The Effect of Data Augmentation on COVID-19 Thoracic CT Images for Semantic Segmentation Using a Deep U-net Convolutional Neural Network

Maximilian Auer maue@kth.se
Kristin Evegård evegard@kth.se
Lukas Frösslund lukasfro@kth.se
Valdemar Gezelius vgez@kth.se

Abstract

The novel coronavirus disease 2019 (COVID-19) pandemic has had destructive effects on global public health. With limitations in current testing, effective methods in screening for COVID-19 and quantifying its harmful effects on lungs remains a considerably important task. Semantic segmentation of computed tomography (CT) scans plays a central role in the quantification and accurate classification of infection caused by COVID-19. A current issue though is the absence of large, annotated datasets of CT-scans from patients diagnosed with the virus. To this end, we investigate the effects of data augmentation, the various techniques to artificially expand a given dataset by methods of modifying existing data, on a small dataset of thoracic COVID-19 CT-scans. We use the U-net network architecture, a network constructed to perform biomedical image segmentation accurately despite few available training samples. In an ablation study, we validate the efficacy of several augmentation methods, including elastic deformations, a method which has shown encouraging results in similar segmentation tasks. These experimental results show that the U-net architecture can achieve accurate segmentation on COVID-19 lesion regions despite lack of data. The use of data augmentation demonstrates a promising increase in performance and our study can act as a foundation for further exploration of the effect of augmentation on COVID-19 CT-scans for semantic segmentation.

1 Introduction

Ever since the outbreak of COVID-19 was declared a pandemic in February 2020 [1], the reports of how this pandemic exposes the healthcare system to an enormous workload have been a common feature of news reporting worldwide. One effective step in detecting COVID-19 in a patient that is suspected to be a carrier, is to do a chest computed tomography (hereinafter referred to as a CT-scan, *see* Figure 2) and analyse the patient's lungs for regions of interest (hereinafter referred to as RIO's) that could be infected areas associated with COVID-19. Accurate segmentation is crucial for reliable quantification of COVID-19 infection in chest CT images [2]. This helps the medical experts to make a decision on to which extent the lungs are infected, and from these CT-scans it is possible to decide which further steps that should be taken and which treatment a patient is in need of [3]. The process of finding RIO's is time consuming and performed by personnel that, in hectic times, has more urgent duties.

For at least the past five years, we have seen an increase in computer-aided diagnosis, i.e. in detecting pulmonary tuberculosis from CT-scans [4]. This, as a result of the increased amount of data collected and AI-systems improving, especially since the introduction of the deep convolutional neural network U-net architecture [5]. The task at hand falls under the category of segmentation. These algorithms segment out specific RIO's, which creates a mask from, in our case a CT-scan, that highlights where in the scan doctors should focus their attention.

We trained a U-net architecture to segment out COVID-19 in CT-scans. Since the data collected on COVID-19 is limited, we implemented several data augmentation techniques to increase the dataset and prevent overfitting [4]. We also implemented the regularization technique; Batch Normalization. This is a technique that has been popularised after the introduction of the U-net, and is therefore not included in the original net architecture [5]. We trained several models mixing different combinations of augmentation- and regularization techniques. The results could indicate that when segmenting CT-scans using a deep CNN encoder/decoder-architecture [5], elastic deformation augmentation techniques could be effective. Our best model uses elastic deformations and scores $\approx$0.85 on *Sensitivity*, $\approx$0.99 on *Specificity* and obtains a *Dice Score* of $\approx$0.89.

2 Related Work

As of writing this report, no cure of COVID-19 has been found. Finding and tracing the infection is the best way of controlling its spread. One of the current and most used methods for testing is the Real-Time Reverse Transcription-polymerase Chain Reaction (Real-Time RT-PCR). This method is time consuming and the demand for the test kits is higher than the supply, which has led to a global shortage [6][7]. The need for alternative methods of diagnosis is large.

A method that has shown signs of promise is the use of CT-scanning of the lungs. This method is already used to identify other diseases that cause pneumonia, which can be a symptom and result of contracting COVID-19. The CT-scans can be analysed to determine how acute the effect on the lungs are and how the disease is progressing. This can be used to determine treatment for patients with COVID-19 or other pneumonia causing diseases. By checking the CT-scan of the lungs for specific patterns of Ground Glass Opacities (GGO's), which are areas of increased deterioration in the lungs, a doctor can deduce whether or not the patient has COVID-19 [8][9][10][11][12]. The problem with this approach is the need for radiologists to highlight ROI's and in turn evaluate them. This process is labour intensive and requires a large amount of work from the already limited healthcare staff [2].

For automating the marking of these ROI's, deep convolutional neural networks (CNN) are often used for their ability to effectively segment images. Segmentation is an important part of a machine learning approach to diagnose and evaluate the disease's impact on the lungs. Segmentation is in the case of COVID-19 used both for identifying the lung and the possible infected areas of it [13]. When segmenting medical images, CNN are the standard approach [13][14] and seems to work best, or at least give the most reasonable results [15][16][17][18].

There are multiple architectures of CNN's for segmenting medical images and many stem from U-Net. U-Net is a CNN architecture created specifically for segmenting medical images, which it is successful at to a reasonable level. Since its first appearance in 2015 it has been widely used [5]. This network, or some variant of it, has been used in the specific task of segmenting COVID-19 CT-scans in multiple cases [3][2][4][19]. A big part of U-Net and making it work on very small datasets is the use of data augmentation [5]. This augmentation of the data, in our case CT-scans, can be as simple as rotating, translating, and shifting the image, or even something much more complex.

The use of data augmentation is important for the learning of invariance [20]. The analysis of CT-scans does however pose some challenges. For example, a photograph of a human is still a human even if we move the person up or down. Making such augmentations does not affect the information in the photograph, and the model will improve the learning of the invariance of the desired feature by using data augmentation [21]. A possible problem arises in the case of CT-scans, where the signs of the disease are heavily dependent on the localization of the above mentioned features in the medical image [22]. Therefore it is of high importance that the effects of data augmentation on the learning process of CNN's on CT-scans are further studied, especially in the case of COVID-19, where the amount of training data is limited and the need for medical support is high.

3 Data

3.1 Data Description

For our experiment we used a COVID-19 axial CT segmentation dataset, curated from the international, open-edit collaborative radiology resource Radiopaedia[1]. The dataset contains 9 whole volumes of axial CTs for a total of 829 slices. Ground-truth segmentation has been performed by professional radiologists and image annotators using their own medical segmentation tool MedSeg[2], constructing feature masks with three labels: ground-glass, consolidation and pleural effusion. Due to drastic data imbalance and usability, all three labels are combined to constitute a COVID-19 lesion region.

3.2 Data Preprocessing and Preparation

A number of preprocessing steps were executed by the same team of trained radiologists. The images were greyscaled and compiled into NIFTI-files, one for each whole volume of CT slices. They were then reversely intensity-normalized to the Hounsfield Unit quantitative radiodensity scale.

We used the neuroimaging python library nibabel[3] to access the NIFTI-files, and concatenated the data to form one cohesive tensor. All images were then resized to $512x512$ in pixel size, and normalized to make all pixel values range between 0 and 1. The data was split randomly into 80% training data and 20% test data, at which point both sets were also randomly shuffled. 10% of the training portion of the data was later set aside for validation.

3.3 Data Augmentation

With only few available training samples, data augmentation can act as an essential component in achieving a robust network model that can generalize effectively. In the sphere of medical image segmentation (most notably CT-scans and microscopical imaging), several augmentation methods have shown promising results on small datasets, including simple transformations such as horizontal flips, small rotations, shifts in height and width, and scaling [5][2][4]. To these, we have added two additional augmentation methods in zooms (in the range 0.9-1.1) and channel shifts. Together they constitute our basic augmentations.

For medical imaging, elastic deformations of the images have achieved especially promising results [5][23]. While the exact reason for these encouraging results have not yet been scientifically examined, it has been argued that its strength stems from how it effectively and realistically simulates the common deformations in human tissue. This study aims to further investigate the usefulness of elastic deformations. As far as we know, augmentation through elastic deformations has never yet been examined for CT-scans with COVID-19 lesion regions.

4 Methods

4.1 Network Architecture

The aim of this project falls under the category of medical image segmentation. In related work on the subject, we found that an already developed convolutional neural network named U-Net has been frequently used [2][5][3]. It is a network that was developed specifically for the segmentation of biomedical images [5], and therefore seemed like the most natural choice of architecture to choose for our project. We also considered using another network architecture called ResNet, which has the capability to mitigate the effect of a problem that is exposed when deeper networks converge [2]. However, due to the frequency of the U-Net architecture in closely related works, and as we aim to study the effect of using different data augmentation methods rather than the implementation of the network, we considered the U-Net architecture to be the most convenient alternative to use.

[1] https://radiopaedia.org/

[2] https://www.medseg.ai/

[3] https://nipy.org/nibabel/

Hence, our network is based on the U-Net architecture [5]. The overall structure is shown in Figure 1 (the picture is borrowed from [5]). As clearly shown, the U-Net architecture is symmetrical with two paths: the contracting path (the encoder) and the expansive path (the decoder).

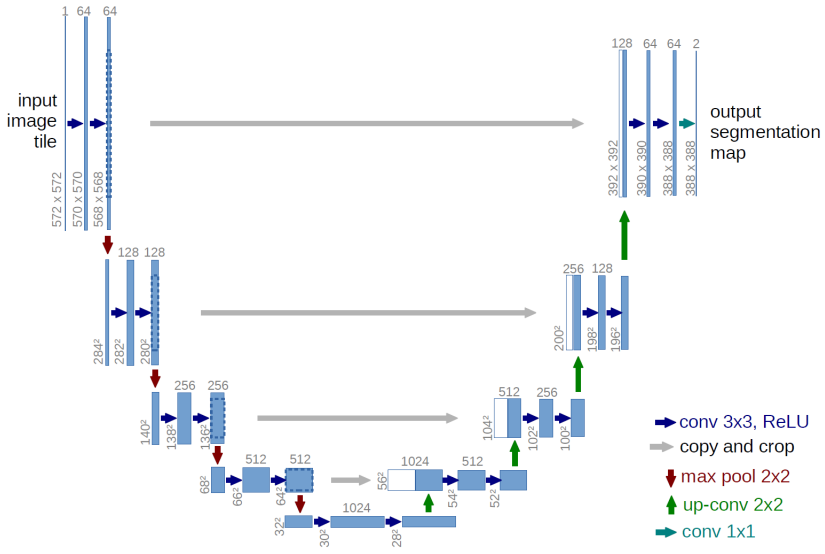

Figure 1: U-Net architecture

The contracting path consists of the repeated application of two 3x3 convolutions. Each convolution is followed by a rectified linear unit (ReLU), and before each ReLU we added Batch Normalization to further optimize and stabilize the learning process. Between the two 3x3 convolutions we have also made it possible to add a Dropout. Finally, a 2x2 Max Pooling operation with stride 2 for downsampling is added. At each downsampling step we doubled the number of feature channels.

The expansive path is not too different from the contracting path, but instead of performing down-sampling it is designed for up-sampling and constructing the segmentation image. Every step in the expansive path consists of an upsampling of the feature map followed by two 3x3 convolutions, each followed by a ReLU. In addition, we have also added Batch Normalization before each ReLU and the possibility to add a Dropout between the two 3x3 convolutions.

At the final layer a 1x1 convolution is used to map each feature vector to the desired number of classes, i.e. to generate the segmentation result.

4.2 Regularization Methods

Apart from our focus regarding different data augmentation methods, we have also looked at different regularization methods for further optimization of our model. We decided to use the two most powerful optimization techniques out today: Dropout and Batch Normalization (BN) [24]. BN has the capability to not only speed up all the modern architectures but also improve upon their strong baselines by acting as regularizers. Due to this, it is now implemented in nearly all recent network structures [24]. Adding Dropout to the network structure is a simple way to prevent the neural network from overfitting. One interesting aspect though, is that it often leads to a worse performance when they are combined together [24]. This is something we further wanted to investigate and analyze through our experiments.

4.3 Loss function

We tested our model using 3 different loss functions: binary cross-entropy loss, weighted binary cross-entropy dice loss and dice loss. In our Ablation study (*see* Table 1) we decided on binary cross-entropy loss ($\mathcal{L}$), since it yielded the best results.

The task of segmenting an image means that every single pixel is given a class, in our case a pixel can be classified as being part of the background or being part of the mask (as when overlaid on the CT-scan highlights the RIO). Following is the formula for binary cross-entropy:

$$\mathcal{L}\left(y, \hat{y}\right) = -\frac{1}{N}\sum_{i=0}^{N}(y_i * log(\hat{y_i})) + (1 - y_i) * log(1 - \hat{y_i}) \tag{1}$$

where y is the label, $\hat{y}$ is the predicted value and N is the amount of pixels in the image. We see that when y_i equals 1, we add $log(\hat{y_i})$ to the loss, a loss that will be big when $\hat{y_i}$ is close to 0 but decrease when $\hat{y_i}$ gets closer to 1. The same reasoning holds true when y_i equals 0, then $log(1 - \hat{y_i})$ is added to the loss. This loss is big when $\hat{y_i}$ is close to 1 but decreases when $\hat{y_i}$ gets closer to 0 (the ground truth for the i-th pixel).

5 Experiments

5.1 Implementation Details

Our network was trained on a single Nvidia Tesla T4 GPU, with the binary crossentropy loss function and the Adam [25] optimizer. Parameters of the optimizer incorporated an initial learning rate of $5e^{-4}$, $\beta_1 = 0.9$, $\beta_2 = 0.99$, $\epsilon = 10e^{-8}$ and a learning rate decay of 0. The U-net model was implemented in Tensorflow and Keras, and was trained for 100 epochs with a batch size of 8. Early stopping was used with a patience set to 10, and the model was validated after each epoch. A model checkpointer was introduced to save the best model from the training process, to then use that same model for the evaluation process on the test data.

5.2 Evaluation Metrics

Evaluation metrics calculated in the model included the number of True Positives (*TP*), True Negatives (*TN*), False Positives (*FP*) and False Negatives (*FN*). These numbers were later used to calculate our three main quantitative measurements for the semantic segmentation performance on the proposed methods: *Dice Score*, *Sensitivity* and *Specificity*. All three quantitative measurements have been chosen due to their effectiveness for the semantic segmentation task, as well as their frequency in closely related research [3][4][2]. Pixel accuracy was deemed unsuitable because of severe class imbalance in the feature masks (a large majority of negative pixels).

Dice Score (also known as the Dice Coefficient or F1 Score) act as a similarity measure, or an overlap rate, between the ground truth and prediction mask, by dividing the size of the overlap by the total size.

$$Dice\ Score = \frac{2TP}{2TP + FP + FN} \tag{2}$$

Sensitivity (also known as the True Positive Rate or Recall), is a proportional measurement of the correctly classified positive pixels, which here refers to COVID-19 lesion regions. *Specificity* (or the True Negative Rate) constitutes the proportion of negative pixels that are correctly classified.

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

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

5.3 Ablation Study

The main experiment of this study consists of an ablation study, summarized in Table 1. We trained 16 models, each with a different combination of input data and regularization techniques. The corner stones of the combinations was input data with no augmentation added, basic augmentation added and elastic deformation augmentation added (as described in 3.3), and regularization techniques

using batch normalization and dropout layers (as described in 4.3).

Table 1: Ablation study

Model	Combination	Sensitivity	Specificity	Dice score
1	na^{*}	0.7691	0.9983	0.7273
2	$na^{*} + bn^{\circ}$	0.7996	0.9994	0.8107
3	$na^{*} + bn^{\circ} + drop^{\bullet}$	0.6576	0.9997	0.7737
4	$na^{*} + drop^{\bullet}$	0.8172	0.9995	0.8657
5	$ba^{\star}$	0.7426	0.9993	0.8060
6	$ba^{\star} + bn^{\circ}$	0.7937	0.9984	0.7928
7	$ba^{\star} + bn^{\circ} + drop^{\bullet}$	0.8627	0.9976	0.7883
8	$ba^{\star} + drop^{\bullet}$	0.6926	0.9995	0.7860
9	$ea^{\diamond}$	0.8630	0.9992	0.8763
10	$ea^{\diamond} + bn^{\circ}$	0.8603	0.9993	0.8792
11	$ea^{\diamond} + bn^{\circ} + drop^{\bullet}$	0.8261	0.9993	0.8594
12	$ea^{\diamond} + drop^{\bullet}$	0.8515	0.9995	0.8855
13	$ba^{\star} + ea^{\diamond}$	0.7471	0.9994	0.8195
14	$ba^{\star} + ea^{\diamond} + bn^{\circ}$	0.8413	0.9989	0.8459
15	$ba^{\star} + ea^{\diamond} + bn^{\circ} + drop^{\bullet}$	0.8360	0.9991	0.8570
16	$ba^{\star} + ea^{\diamond} + drop^{\bullet}$	0.8149	0.9993	0.8526

* No augmentation, $^{\circ}$ Batch normalization layers, $^{\bullet}$ Dropout layers, * Basic augmentation
$^{\diamond}$ Elastic deformation augmentation

5.4 Discussion

After examining the results from the 16 different models presented in Table 1, we can observe that while there are general trends, patterns and differences to be noted, the discrepancy across all quantified metrics between the models is not very significant. Overall we can identify a high Specificity in the trained models, due to severe data imbalance between positive and negative pixels, which results in a large proportional difference between True Negatives and False Positives. While the exceedingly large Specificity values still points to a good performing model, the metric does become flawed because of this. Out of the other two metrics, we deem Dice Score to be the most reliable because it is derived from the more comprehensive overlap rate between the ground-truth mask and the prediction rather than the Sensitivity metric, which only focuses on the pixels classified as positive.

We see that augmentation impacts a models dice score, and particularly elastic deformation augmentation seems to have a positive impact with models $9 - 16$ - all having a dice score > 0.8. Even though we see that the dice scores get better overall on the models that use augmentations on the data, the difference is not significant. This could be because of the U-net architecture's robustness; that it has a high benchmark level on segmenting tasks, but also due to the fact that the test data is very similar to the training data (when no augmentation is added). This could imply that it will perform well on the test data, but would perhaps not generalize well.

We see no real difference between the models that make use of regularization with batch normalization and/or dropout layers, and models without regularization. This could again be due to the relative similarity between training data and test data. As mentioned in section 4.2, other research has indicated that using batch normalization and dropout together in a network often leads to worse performance than when used on its own[24]. In our study we do not encounter this performance issue, neither do we see the opposite effect.

Given that relatively contrasting model configurations resulted in a fairly similar outcome across our performance metrics, one naturally begins to reflect over possible shortcomings in the training process. With it being a segmentation task rather than a classification problem, and with the two output classes being greatly imbalanced, a loss function more tailored to these properties might have yielded more unambiguous results. A modification of the early stopping regularization method, with either a longer patience or a complete removal of this feature, might have allowed for a more significant difference between the models. With the current configuration, models integrating the

basic augmentation methods converged at earlier epochs, with a higher training loss (most likely due to the basic augmentation generating new random modifications for each epoch). This might have resulted in a small dissimilarity between models integrating basic augmentation, and those that didn't.

Figure 2: Qualitative performance of model 12. **Left:** Thoracic CT-scan. **Middle:** Ground Truth Mask. **Right:** Prediction Mask.

6 Conclusion

From our results we see that augmenting the input data to our model helps the learning process, especially with the use of elastic deformations. The use of a U-net architecture suits the task of segmenting RIO's in CT-scans containing COVID-19. Future studies could be made on other augmentation techniques to further improve the task of medical image segmentation, and narrow our study down, focusing on a couple of techniques e.g which basic augmentation that has the most impact instead of bundle them together. An interesting subject is whether the research and knowledge in trained models on the task of segmenting out COVID-19 in CT-scans could transfer to future medical situations.

References

[1] Rolling updates on coronavirus disease (covid-19), world health organization. accessed on: May. 17, 2020. [online] available: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/events-as-they-happen.

[2] Xiaocong Chen, Lina Yao, and Yanyan Zhang. Residual attention u-net for automated multi-class segmentation of covid-19 chest ct images. *ArXiv*, abs/2004.05645, 2020.

[3] Tongxue Zhou, Stéphane Canu, and Su Ruan. An automatic covid-19 ct segmentation network using spatial and channel attention mechanism. *ArXiv*, abs/2004.06673, 2020.

[4] Wei Wu, Xukun Li, Peiwei Du, Guan jing Lang, Min Xu, Kaijin Xu, and Lanjuan Li. A deep learning system that generates quantitative ct reports for diagnosing pulmonary tuberculosis. *ArXiv*, abs/1910.02285, 2019.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[6] Ali Narin, Ceren Kaya, and Ziynet Pamuk. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *ArXiv*, abs/2003.10849, 2020.

[7] Yan Li and Liming Xia. Coronavirus disease 2019 (covid-19): role of chest ct in diagnosis and management. *American Journal of Roentgenology*, pages 1–7, 2020.

[8] Junqiang Lei, Junfeng Li, Xun Li, and Xiaolong Qi. Ct imaging of the 2019 novel coronavirus (2019-ncov) pneumonia. *Radiology*, 295(1):18–18, 2020.

[9] Lin Li, Lixin Qin, Zeguo Xu, Youbing Yin, Xin Wang, Bin Kong, Junjie Bai, Yi Lu, Zhenghan Fang, Qi Song, et al. Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct. *Radiology*, page 200905, 2020.

[10] Ming-Yen Ng, Elaine YP Lee, Jin Yang, Fangfang Yang, Xia Li, Hongxia Wang, Macy Mei-sze Lui, Christine Shing-Yen Lo, Barry Leung, Pek-Lan Khong, et al. Imaging profile of the covid-19 infection: radiologic findings and literature review. *Radiology: Cardiothoracic Imaging*, 2(1):e200034, 2020.

[11] Feng Pan, Tianhe Ye, Peng Sun, Shan Gui, Bo Liang, Lingli Li, Dandan Zheng, Jiazheng Wang, Richard L Hesketh, Lian Yang, et al. Time course of lung changes on chest ct during recovery from 2019 novel coronavirus (covid-19) pneumonia. *Radiology*, page 200370, 2020.

[12] Zheng Ye, Yun Zhang, Yi Wang, Zixiang Huang, and Bin Song. Chest ct manifestations of new coronavirus disease 2019 (covid-19): a pictorial review. *European radiology*, pages 1–9, 2020.

[13] Feng Shi, Jun Wang, Jun Shi, Ziyan Wu, Qian Wang, Zhenyu Tang, Kelei He, Yinghuan Shi, and Dinggang Shen. Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for covid-19. *IEEE Reviews in Biomedical Engineering*, 2020.

[14] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

[15] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017.

[16] Toan Duc Bui, Jae-Joon Lee, and Jitae Shin. Incorporated region detection and classification using deep convolutional networks for bone age assessment. *Artificial intelligence in medicine*, 97:1–8, 2019.

[17] Qinhua Hu, Luís Fabrício de F Souza, Gabriel Bandeira Holanda, Shara SA Alves, Francisco Hércules dos S Silva, Tao Han, and Pedro P Rebouças Filho. An effective approach for ct lung segmentation using mask region-based convolutional neural networks. *Artificial Intelligence in Medicine*, page 101792, 2020.

[18] Gabriele Piantadosi, Mario Sansone, Roberta Fusco, and Carlo Sansone. Multi-planar 3d breast segmentation in mri via deep convolutional neural networks. *Artificial Intelligence in Medicine*, 103:101781, 2020.

[19] Weiyi Xie, Colin Jacobs, Jean-Paul Charbonnier, and Bram van Ginneken. Contextual two-stage u-nets for robust pulmonary lobe segmentation in ct scans of covid-19 and copd patients. *arXiv preprint arXiv:2004.07443*, 2020.

[20] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in neural information processing systems*, pages 766–774, 2014.

[21] Agnieszka Mikołajczyk and Michał Grochowski. Data augmentation for improving deep learning in image classification problem. In *2018 international interdisciplinary PhD workshop (IIPhDW)*, pages 117–122. IEEE, 2018.

[22] John McManigle, Raquel Bartz, and Lawrence Carin. Y-net for chest x-ray preprocessing: Simultaneous classification of geometry and segmentation of annotations. *arXiv preprint arXiv:2005.03824*, 2020.

[23] Eduardo Castro, Jaime S Cardoso, and Jose Costa Pereira. Elastic deformations for data augmentation in breast cancer mass detection. In *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pages 230–234. IEEE, 2018.

[24] Xiang Li, Shuo Chen, Xiaolin Hu, and Jian Yang. Understanding the disharmony between dropout and batch normalization by variance shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2682–2690, 2019.

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.