Semi-supervised Task-driven Data Augmentation for Medical Image Segmentation

Krishna Chaitanya, Neerav Karani, Christian F. Baumgartner, Ertunc Erdil, Anton Becker, Olivio Donati, Ender Konukoglu

***Abstract*—Supervised learning-based segmentation methods typically require a large number of annotated training data to generalize well at test time.In medical applications, curating such datasets is not a favourable option because acquiring a large number of annotated samples from experts is time-consuming and expensive. Consequently, numerous methods have been proposed in the literature for learning with limited annotated examples. Unfortunately, the proposed approaches in the literature have not yet yielded significant gains over random data augmentation for image segmentation, where random augmentations themselves do not yield high accuracy. In this work, we propose a novel task-driven data augmentation method for learning with limited labeled data where the synthetic data generator, is optimized for the segmentation task. The generator of the proposed method models intensity and shape variations using two sets of transformations, as additive intensity transformations and deformation fields. Both transformations are optimized using labeled as well as unlabeled examples in a semi-supervised framework. Our experiments on three medical datasets, namely cardiac, prostate and pancreas, show that the proposed approach significantly outperforms standard augmentation and semi-supervised approaches for image segmentation in the limited annotation setting. The code is made publicly available at https://github.com/krishnabits001/task_driven_data_augmentation.**

***Index Terms*—Data augmentation, semi-supervised learning, machine learning, deep learning, medical image segmentation**

I. Introduction

Accurate image segmentation is important for many clinical applications that rely on medical images. In the recent years, deep neural networks have been successful in yielding high segmentation performance at the expense of requiring large amount of annotated training data. Obtaining many annotated examples is difficult for medical images since getting clinical experts to annotate a large number of segmentation masks, which require per-pixel annotations, is an expensive and time-consuming process. Hence, it is not a preferable solution in clinical settings. At the heart of this issue lies the fundamental gap between generalization performance of humans and current Deep Learning (DL) methods. While humans can generalize well for image segmentation after observing very few examples, even one or two examples seem to suffice in some applications, this is not the case with the current DL algorithms. In this work, we focus on algorithmic approaches aiming to close the mentioned gap for medical image segmentation.

KC, NK, CFB, EE and EK are with the Computer Vision Laboratory, ETH Zurich, Switzerland. Email: {krishna.chaitanya}@vision.ee.ethz.ch. AB and OD are with the University Hospital of Zurich, Switzerland.
Accepted at Medical Image Analysis Journal, 2020.

In the past works, it has been observed that large annotated data are needed for deep neural networks to obtain high segmentation accuracy. When trained with a large number of annotated samples, the target relationships learned by the neural networks between images and segmentation masks become robust to the variations in shape and intensity characteristics of the target and surrounding structures. While algorithms trained on a small number of annotated samples, having not been exposed to a sufficient amount of variation, perform poorly on unseen test images that contain variations not observed during training. These variations in shape arise due to anatomical variations in the population and variations in intensity characteristics are due to differences in (i) tissue properties and its composition, and (ii) image acquisition and scanner protocols, especially in Magnetic Resonance Imaging (MRI).

To address the need for a large number of annotated data to achieve high segmentation performance with DL methods, in this work we propose a task-driven and semi-supervised data augmentation method (shown in Fig. 1). The method is based on learning generative models that can be used to sample deformation fields and additive intensity transformations. Segmentation cost is included during training of these models for the synthesis of image-label pairs, which incorporate the task-driven nature. Semi-supervised nature, on the other hand, is incorporated by including unlabeled data in the training through an adversarial term, which help generators synthesize diverse set of shape and intensity variations present in the population, even in scenarios where the number of labeled examples are extremely low.

The proposed approach in essence aims to optimize the augmentation task and can be intuitively understood by drawing analogies with existing augmentation methods. For example, if we consider random elastic deformations proposed in [1], [2], the augmentation is based on a deformation model with a few number of parameters like Gaussian kernel size and width. These parameters can be seen as hyper-parameters and their values can be optimized by training separate segmentation networks for a number of combinations, and selecting the combination that yields the best performance on validation images. Our approach uses a much more flexible transformation model using networks and optimizes its weights using both labeled and unlabeled training images. Its only hyper-parameter is the number of training iterations which is determined based on performance on validation images. Thus, both the training and validation images play a crucial role. In our experiments, we evaluated the proposed approach using three different publicly

available datasets of cardiac, prostate and pancreas. We present comparisons with existing alternatives as well as an ablation study empirically analyzing the proposed approach.

A preliminary version of this work has been presented at the conference on Information Processing in Medical Imaging [3]. In this extended version we additionally:

- analyze quantitatively how each term of regularization loss, namely adversarial loss and large deviation loss components affects the performance gains obtained in the proposed method (refer to results in Sec 5.A).
- investigate the benefit of optimizing the generator jointly with the segmentation network as compared to independent optimization of generator w.r.t segmentation network (refer to results in Sec 5.B).
- examine the generality of the proposed method by evaluating it on two more datasets, namely prostate and pancreas.
- compare with a larger set of related methods including self-training with conditional random fields and image-level adversarial training.

A. Related work

We broadly classify the relevant literature into two categories in light of the proposed method in this work:

Data Augmentation: Data Augmentation is a simple technique to enlarge the training set based on generating synthetic image-label pairs. The idea is to transform the images in such a way that the label remains the same, or the transformation is well-defined for both the image and the label. The popular approaches are random affine transformations [4], random elastic transformations [1], [2] and random contrast transformations [5], [6]. These methods are very simple to implement and empirically have been shown to reduce overfitting and improve performance on unseen examples. Several recent works trained Generative Adversarial Networks(GANs) [7], using the existing labeled dataset to generate realistic image-label pairs. The idea has been applied to various analysis tasks [8]–[14] including MR image segmentation [15]–[17]. MixUp [18] is different from the above approaches in the fashion that the generated data is not realistic in nature. Here, the additional data is obtained by linearly interpolating available images and corresponding labels, respectively. Despite the unrealistic nature, it seems to improve the performance of the neural networks on standard benchmark datasets [18] and also on medical image segmentation [19] in limited annotation setting. All of the above mentioned methods have parameters that control the generation process. These are either set by experience or learned to generate realistic samples based on the available labeled examples, which itself often requires large number of training samples. None of the methods optimize the parameters with respect to the task performance nor leverage unlabeled data. The proposed approach optimizes the parameters of the generator to get the best segmentation performance and leverages unlabeled data during the optimization.

The closest work to ours was proposed in [20]. In a meta-learning setup, the authors proposed a fully supervised approach that incorporated the classification performance in learning the generator so to generate augmented images that are optimal for the task. Although it is a few-shot learning method, we still need a large number of different classes samples to train the generator. Here, no modeling assumptions are considered in the generator setup, and the augmented images are synthesized directly. While we instead incorporate domain knowledge and model the generator to output deformation and intensity transformations to generate the synthetic images. Due to these assumptions of modeling transformations as well as leveraging the unlabeled data in the generation process, we do not need a large number of labeled examples during training. In the proposed method, we can readily obtain the synthetic mask using the generated transformation which cannot be obtained with the method [20]. We show in the results (Section V and Fig. 3a) that the semi-supervised framework of our method yields significant improvements over using only the labeled examples with task-specific loss as in [20]. In a concurrent work [21], the authors propose a method primarily applicable to classification tasks, where an exhaustive manifold exploration is done to learn affine geometric transformations to find the augmented samples near the decision boundary of the classes. Later, these augmented samples are used with existing training samples to re-train the network, which led to an increase in the robustness (higher accuracy) of the deep learning models. Here, only geometric variations are explored compared to the proposed work where both geometric and intensity variations are addressed. Also, instead of optimizing for the task performance as in our model, here authors use augmentation to reinforce the decision boundaries by generating samples near decision boundaries.

Semi-supervised learning (SSL) methods leverage unlabeled data to accompany limited annotated data during training. The underlying idea is to regularize training by leveraging the unlabeled data and avoid overfitting. We divide SSL works into 3 sub-categories: (i) self-training, (ii) adversarial training, and (iii) learned registration based approaches.

Self-training approaches [22], [23] are based on the concept of iteratively re-training an already-trained network on the estimated labels (pseudo-labels) of unlabeled data as ground truths. In [24] authors show an improvement in the segmentation performance on cardiac MRIs using a self-training approach along with a Conditional Random Field (CRF) model post-processing intermediate predictions before re-training. However, it has also been shown that the self-training approaches can suffer if the initial predictions are erroneous on the unlabeled data [25], [26].

Adversarial training and GANs have been used in the semi-supervised setting both using the generator as the segmentation network and the discriminator in a regularization term [27], and vice-versa [28]. In limited annotation setting, we compare and illustrate that the proposed method outperforms the earlier stated semi-supervised approaches.

Alternatively, in a concurrent work [29], the authors proposed a one-shot data augmentation approach that learns registration between unlabeled image and labeled image where two independent models are trained to learn spatial and appearance transformations for the registration. Later, both learned models are used to generate augmented image-label

pairs which are used to train a segmentation network. As in the other augmentation works, the generator of this model is also not optimized to yield the best task performance. Furthermore, the approach relies on image registration, which, on one hand, is a difficult task for non-brain anatomy, and on the other hand, leads to task-irrelevant background structures substantially influence the augmentation process.

Lastly, with **weakly-supervised learning** the issue of expensive and time-consuming pixel-wise annotations is addressed using weaker labels during training such as scribbles [30] and image-wide labels [31], which are different approaches that can be complementary to data-augmentation.

II. Methods

Let X_L be a set of training images and Y_L be the set of corresponding ground truth segmentation labels, S be a segmentation network and w_s be its trainable parameters. In the supervised learning setting, a loss function $L_s\left(X_L, Y_L\right)$ is defined as the disagreement between the labels predicted by the network S for the set of input images X_L and ground truth labels Y_L. The objective is to minimize the loss function L_s with respect to the parameters w_s as can be stated as in Eq. 1.

$$\min_{w_S} L_S(X_L, Y_L) \tag{1}$$

When data augmentation is used, the minimization becomes

$$\min_{w_S} L_S(X_L \cup X_G, Y_L \cup Y_G) \tag{2}$$

where X_G and Y_G denote the sets of generated images and corresponding labels, which can be generated using the transformations mentioned previously. In this minimization, the effective training set is formed of the augmented sets $X_L \cup X_G$ and $Y_L \cup Y_G$. When model-based transformations are used for generating the augmentation sets X_G and Y_G, such as geometric or contrast transformations, each augmented image $x_G \in X_G$ and the corresponding label $y_G \in Y_G$ are created by a conditional generator function that takes as input an existing labeled pair and applies a random transformation with fixed parameters. We can represent this with the following notation, as in Eq. 3.

$$\begin{aligned} &(x_G, y_G) = G\left((x_L, y_L), z; w_G\right), \\ &(x_L, y_L) \sim p\left(X_L, Y_L\right), \ z \sim p(z), \end{aligned} \tag{3}$$

where (x_L, y_L) are the image-label pair sampled from the set of labeled examples, $G(\cdot, \cdot; w_G)$ is the transformation function, z is the random component of the transformation and w_G are the parameters of the transformation. For instance, for the random elastic deformations proposed in [2], G would be the deformation model, w_G would include the grid spacing between anchor points and standard deviation of the distribution of displacement vectors at each anchor point, while z would correspond to a random draw of displacement vectors. The same transformation would be applied to both x_L and y_L to generate the augmented pairs. As described in the introduction, in the model-based transformations, parameters are often predefined and their number is kept low.

When GANs are used for generation, a neural network is used as the generator as $(x_G, y_G) = G(z; w_G)$, and its parameters are determined by optimizing Eq. 4 as per the adversarial learning framework [7], using an additional discriminator network D with its own set of parameters w_D. (The GAN can also be a conditional image generator defined as $G(z, x_L; w_G)$ and the discussion still holds.)

$$\begin{aligned} &\min_{w_G} \max_{w_D} \mathbb{E}_{x,y \sim p(X_L, Y_L)}[\log D(x, y; w_D)] + \\ &\mathbb{E}_{z \sim p(z)}[\log(1 - D(x_G, y_G; w_D))], \ (x_G, y_G) = G(z; w_G) \end{aligned} \tag{4}$$

The generator G is input only with a random draw from the distribution $p(z)$ to output an image-label pair (x_G, y_G). The labeled pairs are still used but during the training. The discriminator D is optimized to distinguish between generated pairs from G and real pairs (x_L, y_L), while the generator G is optimized to produce (x_G, y_G) such that D can not differentiate between generated and real pairs. This forces the generated set (X_G, Y_G) to be as "realistic" as possible.

In both model-based and GAN-based approaches, generators would be pre-defined or trained in advanced without considering the task, and during training random draws would be sampled from the generator models to create X_G and Y_G.

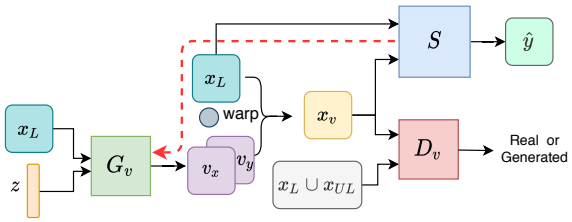
A. Semi-Supervised and Task-Driven Data Augmentation

In this work, we propose to generate augmented image-label pairs that are optimized for the segmentation task (Fig. 1) and the method is summarized in Algorithm 1. We achieve this by optimizing the generator function, similar to the GAN-based approach, but with a crucial difference, we integrate the task loss and leverage unlabeled images in the process. The proposed model optimizes Eq. 5 instead of Eq. 4.

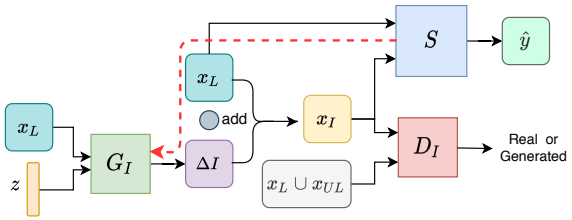
$$\min_{w_G} \big(\min_{w_S} L_S\left(X_L \cup X_G, Y_L \cup Y_G\right) + L_{\text{reg},G}(X_{UL}) \big), \tag{5}$$

where X_{UL} denotes a set of unlabeled images. Furthermore, we would like to be able to optimize the parameters with limited number of labeled examples. To this end, we integrate domain knowledge into the generation process, similar to the model-based approaches, but allowing a network parameterization of the transformation models to increase flexibility while remaining trainable. We define two conditional generators to model shape and intensity variations using *deformation fields generator* (non-affine spatial transformations) and *intensity fields generator*, respectively. Crucially, both models are constructed such that the segmentation mask y_G corresponding to an augmented image x_G is obtained by applying the same transformation to the input image mask y_L.

With this optimization, we want to incorporate two sets of ideas with the two loss-terms in Eq. 5. The first term ensures that the model generates set of augmented pairs (X_G, Y_G) such that they are helpful for the minimization of segmentation loss, which is the *task-driven* nature of the approach. The second term is a regularization term built on adversarial loss and integrates a preference for larger transformations leveraging the unlabeled images in the generation process, which is the *semi-supervised* component of the method. Also, when using only segmentation loss (task-driven component), generators' may produce images that are easy to segment. Hence, to



(a) Deformation Field conditional GAN



(b) Additive Intensity Field conditional GAN

Fig. 1: Data augmentation modules that generate augmented image-label pair with task-driven optimization defined in a semi-supervised framework. Here, the dotted-red line indicates the inclusion of segmentation loss for generator optimization.

prevent this case, it becomes crucial to use the regularization loss that comprises of adversarial loss and large deformations loss. Adding large deformations loss and adversarial loss ensures that the deformations and intensity fields generated are larger, and the generated images match the distribution of the unlabeled set. This regularization ensures that the generated images have large changes compared to the input labeled image while also being realistic yet non-trivial to segment for segmentation network.

1) Deformation Field Generator: The deformation field generator G_V is trained to output a deformation field transformation. The conditional generator G_V, a network with parameters w_{G_V}, takes as input a labeled image x_L and a z vector randomly drawn from a unit Gaussian distribution to produce a dense per-pixel deformation field $\mathbf{v} = G_V(x_L, z; w_{G_V})$. Later, the input image and its corresponding label (in one-hot encoding form) are warped using bi-linear interpolation based on the generated deformation field $\mathbf{v}$ to produce the augmented image-label pair x_{G_V} and y_{G_V}, respectively. The augmented image and label sets are denoted by X_{G_V} and Y_{G_V}, and each sample pair is defined using the following: $x_{G_V} = \mathbf{v} \circ x_L$, $y_{G_V} = \mathbf{v} \circ y_L$, where $\circ$ denotes warping operation.

2) Additive Intensity Field Generator: Similar to G_V, here the generator G_I is trained to output an additive intensity mask transformation. The generator G_I, again a network with parameters w_{G_I}, takes as input a labeled image from x_L and a z vector randomly drawn from a unit Gaussian distribution to output an additive intensity mask $\Delta I = G_I(x_L, z; w_{G_I})$. Then ΔI is added to the input image x_L to obtain the augmented image x_{G_I}, and its corresponding segmentation mask y_{G_I} remains unchanged as the initial mask y_L. The augmented image-label set is denoted by $\{X_{G_I}, Y_{G_I}\}$ and one sample pair is defined using the following: $x_{G_I} = x_L + \Delta I$, $y_{G_I} = y_L$.

3) Regularization Loss: The regularization term L_{reg} is defined as in Eq. 6 for both conditional generators.

$$L_{\text{reg},G_C} = \lambda_{\text{adv}} L_{\text{adv},G_C} + \lambda_{\text{LD}} L_{\text{LD},G_C}, \text{ for } C = V, I. \quad (6)$$

The first term is the following standard adversarial loss

$$L_{\text{adv},G_C} = \max_{w_{D_C}} \mathbb{E}_{x \sim p(X_U \cup X_{UL})}[\log D_C(x; w_{D_C})] + \mathbb{E}_{z \sim p(z), x_L \sim p(X_L)}[\log(1 - D_C(G_C(x_L, z; w_{G_C}); w_{D_C}))] \quad (7)$$

where D_C, w_{D_C} denotes the discriminator network and its weights. This first term incorporates the semi-supervised nature of the model by including the set of unlabeled images X_{UL} in the optimization. It allows samples in the unlabeled set that show different shape and intensity variations than the labeled examples guide the optimization of the generators. The second term in Eq. 6 is the *Large Deviation* (LD) term that embeds a preference for large transformations. The LD term prevents the generator from producing relatively smaller deformations and intensity fields, which would satisfy the adversarial loss as well as lead to lower segmentation loss in the cost given in Eq. 5. The definition of L_{LD,G_C} depends on the generator type. We define the two terms for the deformation and intensity field generators as $L_{\text{LD},G_V} = -\|\mathbf{v}\|_1$ and $L_{\text{LD},G_I} = -\|\Delta_I\|_1$. The negative signs ensures that minimizing the LD term maximizes the L_1 norms. The LD term forces the generator to produce large transformations while the adversarial loss tries to constrain them. Optimizing for both terms yield augmented images that differ substantially from the input labeled images for both the generators.

Finally, the weighting terms λ_{adv} and λ_{LD} balances the effect of the two terms on the optimization.

Discriminator: The discriminator's role is to ensure that the augmented images obtained by applying transformations produced by the generators are indistinguishable from real unlabeled images in the distribution sense. This loss competes against segmentation loss to ensure that the generated images are relatively realistic and are not too easy to segment images either.

4) Optimization Sequence: Before the optimization of the conditional generators, the segmentation network (S) in the proposed setup is pre-trained on only the labeled data for a few epochs as per Eq. 1 and later optimized with both the labeled and generated data from the conditional generators as per Eq. 2. This is done to ensure that S produces reasonable segmentation masks for the labeled data when the optimization of the generators begins. Following the pre-training of S, both generative models for deformation and intensity fields are trained separately by minimizing the cost given in Eq. 5 with corresponding regularization terms. Note that this minimization trains over S, the generators and the discriminators. This implies training of the set of networks (S, G_V, D_V) and (S, G_I, D_I) independently for deformation and intensity fields generation, respectively. The optimization sequence is run for a fixed number of iterations, and the segmentation network is evaluated on the validation images using the Dice's similarity coefficient (DSC) at every iteration. Once the optimization is complete, we fix both the generators G_V and G_I with the

parameters that yielded the best mean DSC over the validation images. Then, the segmentation network S is optimized using Eq. 2 once again from a random initialization. The data used for this final training comprises of both the original labeled sets, X_L and Y_L, and the augmented sets, X_G and Y_G sampled from the trained generators G_V or G_I or both (where we perform two back to back transformations, e.g. G_I is applied over the deformed image obtained from G_V).

The validation images play a crucial role in the defined optimization as they determine the parameters of the generator model chosen to generate the augmented data that is later used for the independent optimization of the segmentation network.

Algorithm 1: Training steps of the proposed method

Available training data: labeled set (X_L, Y_L) and unlabeled set (X_{UL}).

Step 1:

(a) train deformation field generator (G_V) as per Eq. (5) using the available data.

(b) train intensity field generator (G_I) as per Eq. (5) using the available data.

Step 2:

Post optimization, the generators are used to sample augmented image, label pairs conditioned on the labeled set.

(a) The shape transformed image, label pairs (X_{G_V}, Y_{G_V}) are sampled using generator (G_V).

(b) The intensity transformed image, label pairs (X_{G_I}, Y_{G_I}) are sampled using generator (G_I).

(c) The image, label pairs that contain both the shape and intensity transformations (denoted by $(X_{G_{VI}}, Y_{G_{VI}})$) are obtained by inputting the sampled shape transformed image, label pairs (X_{G_V}, Y_{G_V}) from (G_V) through (G_I).

Step 3:

Train the segmentation network with all the available labeled and generated augmented data. The training set includes: original labeled data (X_L, Y_L), affine transformed data (X_{Aff}, Y_{Aff}), shape transformed data (X_{G_V}, Y_{G_V}), intensity transformed data (X_{G_I}, Y_{G_I}) and data with both shape and intensity transformations applied $(X_{G_{VI}}, Y_{G_{VI}})$.

III. Dataset and Network details

A. Dataset details

Cardiac Dataset: This is a publicly available dataset hosted as part of MICCAI'17 ACDC challenge [32] [1]. It comprises of 100 subjects' short-axis cardiac cine-MR images captured using 1.5T and 3T scanners. The in-plane resolution ranges from 0.70x0.70mm to 1.92x1.92mm and through-plane resolution ranges from 5mm to 10mm. The segmentation masks are provided for left ventricle (LV), myocardium (Myo) and right ventricle (RV) for both end-systole (ES) and end-diastole (ED) phases of each subject. This dataset is divided into 5 sub-groups (details in [32] [2]), each comprising of 20 subjects, respectively. For our experiments in the main article, we only used the ES images, which was a random choice. The evaluation of ED images for the proposed method are presented in table III in the appendix.

Prostate Dataset: This is a public dataset made available as part of MICCAI'18 medical segmentation decathlon challenge [3]. It comprises of 48 subjects T2 weighted MR scans of prostate. The in-plane resolution ranges from 0.60x0.60mm to 0.75x0.75mm and through-plane resolution ranges from 2.99mm to 4mm. Segmentation masks comprise of two adjoint regions: peripheral zone (PZ) and central gland (CG).

Pancreas Dataset: This dataset also is from medical decathlon MICCAI'18 challenge [4]. It comprises of 282 subjects CT scans. Segmentation masks available comprise of labels with large (background), medium (pancreas) and small (tumor) structures. The in-plane resolution ranges from 0.6x0.6mm to 0.97x0.97mm and through-plane resolution ranges from 0.7mm to 7.5mm. In this work, we create two labels for segmentation task, where label 1 denotes the foreground which was created by merging the labels of pancreas and tumor, and label 0 denotes the background label.

B. Pre-processing

N4 [33] bias correction was performed on the cardiac and prostate datasets. The below pre-processing was applied to all images of all the datasets. (i) intensity normalization: all the volumes were normalized using min-max normalization according to: $(x - x_2)/(x_{98} - x_2)$, where x_2 and x_{98} denote the 2^{nd} and 98^{th} intensity percentiles of the 3D volume. (ii) re-sampling: all 2D image slices and their corresponding label maps were re-sampled to a fixed in-plane resolution r using bi-linear and nearest-neighbour interpolation, respectively and then cropped or padded with zeros to a fixed size of f_z. The resolution r and fixed size f_z were chosen empirically for each dataset, and the values were: (a) cardiac dataset: r =1.367x1.367mm and f_z =224x224, (b) prostate dataset: r =0.6x0.6mm and f_z =224x224, (c) pancreas dataset: r =0.8x0.8mm and f_z =320x320.

C. Network Architecture

There are 3 types of networks in the proposed method (see Fig. 1): a segmentation network S, a discriminator network D and a generator network G. We describe the architectures of these networks below. G_V and G_I use the same architecture except at the last layer, which are used to model the deformation field and the intensity mask, respectively.

Generator: Generator G takes an image x_L and a randomly drawn z vector of dimension 100 as input. Both inputs are initially passed through 2 sub-networks namely $G_{subnet,X}$ and $G_{subnet,z}$. $G_{subnet,X}$ comprises of 2 convolutional layers and $G_{subnet,z}$ comprises of a fully-connected layer, followed by reshaping of output into down-sampled image dimensions. Then a set of bi-linear upsampling and convolutional layers are

[1] https://www.creatis.insa-lyon.fr/Challenge/acdc

[2] https://www.creatis.insa-lyon.fr/Challenge/acdc

[3] http://medicaldecathlon.com/index.html

[4] http://medicaldecathlon.com/index.html

applied consecutively to output feature maps of image dimensions. The resulting outputs of both the sub-networks, which are of same dimensions, are then concatenated and passed through a common sub-network $G_{subnet,common}$, which consists of 4 convolutional layers where the last layer is different for G_V and G_I. The final convolution for G_V yields two feature maps that correspond to dense per-pixel deformation field $\mathbf{v}$, while for G_I, it outputs a single feature map that corresponds to the intensity mask ΔI. The final layer of G_I uses $tanh$ activation to restrict the range of values in the intensity mask. All the convolutional layers except the final layers are followed by batch normalization layers and ReLU activation. All convolutional layers' kernels are 3x3 except the final layers' which are 1x1.

Discriminator: D has an architecture similar to the DCGAN [34], which comprises of five convolutional layers each with a kernel size of 5x5 and a stride of 2. The convolutions are followed by batch normalization layers and leaky ReLU activations with the leak value of the negative slope set to 0.2. After the convolutional layers, the output is reshaped and passed through three fully-connected layers where the final layer has an output size of 2 that predicts the probability of the input being real or fake.

Segmentation Network: We chose the architecture of segmentation network S similar to U-Net [2]. It comprises of encoding and decoding paths. The encoder comprises of four convolution blocks where each block has two 3x3 convolutions followed by a 2x2 max-pool layer. The decoder comprises of four convolution blocks where each block comprises of the concatenation of features from the corresponding level of the encoder, followed by two 3x3 convolutions and bi-linear upsampling by a factor of 2. Except for the last layer, all the layers have batch normalization and ReLU activation.

D. Training Details

The segmentation loss (L_S) used is weighted cross entropy. We empirically set the weights of background pixels as 0.1 and foreground pixels as 0.9 since the number of pixels belonging to the foreground are fewer in quantity and are of primary interest for the segmentation task at hand. For the datasets with more than one foreground label, we divided the foreground weight of 0.9 equally among all the labels. With this rationale, the weights of the output labels are as follows: (a) 0.1 for background and 0.3 for each of the three foreground classes of the cardiac dataset, (b) 0.1 for background and 0.45 for each of the two foreground classes of the prostate dataset, and (c) 0.1 for background and 0.9 for one foreground class of the pancreas dataset.

We split the data into training pool, validation, unlabeled and test sets. We empirically set λ_{adv} as 1 to match the magnitude of adversarial loss to the segmentation loss. To determine λ_{LD} parameter, we randomly sampled one 3D volume from the training pool of cardiac dataset (the sampled volume is one possibility of all training combinations used in full analysis) and trained the network, and evaluated performance on the validation images for three values of λ_{LD}: $10^{-2}, 10^{-3}, 10^{-4}$. The value of 10^{-3} for λ_{LD} yielded the best validation performance for this experiment. So, we used this set values ($\lambda_{adv} = 1, \lambda_{LD} = 10^{-3}$) for all future experiments on all datasets. Owing to the computationally expensive nature of the proposed method, we did not perform an exhaustive grid search on many combinations of weights of the loss terms (λ_{adv}, λ_{LD}) on the validation set. But if one has enough computational resources, one could do a grid search of these hyper-parameters for each dataset to potentially obtain higher performance gains. The training of the GANs to convergence with limited data with stability can be complicated, as seen in many prior applications. We did not face any major training issues as: (1) we incorporated the concluding remarks from earlier works like DCGAN [34] into the designing of our generator and discriminator networks and choosing appropriate respective learning rates, and also here (2) the generators are trained to generate deformation/intensity fields instead of images directly as done in earlier works, which could be relatively easier to train.

The batch size and the total number of iterations are set as 20 and 10000, respectively, based on the evolution of the training curves. For all the networks, while training, the iteration where the model yields the best performance on the validation images is saved for the evaluation. AdamOptimizer [35] is used for the optimization of all the networks with learning rate of 10^{-3} and default beta values ($\beta_1 = 0.9$, $\beta_2 = 0.999$).

IV. Experiments

We evaluated the proposed method on three datasets: cardiac, prostate and pancreas. For each dataset, we split the data into 4 sets: labeled training ($X_{L,total}$), unlabeled training (X_{UL}), test (X_{ts}) and validation (X_{vl}). The size of each set is denoted by N followed by a subscript indicating the set. The validation set consists of two 3D volumes (N_{vl}=2) for all datasets. For the cardiac, prostate and pancreas datasets, the number of 3D volumes (N_{UL}, N_{ts}) for unlabeled and test sets are (25, 20), (20, 13) and (25, 20) respectively. X_{UL}, X_{ts} and X_{vl} sets are selected randomly a-priori and fixed for all experiments. The remaining 3D volumes constitute the training pool $X_{L,total}$. Note that the entire training pool is never utilized for training. Rather a small number of training images (N_L) is sampled from $X_{L,total}$ for each experiment. As the interest of this work is to analyze the performance in the limited annotation setting, we set $N_L = 1$ or 3. Each experiment is run five times with different 3D training volumes. Further, to account for the variations in the random initialization and convergence of the networks, each of the five experiments is run three times. Thus, overall, we have 15 runs for each experiment. Since the cardiac dataset has five sub-groups of images (see Sec. III-A), we ensure that each set contains an equal number of images from each sub-group, and, when $N_L = 1$ is run five times, each time the 3D volume is selected from a different sub-group.

Segmentation performances of the following models were compared:

No data augmentation (no aug): S is trained without any data augmentation.

Affine data augmentation (Aff): S is trained with data augmentation comprising of affine transformations such as: (a) scaling (random scale factor is chosen from a uniform distribution with min and max value as 0.9 and 1.1), (b) flipping along x-axis, (c) rotation (randomly a value is chosen between -15 and +15 degrees and another type of rotation that is multiple of 45 degrees (defined as 45 deg*N where N is randomly chosen between 0 and 8)). For each slice in the batch, we apply one of the above random transformation 80% of the time and 20% of the time we use the image as is.

For all the subsequent data augmentation methods, MixUp and semi-supervised methods, we include the random affine data augmentations by default as described above. For training of S, half of each batch was composed random affine augmentation and the remaining half was chosen from the specific augmentation technique.

Random elastic deformations (RD): Random elastic augmented images are created as stated in [2], where a deformation field is created using a matrix of size 3x3x2. Each element of this matrix is sampled from a Gaussian distribution with a mean of 0 and standard deviation (sigma) of 10 and is then re-sampled to image dimensions using bi-cubic interpolation. (refer to Fig. 6 in the appendix for results with different sigma & matrix sizes)

Random contrast and brightness fluctuations [5], [6] **(RI)**: These augmented images are created with the help of contrast adjustment step ($x = (x - \bar{x}) * c + \bar{x}$) and brightness adjustment step ($x = x + b$), where c and b are sampled uniformly from [0.8,1.2] and [-0.1,0.1], respectively and $\bar{x}$ denotes mean of 2D image. (refer to Fig. 7 in the appendix for more combinations of c and b evaluated)

Deformation field transformations (GD): The deformation field generator trained with the proposed method G_V is used to generate the augmented data i.e., X_{G_V}.

Intensity field transformations (GI): The intensity field generator trained with the proposed method G_I is used to generate the augmented data i.e., X_{G_I}.

Both deformation and intensity field transformations (GD+GI): Augmented data included both X_{G_V} and X_{G_I}, obtained from the generators G_V and G_I, respectively. We also generated additional images $X_{G_{VI}}$ which have both the deformation and intensity transformations. These are obtained by applying intensity transformation using generator G_I on the deformation field transformed images X_{G_V}. The augmented data consists of all the images generated $\{X_{G_V}, X_{G_I}, X_{G_{VI}}\}$.

MixUp [18] **(Mixup)**: The augmentation sets X_G and Y_G consist of the linear combination of available labeled images using the Mixup formulation as stated in Eq. 8 [18].

$$x_{Gi} = \lambda x_{Li} + (1 - \lambda) x_{Lj}, \quad y_{Gi} = \lambda y_{Li} + (1 - \lambda) y_{Lj} \quad (8)$$

where λ is sampled from beta distribution Beta(α, α) with $\alpha \in (0, \infty)$ and $\lambda \in [0, 1)$. The α value of 0.2 yielded the best results for the datasets considered. λ controls the ratio of mixing of two image-label pairs (x_{Li}, y_{Li}), (x_{Lj}, y_{Lj}) which are randomly sampled from the labeled image set.

Mixup over deformation and intensity field transformations (GD+GI+Mixup): These set of images are obtained by applying Mixup over all possible pairs of available images: original data (X_L), their affine transformations and the generated images using deformation and intensity field generators

Adversarial Training (Adv_tr): For comparison, we investigate previously proposed adversarial training methods with the discriminator trained to operate on: (i) the image level discrimination [27] and (ii) the pixel level discrimination [28] in a semi-supervised (SSL) setting.

Self-training (Self_tr): The self-training based method as proposed in [24] is evaluated on the datasets.

Ablation Studies: In addition to the aforementioned comparisons, we carried out additional ablation studies as described below. These studies were done only on the cardiac dataset owing to lack of computational resources, as each experiment for each ablation study and dataset requires 15 runs.

A. Effects of adversarial (λ_{adv}) and large deviation (λ_{LD}) loss terms of regularization loss on segmentation performance: We investigate the effect of each term of regularization on the performance of the proposed method. Different values of λ_{adv} and λ_{LD} are considered to examine how much each term impacts the performance. The training case of $\lambda_{adv} = \lambda_{LD} = 0$ is similar to earlier work [20].

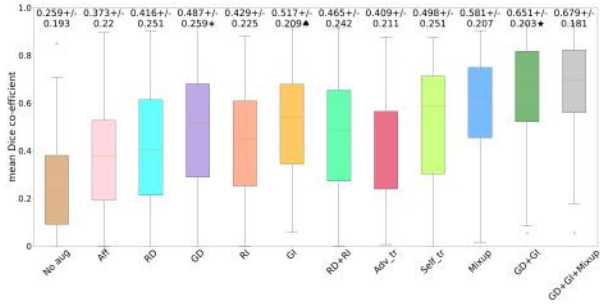
B. Independent optimization of the generator and the segmentation networks: Here, we optimize both G_V and G_I without the segmentation loss similar to [16], [17]. Later, augmented data created from these optimized generators are used for the independent training of the segmentation network. This experiment reveals the value of the segmentation loss.

C. Varying the number of unlabeled data: We used different number of unlabeled volumes in the training of the generators. The number of 3D volumes studied (N_{UL}) were: 1, 3, 5, 10, 20, 25 and 50.

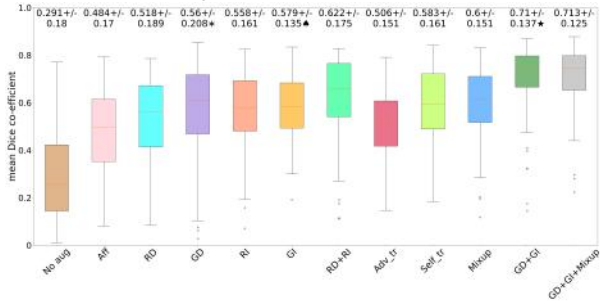
D. Varying the number of labeled 3D volumes used in training: The number of 3D volumes considered (N_L) were: 1, 3, 5, 10, 15, 40. This experiment is done to study if any improvement in Dice score is obtained when a large number of annotated volumes are available for training.

E. Different set of train, validation, test and unlabeled 3D volumes: We randomly sample another training, validation, test, and unlabeled image sets from the cardiac dataset different from the earlier sets and re-run learning deformation and intensity field transformations for this new set for one 3D training volume case ($N_L = 1$). This experiment is done to analyze if the proposed method overfits to a specific dataset split or generalizes for any split.

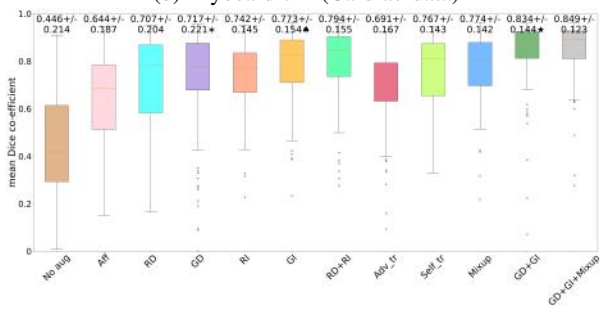
F. No validation images: Lastly, we report the performance observed when we do not use any validation images in the training of the generators. In this case, the training is stopped after running the model for some predefined number of iterations and these model parameters are used for generating images for augmentation.



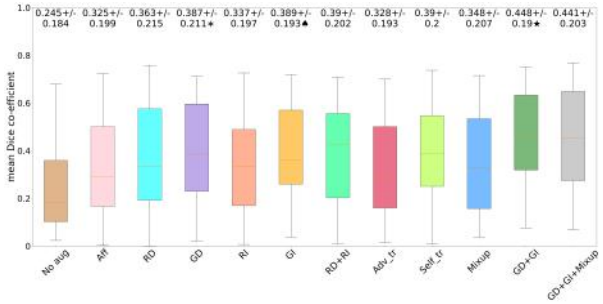
(a) Right Ventricle (Cardiac data)



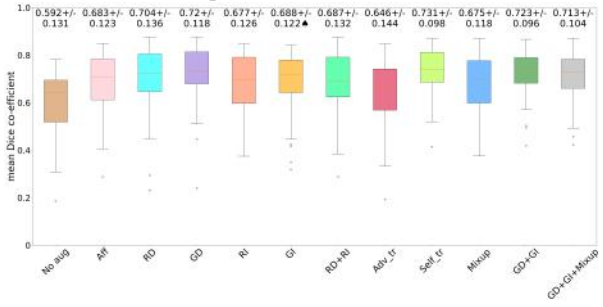
(b) Myocardium (Cardiac data)



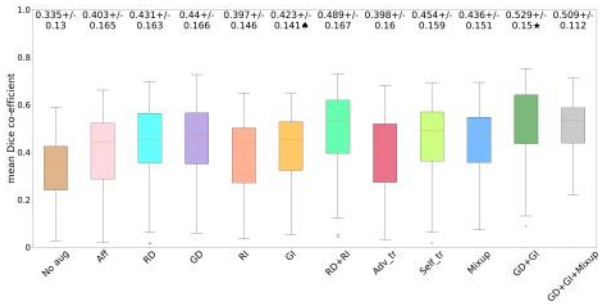
(c) Left Ventricle (Cardiac data)



(d) Peripheral Zone (Prostate data)



(e) Central Gland (Prostate data)



(f) Pancreas+Tumor (Pancreas data)

Fig. 2: The segmentation performance of the proposed augmentation method (GD+GI) and several relevant works for three datasets are presented using Dice score (DSC). The number of labelled 3d volumes (N_L) used for training on cardiac, prostate, and pancreas datasets is one, one, and three, respectively (mean DSC and standard deviation values are reported on top of each boxplot). $*, \spadesuit, \star$ denotes the statistical significance of GD over RD, GI over RI, and $GD+GI$ over best performing related work, respectively (Wilcoxon signed rank test with threshold p value of 0.05).

Evaluation: Dice's similarity coefficient (DSC) is used to evaluate the segmentation performance of each method. The performance reported is obtained on N_{ts} number of test images for the structures of each dataset as stated earlier.

V. Results and Discussion

Figures (2a-to-2c),(2d-to-2e) and 2f present the quantitative results of the experiments on the cardiac, prostate and pancreas datasets, respectively. The mean DSC and standard deviation values over all the runs are reported on the top of each boxplot in these figures. We observe that the proposed method of augmentation (i.e.,GD+GI) outperforms the other data augmentation and semi-supervised learning methods considered here. For qualitative inspection, we present some examples of visual results in Fig. 4. In the rest of this section, we present further analysis of the experimental results.

As expected, the lowest performance was observed when no data augmentation is used. Employing affine augmentation alone provided a substantial boost in performance. Adding random elastic deformations or random intensity fluctuations on top of affine augmentation yielded further improvements.

Using the proposed learned deformation fields(GD) for augmentation yielded higher performance compared to random elastic deformations(RD). This results suggest that the proposed approach to learn a deformation field generator, by optimizing the segmentation accuracy along with a regularization term that leverages unlabeled examples, provided augmented examples more useful for obtaining high segmentation performance than random deformations. Some samples of the generated deformed images are illustrated in the Fig. 5. Surprisingly, we observed that the generated images did not always have realistic anatomical shapes. This is contrary to the popular belief, but generating realistic images may not be necessary nor optimal to obtain the best segmentation network.

Similar to the deformation case, the proposed additive intensity mask(GI) based augmentations performed better than random intensity fluctuations(RI). Here as well, the result suggest benefits of optimizing the intensity transformation generator using the proposed approach. Fig. 5 illustrates that

the images generated from the learned intensity transformation generator are not necessarily realistic.

Since both the generators G_V and G_I are modeled to encapsulate different characteristics of the entire population, using both the augmentations is expected to produce higher performance gains over using only one of the augmentation separately. In our experiments, we indeed observed a substantial improvement in DSC when both are used together.

To our surprise, we observed that the Mixup augmentation yielded substantial performance gain over the affine transformations and random elastic deformations. This improvement was despite the augmented images being unrealistic. We attribute the performance improvement to the fact that Mixup creates soft probability maps for augmented images, which has been hypothesized to assist the optimization by providing additional information for training samples [36]. Applying Mixup over the augmented data obtained from the trained generators G_V and G_I yielded further marginal improvements, which suggests both approaches have complementary benefits.

With self-training, we observed an improvement in DSC over the affine augmentations as illustrated in Fig. 2. This has been well-documented in SSL literature [22], [23] that retraining the neural network with the estimated predictions of the unlabeled data can assist in improving the segmentation performance [24] in limited annotation setting. Although this yields some improvement over the affine augmentations, it did not outperform the proposed augmentations (GD+GI) except for the structure central gland of the prostate dataset.

The semi-supervised adversarial training [27], [28] provided marginal performance gains over the baseline with affine augmentation (Results of [28] that uses image-level adversarial training are not reported as the GAN training did not converge to reasonable performance for the case of one labeled volume). This observation is not surprising as it has also been shown for other tasks in [37], SSL training may yield minimal performance gains when affine augmentations are included in the training.

In the Appendix, we provide additional analysis plots such as: (a) the performance improvement seen per test subject averaged over all the runs for each dataset in Fig. 9. We observed that for the majority of test subjects, the proposed method performs better or equal to random augmentations. (b) A sample of generated deformation and additive intensity fields obtained from the trained generators on the cardiac dataset are provided in Fig. 10 and Fig. 11, respectively.

Despite getting performance improvements using the proposed method for the three datasets evaluated in this work, it is important to note that it is a computationally expensive method to deploy on any new dataset. This is because one would need to search for the optimal hyper-parameters (λ_{adv}, λ_{LD}) for the loss terms.

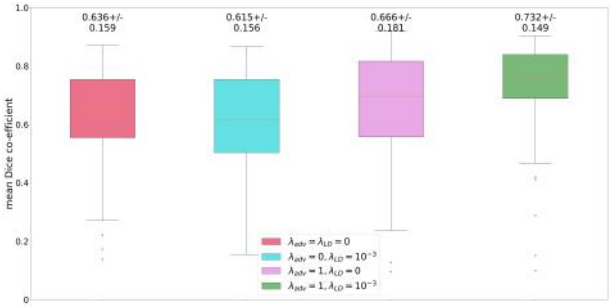
Also, we want to state that we do not think our model and experiments presented can provide any insight or address the intensity differences problem arising from scanner differences and nor does this study focuses on addressing this. We are modeling augmentations to maximize performance given a limited labeled and unlabeled set and evaluate the model with images from one type of scanner. Hence, we do not expect this setup to address the scanner differences problems.

Ablation studies:

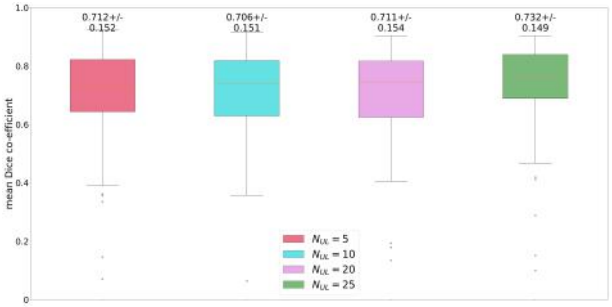
A. Effects of adversarial (λ_{adv}) and large deviation (λ_{LD}) loss terms of regularization loss on segmentation performance: In this experiment, we analyze the effect of each term of the regularization loss on the performance by varying the values of λ_{adv} and λ_{LD} in the training of the generators G_V and G_I. Fig. 3a presents the quantitative results of the analysis on the cardiac dataset for GD+GI augmentations. We observed the least performance gain over baseline when we disabled the whole regularization with $\lambda_{adv} = \lambda_{LD} = 0$, the setup similar to the work in [20]. This setup yielded performance similar to the case when both random deformations and intensity fluctuations were leveraged for augmentation. The performance gain we observe when we enable only the adversarial loss in the regularization, i.e. ($\lambda_{adv} = 1, \lambda_{LD} = 0$), can be attributed to enforcing the model to match the distribution of generated images to that of unlabeled images. This matching propels the generator to synthesize examples showing the diverse set of shape and intensity variations present in the unlabeled data. We observed a deterioration in performance when only large-deviation loss is enabled on deformation and intensity fields ($\lambda_{adv} = 0, \lambda_{LD} = 10^{-3}$). This setup encourages the generators to explicitly produce larger deformation and intensity fields without any control from the adversarial term. Transformations output from these generators yields very unrealistic samples, in the most extreme case moving all the foreground pixels out of the frame. Such synthetic examples, naturally, are not useful for the segmentation task.

We observed the highest performance boost when both terms were enabled($\lambda_{adv} = 1, \lambda_{LD} = 10^{-3}$). Effectively, for the proposed method to yield stated gains, one needs to use all three losses simultaneously, where they compete against each other. Also, this indicates the value of the combination of the terms and their complementary behavior. The large-deviation loss, when used in addition to the adversarial loss, prevents the network from simply replicating the training data with generating relatively smaller transformations. Instead, it compels the generator to produce larger fields as long as they satisfy discriminator's objective. Effectively, producing augmented image-label pairs that are very different from the labeled image-label pairs. Adversarial loss on the other hand, contains the effects of the large-deviation loss by not allowing models to generate extreme transformations.

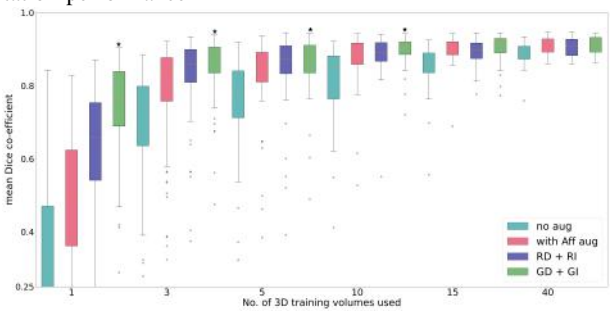
B. Independent optimization of the generator and the segmentation networks: One of the biggest claims of the proposed method is the benefit of using the segmentation cost function for learning the generators. This leads to a joint optimization of the segmentation network's parameters along with the generator's. Here, we examine the impact on segmentation performance when the generators are optimized without the segmentation loss, only using the regularization term with adversarial and large-deviation terms. Results shown in Table I show that when the segmentation loss is included in the learning of the generators, i.e., joint optimization leads to much higher DSC values than independent optimization. Thus, empirically proving our point that task-driven optimization is



(a) Effect of adversarial (λ_{adv}) and large deviation (λ_{LD}) loss terms of the regularization loss on the segmentation performance of the proposed method. The DSC with random augmentations (RD+RI) is 0.627 from Fig. 1 for reference to compare with the combinations of loss terms evaluated here.



(b) Effect of number of 3D unlabeled images (N_{UL}) on the segmentation performance



(c) Effect of number of 3D labeled images (N_L) on the segmentation performance

Fig. 3: Results of segmentation performance on cardiac dataset quantifying: (a) effects of adversarial (λ_{adv}) and large deviation (λ_{LD}) loss terms of the regularization loss for $N_L = 1$ (see V-A), (b) effect of varying the number of 3D unlabeled images N_{UL} for $N_L = 1$ (see V-C) (mean DSC and standard deviation values reported on top of each boxplot) and (c) effect of varying the number of 3D labeled volumes ($N_L = 1, 3, 5, 10, 15, 40$) in the training (see V-D) and $\star$ here indicates statistically significant improvements between proposed method (GD+GI) against best of RD+RI and Aff augmentations using wilcoxon's paired t-test with threshold value of 0.05.

better than the traditional independent optimization, which was the approach taken in earlier works [16], [17] to generate the augmented data independent of the down-stream task.

C. Varying number of unlabeled images: We investigate how varying the number of unlabeled 3D volumes for the training of the proposed method can influence the segmentation performance. This experiment is illustrated in Fig. 3b for the cardiac dataset. For extremely low values of unlabeled volumes of 1 and 3, we observe a drop in performance. While for a higher value of 50 volumes, we observed negligible improvements compared to using 25 volumes. We observe that the improvements in segmentation accuracy do not change significantly for other values of unlabeled volumes from 5 to 25. This indicates that we need some amount of unlabeled volumes to obtain stated performance gains, and beyond a certain number of unlabeled volumes we do not get extra benefits.

D. Varying number of labeled images: Here, we investigated how the performance gap between affine, random, and proposed augmentations varies as we increase the number of training volumes involved in the training. We observe that the performance gap between the augmentation approaches reduces as we increase the number of labeled training volumes as shown in Fig. 3c. This is expected since as the labeled examples increase, the network sees larger number cases and gains robustness to variations present in these images. For the case of 40 3D training volumes, we see that the performance of affine augmentations is almost similar to the proposed augmentations. We observe that the statistically significant improvements exist until 10 labeled examples. Also, we see that with the proposed model, segmentation accuracy using 10 labeled examples is similar to using 40 examples using random augmentations.

E. Different set of train, validation, test, and unlabeled 3D volumes: Lastly, we show that the results hold for any randomly chosen dataset split, and does not overfit to a specific set of validation images as illustrated in Table II.

F. No validation images: Additionally, we also present results for the case when no validation images were used in the training in Fig. 8 in the appendix. Here, the chosen model parameters are obtained after training the generators for predefined number of iterations. Surprisingly, we observe that the performance obtained without validation images does not vary much w.r.t the performance obtained using validation images.

VI. Conclusion

In the clinical setting, deployment of successful deep learning algorithms for medical image analysis is limited due to the difficulty of assembling large-scale annotated datasets. In this work, we proposed a semi-supervised task-driven data augmentation method to tackle the issue of obtaining robust

methods of optimization	$N_L = 1$			$N_L = 3$		
	RV	Myo	LV	RV	Myo	LV
independent	0.527 (0.266)	0.553 (0.218)	0.719 (0.23)	0.793 (0.187)	0.797 (0.097)	0.909 (0.09)
joint	0.651 (0.23)	0.710 (0.157)	0.834 (0.171)	0.832 (0.148)	0.823 (0.076)	0.922 (0.072)

TABLE I: Effect on the segmentation performance when the generator is optimized jointly with the segmentation networks' loss against the independent optimization of generator without segmentation loss. Mean Dice score with standard deviations in brackets is presented for cardiac dataset (see V-B).

Methods	RV	Myo	LV
RD+RI	0.506 (0.213)	0.647 (0.159)	0.819 (0.149)
GD+GI	0.683 (0.212)	0.693 (0.139)	0.842 (0.136)

TABLE II: Mean Dice scores with standard deviations for the cardiac dataset for a different set of train, validation, test, and unlabeled volumes for $N_L = 1$ (see V-E).

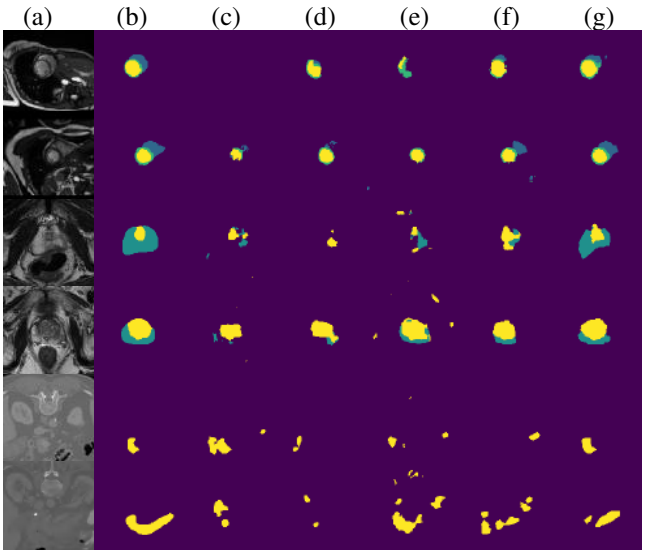


Fig. 4: Qualitative comparison of the proposed method with other approaches is illustrated for two images each from cardiac, prostate, and pancreas datasets in the order of top to bottom: (a) input image, (b) ground truth, (c) Aff, (d) RD+RI, (e) Adv_tr [27], (f) Mixup [18], (g) GD+GI (Ours).

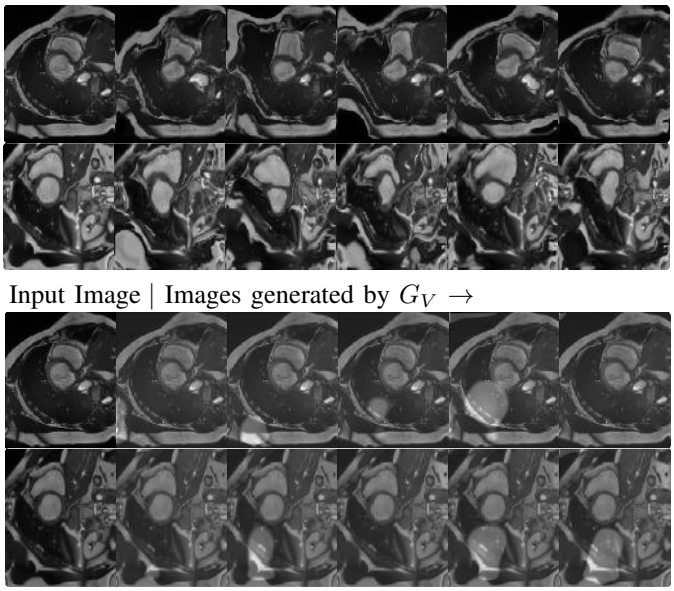


Fig. 5: Generated augmentation images from the deformation field generator G_V (top) and the intensity field generator G_I (bottom) for the cardiac dataset.

segmentation in limited data setting for training. To achieve this we proposed two novel contributions: (i) task-driven based optimization where the generation of the augmentation data is optimal for the segmentation performance, and (ii) semi-supervised nature is induced by using the unlabeled data in the generative modeling setup, where we design two conditional generative models to output transformations that capture two factors of variations: shape and intensity characteristics present in the population. Using three publicly available datasets, we demonstrated the proposed method for segmenting the cardiac, prostate and pancreas using limited annotated examples, reporting substantial performance gains over existing methods. Surprisingly, the augmented images generated via the proposed task-driven approach were not necessarily realistic yet yielded improved segmentation performance, questioning the validity of the assumption that generating realistic examples is the optimal way.

VII. Acknowledgements

The presented work is partially funded by: 1. Swiss Data Science Center (DeepMicroIA), 2. Clinical Research Priority Program Grant (CRPP) on Artificial Intelligence in Oncological Imaging Network, University of Zurich, 3. Platform for Advanced Scientific Computing (PASC) - project HCP-Predict, 4. Personalized Health Related Technologies - project number 222, ETH. We thank Nvidia for their GPU donation.

References

[1] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *null*. IEEE, 2003, p. 958.

[2] O. Ronneberger, P. Fischer, and 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.

[3] K. Chaitanya, N. Karani, C. F. Baumgartner, A. Becker, O. Donati, and E. Konukoglu, "Semi-supervised and task-driven data augmentation," in *Information Processing in Medical Imaging*, A. C. S. Chung, J. C. Gee, P. A. Yushkevich, and S. Bao, Eds. Cham: Springer International Publishing, 2019, pp. 29–41.

[4] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "High-performance neural networks for visual object classification," *arXiv preprint arXiv:1102.0183*, 2011.

[5] J. Hong, B.-y. Park, and H. Park, "Convolutional neural network classifier for distinguishing barrett's esophagus and neoplasia endomicroscopy images," in *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*. IEEE, 2017, pp. 2892–2895.

[6] F. Perez, C. Vasconcelos, S. Avila, and E. Valle, "Data augmentation for skin lesion analysis," in *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*. Springer, 2018, pp. 303–311.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] H. Salehinejad, S. Valaee, T. Dowdell, E. Colak, and J. Barfett, "Generalization of deep neural networks for chest pathology classification in x-rays using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 990–994.

[9] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Synthetic data augmentation using gan for improved liver lesion classification," in *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*. IEEE, 2018, pp. 289–293.

[10] J. T. Guibas, T. S. Virdi, and P. S. Li, "Synthetic medical images from dual generative adversarial networks," *arXiv preprint arXiv:1709.01872*, 2017.

[11] L. Hou, A. Agarwal, D. Samaras, T. M. Kurc, R. R. Gupta, and J. H. Saltz, "Unsupervised histopathology image synthesis," *arXiv preprint arXiv:1712.05021*, 2017.

[12] E. Wu, K. Wu, D. Cox, and W. Lotter, "Conditional infilling gans for data augmentation in mammogram classification," in *Image Analysis for Moving Organ, Breast, and Thoracic Images*. Springer, 2018, pp. 98–106.

[13] S. Liu, J. Zhang, Y. Chen, Y. Liu, Z. Qin, and T. Wan, "Pixel level data augmentation for semantic image segmentation using generative adversarial networks," *arXiv preprint arXiv:1811.00174*, 2018.

[14] L. Sixt, B. Wild, and T. Landgraf, "Rendergan: Generating realistic labeled data," *Frontiers in Robotics and AI*, vol. 5, p. 66, 2018.

[15] P. Costa, A. Galdran, M. I. Meyer, M. Niemeijer, M. Abràmoff, A. M. Mendonça, and A. Campilho, "End-to-end adversarial retinal image synthesis," *IEEE transactions on medical imaging*, vol. 37, no. 3, pp. 781–791, 2018.

[16] H.-C. Shin, N. A. Tenenholtz, J. K. Rogers, C. G. Schwarz, M. L. Senjem, J. L. Gunter, K. P. Andriole, and M. Michalski, "Medical image synthesis for data augmentation and anonymization using generative adversarial networks," in *International Workshop on Simulation and Synthesis in Medical Imaging*. Springer, 2018, pp. 1–11.

[17] C. Bowles, L. Chen, R. Guerrero, P. Bentley, R. Gunn, A. Hammers, D. A. Dickie, M. V. Hernández, J. Wardlaw, and D. Rueckert, "Gan augmentation: Augmenting training data using generative adversarial networks," *arXiv preprint arXiv:1810.10863*, 2018.

[18] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[19] Z. Eaton-Rosen, F. Bragman, S. Ourselin, and M. J. Cardoso, "Improving data augmentation for medical image segmentation," in *International Conference on Medical Imaging with Deep Learning*, 2018.

[20] Y.-X. Wang, R. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7278–7286.

[21] M. Paschali, W. Simson, A. G. Roy, R. Göbl, C. Wachinger, and N. Navab, "Manifold exploring data augmentation with geometric transformations for increased performance and robustness," in *International Conference on Information Processing in Medical Imaging*. Springer, 2019, pp. 517–529.

[22] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd annual meeting of the association for computational linguistics*, 1995.

[23] M. Li and Z.-H. Zhou, "Setred: Self-training with editing," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2005, pp. 611–621.

[24] W. Bai, O. Oktay, M. Sinclair, H. Suzuki, M. Rajchl, G. Tarroni, B. Glocker, A. King, P. M. Matthews, and D. Rueckert, "Semi-supervised learning for network-based cardiac mr image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 253–260.

[25] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[26] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synthesis lectures on artificial intelligence and machine learning*, vol. 3, no. 1, pp. 1–130, 2009.

[27] Y. Zhang, L. Yang, J. Chen, M. Fredericksen, D. P. Hughes, and D. Z. Chen, "Deep adversarial networks for biomedical image segmentation utilizing unannotated images," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2017, pp. 408–416.

[28] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5689–5697.

[29] A. Zhao, G. Balakrishnan, F. Durand, J. V. Guttag, and A. V. Dalca, "Data augmentation using learned transforms for one-shot medical image segmentation," *arXiv preprint arXiv:1902.09383*, 2019.

[30] Y. Can, K. Chaitanya, B. Mustafa, L. Koch, E. Konukoglu, and C. Baumgartner, "Learning to segment medical images with scribble-supervision alone," *arXiv preprint arXiv:1807.04668v1*, 2018.

[31] S. Andermatt, A. Horváth, S. Pezold, and P. Cattin, "Pathology segmentation using distributional differences to images of healthy origin," *arXiv preprint arXiv:1805.10344*, 2018.

[32] O. Bernard, A. Lalande, C. Zotti, F. Cervenansky, X. Yang, P.-A. Heng, I. Cetin, K. Lekadir, O. Camara, M. A. G. Ballester *et al.*, "Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved?" *IEEE Transactions on Medical Imaging*, 2018.

[33] N. J. Tustison, B. B. Avants, P. A. Cook, Y. Zheng, A. Egan, P. A. Yushkevich, and J. C. Gee, "N4itk: improved n3 bias correction," *IEEE transactions on medical imaging*, vol. 29, no. 6, pp. 1310–1320, 2010.

[34] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

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

[36] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[37] A. Oliver, A. Odena, C. Raffel, E. D. Cubuk, and I. J. Goodfellow, "Realistic evaluation of deep semi-supervised learning algorithms," *arXiv preprint arXiv:1804.09170*, 2018.

VIII. Appendix

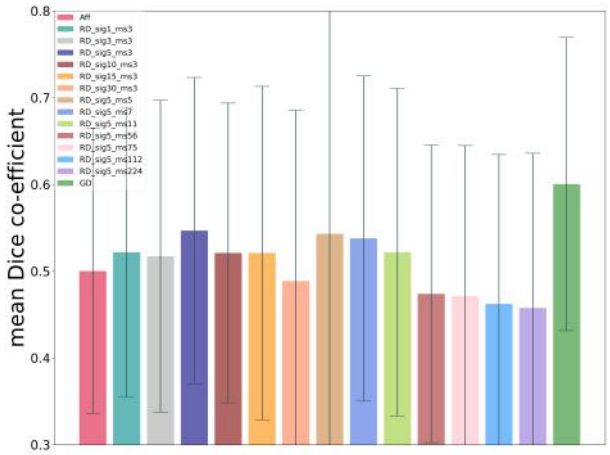


Fig. 6: The segmentation performance was obtained using random elastic augmented images (RD). These RD images are generated using elastic deformation fields as defined in [2] applied to the input image. We evaluate different initial matrix sizes (ms x ms x 2) and sigma values that are later interpolated to image dimensions to obtain the deformation fields. These matrix values are randomly sampled from a Gaussian distribution of zero mean and standard deviation of sigma. Here, GD denotes that the augmented images are generated using the learned deformation field generator G_V. (mean Dice scores are reported for $N_L = 1$ for the cardiac dataset).

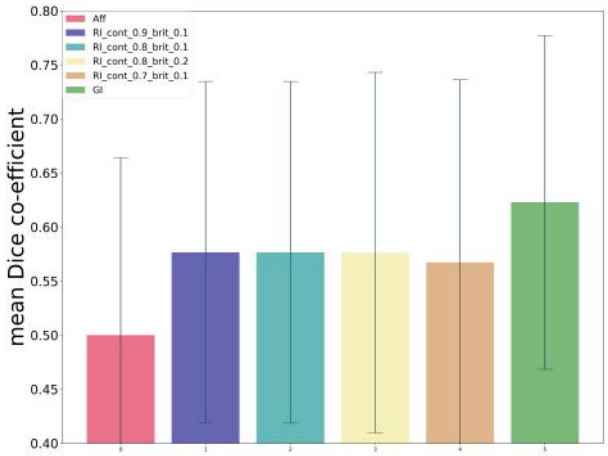


Fig. 7: The segmentation performance obtained using different combinations of contrast (cont) and brightness (brit) values used to generate intensity transformation fields as defined in [5], [6]. Here, GI denotes that the augmented images are generated using the learned intensity field generator G_I. (mean Dice scores are reported for $N_L = 1$ for the cardiac dataset).

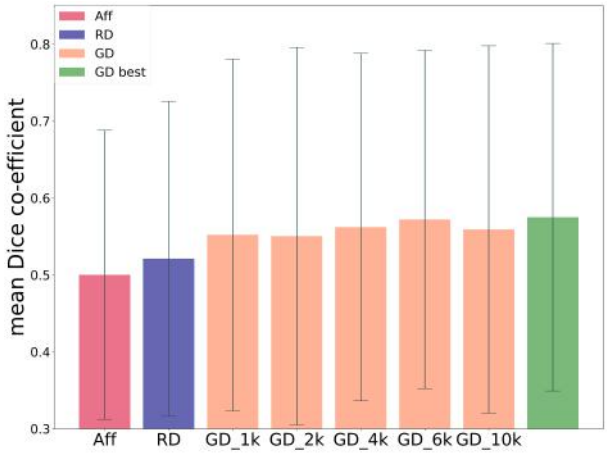
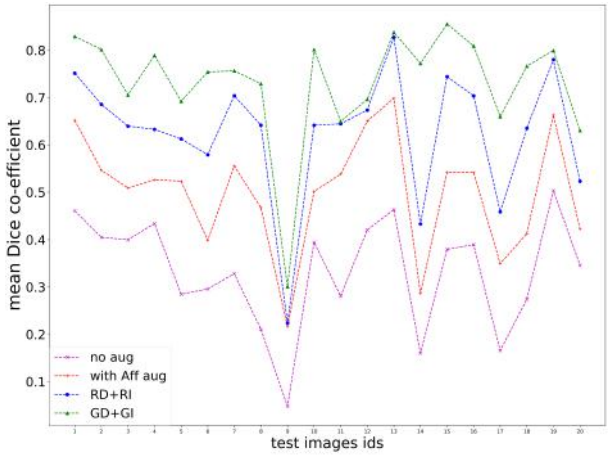


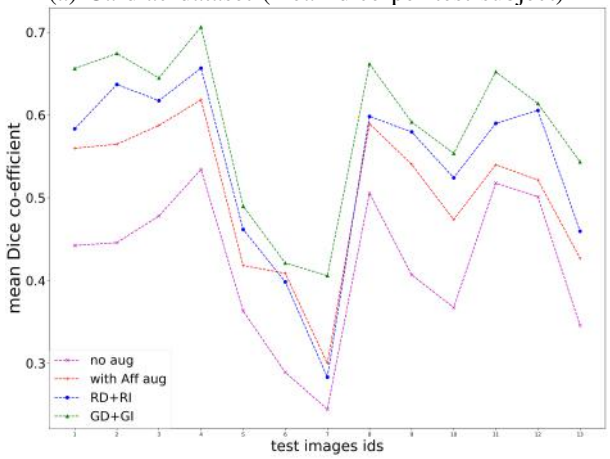
Fig. 8: The segmentation performance obtained for the case of using deformation field generator (GD) when no validation images are used during the training of the generator, and the training is stopped after a pre-defined number of training iterations. Here, $\_xk$ denotes the pre-defined number of training iterations, where $\_1k, \_2k, \_4k, \_6k, \_10k$ denotes that the model is trained for 1000, 2000, 4000, 6000, 10000 iterations, respectively. $\_best$ denotes the case where two 3D validations volumes are used to determine the model parameters based on best validation Dice score observed through all the 10000 training iterations. We additionally compare it against random deformations (RD) (mean Dice scores is presented for $N_L = 1$ for the cardiac dataset).

method	RV	Myo	LV
random deformations + intensities (RD + RI)	0.241 (0.224)	0.368 (0.246)	0.44 (0.296)
learned deformations + intensities (GD + GI)	0.411 (0.28) $\star$	0.587 (0.248) $\star$	0.644 (0.249) $\star$

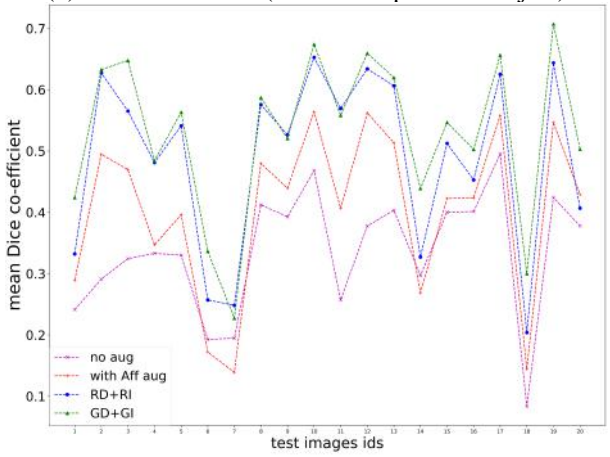
TABLE III: ACDC dataset - mean DSC (std dev) over 15 runs is reported for 20 test **ED** images when segmentation network was trained with 1 labeled **ED** volume with the proposed learned augmented images (GD+GI) which is compared to random augmentations (RD+RI) ($\star$ denotes the statistical significance of the proposed method (GD + GI) over random augmentations (RD + RI) illustrated using the Wilcoxon signed-rank test with a threshold p-value of 0.05).



(a) Cardiac dataset (mean dice per test subject)



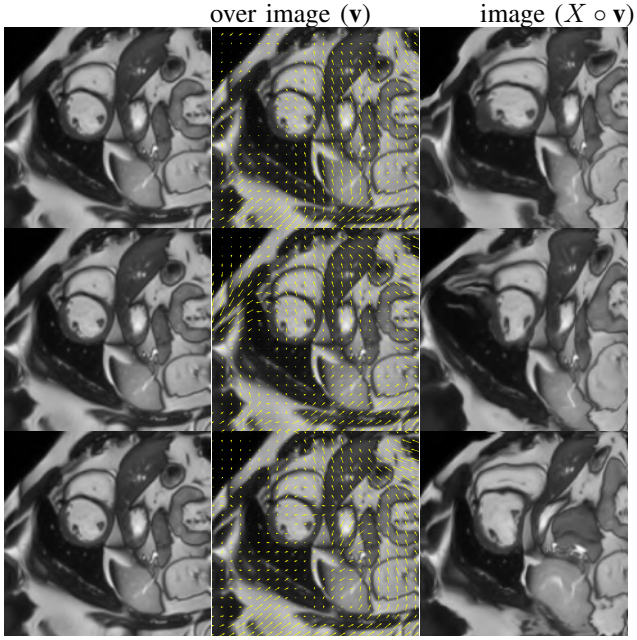
(b) Prostate dataset (mean dice per test subject)



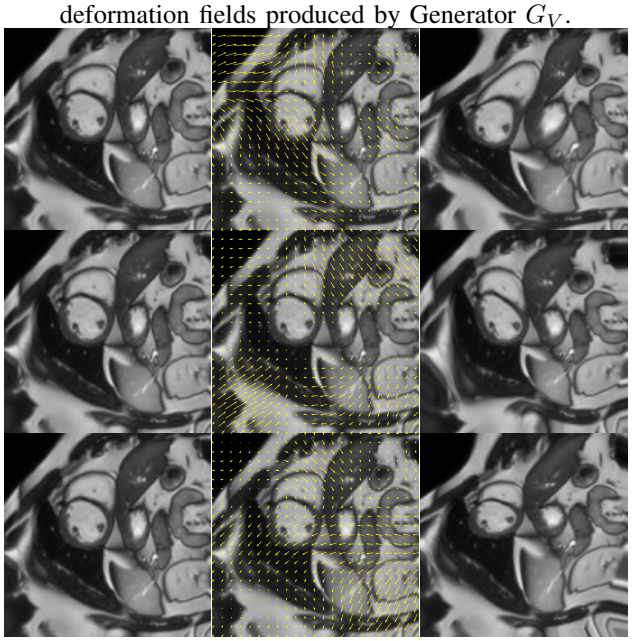
(c) Pancreas dataset (mean dice per test subject)

Fig. 9: Comparison of affine, random, and proposed augmentations' performance for each test subject. Mean Dice scores over 15 runs for each test subject is presented for $N_L = 1$ for the cardiac,prostate dataset and $N_L = 3$ for the pancreas dataset.

(a) image (X) (b) deformation field over image ($\mathbf{v}$) (c) transformed image ($X \circ \mathbf{v}$)



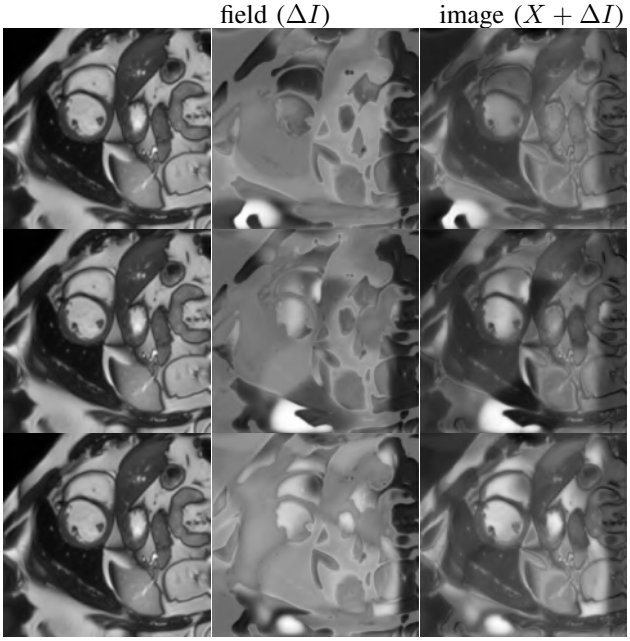
(i) Transformed images are obtained by applying deformation fields produced by Generator G_V.



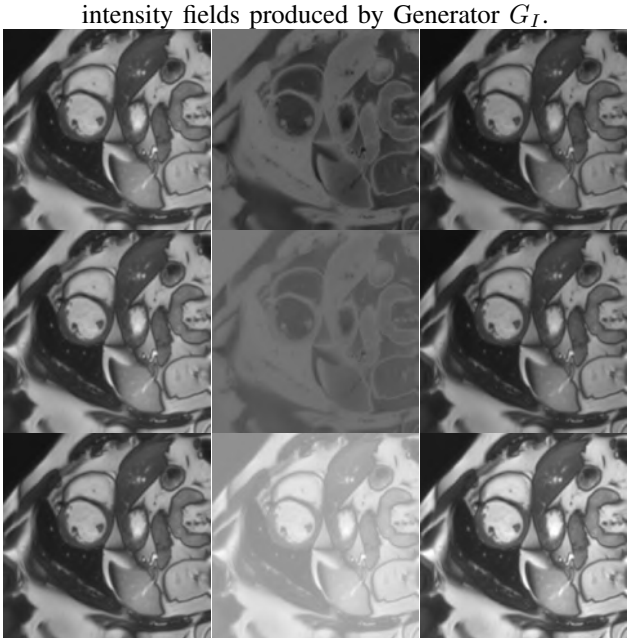
(ii) Transformed images are obtained by applying random elastic deformations.

Fig. 10: Generated deformation fields ($\mathbf{v}$) and corresponding shape transformed augmentation images ($X \circ \mathbf{v}$) obtained from the deformation field generator G_V or random elastic deformations for an input image (X) from the cardiac dataset.

(a) image (X) (b) additive intensity field (ΔI) (c) transformed image ($X + \Delta I$)



(i) Transformed images are obtained by applying additive intensity fields produced by Generator G_I.



(ii) Transformed images are obtained by applying random contrast and brightness values.

Fig. 11: Generated additive intensity fields (ΔI) and corresponding intensity transformed augmentation images ($X+\Delta I$) obtained from the intensity field generator G_I or random contrast and brightness for an input image (X) from the cardiac dataset.