Orientation Regression in Hand Radiographs: A Transfer Learning Approach

Ivo M. Baltruschat[a],[c], Axel Saalbach[b], Mattias P. Heinrich[a], Hannes Nickisch[b], and Sascha Jockel[c]

[a]Institute of Medical Informatics, Universität zu Lübeck, Lübeck, Germany;
[b]Philips GmbH Innovative Technologies, Hamburg, Germany;
[c]Philips Medical Systems DMC GmbH, Hamburg, Germany;

ABSTRACT

Most radiologists prefer an upright orientation of the anatomy in a digital X-ray image for consistency and quality reasons. In almost half of the clinical cases, the anatomy is not upright orientated, which is why the images must be digitally rotated by radiographers. Earlier work has shown that automated orientation detection results in small error rates, but requires specially designed algorithms for individual anatomies. In this work, we propose a novel approach to overcome time-consuming feature engineering by means of Residual Neural Networks (ResNet), which extract generic low-level and high-level features, and provide promising solutions for medical imaging. Our method uses the learned representations to estimate the orientation via linear regression, and can be further improved by fine-tuning selected ResNet layers. The method was evaluated on 926 hand X-ray images and achieves a state-of-the-art mean absolute error of 2.79°.

Keywords: deep learning, transfer learning, orientation regression, X-ray, residual neural network

1. INTRODUCTION

A major challenge in medical imaging is workflow optimization as hospitals are keen to improve their patient throughput at a radiography system. In the radiography workflow of most examinations, a common task is to manually rotate digital X-ray images to a preferred orientation suitable for diagnostic reading. This reduces the number of patients that can be analyzed in a given time frame. An automatic alignment system for clinical X-ray images can help to ensure the correct orientation, but is still an open problem. We identify three major challenges: First, in clinical routine we observed up to 23 different examinations, consisting of 13 anatomies (e.g. hand, chest, ...) and 4 projection types (e.g. posterior-anterior (PA), anterior-posterior (AP), lateral, and oblique), makes this task very diverse. Second, computation time must be as small as possible and third, a high alignment accuracy is needed.

A lot of work has been done for orientation detection in X-ray imaging, in particular for chest examinations[1234] as they are most common examination. However, most of these methods allow only for the identification of a limited set of orientations. Furthermore, they are often specifically designed for specific examinations. Luo and Luo[5] presented a generic framework for orientation detection which supports multiple examination types. They tested the method on a large dataset of 12000 radiographies and reported a success rate of 96.1%. In contrast to our approach, they only perform a classification in four regions (e.g. 0°, 90°, 180°, 270°) and not a precise orientation prediction. An in-house study conducted at Philips Healthcare investigated specifically designed methods for orientation classification in chest and hand AP/PA images. The first method employed hand-crafted features, while the later used the Generalised Hough transform[6] in combination with a Canny edge detection. The methods achieved a good accuracy, but both approaches are not generic and thus cannot easily extended to other examination types. The development of such hand-crafted approaches has to be repeated for every examination type, which is a difficult and time intensive task. In addition, the method for hand AP/PA suffered from long computation time of 202ms. In early work, we demonstrated that a simple geometrical method with hand-crafted features for hand AP/PA and oblique examinations can reduce the computation time down to 2ms but at the cost of alignment accuracy.[7] Anatomy orientations were classified in four categories, with a subsequent alignment in 90° steps. We tested this feature-engineered method twice. First, only on AP/PA and oblique examination, which resulted in a mean accuracy of 96.73%. Secondly, we also added lateral examinations

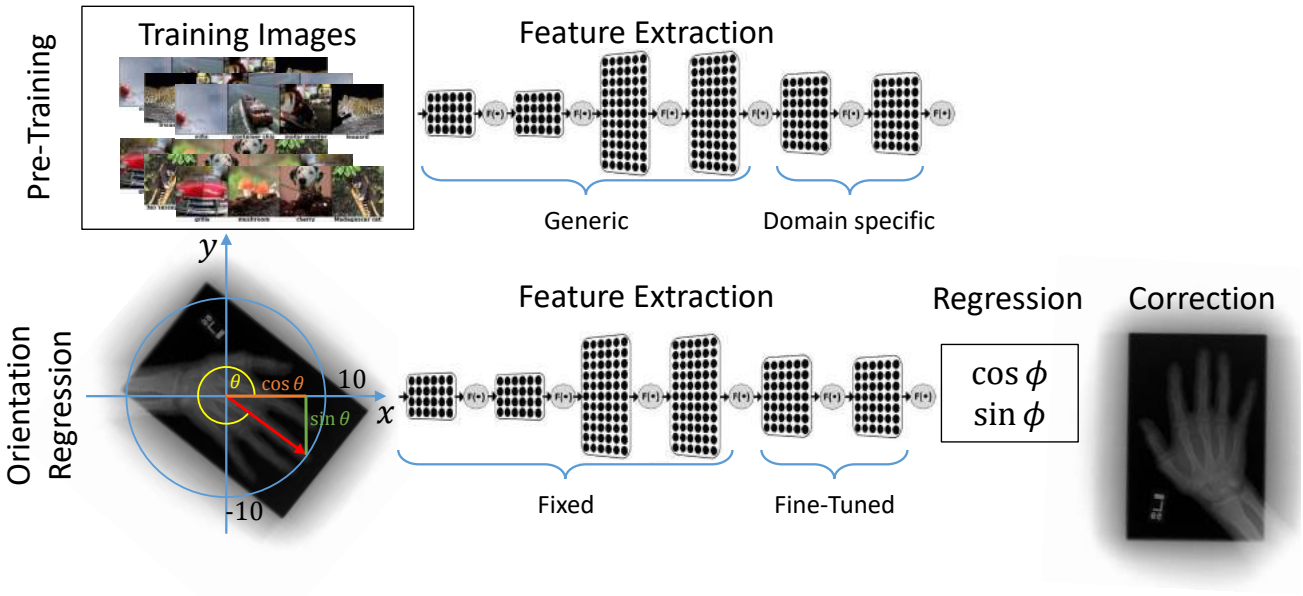


Figure 1: Top row: A neural network is trained on a large dataset of natural images (ImageNet). The model learns low- and high-level features. Bottom row: The model trained until convergence for natural image classification, is adapted for an orientation regression in medical image domain and a linear transformation is used to correct the image orientation in a final step. Early layers of the ImageNet model are directly transferred with same weights, while later layers are fine-tuned for our medical imaging application.

to the test set. Thus, we evaluated on the same data as in our later experiments. This reduced the mean accuracy to 92.14%.

Our goal is to present a generic method for orientation estimation in clinical X-ray images while retaining a high alignment accuracy and low computation time. To deal with these challenges we propose to employ deep Convolutional Neural Networks (CNNs) together with transfer learning.[8,9] CNNs are a generic method for machine learning and usually need large amounts of annotated image data for training. Recent work showed that transfer learning can be used for medical applications[9,10,11,12] without the need of a large training dataset. In this contribution, we present a novel approach for orientation regression and automatic alignment of X-ray images. We use a ResNet-50[13,14] trained on more than 1 million images with 1000 class labels on the ImageNet dataset, popular in computer vision,[15] as generic high-level feature extractor for an orientation regression in the medical domain. When applied to unseen data such as X-ray images, our approach automatically estimates a real-valued orientation and can therefore align examinations into any preferred orientation. An overview of our method is given in Figure 1. Importantly, using this approach, our model can be trained even for examination types where only a small labeled dataset exists.

Open-source datasets are rare in the medical domain compared to computer vision. As no dataset met our requirements nor even existed for training and testing deep neural networks regarding our objective, we created a dedicated dataset for orientation detection. We restricted the dataset to three examination types and labeled 424 hand AP/PA, 397 hand oblique, and 105 hand lateral images. Therefore, our orientation dataset contains 926 images in total. The distribution of the original image orientations in our dataset is given in Figure 2. In our dataset images are 16-bit gray scale and have a varying pixel size of 3001×3001, 3000×2372, or 2846×2330 depending on the size of the X-ray detector used during acquisition. We labeled each image with a precise orientation, which was measured between x-axis and main orientation line. Figure 3 illustrates our labelling strategy on some examples from our dataset. All X-ray images in this work are presented and learned without any post-processing. Hence, the shutter area is not shown in black and the image contrast is not enhanced. We applied our automatic alignment method to the above data and show significantly improved accuracy compared

to hand-crafted feature methods and off-the-shelf feature extractor in Section 3.

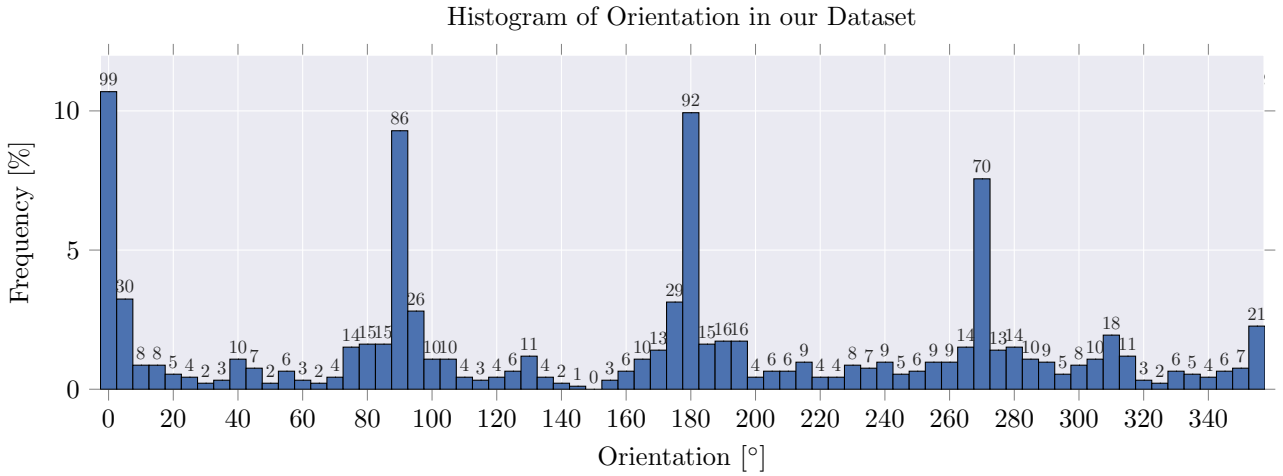
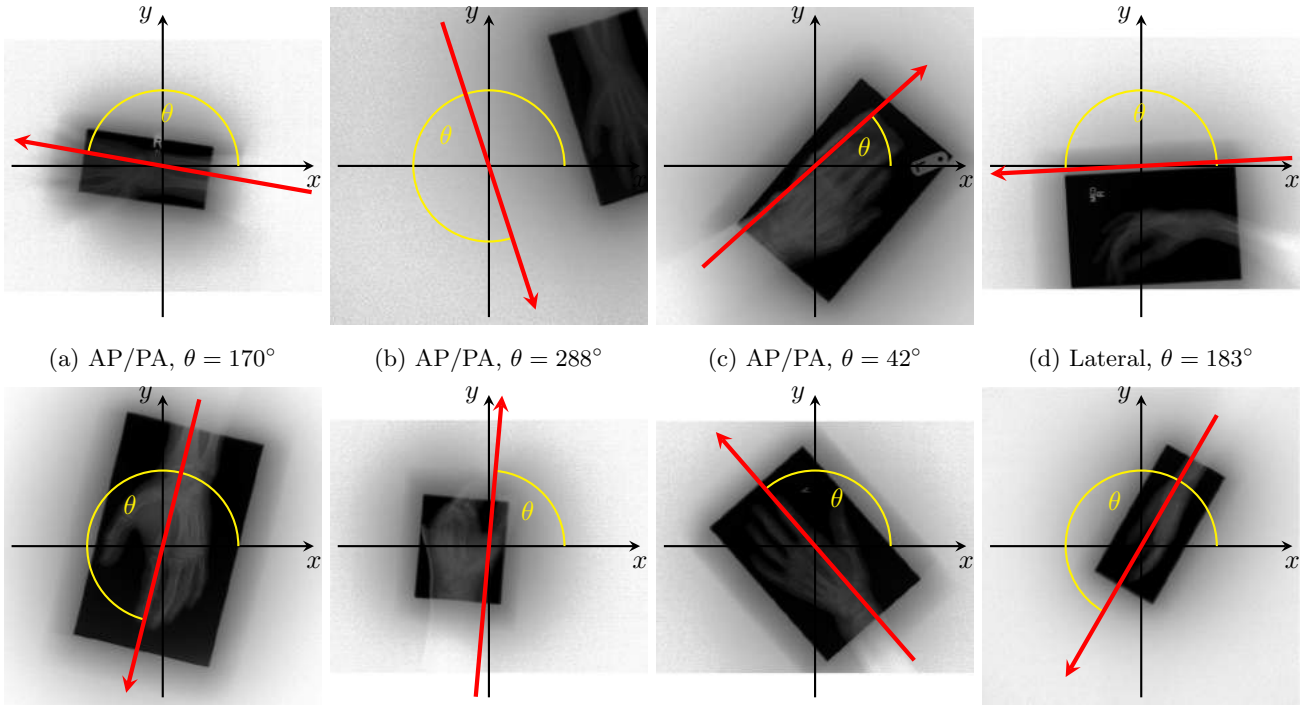


Figure 2: We labeled 926 hand X-ray images with a precise angle for anatomy orientation. The histogram is plotted over these labeled orientations. Each bin covers a width of five degree orientation angle. The values of the y-axis are normalized by the total number of samples and the x-axis is in degrees. In addition, above each bin its total count in our dataset is plotted.



(a) AP/PA, $\theta = 170°$ (b) AP/PA, $\theta = 288°$ (c) AP/PA, $\theta = 42°$ (d) Lateral, $\theta = 183°$

(e) Oblique, $\theta = 256°$ (f) Oblique, $\theta = 85°$ (g) Oblique, $\theta = 131°$ (h) Lateral, $\theta = 240°$

Figure 3: Eight examples illustrating a large diversity of appearance in hand X-rays. Images are shown without any post-processing. The X-ray detector size my vary between examples and hence image size is also different (e.g. (a) has a smaller height than (b)). In addition, our labeling strategy is shown by a red arrow as main orientation line and a measured angle θ to x-axis. The coordinate system's origin is in the center of the X-ray detector for each example. Beneath each image, its projection type and annotated angle θ is plotted.

2. METHOD

Deep CNNs achieve state-of-the-art results in a variety of image processing tasks. These CNNs are trained for a task in a specific image domain. In a CNN, earlier layers extract generic low-level features and later layers extract domain and task specific features as illustrated in Figure 1. Transferring a pre-trained neural network to a new task in a different image domain is possible, if either the domains or tasks share similarities.[9,10] In a transfer learning setup, parts of the network are replaced or retrained. If only the last layer is adapted, and the original network is used primarily as a feature extractor, this is often considered as an off-the-shelf approach. Using fine-tuning, additional (i.e. deep layer) of the network are retrained as well, based on the new data.

For the automatic alignment of the input images F to a preferred orientation $\alpha \in [0°, 360°)$, we use a two-step approach. First, we employ the adapted ResNet-50[13,14] (see Table 1) for orientation regression, where an angle $\phi_i \in [0°, 360°)$ is predicted for each test sample F_i. Second, we calculate the aligned image F_i' by a linear transformation $T_{\phi_i}^{rot} : \mathbb{R}^2 \to \mathbb{R}^2$ with

$$T_{\phi_i}^{rot} = \begin{pmatrix} \cos(\phi_i - \alpha) & -\sin(\phi_i - \alpha) \\ \sin(\phi_i - \alpha) & \cos(\phi_i - \alpha) \end{pmatrix} \tag{1}$$

and $F_i' = T_{\phi_i}^{rot}(F_i)$. Figure 1 illustrates an overview of our automatic alignment method at the bottom row.

Orientation Regression: For learning purposes, ground truth information (i.e. the angle) was estimated manually for a representative set of images, resulting in N training samples $\{F_j, \theta_j\}, j \in N$. During training, we use the mean squared error (MSE) as a loss function to compare the ground truth angle θ with the estimate ϕ. The MSE puts more emphasis on large errors but cannot handle a circular angle representation as it assigns a large penalty to the pair $(2°, 359°)$ even though they are very similar. We hence represent angles θ in Cartesian coordinates in the complex plane via the embedding $z(\theta) = (\cos(\theta), \sin(\theta))$ resulting in the empirical average:

$$E = \frac{1}{N} \sum_{j=1}^{N} \|z(\theta_j) - z(\phi_j)\|_2^2 = \frac{1}{N} \sum_{j=1}^{N} [\cos(\theta_j) - \cos(\phi_j)]^2 + [\sin(\theta_j) - \sin(\phi_j)]^2 \tag{2}$$

which we use as a training criterion. The classifier part is the last fully-connected layer in a ResNet. For the prediction of the orientation, we replaced the last layer by a new fully-connected layer with two outputs for a regression of the 2-D Cartesian coordinates $z(\theta)$. Table 1 shows our new network architecture compared to the original network.

3. EXPERIMENTS

We evaluated both transfer learning methods on our dataset and compared their results. Due to the limited size of our dataset, we employed a five times random sub-sampling and split the dataset into 70% training and 30% testing data in order to assess the performance of the method.[16] In our fine-tuning experiment, we retrained the conv5-layers as shown in Table 1.

3.1 Preprocessing and data augmentation

ResNet expects a $224 \times 224 \times 3$ dimensional input image of 8-bit color depth. We thus performed some pre-processing steps to meet the network's input requirement. First, a linear intensity transformation from 16-bit to 8-bit color depth is applied. Secondly, each image is resized to 256×256 pixels and converted into a 3-channel image by copying the same information in all channels. Hence, we did not preserve image aspect ratios. In this work, geometric transformations are applied as data augmentation methods. Each image is rotated by 90, 180, and 270 and then flipped horizontally. Furthermore, a random cropping from $256 \times 256 \times 3$ to $224 \times 224 \times 3$ pixels was employed.

Table 1: In our experiments, we used the ResNet-50. This table shows differences between the original architecture and ours (off-the-shelf and fine-tuned ResNet-50). If there is no difference to the original network, the word "same" is written in the table. The violet text emphasizes, which parts of the network are changed for our application. The conv3_1, conv4_1, and conv5_1 layers perform a down-sampling of the spatial size with a stride of 2.

Layer name	**Output size**	**Original 50-layer**	**Off-the-shelf 50-layer**	**Fine-tuned 50-layer**
conv1	112×112	7×7, 64-d, stride 2	same	same
pooling1	56×56	3×3, 64-d, max pool, stride 2	same	same
conv2_x	56×56	$\begin{bmatrix} 1 \times 1, & 64\text{-d} \\ 3 \times 3, & 64\text{-d} \\ 1 \times 1, & 256\text{-d} \end{bmatrix} \times 3$	same	same
conv3_x	28×28	$\begin{bmatrix} 1 \times 1, & 128\text{-d} \\ 3 \times 3, & 128\text{-d} \\ 1 \times 1, & 512\text{-d} \end{bmatrix} \times 4$	same	same
conv4_x	14×14	$\begin{bmatrix} 1 \times 1, & 256\text{-d} \\ 3 \times 3, & 256\text{-d} \\ 1 \times 1, & 1024\text{-d} \end{bmatrix} \times 6$	same	same
conv5_x	7×7	$\begin{bmatrix} 1 \times 1, & 512\text{-d} \\ 3 \times 3, & 512\text{-d} \\ 1 \times 1, & 2048\text{-d} \end{bmatrix} \times 3$	same	**fine-tuned**
pooling2	1×1	7×7, 2048-d, average pool, stride 1	same	same
fc	1×1	1000-d, fully-connected	**2-d, fully-connected**	
loss	1×1	1000-d, softmax	**2-d, MSE**	

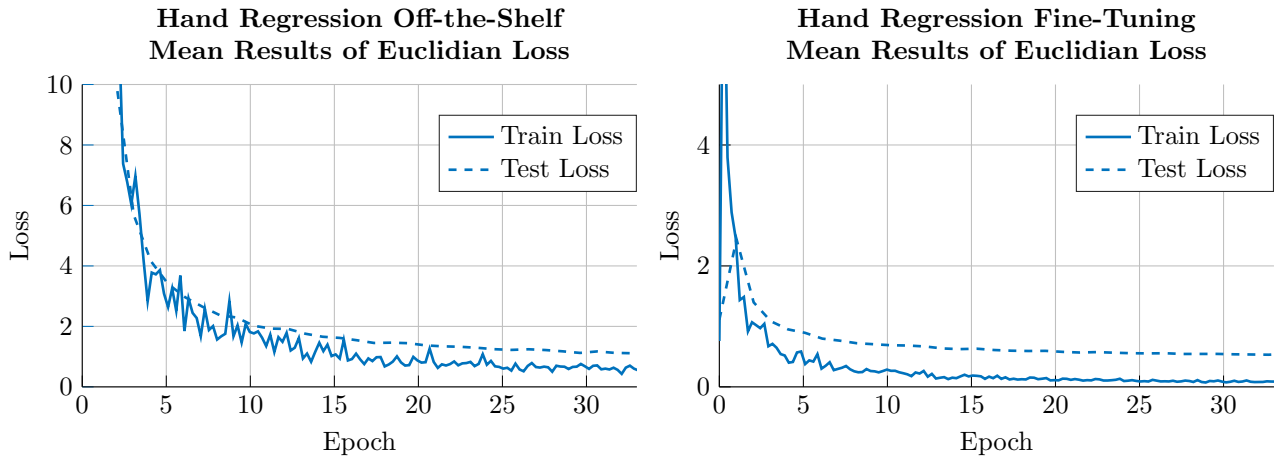
Training Parameters: We followed mostly the training of the original ResNet-50. In the training process, our data is zero-centered by subtracting the mean. We initialize the weights with the converged ResNet-50 of He et al.[13] and trained the last fully-connected layer from scratch after random initialization. As optimization method, we employed Adaptive Moment Estimation (Adam) with a mini-batch size of 256. The parameters of Adam were set to $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We reduced the learning rate η by a factor of 10 compared to the original training and set it to $\eta = 0.01$. After each epoch, the training data is shuffled.

4. RESULTS

The runtime of our algorithm is a crucial aspect for the application and therefore we measured the runtime of 50 applications of the network to an image. The test is performed on the Intel Xeon CPU E3-1246 v3 with four 3.5Ghz cores, eight threads and a single image. Our tests resulted in an average runtime of 176.30ms.

Off-the-shelf: Figure 4a illustrates the training process. It can be seen that the train and test loss decreased over time, while after epoch 18 only minor improvements could be observed. The train loss fluctuated around 0.51 after epoch 26 and the test lost saturated at 1.17 after epoch 30. Because of this, the training was stopped at epoch 32. Since the test loss saturated after epoch 30 on average, the corresponding networks were used for further investigation and they produced the following results. The off-the-shelf ResNet-50 demonstrates impressive initial results for a precise angle prediction. Figure 5a shows that the histogram of deviations is almost centered around zero, but the standard deviation of 9.47° is high. This is caused by 0.66% outliers in the range $[-180, -30)$ and $(30, 180]$, which heavily influence the standard deviation. Nonetheless, 64.1% of the samples are within $[-4°, 4°]$ and the mean absolute error (MAE) is only 4.25°.

Fine-tuning: Figure 4b presents the results of the fine-tuning process. The test loss at epoch 0 is at 1.2, as the final off-the-shelf ResNet-50 was used as initialization. After an initial increase of train and test loss, both losses decreased again. The test loss saturated around 0.56 after epoch 21, whereas the train loss decreased continually until the training was stopped at epoch 32. It reached a minimum loss of 0.07. As a result, we concluded that

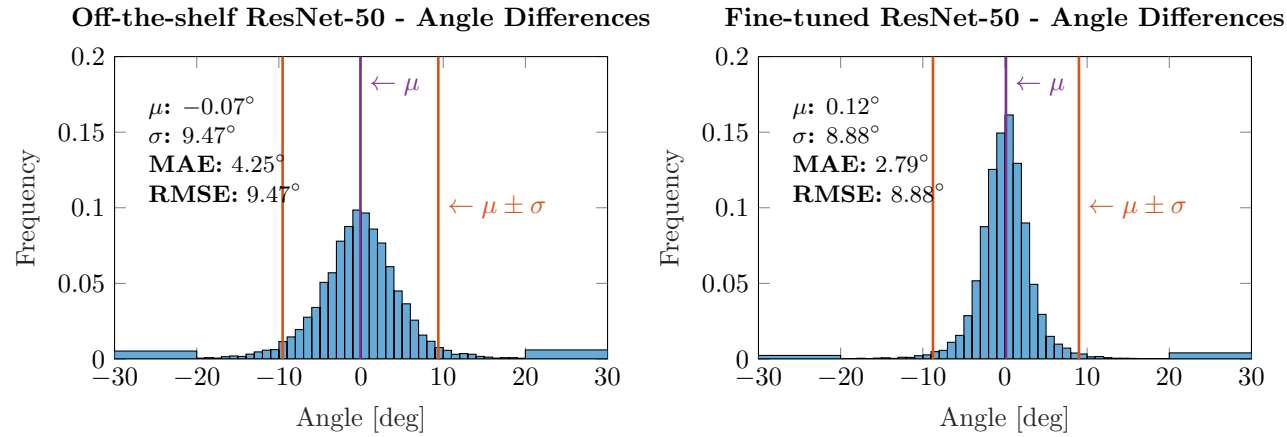


(a) (b)

Figure 4: We trained five off-the-shelf and five fine-tuned ResNet-50. Each of the five with a different re-sample split, but same split between off-the-shelf and fine-tuned. In scatter plot (a) and (b), averaged training and testing curves with Euclidean loss (y-axis) against training epoch (x-axis) are shown. (a) shows training from the off-the-shelf experiment and achieved the best test loss at epoch 30. (b) shows training of our fine-tuning experiment and achieved the best test loss at epoch 21.

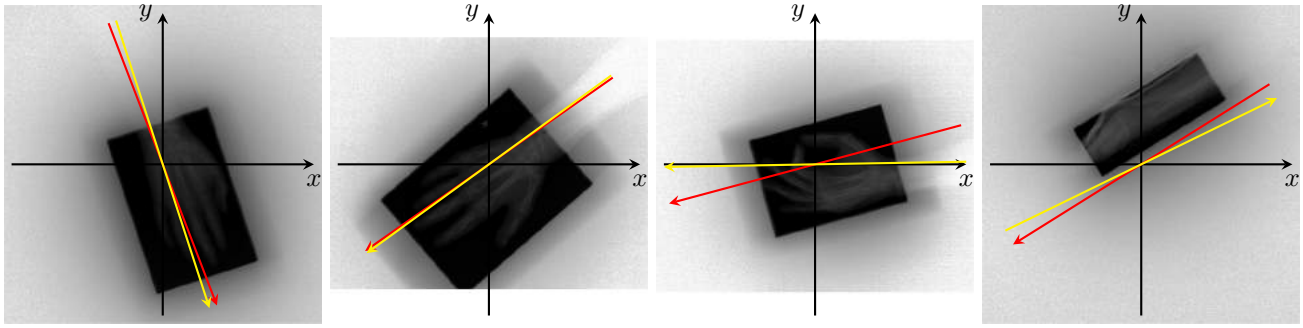
the trained network at epoch 21 merits further investigation. The results show that fine-tuning the regression ResNet-50 improved the outcome in terms of precision of angle prediction. In Figure 5b, the standard deviation of the distribution is 0.59° smaller than that of the off-the-shelf ResNet-50. In addition, 84.4% of the samples are in the interval $[-4°, 4°]$ and the MAE improved to 2.79°. Comparing this to the off-the-shelf version the MAE decreased by 1.46 percentage points.

Previous methods only classify orientation in four classes. Hence, we binned our results to four classes and calculated a mean accuracy of $98.04 \pm 0.44\%$.

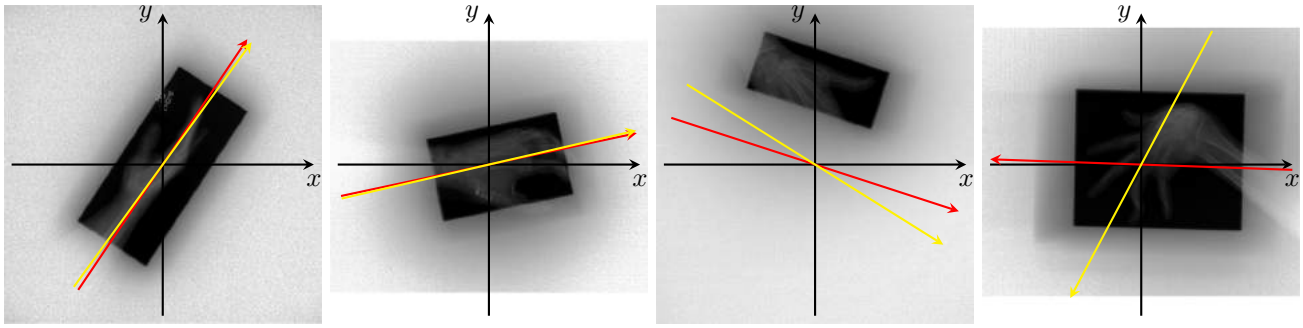


(a) (b)

Figure 5: As evaluation metric, we calculated the difference e between predicted angle and our labeled angle for each test sample. Panel (a) shows the resulting histogram from the off-the-shelf experiment and panel (b) displays the resulting histogram of our fine-tuning experiment. e is on the x-axis in degrees and the frequency of each bin is on the y-axis. In both histograms, σ is heavily influenced by outliers in the range $[-180°, -30°)$ and $(30°, 180°)$. The bin size for the interval $[-20°, 20°]$ is one degree, whereas the intervals $[-180°, -20°)$ and $(20°, 180°]$ are reduced to a single bin for visualization purposes.



(a) $\|e\| = 3°$ (b) $\|e\| = 1°$ (c) $\|e\| = 14°$ (d) $\|e\| = 186°$



(e) $\|e\| = 2°$ (f) $\|e\| = 1°$ (g) $\|e\| = 14°$ (h) $\|e\| = 64°$

Figure 6: Eight results calculated on samples from our test set. The images size varies because different X-ray detectors were used. Center point of the coordinate system lies in the center point of the detector. The red arrow shows our labeled orientation and the yellow arrow displays the predicted orientation. Image orientations in the fist two columns are correctly predicted with respect to a small error $\|e\| = \|\theta - \phi\| \in [0°, 3°]$. In the third column, two examples with a medium error $\|e\| \in (3°, 45°]$ are shown. The last column presents examples with a large error $\|e\| \in (45°, 180°]$.

5. CONCLUSION

Our work demonstrates that a pre-trained ResNet-50 has remarkable potential for orientation regression of anatomical structures in clinical X-ray images. The learned representations of the ResNet are an excellent starting point for an orientation regression. Our empirical results show that a fine-tuned regression ResNet-50 has a higher mean accuracy of $98.04 \pm 0.44\%$ than the previous state-of-the-art of 92.14% from Baltruschat et al. (2016) when applied to the same dataset. We tested the method a second time only on AP/PA and oblique examinations. This resulted in a mean accuracy of 96.73% which is still 1.31 percentage points lower than our best result. In addition, the development time was much lower and this novel approach can be applied to several types of examinations. Comparing the e of our method to others, our method has a high runtime. This can be easily reduced by a magnitude, if GPUs were used. Future work will include the investigation of additional examination types (e.g. chest and foot) and a comparison of different training techniques (e.g. fine-tuning vs. auto-encoders).

ACKNOWLEDGMENTS

The work has been carried out at Philips Medical Systems DMC GmbH, Hamburg, Germany.

REFERENCES

[1] Boone, J. M., Seshagiri, S., and Steiner, R. M., "Recognition of chest radiograph orientation for picture archiving and communications systems display using neural networks," *Journal of Digital Imaging* **5**(3), 190–193 (1992).

[2] Pietka, E. and Huang, H., "Orientation correction for chest images," *Journal of Digital Imaging* **5**(3), 185–189 (1992).

[3] Luo, H., Hao, W., Foos, D. H., and Cornelius, C. W., "Automatic image hanging protocol for chest radiographs in pacs," *IEEE Transactions on Information Technology in Biomedicine* **10**(2), 302–311 (2006).

[4] Nose, H., Unno, Y., Koike, M., and Shiraishi, J., "A simple method for identifying image orientation of chest radiographs by use of the center of gravity of the image," *Radiological physics and technology* , 1–6 (2012).

[5] Luo, H. and Luo, J., "Robust online orientation correction for radiographs in pacs environments," *IEEE transactions on medical imaging* **25**(10), 1370–1379 (2006).

[6] Ballard, D., "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition* **13**(2), 111 – 122 (1981).

[7] Baltruschat, I. M., Hensel, M., and Heinrich, M., "Automatic orientation detection of hand structures in digital X-ray images," *Student Conference 2016, Medical Engineering Science and Medical Informatics* , 203–206 (2016).

[8] Yosinski, J., Clune, J., Bengio, Y., and Lipson, H., "How transferable are features in deep neural networks?," in [*NIPS*], (2014).

[9] Menkovski, V., Aleksovski, Z., Saalbach, A., and Nickisch, H., "Can pretrained neural networks detect anatomy?," *arXiv preprint arXiv:1512.05986* **abs/1512.05986** (2015).

[10] Bar, Y., Diamant, I., Wolf, L., Lieberman, S., Konen, E., and Greenspan, H., "Chest pathology detection using deep learning with non-medical training," *IEEE* (2015).

[11] Shin, H.-C., Roth, H., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D. J., and Summers, R. M., "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging* **35 5**, 1285–98 (2016).

[12] van Ginneken, B., Setio, A. A. A., Jacobs, C., and Ciompi, F., "Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans," in [*Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on*], 286–289, IEEE (2015).

[13] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* , 770–778 (2016).

[14] He, K., Zhang, X., Ren, S., and Sun, J., "Identity mappings in deep residual networks," in [*ECCV*], (2016).

[15] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L., "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)* **115**(3), 211–252 (2015).

[16] Molinaro, A. M., Simon, R., and Pfeiffer, R. M., "Prediction error estimation: A comparison of resampling methods," *Bioinformatics* **21**(15), 3301–3307 (2005).