

Whole Image Synthesis Using a Deep Encoder-Decoder Network

Vasileios Sevetlidis^{1,2}(✉), Mario Valerio Giuffrida^{1,2},
and Sotirios A. Tsaftaris^{1,2}

¹ PRIAn, IMT School Advanced Studies Lucca, Lucca, Italy
{vasileios.sevetlidis, valerio.giuffrida}@imtlucca.it

² School of Engineering, University of Edinburgh, Edinburgh, UK
s.tsaftaris@ed.ac.uk

Abstract. The synthesis of medical images is an intensity transformation of a given modality in a way that represents an acquisition with a different modality (in the context of MRI this represents the synthesis of images originating from different MR sequences). Most methods follow a patch-based approach, which is computationally inefficient during synthesis and requires some sort of ‘fusion’ to synthesize a whole image from patch-level results. In this paper, we present a whole image synthesis approach that relies on deep neural networks. Our architecture resembles those of encoder-decoder networks, which aims to synthesize a source MRI modality to an other target MRI modality. The proposed method is computationally fast, it doesn’t require extensive amounts of memory, and produces comparable results to recent patch-based approaches.

Keywords: Image synthesis · MRI · Stacked neural network · Autoencoder

1 Introduction

Image synthesis has attracted a lot of attention lately due to exciting potential applications in medical imaging, since synthesized images for example may be used to impute missing images (in a large database, e.g., as in [22]), to derive images lacking a particular pathology, which is not present in the input modality (for detection purposes, e.g., [29]), to increase the resolution of input data (e.g., [14]), to perform attenuation correction (e.g., [3]), and others.

Early works in image synthesis followed a physics driven approach [6, 26], using the physical models of the acquisition, they applied directly the transformation on images. Polynomial mixture models like in [7] or non parametric approaches like joint histogram [15] optimized a transformation map from a single image into an other modality. The idea of using raw data directly within the synthesis started with [20], which utilized non-local pair-wise interactions in a super-resolution context. This idea, together with the pioneering work in image

V. Sevetlidis and M.V. Giuffrida—Equal contribution.

analogies [8], spawned several works in data driven synthesis. Typically this happens in a supervised fashion¹, where pairs of images (or volumes) corresponding to the same subject but of different modality are being used. Modality can refer to different physical imaging schemes such as CT, MRI, US, PET but also within an MRI context to distinguish images acquired by different sequences for example T_1 vs. T_2 . Typically, these approaches break the available training data in patches and construct a database linking patches among each other. During inference, the query image is used to find similar in appearance patches in the database and the synthesized modality is generated by fusing the matched patches. For example, similar to label propagation [2, 19, 28], the authors in [11, 29] approached the problem of synthesis with nearest neighbors patch-matching. Similarly, the work in [4] used a generative model of image synthesis using a probabilistic framework. These methods are simple but they require a lot of memory and computational time during inference. Also, the process of how individual patch results are fused together into a final image may have undesired effects. Blending the patches by using simple averaging leads to smoothing, and approaches which they use only the central pixel of the patches, only they may lead to noisy and locally discontinuous outcomes.

One approach which at least reduces the computational cost at run-time during inference, is treating the image synthesis as a regression [23]. A mapping is learned to relate features around a local neighborhood from the input modality to a pixel in the target modality, for example with the use of a neural network [23]. Another example is in [14], where they learn the joint probability between high resolution and low resolution patches, for the purpose of super resolution. Similarly, coupled sparse representation [21], and random forest approaches [1, 12] use patches from the source images for regression analysis, in order to perform the synthesis of a target modality. These approaches are usually less computationally intensive, because they usually store only the mapping function. Also, depending on the approach, inference can be simple, but still they operate at the patch or pixel level.

In this paper, motivated by the above shortcomings, we introduce a new deep learning approach which we term *Deep Encoder-Decoder Image Synthesizer* (DEDIS). Compared to patch-based methods, DEDIS retains low computational/memory requirements, and it is capable of predicting the *whole* image directly, and hence it provides homogeneous and sharp synthetic images. This multi-output regression is achieved based on a deep encoder network which draws inspiration from Stacked Denoising Autoencoders [16]. Essentially, DEDIS (with its architecture shown in Fig. 1), given an input source image, provides as an output a synthesized modality of the same size as the input. Our training input consists of input-output pairs of imaging data. To prevent over-fitting, which can occur when networks are deep, we use dataset augmentation and bottleneck middle layers, which they compress and find useful representations [25]. To initialize the network with reasonable weights (filters), ensuring better convergence,

¹ There is also the recent exciting unsupervised work by [24], however for ease of introducing the reader to the topic this is not discussed here.

we rely on layer-wise pre-training with Restricted Boltzmann Machine (RBM), because it has been shown in the literature the relationship between RBM and Autoencoders [13]. The fine tuning of the complete network is obtained via back-propagation such that the filters are being updated, in order the reconstructed (synthesized) modality to match the desired output.

To evaluate our method we used the SISS brain multimodal MR dataset from the ISLES 2015 workshop [17] and we compared with a classical patch-based approach [29]. Overall, our findings show that the proposed method is capable of preserving anatomical details. and the quantitative analysis, which has been measured with classical measures according to the literature, showed similar performance amongst the two. DEDIS can synthesize a full volume in ~ 0.63 s.

The rest of the paper is organized as follows. In Sect. 2, the proposed DEDIS architecture is described alongside its pre-training and inference steps. Section 3 presents experimental results, while Sect. 4 offers conclusions.

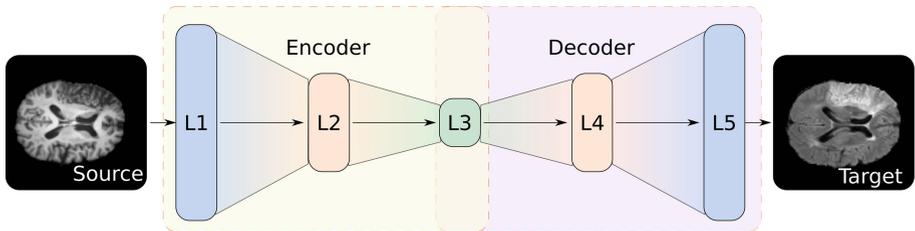


Fig. 1. The proposed DEDIS network, which is able to synthesize a source image into the target modality. Pre-training is performed using Restricted Boltzmann Machines. The GB-RBM and the BB-RBM are being used greedily to initialize the weights between $L1$, $L2$, and $L3$, using the source modality images. The same protocol is followed for $L5$, $L4$, and $L3$, but instead with the use of the target modality.

2 Deep Encoder-Decoder Image Synthesizer (DEDIS)

Our image synthesis approach relies essentially on finding a mapping between an input image \mathbf{I}_S and a desired \mathbf{I}_T , which they correspond to the source and target modalities. Assuming that the sizes are identical, this is a multi-input multi-output regression problem. To find this mapping we rely on a Deep Neural Network architecture. We present sequentially its description, the pre-training of the layer weights, the fine-tuning of the network parameters, and finally we describe inference at test time.

2.1 The DEDIS Architecture

The architecture we devised is inspired by *Stacked Autoencoders* [16] and it is shown in Fig. 1. It can be easily trained in supervised fashion to learn the non-linear mapping, which relates two modalities.

Our goal is to find the relationship which associates the representations of two different modalities. Thus, we added an intermediate layer between the encoding/decoding layers, in order to observe this latent relationship. Specifically, the input layer $L1$ receives a full slice of a 2D image \mathbf{I}_S from the source modality’s volume acquisition. Then, $L2$ and $L3$ aim to “encode” the input. Observe that, as depicted in Fig. 1, they have lower dimensions. This is deliberate, as by adding this bottleneck improves regularization and reduces over-fitting [25]. The subsequent layers $L4$ and $L5$ essentially “decode” the information coming from previous layers, providing at the output the desired target modality.

The entire set of parameters in our network is $\Theta = \{\mathbf{W}^{(l)}, \mathbf{b}^{(l)}\}$, for all $1 \leq l \leq 4$, where l denotes the number of the layer, $\mathbf{W}^{(l)}$ is the weight matrix that connects two consecutive layers l and $l + 1$, and $\mathbf{b}^{(l)}$ denotes the bias term. These parameters are being optimized using the back-propagation algorithm [27].

Particularly, we use back-propagation to optimize the whole network with pre-trained weights, as described below, in order to improve the correspondence between the source and target modality. Moreover, the back-propagation will allow the layer to share information by capturing the non-linearity that the pre-training could not characterize. In order to be compatible with the pre-training phase, we used the sigmoid as activation function of the layers $L2$ - $L4$ (cf. Eq. (1)) namely,

$$\mathbf{a}^{l+1} = \sigma \left(\mathbf{W}^{(l)} \mathbf{a}^{(l)} + \mathbf{b}^{(l)} \right), \quad 1 \leq l \leq 3, \quad (1)$$

whereas for the output $L5$ we adopted the linear activation function (cf. Eq. (2)),

$$\hat{\mathbf{I}}_T = \mathbf{W}^{(4)} \mathbf{a}^{(4)} + \mathbf{b}^{(4)}. \quad (2)$$

Note that $\mathbf{a}^{(1)} \equiv \mathbf{I}_S$, specifically it is a slice of the source modality. These functions are differentiable and their derivative is known and easy to compute. The back-propagation error optimizes the parameters in Θ by minimizing the following cost function

$$J(\mathbf{I}_S, \mathbf{I}_T; \Theta) = \frac{1}{2} \|\hat{\mathbf{I}}_T - \mathbf{I}_T\|_2^2, \quad (3)$$

where $\hat{\mathbf{I}}_T$ is a function of the parameters Θ (feed-forward step).

2.2 Pre-training

Learning the mapping of two different modalities is a complex task. To assist our network architecture in optimizing the parameters, we initialize the weights with pre-trained ones. For this purpose, we pre-train the weights per-layer relying on the unsupervised learning power of Restricted Boltzmann Machine (RBM) [10], leveraging the tight relationship between Autoencoders and RBM [13]. Specifically, we adopted a *Gaussian-Bernoulli RBM* (GB-RBM) [5] to pre-train the weights and the bias connecting $L1$ and $L2$. Then, the output of this network is provided to a *Bernoulli-Bernoulli RBM* (BB-RBM) [9] that pre-trains the connections between $L2$ and $L3$. The same process is being followed for the layers $L4$ and $L5$, as well as for the layers $L3$ and $L4$, while here there are being used

as training images the ones of the target modality instead. Once the parameters in Θ are pre-trained, we perform the fine-tuning and we train the network with back-propagation as discussed previously.

2.3 Inference

At test time, we provide an image from the source modality to the layer $L1$ and we perform a single feed-forward step into the entire network. The activations on the layer $L5$ are the output of the network and represent the actual synthetic image in the target modality domain. This demonstrates the simplicity and elegance of this holistic approach to synthesis.

Retaining this network requires less memory than retaining a patch database. While training the network is computationally demanding this happens offline. Performing a feed-forward step at inference is significantly more efficient than the nearest neighbours being used in most patch-based approaches.

3 Experiments and Results

In this section, we evaluate the proposed deep neural network approach by generating T2 scans from T1 scans and DWI scans from T2. We follow evaluation settings and metrics that there were recently had been used in [29]. Furthermore, we compare with the patch-based method of [29].

Dataset: We used the SISS dataset from the ISLES 2015 workshop [17]. Specifically, we used the training dataset, which includes 28 subjects. For every subject, there are images of four modalities: T1, T2, VFlair, and DWI, of dimension approximately $230 \times 230 \times 150$. We rely on this dataset since it has been already preprocessed and the subjects are co-registered. If we had chosen another popular dataset. It would be necessary to perform various steps of pre-processing. The pre-processing could vary according to the implementation and the tools might had been used, eventually leading to potential bias and inability to compare directly among papers in the same area.

Model of Comparison: Modality Propagation (MP) [29] is a patch-based method for medical image synthesis, which is commonly used for comparisons, even in the most recent papers which use neural networks [23, 24]. This method comprises of a database with paired images of different modalities. Assuming fine alignment between the input image and those in the database, the synthesis of the target image is being made through patch matching nearest neighbor search. To reduce the search space within the database, the method uses techniques to reduce both the population and the area searched around a specific point location. Additionally, the method introduces an iterative regularization term that takes advantage of the produced synthetic image. We implemented this method de novo and we used the same parameters as the authors.

Preprocessing: The portion occupied by brain matter is less than the actual size of the image and this does not affect our learning. For a more efficient computation, we crop all the images using the biggest bounding box that encloses

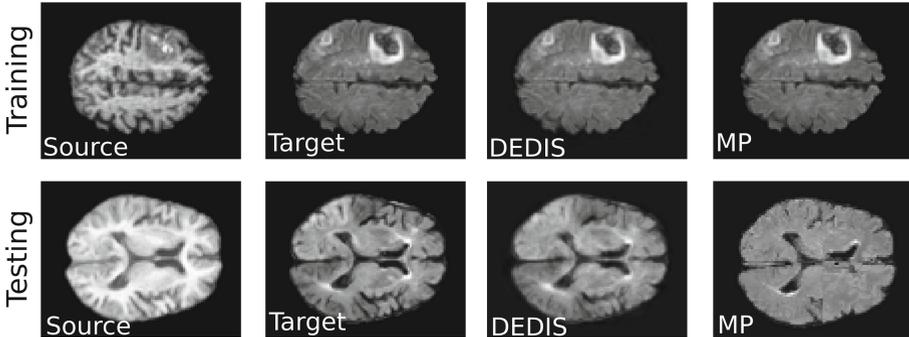


Fig. 2. Example of reconstruction with the proposed DEDIS network and Modality Propagation (MP) [29]. In this picture, we have a training (first row) and testing subject (second row). The first column shows the source modality images T1, the second column shows the ground-truth image in the target modality. Then, the third column shows the output of our DNN. The last column is the result from MP.

the area covered by the brain across all the subjects. Subsequently, plane slices have been rescaled by half. Even though our method is computationally fast, our implementation of Modality Propagation is not. Thus, to keep the comparison fair we used the same image sizes for both methods. After these operations, all the images have dimension 79×100 . Then, each image is individually normalized, by removing the mean and dividing by the standard deviation of the intensities within the image. To increase the data to train on and add some bilateral invariance, we augmented the dataset. The augmentation have been made by flipping all training data across the horizontal line, since the brain is almost symmetric for healthy subjects along the interhemispheric fissure, doubling the dataset size.

Experimental Setup: We selected the size of our network as multiples of the input layers, which corresponds to the number of pixels within one slice after preprocessing. The chosen architecture is summarized as follows: [7900, 2250, 1125, 2250, 7900]. Other configurations, e.g. over-complete setup scenarios, that is the layers in between ($L2$ and $L3$) were bigger than the input layer, induced redundant information and were prone to over-fitting. In the other hand, when we reduced the dimensions of the inner layers, we observed that the network had learned better representations.

We adopted 7-fold cross-validation, in which 24 subjects had been used for training, and the remaining 4 had been used for testing. The training, included only the slices containing brain matter. We iterated the pre-training of RBMs with 200 epochs, whereas the fine-tuning had been performed with 300 iterations. We built our network with the *Deep Learning Toolbox*² and thus our code had been based mostly in Matlab. We ran our experiments on an Intel Xeon 3.5 GHz CPU with a GeForce GTX Titan X GPU running in Debian.

² Freely available at <https://github.com/rasmusbergpalm/DeepLearnToolbox> [18]. We modified the current implementation to enable also GPU (CUDA) processing.

Table 1. Experimental results of our proposed method, when it is trained from modality T1 to VFlair, input and target modality respectively. We compared test results with Modality Propagation [29]. Values are *mean (std)*.

	DEDIS		Modality Propagation [29]	
	Training	Testing	Training	Testing
M.A.E	0.1261 (0.0558)	0.2400 (0.0490)	-	0.1196 (0.0467)
M.S.E	0.0696 (0.0863)	0.2212 (0.1226)	-	0.1501 (0.1151)
Norm. X-Corr	0.9652 (0.0454)	0.8886 (0.0652)	-	0.9292 (0.0559)

Table 2. Similar to Table 1 but synthesizing T2 from DWI.

	DEDIS		Modality Propagation [29]	
	Training	Testing	Training	Testing
M.A.E	0.0544 (0.0123)	0.2697 (0.2898)	-	0.1456 (0.0441)
M.S.E	0.0118 (0.0255)	0.2898 (0.1360)	-	0.2008 (0.1897)
Norm. X-Corr	0.9940 (0.0145)	0.8573 (0.0750)	-	0.9096 (0.0784)

Evaluation Metrics: As commonly done by other works in this area, we adopted three different metrics to quantitatively evaluate our method: (i) mean absolute error (M.A.E.), (ii) mean squared error (M.S.E.), and (iii) normalized cross-correlation (Norm. X-Corr), where for the first two the lower the better and for the third the higher the better.

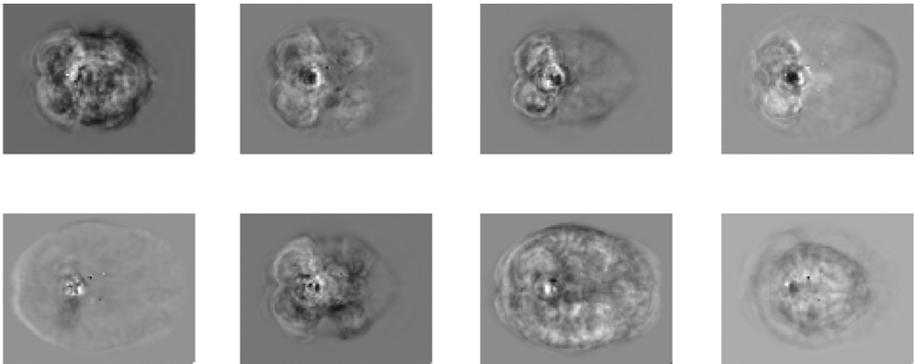


Fig. 3. The first 8 filters from the weight matrix $W^{(1)}$ connecting $L1$ and $L2$ of DEDIS after 300 iterations of fine-tuning.

Results and Discussion: In Fig. 2 we show a visual example of synthesized images using DEDIS, where images of T1 were used as source modality and as

target modality the corresponding VFlair contrast. In this example, we show images from the training and also testing, where both approaches (DEDIS and MP) have not seen the input image. In the last two columns we present estimated instances of DNN and the MP respectively. The first and second row show training and testing examples accordingly. The proposed method is able to preserve anatomical details (e.g., cortical folds), both in training and testing examples. This is particularly evident in the testing case: the MP was unable to reconstruct fully the lateral ventricle, most likely attributed to the large cell size we used. This demonstrates the benefit of using our whole image approach to synthesis. In fact, as Fig. 3 shows, filters learned by DEDIS, preserve both anatomical and contrast related information.

Quantitative results across the study population and the cross-validation are reported in Table 1, for VFlair synthesis with T1 as input, and in Table 2 for T2 synthesis given DWI as input. We show training and testing performance for DEDIS, as well as the testing error of MP [29]. As done in [23] we do not report the training performance of MP as it is not applicable. Whilst training results are above MP, at testing we are slightly lower. Perhaps our network still over-fits and strategies to mitigate that in the future could improve performance.

Critically though, we do gain in computational performance at inference. DEDIS takes 0.004 seconds per slice at test time³, orders of magnitude less than the 80 seconds required by MP. The implications of the k-NN search required by MP within the image database and the local patch-based search are evident. Note that the search time scales (albeit linearly) with the size of the database (or the window) and as such the more the images in the database the more the computational time is in demand. In contrast, our approach synthesizes the whole image while at the same time, since it is independent of database size for inference.

Similar argument holds for memory size requirements. Our network at these settings occupies 300 MB in memory, whilst MP 800 MB, almost 3-fold more. As database size increases, e.g. the number of training images increases, memory requirements for MP increase, whereas ours remains the same.

4 Conclusion

In this paper, we introduced a Deep Neural Network that can learn to synthesise a modality. Our network is optimized via back-propagation and it needs a set of data belonging to the input and target modality, as it is illustrated in Fig. 1. We pre-trained the network using Restricted Boltzmann Machines, which learned the pair-wise weight matrices that were fine-tuned at later stage. Example-based methods are expensive both in time and memory resources. Instead our

³ We use only the CPU and not GPU to permit fair comparison with our MP implementation which does not use GPU.

approach, which we termed DEDIS, synthesizes whole images treating the problem as a multi-output regression. Overall, we show that our method gives comparable results with a patch-based method (Modality Propagation) [29] when trained on a preprocessed dataset. But our method is almost 1000 times faster. Being fast is important when for example the synthesis method will be used within a data imputation pipeline of a very large database (e.g., biobank).

Relying on the advantages of the proposed network, the future orientation to explore is the one enabling DEDIS architecture to synthesize a whole volume at once, instead of slice by slice.

Acknowledgement. We thank NVIDIA Inc. for providing us with a Titan X GPU used for our experiments.

References

1. Alexander, D.C., Zikic, D., Zhang, J., Zhang, H., Criminisi, A.: Image quality transfer via random forest regression: applications in diffusion MRI. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014, Part III. LNCS, vol. 8675, pp. 225–232. Springer, Heidelberg (2014)
2. Artaechevarria, X., Munoz-Barrutia, A., Ortiz-de-Solórzano, C.: Combination strategies in multi-atlas image segmentation: application to brain MR data. *IEEE Trans. Med. Imaging* **28**(8), 1266–1277 (2009)
3. Burgos, N., Cardoso, M.J., Thielemans, K., Modat, M., Pedemonte, S., Dickson, J., Barnes, A., Ahmed, R., Mahoney, C.J., Schott, J.M., et al.: Attenuation correction synthesis for hybrid PET-MR scanners: application to brain studies. *IEEE Trans. Med. Imaging* **33**(12), 2332–2341 (2014)
4. Cardoso, M.J., Sudre, C.H., Modat, M., Ourselin, S.: Template-based multimodal joint generative model of brain data. In: Ourselin, S., Alexander, D.C., Westin, C.-F., Cardoso, M.J. (eds.) IPMI 2015. LNCS, vol. 9123, pp. 17–29. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-19992-4_2](https://doi.org/10.1007/978-3-319-19992-4_2)
5. Cho, K.H., Ilin, A., Raiko, T.: Improved learning of Gaussian-Bernoulli restricted Boltzmann machines. In: Honkela, T., Duch, W., Girolami, M., Kaski, S. (eds.) ICANN 2011. LNCS, vol. 6791, pp. 10–17. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-21735-7_2](https://doi.org/10.1007/978-3-642-21735-7_2)
6. Fischl, B., Salat, D.H., van der Kouwe, A.J., Makris, N., Ségonne, F., Quinn, B.T., Dale, A.M.: Sequence-independent segmentation of magnetic resonance images. *Neuroimage* **23**, S69–S84 (2004)
7. Guimond, A., Roche, A., Ayache, N., Meunier, J.: Three-dimensional multimodal brain warping using the demons algorithm and adaptive intensity corrections. *IEEE Trans. Med. Imaging* **20**(1), 58–69 (2001)
8. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, pp. 327–340. ACM (2001)
9. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. *Neural Comput.* **14**(8), 1771–1800 (2002)
10. Hinton, G.E., Osindero, S., Teh, Y.W.: A fast learning algorithm for deep belief nets. *Neural Comput.* **18**(7), 1527–1554 (2006)

11. Iglesias, J.E., Konukoglu, E., Zikic, D., Glocker, B., Van Leemput, K., Fischl, B.: Is synthesizing MRI contrast useful for inter-modality analysis? In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013, Part I. LNCS, vol. 8149, pp. 631–638. Springer, Heidelberg (2013)
12. Jog, A., Roy, S., Carass, A., Prince, J.L.: Magnetic resonance image synthesis through patch regression. In: IEEE 10th ISBI, pp. 350–353. IEEE (2013)
13. Kamyshanska, H., Memisevic, R.: The potential energy of an autoencoder. IEEE Trans. PAMI **37**(6), 1261–1273 (2015)
14. Konukoglu, E., van der Kouwe, A., Sabuncu, M.R., Fischl, B.: Example-based restoration of high-resolution magnetic resonance image acquisitions. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013, Part I. LNCS, vol. 8149, pp. 131–138. Springer, Heidelberg (2013)
15. Kroon, D.J., Slump, C.H.: MRI modality transformation in demon registration. In: IEEE International Symposium on Biomedical Imaging: From Nano to Macro, ISBI 2009, pp. 963–966. IEEE (2009)
16. Larochelle, H., Erhan, D., Courville, A., Bergstra, J., Bengio, Y.: An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th ICML, pp. 473–480 (2007)
17. Maier, O., Wilms, M., von der Gablentz, J., Krämer, U.M., Münte, T.F., Handels, H.: Extra tree forests for sub-acute ischemic stroke lesion segmentation in MR sequences. J. Neurosci. Methods **240**, 89–100 (2015)
18. Palm, R.B.: Prediction as a candidate for learning deep hierarchical models of data. Master’s thesis (2012)
19. Rohlfing, T., Russakoff, D.B., Maurer, C.R.: Expectation maximization strategies for multi-atlas multi-label segmentation. In: Taylor, C.J., Noble, J.A. (eds.) IPMI 2003. LNCS, vol. 2732, pp. 210–221. Springer, Heidelberg (2003)
20. Rousseau, F.: Brain hallucination. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 497–508. Springer, Heidelberg (2008)
21. Roy, S., Carass, A., Prince, J.: A compressed sensing approach for MR tissue contrast synthesis. In: Székely, G., Hahn, H.K. (eds.) IPMI 2011. LNCS, vol. 6801, pp. 371–383. Springer, Heidelberg (2011)
22. Tulder, G., Bruijne, M.: Why does synthesized data improve multi-sequence classification? In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9349, pp. 531–538. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-24553-9_65](https://doi.org/10.1007/978-3-319-24553-9_65)
23. Nguyen, H., Zhou, K., Vemulapalli, R.: Cross-domain synthesis of medical images using efficient location-sensitive deep network. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9349, pp. 677–684. Springer, Heidelberg (2015). doi:[10.1007/978-3-319-24553-9_83](https://doi.org/10.1007/978-3-319-24553-9_83)
24. Vemulapalli, R., Van Nguyen, H., Zhou, S.K.: Unsupervised cross-modal synthesis of subject-specific scans. In: Proceedings of the IEEE ICCV, pp. 630–638 (2015)
25. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res. **11**, 3371–3408 (2010)
26. Wein, W., Brunke, S., Khamene, A., Callstrom, M.R., Navab, N.: Automatic CT-ultrasound registration for diagnostic imaging and image-guided intervention. Med. Image Anal. **12**(5), 577–585 (2008)
27. Williams, D., Hinton, G.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986)

28. Wolz, R., Chu, C., Misawa, K., Mori, K., Rueckert, D.: Multi-organ abdominal CT segmentation using hierarchically weighted subject-specific atlases. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) MICCAI 2012, Part I. LNCS, vol. 7510, pp. 10–17. Springer, Heidelberg (2012)
29. Ye, D.H., Zikic, D., Glocker, B., Criminisi, A., Konukoglu, E.: Modality propagation: coherent synthesis of subject-specific scans with data-driven regularization. In: Mori, K., Sakuma, I., Sato, Y., Barillot, C., Navab, N. (eds.) MICCAI 2013, Part I. LNCS, vol. 8149, pp. 606–613. Springer, Heidelberg (2013)