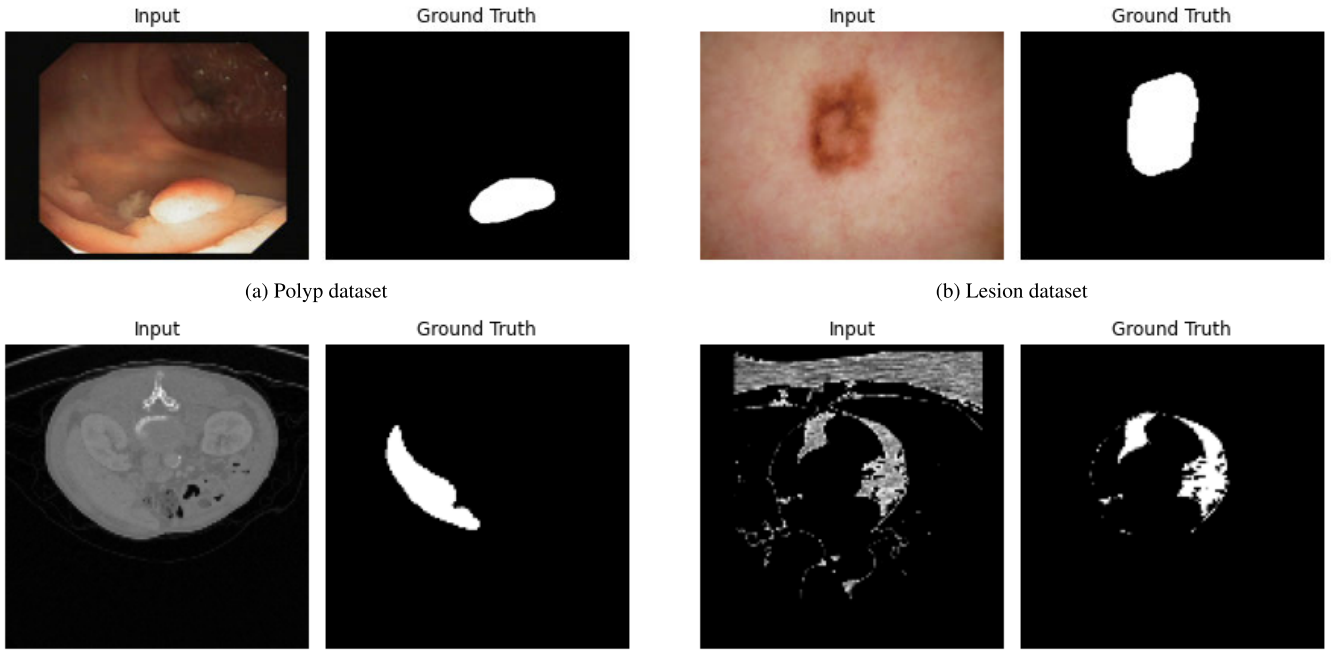
We used four different datasets to train the network. In this section, we give an overview of each used dataset and how it was preprocessed. Note that for training the center-point predictor, the input images were resized to a resolution of 256×256, while the generated heatmaps were resized to 64×64 pixels. Otherwise, all preprocessing steps described here are applied to the center-point model datasets as well. Each dataset was normalized and zero-centered to better facilitate network convergence.

1) POLYP DATASET

The CVC-ClinicDB dataset [25] contains 612 RGB colonoscopy images with the resolution 288×384 with labeled polyps from MICCAI 2015. We normalize each image to a range of $[-0.5, 0.5]$. We use the original image resolution to train all networks except the centerpoint network. As is used in [11], we use an 80%, 10% and 10% split for training, validation and testing datasets, respectively. An example of the dataset is shown in Fig. 6(a).

2) LIVER DATASET

The second dataset we use is the LiTS dataset [26] from the Liver Tumor Segmentation Challenge from MICCAI 2017.



(a) Polyp dataset

(b) Lesion dataset

(c) Liver dataset

(d) EAT dataset

FIGURE 6. Example input images and ground-truth labels for each dataset used in our experiments.

The dataset contains 131 CT scans of patients with hepatocellular carcinoma, with the liver as well as tumor lesions labeled by experts. In our experiments, we disregard the lesion segmentation labels and treat the dataset as a binary liver segmentation problem. In addition, we removed all slices that did not contain a ground-truth liver segmentation label, resulting in a dataset of roughly 15,000 slices. Each axial slice is thresholded to a Hounsfield scale range of [0, 200] HU that contains the liver. Next, the slices are normalized to a [0, 1] range and zero-centered by subtracting the global intensity of all training slices (0.1). We then proceed to train the networks on each axial slice separately. We use 101 scans for training, 15 scans for validation, and the remaining 15 scans for testing. Example liver segmentation images are shown in Fig. 6(c).

3) LESION DATASET

The third dataset we use is the ISIC 2018 Lesion Boundary Segmentation dataset [27], [28] which contains 2,694 dermatoscopy images of skin lesions with expert labels of the lesions from various anatomic sites and several different institutions. We resize each image to a resolution of 384 × 512 and use a training, validation and test split of 80%, 10% and 10%, respectively. This is consistent with [11]. Additionally, we normalize each image to a range of [−0.5, 0.5]. An example of a lesion input image and its corresponding label is shown in Fig. 6(b).

4) EAT DATASET

Finally, we also train on a dataset of labeled EAT regions from 20 patients' cardiac CT scans from the Cardiac Fat Database [29]. The dataset has three classes labeled: the pericardium, EAT, and pericardial adipose tissue. We disregard all original labels except EAT and treat the dataset as a binary EAT segmentation dataset. The dataset is first split into training (10 patients), validation (5 patients), and test (5 patients) datasets. In the original dataset, each slice is thresholded to the adipose tissue range of [−200, −30] HU and registered so that anatomical structures have the same locations. In addition to these original pre-processing steps, we normalize each slice to a [0, 1] range and zero-center the dataset by subtracting a global mean intensity of the training set (0.1). We then train on each CT slice separately. An input image of the EAT dataset and its corresponding label is shown in Fig. 6(d).

B. IMPLEMENTATION DETAILS

We use the OpenCV linear polar transformation implementation. Each model is implemented and trained using PyTorch 1.7.1 on an NVIDIA GeForce RTX 3080 GPU. For all networks, we use the Adam optimizer with a learning rate of 10^{-3}. A batch size of 8 was used for all networks except the center-point model, where a batch size of 6 was used for the lesion and liver datasets, and 8 for all remaining datasets. We trained all models up to a maximum of 200 epochs and used checkpoints after each epoch to store the model with the best validation loss. We modify the Dice coefficient to act as a loss function as shown in (5).

$$DSC_{loss} = 1 - \frac{2|X \cap Y| + \lambda}{|X| + |Y| + \lambda} \tag{5}$$

TABLE 1. Results of our proposed approaches for polyp segmentation on the CVC-ClinicDB dataset [25] for three different neural network architectures. The cartesian network is the network trained on Cartesian images. "GT centers" refers to obtaining a polar origin from the ground-truth labels and segmentation using the polar network. "Cartesian centers" refers to predicting the polar origins from the Cartesian network and then performing segmentation using the polar network. "Model centers" refers to using the center-point predictor to obtain polar origins.

Segm. net.	Method	DSC	mIoU	Prec.	Rec.
U-Net	Cartesian	0.8315	0.7604	0.8513	0.8334
	GT centers	0.9484	0.9141	0.9563	0.9442
	Cart. centers	0.8973	0.8571	0.8996	0.8998
	Model centers	0.9374	0.8977	0.9488	0.9368
Res-U-Net++	Cartesian	0.8356	0.7636	0.9004	0.8256
	GT centers	0.9557	0.9260	0.9583	0.9554
	Cart. centers	0.9063	0.8685	0.9243	0.9027
	Model centers	0.9332	0.8924	0.9477	0.9321
DeepLabV3+	Cartesian	0.8706	0.8013	0.8857	0.8876
	GT centers	0.9593	0.9296	0.9576	0.9682
	Cart. centers	0.9212	0.8823	0.9179	0.9397
	Model centers	0.9338	0.8967	0.9436	0.9347

TABLE 2. Results of our proposed approaches for lesion segmentation on the ISIC 2018 Lesion Boundary Segmentation dataset [27], [28] for three different neural network architectures. The cartesian network is the network trained on Cartesian images. "GT centers" refers to obtaining a polar origin from the ground-truth labels and segmentation using the polar network. "Cartesian centers" refers to predicting the polar origins from the Cartesian network and then performing segmentation using the polar network. "Model centers" refers to using the center-point predictor to obtain polar origins.

Segm. net.	Method	DSC	mIoU	Prec.	Rec.
U-Net	Cartesian	0.8256	0.7393	0.8407	0.8712
	GT centers	0.9320	0.8824	0.9261	0.9541
	Cart. centers	0.8836	0.8317	0.8746	0.9492
	Model centers	0.9224	0.8699	0.9165	0.9494
Res-U-Net++	Cartesian	0.8664	0.7925	0.8728	0.9122
	GT centers	0.9439	0.9014	0.9418	0.9584
	Cart. centers	0.9125	0.8653	0.9075	0.9540
	Model centers	0.9253	0.8743	0.9253	0.9464
DeepLabV3+	Cartesian	0.8717	0.7984	0.8807	0.9068
	GT centers	0.9459	0.9059	0.9418	0.9632
	Cart. centers	0.9162	0.8686	0.9097	0.9536
	Model centers	0.9235	0.8721	0.9125	0.9570

TABLE 3. Results of our proposed approaches for liver segmentation on the LiTS dataset [26] for three different neural network architectures. The cartesian network is the network trained on Cartesian images. "GT centers" refers to obtaining a polar origin from the ground-truth labels and segmentation using the polar network. "Cartesian centers" refers to predicting the polar origins from the Cartesian network and then performing segmentation using the polar network. "Model centers" refers to using the center-point predictor to obtain polar origins.

Segm. net.	Method	DSC	mIoU	Prec.	Rec.
U-Net	Cartesian	0.8976	0.8505	0.8997	0.9201
	GT centers	0.9553	0.9227	0.9595	0.9569
	Cart. centers	0.9302	0.8985	0.9279	0.9429
	Model centers	0.9125	0.8828	0.9108	0.9219
Res-U-Net++	Cartesian	0.8908	0.8463	0.8936	0.9085
	GT centers	0.9548	0.9215	0.9492	0.9661
	Cart. centers	0.9219	0.8898	0.9119	0.9428
	Model centers	0.9109	0.8795	0.9009	0.9306
DeepLabV3+	Cartesian	0.8868	0.8341	0.8995	0.8959
	GT centers	0.9518	0.9171	0.9547	0.9550
	Cart. centers	0.9253	0.8932	0.9244	0.9361
	Model centers	0.9092	0.8783	0.9075	0.9199

TABLE 4. Results of our proposed approaches for EAT segmentation on the Cardiac Fat database [29] for three different neural network architectures. The cartesian network is the network trained on Cartesian images. "GT centers" refers to obtaining a polar origin from the ground-truth labels and segmentation using the polar network. "Cartesian centers" refers to predicting the polar origins from the Cartesian network and then performing segmentation using the polar network. "Model centers" refers to using the center-point predictor to obtain polar origins.

Segm. net.	Method	DSC	mIoU	Prec.	Rec.
U-Net	Cartesian	0.7544	0.5812	0.7190	0.6949
	GT centers	0.8088	0.6607	0.7986	0.7675
	Cart. centers	0.7835	0.6227	0.7455	0.7208
	Model centers	0.7840	0.6252	0.7451	0.7302
Res-U-Net++	Cartesian	0.3410	0.1743	0.2700	0.3294
	GT centers	0.8030	0.6827	0.7939	0.8043
	Cart. centers	0.5466	0.3980	0.5286	0.5066
	Model centers	0.7740	0.6140	0.7156	0.7453
DeepLabV3+	Cartesian	0.6380	0.4246	0.5665	0.5940
	GT centers	0.6952	0.5123	0.6519	0.6779
	Cart. centers	0.6696	0.4716	0.5988	0.6454
	Model centers	0.6720	0.4779	0.6070	0.6488

where X and Y are the input and predicted images, respectively, and λ is a smoothing parameter set to 1 in our experiments. This loss function is used to train all models except the center-point model.

The centerpoint model outputs eight heatmaps [23]. We use a loss function that is the mean of the mean squared errors between each of the heatmaps and the ground truth heatmap. The code used for all experiments is available at github.com/marinbenc/medical-polar-training.

IV. RESULTS

We evaluate segmentation performance along with four key metrics: the Dice coefficient (DSC), the median intersection-over-union score (mIoU), precision, and accuracy. Precision and accuracy are both calculated pixel-wise. The results of training the different approaches presented in III are shown in Table 1 for polyp segmentation, Table 2 for lesion segmentation, Table 3 for liver segmentation and Table 4 for EAT segmentation. In all cases, training on polar coordinates

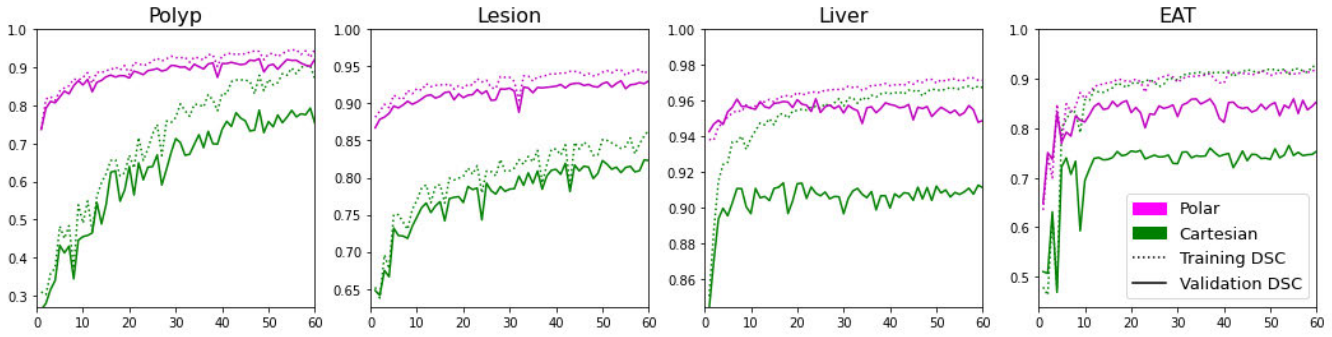


FIGURE 7. The training and validation Dice coefficient (DSC) of the polar and Cartesian U-Net models during training.

TABLE 5. A comparison between our method (approach with best results) and the state of the art on the same datasets.

Dataset	Method	DSC	mIoU	Prec.	Rec.
Polyp	PraNet [16]	0.8990	0.8490	-	-
	FANet [14]	-	0.8937	0.9401	0.9339
	Our method	**0.9374**	**0.8977**	**0.9488**	**0.9368**
Lesion	DeepLabV3+ [4]	0.8717	0.7984	0.8807	0.9068
	DoubleU-Net [11]	0.8962	0.8212	**0.9459**	0.8780
	Our method	**0.9253**	**0.8743**	0.9253	**0.9464**
Liver	U-Net [1]	0.8976	0.8505	0.8997	0.9201
	KiU-Net 3D [20]	**0.9423**	0.8946	-	-
	Our method	0.9302	**0.8985**	0.9279	0.9429
EAT	U-Net. [1]	0.7544	0.5812	0.7190	0.6949
	Zhang et al. [21]	**0.9119**	**0.8425**	-	-
	Our method	0.7840	0.6252	0.7451	0.7302

TABLE 6. Ablation study of our approach for the polyp dataset.

Method	DSC	Difference
Cartesian	0.8315	-
Polar (Cartesian origins)	0.8918	+0.0603
Polar (Centerpoint predictor)	0.9094	+0.0176
+ centerpoint augmentation	0.9288	+0.0194
+ polar network training augmentation	0.9374	+0.0086

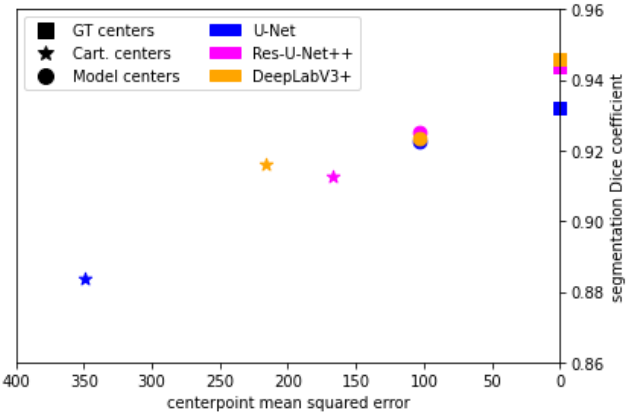


FIGURE 8. The relationship between mean squared errors of the centers used for the polar transformation and segmentation performance of the polar network on the lesion dataset. The mean squared errors are calculated compared to the ground-truth centers.

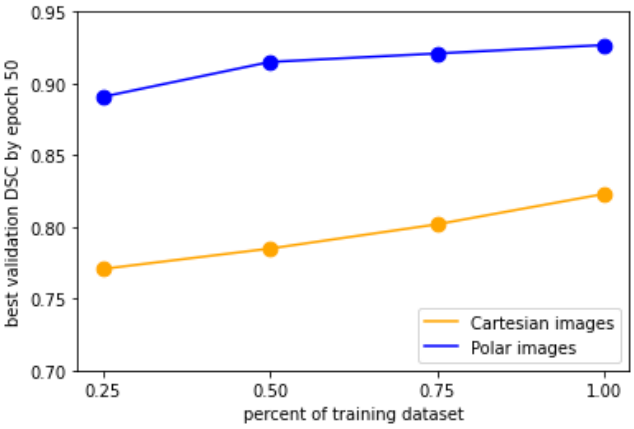
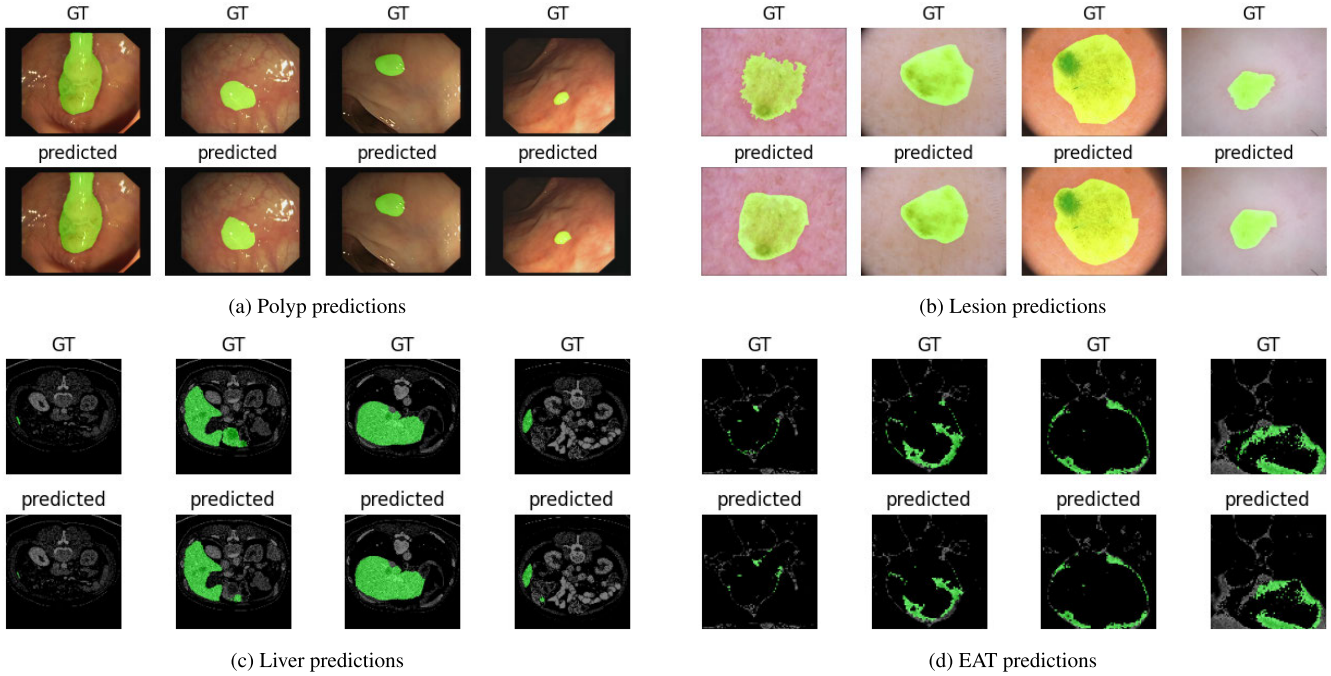


FIGURE 9. The best Dice coefficient by epoch 50 for models trained on subsets of the lesion training dataset.

improves the segmentation in all metrics when compared to training the same model on Cartesian coordinates. As is to be expected, testing the polar network on images transformed using the ground truth polar origins produces the best results. A close second is predicting the polar origin from the center-point predictor. Predicting polar origins from the Cartesian model leads to less accurate polar origins, and the results are worse, however, they are still better than using only the Cartesian model.

We also compare our methods to other state-of-the-art methods that use the same datasets, shown in Table 5. We achieve state-of-the-art results for the polyp and liver datasets. Additionally, we achieve state-of-the-art liver segmentation when compared to other per-slice methods, and nearly state-of-the-art results when compared to 3D-based methods. For EAT segmentation, our approach outperforms standard medical image segmentation networks but does not



(a) Polyp predictions

(b) Lesion predictions

(c) Liver predictions

(d) EAT predictions

FIGURE 10. A random sampling of inverse polar transformed predictions from the polar network with the polar origins predicted from the centerpoint predictor for various datasets. The prediction is shown in green and overlaid on top of the original input image. EAT predictions (d) are cropped and zoomed to better show the predictions.

achieve state-of-the-art performance due to segmenting EAT directly and not first segmenting the pericardium.

A training graph for a polar and Cartesian U-Net-based network is shown in Fig. 7.

Additionally, we evaluate the accuracy of the different ways of obtaining the polar origin. This accuracy is compared with segmentation performance in Fig. 8.

We also train several models with both polar and Cartesian coordinates on subsets of the training dataset. Namely, we trained models on 25%, 50%, 75%, and 100% of the lesion training dataset for 50 epochs. The results of this training are shown in Fig. 9. The polar network is much more data efficient and achieves better results than the cartesian network even with only 25% of the data.

V. DISCUSSION

We obtain state-of-the-art results for polyp and lesion segmentation by training common biomedical image segmentation models.

In the liver dataset, we achieve state-of-the-art results when compared to other 2D methods, but 3D methods achieve the same or slightly better results [20]. The liver dataset is by far the largest dataset we evaluated. As such, improvements gained from encoding localization information and reducing dimensionality might not be as large as in smaller datasets, since the network has enough data to learn these complex structures. The EAT dataset is one where the task is not to find a single object, but instead, segment multiple smaller pockets of tissue around the heart. This task is more challenging for common models like U-Net and requires a more complex approach [21]. It is possible that combining these existing approaches, namely segmenting the pericardium first, with training on polar coordinates would lead to an improvement in the state of the art.

We also show that training on polar images leads to a significant improvement in segmentation performance when compared to training on Cartesian images for the same network architecture. Additionally, as shown in Fig. 7, the polar network portions of our approach converge in much fewer epochs than the Cartesian networks. This is in part due to the location information being encoded in the image itself via the polar origin, and in part due to a possible data dimensionality reduction, allowing the network to more easily optimize the loss function. The polar networks are also more robust to low dataset sample size. This is especially important in biomedical image segmentation where the availability of large labeled datasets is often very limited. Training curves for all of our experiments are included at github.com/marinbenc/medical-polar-training.

Predicting the center point from the Cartesian model, while still an improvement over the plain Cartesian network, leads to worse results than those obtained by the center point predictor model. We conclude that segmentation is highly dependent on choosing the correct polar origin. This dependency is somewhat loosened by adding polar origin augmentation when training the polar network.

Fig. 10 shows a random sampling of predictions from the polar network using the center point predictor for polar origins. Qualitatively, we conclude that the network achieves very good segmentation results, leading to a very high overlap

with the target object. The network successfully segments both small and elliptical as well as large and unevenly shaped polyps. On the lesion images, the network predicts a smooth border when sometimes the actual border of the lesion is rough, as shown in the left-most example on Fig. 10(b), however, the network still does a good job of delineating a lesion border even when the color of the lesion is very similar to the surrounding skin. The network successfully predicts a liver border both when the liver is very large and very small on the image, showing good scale invariance, but sometimes under segments the liver when multiple connected components are needed. On the EAT dataset, the network successfully learns to segment EAT despite its highly discontinuous and sparse distribution. However, the network sometimes under segments EAT.

Finally, we also perform an ablation study shown in Table 6. Training on the polar coordinates with the polar origins predicted from the cartesian network yields the largest performance improvement. Predicting the polar origin from the center point predictor as well as adding center point augmentation to the predictor play a roughly equally important role in the performance. Lastly, a small performance improvement is further achieved by using data augmentation when training the polar network.

A potential improvement of our method is to train a single neural network that combines the center-point predictor and the segmentation network and is trained end-to-end. In our approach, polar origins are always optimized towards the center of mass of the segmented object. Training an end-to-end network would allow the polar origins to be optimized for that specific segmentation task. Additionally, the center points could be obtained manually from experts, creating a user-guided segmentation approach similar to [8]. The center points could also be obtained by a more basic segmentation approach like thresholding or other traditional image processing method, leading to a possible reduction in the number of required neural network parameters to achieve good segmentation. Furthermore, in our experiments, we found that the segmentation is dependent on choosing the correct standard deviation of the generated heatmaps for training the center point predictor. An improvement to our method could be made by developing a method to automatically estimate the standard deviation from the training or validation data without needing to first train the center point predictor.

VI. CONCLUSION

We explored training neural networks for biomedical image segmentation on polar transformations on images. We hypothesized that polar transformations would reduce the dimensionality of the input images, and allow the network to separately learn localization and fine segmentation of an object. We showed that training time improves when training on polar images for tasks where a single object which is roughly elliptical in shape or distribution needs to be segmented. Additionally, we show that training on polar images achieves state-of-the-art results on small datasets, and achieves near state-of-the-art results on larger datasets using generic low-parameter-count models like U-Net. We also noted that choosing the correct polar origin is essential for improving performance on polar images. Therefore, we proposed two different ways of obtaining the polar origin automatically from unlabeled input images. We trained a center-point predictor which predicts a heatmap to produce a polar origin, and showed that its performance is better than predicting the origin from a segmentation network trained on Cartesian images. We noted that sometimes our method under segments in examples where multiple objects need to be segmented.

While our approach already produces state-of-the-art results in some cases, our results could be further improved. Our approach can be used as a pre-processing step for existing and future semantic segmentation methods that use neural networks to provide additional segmentation improvement. Therefore, it is possible that our approach could be used in a variety of different biomedical and non-medical segmentation applications.

REFERENCES

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI*, vol. 9351, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham, Switzerland: Springer, 2015, pp. 234–241.

[2] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, vol. 11045, D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan, and A. Madabhushi, Eds. Cham, Switzerland: Springer, 2018, pp. 3–11.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[4] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with Atrous separable convolution for semantic image segmentation," in *Computer Vision—ECCV*, vol. 11211, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 833–851.

[5] Q. Liu, X. Hong, W. Ke, Z. Chen, and B. Zou, "DDNet: Cartesian-polar dual-domain network for the joint optic disc and cup segmentation," 2019, *arXiv:1904.08773*. [Online]. Available: http://arxiv.org/abs/1904.08773

[6] H. Salehinejad, S. Valaee, T. Dowdell, and J. Barfett, "Image augmentation using radial transform for training deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2018, pp. 3016–3020.

[7] J. Kim, W. Jung, H. Kim, and J. Lee, "CyCNN: A rotation invariant CNN using polar mapping and cylindrical convolution layers," 2020, *arXiv:2007.10588*. [Online]. Available: http://arxiv.org/abs/2007.10588

[8] B. Kim, J. Sun, S. Kim, M. Kang, and S. Ko, "CNN-based UGS method using Cartesian-to-polar coordinate transformation," *Electron. Lett.*, vol. 54, no. 23, pp. 1321–1322, Nov. 2018.

[9] C. Esteves, C. Allen-Blanchette, X. Zhou, and K. Daniilidis, "Polar transformer networks," 2017, *arXiv:1709.01889*. [Online]. Available: http://arxiv.org/abs/1709.01889

[10] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," 2015, *arXiv:1506.02025*. [Online]. Available: http://arxiv.org/abs/1506.02025

[11] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, "DoubleU-Net: A deep convolutional neural network for medical image segmentation," in *Proc. IEEE 33rd Int. Symp. Comput.-Based Med. Syst. (CBMS)*, Jul. 2020, pp. 558–564.

[12] R. Azad, M. Asadi-Aghbolaghi, M. Fathy, and S. Escalera, "Bi-directional ConvLSTM U-Net with Densley connected convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 406–415.

[13] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[14] N. K. Tomar, D. Jha, M. A. Riegler, H. D. Johansen, D. Johansen, J. Rittscher, P. Halvorsen, and S. Ali, "FANet: A feedback attention network for improved biomedical image segmentation," 2021, *arXiv:2103.17235*. [Online]. Available: http://arxiv.org/abs/2103.17235

[15] N. Ibtehaz and M. S. Rahman, "MultiResUNet : Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Netw.*, vol. 121, pp. 74–87, Jan. 2020.

[16] D.-P. Fan, G.-P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen, and L. Shao, "PraNet: Parallel reverse attention network for polyp segmentation," in *Medical Image Computing and Computer Assisted Intervention—MICCAI*, vol. 12266, A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, D. Racoceanu, and L. Joskowicz, Eds. Cham, Switzerland: Springer, 2020, pp. 263–273.

[17] Y. Fang, C. Chen, Y. Yuan, and K.-Y. Tong, "Selective feature aggregation network with area-boundary constraints for polyp segmentation," in *Medical Image Computing and Computer Assisted Intervention—MICCAI*, vol. 11764, D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, Eds. Cham, Switzerland: Springer, 2019, pp. 302–310.

[18] C.-H. Huang, H.-Y. Wu, and Y.-L. Lin, "HarDNet-MSEG: A simple encoder-decoder polyp segmentation neural network that achieves over 0.9 mean dice and 86 FPS," 2021, *arXiv:2101.07172*. [Online]. Available: http://arxiv.org/abs/2101.07172

[19] P. Chao, C.-Y. Kao, Y. Ruan, C.-H. Huang, and Y.-L. Lin, "HarDNet: A low memory traffic network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3551–3560.

[20] J. M. J. Valanarasu, V. A. Sindagi, I. Hacihaliloglu, and V. M. Patel, "KiU-Net: Overcomplete convolutional architectures for biomedical image and volumetric segmentation," 2020, *arXiv:2010.01663*. [Online]. Available: http://arxiv.org/abs/2010.01663

[21] Q. Zhang, J. Zhou, B. Zhang, W. Jia, and E. Wu, "Automatic epicardial fat segmentation and quantification of CT scans using dual U-Nets with a morphological processing layer," *IEEE Access*, vol. 8, pp. 128032–128041, 2020.

[22] F. Commandeur, M. Goeller, J. Betancur, S. Cadet, M. Doris, X. Chen, D. S. Berman, P. J. Slomka, B. K. Tamarappoo, and D. Dey, "Deep learning for quantification of epicardial and thoracic adipose tissue from non-contrast CT," *IEEE Trans. Med. Imag.*, vol. 37, no. 8, pp. 1835–1846, Aug. 2018.

[23] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. ECCV* in Lecture Notes in Computer Science, vol. 9912. Berlin, Germany: Springer, 2016, pp. 483–499.

[24] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, p. 125, Feb. 2020.

[25] J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, D. Gil, C. Rodríguez, and F. Vilari no, "WM-DOVA maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians," *Comput. Med. Imag. Graph.*, vol. 43, pp. 99–111, Jul. 2015.

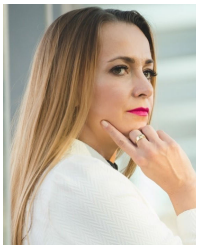
[26] P. Bilic *et al.*, "The liver tumor segmentation benchmark (LiTS)," 2019, *arXiv:1901.04056*. [Online]. Available: http://arxiv.org/abs/1901.04056

[27] N. C. F. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, and A. Halpern, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC)," in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 168–172.

[28] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Sci. Data*, vol. 5, no. 1, Dec. 2018, Art. no. 180161.

[29] É. O. Rodrigues, F. F. C. Morais, N. A. O. S. Morais, L. S. Conci, L. V. Neto, and A. Conci, "A novel approach for the automated segmentation and volume quantification of cardiac fats on computed tomography," *Comput. Methods Programs Biomed.*, vol. 123, pp. 109–128, Jan. 2016.

MARIN BENČEVIĆ received the Bachelor of Computer Engineering degree and the Master of Computer Engineering degree in information and data science from the Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, where he is currently pursuing the Ph.D. degree. He is the author and editor for an online publication of mobile application development books and tutorials. His research interests include image processing and computer vision in medical images focused on the human cardiovascular systems.



IRENA GALIĆ was born in Osijek, Croatia. She received the Diploma degree in mathematics and computer science from J. J. Strossmayer University of Osijek, Croatia, in 1999, the M.Sc. degree in computer science from Saarland University, Saarbrücken, Germany, in 2004, and the Ph.D. degree from J. J. Strossmayer University of Osijek, in 2011. She is currently a Professor of computer science with the Faculty of Electrical Engineering, Computer Science, and Information Technology, J. J. Strossmayer University of Osijek. Her research interest includes visual computing.

MARIJA HABIJAN is currently pursuing the Ph.D. degree in medical image processing. Since 2018, she has been working as a Junior Researcher and Teaching Assistant with the Faculty of Electrical Engineering, Computer Science and Information Technology Osijek. Her research interests include image processing, computer vision, and deep learning. Her work mainly promotes the development of medical image processing and analysis applications on deep learning.

DANILO BABIN received the Master of Science degree in telecommunications and signal processing from the University of Novi Sad, Serbia, in 2007, and the Ph.D. degree in computer science engineering from Ghent University, Belgium, in 2013. The topic of his Ph.D. dissertation was "Segmentation and Skeletonization Techniques for Cardiovascular Image Analysis." Since 2014, he has been a Postdoctoral Researcher in charge of medical image analysis with the Image Processing and Interpretation Group, Ghent University. Since 2016, he has been affiliated with UGent-imec as a Postdoctoral Researcher. His research interests include medical image analysis, processing of cardiovascular images, image segmentation, and skeletonization.