CHAPTER

16 Deep Networks and Mutual Information Maximization for Cross-Modal Medical Image Synthesis

Raviteja Vemulapalli[*], **Hien Van Nguyen**[†], **S. Kevin Zhou**[‡]

University of Maryland, College Park, MD, United States[*] *Uber Advanced Technology Center, Pittsburgh, PA, United States*[†] *Siemens Healthineers Technology Center, Princeton, NJ, United States*[‡]

CHAPTER OUTLINE

Deep Learning for Medical Image Analysis. DOI: 10.1016/B978-0-12-810408-8.00022-5

CHAPTER POINTS

- Location-sensitive deep network integrates spatial information with intensity-based features by modeling the first hidden layer nodes as products of certain location-sensitive functions and nonlinear features computed from voxel intensity values
- The mutual information maximization-based approach can be used for synthesis in the unsupervised setting by combining it with cross-modal nearest neighbor search
- The presented approaches show promising synthesis capabilities when evaluated by synthesizing MR contrast

16.1 INTRODUCTION

Currently, a multitude of imaging modalities such as Computed Tomography, X-ray, Magnetic Resonance Imaging (MRI), Ultrasound, etc., are being used for medical imaging research and clinical diagnosis. Each of these modalities has different contrast and noise mechanisms and hence captures different characteristics of the underlying anatomy. The intensity transformation between any two modalities is highly nonlinear.

Due to variations in the image characteristics across modalities, medical image analysis algorithms trained with data from one modality may not work well when applied to data from a different modality. A straightforward way to address this issue is to collect a large amount of training data in each modality. But, this is impractical since collecting medical images is time consuming, expensive, and involves exposing patients to radiation. Hence, it is highly desirable to have a general cross-modal synthesis approach that generates subject-specific scans in the desired target modality from the given source modality images. The ability to synthesize medical images without real acquisitions has various potential applications such as super-resolution [10,21], atlas construction [5], multimodal registration [4,18,19], segmentation [9,20], and virtual enhancement [15].

Due to its potential applications, cross-modal synthesis has received significant attention from the medical imaging community in the recent past, and various approaches have been proposed for this task. In [12], a set of paired low/high resolution images was used to learn the joint probability distribution of low/high resolution patches. This probability distribution was then used for super-resolution. A regression-based synthesis approach was proposed in [11] using random forest. In [4], a cross-modal image synthesis approach was proposed based on coupled sparse representation. While training, this approach used paired data from source and target modalities to learn coupled dictionaries that establish cross-modal correspondences. These learned dictionaries were then used for sparse coding-based image synthesis. Similar sparse coding-based approaches have also been used in [3, 19,21,24] for image synthesis applications. In [20], a codebook of paired training patches was used for MR contrast synthesis. For each test patch, few best matches

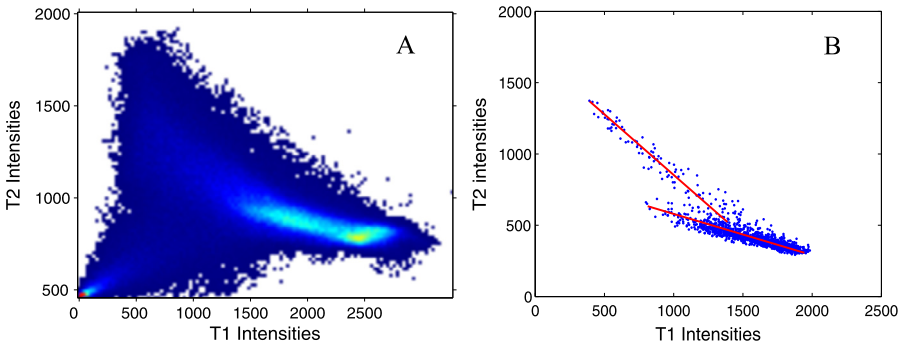

FIGURE 16.1

(A) Intensity correspondences between T1-MRI and T2-MRI over an entire image. Brighter color indicates higher density region. (B) Intensity correspondences over a region of $10 \times 10 \times 10$ voxels. Red lines indicate the main directions of variation. Images are registered using rigid transformations.

(among the source modality patches) were found from the codebook and the target patches corresponding to the best matches were averaged to generate target MR contrast. A similar approach based on image analogies [7] was also used in [9]. A label propagation [17] based iterative synthesis approach, called modality propagation, was proposed in [27]. Alternative to cross-modal medical image synthesis approaches, compressed sensing-based methods such as magnetic resonance fingerprinting [13] have also been proposed in the recent past to reduce scanning time and cost.

Most of the existing synthesis approaches either do not use the spatial information or use it in the form of a hard constraint. For example, modality propagation [27] restricts the nearest neighbor search to a small window around the voxel's location to incorporate the spatial information. To see the importance of spatial information, we plot the T1–T2 MRI intensity correspondences in Fig. 16.1A using the (registered) brain scans of a subject from the NAMIC multimodality database. We can notice that the intensity transformation is not only nonlinear but also far from unique, i.e., there are several feasible intensity values in T2-MRI modality for one intensity value in T1-MRI modality, and vice versa. This non-uniqueness comes from the fact that the intensity transformation depends on the region in which voxels reside. If we restrict the region of interest to a local neighborhood, say of size $10 \times 10 \times 10$ voxels, the intensity transformation becomes much simpler as shown in Fig. 16.1B. This shows that spatial information helps in simplifying the transformations between modalities, which in turn enables more accurate predictions. Unfortunately, most of the existing approaches do not have a systematic way to incorporate spatial information.

Section 16.2 of this chapter presents a novel deep network architecture, referred to as location-sensitive deep network (LSDN), that integrates image intensity-based

features and spatial information in a principled manner. This network, which was initially proposed in [14], models the first hidden layer nodes as products of feature responses computed from voxel intensity values and certain location-sensitive functions. As a result, LSDN is able to capture the joint distribution of intensity features and spatial information. In addition, Section 16.2 also presents a network simplification method for reducing the LSDN synthesis time while maintaining the prediction accuracy.

All the above mentioned synthesis approaches work under the supervised setting, i.e., they require training data in both source and target modalities from the same set of subjects. The availability of such paired data is limited in many cases. Also, collecting multiple scans from each subject is not desirable especially when patients are exposed to radiation. Hence, there is a need for a cross-modal synthesis approach that works in the unsupervised setting, i.e., when paired training data is not available.

Section 16.3 of this chapter presents an approach based on cross-modal nearest neighbor search and mutual information maximization, which can be used in the unsupervised setting. Given a source modality image, this approach first generates multiple target modality candidate values independently for each voxel by performing cross-modal nearest neighbor search over the training database. Then, the best candidate values are selected jointly for all the voxels by simultaneously maximizing a global mutual information-based cost function and a local spatial consistency cost function. To the best of our knowledge, this approach, which was initially proposed in [23], is the first approach that addresses the cross-modal image synthesis problem in an unsupervised setting. This approach can also be used in the supervised setting by replacing the cross-modal nearest neighbor search with source-modal nearest neighbor search.

Notations. Matrices are denoted using boldface capital letters and vectors are denoted using boldface small letters. The ℓ_0 and ℓ_2 norms of a vector $\boldsymbol{v}$ are denoted by $\|\boldsymbol{v}\|_0$ and $\|\boldsymbol{v}\|_2$, respectively. We use $\odot$ to denote the Hadamard product. The transpose and Frobenius norm of a matrix $\boldsymbol{A}$ are denoted by $\boldsymbol{A}^\top$ and $\|\boldsymbol{A}\|_F$, respectively. The ith column of a matrix $\boldsymbol{A}$ is denoted using $\boldsymbol{A}(:, i)$. We use $\mathbb{I}$ to denote the indicator function and P to denote probability.

16.2 SUPERVISED SYNTHESIS USING LOCATION-SENSITIVE DEEP NETWORK

Cross-modal image synthesis can be seen as a regression problem with the image in source modality as input and the image in target modality as output. In this section, we will use a neural network as the regressor. It is ineffective to train a network that operates on the entire image since the number of parameters would be too large for the network to generalize well. Hence, we will use a network that takes a source modality image patch as input and predicts the target modality intensity

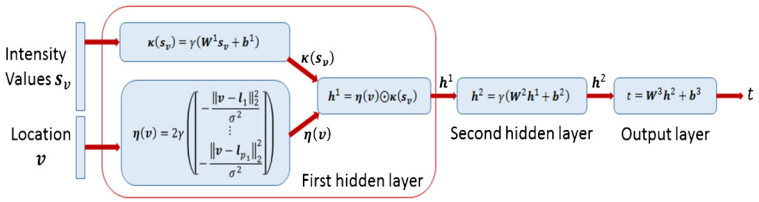

FIGURE 16.2

Location-sensitive deep network with two hidden layers.

value for the center voxel of the patch. Note that the synthesized target modality image will be produced in the same coordinate system as the given source modality image.

Let $\boldsymbol{s_v}$ represent the vectorized version of a $d \times d \times d$ patch centered on voxel $\boldsymbol{v} = (x, y, z)$ in the source modality input image. As discussed earlier (Fig. 16.1), the intensity transformation between modalities depends on voxel's spatial location.[1] Hence, the neural network regressor should take the voxel location $\boldsymbol{v}$ as an input along with the intensity features $\boldsymbol{s_v}$. One straightforward way to do this is to concatenate $\boldsymbol{s_v}$ and $\boldsymbol{v}$ into a single vector and train a standard multilayer perceptron (MLP) with the concatenated vector as input. However, this strategy does not work well (as shown later in Section 16.2.3) since the intensity features and spatial location are two heterogeneous sources of information.

In the section, we present a new network architecture, referred to as location-sensitive deep network, which combines the intensity input $\boldsymbol{s_v}$ and the spatial location $\boldsymbol{v}$ in a principled manner. This network is similar to an MLP with sigmoid nonlinearity except that the first hidden layer nodes are modeled as products of feature responses computed from the intensity input $\boldsymbol{s_v}$ and certain spatial response functions. Fig. 16.2 shows an LSDN with two hidden layers. The ith node in the first hidden layer performs the following computations:

$$\begin{aligned} h_i^1 &= \eta_i(\boldsymbol{v})\kappa_i(\boldsymbol{s_v}), \quad \eta_i(\boldsymbol{v}) = 2\gamma\left(-\frac{\|\boldsymbol{v}-\boldsymbol{l}_i\|_2^2}{\sigma^2}\right), \\ \kappa_i(\boldsymbol{s_v}) &= \gamma\left(\boldsymbol{w}_i^{1\top}\boldsymbol{s_v} + b_i^1\right). \end{aligned} \tag{16.1}$$

Here, γ denotes the standard sigmoid function, $\eta_i(\boldsymbol{v})$ is a spatial response function parametrized by a location $\boldsymbol{l}_i$ which will be learned during training and a constant σ (we could also use a different σ for each node and learn them during training), and $\kappa_i(\boldsymbol{s_v})$ is an intensity-based feature response function parametrized by a linear filter $\boldsymbol{w}_i^1$ and bias b_i^1 which will be learned during training. The computations performed

by the entire first hidden layer with p_1 nodes can be written as

$$\boldsymbol{h}^1 = \begin{bmatrix} h_1^1 \\ \vdots \\ h_{p_1}^1 \end{bmatrix} = \boldsymbol{\eta}(\boldsymbol{v}) \odot \boldsymbol{\kappa}(\boldsymbol{s_v}), \quad \boldsymbol{\eta}(\boldsymbol{v}) = \begin{bmatrix} \eta_1(\boldsymbol{v}) \\ \vdots \\ \eta_{p_1}(\boldsymbol{v}) \end{bmatrix} = 2\gamma \left(\begin{bmatrix} -\frac{\|\boldsymbol{v}-\boldsymbol{l}_1\|_2^2}{\sigma^2} \\ \vdots \\ -\frac{\|\boldsymbol{v}-\boldsymbol{l}_{p_1}\|_2^2}{\sigma^2} \end{bmatrix} \right),$$
$$\boldsymbol{\kappa}(\boldsymbol{s_v}) = \begin{bmatrix} \kappa_1(\boldsymbol{s_v}) \\ \vdots \\ \kappa_{p_1}(\boldsymbol{s_v}) \end{bmatrix} = \gamma \left(\boldsymbol{W}^1 \boldsymbol{s_v} + \boldsymbol{b}^1 \right), \quad \boldsymbol{W}^1 = \begin{bmatrix} \boldsymbol{w}_1^{1\top} \\ \vdots \\ \boldsymbol{w}_{p_1}^{1\top} \end{bmatrix}, \quad \boldsymbol{b}^1 = \begin{bmatrix} b_1^1 \\ \vdots \\ b_{p_1}^1 \end{bmatrix}. \tag{16.2}$$

All the other hidden layers of an LSDN are similar to the hidden layers of a standard MLP, i.e., they perform the following computations:

$$\boldsymbol{h}^k = \gamma \left(\boldsymbol{W}^k \boldsymbol{h}^{k-1} + \boldsymbol{b}^k \right). \tag{16.3}$$

The output layer of an LSDN with K layers (i.e., $K-1$ hidden layers) computes the target modality intensity value of the center voxel of the input patch using

$$t = \boldsymbol{W}^K \boldsymbol{h}^{K-1} + \boldsymbol{b}^K. \tag{16.4}$$

Note that the ith node of first hidden layer will be active only when the voxel location $\boldsymbol{v}$ is close enough to the location $\boldsymbol{l}_i$. Hence, based on the voxel location $\boldsymbol{v}$ some of the first hidden layer nodes will be active and the others will not be. Different combinations of on/off nodes effectively create multiple sub-networks, each of which is tuned to a small spatial region in the image. This novel property is the main advantage of an LSDN compared to a standard MLP. Recall the observation from Fig. 16.1 that the input–output mapping becomes simpler when restricted to a smaller spatial region. Therefore, LSDN has the potential to yield more accurate predictions.

To ensure that the spatial location $\boldsymbol{v}$ conveys meaningful information, training and test images are registered to a reference image using rigid transformations. After the registration, the same voxel location in different images corresponds to roughly the same anatomical region. Alternatively, one could eliminate the need for registration by using relative coordinates with respect to some landmarks.

16.2.1 BACKPROPAGATION

Similar to an MLP, an LSDN can also be trained using stochastic gradient descent by computing the gradients using standard backpropagation. In this section, we show how to backpropagate the error derivatives through the first hidden layer of an LSDN. We skip the derivative formulas corresponding to the subsequent layers since they are standard MLP layers.

The derivatives $dE/d\boldsymbol{h}^1$ of the error function E with respect to the first hidden layer outputs $\boldsymbol{h}^1$ can be computed by backpropagating the error derivatives from the output layer till the second hidden layer. Given these derivatives, we can compute the derivatives of E with respect to $\boldsymbol{\eta}(\boldsymbol{v})$ and $\boldsymbol{\kappa}(\boldsymbol{s_v})$ using

$$\frac{dE}{d\boldsymbol{\eta}(\boldsymbol{v})} = \frac{dE}{d\boldsymbol{h}^1} \odot \boldsymbol{\kappa}(\boldsymbol{s_v}), \qquad \frac{dE}{d\boldsymbol{\kappa}(\boldsymbol{s_v})} = \frac{dE}{d\boldsymbol{h}^1} \odot \boldsymbol{\eta}(\boldsymbol{v}). \tag{16.5}$$

Given the derivatives $dE/d\boldsymbol{\eta}(\boldsymbol{v})$, we can compute the derivatives of E with respect to the network location parameters $\boldsymbol{l}_i$ using

$$\frac{dE}{d\boldsymbol{l}_i} = \frac{4}{\sigma^2}\gamma\left(-\frac{\|\boldsymbol{v}-\boldsymbol{l}_i\|_2^2}{\sigma^2}\right)\left(1-\gamma\left(-\frac{\|\boldsymbol{v}-\boldsymbol{l}_i\|_2^2}{\sigma^2}\right)\right)\frac{dE}{d\boldsymbol{\eta}_i(\boldsymbol{v})}\left(\boldsymbol{v}-\boldsymbol{l}_i\right). \tag{16.6}$$

Given the derivatives $dE/d\boldsymbol{\kappa}(\boldsymbol{s_v})$, the derivatives $dE/d\boldsymbol{W}^1$ and $dE/d\boldsymbol{b}^1$ can be computed by backpropagating through the computation of $\boldsymbol{\kappa}(\boldsymbol{s_v})$ which is nothing but a standard MLP layer.

16.2.2 NETWORK SIMPLIFICATION

Applying LSDN on every $d \times d \times d$ patch during the synthesis process could be computationally expensive since medical images usually contain hundreds of thousands of voxels. In this section, we present a post-processing method for simplifying the network in order to improve the speed of LSDN without losing much in terms of prediction accuracy. At each hidden layer of the network, this method tries to find a small subset of features/nodes that can be used to reconstruct the entire layer approximately.

Let $\mathcal{I}^k$ denote the index set of such a subset at the kth hidden layer. Then, we have

$$h_{i,n}^k \approx \sum_{j\in\mathcal{I}^k} \alpha_{ij}^k h_{j,n}^k, \quad i \in \{1,2,\ldots,p_k\}, \;\; n \in \{1,2,\ldots,N\}, \tag{16.7}$$

where p_k is the number of nodes in the kth hidden layer, N is the number of training samples, $h_{i,n}^k$ is the response of the ith node in the kth hidden layer for the nth training sample, and α_{ij}^k are the reconstruction coefficients. The index set $\mathcal{I}^k$ and coefficients α_{ij}^k can be obtained by solving the following optimization problem:

$$\underset{\boldsymbol{A}^k}{\text{minimize}} \;\; \|\boldsymbol{H}^k - \boldsymbol{A}^k\boldsymbol{H}^k\|_F^2, \quad \text{subject to} \;\; \|\boldsymbol{A}^k\|_{col-0} \le T^k, \tag{16.8}$$

$$\boldsymbol{A}_{ij}^k = \begin{cases} \alpha_{ij}^k, & \text{if } j \in \mathcal{I}^k, \\ 0, & \text{otherwise,} \end{cases} \qquad \boldsymbol{H}^k = \begin{pmatrix} h_{1,1}^k & \cdots & h_{1,N}^k \\ \vdots & \ddots & \vdots \\ h_{p_k,1}^k & \cdots & h_{p_k,N}^k \end{pmatrix}. \tag{16.9}$$

By constraining the quasi-norm $\|\mathbf{A}^k\|_{col-0}$, which is the number of nonzero columns, to be less than a certain threshold T^k, the above optimization problem identifies a small subset of nodes that can approximately reconstruct the entire kth layer. Since the formulation in (16.8) is a special case of simultaneous sparsity, simultaneous orthogonal matching pursuit [22] can be used to efficiently minimize the cost function. The index set $\mathcal{I}^k$ can be obtained from the indices of nonzero columns in $\boldsymbol{A}^k$.

Once we have the index sets $\mathcal{I}^k$ and the corresponding reconstruction coefficients, we can simplify the LSDN by shrinking the number of connections in each layer, also referred to as *ShrinkConnect* in the rest of this chapter. This can be done by removing the hidden layer nodes whose indices are not in the index sets $\mathcal{I}^k$. Let $\boldsymbol{h}^k_{\mathcal{I}^k}$ represent the kth hidden layer in the simplified network. Note that the output of the kth hidden layer is the input to the $(k+1)$th hidden layer. Hence, when we remove nodes from the kth hidden layer, we need to update $\boldsymbol{W}^{k+1}$ such that the outputs of applying the new filters on $\boldsymbol{h}^k_{\mathcal{I}^k}$ are approximately equal to the original LSDN outputs $\boldsymbol{W}^{k+1}_{row\in\mathcal{I}^{k+1}}\boldsymbol{h}^k$. Using the approximation in (16.8), the entire ShrinkConnect operation can be described by the following update rule:

$$\boldsymbol{W}^{k+1} \leftarrow \boldsymbol{W}^{k+1}_{\text{row}\in\mathcal{I}^{k+1}}\boldsymbol{A}^k_{\text{column}\in\mathcal{I}^k}, \qquad \boldsymbol{b}^{k+1} \leftarrow \boldsymbol{b}^{k+1}_{\text{ind}\in\mathcal{I}^{k+1}}, \tag{16.10}$$

where $\boldsymbol{W}^{k+1}_{\text{row}\in\mathcal{I}^{k+1}}$ is the matrix formed by the rows of $\boldsymbol{W}^{k+1}$ whose indices are in $\mathcal{I}^{k+1}$, $\boldsymbol{A}^k_{\text{column}\in\mathcal{I}^k}$ is the matrix formed by the columns of $\boldsymbol{A}^k$ whose indices are in $\mathcal{I}^k$, and $\boldsymbol{b}^{k+1}_{\text{ind}\in\mathcal{I}^{k+1}}$ is a vector formed by the entries of $\boldsymbol{b}^{k+1}$ whose indices are in $\mathcal{I}^{k+1}$.

16.2.3 EXPERIMENTS

In this section, we evaluate the LSDN architecture by generating T1-MRI scans from T2-MRI scans, and vice versa.

Dataset and Pre-Processing. We used the T1 and T2 MRI scans of 19 subjects from the NAMIC brain multimodality database (http://hdl.handle.net/1926/1687). Along with the MRI scans, this database also provides brain masks for skull-stripping. Following [27], all the images were skull stripped, linearly registered, inhomogeneity corrected, histogram matched within each modality, and resampled to 2 mm resolution. For registration, we used the first subject as reference. First, the T2 scan of the reference subject was registered to the corresponding T1 scan, and then the T1 and T2 scans of the remaining subjects were registered to the reference subject. We used 3D Slicer (www.slicer.org) software for data pre-processing.

Evaluation Setting and Metric. We used leave-one-out cross-validation setting, in which 18 image pairs were used for training and the remaining one was used for testing. Since the dataset has both T1 and T2 MRI scans for each subject, we directly compare the synthesized and ground truth target modality scans for evaluation. We use signal-to-noise ratio (SNR) as evaluation metric.

Parameters. The LSDN input patch size d and the spatial response function parameter σ were chosen as 3 and 0.5, respectively. When $d = 3$, the input to LSDN is

Table 16.1 SNR values and computation times for various different approaches

Approach	SNR (T1→T2) (dB)	SNR (T2→T1) (dB)	Training (h)	Synthesis (s)
MP [27]	13.64 ± 0.67	15.13 ± 0.88	n/a	928
CSR [4]	13.72 ± 0.64	15.24 ± 0.85	2.8	245
VDN	12.67 ± 0.6	14.19 ± 0.82	1.2	23.5
CDN	13.79 ± 0.68	15.36 ± 0.88	1.2	23.6
LSDN-small	12.53 ± 0.75	13.85 ± 0.86	0.6	9.2
LSDN-1	**14.82 ± 0.72**	**17.09 ± 0.94**	**1.4**	**29.5**
LSDN-2	**14.93 ± 0.73**	**17.39 ± 0.91**	**2.5**	**68.0**
LSDN-1+ShrinkConnect	**14.79 ± 0.72**	**17.05 ± 0.91**	**1.4**	**9.2**
LSDN-2+ShrinkConnect	**14.80 ± 0.74**	**17.1 ± 0.86**	**2.5**	**21.5**

30-dimensional (27 intensity values and 3 spatial coordinates). We investigated three network configurations denoted LSDN-small, LSDN-1, and LSDN-2, whose sizes are [30-50-5-1], [30-200-20-1], and [30-400-40-1], respectively. In ShrinkConnect, the threshold T^k for each hidden layer was set to one-fourth of the layer's original size.

Training. For training, we randomly sampled around one million patches from the training images. We trained the networks using stochastic gradient descent for 300 epochs with mean squared error as loss function. We used an initial learning rate of 0.25 and slowly decreased the learning rate by multiplying it with 0.99 after each iteration. After ShrinkConnect, we fine-tuned the simplified networks for 10 epochs.

Testing. To synthesize a target modality image, LSDN was applied to the source modality image in a sliding window fashion.

Comparisons. We compare LSDN with the following approaches:

- Vanilla deep network (VDN). Standard MLP of size [27-400-40-1] that takes only the intensity values as input.
- Concatenation deep network (CDN). Standard MLP of size [30-400-40-1] that takes both intensity values and spatial coordinates as input.
- Modality propagation (MP) [27] and coupled sparse representation (CSR) [4].

Results. Table 16.1 shows the average SNR values for various different approaches. As we can see, the SNR values for LSDN-1 and LSDN-2 are higher when compared to all the other approaches. It is interesting to see that synthesizing T1 from T2 produces much better results compared to synthesizing T2 from T1. We conjecture that more details of the brain are visible under T2 than T1. As expected, VDN which uses only the intensity values as input performs poorly. While CDN which uses both intensity values and spatial location as input performs better than VDN, it still performs poorly compared to LSDN-2 which has the same size as CDN. This clearly shows that standard MLPs are not appropriate for fusing intensity values and spatial

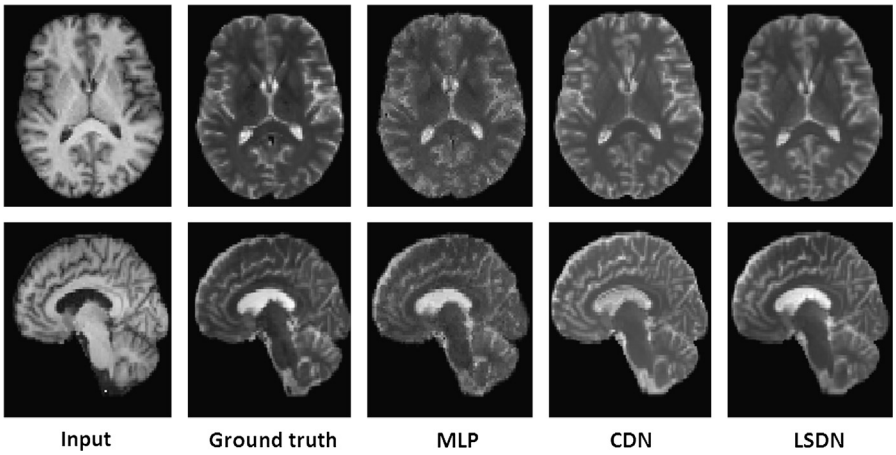

FIGURE 16.3

Comparison of various supervised synthesis approaches. T2-MRI synthesis from T1-MRI.

locations. ShrinkConnect reduces the size of LSDN-1 to [30-50-5-1] and the size of LSDN-2 to [30-100-10-1] without losing much in terms of prediction accuracy (refer to the last two rows of Table 16.1). Note that the size of LSDN-small is same as the size of (LSDN-1 + ShrinkConnect). The poor performance of LSDN-small shows that training a bigger network first and then reducing its size using ShrinkConnect is more effective than directly training the smaller network. The results of the LSDN networks also indicate that increasing the network size would improve the prediction at the cost of computation time. Figs. 16.3 and 16.4 show some visual examples comparing various different approaches.

Table 16.1 also provides the training and synthesis times for different approaches. The experiments were run on a 20-core machine with Intel X5650 processor using MATLAB implementation. The average time LSDN-1 takes to synthesize an image is 29.5 s. With ShrinkConnect, the synthesis time is reduced to 9.2 s per image, which is 26× faster than CSR and 100× faster than modality propagation.

16.3 UNSUPERVISED SYNTHESIS USING MUTUAL INFORMATION MAXIMIZATION

The LSDN-based approach presented in the above section is a supervised synthesis approach as it uses paired training data from source and target modalities to learn the network parameters. In this section, we present an unsupervised cross-modal medical image synthesis approach that works without paired data. Given a source modality image of a subject, this approach synthesizes the corresponding target modality image by making use of target modality images from a different set of subjects.

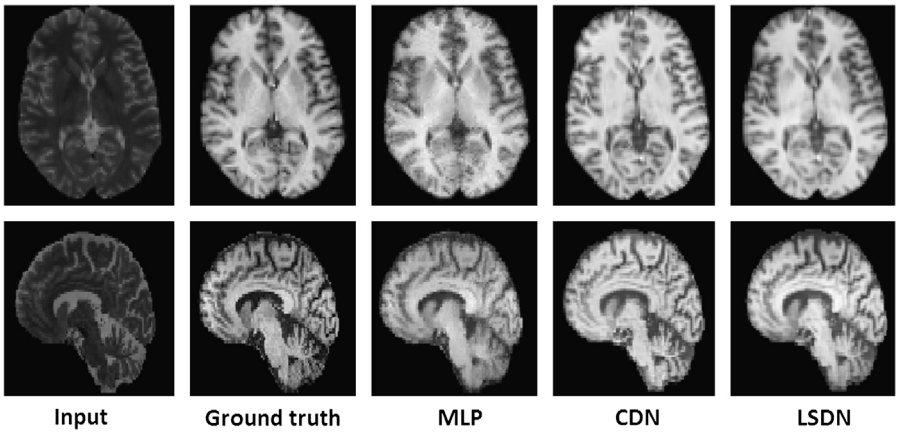

FIGURE 16.4

Comparison of various supervised synthesis approaches. T1-MRI synthesis from T2-MRI.

Synthesizing an image with D voxels can be seen as estimating a D-dimensional quantity. The following are two possible (extreme) strategies for solving this problem: (i) Estimating the intensities of all the voxels jointly, (ii) Estimating the intensity of each voxel independently. Each strategy has its own advantages and disadvantages. While the first one takes interactions between voxels into account, it is fairly complex given the large set of possible values for each voxel. While the second one simplifies the problem by considering each voxel independently, it does not take into account the image level context.

The presented approach takes advantage of both these strategies by following a two-step synthesis strategy. In the first step, multiple target modality candidate values are generated for each voxel independently. In the second step, a full target modality image is synthesized by selecting the best candidate values jointly for all the voxels by taking the image level context into account. The rationale behind generating multiple candidates in the first step is that at least one of the top K candidates would be an appropriate value for synthesis when the image context is considered in the second step. This can also be interpreted as restricting the set of possible intensity values for each voxel so that the joint estimation step becomes more tractable.

Since this approach works without paired training data, the quality of the synthesized images would be usually low when compared to the supervised approaches. These results could be improved further by using coupled sparse representation as a refinement step. Recently, various CSR-based approaches have been successfully used for synthesis in the supervised setting [3,4,8,19]. Fig. 16.5 shows the block diagram of the overall synthesis process.

Notations. Let $\Phi^{\boldsymbol{v}}$ denote the set consisting of voxel $\boldsymbol{v}$ and its six neighbors that are at unit distance from $\boldsymbol{v}$. We use the notation $\Phi^{\boldsymbol{v}}(p, q, r)$ to represent the elements of $\Phi^{\boldsymbol{v}}$.

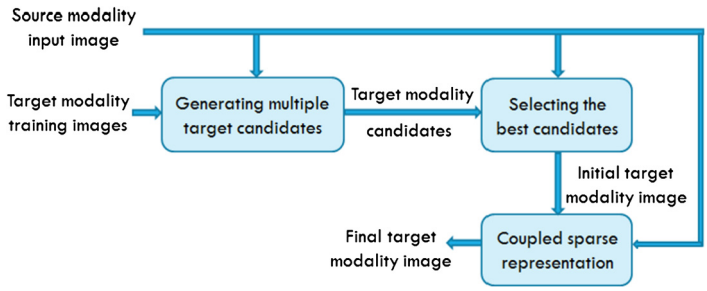

FIGURE 16.5

Unsupervised cross-modal image synthesis approach.

Here, $\Phi^{\boldsymbol{v}}(p, q, r)$ refers to the voxel $(\boldsymbol{v} + (p, q, r))$. We use the notation $\boldsymbol{v} \backsim \boldsymbol{v}'$ to indicate that voxels $\boldsymbol{v}$ and $\boldsymbol{v}'$ are neighbors.

16.3.1 GENERATING MULTIPLE TARGET MODALITY CANDIDATES

In the first step, given a source modality image I_s, multiple target modality candidate intensity values are generated for the set $\Phi^{\boldsymbol{v}}$ at each voxel independently. To generate the target values for $\Phi^{\boldsymbol{v}}$, a $d_1 \times d_1 \times d_1$ patch centered on $\boldsymbol{v}$ extracted from the given source image I_s is used. If we have paired *Source–Target* images during training, we could possibly learn a predictor/regressor

$$g : (\text{Source modality patch at voxel } \boldsymbol{v}) \longrightarrow$$
$$(\text{Multiple target modality candidate values for } \Phi^{\boldsymbol{v})}.$$

But, since such paired training data in not available in the unsupervised setting, the target modality candidates are obtained using cross-modal nearest neighbor search. For each $d_1 \times d_1 \times d_1$ patch from the given source image I_s, K nearest $d_1 \times d_1 \times d_1$ target patches are obtained by searching across the target modality training images. The intensity values of the center voxel and its neighbors from these K nearest patches are used as target candidate values for the set $\Phi^{\boldsymbol{v}}$.

For cross-modal nearest neighbor search, we need a similarity measure that is robust to changes in modality. One such measure is the voxel intensity-based mutual information, which has been successfully used in the past as a cross-modal similarity measure for medical image registration [16]. Given two image patches A and B, their mutual information is given by

$$MI(A, B) = H(X_a) + H(X_b) - H(X_a, X_b), \tag{16.11}$$

where H denotes the Shannon entropy function, X_a and X_b are random variables representing the voxel intensities in patches A and B, respectively.

16.3.2 FULL IMAGE SYNTHESIS USING BEST CANDIDATES

In the second step, given K target modality candidate intensity values for the set $\Phi^{\boldsymbol{v}}$ at each voxel, a full target modality image $\tilde{I}_t$ is synthesized by selecting one among the K candidates at each voxel. The value of $\Phi^{\boldsymbol{v}}(0, 0, 0)$ from the selected candidate is used to synthesize voxel $\boldsymbol{v}$ in $\tilde{I}_t$.

Let X_s and X_t be two random variables with support $\Psi = \{f_1, \ldots, f_L\}$, representing the voxel intensity values of images I_s and $\tilde{I}_t$, respectively. Let $I_s(\boldsymbol{v})$ denote the intensity value of voxel $\boldsymbol{v}$ in image I_s. Let V represent the set of all voxels with cardinality D. Let $\{\phi^{\boldsymbol{v}1}, \ldots, \phi^{\boldsymbol{v}K}\}$ denote the K target modality candidate values for the set $\Phi^{\boldsymbol{v}}$ at voxel $\boldsymbol{v}$. Let $w_{\boldsymbol{v}k} = \mathbb{I}[\text{Candidate } \phi^{\boldsymbol{v}k} \text{ is selected at voxel } \boldsymbol{v}]$.

Since the candidates have been obtained for each voxel independently, the selection problem is solved jointly for all the voxels based on the following criteria: (i) Mutual information maximization, which is a global criterion, and (ii) Spatial consistency maximization, which is a local criterion.

16.3.2.1 *Global Mutual Information Maximization*

Motivated by the assumption that regions of similar tissues (and hence similar gray values) in one modality image would correspond to regions of similar gray values in the other modality image (though the values could be different across modalities), mutual information has been successfully used in the past as a cost function for cross-modal medical image registration [16]. Motivated by this, here, mutual information is used as a cost function for cross-modal medical image synthesis. Since we are interested in generating subject-specific scans, the synthesized target modality image $\tilde{I}_t$ should have high mutual information with the given source modality image I_s, i.e., the amount of information I_s and $\tilde{I}_t$ contain about each other should be maximal. This global criterion helps in transferring the image level structure across modalities.

The mutual information between images I_s and $\tilde{I}_t$ is given by

$$MI(X_s, X_t) = H(X_s) + H(X_t) - H(X_s, X_t). \tag{16.12}$$

Since the entropy $H(X_s)$ is constant for a given source modality image, maximizing $MI(X_s, X_t)$ is equivalent to maximizing $H(X_t) - H(X_s, X_t)$, where

$$\begin{aligned}
H(X_t) &= -\sum_{b=1}^{L} P_b \log[P_b], \\
P_b &= P(X_t = f_b) = \frac{1}{D} \sum_{\boldsymbol{v} \in V} \sum_{k=1}^{K} w_{\boldsymbol{v}k} \mathbb{I}[\phi^{\boldsymbol{v}k}(0, 0, 0) = f_b], \\
H(X_s, X_t) &= -\sum_{a,b=1}^{L} P_{ab} \log[P_{ab}], \\
P_{ab} &= P(X_s = f_a, X_t = f_b) = \frac{1}{D} \sum_{\boldsymbol{v} \in V} \sum_{k=1}^{K} w_{\boldsymbol{v}k} \mathbb{I}[I_s(\boldsymbol{v}) = f_a, \phi^{\boldsymbol{v}k}(0, 0, 0) = f_b].
\end{aligned} \tag{16.13}$$

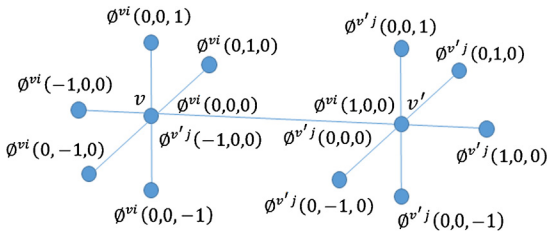

FIGURE 16.6

The values assigned by ϕ^{vi} and $\phi^{v'j}$ to the sets Φ^{v} and $\Phi^{v'}$, respectively.

16.3.2.2 Local Spatial Consistency Maximization

Let $\boldsymbol{v}, \boldsymbol{v}' \in V$ be two neighboring voxels. If we select a candidate ϕ^{vi} at voxel $\boldsymbol{v}$, along with assigning the value $\phi^{vi}(0,0,0)$ to voxel $\boldsymbol{v}$, it also assigns the value $\phi^{vi}(\boldsymbol{v}'-\boldsymbol{v})$ to the neighboring voxel $\boldsymbol{v}'$. Similarly if we select a candidate $\phi^{v'j}$ at voxel $\boldsymbol{v}'$, along with assigning the value $\phi^{v'j}(0,0,0)$ to voxel $\boldsymbol{v}'$, it also assigns the value $\phi^{v'j}(\boldsymbol{v}-\boldsymbol{v}')$ to the neighboring voxel $\boldsymbol{v}$. Fig. 16.6 shows this pictorially. In this case, we would ideally want the selected candidates ϕ^{vi} and $\phi^{v'j}$ to be spatially consistent, i.e.,

$$\phi^{vi}(0,0,0) = \phi^{v'j}(\boldsymbol{v}-\boldsymbol{v}'), \quad \phi^{v'j}(0,0,0) = \phi^{vi}(\boldsymbol{v}'-\boldsymbol{v}). \tag{16.14}$$

Hence, in order to promote spatial consistency among the selected candidates, the following cost function (note the minus sign in the cost) is maximized:

$$SC(\boldsymbol{W}) = -\sum_{\substack{\boldsymbol{v},\boldsymbol{v}'\in V \\ \boldsymbol{v}\backsim\boldsymbol{v}'}} \begin{bmatrix} w_{\boldsymbol{v}1} & \dots & w_{\boldsymbol{v}K} \end{bmatrix} \begin{bmatrix} C_{11}^{\boldsymbol{v}\boldsymbol{v}'} & \dots & C_{1K}^{\boldsymbol{v}\boldsymbol{v}'} \\ \vdots & \ddots & \vdots \\ C_{K1}^{\boldsymbol{v}\boldsymbol{v}'} & \dots & C_{KK}^{\boldsymbol{v}\boldsymbol{v}'} \end{bmatrix} \begin{bmatrix} w_{\boldsymbol{v}'1} \\ \vdots \\ w_{\boldsymbol{v}'K} \end{bmatrix}, \tag{16.15}$$

where $C_{ij}^{\boldsymbol{v}\boldsymbol{v}'} = \sqrt{\left(\phi^{vi}(0,0,0) - \phi^{v'j}(\boldsymbol{v}-\boldsymbol{v}')\right)^2 + \left(\phi^{v'j}(0,0,0) - \phi^{vi}(\boldsymbol{v}'-\boldsymbol{v})\right)^2}$.

Note that here $C_{ij}^{\boldsymbol{v}\boldsymbol{v}'}$ is the spatial consistency cost between neighbors $\boldsymbol{v}$ and $\boldsymbol{v}'$ when ϕ^{vi} is selected at $\boldsymbol{v}$ and $\phi^{v'j}$ is selected at $\boldsymbol{v}'$.

16.3.2.3 Combined Formulation

Combining the global mutual information cost and the local spatial consistency cost, the candidate selection step can be formulated as the following optimization problem:

$$\begin{aligned} &\underset{\{w_{\boldsymbol{v}k}\}}{\text{maximize}} \;\; H(X_t) - H(X_s, X_t) + \lambda SC(\boldsymbol{W}) \\ &\text{subject to} \;\; \sum_{k=1}^{K} w_{\boldsymbol{v}k} = 1, \;\; w_{\boldsymbol{v}k} \in \{0,1\}, \;\; \text{for} \;\; k = 1,\dots,K, \;\; \forall \boldsymbol{v} \in V, \end{aligned} \tag{16.16}$$

where λ is a trade-off parameter.

The optimization problem (16.16) is combinatorial is nature due to the binary integer constraints on $w_{\boldsymbol{v}k}$ and is difficult to solve. Hence, the binary integer constraints are relaxed to positivity constraints to get the following relaxed problem:

$$\begin{aligned} &\underset{\{w_{\boldsymbol{v}k}\}}{\text{maximize}} \;\; H(X_t) - H(X_s, X_t) + \lambda SC(\boldsymbol{W}) \\ &\text{subject to} \;\; \sum_{k=1}^{K} w_{\boldsymbol{v}k} = 1, \;\; w_{\boldsymbol{v}k} \geq 0, \;\; \text{for} \;\; k = 1, \ldots, K, \;\; \forall \boldsymbol{v} \in V. \end{aligned} \tag{16.17}$$

16.3.2.4 Optimization

The cost function $H(X_t) - H(X_s, X_t) + \lambda SC(\boldsymbol{W})$ is differentiable and its derivative with respect to $w_{\boldsymbol{v}k}$ can be computed using:

$$\begin{aligned} \frac{dH(X_t)}{dw_{\boldsymbol{v}k}} &= -\sum_{b=1}^{L} (1 + \log[P(X_t = f_b)]) \frac{d}{dw_{\boldsymbol{v}k}} P(X_t = f_b) \\ &= -\frac{1}{D} \sum_{b=1}^{L} (1 + \log[P(X_t = f_b)]) \, \mathbb{I}[\phi^{\boldsymbol{v}k}(0,0,0) = f_b] \\ &= -\frac{1}{D} \left(1 + \log[P(X_t = \phi^{\boldsymbol{v}k}(0,0,0))]\right), \\ \frac{dH(X_s, X_t)}{dw_{\boldsymbol{v}k}} &= -\sum_{a,b=1}^{L} (1 + \log[P_{ab}]) \frac{dP_{ab}}{dw_{\boldsymbol{v}k}} \\ &= -\frac{1}{D} \sum_{a,b=1}^{L} (1 + \log[P_{ab}]) \, \mathbb{I}[I_s(\boldsymbol{v}) = f_a, \phi^{\boldsymbol{v}k}(0,0,0) = f_b] \\ &= -\frac{1}{D} \left(1 + \log[P(X_s = I_s(\boldsymbol{v}), X_t = \phi^{\boldsymbol{v}k}(0,0,0))]\right), \\ \frac{dSC(\boldsymbol{W})}{dw_{\boldsymbol{v}k}} &= \sum_{\boldsymbol{v}' \backsim \boldsymbol{v}} \left(\sum_{p=1}^{K} C_{kp}^{\boldsymbol{v}\boldsymbol{v}'} w_{\boldsymbol{v}'p} \right). \end{aligned} \tag{16.18}$$

The optimization problem (16.17) has a differentiable cost function with linear equality and inequality constraints. Hence, it can be solved using reduced gradient ascent approach, in which the gradient computed from (16.18) is projected onto the constraint set in each iteration. Once we obtain $w_{\boldsymbol{v}k}$, we use $\phi^{\boldsymbol{v}k^*}(0,0,0)$ to synthesize voxel $\boldsymbol{v}$ in $\tilde{I}_t$, where $k^* = \text{argmax}_k \, w_{\boldsymbol{v}k}$.

Note that though the optimization problem (16.17) is a relaxation of problem (16.16), the unit ℓ_1-norm constraints on the weights promote a sparse solution [2,6] pushing most of $w_{\boldsymbol{v}k}$ toward zero. Since the cost function in (16.17) is non-convex, the reduced gradient ascent approach is not guaranteed to find the global optimum. In the experiments, we use the local optimum obtained by initializing all

the variables $w_{\boldsymbol{v}k}$ with a value of $\frac{1}{K}$. This initialization can be interpreted as giving equal weight to all the K candidates at the beginning of the optimization. During the optimization, in each iteration, along with projecting the gradient on to the constraint set, the ascent direction is also adjusted such that the variables satisfying $w_{\boldsymbol{v}k} = 0$ remain as zero. In each iteration, the learning rate is chosen as the maximum possible value such that none of the variables $w_{\boldsymbol{v}k}$ goes below zero.

16.3.3 REFINEMENT USING COUPLED SPARSE REPRESENTATION

Recently, coupled sparse representation has been shown to be a powerful model when dealing with coupled signal spaces in applications like super-resolution [21,25,26], cross-modal image synthesis [4,8,19], etc. Sparse representations are robust to noise and artifacts present in the data. Hence, CSR is used to refine the synthesized target modality image $\tilde{I}_t$ and generate the final target modality image I_t.

At each voxel $\boldsymbol{v} \in V$, small $d_2 \times d_2 \times d_2$ patches are extracted from the given source modality image I_s and the synthesized target modality image $\tilde{I}_t$. Let $Z^s_{\boldsymbol{v}}$ and $Z^t_{\boldsymbol{v}}$ denote the patches at voxel $\boldsymbol{v}$ from images I_s and $\tilde{I}_t$, respectively. Using $\{(Z^s_{\boldsymbol{v}}, Z^t_{\boldsymbol{v}}) \mid \boldsymbol{v} \in V\}$ as signal pairs from the source and target modalities, coupled sparse representation can be formulated as the following optimization problem:

$$\begin{aligned} &\underset{\boldsymbol{D}_s, \boldsymbol{D}_t, \{\alpha_{\boldsymbol{v}}\}}{\text{minimize}} \sum_{\boldsymbol{v} \in V} \left(\| Z^s_{\boldsymbol{v}} - \boldsymbol{D}_s \alpha_{\boldsymbol{v}} \|_2^2 + \| Z^t_{\boldsymbol{v}} - \boldsymbol{D}_t \alpha_{\boldsymbol{v}} \|_2^2 \right) \\ &\text{subject to } \ \| \alpha_{\boldsymbol{v}} \|_0 \leq T_0 \ \ \forall \boldsymbol{v} \in V, \\ &\qquad\qquad \| \boldsymbol{D}_s(:, j) \|_2 = 1, \ \| \boldsymbol{D}_t(:, j) \|_2 = 1 \ \ \forall j, \end{aligned} \tag{16.19}$$

where $\boldsymbol{D}_s$ and $\boldsymbol{D}_t$ are over-complete dictionaries with M atoms in the source and target modalities respectively, $\alpha_{\boldsymbol{v}}$ is the coupled sparse code for signals $Z^s_{\boldsymbol{v}}$ and $Z_{\boldsymbol{v}} v^t$ in their respective dictionaries, and T_0 is the sparsity parameter.

The dictionaries $\boldsymbol{D}_s$, $\boldsymbol{D}_t$ and the coupled sparse codes $\alpha_{\boldsymbol{v}}$ can be learned by solving the optimization problem (16.19) using the K-SVD [1] algorithm with explicitly re-normalizing the columns of $\boldsymbol{D}_s$ and $\boldsymbol{D}_t$ to unit norm after each iteration. Once the dictionaries and sparse codes are obtained, the target modality patches are reconstructed at every voxel using $\hat{Z}^t_{\boldsymbol{v}} = \boldsymbol{D}_t \alpha_{\boldsymbol{v}}$, and the center voxel value from $\hat{Z}^t_{\boldsymbol{v}}$ is used to synthesize voxel $\boldsymbol{v}$ in the final target modality image I_t.

16.3.4 EXTENSION TO SUPERVISED SETTING

The above described unsupervised synthesis approach can be extended to the supervised setting by simply replacing the cross-modal nearest neighbor search in the candidate generation step with source-modal nearest neighbor search. For each voxel $\boldsymbol{v} \in V$, a $d_3 \times d_3 \times d_3$ patch centered on $\boldsymbol{v}$ is extracted from the given source modality test image I_s and K nearest $d_3 \times d_3 \times d_3$ patches are found from the source modality training images using standard Euclidean distance. Note that a nearest neighbor search (even within source modality) needs to be performed because the training and

Table 16.2 SNR values for the unsupervised synthesis approach

SNR (T1→T2)		SNR (T2→T1)	
No CSR	**CSR**	**No CSR**	**CSR**
12.78 ± 0.61	13.35 ± 0.65	16.23 ± 0.64	16.52 ± 0.70

test images are from different subjects. Once the K nearest source modality training patches are found, the corresponding target modality training patches are used for generating the target modality candidates for $\Phi^{\mathbf{v}}$.

In the CSR step, the paired training data is used to learn coupled dictionaries, and the learned dictionaries are used for refining the synthesized target modality images.

16.3.5 EXPERIMENTS

In this section, we evaluate the above described image synthesis approach by generating T1-MRI scans from T2-MRI scans, and vice versa. We use the same dataset and evaluation metric described in Section 16.2.3. We follow the leave-one-subject-out test setting in which one subject is used for testing and the target modality scans of the remaining 18 subjects are used for training.

Implementation Details. Since exhaustively searching the images (to find nearest neighbors) is highly computational and all the images in the dataset are roughly aligned, we restricted the search in each image to an $h \times h \times h$ (with $h = 7$) region around the voxel of interest. The patch sizes d_1 and d_3 used for cross-modal and source-modal nearest neighbor searches were chosen as 9 and 3, respectively. The patch size used for cross-modal search is much larger than the patch size used for source-modal search because for reliable estimation of mutual information, the patch should have sufficient number of voxels. The number of nearest neighbors K was chosen as 10. Since MRI scans have a high dynamic range, the mutual information computed using the original intensity values would be highly unreliable. Hence, we quantized the intensity values to $L = 32$ levels for computing the mutual information.

Note that the spatial consistency cost (16.15) involves the sum of errors over all pairs of neighboring voxels. As the number of pairs in an image is very large, the magnitude of (16.15) will be much higher than the mutual information cost. Hence, we chose the value of parameter λ in (16.17) such that the mutual information and spatial consistency costs have values that are of the same order of magnitude. For the unsupervised setting, we used $\lambda = 10^{-8}$ and for the supervised setting we used $\lambda = 10^{-7}$. For the CSR step, we used patches with $d_2 = 3$. The sparsity parameter T_0 and the number of dictionary atoms M were chosen as 5 and 500, respectively.

Results. Table 16.2 shows the average SNR values. Figs. 16.7 and 16.8 show some visual examples comparing the unsupervised synthesis results with the ground truth. Similar to the LSDN results, synthesis of T1-MRI from T2-MRI produces better results compared to the synthesis of T2-MRI from T1-MRI. Images without CSR look a bit noisy compared to the images with CSR (please zoom Figs. 16.7 and 16.8).

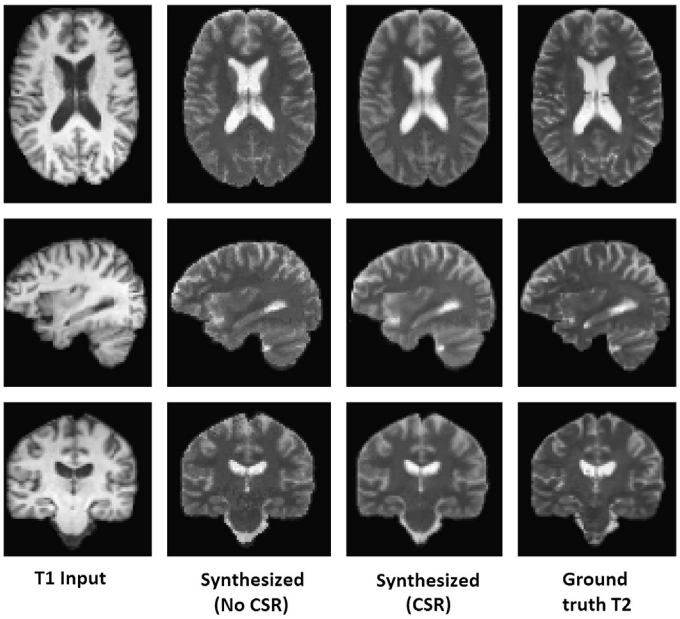

FIGURE 16.7

Comparison of the unsupervised synthesis results with ground truth. T2-MRI synthesis from T1-MRI.

Comparisons. To the best of our knowledge this approach is the first unsupervised cross-modal synthesis approach, and hence there is no existing state-of-the-art to compare with under the unsupervised setting. To show the effectiveness of the candidate selection approach presented in Section 16.3.2, we compare it with the following methods:

1. **First nearest neighbor (F-NN).** We use the center voxel value of the first nearest neighbor for synthesis.
2. **Average of nearest neighbors (A-NN).** We use the average of the center voxel values of all the K nearest neighbors for synthesis.
3. **Candidate selection using only mutual information (MI-only).** We use the center voxel value of the best candidate selected by optimizing only the global mutual information cost. This is equivalent to removing $SC(\boldsymbol{W})$ from optimization problem (16.17).
4. **Candidate selection using only spatial consistency (SC-only).** We use the center voxel value of the best candidate selected by optimizing only the local spatial consistency cost. This is equivalent to removing $H(X_t) - H(X_s, X_t)$ from optimization problem (16.17).

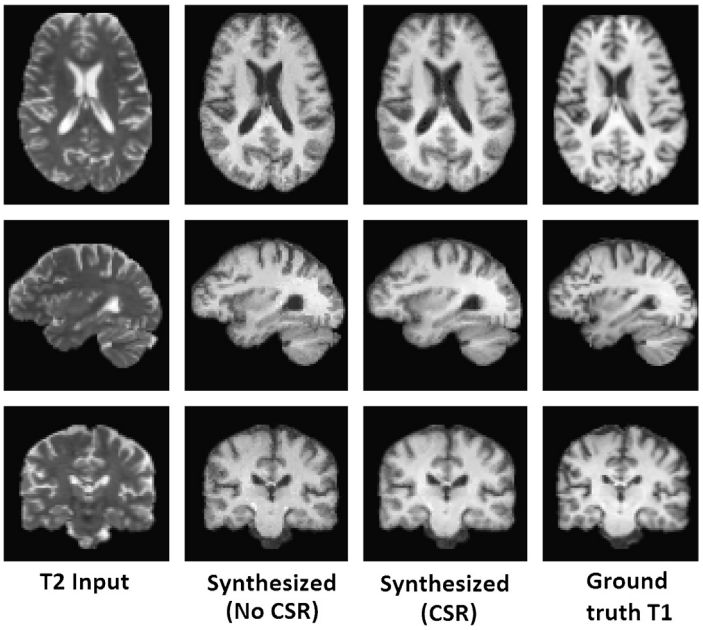

FIGURE 16.8

Comparison of the unsupervised synthesis results with ground truth. T1-MRI synthesis from T2-MRI.

Table 16.3 SNR values for various candidate selection approaches

Approach	SNR (T1→T2)	SNR (T2→T1)
F-NN	10.10 ± 0.45	13.30 ± 0.34
A-NN	12.41 ± 0.61	15.45 ± 0.57
MI only	11.72 ± 0.61	14.88 ± 0.59
SC only	12.11 ± 0.56	15.19 ± 0.53
MI + SC (No CSR)	**12.78 ± 0.61**	**16.23 ± 0.64**

Table 16.3 compares various candidate selection approaches in terms of average SNR values and Fig. 16.9 shows some visual examples. We can clearly see that the presented approach (MI + SC, No CSR) gives the best synthesis results. The low SNR values of F-NN and A-NN methods indicate that directly using the first nearest neighbor or the average of K nearest neighbors is not sufficient for obtaining good synthesis results. While the F-NN method produces very noisy images with spurious structures, the A-NN method produces blurred images. The low SNR values of MI-only and SC-only methods suggest that using only the global mutual information criterion or only the local spatial consistency criterion would produce inferior synthesis results compared to using both criteria. While the images synthesized by the

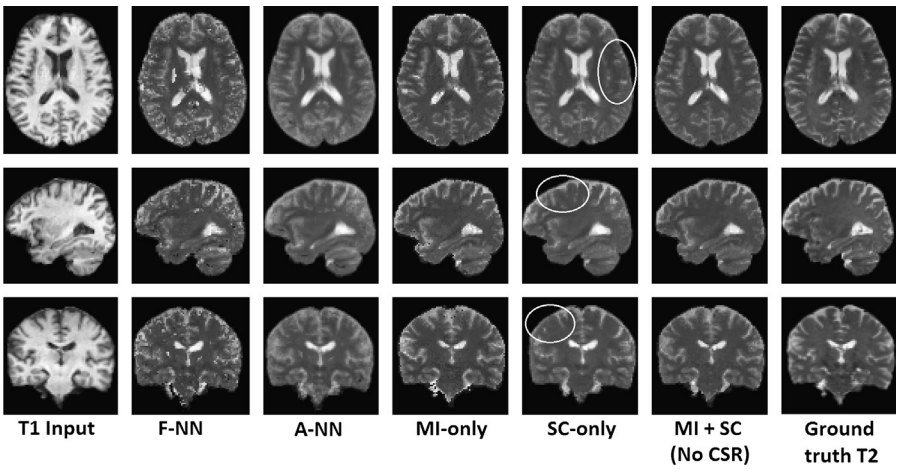

FIGURE 16.9

Comparison of various candidate selection approaches.

Table 16.4 Comparison with supervised approaches

Method	SNR (T1→T2)	SNR (T2→T1)
CSR [4]	13.72 ± 0.64	15.24 ± 0.85
MP [27]	13.64 ± 0.67	15.13 ± 0.88
LSDN [14]	14.93 ± 0.73	17.39 ± 0.91
MI + SC + CSR (unsupervised)	13.35 ± 0.65	16.52 ± 0.70
MI + SC + CSR (supervised)	**15.30 ± 0.94**	**18.33 ± 1.15**

MI-only method are corrupted by salt and pepper type noise, the images synthesized by the SC-only method are missing a lot of structural details (see the circled areas in Fig. 16.9). The presented approach, which uses both MI and SC criteria, is able to get rid of the noise without loosing the structural details.

Supervised Synthesis Results. Table 16.4 compares the synthesis results of the presented approach under the supervised and unsupervised settings with modality propagation [27], coupled sparse representation [4] and location-sensitive deep network [14] methods. We can clearly see that the presented approach outperforms all the three methods under the supervised setting. In fact, the unsupervised approach is able to outperform the supervised modality propagation and CSR methods while synthesizing T1-MRI from T2-MRI.

Computation Time. When ran on a machine with Intel X5650 processor (2.66 GHz, 20 cores), the candidate generation step took 17 min, the candidate selection step took 15 min, and the sparse coding step took 7 min.

16.4 CONCLUSIONS AND FUTURE WORK

In this chapter, we presented two approaches for cross-modal medical image synthesis. The first approach is based on a deep network architecture, referred to as location-sensitive deep network, which integrates spatial information with intensity-based features in a principled manner. LSDN models the first hidden layer nodes as products of certain location-sensitive functions and nonlinear features computed from voxel intensity values. LSDN is a supervised synthesis approach since paired training data is used to learn the network parameters. Along with LSDN, we also presented a sparse coding-based network simplification approach, referred to as ShrinkConnect, to reduce the size of LSDN without losing much in terms of prediction accuracy.

The second approach is based on the principle of mutual information maximization. Given a source modality image, this approach first generates multiple target modality candidate values independently for each voxel by performing nearest neighbor search over the training database. Then, the best candidate values are selected jointly for all the voxels by simultaneously maximizing a global mutual information-based cost function and a local spatial consistency cost function. Finally, coupled sparse representation is used to further refine the synthesized images. This approach can be used under both the unsupervised and supervised settings. We experimentally demonstrated the synthesis capabilities of both the approaches by generating T1-MRI scans from T2-MRI scans and vice versa.

In our experiments, we mainly focused on MR contrast synthesis. In the future, we will apply the presented approaches to other medical imaging modalities. We also plan to use the synthesized images for improving image analysis algorithms like detection and segmentation.

ACKNOWLEDGEMENTS

We would like to acknowledge Siemens Healthcare Technology Center for funding this research.

REFERENCES

1. M. Aharon, M. Elad, A. Bruckstein, K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (11) (2006) 4311–4322.
2. E.J. Candès, J.K. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information, IEEE Trans. Inf. Theory 52 (2) (2006) 489–509.
3. T. Cao, V. Jojic, S. Modla, D. Powell, K. Czymmek, M. Niethammer, Robust multimodal dictionary learning, in: MICCAI, 2013.
4. T. Cao, C. Zach, S. Modla, D. Powell, K. Czymmek, M. Niethammer, Multi-modal registration for correlative microscopy using image analogies, Med. Image Anal. 18 (6) (2014) 914–926.

5. O. Commowick, S.K. Warfield, G. Malandain, Using Frankenstein's creature paradigm to build a patient specific atlas, in: MICCAI, 2013.
6. D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
7. A. Hertzmann, C.E. Jacobs, N. Oliver, B. Curless, D.H. Salesin, Image analogies, in: Annual Conference on Computer Graphics and Interactive Techniques, 2001.
8. D. Huang, Y.F. Wang, Coupled dictionary and feature space learning with applications to cross-domain image synthesis and recognition, in: ICCV, 2013.
9. J.E. Iglesias, E. Konukoglu, D. Zikic, B. Glocker, K.V. Leemput, B. Fischl, Is synthesizing MRI contrast useful for inter-modality analysis?, in: MICCAI, 2013.
10. A. Jog, A. Carass, J.L. Prince, Improving magnetic resonance resolution with supervised learning, in: ISBI, 2014.
11. A. Jog, S. Roy, A. Carass, J.L. Prince, Magnetic resonance image synthesis through patch regression, in: ISBI, 2013.
12. E. Konukoglu, A.J.W. van der Kouwe, M.R. Sabuncu, B. Fischl, Example-based restoration of high resolution magnetic resonance image acquisitions, in: MICCAI, 2013.
13. D. Ma, V. Gulani, N. Seiberlich, K. Liu, J.L. Sunshine, J.L. Duerk, M.A. Griswold, Magnetic resonance fingerprinting, Nature 495 (2013) 187–192.
14. H.V. Nguyen, S.K. Zhou, R. Vemulapalli, Cross-domain synthesis of medical images using efficient location-sensitive deep network, in: MICCAI, 2015.
15. J. Nuyts, G. Bal, F. Kehren, M. Fenchel, C. Michel, C. Watson, Completion of a truncated attenuation image from the attenuated PET emission data, IEEE Trans. Med. Imaging 32 (2) (2013) 237–246.
16. J.P.W. Pluim, J.B.A. Maintz, M.A. Viergever, Mutual information-based registration of medical images: a survey, IEEE Trans. Med. Imaging 22 (8) (2003) 986–1004.
17. F. Rousseau, P.A. Habas, C. Studholme, A supervised patch-based approach for human brain labeling, IEEE Trans. Med. Imaging 30 (10) (2011) 1852–1862.
18. S. Roy, A. Carass, A. Jog, J.L. Prince, J. Lee, MR to CT registration of brains using image synthesis, in: SPIE Medical Imaging, 2014.
19. S. Roy, A. Carass, J.L. Prince, Magnetic resonance image example-based contrast synthesis, IEEE Trans. Med. Imaging 32 (12) (2013) 2348–2363.
20. S. Roy, A. Carass, N. Shiee, D.L. Pham, J.L. Prince, MR contrast synthesis for lesion segmentation, in: ISBI, 2010.
21. A. Rueda, N. Malpica, E. Romero, Single-image super-resolution of brain MR images using overcomplete dictionaries, Med. Image Anal. 17 (1) (2013) 113–132.
22. J.A. Tropp, A.C. Gilbert, M.J. Strauss, Simultaneous sparse approximation via greedy pursuit, in: ICASSP, 2005.
23. R. Vemulapalli, H.V. Nguyen, S.K. Zhou, Unsupervised cross-modal synthesis of subject-specific scans, in: ICCV, 2015.
24. S. Wang, L. Zhang, Y. Liang, Q. Pan, Semi-coupled dictionary learning with applications to image super-resolution and photo-sketch synthesis, in: CVPR, 2012.
25. J. Yang, Z. Wang, Z. Lin, X. Shu, T.S. Huang, Bilevel sparse coding for coupled feature spaces, in: CVPR, 2012.
26. J. Yang, J. Wright, T.S. Huang, Y. Ma, Image super-resolution via sparse representation, IEEE Trans. Image Process. 19 (11) (2010) 2861–2873.
27. D.H. Ye, D. Zikic, B. Glocker, A. Criminisi, E. Konukoglu, Modality propagation: coherent synthesis of subject-specific scans with data-driven regularization, in: MICCAI, 2013.

NOTE

1. Fig. 16.1 shows the intensity mapping of voxels within a small window across modalities. Small window size means that all the voxels are closer to a specific (x, y, z) location. Hence, the importance of smaller window size is effectively same as the importance of a voxel's spatial location.