Review of Data Types and Model Dimensionality for Cardiac DTI SMS-Related Artefact Removal

Michael Tänzer[1(✉)], Sea Hee Yook[1,2], Pedro Ferreira[1,2], Guang Yang[1,2(✉)], Daniel Rueckert[1,3], and Sonia Nielles-Vallespin[1,2]

[1] Imperial College London, London, England
{m.tanzer,g.yang}@imperial.ac.uk
[2] Royal Brompton and Harefield Hospital, London, England
[3] Klinikum rechts der Isar, Technische Universität München (TUM), Munich, Germany

Abstract. As diffusion tensor imaging (DTI) gains popularity in cardiac imaging due to its unique ability to non-invasively assess the cardiac microstructure, deep learning-based Artificial Intelligence is becoming a crucial tool in mitigating some of its drawbacks, such as the long scan times. As it often happens in fast-paced research environments, a lot of emphasis has been put on showing the capability of deep learning while often not enough time has been spent investigating what input and architectural properties would benefit cardiac DTI acceleration the most. In this work, we compare the effect of several input types (magnitude images vs complex images), multiple dimensionalities (2D vs 3D operations), and multiple input types (single slice vs multi-slice) on the performance of a model trained to remove artefacts caused by a simultaneous multi-slice (SMS) acquisition. Despite our initial intuition, our experiments show that, for a fixed number of parameters, simpler 2D real-valued models outperform their more advanced 3D or complex counterparts. The best performance is although obtained by a real-valued model trained using both the magnitude and phase components of the acquired data. We believe this behaviour to be due to real-valued models making better use of the lower number of parameters, and to 3D models not being able to exploit the spatial information because of the low SMS acceleration factor used in our experiments.

Keywords: MRI · Deep learning · Diffusion tensor imaging · Cardiac MRI

1 Introduction

Cardiac diffusion tensor imaging is growing as a novel imaging modality as it is capable of interrogating the microstructure of the beating heart with no invasive

This work was supported in part by the UKRI CDT in AI for Healthcare http://ai4health.io (Grant No. EP/S023283/1), the British Heart Foundation (RG/19/1/34160), and the UKRI Future Leaders Fellowship (MR/V023799/1).
G. Yang, D. Rueckert and S. Nielles-Vallespin—Co-last senior authors.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
O. Camara et al. (Eds.): STACOM 2022, LNCS 13593, pp. 123–132, 2022.
https://doi.org/10.1007/978-3-031-23443-9_12

surgery and without the use of any contrast agent [11], making it accessible for patients with reduced kidney functionality [17] or for frequent scans. In clinical research studies, cardiac DTI has been shown to be useful in phenotyping several cardiomyopathies such as hypertrophic cardiomyopathy (HCM) and dilated cardiomyopathy (DCM) by quantitatively analysing the microstructural organisation and orientation of cardiomyocytes within the myocardium.

As cardiac DTI is becoming more and more studied, the use of deep learning-based (DL) approaches applied to it is similarly increasing [4–6,8,14,18,21,22]. Most of the recent work in the field, although, suffer from a common crucial shortcoming: in order to quickly show the potential of deep learning on improving cardiac DTI, many publications don't go beyond applying some well-known general-purpose architecture to the data they have available, often ignoring inherent properties of the acquisition method. As an example, in much of the recently published work, we find a widespread use of out-of-the-box models such as the popular U-Net [16], which is a 2D model designed for real data. MRI data, on the other hand, is complex by definition and the subject of the acquisition is often 3D rather than 2D, making the standard U-Net ill-fitting for the task.

In this work, we compare the effect of making relatively small architectural changes to the popular U-Net model when applied to a general image-to-image task in DTI. Specifically, we choose the task of removing artefacts from cardiac DTI images acquired with an SMS protocol, and we compare the classic 2D magnitude-only U-Net with 3D and complex versions of the same model, with the goal of providing future researchers with a better starting point for their experimental work. To this end, we also make the code for all of our tested models available on GitHub[1].

2 Background

2.1 Cardiac Diffusion Tensor Imaging

Diffusion tensor imaging measures the diffusion of water molecules for every voxel in the imaged tissue and approximates as a 3D tensor. As the free diffusion of water in the tissue is constrained by the shape of cardiac muscle microstructure, studying such tensors has been shown to give us information related to the shape and orientation of the cardiomyocytes in the imaged tissues.

The cardiac diffusion tensor information is commonly visualised and quantified through four per-voxel metric maps: Mean Diffusivity (MD) that quantifies the total diffusion in the voxel (higher corresponds to more diffusion), Fractional Anisotropy (FA) that quantifies the level of organisation of the tissue (higher corresponds to a higher organisation), Helix Angle (HA) and Second Eigenvector (E2) Angle (E2A) that quantify the 3D orientation and shape of the tissue in the voxel [2,10]. These maps have been shown to be a promising tool for phenotyping many cardiac pathology in a clinical setting [3,12,13].

[1] https://github.com/Michael-Tanzer/architectures-tanzer-stacom22.

2.2 Simultaneous Multi-slice Acquisition (SMS)

Simultaneous multi-slice techniques have been used with great success to reduce the acquisition time in brain diffusion tensor imaging (DTI) [19]. SMS uses a multi-band excitation pulse to simultaneously excite several 2D slices within the imaged tissue. In SMS, each receiver coil collects a single frequency-domain image for all the excited slices, the signal received is a weighted sum of the signals that would be emitted by exciting each slice individually.

By making use of the redundant information from the receiver coils that surround the tissue we want to image, we can then separate the signal from the excited slices using modified versions of the GRAPPA [7] and SENSE [15] algorithms used for in-plane acceleration. Unfortunately, as the information is often insufficient and as the problem is not fully characterised, the slice separation algorithms introduce artefacts in the separated slices [1]. The artefacts are often referred to as interslice-leakage artefact as they arise when information from one slice erroneously ends up on a different slice. Because of the disposition in space of the MRI coils, the leakage between two slices is more evident when the slices are closer in the imaged tissue.

2.3 Deep Learning in DTI

As cardiac DTI grows in popularity, a lot of work has been published trying to improve the acquisition quality or trying to shorten its long scan-times. Among others, Ferreira et al. [5], Tänzer et al. [21], and Phipps et al. [14] reduced the number of repetitions needed to increase the low SNR by using a de-noising framework to restore high image quality from less acquired data. Schlemper et al. [18] applied a cascade of convolutional neural networks to fill-in k-space entries acquired with a compressed sensing protocol, further reducing the scan times.

Ferreira et al. [6] propose a U-Net for the segmentation of the left ventricle, automating part of the DTI maps computation, Cao et al. [4] show how a GAN model can be used as a de-aliasing model for DTI, and Tian et al. [20] work on self-supervised DTI de-noising is also based on a modified U-Net model.

Most of the examples reported above, despite all being a great contribution to the field of deep-learning-accelerated cardiac DTI, have a 2D real-valued U-Net model at their core. This shows that the choice of data type and dimensionality is not the top priority of many influential publications.

3 Methods

3.1 Complex Neural Networks

As MRI data is inherently complex, we explore the possibility of training using complex data. Traditionally, there are two main ways to achieve this: separating the complex data into non-complex components, or using a model that performs complex operations.

The former is a more straight-forward approach: the complex data is split into its real and imaginary components or into magnitude and phase and then a real-values model is trained using the split data. This can be done in multiple ways: either by treating the components as different channels, or by training two separate models, one for each component. In our comparison we split the data in magnitude and phase as they are more meaningful in the physics of MRI and we use them as separate input channels to a unified model.

The latter option requires more thinking: while some operations naturally extend to the complex domain, some others do not and need to be re-designed. We report the main changes to the used operators in Table 1. In the table $z = a + ib$ where $i = \sqrt{-1}$, c is the convolution layer, and tc is the transpose convolution layer.

Table 1. Neural networks operators and their complex counterparts. Marked with a checkmark those operations that don't need substantial modification to extend their usage to complex space.

Operation		Naive imp	Complex equivalent
Multiplication	Inner product	$\sum_i w_i z_i$	✓
	Convolution	$\mathrm{c}(a + ib)$	$\mathrm{c}_r(a) - \mathrm{c}_i(b) + i(\mathrm{c}_i(a) + \mathrm{c}_r(b))$
	Trans. conv	$\mathrm{tc}(a + ib)$	$\mathrm{tc}_r(a) + \mathrm{tc}_i(b) + i(\mathrm{tc}_i(a) - \mathrm{tc}_r(b))$
Activation	Sigmoid	$\frac{1}{1+e^{-(a+ib)}}$	$\mathrm{sigmoid}(a) + i\,\mathrm{sigmoid}(b)$
	ReLU	$\max(0, a + ib)$	$\begin{cases} (\lvert z\rvert + b)\frac{z}{\lvert z\rvert} & \text{if } \lvert z\rvert + b \geq 0 \\ 0 & \text{if } \lvert z\rvert + b < 0 \end{cases}$
	Dropout	$\mathrm{DO}(a) + i\,\mathrm{DO}(b)$	$\mathrm{DO}(a + ib)$
Pooling	Max	$\max_i z_i$	z_k where $\mathrm{argmax}_k\{\lvert z_k\rvert\}$
	Average	$\frac{1}{N}\sum_i z_i$	✓
Normalisation	Batch norm	$\hat{\mathbf{z}} = \frac{\mathbf{z} - \mathbb{E}[\mathbf{z}]}{\sqrt{\mathbb{V}[\mathbf{z}] + \epsilon}}$	✓
Loss	Euclidean	$\lvert\lvert Y - Z\rvert\rvert_2^2$	$\lvert\lvert \mathrm{abs}(Y) - \mathrm{abs}(Z)\rvert\rvert_2^2$
	L1	$\lvert\lvert Y - Z\rvert\rvert_1$	$\lvert\lvert \mathrm{abs}(Y) - \mathrm{abs}(Z)\rvert\rvert_1$

Once we have defined these basic operations we can then build a model that uses them and is therefore fit to process complex numbers. The advantage of this approach is that the model makes use of properties of complex numbers during the training instead of letting the network learn a relationship between the input channels like in the previous method.

3.2 Experimental Setting

In order to provide a better understanding on the effect of using a 3D or complex model, we compare all combinations of architectures obtained by modifying the following properties:

- 2D vs 3D: whether the model uses 2D operations or 3D operations.
- Magnitude vs complex data: whether we train the model with the absolute component alone or whether we use the full complex representation of the data. Notice that "complex data" is further split into 1. fully complex models that use complex operations and 2. models that keep standard operations, but use the phase data as a separate input channel. We refer to the former magnitude-only models as "Mag", fully complex models as "Comp", and magnitude-and-phase models as "MagPhs" for brevity.
- All slices vs individual slices: whether the model is trained to correct all SMS-acquired slices simultaneously or whether to correct a single slice at a time. When all the slices are used and the model is 3D, the slices are arranged in the third spatial dimension, while for 2D models the slices are treated as image channels.

This comparison results in 12 combinations of dimensionality (2 types), data (3 types), and input data (2 types). When a single model is referred to, we often use a shorthand: for instance, if we refer to a 2D fully-complex model trained on all the SMS slices, we will shorten it as "2D-All-Comp".

All the models were trained for 200 epochs with the Adam optimiser [9], a learning rate of 0.0003 that was then lowered by a factor of 10 after 100 epochs, a batch size of 16, mean absolute error loss, and residual learning. The data was padded and normalised in the range 0 to 1 and then randomly augmented with random rotation and random vertical and horizontal flipping. All the results were computed on the test set for the epoch in which the validation MAE was lowest.

In order to ensure consistency, we kept the model size and architecture as fixed as possible by choosing a number of parameters, 3 millions, and a general architecture (U-Net with 5 layers with a doubling number of channels in each encoding layer) and subsequently adjusting the initial number of channels. The models have the following starting number of channels:

- 2D Mag and MagPhs: 28
- 3D Mag and MagPhs: 16
- 2D Complex: 20
- 3D Complex: 11

The data used for the training came from 31 ex-vivo swine hearts and was acquired with SMS factor equal to 2 and distance factor equal to 400%. Each heart was scanned in multiple locations to cover as much of the volume as possible. The ground truth images were obtained by scanning each heart again with the same protocol but no SMS acceleration. This results in around 43,000 2D complex slices for the training, 1200 slices for validation and 1200 slices for testing.

3.3 Results Evaluation

When working with DTI data, there are two main components we are interested in evaluating: the acquired images and the DTI maps derived from the images.

The latter are particularly important as they are the main tools a clinician would use in a clinical setting. When evaluating the artefact-removal results of our models we therefore need to take both into account.

To evaluate the image quality we use mostly standard wide-spread metrics:

- Mean Absolute Error (MAE) ↓: $\frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left|X^{(i,j)} - Y^{(i,j)}\right|$ where x and y are the predicted and target images respectively. For complex images the MAE is computed with respect to their magnitude and for MagPhs the phase information is not taken into account.
- Peak Signal to Noise Ratio (PSNR) ↑: $10\log_{10}\left(\frac{\mathrm{MAX}_X^2}{\mathrm{MSE}}\right)$ where MAX_X is maximum pixel intensity value across the image and MSE is the mean squared error between the predicted image and the target image. In the case of complex images we compute the PSNR on their magnitude and for the MagPhs case we discard the phase data.
- Structural Similarity Index (SSIM) ↑: SSIM measures the perceived image degradation based on the loss of structure between the output and target images. It is computed as follows:

$$\mathrm{SSIM}(X,Y) = \frac{(2\mu_X\mu_X + c_1)\,(2\sigma_{XY} + c_2)}{(\mu_X^2 + \mu_Y^2 + c_1)\,(\sigma_X^2 + \sigma_X^2 + c_2)} \tag{1}$$

  Where μ_I represents the mean over I, σ_I the standard deviation over I, σ_{IJ} the covariance, and c_1 and c_2 are fixed scalars used for numerical reasons.

When analysing the DTI maps, we need to distinguish between scalar maps (MD and FA) and angular maps (HA and E2A). While for scalar maps we can use MAE (↓) as an error metric, angular maps are defined in the range $[-90°, 90°)$ and they wrap around at the two extrema of this range. For these maps we use the Mean Angle Absolute Error (MAAE, ↓) defined below instead

$$\mathrm{MAAE}(X,Y) = \frac{1}{NM}\sum_{i=0}^{N}\sum_{j=0}^{M}\begin{cases}\left|X^{(i,j)} - Y^{(i,j)}\right|, \; if \;\; \left|X^{(i,j)} - Y^{(i,j)}\right| < 90° \\ 180° - \left|X^{(i,j)} - Y^{(i,j)}\right|, \; \text{otherwise}\end{cases}$$

Moreover, as the DTI maps are not well-defined for the background voxels, we only consider the metrics computed on voxels belonging to the cardiac tissue.

All the results are reported as the median over the means on a per-slice basis and its inter-quartile range as *median [iqr]*. When we perform a statistical significance test we use the Wilcoxon rank test with $P = 0.05$.

4 Results

In Tables 2 and 3 we report the performance metrics for the output images and for the DTI maps on the test set for all our models. The values of MD have been scaled by 10^5 and the values of FA by 10^2 to improve readability. In the table we mark in bold the best result across all models and we underline the second-best result.

We also visually report an example chosen from the test set in Fig. 1.

Table 2. Numerical results related to the DTI maps computed from the AI-processed data. We report MAE for MD and FA and MAAE for HA and E2A. In bold and underlined, respectively, the best and second-best results for each metric.

Run name	HA		E2A		MD ($\times 10^5$)		FA ($\times 10^2$)	
	Median	IQR	Median	IQR	Median	IQR	Median	IQR
2D-ALL-ABS	16.9	5.0	26.1	4.8	5.44	1.51	5.95	1.19
2D-ALL-Comp	18.0	4.6	26.4	4.7	5.50	0.97	**5.41**	1.33
2D-ALL-MagPhs	16.7	2.6	**24.5**	4.4	5.55	1.03	5.86	0.40
3D-ALL-ABS	16.9	4.3	26.4	3.5	5.98	1.26	6.24	0.59
3D-ALL-Comp	19.4	4.8	27.8	5.1	6.19	1.47	6.03	3.26
3D-ALL-MagPhs	**15.0**	3.8	25.5	4.8	4.89	0.67	5.46	1.35
2D-Single-ABS	16.4	4.2	25.5	5.5	5.26	1.28	5.83	0.51
2D-Single-Comp	18.6	4.7	27.3	5.0	5.22	0.95	5.63	2.25
2D-Single-MagPhs	18.9	3.8	27.2	4.7	5.71	1.17	5.93	0.37
3D-Single-ABS	17.8	3.1	27.0	5.4	6.41	1.92	5.76	0.34
3D-Single-Comp	18.9	5.2	27.0	5.3	**4.82**	0.96	5.48	1.12
3D-Single-MagPhs	18.4	3.2	27.9	6.0	6.28	1.88	5.88	0.72

Table 3. Numerical results related to the artefact-removal output of our proposed AI models. The results here refer to the images produced by our model. In bold and underlined, respectively, the best and second-best results for each metric.

Run name	MAE ($\times 10^3$)		PSNR		SSIM	
	Median	IQR	Median	IQR	Median	IQR
2D-ALL-ABS	1.75	0.73	37.2	3.56	0.917	0.021
2D-ALL-Comp	1.82	0.78	36.6	3.26	0.911	0.026
2D-ALL-MagPhs	1.84	0.78	**37.4**	2.85	0.921	0.020
3D-ALL-ABS	1.74	0.62	37.0	2.80	0.920	0.018
3D-ALL-Comp	7.15	3.92	31.1	1.41	0.618	0.012
3D-ALL-MagPhs	1.71	0.71	34.9	3.94	0.922	0.020
2D-Single-ABS	1.68	0.76	37.2	4.52	0.923	0.022
2D-Single-Comp	2.04	0.75	36.2	3.26	0.892	0.024
2D-Single-MagPhs	1.67	0.75	36.6	5.31	**0.925**	0.022
3D-Single-ABS	**1.66**	0.72	27.2	5.46	0.909	0.028
3D-Single-Comp	5.24	4.19	36.2	3.37	0.889	0.020
3D-Single-MagPhs	1.70	0.77	37.1	3.83	0.923	0.026

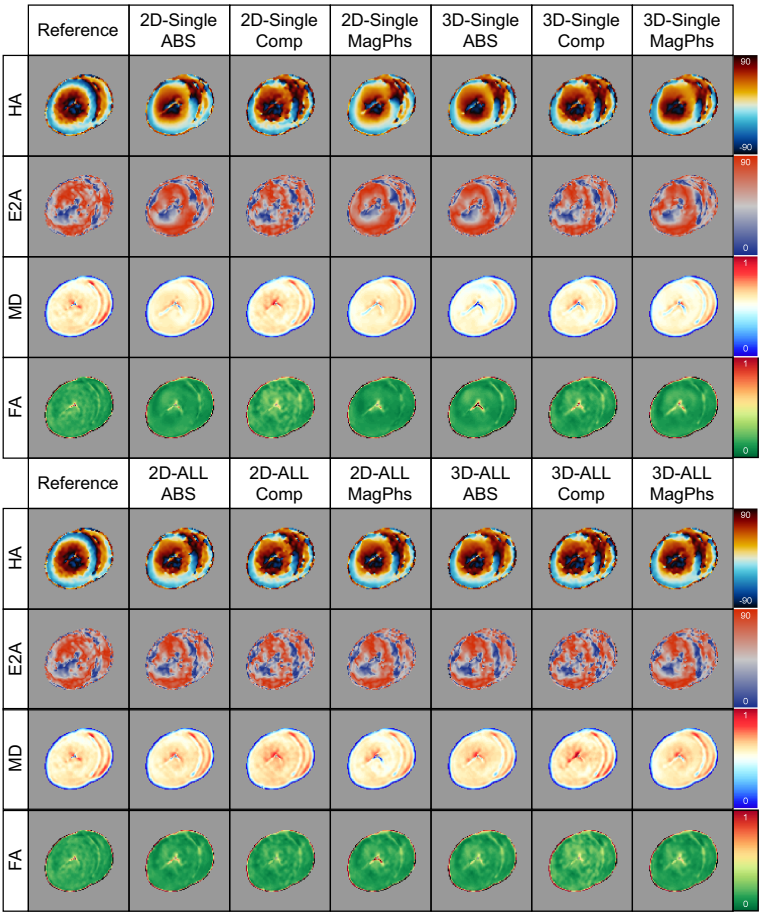

Fig. 1. Example DTI maps for an example from the test set.

5 Discussion

Analysing Tables 2 and 3, we can see how simpler 2D models that only use magnitude data seem to be extremely stable and easy to train, resulting in very good performance overall. Moreover, these models were also faster to train compared to 3D or complex models. 2D MagPhs models also performed remarkably well given how little the architectures differs from the 2D real-values model.

Fully complex and 3D models are significantly slower to train on average due to the higher number of operations needed. Moreover, they are also associated with worse performance especially for the DTI maps results.

There is an important point to be made about the worse performances of the more advanced models: as we aimed to keep the number of parameters fixed, we reduced the number of channels for each layer in the more advanced models,

reducing the effective capacity of these models and therefore negatively affecting their performances. It can be hypothesised that these more advanced models would perform better given a higher number of fixed parameters and a longer training time.

Moreover, as our dataset was acquired with SMS factor equal to 2, the 3D architectures have little additional spatial information to exploit when removing the artefacts from the slices. If the acceleration factor were higher, it is likely that better exploiting the 3D spatial information would produce better performances.

Visually, from Fig. 1, we can notice how the AI-derived maps all show extremely similar flaws regardless of the model used (e.g. top-left quadrant of the HA maps), suggesting that 1. the information learned by the models is similar, and 2. almost none of the models is able to overcome the incorrect information present in the SMS version of the images used as input.

6 Conclusion

As cardiac DTI becomes closer and closer to a clinical reality, deep learning is also becoming a vital tool in alleviating some of its downsides such as long scan times and low SNR. In the hurry associated with a new emerging field, many authors prioritise proof-of-concept work to showcase innovative ideas over exploring and comparing known and common options. In this work we lay out the basis of a comparison between input types and model dimensionality and we show how, despite our initial assumption, given a fixed number of parameters and a reasonable train time, 2D models vastly outperform their 3D counterparts and complex-valued networks are not preferable. On the other hand, the seemingly naive use of separate input channels for magnitude and phase data has beneficial performance over discarding the phase information as it is commonly done.

As an advice for future studies, we suggest the use of 2D models and, if available, the of phase information together with the absolute information for their model-development starting point.

References

1. Barth, M., Breuer, F., Koopmans, P.J., Norris, D.G., Poser, B.A.: Simultaneous multislice (SMS) imaging techniques. Magn. Reson. Med. **75**(1), 63–81 (2016). https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.25897
2. Basser, P.J.: Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. NMR Biomed. **8**(7), 333–344 (1995)
3. Bihan, D.L., et al.: Diffusion tensor imaging: Concepts and applications. J. Magn. Reson. Imaging **13**(4), 534–546 (2001). https://onlinelibrary.wiley.com/doi/abs/10.1002/jmri.1076
4. Cao, Y., et al.: CS-GAN for High-Quality Diffusion Tensor Imaging. Tech. rep., Guizhou University (Sep 2020). https://doi.org/10.21203/rs.3.rs-65572/v1,https://www.researchsquare.com/article/rs-65572/v1
5. Ferreira, P.F., et al.: Accelerating cardiac diffusion tensor imaging with a u-net based model: toward single breath-hold. J. Magn. Reson. Imaging (2022). https://onlinelibrary.wiley.com/doi/abs/10.1002/jmri.28199

6. Ferreira, P.F., et al.: Automating in vivo cardiac diffusion tensor postprocessing with deep learning-based segmentation. Magn. Reson. Med. **84**(5), 2801–2814 (2020). https://onlinelibrary.wiley.com/doi/abs/10.1002/mrm.28294
7. Griswold, M.A., et al.: Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn. Reson. Med. **47**(6), 1202–1210 (2002). http://doi.wiley.com/10.1002/mrm.10171
8. Karimi, D., Gholipour, A.: Diffusion tensor estimation with transformer neural networks. Artifi. Intell. Med. **130**, 102330 (2022). https://www.sciencedirect.com/science/article/pii/S0933365722000951
9. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv:1412.6980 [cs] (Jan 2017)
10. Kung, G.L., et al.: The presence of two local myocardial sheet populations confirmed by diffusion tensor mri and histological validation. J. Magn. Reson. Imaging **34**(5), 1080–1091 (2011)
11. Mori, S., Zhang, J.: Principles of diffusion tensor imaging and its applications to basic neuroscience research. Neuron **51**(5), 527–539 (2006). https://www.cell.com/neuron/abstract/S0896-6273(06)00634-9
12. Nielles-Vallespin, S., et al.: Assessment of myocardial microstructural dynamics by in vivo diffusion tensor cardiac magnetic resonance. J. Am. College Cardiol. **69**(6), 661–676 (2017). https://linkinghub.elsevier.com/retrieve/pii/S0735109716373272
13. Nielles-Vallespin, S., Scott, A., Ferreira, P., Khalique, Z., Pennell, D., Firmin, D.: Cardiac diffusion: technique and practical applications. J. Magn. Reson. Imaging **52**(2), 348–368 (2020). https://onlinelibrary.wiley.com/doi/abs/10.1002/jmri.26912
14. Phipps, K., et al.: Accelerated in vivo cardiac diffusion-tensor mri using residual deep learning-based denoising in participants with obesity. Radiol. Cardiothoracic Imag. **3**(3), e200580 (2021). https://pubs.rsna.org/doi/abs/10.1148/ryct.2021200580
15. Pruessmann, K.P., Weiger, M., Scheidegger, M.B., Boesiger, P.: SENSE: sensitivity encoding for fast MRI. Magn. Reson. Med. **42**(5), 952–962 (1999)
16. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. arXiv:1505.04597 [cs] (May 2015)
17. Schlaudecker, J.D., Bernheisel, C.R.: Gadolinium-Associated Nephrogenic Systemic Fibrosis. Am. Family Phys. **80**(7), 711–714 (2009). https://www.aafp.org/afp/2009/1001/p711.html
18. Schlemper, J., et al.: Stochastic deep compressive sensing for the reconstruction of diffusion tensor cardiac MRI. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11070, pp. 295–303. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00928-1_34
19. Setsompop, K., et al.: Improving diffusion MRI using simultaneous multi-slice echo planar imaging. NeuroImage **63**(1), 569–580 (2012). https://linkinghub.elsevier.com/retrieve/pii/S1053811912006477
20. Tian, C., Fei, L., Zheng, W., Xu, Y., Zuo, W., Lin, C.W.: Deep Learning on Image Denoising: An overview. arXiv:1912.13171 [cs, eess] (Aug 2020)
21. Tänzer, M., et al.: Faster diffusion cardiac MRI with deep learning-based breath hold reduction. In: Medical Image Understanding and Analysis, pp. 101–115. LNCS. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-12053-4_8
22. Weine, J., van Gorkum, R.J.H., Stoeck, C.T., Vishnevskiy, V., Kozerke, S.: Synthetically trained convolutional neural networks for improved tensor estimation from free-breathing cardiac DTI. Comput. Med. Imaging Graph. **99**, 102075 (2022). https://www.sciencedirect.com/science/article/pii/S0895611122000489