GAN-based Synthetic Medical Image Augmentation for increased CNN Performance in Liver Lesion Classification

Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan, *Member, IEEE*

***Abstract*—Deep learning methods, and in particular convolutional neural networks (CNNs), have led to an enormous breakthrough in a wide range of computer vision tasks, primarily by using large-scale annotated datasets. However, obtaining such datasets in the medical domain remains a challenge. In this paper, we present methods for generating synthetic medical images using recently presented deep learning Generative Adversarial Networks (GANs). Furthermore, we show that generated medical images can be used for synthetic data augmentation, and improve the performance of CNN for medical image classification. Our novel method is demonstrated on a limited dataset of computed tomography (CT) images of 182 liver lesions (53 cysts, 64 metastases and 65 hemangiomas). We first exploit GAN architectures for synthesizing high quality liver lesion ROIs. Then we present a novel scheme for liver lesion classification using CNN. Finally, we train the CNN using classic data augmentation and our synthetic data augmentation and compare performance. In addition, we explore the quality of our synthesized examples using visualization and expert assessment. The classification performance using only classic data augmentation yielded 78.6% sensitivity and 88.4% specificity. By adding the synthetic data augmentation the results increased to 85.7% sensitivity and 92.4% specificity. We believe that this approach to synthetic data augmentation can generalize to other medical classification applications and thus support radiologists' efforts to improve diagnosis.**

***Index Terms*—Image synthesis, data augmentation, convolutional neural networks, generative adversarial network, deep learning, liver lesions, lesion classification.**

I. Introduction

THE greatest challenge in the medical imaging domain is how to cope with the small datasets and limited amount of annotated samples [1]–[5], especially when employing supervised machine learning algorithms that require labeled data and larger training examples. In medical imaging tasks, annotations are made by radiologists with expert knowledge on the data and task. Most annotations of medical images are time consuming. This is especially true for precise annotations, such as the segmentations of organs or lesions into multiple 2-D slices and 3-D volumes. Although public medical datasets are available online, and grand challenges have been publicized, most datasets are still limited in size and only applicable to specific medical problems. Collecting medical data is a complex and expensive procedure that requires the collaboration of researchers and radiologists [3].

Researchers attempt to overcome this challenge by using data augmentation. The most common data augmentation methods include simple modifications of dataset images such as translation, rotation, flip and scale. Using classic data augmentation to improve the training process of networks is a standard procedure in computer vision tasks [6]. However, little additional information can be gained from small modifications to the images (e.g. the translation of the image a few pixels to the right). Synthetic data augmentation of high quality examples is new, sophisticated type of data augmentation. Synthetic data examples learned using a generative model enable more variability and enrich the dataset to further improve the system training process.

One such promising approach inspired by game theory for training a model that syntheses images is known as Generative Adversarial Networks (GANs) [7]. The model consists of two networks that are trained in an adversarial process where one network generates fake images and the other network discriminates between real and fake images repeatedly. GANs have gained great popularity in the computer vision community and different variations of GANs were recently proposed for generating high quality realistic natural images [8]–[11]. Interesting applications of GAN include generating images of one style from another style (image-to-image translation) [12] and image inpainting using GAN [13].

Recently, several medical imaging applications have applied the GAN framework [14]–[20]. Most studies have employed the image-to-image GAN technique to create label-to-segmentation translation, segmentation-to-image translation or medical cross modality translations. Costa et al. [14] trained a fully-convolutional network to learn retinal vessel segmentation images. Then they learned the translation from the binary vessel tree to a new retinal image. Dai et al. [15] trained GAN to create segmentation images of the lung fields and the heart from chest X-ray images. Xue et al. [16] referred to the two GAN networks as a Segmentor and Critic, and learned the translation between brain MRI images and a brain tumor binary segmentation map. In Nie et al. [17], A patch-based GAN was trained for translation between brain CT images and the corresponding MRI images. They further suggested an auto-context model for image refinement. Ben-Cohen et

M. Frid-Adar, I. Diamant and H. Greenspan are with the Department of Biomedical Engineering, Tel Aviv University, Tel Aviv, Israel (e-mail: maayan.frid@gmail.com; iditdiamant@gmail.com; hayit@eng.tau.ac.il).

E. Klang and M. Amitai are with the Department of Diagnostic Imaging, The Chaim Sheba Medical Center, Tel-Hashomer, Israel (e-mail: eyalkla@hotmail.com; michal.amitai@sheba.health.gov.il).

J. Goldberger is with the Faculty of Engineering, Bar-Ilan University, Ramat-Gan, Israel (e-mail: jacob.goldberger@biu.ac.il).

al. [20] also introduced a cross modality image generation using GAN, from abdominal CT image to a PET scan image that highlights liver lesions. Some studies have been inspired by the GAN method for image inpainting. Schlegl et al. [18] trained GAN with healthy patches of the retinal area to learn the data distribution of healthy tissue. Then they tested the GAN on patches of both unseen healthy and anomalous data for anomaly detection in retinal images.

The problem of limited data in the medical imaging field prompted us to explore methods for synthetic data augmentation to enlarge medical datasets. In the current study, we focus on improving results in the specific task of liver lesion classification. We applied the GAN framework to synthesize high quality liver lesion images (hereon we use interchangeably the terms lesion images and lesion ROIs).

The liver is one of three most common sites for metastatic cancer along with the bone and lungs [21]. According to the World Health Organization, in 2012 alone, cancer accounted for 8.2 million deaths worldwide of which 745,000 were caused by liver cancer [22]. Focal liver lesions can be malignant and manifest metastases, or be benign (e.g. hemangioma or hepatic cysts). Computed tomography (CT) is one of the most common and robust imaging techniques for the detection, diagnosis and follow up of liver lesions [23]. Thus, there is a great need and interest in developing automated diagnostic tools based on CT images to assists radiologists in the diagnosis of liver lesions.

Previous studies have presented methods for automatic classification of focal liver lesions in CT images [24]–[30]. Gletsos et al. [24] used texture features for liver lesion classification into four categories including the normal liver parenchyma class. They applied a hierarchical classifier of neural networks at each level. Chang et al. [26] obtained three kind of features for each tumor, including texture, shape, and kinetic curve on segmented tumors. Backward elimination was used to select the best combination of features through binary logistic regression analysis to classify the tumors. Diamant et al. [29] applied the bag-of-visual-words (BoVW) method learned from image patches. They used two dictionaries for lesion interior and boundary regions. Based on the two dictionaries they generated histograms for each lesion ROI. The final classification was made using SVM.

In the current work we used deep learning methodology for the task of liver lesion classification. Deep learning convolutional neural networks (CNNs) has emerged as a powerful tool in computer vision. In recent years many medical imaging studies have applied CNNs and reported improved performance for a broad range of medical tasks [3]. We combine synthetic liver lesion generation using GAN with our proposed CNN for liver lesion classification.

The contributions of this work are the following:

1) Synthesis of high quality focal liver lesions from CT images using generative adversarial networks (GANs).
2) Design of a CNN-based solution for the liver lesion classification task, with comparable results to state-of-the-art methods.
3) Augmentation of the CNN training set, using the generated synthetic data - for improved classification results.

II. Liver Lesion Classification

In this section we first describe the data and their characteristics. Then we elaborate on the CNN architecture for the liver lesion classification task. The main challenge is the small amount of data available for training the CNN. In the next section we describe methods to artificially enlarge the data.

A. Data

The dataset used in this work contains cases of liver lesions collected from Sheba Medical Center by searching medical records for cases of cysts, metastases and hemangiomas. Cases were acquired from 2009 to 2014 using two CT scanners: a General Electric (GE) Healthcare scanner and a Siemens Medical System scanner, with the following parameters: 120kVp, 140-400mAs and 1.25-5.0 mm slice thickness. Cases were collected with the approval of the institution's Institutional Review Board.

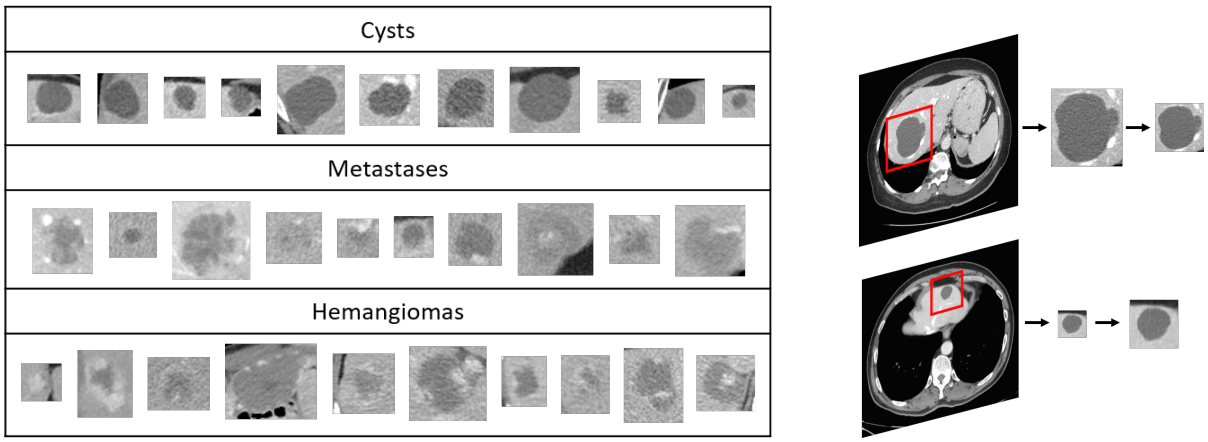
Figure 1 shows examples of the input data and the ROI extraction process. The dataset was made up of 182 portal-phase 2-D CT scans (Figure 1a): 53 cysts, 64 metastases, 65 hemangiomas. An expert radiologist marked the margin of each lesion and determined its corresponding diagnosis which was established by biopsy or a clinical follow-up. This serves as our ground truth.

Liver lesions vary considerably in shape, contrast and size (10 - 102mm). They also vary within categories. In addition, they are located in interior sections of the liver or near its boundary where the surrounding parenchyma tissue of the lesions changes. Each type of lesion has its own characteristics: Cysts are non-enhancing water-attenuation circumscribed lesions. Metastases are hypoattenuating, have soft-tissue attenuation and less well-defined margins than cysts, and hemangiomas show typical features of discontinuous nodular peripheral enhancement, with fill-in on delayed images [31]. Despite this detailed description, some characteristics may be confusing, in particular for metastasis and hemangioma lesions (see Figure 1a). Metastases can contain areas of higher density, probably prominent blood vessels or calcifications that can be mistaken for hemangiomas attributes. Hemangiomas are benign tumors and metastases are malignant lesions derived from different primary cancers. Thus, the correct identification of a lesion as metastasis or hemangioma is especially important.

The input to our classification system are ROIs of lesions cropped from CT scans using the radiologist's annotations. The ROIs are extracted to capture the lesion and its surrounding tissue relative to its size. Due to the large variability in lesion sizes, this results in varying size ROIs (Figure 1b).

B. CNN Architecture

The architecture of the liver lesion classification system we propose is shown in Figure 2. CNNs are widely used for solving image classification tasks in computer vision [6]. CNN architectures for medical imaging have also been introduced [1], [32], [33], usually containing fewer convolutional layers because of the small datasets and smaller input size. Our classification CNN gets fixed size input ROIs of 64×64, with



(a) (b)

Fig. 1. (a) Dataset examples of cyst, metastasis and hemangioma liver lesions. (b) ROI extraction process from a 2-D CT slice of the liver. All ROIs are resized to a uniform size.

an intensity range rescaled to $(0, 1)$. The architecture consists of three pairs of convolutional layers where each convolutional layer is followed by a max-pooling layer, and two dense fully-connected layers ending with a soft-max layer to determine the network predictions to classify lesions into three classes. We use ReLU as activation functions. The network had approx. 1.3M parameters. In addition, to further reducing overfitting, we incorporated a dropout layer [34] with a probability of 0.5 during training.

Training Procedure. The mean value of the training images was subtracted from each image fed into the CNN. For training we used a batch size of 64 with a learning rate of 0.001 for 150 epochs. We used stochastic gradient descent optimization with Nesterov momentum updates [35], where instead of evaluating the gradient at the current position we evaluated it at the "look-ahead" position which improves the optimization process.

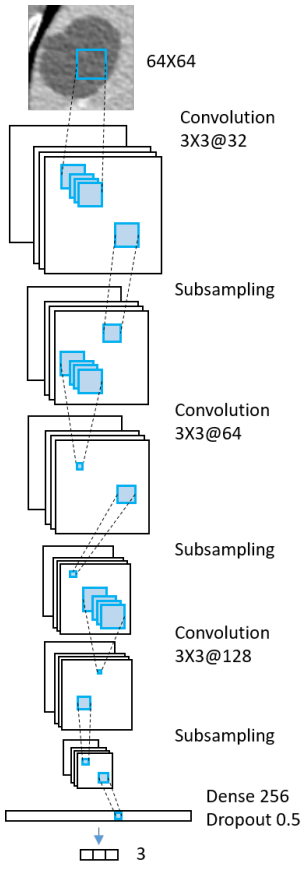


Fig. 2. The architecture of the liver lesion classification CNN.

III. Generating Synthetic Liver Lesions

The main problem in training the network described above is the lack of a large labeled training dataset. To enlarge the training data and improve the classification results in the liver lesion classification task, we augmented the data in two ways: 1) Classic augmentation that includes varieties of known image manipulations on given data examples; 2) Synthesis of new examples which are learned from the data examples using generative models. We start with an overview of standard data augmentation techniques and then describe our new method of generating synthetic liver lesion images using generative adversarial networks (GANs).

A. Classic Data Augmentation

Even a small CNN has thousands of parameters that need to be trained. When using deep networks with multiple layers or dealing with limited numbers of training images, there is a danger of overfitting. The standard solution to reduce overfitting is data augmentation that artificially enlarges the dataset [6]. Classic augmentation techniques on gray-scale images include mostly affine transformations such as translation, rotation, scaling, flipping and shearing [1], [33]. In order to preserve the liver lesion characteristics we avoided transformations that cause shape deformation (like shearing). In addition, we kept the ROI centered around the lesion.

Each lesion ROI was first rotated N_{rot} times at random angles $\theta = [0°, ..., 180°]$. Afterwards, each rotated ROI was flipped N_{flip} times (up-down,left-right), translated N_{trans} times where we sampled random pairs of $[x, y]$ pixel values between $(-p, p)$ related to the lesion diameter (d) by

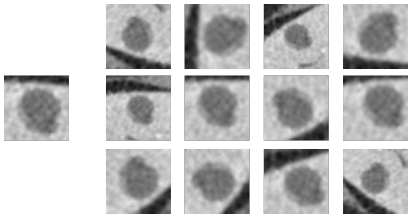


Fig. 3. Lesion ROI and augmentation examples of translation, rotation, flipping and scaling.

$p = min(4, 0.1 \times d)$. Finally the ROI was scaled N_{scale} times from a stochastic range of scales $s = [0.1 \times d, 0.4 \times d]$. The scale was implemented by changing the amount of tissue around the lesion in the ROI. As a result of the augmentation process, the total number of augmentations was $N = N_{rot} \times (1 + N_{flip} + N_{trans} + N_{scale})$. An example lesion and its corresponding augmentations are shown in Figure 3. All the ROIs were resized to fit a uniform size of 64×64 pixels using bicubic interpolation.

B. Generative Adversarial Networks for Lesion Synthesis

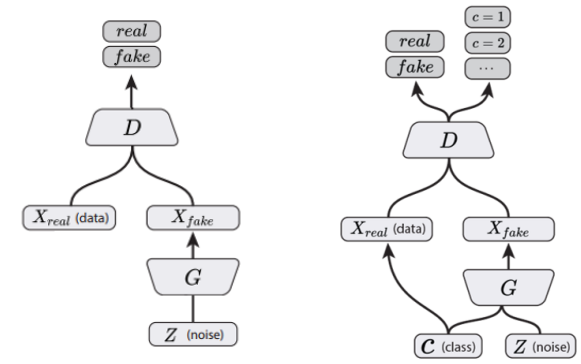
GANs [7] are a specific framework of a generative model. The generative model aims to implicitly learn the data distribution p_{data} from a set of samples $x^{(1)}, ..., x^{(m)}$ (e.g. images) to further generate new samples drawn from the learned distribution. We explored two variants of GANs for synthesizing labeled lesions, as shown in Figure 4: one that generates labeled examples for each lesion class separately and the other that incorporates class conditioning to generate labeled examples all at once.

We started with the first GAN variant, the Deep Convolutional GAN (DCGAN). We followed the architecture proposed by Radford et al. [8], where both the G and D networks are deep CNNs. They suggested architectural guidelines for stable GAN training and modifications of the original GAN proposed by Goodfellow et al. [7], which have become the basis for many recent GAN papers [11], [13], [36]. The model consists of two neural networks that are trained simultaneously (see Figure 4a). The first network is termed the discriminator and is denoted D. The role of the discriminator is to discriminate between the real and fake samples. It is inputted a sample x and outputs $D(x)$, its probability of being a real sample. The second network is termed the generator and is denoted G. The generator synthesizes samples that D will consider to be real samples with high probability. G gets input samples $z^{(1)}, ..., z^{(m)}$ from a known simple distribution p_z, usually a uniform distribution, and maps $G(z)$ to the image space of distribution p_g. The goal of G is to achieve $p_g = p_{data}$.

Adversarial networks are trained by optimizing the following loss function of a two-player minimax game:

$$\min_G \max_D \mathbb{E}_{x \sim p_{data}} \log D(x) + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z)))] \quad (1)$$

The discriminator is trained to maximize $D(x)$ for images with $x \sim p_{data}$ and to minimize $D(x)$ for images with $x \nsim p_{data}$. The generator produces images $G(z)$ to fool D during training such that $D(G(z)) \sim p_{data}$. Therefore, the generator is trained to maximize $D(G(z))$, or equivalently minimize $1 - D(G(z))$. During training the generator improves in its ability to synthesize more realistic images while the discriminator improves in its ability to distinguish the real from the synthesized images. Hence the moniker of adversarial training.



(a) (b)

Fig. 4. (a) DCGAN architecture. (b) ACGAN architecture (Figure is taken from [11]).

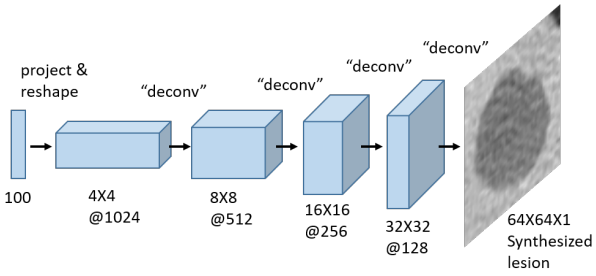


Fig. 5. Generator architecture (of deep convolutional GAN).

Generator Architecture: The generator network takes a vector of 100 random numbers drawn from a uniform distribution as input and outputs a liver lesion image of size $64 \times 64 \times 1$ as shown in Figure 5. The network architecture [8] consists of a fully connected layer reshaped to size $4 \times 4 \times 1024$ and four *fractionally-strided convolutional* layers to up-sample the image with a 5×5 kernel size. A fractionally-strided convolution (known also as 'deconvolution') can be interpreted as expanding the pixels by inserting zeros in between them. Convolution over the expanded image will result in a larger output image. *Batch-normalization* is applied to each layer of the network, except for the output layer. Normalizing responses to have zero mean and unit variance over the entire mini-batch stabilizes the GAN learning process and prevents the generator from collapsing all samples to a single point [37]. ReLU activation functions are applied to all layers except the output layer which uses a tanh activation function.

Discriminator Architecture: The discriminator network has a typical CNN architecture that takes the input image of size $64 \times 64 \times 1$ (lesion ROI), and outputs one decision: is this lesion real or fake? The network consists of four convolution layers with a kernel size of 5×5 and a fully connected layer.

Strided convolutions are applied to each convolution layer to reduce spatial dimensionality instead of using pooling layers. Batch-normalization is applied to each layer of the network, except for the input and output layers. *Leaky ReLU* activation functions $f(x) = \max(x, leak \times x)$ are applied to all layers except the output layer which uses the Sigmoid function for the likelihood probability $(0, 1)$ score of the image.

Training Procedure: We trained the DCGAN to synthesize liver lesion ROIs for each lesion category separately. The training process was done iteratively for the generator and the discriminator. We used mini-batches of m=64 lesion ROI examples $x_l^{(1)}, ..., x_l^{(m)}$ for each lesion type $l \in (Cyst, Metastasis, Hemangioma)$ and m=64 noise samples $z^{(1)}, ..., z^{(m)}$ drawn from uniform distribution between $[-1, 1]$. The only preprocessing steps used involved scaling the training images to the range of the tanh activation function $(-1, 1)$. In the Leaky ReLU, the slope of the leak was set to $leak = 0.2$. Weights were initialized to a zero-centered normal distribution with standard deviation of 0.02. We applied stochastic gradient descent with the Adam optimizer [38], an adaptive moment estimation that incorporates the first and second moments of the gradients, controlled by parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$ respectively. We used a learning rate of 0.0002 for 70 epochs.

C. Conditional Lesion Synthesis

The second GAN variant is the Auxiliary Classifier GAN (ACGAN). Conditional GANs are an extension of the GAN model, that enable the model to be conditioned on external information to improve the quality of the generated samples. GAN architectures that incorporate the class labels to produce labeled samples were introduced by [10], [11], [36]. Odena et al. [11] suggested that instead of feeding the discriminator with side information [10], the discriminator should be tasked with reconstructing side information. This is done by modifying the discriminator to contain an auxiliary decoder network that outputs the class label in addition to the real or fake decision (see Figure 4b). We followed the architecture proposed in [11] with minor modifications for synthesizing the labeled lesions of all three types. ACGANs generator architecture is similar to the DCGANs architecture described in section III-B with class embedding in addition to the input noise samples. The ACGAN discriminator architecture modified the DCGAN to have kernels of size 3×3 with strided convolutions every odd layer and incorporates a dropout of 0.5 in every layer except for the last layer. We use the ACGAN discriminator without these modifications after optimizing for our small dataset. The discriminator auxiliary decoder classified the three classes of lesions.

Training Procedure: The training parameters were similar to the ones described in III-B except that we used a learning rate of 0.0001 for 50 epochs. Our training inputs included liver lesion ROIs and their corresponding labels $(x_l, y_l)^{(1)}, ..., (x_l, y_l)^{(m)}$ for all lesion types $l \in (Cyst, Metastasis, Hemangioma)$, and noise samples $z^{(1)}, ..., z^{(m)}$ drawn from uniform distribution between $[-1, 1]$. The loss function needed to be modified to incorporate the label information. For simplification, let us write the basic GAN discriminator maximization equation over the log-likelihood (similar to Equation 1) as:

$$L = \mathbb{E}[\log P(S = real|X_{real})] + \mathbb{E}[\log P(S = fake|X_{fake})]$$

where $P(S|X) = D(X)$ and $X_{fake} = G(z)$. The generator is trained to minimize that objective. In ACGAN, the discriminator outputs $P(S|X), P(C|X) = D(X)$, and $X_{fake} = G(c, z)$ where C is the class label. The loss has two parts:

$$L_s = \mathbb{E}[\log P(S = real|X_{real})] + \mathbb{E}[\log P(S = fake|X_{fake})]$$
$$L_c = \mathbb{E}[\log P(C = c|X_{real})] + \mathbb{E}[\log P(C = c|X_{fake})]$$

The discriminator is trained to maximize $L_s + L_c$ and the generator is trained to maximize $L_c - L_s$.

IV. Experiments and Results

In the following we present a set of experiments and results. To test the classification results, we employed the CNN architecture described in Section II-B. We then analyzed the effects of data augmentation using synthetic liver lesions, as compared to classical data augmentation methodology. We implemented the two methods for synthetic lesion generation, as described in Sections III-B and III-C. In our experimentations we found that the Deep Convolutional GAN (DCGAN) method performed better. We therefore focus on that method in the results presented below. A comparison between the ACGAN and the DCGAN results will be presented in Section IV-E.

A. Dataset Evaluation and Implementation Details

In all experiments and evaluations we used 3-fold cross validation with case separation at the patient level. The number of examples in each fold was $(63, 63, 62)$ and each contained a balanced number of cyst, metastasis and hemangioma lesion ROIs. We evaluated the classification performance using a total classification accuracy measure. Additionally, we calculated confusion matrices and sensitivity and specificity measures for each lesion category. All these measures are presented in the following equations:

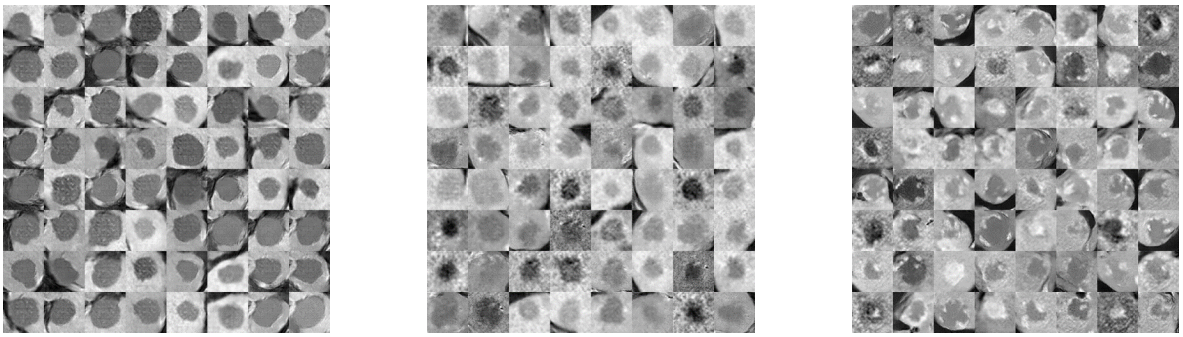
$$Total\ Accuracy = \frac{\sum TP}{Amount\ of\ lesions} \tag{2}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{4}$$

where for each lesion category, positives (P) are examples from this category and negatives (N) are examples from the other two categories.

For the implementation of the liver lesion classification CNN we used the Keras framework [39]. For the implementation of the GAN architectures we used the TensorFlow framework [40]. All training processes were performed using an NVIDIA GeForce GTX 980 Ti GPU.



(a) (b) (c)

Fig. 6. Synthetic liver lesion ROIs generated with DCGAN for each category: (a) Cyst examples (b) Metastasis examples (c) Hemangioma examples.

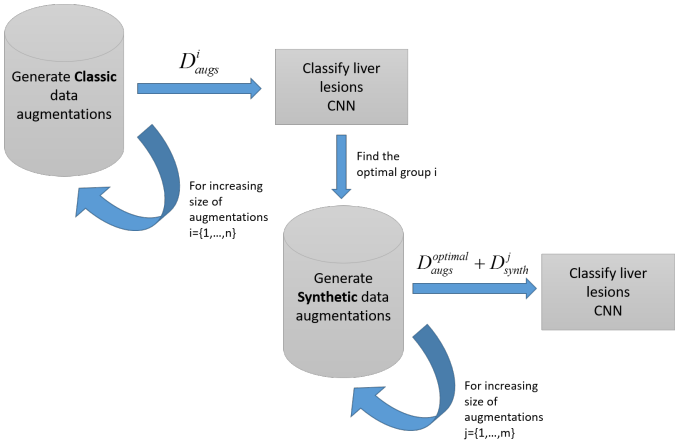


Fig. 7. Experiment flowchart for evaluating synthetic data augmentation in the task of classifying liver lesion ROIs.

B. Evaluation of the Synthetic Data Augmentation

Figure 7 presents the flowchart for the experiment conducted to evaluate the results from synthetic data augmentation: We started by examining the effects of using only classic data augmentation for the liver lesion classification task (our baseline). We then synthesized liver lesion ROIs using GAN and examined the classification results after adding the synthesized lesion ROIs to the training set. A detailed description of each step is provided next.

1) Classical data augmentation: As our baseline, we used classical data augmentation (see section III-A). We refer to this network as CNN-AUG. We recorded the classification results for the liver lesion classification CNN for increasing amounts of data augmentation over the original training set. We trained the network and evaluated the results separately for each set of data images (that included the original images and additional classic augmented images), as follows: Let $\{D_{aug}\}_{i=1}^{9}$ be the data groups that include increasing amounts of augmented examples for each training. During testing time, we used the same data examples for all evaluations. In order to examine the effect of adding increasing numbers of examples, we formed the data groups additively such that $D^1_{aug} \subset D^2_{aug} \subset ... \subset D^9_{aug}$. The first data group was only made up of the original ROIs. For each original ROI, we produced a large number of augmentations ($N_{rot} = 30$, $N_{flip} = 3$, $N_{trans} = 7$ and $N_{scale} = 5$), resulting in $N = 480$ augmented images per lesion ROI and overall $\sim 30,000$ examples per folder. Then, we selected the images for the data groups by sampling randomly augmented examples such that for each original lesion we sampled the same augmentation volume. To summarize the augmentation data group preparation process, the number of samples added to each fold (in our 3-folds) was $\{0, 500, 1000, 2000, 3000, 5000, 7500, 10000, 15000\}$. The training process was conducted by cross-validation over 3-folds, such that for each training group, the training examples were from two folds.

2) Synthetic data augmentation: The second step of the experiment consisted of generating synthetic liver lesion ROIs for data augmentation using GAN. We refer to this network as CNN-AUG-GAN. We took the optimal point for the classic augmentation $D^{optimal}_{aug}$ and used this group of data to train the GAN. Since our dataset was too small for effective training, we incorporated classic augmentation for the training process. We employed the DCGAN architecture to train each lesion class separately, using the same 3-fold cross validation process and the same data partition. After the generator had learned each lesion class data distribution separately, it was able to synthesize new examples by using an input vector of normal distributed samples ("noise"). Figure 6 presents examples of synthesized liver lesion ROIs from each class. The same approach that was applied in step one of the experiment when constructing the data groups was also applied in step two: We collected large numbers of synthetic lesions for all three lesion classes, and constructed data groups $\{D_{synth}\}_{j=1}^{6}$ of synthetic examples additively. To keep the classes balanced, we sampled the same number of synthetic ROIs for each class. To summarize the synthetic augmentation data group preparation process, the number of samples added to each fold (in our 3-folds) was $\{100 \times 3, 500 \times 3, 1000 \times 3, 2000 \times 3, 3000 \times 3, 4000 \times 3\}$.

Results of the GAN-based synthetic augmentation experiment are shown in Figure 8. The baseline results (classical augmentation) are shown in red. We see the total accuracy results for the lesion classification task, for each group of data.

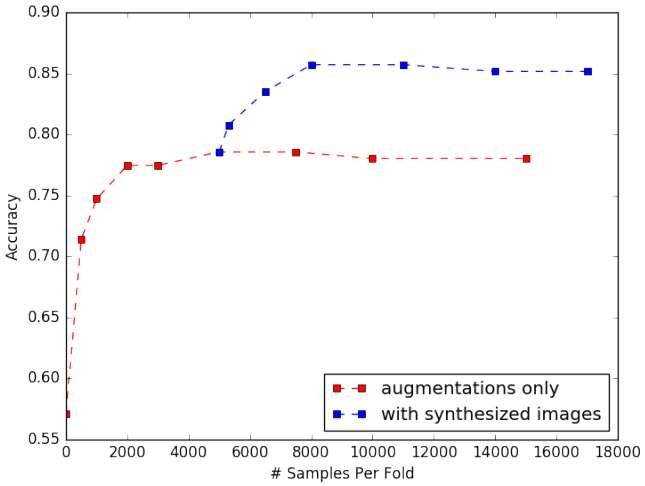


Fig. 8. Total accuracy results for liver lesion classification of cysts, metastases and hemangiomas with the increase of training set size. The red line shows the effect of adding classic data augmentation and the blue line shows the effect of adding synthetic data augmentation.

TABLE I
CONFUSION MATRIX FOR THE OPTIMAL CLASSICAL DATA AUGMENTATION GROUP (CNN-AUG)

True \ Auto	Cyst	Met	Hem	Sensitivity
Cyst	52	1	0	98.1%
Met	2	44	18	68.7%
Hem	0	18	47	72.3%
Specificity	98.4%	83.9%	84.6%	

TABLE II
CONFUSION MATRIX FOR THE OPTIMAL SYNTHETIC DATA AUGMENTATION GROUP (CNN-AUG-GAN)

True \ Auto	Cyst	Met	Hem	Sensitivity
Cyst	53	0	0	100%
Met	2	52	10	81.2%
Hem	1	13	51	78.5%
Specificity	97.7%	89%	91.4%	

When no augmentations were applied, a result of 57% was achieved; this may be due to overfitting over the small number of training examples ($\sim$ 63 samples per fold). The results improved as the number of training examples increased, up to saturation around 78.6% where adding more augmented data examples failed to improve the classification results. We note that the saturation starts with $D^6_{aug} = 5000$ samples per fold. We define this point as i=optimal where the smallest number of augmented samples were used. The confusion matrix for the optimal point appears in Table I.

The blue line in Figure 8 shows the total accuracy results for the lesion classification task for the synthetic data augmentation scenario. The classification results improved from 78.6% with no synthesized lesions to 85.7% for $D^{optimal}_{aug} + D^3_{synth} = 5000 + 3000 = 8000$ samples per fold. The confusion matrix for the best classification results using synthetic data augmentation is presented in Table II.

C. Visualization using t-SNE

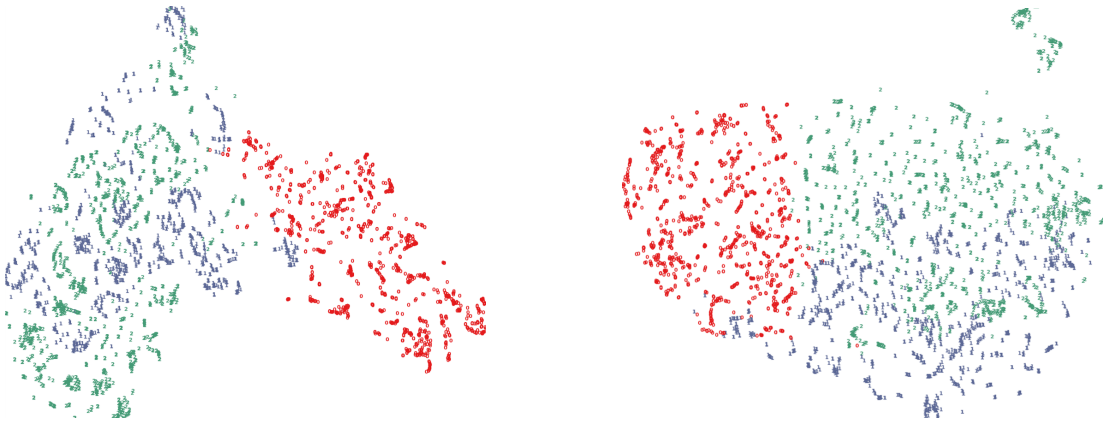
To further analyze the results, we used the t-SNE visualization. The t-SNE algorithm for dimensionality reduction enables the embedding of high-dimensional data into a two dimensional space [41]. The high-dimensional data for visualization are features extracted from the last layer of a trained liver lesion classification CNN. We trained the CNN in two scenarios: one with the classic augmented data examples (CNN-AUG) and one with the synthesized data examples (CNN-AUG-GAN). Afterwards, for each scenario, we extract the features of real images from the test set and their classic augmentations. We then used the t-SNE to illustrate the features, as shown in Figure 9 (a) and (b), respectively.

We note that the cyst category, shown in red, shows a more distinct localization in the t-SNE space. This characteristic correlates well with the more distinctive features of the cyst class as compared to metastases or hemangiomas. Metastases and hemangiomas have confusing features, which is indicated here in the perceived overlap and accounts for the lower sensitivity and specificity results than in the cyst class. When using the synthetic data augmentation, the t-SNE visualization exhibited in general better separating power. This can provide intuition for the increase in classification performance.

D. Expert Assessment of Synthetic Data

Human annotators have been shown to evaluate the visual quality of samples generated by GANs [9], [36]. In our study, we were interested to explore two key points: Is the synthesized lesions appearance a realistic one? Is the set of lesions generated sufficiently distinct to enable classification amongst the three lesion categories? These issues were explored with the help of two expert radiologists.

We created an automatic application which was presented to two independent radiologists, with two tasks. One task was to classify each presented lesion ROI image into one of three classes: cyst, metastasis or hemangioma. The second task was



(a) (b)

Fig. 9. T-SNE embedding of Cysts (red), Metastases (blue) and Hemangiomas (green) real lesion ROIs. (a) Features extracted from CNN-AUG (b) Features extracted from CNN-AUG-GAN.

TABLE III
SUMMARY OF EXPERTS' ASSESSMENT OF LESION ROI

	Classification Accuracy			Is ROI Real?
	Real	Synthetic	Total Score	Total Score
Expert 1	78%	77.5%	235\302=77.8%	189\302=62.5%
Expert 2	69.2%	69.2%	209\302=69.2%	177\302=58.6%

TABLE IV
PERFORMANCE COMPARISON FOR LIVER LESION CLASSIFICATION BETWEEN GENERATIVE MODELS

Method	Sensitivity	Specificity
CNN-AUG-GAN (DCGAN)	**85.7%**	**92.4%**
CNN-AUG-GAN (ACGAN)	81.3%	90.0%
ACGAN discriminator	79.1%	88.8%

to distinguish between real lesion images and synthetic lesion images. The experts were given, in random order, lesion ROIs from the original dataset of 182 real lesions and 120 additional synthesized lesions. Both our algorithm results and the expert radiologists' results were compared against the ground truth classification.

Table III summarizes the experts' results. We note the overall low results, on the order of 60%, in identifying whether the lesions shown were true or fake. In the lesion categorization task, Expert 1 and Expert 2 classified correctly in 77.8% and 69.2% of the cases, respectively. Overall, the radiologists agreed on the lesion class on 222 out of 302 lesions (73.5%), with a correct classification of 185 out of 302 lesions. In addressing these results, it is important to note that the task we defined was not consistent with existing clinical workflow. The radiologist is trained to make a decision based on the entire 3D volume, with support from additional anatomical context, medical history context, and more. Here, we challenged the radiologists to reach a decision based on a single 2-D ROI image. In this scenario, the baseline CNN solution is similar in performance to the human expert. Using the GAN-based augmentation, an increase of approx 7% is achieved.

As a final note, we observe that for both experts, the classification performances for the real lesions and the synthesized lesions were similar. which suggests that our synthetic generated lesions were meaningful in appearance.

E. Comparison with Other Classification Methods

Table IV compares the best classification results between the DCGAN and ACGAN models. As described above, we used synthetic augmentations generated using the DCGAN for training the classification CNN (CNN-AUG-GAN). Training the classification CNN with synthetic augmentations generated using the ACGAN, yield improved results in comparison of using only classic augmentations, but degraded results in comparison to the DCGAN. The ACGAN discriminator contains an auxiliary classifier. Thus, after training the ACGAN, we can use the learned discriminator as an autonomous component to test directly the test set performance. Using this method resulted in $\sim 2\%$ decrease in performance.

In our final experiment, we compared our CNN classification results for classic augmentation (CNN-AUG) and synthetic augmentation (CNN-AUG-GAN), to a recently published state-of-the-art liver lesion categorization method, termed BoVW-MI [30]. The BoVW-MI method is an enhancement of the BoVW model. It learns a task-driven dictionary of the most relevant visual words per task using a mutual information measure. In order to compare between the approaches, using the datasets of the current work, we ran the BoVW-MI method using the specified optimized parameters for the liver lesion classification task, as found in [30]: A patch size of 11×11, a word size with a 10 PCA coefficient, a dictionary size of 750 words and a MI threshold of 35%. We trained the BoVW-MI in 3-fold cross validation using the same lesion partitions. Table V compares the sensitivity and specificity results of our best results to the BOVW-MI results.

V. DISCUSSION AND CONCLUSIONS

This work focused on generating synthetic medical images with GAN for data augmentation to enlarge small datasets and

TABLE V
PERFORMANCE COMPARISON FOR LIVER LESION CLASSIFICATION BETWEEN CNN AND BOVW-MI

	CNN-AUG-GAN		CNN-AUG		BOVW-MI	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
Cysts	100%	97.7%	98.1%	98.4%	96.3%	96.9%
Metastases	81.2%	89.0%	68.7%	83.9%	75.0%	82.2%
Hemangiomas	78.5%	91.4%	72.3%	84.6%	66.1%	87.2%
Weighted Average	**85.7%**	**92.4%**	78.6%	88.4%	78.0%	88.3%

improve performance on classification tasks using CNN. Our relatively small dataset reflects the size of datasets available to most researchers in the medical imaging community (by contrast to the computer vision community where large scale datasets are available).

We tested our hypothesis that adding synthesized examples would improve classification results. The experimental setup is depicted in Figure 7. The experiment was carried out on a limited dataset of three liver lesion categories of cysts, metastases and hemangiomas. Each class has its unique features but there is also considerable intra-variability between classes, mostly for the metastases and hemangiomas. We classified the three categories using a CNN architecture. In running the experiment, we found that increasing the size of the training data groups with the standard augmentation (translation, rotation, flip, scale), improved training results up to a certain volume of augmented data, where adding more data did not improve the results (Figure 8). Table I shows the results for the optimal point achieved using the commonly used classic augmentation.

In the second step of the experiment we used GANs to generate new examples learned from our small dataset. The best generated liver lesion samples were produced by using the Deep Convolution GAN (DCGAN) for each lesion class separately. Starting from the optimal point where classic augmentation reached saturation, we applied increasing sizes of synthetic data. We saw an improvement in the classification results from 78.6% to 85.7% total accuracy (Figure 8). We see increase in the sensitivity and specificity of the metastasis and hemangiomas classes. It seems that the synthetic data samples generated from a given dataset distribution, using GAN, can add additional variability to the input dataset (Figure 9), that in turn leads to better performance.

Evaluations of the quality of the synthesized liver lesions were made by two expert radiologists. Although the experiment was not conducted in the regular radiologist working environment, and proved to be a challenging task for them, we find it of interest that both experts had the same classification accuracy results for the real set, as well as the synthesized lesions set (Table III), indicating to us the validity of the lesion generation process.

In this study, our goal was to assess to what extent synthesized lesions can improve the performance of another system behind the scenes. Our results show that the synthesized lesions have meaningful visualizations and more importantly meaningful features and can be incorporated into computer aided algorithms.

We tested another generative model that incorporated labels in the training process. Both GANs were trained using supervised learning with liver lesion class labels. The DCGAN trained each lesion class separately while the ACGAN trained all three lesion classes at once. In recent computer vision studies [11], [36], training a GAN that combines label information improved the visualization quality of samples over GANs that did not utilize the label information to generate samples of many classes together. Somewhat surprisingly, we found that for our dataset, challenging the discriminator network to perform two tasks (distinguishing real or fake and classifying lesions into 3 categories), resulted in poor results in comparison the DCGAN model. Using synthetic augmentation generated using the ACGAN, we were not able to improve the results over the CNN-AUG-GAN (Table IV).

As a final experiment, we compared the performance of the CNN - based system which we propose in this work, to non-network state-of-the-art methods for liver lesion classification (Table V). Our suggested CNN architecture for classification that employs classic augmentation performed on a par with the BoVW-MI method [30] with the same ROI input. Using synthetic data augmentation in our CNN architecture led to the best performance.

There are several limitations to this work. One possible extension could be an increase from 2-D to 3-D input volumes, using 3-D analysis CNN. We trained separate GANs for each lesion class which increased the training complexity. Investigation of GAN architectures that generate multi-class samples together would be worthwhile. The quality of the generated lesion samples could possibly be improved by incorporating unlabeled data to improve the GAN learning process [36]. Further analysis into modifications of the training loss to incorporate regularization terms for the L1-norm or L2-norm, can be investigated as well [13], [18]. In the future, we plan to extend our work to additional medical domains that can benefit from synthesis of lesions for improved training.

In conclusion, we presented a method that uses the generation of synthetic medical images for data augmentation to improve performance on a medical problem with limited data. We demonstrated this technique on a liver lesion classification task and achieved an improvement of $\sim 7\%$ using synthetic augmentation over the classic augmentation. We introduced a CNN-based architecture for the liver lesion classification task, that achieves state-of-the-art results. We believe that other medical problems can benefit from using synthetic

augmentation, and that the presented approach can lead to stronger and more robust radiology support systems.

REFERENCES

[1] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M. Summers, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1170–1181, May 2016.

[2] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *arXiv preprint arXiv:1702.05747*, 2017.

[3] H. Greenspan, B. van Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1153–1159, May 2016.

[4] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1299–1312, May 2016.

[5] J. Shi, S. Zhou, X. Liu, Q. Zhang, M. Lu, and T. Wang, "Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset," *Neurocomputing*, vol. 194, pp. 87–94, 2016.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

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

[9] E. L. Denton, S. Chintala, R. Fergus *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Advances in neural information processing systems*, 2015, pp. 1486–1494.

[10] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[11] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," *arXiv preprint arXiv:1610.09585*, 2016.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint arXiv:1611.07004*, 2016.

[13] R. Yeh, C. Chen, T. Y. Lim, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with perceptual and contextual losses," *arXiv preprint arXiv:1607.07539*, 2016.

[14] P. Costa, A. Galdran, M. I. Meyer, M. D. Abràmoff, M. Niemeijer, A. M. Mendonça, and A. Campilho, "Towards adversarial retinal image synthesis," *arXiv preprint arXiv:1701.08974*, 2017.

[15] W. Dai, J. Doyle, X. Liang, H. Zhang, N. Dong, Y. Li, and E. P. Xing, "Scan: Structure correcting adversarial network for chest x-rays organ segmentation," *arXiv preprint arXiv:1703.08770*, 2017.

[16] Y. Xue, T. Xu, H. Zhang, R. Long, and X. Huang, "Segan: Adversarial network with multi-scale l_1 loss for medical image segmentation," *arXiv preprint arXiv:1706.01805*, 2017.

[17] D. Nie, R. Trullo, C. Petitjean, S. Ruan, and D. Shen, "Medical image synthesis with context-aware generative adversarial networks," *arXiv preprint arXiv:1612.05362*, 2016.

[18] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*. Springer, 2017, pp. 146–157.

[19] V. Alex, M. S. KP, S. S. Chennamsetty, and G. Krishnamurthi, "Generative adversarial networks for brain lesion detection," in *SPIE Medical Imaging*. International Society for Optics and Photonics, 2017, pp. 101 330G–101 330G.

[20] A. Ben-Cohen, E. Klang, S. P. Raskin, M. M. Amitai, and H. Greenspan, "Virtual pet images from ct data using deep convolutional networks: Initial results," *arXiv preprint arXiv:1707.09585*, 2017.

[21] National cancer institute. [Online]. Available: https://www.cancer.gov/types/metastatic-cancer

[22] J. Ferlay, I. Soerjomataram, R. Dikshit, S. Eser, C. Mathers, M. Rebelo, D. M. Parkin, D. Forman, and F. Bray, "Cancer incidence and mortality worldwide: sources, methods and major patterns in globocan 2012," *International journal of cancer*, vol. 136, no. 5, 2015.

[23] T. Murakami, Y. Imai, M. Okada, T. Hyodo, W.-J. Lee, M.-J. Kim, T. Kim, and B. I. Choi, "Ultrasonography, computed tomography and magnetic resonance imaging of hepatocellular carcinoma: toward improved treatment decisions," *Oncology*, vol. 81, no. Suppl. 1, pp. 86–99, 2011.

[24] M. Gletsos, S. G. Mougiakakou, G. K. Matsopoulos, K. S. Nikita, A. S. Nikita, and D. Kelekis, "A computer-aided diagnostic system to characterize ct focal liver lesions: design and optimization of a neural network classifier," *IEEE Transactions on Information Technology in Biomedicine*, vol. 7, no. 3, pp. 153–162, Sept 2003.

[25] A. Adcock, D. Rubin, and G. Carlsson, "Classification of hepatic lesions using the matching metric," *Computer vision and image understanding*, vol. 121, pp. 36–42, 2014.

[26] C.-C. Chang, H.-H. Chen, Y.-C. Chang, M.-Y. Yang, C.-M. Lo, W.-C. Ko, Y.-F. Lee, K.-L. Liu, and R.-F. Chang, "Computer-aided diagnosis of liver tumors on computed tomography images," *Computer Methods and Programs in Biomedicine*, vol. 145, pp. 45–51, 2017.

[27] M. Bilello, S. B. Gokturk, T. Desser, S. Napel, R. B. Jeffrey, and C. F. Beaulieu, "Automatic detection and classification of hypodense hepatic lesions on contrast-enhanced venous-phase ct," *Medical physics*, vol. 31, no. 9, pp. 2584–2593, 2004.

[28] S. G. Mougiakakou, I. K. Valavanis, A. Nikita, and K. S. Nikita, "Differential diagnosis of ct focal liver lesions using texture features, feature selection and ensemble driven classifiers," *Artificial Intelligence in Medicine*, vol. 41, no. 1, pp. 25–37, 2007.

[29] I. Diamant, A. Hoogi, C. F. Beaulieu, M. Safdari, E. Klang, M. Amitai, H. Greenspan, and D. L. Rubin, "Improved patch-based automated liver lesion classification by separate analysis of the interior and boundary regions," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 6, pp. 1585–1594, Nov 2016.

[30] I. Diamant, E. Klang, M. Amitai, E. Konen, J. Goldberger, and H. Greenspan, "Task-driven dictionary learning based on mutual information for medical image classification," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 6, pp. 1380–1392, June 2017.

[31] S. A. Napel, C. F. Beaulieu, C. Rodriguez, J. Cui, J. Xu, A. Gupta, D. Korenblum, H. Greenspan, Y. Ma, and D. L. Rubin, "Automated retrieval of ct images of liver lesions on the basis of image similarity: Method and preliminary results," *Radiology*, vol. 256, no. 1, pp. 243–252, 2010.

[32] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016.

[33] A. A. A. Setio, F. Ciompi, G. Litjens, P. Gerke, C. Jacobs, S. J. van Riel, M. M. W. Wille, M. Naqibullah, C. I. Snchez, and B. van Ginneken, "Pulmonary nodule detection in ct images: False positive reduction using multi-view convolutional networks," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1160–1169, May 2016.

[34] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[35] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence o (1/k2)," in *Doklady an SSSR*, vol. 269, no. 3, 1983, pp. 543–547.

[36] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, 2016, pp. 2234–2242.

[37] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015, pp. 448–456.

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

[39] F. Chollet *et al.* (2015) Keras. [Online]. Available: https://github.com/fchollet/keras

[40] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[41] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.