Graphical User Interface for Medical Deep Learning - Application to Magnetic Resonance Imaging

Sebastian Milde^{*}, Annika Liebgott^{*†}, Ziwei Wu^{*}, Wenyi Feng^{*}, Jiahuan Yang^{*}, Lukas Mauch^{*}, Petros Martirosian[‡], Fabian Bamberg[†], Konstantin Nikolaou[†], Sergios Gatidis[†], Fritz Schick[‡], Bin Yang^{*} and Thomas Küstner^{*‡§}

* Institute of Signal Processing and System Theory, University of Stuttgart, Germany
[†] Department of Diagnostic and Interventional Radiology, University Hospital of Tübingen, Germany

Department of Diagnostic and interventional Radiology, University Hospital of Tubingen, Germany

[‡] Section on Experimental Radiology, University Hospital of Tübingen, Germany

§ School of Biomedical Engineering & Imaging Sciences, King's College London, St. Thomas' Hospital, UK

Abstract-In clinical diagnostic, magnetic resonance imaging (MRI) is a valuable and versatile tool. The acquisition process is, however, susceptible to image distortions (artifacts) which may lead to degradation of image quality. Automated and referencefree localization and quantification of artifacts by employing convolutional neural networks (CNNs) is a promising way for early detection of artifacts. Training relies on high amount of expert labeled data which is a time-demanding process. Previous studies were based on global labels, i.e. a whole volume was automatically labeled as artifact-free or artifact-affected. However, artifact appearance is rather localized. We propose a local labeling which is conducted via a graphical user interface (GUI). Moreover, the GUI provides easy handling of data viewing, preprocessing (labeling, patching, data augmentation), network parametrization and training, data and network evaluation as well as deep visualization of the learned network content. The GUI is not limited to these features and will be extended in the future. The developed GUI is made publicly available and features a modular outline to target different applications of machine learning and deep learning, such as artifact detection, classification and segmentation.

I. INTRODUCTION

In modern medicine, magnetic resonance imaging (MRI) is a versatile tool in clinical diagnostic. Its capability of assessing processes within the human body and anatomical structures in a non-invasive manner makes it a valuable technique. However, due to its long acquisition time, MRI is prone to a manifold of artifacts which can degrade image quality significantly. Fig. 1 shows images which are distorted by patient-induced motion in comparison to a desired motion-free image. Motion artifacts are a common source for image degradation, since the long acquisition time can cause patients to move instead of



Fig. 1: Artifact-free and artifact-affected images in the head and abdomen. Arrows indicate the motion artifacts.

lying perfectly still (rigid motion) and necessary physiological functions such as respiration or heartbeat result in unavoidable movements (non-rigid motion), which may manifest as motion artifacts in the image. Depending on their cause, the appearances of patient-induced motion artifacts differ. Rigid movement (e.g. head movement) results in Nyquist ghosting, i.e. additional manifestations of the original image which are slightly rotated or shifted. Non-rigid movement (e.g. respiration) appears as blurring in the affected image regions because internal structures change their location, shape and size. For a precise diagnostic, a good image quality of the acquired data is crucial. To determine if the quality is sufficient, it is often necessary to have a human expert assessing the acquired images. Such a time-demanding and expensive manual inspection is not feasible in cases of retrospective analysis of high data quantities, e.g. for large epidemiological cohort studies like the German National Cohort [1] or UK Biobank [2]. Other possible problems concern patient comfort and clinical throughput. In some cases the quality inspection by an MR specialist is performed after the patient has already left and it may be necessary to schedule an additional examination.

For an automated assessment of MR image quality, a reference-free and spatially resolved detection and quantification of artifacts would be desired. That way, a prospective quality assurance or a retrospective quality control can be performed. The obtained quality information respectively artifact appearance can enable mechanism like prospective [3]–[5] or retrospective [6]–[9] correction of patient-inducted motion artifacts. Moreover, an automated adaptation of acquisition parameters with a directly repeated scan of insufficient quality can improve data fidelity.

We thus proposed an approach based on convolutional neural networks (CNN) which were trained to learn the appearances of artifacts in whole-body MR images. The first step consisted of successfully training 2D CNNs to detect motion in 2D image patches [10], which was subsequently extended to the 3D case to cope with through-plane motion [9], [11]–[14], resulting in a substantial increase in classification accuracy. Furthermore, we recently proposed a first approach on training a CNN to detect multiple artifacts simultaneously [15].

Although good results were achieved in our previous studies, the detection was limited by the provided global labeling. In global labeling, the entire volume is labeled as artifact-free or artifact-affected. Since artifact appearance is localized and to ensure a high quality of the labels used for training, it is necessary that an experienced radiologist inspects the images for artifacts and provides local labels for the appearance of artifacts. In this work, we will examine the benefits of a local labeling over a global labeling for the task of MR artifact detection.

Moreover, the emerge of machine learning and deep learning applications in the medical area rise the need for an easy processing setup: from data viewing over network training towards analysis of the obtained results. We therefore propose in this work an easy-to-use graphical user interface (GUI) which provides the following functionalities: data viewing (DI-COM, JPEG, PNG, TIFF), preprocessing (labeling, patching, data augmentation, data splitting), network training, hyperparameter setting, test data evaluation (accuracy plots, confusion matrix, probability map), network visualization (feature maps, deep visualization). Several applications can be supported by this GUI such as classification, regression, reconstruction, correction and/or segmentation. We chose to implement the GUI in PyQt, so that it can be used platform independent with any Python-based deep learning implementation (keras [16], Tensorflow [17], PyTorch [18], Caffe [19]). In our work we show its usage with keras. In this work we present the modular structure of the GUI and its core functionalities. We will highlight its usage for three applications: MR motion artifact detection with derived local labels in comparison to a global labeling, MR motion correction via variational autoencoder (VAE) and semantic segmentation of organs in whole-body MRI.

II. GUI

The GUI is built up in PyQt as a model-view control architecture and is supported for both Python 2.7 and Python>3.5. The currently provided modules are *Data Viewing*, *Network Training*, *Network Visualization* and *Prediction Viewing* which can be accessed via tabs. Due to the modular structure of the user interface, it can be extended by further modules as required. Furthermore, the GUI can also be accessed via web browser (hosted on a Python server) making it independent of an installed Python environment which eases operation and access. The structure of the GUI with its modules is shown in Figure 2.

A. Data Viewing

The *Data Viewing* module includes all functions to display images (DICOM, JPEG, PNG, TIFF, ...) in a suitable layout. Within this module, the GUI ensures a flexible display of images on a grid layout in both 2D mode (Fig. 3 (a)) and 3D mode (Fig. 3 (b)). In 2D mode, different images can be loaded, next to each other. In 3D mode, the various views (sagittal, coronal, axial) of a 3D image are displayed. The user has several image operations such as zooming, brightness or contrast adjustment as well as scrolling through the different slices. They can be invoked individually on each image or coupled on all loaded images. For labeling, several drawing



Fig. 2: Structure of Graphical User Interface (GUI) with its modules designed for deep learning of medical data.

tools are available to the user: rectangle-tool, ellipse-tool, lasso-tool to mark regions of interest (ROIs) and volumes of interest (VOIs). In addition the user can assign a class label to these marked regions/volumes. The set markers are stored in a database immediately after creation. They can be reloaded and modified later via the GUI. The colouring of the label can be individually changed. For each mark, the information is stored in a JSON file (data format for the purpose of data exchange between applications), making it available in the database at any time for data preprocessing.

B. Network Training

For training and network validation, the user can set parameters for data preprocessing, data splitting, and training using the GUI's *Network Training* tab (Fig. 3 (c)).

For data preprocessing, the gray values of an image can be normalized either into a range of $[I_{min}, I_{max}]$ or using zeromean and standard deviation. Afterwards, the image can be patched into sizes $[p_x \times p_y \times p_z]$ ([left-right × anterior-posterior × superior-inferior]) with an overlap $[o_x, o_y, o_z]$. Since some images can not be completely subdivided into patches, they must be enlarged by zero-padding. The zero-padded image then has the size $[n_x \times n_y \times n_z]$. The number of patches per image can be calculated as

$$N_{\text{patches}} = \left(\frac{n_x - p_x}{o_x} + 1\right) \cdot \left(\frac{n_y - p_y}{o_y} + 1\right) \cdot \left(\frac{n_z - p_z}{o_z} + 1\right)$$

For data augmentation, the user has the choice between mirroring along all dimensions, translation by a specified vector of displacements, rotation by a vector of angles, rescaling to a vector of desired resolutions, zero component analysis whitening for decorrelating patches, histogram equalization for intensity variation and contrast stretching for contrast adaptations.

For a supervised training, the images respectively patches are assigned with their respective labels from the database. Training any previously defined architecture is done by handing over a *.py* file to the GUI which has a specified interface. This file is then executed and monitored from the GUI. This approach makes the GUI independent of the desired machine/deep learning implementation. In addition, the user can select certain parameters such as batch size, optimizer, learning rate or number of epochs. Moreover, the user can select the data splitting into training, validation and test set: random splitting with a split percentage or k-fold crossvalidation with leave-one-subject-out approach. During the training process, curves of the accuracy and loss function are displayed in the GUI and are updated after each epoch. This enables fast performance evaluations and validity checks.

C. Network Visualisation

From the provided *.py* file, the structure (convolutional layers, activation functions, connections, ...) of the CNN architecture can be visualized in the GUI (Fig. 3 (d)). Moreover, after training the user can also examine the learned network content in each layer by visualizing the trained filter weights and obtained feature maps for 2D and 3D.

If a test database or a test images is selected, a humaninterpretable visualization of the network content is retrieved via a deep visualization [10]. A most-likely input patch for a given output class is generated and convolved with the significant and sparse attractor points in the test image/database. This yields an interpretable significance scale to which areas the network is most attracted.



(a) Data Viewing in 2D mode

(b) Data Viewing in 3D mode



(c) Network Training

(d) Network Visualization

Fig. 3: The Graphical User Interface: Operating tabs of Data Viewing in (a) 2D mode and (b) 3D mode, (c) Network Training and (d) Network Visualization. The user can easily switch between the tabs and the 2D/3D viewing modes. In Network Training, parameters for data preprocessing and training can be set. In the Network visualization tab, the user can examine the inner structure of the network and the obtained feature maps as well as a deep visualization.

D. Prediction Analysis

The training and prediction results of the CNN architectures can be assessed using retrospective analysis methods such as probability map overlay, confusion matrices and/or derived predictive metrics (accuracy, loss, sensitivity/TPR, specificity/TNR, ...). In case of a voxel-wise or patch-wise classification/regression, the estimated masks/maps can be displayed and overlaid to the images (Fig. 4). The user can specify the overlay colours transparency and hatching.

For network performance, the training accuracy/loss and the test accuracy/loss can be plotted over the epochs during training. Predictive test metrics derived from the confusion matrix show the test performance. In case of multiple classes, a one-versus-all metric calculation is provided.



Fig. 4: Prediction Analysis: Estimated probability maps overlaid on MR image with user-specified colouring and hatching.

III. USE CASES

The following use cases have been implemented for the GUI: MR motion artifact detection, MR motion correction

via VAE and semantic segmentation of organs in whole-body MRI. For the use cases MR motion artifact detection and motion correction we used MRI scans of the head, abdomen and pelvis of 18 healthy volunteers (3 female, age 27.3 ± 6.3 years) to evaluate the performance of CNN architectures in different settings. Table I summarizes the acquisition parameters. For each volunteer and body region, a reference image and an artifact image are created by either instructing the volunteer to hold still/their breath or move their head or pelvis/breathe freely.

For semantic segmentation we used the epidemiological imaging KORA database [20]. From the imaging protocol of the KORA study, we used the whole body T1-weighted dual gradient echo sequence which was acquired with isotropic resolution of $1.7 \times 1.7 \times 1.7$ mm in coronal orientation in a FOV of 490 × 490 × 272 mm³ which results in a 3D image size of 288 × 288 × 160. Further imaging parameters are: echo times, 1.26/2.52 ms; repetition time, 4.06 ms; flip angle, 9°; bandwidth, 755 Hz/px. In-phase and opposed-phase images were obtained in each measurement from which water and fat images were obtained via a 2-point DIXON. A total number of 173 labeled patient sets were available.

A. MR motion artifact detection

For the MR motion artifact detection, the artifact-affected areas are marked with the drawing tools of the GUI (local labeling). To compare local labeling with global labeling, two networks are used: The first one is the 2D-CNN which we employed in a previous study [10]. Its structure consists of four stages. The first three stages have 2D convolutional layers with ReLU activation and the last stage consists of a fully connected layer with softmax output. The second network is a 3D network which was employed in another study [11], [12] and is denoted as VNetArt. It inspired by VNet [21] and consists of four stages. Each of the first three stages has two convolutional layers with parametric ReLU activation and a concatenation layer which combines the output after the two convolution operations with the feedforward input. Each stage is completed by a max-pooling downsampling. The fourth stage is a fully connected dense layer with softmax output.

Each patch is thus assigned an artifact probability p_{art} . 2D patches $[p_x \times p_y] = [40 \times 40]$ with $o_x = o_y = 0.5$, and 3D patches $[p_x \times p_y \times p_z] = [40 \times 40 \times 10]$ with $o_x = o_y = o_z = 0.5$ are used. The label corresponding to "artifact" is assigned if more than 50%/75% of a patch is marked as artifact in the 2D/3D case, respectively. For each body region, separate networks are trained using either only the body-specific training data or all of it. In all scenarios, both architectures with global and local labeling are trained for a comparative performance analysis. The following parameters are used for training: learning rate = 0.001, epochs = 100, Adam optimizer and batch size = 64. A leave-one-subject-out cross-validation with 2 runs was conducted. After the training is completed, the patches of a volunteer who is unknown to the networks are classified. A higher patching overlap of $o_x = o_y = o_z = 0.9$ is chosen for this evaluation to create a probability map with detailed localization and quantification of motion artifacts.

Quantitative results in terms of accuracy, true negative rate (TNR) and true positive rate (TPR) for all body regions are depicted in Figure 5. For the 2D-CNN with local labeling (LL), an accuracy of 79.4% is calculated. This represents an improvement of 9.4% compared to the previous approach of global labeling (GL). Local labeling combined with VNetArt achieves an accuracy of 90.7%, which increases former results for global labeling by 8.1%.

Fig. 6 shows the probability maps (respective top rows) and the decided artifact region (respective bottom rows) for the 2D-CNN and the VNetArt with both global and local labeling in the head (rigid motion), abdomen (non-rigid motion) and pelvis (rigid motion). The column on the far left depicts the labeling, i.e. the marking of the region affected by artifacts (upper image) and the resulting hard-thresholded artifact region (lower image). The probability maps are the estimated p_{art} for each patch.

In the head (Fig. 6 (a)), for both 2D-CNN and VNetArt, local labeling improves the artifact localization compared to global labeling which is also reflected in a higher accuracy and TPR.

The abdominal images (Fig. 6 (b)) show only a weak expression of artifacts in the anterior-posterior direction caused by non-rigid motion, especially in the lung area and in the background. This is reflected in the comparably low results in Fig. 5. Nevertheless patch-wise labeling leads to significant improvements. The classification accuracy is 67% for the 2D-CNN with local and 63% with global labeling. VNetArt shows higher accuracies for both labeling approaches. Local labeling results in an accuracy of 85.9%, while global labeling leads to an accuracy of 76.7%, corresponding to an improvement of 12%. The 2D-CNN with local labeling improves recognition of non-rigid movements in the abdomen.

	head T1-weighted	abdomen T1-weighted	pelvis T1-weighted	pelvis T2-weighted
TE [ms]	8.4	10	11	86
TR [ms]	750	800	900	4200
flip angle [°]	140	150	160	150
bandwith [Hz/px]	260	190	190	200
averages	2	2	2	2
phase-encoding direction	left-right	anterior-posterior	anterior-posterior	anterior-posterior
matrix size	256×196×40	320×192-256×30	320×192-240×40	320×192-240×40
voxel size [mm ³]	$1 \times 1 \times 3$	$1.25 \times 1.25 \times 5$	$1.25 \times 1.25 \times 5$	1.25×1.25×5

TABLE I: Acquisition parameters of MR data for motion artifact detection and correction.



Fig. 5: MR motion artifact detection: Obtained accuracies, true negative rate (TNR) and true positive rate (TPR) for both networks (2D-CNN, VNetArt) with global (GL) and local (LL)

The resulting accuracies for T1-weighted images of the pelvis are given in Fig. 5 and show very little difference between both labeling approaches and for either network. The 2D-CNN with global labeling even exceeds the results achieved with local labeling by 1.2%. The reason for that is, that the pelvic region in general shows strong motion artifacts in all images, with only small areas free of artifacts. The overall best results are provided by the combination of VNetArt and local labeling with an accuracy of 96%. Figure 6 (c) shows the probability maps. With local labeling, the network is able to better distinguish artifact-free regions from artifact-affected regions. Using T2-weighted images leads to similar results (see Fig. 5).

For the MR motion artifact detection the local labeling exceeds global labeling on average by 5%.

Comparing the 2D and 3D networks, the 3D VNetArt outperforms on average the 2D-CNN in terms of accuracy, 87.5% compared to 76.3% for the global labeling and 91.9% compared to 80.0% with local labeling. This corresponds to an average increase of 15% for both labeling approaches. Overall, local labeling increases classifier performance for 2D-CNN and VNetArt by 4.8% and 5.1%, respectively.

B. MR motion correction via variational autoencoder

Once the networks are trained to detect motion artifacts, they have embedded some information on how motion looks like in the image. So instead of only detecting the artifacts, an additional network can be appended to the structure which performs a correction step [12]. This end-to-end trained architecture is built up as a VAE with VNetArt architecture in the individual parts, denoted as VAE-VNetArt. The VAE-VNetArt is separated into three parts as shown in Fig. 8: standalone encoder, shared encoder, decoder.

The encoding parts resemble the encoding branch of the proposed VNetArt whilst the decoding part is the direct composite decoding branch of the VNetArt. The network is trained on motion-free and motion-affected patches. After training the image-translation step, the network can estimate a motion-corrected image from a motion-affected image.

In the training, motion-free and motion-affected patches are firstly input into separate encoder branches which consist of the first encoding stage: 2D convolutional layers with ReLU activation function and residual path. Afterwards the individual motion-free and motion-affected feature maps are concatena-



(c) pelvis

Fig. 6: MR motion artifact detection: Probability mask overlay with decided artifact region for 2D-CNN and VNetArt with local labeling (LL) and global labeling (GL) in (a) head (rigid motion), (b) abdominal (non-rigid motion) and (c) pelvic (rigid motion) images.



SSIM=0.89, NMI=0.43 SSIM=0.90, NMI=0.66

Fig. 7: MR motion correction: VAE-VNetArt generated motion-corrected images in head (rigid motion) and abdomen (non-rigid motion) which uses the respective motion-affected images as input. The structural similarity index (SSIM) and normalized mutual information (NMI) are calculated in comparison to the motion-free (reference) images.



Fig. 8: MR motion correction: Proposed VAE-VNetArt consisting of two standalone encoders for motion-free and motionaffected patches, shared encoder, dense net in latent space and decoder to retrieve motion-corrected patches.

ted along the channel direction and processed simultaneously in the shared encoder block. This block consists of the two remaining VNetArt encoding stages. In the latent space two dense layers perform a separate mapping to mean and variance from which a subsequent selection layer depicts the values. This built up feature map is processed in the decoding block outputting a motion-corrected patch. Skip connections between layers and blocks improve stability and share information amongst different resolution levels. An Adam optimizer was used with $\beta_1 = 0.9, \beta_2 = 0.999$ and $\varepsilon = 1e - 08$. The loss function is composed of a perceptual loss [22], a Charbonnier loss [23] and a gradient entropy loss [24]. The perceptual loss uses a separate VGG-19 network [25] which was pretrained on the ImageNet database.

The presented results in Fig. 7 show a very good visual as well as quantitative performance in terms of structural similarity index (SSIM) [26] and normalized mutual information (NMI) for the correction of rigid motion artifacts in the head. The correction of non-rigid motion in the abdominal case is still acceptable, but suffers from through-plane motion which cannot be perfectly corrected by the current 2D network. Future studies focus on the 3D extension which demand however careful parametrization to guarantee convergence.

C. Semantic segmentation of organs in whole-body MRI

Another use case for the GUI are segmentation tasks. We looked at semantic segmentation of specific organs in wholebody MRI. The goal is to implement and validate a CNN-based 3D semantic segmentation of the liver, spleen and background on multi-contrast MR data. From the KORA study we used the whole-body T1-weighted dual gradient echo sequence of 173 patients. Data includes the image images of patients with and without diabetes. To obtain the ground truth data for training and validation, the liver and spleen of all available data sets were manually segmented using the proposed GUI labeling tool. The resulting masks were stored for further processing.

The proposed CNN-based network [27], denoted as DCNet, includes several schemes for optimizing semantic segmentation: UNet for pixel-wise localization [28], VNet for volumetric medical image segmentation [21], ResNet to cope with vanishing gradients and degradation problem [29] and DenseNet, to enable deep supervision [30]. The structure of the DCNet consists of an encoding and decoding branch. Each stage consists of N_B dense blocks with *L DenseConv* layers and N_T *TransitionLayerPool/Up* blocks with concatenations in between. This resulted in 152 layers in total. For training, the number of epochs has been set to 50 and the batch size to 48. Images were patched into sizes of $32 \times 32 \times 32$ and 50% overlap. A 4-fold patient-leave-out cross-validation was performed.

Figure 9 shows the comparison of the ground truth (labeling) with the 3D semantic segmentation. Organ boundaries and background region were precisely differentiated. Test accuracy reached a maximum of 99.7% and 99.9% in the liver and spleen, respectively.



Fig. 9: Semantic segmentation: 3D segmentation of liver (red) and spleen (green) in four diabetic patients via proposed DCNet in comparison to labeled ground truth (labeling).

IV. CONCLUSIONS

In this work, a GUI was implemented, which is composed of the modules *Data Viewing*, *Network Training*, *Network Visualization* and *Prediction Analysis*. The GUI is a powerful and easy-to-use tool suitable for processing and visualizing data. Processing of MRI data with the GUI was examined in the uses cases: MR motion artifact detection, MR motion correction via VAE and semantic segmentation of organs in whole-body MRI.

All use cases were performed in conjunction with the GUI and users reported an easier data and result handling as well as operation of the training itself. The proposed GUI can assist researchers and radiologists in the field of machine/deep learning to enable an easy-to-handle and maintain processing. A flexible data display with possible network output overlay allows smooth network performance evaluation. Moreover, the performance can be examined via confusion matrices, derived metrics and by displaying its trained visual content. Our GUI provides several core functionalities and is due to its modular structure expandable by new modules with further functions to target different applications such as classification, regression, reconstruction, correction and/or segmentation tasks. The GUI as well as the CNNs are publicly available under Apache 2 license: https://github.com/thomaskuestner/CNNArt

REFERENCES

- F. Bamberg, H. U. Kauczor, S. Weckbach, C. L. Schlett, M. Forsting, S. C. Ladd, K. H. Greiser, M. A. Weber, J. Schulz-Menger, T. Niendorf, T. Pischon, S. Caspers, K. Amunts, K. Berger, R. Bulow, N. Hosten, K. Hegenscheid, T. Kroncke, J. Linseisen, M. Gunther, J. G. Hirsch, A. Kohn, T. Hendel, H. E. Wichmann, B. Schmidt, K. H. Jockel, W. Hoffmann, R. Kaaks, M. F. Reiser, and H. Volzke, "Whole-Body MR Imaging in the German National Cohort: Rationale, Design, and Technical Background," *Radiology*, vol. 277, no. 1, pp. 206–20, 2015.
- [2] W. Ollier, T. Sprosen, and T. Peakman, "UK Biobank: from concept to reality," *Pharmacogenomics*, vol. 6, no. 6, pp. 639–646, 2005.
- [3] M. Zaitsev, J. Maclaren, and M. Herbst, "Motion artifacts in MRI: A complex problem with many partial solutions," *J. Magn. Reson. Imaging.*, vol. 42, no. 4, pp. 887–901, 2015.
- [4] J. Maclaren, M. Herbst, O. Speck, and M. Zaitsev, "Prospective motion correction in brain imaging: a review," *Magn. Reson. Med.*, vol. 69, no. 3, pp. 621–36, 2013.
- [5] F. Godenschweger, U. Kagebein, D. Stucht, U. Yarach, A. Sciarra, R. Yakupov, F. Lusebrink, P. Schulze, and O. Speck, "Motion correction in MRI of the brain," *Phys. Med. Biol.*, vol. 61, no. 5, pp. R32–56, 2016.
- [6] J. Y. Cheng, T. Zhang, N. Ruangwattanapaisarn, M. T. Alley, M. Uecker, J. M. Pauly, M. Lustig, and S. S. Vasanawala, "Free-breathing pediatric MRI with nonrigid motion correction and acceleration," *J. Magn. Reson. Imaging.*, vol. 42, no. 2, pp. 407–20, 2015.
- [7] G. Cruz, D. Atkinson, C. Buerger, T. Schaeffter, and C. Prieto, "Accelerated motion corrected three-dimensional abdominal MRI using total variation regularized SENSE reconstruction," *Magn. Reson. Med.*, vol. 75, no. 4, pp. 1484–98, 2016.
- [8] L. Feng, R. Grimm, K. T. Block, H. Chandarana, S. Kim, J. Xu, L. Axel, D. K. Sodickson, and R. Otazo, "Golden-angle radial sparse parallel MRI: combination of compressed sensing, parallel imaging, and goldenangle radial sampling for fast and flexible dynamic volumetric MRI," *Magn. Reson. Med.*, vol. 72, no. 3, pp. 707–17, 2014.

- [9] T. Küstner, C. Würslin, M. Schwartz, P. Martirosian, S. Gatidis, C. Brendle, F. Seith, F. Schick, N.F. Schwenzer, B. Yang, and H. Schmidt, "Self-navigated 4D cartesian imaging of periodic motion in the body trunk using partial k-space compressed sensing," *Magn. Reson. Med.*, vol. 78, no. 2, pp. 632–644, 2017.
- [10] T. Küstner, A. Liebgott, L. Mauch, P. Martirosian, F. Bamberg, K. Nikolaou, B. Yang, F. Schick, and S. Gatidis, "Automated reference-free detection of motion artifacts in magnetic resonance images," *Magn. Reson. Mater. Phys. Biol. Med.*, Sep 2017.
- [11] T. Küstner, M. Jandt, A. Liebgott, L. Mauch, P. Martirosian, F. Bamberg, K. Nikolaou, S. Gatidis, F. Schick, and B. Yang, "Automatic motion artifact detection for whole-body magnetic resonance imaging," in *Proceedings of the IEEE International Conference on Acoustics, Speech* and Signal Processing (ICASSP), Calgary, Canada, April 2018.
- [12] Thomas Küstner, Marvin Jandt, Annika Liebgott, Lukas Mauch, Petros Martirosian, Fabian Bamberg, Konstantin Nikolaou, Sergios Gatidis, Bin Yang, and Fritz Schick, "Motion artifact quantification and localization for whole-body MRI," in *Proceedings of the International Society for Magnetic Resonance in Medicine (ISMRM)*, Paris, France, June 2018.
- [13] Thomas Küstner, Annika Liebgott, Lukas Mauch, Petros Martirosian, Konstantin Nikolaou, Fritz Schick, Bin Yang, and Sergios Gatidis, "Automatic reference-free detection and quantification of mr image artifacts in human examinations due to motion," in *Proceedings of the International Society for Magnetic Resonance in Medicine (ISMRM)*, Honolulu, Hawaii, USA, Apr. 2017.
- [14] Thomas Küstner, Annika Liegbott, Lukas Mauch, Petros Martirosian, Fritz Schick, Fabian Bamberg, Konstantin Nikolaou, Bin Yang, and Sergios Gatidisi, "Automatic reference-free motion artifact detection and quantification in t1-weighted mr images of the head and abdomen," in *Proceedings of the European Society for Magnetic Resonance in Medicine and Biology (ESMRMB)*, Barcelona, Spain, Oct. 2017.
- [15] Thomas Küstner, Ke Liu, Annika Liebgott, Lukas Mauch, Petros Martirosian, Fabian Bamberg, Konstantin Nikolaou, Bin Yang, Fritz Schick, and Sergios Gatidis, "Simultaneous detection and identification of MR artifact types in whole-body imaging," in *Proceedings of the International Society for Magnetic Resonance in Medicine (ISMRM)*, Paris, France, June 2018.
- [16] François Chollet et al., "Keras," https://keras.io, 2015.
- [17] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, Software available from tensorflow.org.
- [18] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer, "Automatic differentiation in pytorch," in *Neural Information Processing Systems (NIPS)*, 2017.
- [19] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.
- [20] R Holle, M Happich, H Löwel, HE Wichmann, MONICA/KORA study

group, et al., "Kora-a research platform for population based health research," *Das Gesundheitswesen*, vol. 67, no. S 01, pp. 19–25, 2005.

- [21] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," *ArXiv e-prints*, June 2016.
- [22] Alexey Dosovitskiy and Thomas Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances* in *Neural Information Processing Systems*, 2016, pp. 658–666.
- [23] Jonathan T Barron, "A more general robust loss function," arXiv preprint arXiv:1701.03077, 2017.
- [24] K. P. McGee, A. Manduca, J. P. Felmlee, S. J. Riederer, and R. L. Ehman, "Image metric-based correction (autocorrection) of motion effects: analysis of image metrics," *J. Magn. Reson. Imaging*, vol. 11, no. 2, pp. 174–181, 2000.
- [25] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint* arXiv:1409.1556, 2014.
- [26] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
- [27] T. Küstner, S. Müller, M. Fischer, J. Weiß, K. Nikolaou, F. Bamberg, B. Yang, F. Schick, and S. Gatidis, "Semantic organ segmentation in 3d whole-body mr images," in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Athen, Greece, 2018.
- [28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Conference of Medical Image Computing and Computer Assisted Intervention (MIC-CAI)*. Springer, 2015, pp. 234–241.
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE* conference on computer vision and pattern recognition (CVPR), 2016, pp. 770–778.
- [30] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten, "Densely connected convolutional networks," arXiv preprint arXiv:1608.06993, 2016.