



Auto-encoded Latent Representations of White Matter Streamlines for Quantitative Distance Analysis

Shenjun Zhong^{1,5} · Zhaolin Chen^{1,4} · Gary Egan^{1,2,3}

Accepted: 10 June 2022 / Published online: 22 June 2022
© The Author(s) 2022

Abstract

Parcellation of whole brain tractograms is a critical step to study brain white matter structures and connectivity patterns. The existing methods based on supervised classification of streamlines into predefined streamline bundle types are not designed to explore sub-bundle structures, and methods with manually designed features are expensive to compute streamline-wise similarities. To resolve these issues, we propose a novel atlas-free method that learns a latent space using a deep recurrent auto-encoder trained in an unsupervised manner. The method efficiently embeds any length of streamlines to fixed-size feature vectors, named streamline embedding, for tractogram parcellation using non-parametric clustering in the latent space. The method was evaluated on the ISMRM 2015 tractography challenge dataset with discrimination of major bundles using clustering algorithms and streamline querying based on similarity, as well as real tractograms of 102 subjects Human Connectome Project. The learnt latent streamline and bundle representations open the possibility of quantitative studies of arbitrary granularity of sub-bundle structures using generic data mining techniques.

Keywords Streamline tractography · Streamline embedding · Streamline clustering · Diffusion MRI · Recurrent auto-encoder

Introduction

Diffusion tractography is a post-processing technique that generates streamlines as a proxy to study the underlying white matter fiber networks. Diffusion-weighted imaging (DWI) is a magnetic resonance imaging (MRI) technique that measures the random motion of water molecules in biological tissues (Jones, 2010). In brain imaging, DWI is used to characterize the distribution and orientation of the underlying white matter pathways. DWI data provides quantitative

diffusion measurements, including the apparent diffusion coefficient (ADC) and fractional anisotropy (FA), that can be further analysed to model the voxel-wise magnitude and orientation information. Current techniques include diffusion tensor imaging (DTI) and spherical harmonics based methods (e.g. constrained spherical deconvolution (Tournier et al., 2008)) for high angular resolution diffusion imaging data (HARDI). Together with fiber tractography, DWI data enable investigation of brain network connectivity using three dimensional (3D) reconstruction of streamlines to represent white matter fibre pathways in the brain.

Streamlines consist of a series of 3D spatial positions that can be grouped into meaningful bundles using streamlining clustering or segmentation algorithms. Since manual streamline anatomical labelling is extremely time-consuming, many automated or semi-automated labelling solutions have been proposed to identify and label streamlines to pre-defined anatomical white matter fiber bundles, e.g. the corticospinal tract (CST) and corpus callosum (CC). The automated solutions to the streamline labeling problem can be: (i) region-of-interest (ROI) based filtering, i.e. the assignment of a streamline is determined by defining ROIs where the streamlines should pass, bypass and terminate (Cammoun et al., 2012;

✉ Shenjun Zhong
shenjun.zhong@monash.edu

¹ Monash Biomedical Imaging, Monash University, Melbourne, VIC, Australia
² School of Psychological Sciences, Monash University, Melbourne, VIC, Australia
³ ARC Center of Excellence for Integrative Brain Functions, Monash University, Melbourne, VIC, Australia
⁴ Department of Data Science and AI, Faculty of Information Technology, Monash University, Melbourne, VIC, Australia
⁵ National Imaging Facility, Brisbane, QLD, Australia

Colon-Perez et al., 2016; Jones & Pierpaoli, 2005; Wakana et al., 2007; Wassermann et al., 2016; Yendiki et al., 2011); (ii) distance based methods that label a streamline to its spatially closest reference streamline tract (Clayden et al., 2007; Corouge et al., 2004; Labra et al., 2017; Maddah et al., 2005); and (iii) classifier-based methods that learn a model from annotated streamline data to correspond to a streamline predefined bundle class, with deep learning (DL) based classifiers with convolutional neural networks (CNNs) (Gupta et al., 2017, 2018; Ugurlu et al., 2018; Zhang et al., 2020) and graph convolutional networks (GCNs) (Liu et al., 2019) used for supervised learning. The ROI-based and distance-based approaches require carefully defined ROIs and reference tracts, and are difficult to adapt to group studies, due to inter-subject variability and time consuming manual annotation. The classifier based approaches require a large number of labeled streamlines to train the DL models, and are only limited to predict a fixed number of predefined streamline bundles according to the manual annotations for model training in the dataset.

Supervised learning approaches are limited to predefined streamline bundle types and require a large number of annotated streamlines via manual or semi-automated annotations. Conversely, unsupervised learning methods are able to cluster streamlines in various granularities without knowledge of the labels. Early methods focused on the design of spatial distance functions. The mean closest point distance (Corouge et al., 2004) was widely used to compute a distance matrix of all the streamline pairs, i.e. inter-streamline distances, and conduct clustering algorithms to decompose streamlines into plausible bundles, including spectral clustering (Ziyan et al., 2009) and hierarchical clustering (Jianu et al., 2009; Zhang et al., 2008). The method calculates the Euclidean distances of each point on a streamline to all points in another streamline. A more computational efficient approach, QuickBundles (Garyfallidis et al., 2012), uses the minimum average direct-flip (MDF) distance, and the element-wise distance measurements of two streamlines that are required to have the same length. The method does not handle different streamline lengths and spatially various endpoints where streamlines belong to the same bundle types but have endpoints that are not apart from each other.

The efficiency of streamline clustering can be improved using a feature space which encodes the streamlines into corresponding feature representations, termed streamline embedding. Measuring the distances between a streamline and the reference streamlines (i.e. the landmarks) can capture the relative position and shape information of a streamline. By selecting a predefined number of reference streamlines, a distance-based feature vector can be constructed with the same length. Olivetti et al. (2012) designed three different policies to select reference streamlines, and used a symmetric minimum average distance

(Zhang et al., 2008) to encode each streamline to a feature vector. Bertò et al. (2021) selected reference landmarks from global and local perspectives, and considered the distances of the streamline endpoints to the global reference landmarks and the predefined ROIs. Lam and co-workers (Lam et al., 2018) also included curvature and torsion as features to represent a streamline. Cabeen et al. (2021) proposed an algorithmic way to select landmarks and represent each streamline with the sparse closest point transform that concatenated the closest points on the streamline curve to form a feature vector. However, methods that rely on landmark selections assume all streamlines including the reference landmarks, have the same length and the embedding performance is related to the number of reference landmarks.

The shape and position of streamlines can be directly modelled by manually designing the feature scheme. Brun et al. (2004) captured the position, shape and connectivity information of streamlines by constructing a 9D vector consisting of the 3D mean vector and the lower triangular part of the covariance matrix of the streamline points. In the work from Maddah et al. (2006), the coefficients of the 3D quintic B-spline were used as the feature vector representation of a streamline. Batchelor et al. (2006) and Chung et al. (2010) used Fourier series as the basis function and the shape of streamlines were represented using a series of Fourier coefficients. This representation model was adopted by Wu et al. (2020) with dictionary prototypes learnt for each predefined bundle type and streamlines classified based on the minimized sparse coding residuals. In the work of FiberMap (Zhang et al., 2020), the sequential information of a streamline was converted to a 2D feature map by splitting the coordinates in the x, y and z axis as three channels, flipping and repeating the spatial coordinates. However, the FiberMap representations of streamlines lack a distance measurement definition so cannot be directly used for quantitative analysis, and the streamlines are assumed to have the same length. Nevertheless, the 2D feature map can be with 2D CNN for supervised learning tasks.

Embedding or representation in a latent space is an effective and robust method for data-driven learning. A number of basis functions have been used to represent streamlines as combinations of the basis functions. Continuous basis functions can be learned with dictionary learning techniques and streamlines can be represented as a combination of dictionaries, to encode streamlines as fixed-size coefficients that represent the feature vector (Alexandroni et al., 2017; Kumar et al., 2019). Similar to our previous work (Zhong et al., 2020), Legarreta et al. (2021) used a 1D CNN-based auto-encoder to learn a latent space to measure distances from the latent streamline representations to manually selected reference streamlines. However, convolutional neural networks do not capture the sequential nature of streamlines

and require fixed size inputs, which limits the ability of the neural network to handle variable streamline lengths.

In this paper, we extended our previous work (Zhong et al., 2020), and proposed and validated a novel recurrent neural network (RNN) to learn latent streamline representations. The key idea is motivated by the successful applications of sequential modelling in natural language processing (NLP) where recurrent neural networks perform well on capturing long sequential information and are proven to effectively model latent features (Cho et al., 2014; Le & Mikolov, 2014; Mikolov et al., 2013a, b). This work uses a Long Short Term Memory (LSTM) RNN to auto-encode any length of streamlines into fixed-size latent vectors. Similarities are measured in the latent space by simply computing the Euclidean distances between the streamlines. Compared to the existing methods, our method can handle streamlines of any length without resampling to ensure equal streamline lengths, and capture the sequential information of streamlines by the latent representation. Furthermore, we propose an effective way to construct group or streamline bundle embedding via averaging latent vectors. Embedding the streamlines and bundles enables clustering, querying, filtering and downstream quantitative analysis of the white matter streamlines.

Methods

In this section, we present the proposed neural network architecture and training scheme, introduce the NN-based streamline embedding mechanism to convert any length of streamline to a multi-dimensional feature vector, show the effective way to obtain embedded streamline bundling from the streamline embedding, and describe the experiments and corresponding validation processes.

Recurrent Autoencoder

Long Short Term Memory (LSTM) is a type of recurrent neural network (RNN) unit that encodes sequential information from variable length data samples (Hochreiter & Schmidhuber, 1997). It consists of several gates, and output

a hidden state h and an output o . At each time step t , the hidden state $h(t)$ is updated by feeding the corresponding inputs and the state from the previous time step $h(t - 1)$, as

$$h(t) = LSTM(h(t - 1), x(t))$$

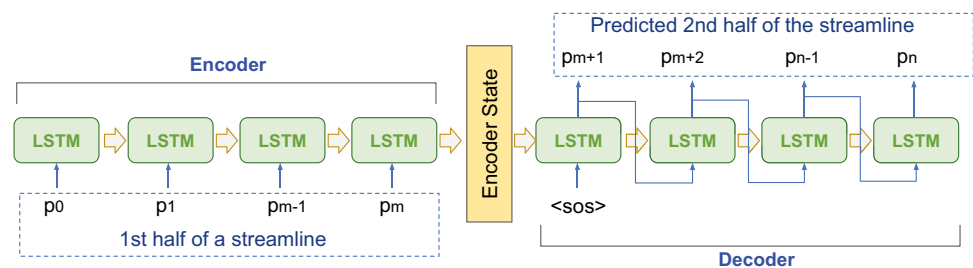
where $x(t)$ is the input data of the current time step. Models with LSTM can learn to predict the next element in a sequence, by fitting the probability distribution, $p(x_t|x_{t-1}, x_{t-2}, \dots, x_1)$ over a sequence. Therefore, LSTM has been widely used in areas where the data have a sequential temporal component, e.g. encoding video features (Bin et al., 2018; Gao et al., 2017), natural language processing (Cho et al., 2014; Sundermeyer et al., 2012) and brain functional analysis (Oota et al., 2019).

Due to the temporal memorization property, LSTM units are successfully used in encoder-decoder neural networks to map an input sequence to a target sequence, which is commonly known as the Seq2Seq auto-encoder network (Sutskever et al., 2014). The encoder transforms the input sequence into a fixed-length latent vector, by feeding each element of an input sequence one by one and updating the hidden states where the last hidden state is considered to contain the summarized information of the entire input sequence. On the other hand, the decoder takes the summary vector from the encoder and generates elements sequentially. During training, encoder-decoder networks are trained in an end-to-end manner, while the trained encoder and decoder can be used independently or together during inference. The encoder can be used to embed an input sequence into a latent vector alone.

Network Architecture and Training

As shown in Fig. 1, a LSTM-based seq2seq auto-encoder was used to embed the streamlines into a fixed length latent representation. The encoder received the first half of each streamline and output a feature vector; and the decoder was trained to predict the second half of the same streamline during the training. The encoder and decoder networks both contained a single layer of LSTM, with 128 hidden dimensions. A seq2seq model with 1-layer LSTM encoder and

Fig. 1 LSTM-based auto-encoder architecture. The first half of the streamline was fed into the encoder of the network sequentially and the hidden state of the encoder was passed to the decoder. The decoder learned to point-by-point predict the second half of the streamline



1-layer LSTM decoder had 136,579 trainable parameters in total. We used Adam optimizer to minimize the loss (i.e. Mean Square Loss) between the predicted and the ground truth, using a learning rate of $1e-3$ and a batch size of 128.

For each training iteration, the input sequence (i.e. the first half of a streamline) was appended with a special token, (0,0,0) to indicate the end of the sequence and the same token was appended to the start of the output sequence (i.e. the second half of the same streamline). The training samples in a batch were padded with zeros to match the length of the longest streamline in the batch. The LSTM hidden and cell states computed from the encoder were passed to the decoder. During decoding, a special token was fed, with value, (0,0,0) to indicate the start of the decoding process, then the decoding loop predicted the next position iteratively. No teacher forcing technique was used, meaning that the input to predict the next position during decoding was the LSTM output from the previous decoding step. The loss was computed by aggregating the individual losses between each predicted position of the second half streamline and the ground truth. A gradient clip of 1.0 was used to prevent gradient explosion, and the training samples were ran-

domly shuffled between each epoch. Besides, each training streamline had 50% chance of reversing the sequence on the fly during training, to encode the streamlines from both directions.

Streamline Embedding

After the model was trained, the encoder was used to embed the streamlines into fixed-length latent vectors, i.e. the streamline length was converted into the corresponding latent representation as shown in Fig. 2. The length of the latent representation could be configured during model training, by setting the hidden dimension in the LSTM cell, in this case, we used a 128 hidden dimension which resulted in the latent vector to be the length of 128. A larger latent dimension, e.g. 256 or 512, leads to a larger parameter space for the auto-encoder, and consequently more computational resources during training and inference. On the other hand, a small latent space, e.g. 32 or 64, requires less computing power, but it is prone to errors if the model is under-fitted.

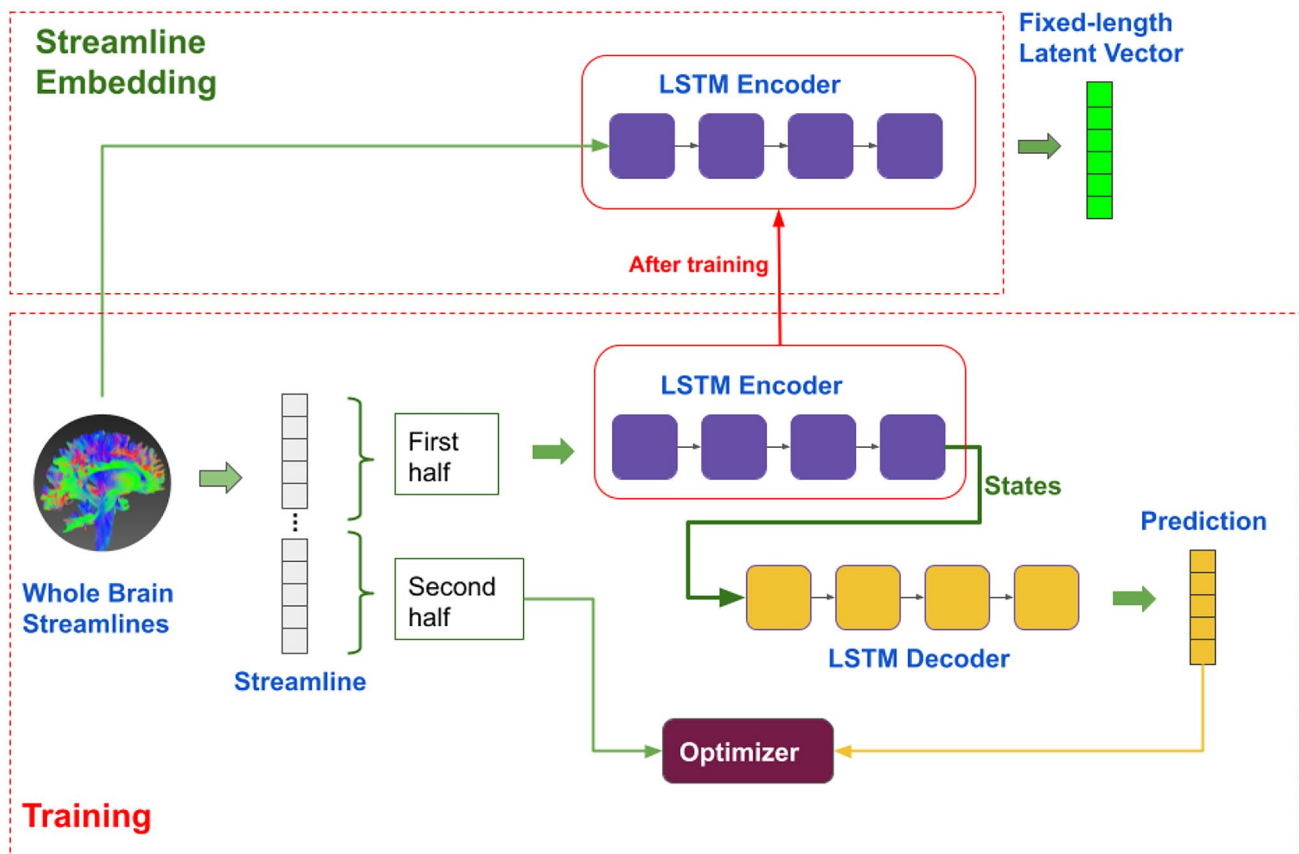


Fig. 2 Training and validation workflow. During training, the whole brain streamlines were used to train the auto-encoder. After trained, only the encoder part was used for embedding any length of streamlines into their fixed-size latent vectors

When training the model with augmented streamlines by randomly reversing their sequence orders, the latent vector was made up of two parts: the encoding of the first half of the streamline and the encoding of the second half with reversed order. By averaging the vectors of the two half streamlines, it resulted in a full latent vector with the same dimension as the LSTM hidden dimension. Alternatively, the two parts could be concatenated to form the full latent vector that contained bi-directional information and was twice the length of the LSTM hidden dimension.

Streamline Bundle Embedding

The embedding of each streamline bundle can be formed to represent streamline groups, by averaging the latent representations of the streamlines in the bundle or group. The bundle embedding, B was defined as the explicit average of the streamline embeddings, S , that belonged to that bundle, i.e.

$$B = \frac{1}{n} \sum_{i=1}^n S_i$$

where n is the number of streamlines in a bundle. In this way, the embedded bundle had the same vector length as the streamlines, which enabled similarity measurements to be computed between the streamline and bundles.

Similarity Measurement and Streamline Clustering

The similarity between two streamlines can be directly measured by calculating the Euclidean distance of their corresponding high-dimensional latent vectors. Similar to the application in NLP where semantically similar words, sentences or paragraphs are closer to each other, the auto-encoded streamlines have the property that anatomically similar streamlines are located closer in the latent space. Hence, the streamline-wise similarity can be calculated using the Euclidean distance metric, which has the computing complexity of $O(n)$. The complexity is only dependent on the length of the latent vector which is customizable when training the auto-encoder, and a smaller hidden dimension can be used, for example 32 or 64, to reduce the distance metric computation.

Similarly, bundle-wise distance and similarity can be measured and any streamline can be classified to defined bundles by the sorted distances. As explained in “[Streamline Bundle Embedding](#)”, streamline bundle embedding can be represented as the average of all or a subset of the embedded streamlines. The similarity between different bundles can be calculated with the Euclidean distance and the distance between a streamline and a bundle can be computed using their corresponding embedded vectors. In this way, if the

bundles are determined, any streamlines can be easily classified to the bundle types by simply measuring the distances.

The latent embedding enables streamline clustering with any clustering algorithm in an unsupervised manner. Each streamline is embedded to a feature vector using the trained encoder, and can be treated as delimited data where each row represents a streamline and columns are the latent features. Since the embedding part is a separate step, any clustering algorithms can be used to segment the streamlines into groups. In our work, the k-means algorithm was used to cluster streamlines into anatomically plausible groups.

Dataset

The first the dataset used in the study was from the ISMRM 2015 Tractography Challenge (Maier-Hein et al., 2015) which contained 25 predefined types of streamlines in the TCK format. There were in total 200,432 streamlines available in the dataset, which covered 25 different types, such as corpus callosum (CC), left and right superior cerebellar peduncle (SCP). The streamlines were further split into training (80%) and validation (20%) datasets. The model with the minimal loss value was selected for the subsequent experiments, where all of the streamlines were used.

The proposed method was also trained and validated on the tractogram dataset generated from Human Connectome Project (HCP) young adult dataset (Van Essen et al., 2013), and the dataset contained segmentations of 72 white matter tracts generated from 105 subjects (Wasserthal et al., 2018), where 102 subjects were used for this work. In the experiments, the subject with HCP ID of 861,456 was randomly chosen for training, while the other 101 subjects were used for testing the generalization of the learnt model.

Experiments and Validation

To validate the latent representations that capture useful information about the streamlines, several streamline clustering experiments were conducted. Streamlines were clustered in the latent space, using the basic K-means algorithm, and assigned to k clusters where k was customizable. In our work, the experiments included: (i) segment symmetric bundles, e.g. left and right SCP, left and right ICP, anterior and posterior commissures, and tests whether the model can separate left/right or anterior/posterior counterparts; (ii) dimension reduction of latent vectors into 2D using t-SNE algorithm and visual check that the features were separable even in lower dimension; (iii) decomposition of the bundles on the left hemisphere (excluding CC, fornix and middle cerebellar peduncle (MCP)) into ten clusters and validation that the clustered streamlines could be mapped back to the ground truth bundle types; (iv) query and filtering of streamlines based on distance measurements; and (v) validation

of the streamline classification accuracy by sorting of the distances in the latent space. The clustered streamlines were further converted back and stored as TCK files for visualization and subsequent processing tasks.

Further experiments were completed on a non-synthetic dataset for validating the performance of the method and examining the generalization of the learnt model. Tractograms of 102 subjects from HCP dataset were used, where data from one subject was used as the training data while the other 101 subjects were kept for testing. In this case, the model was scaled up with 2 layers of LSTMs for both encoder and decoder, and the hidden states output from the second LSTM layer in the encoder were used to form the latent vectors. After the model was trained, 72 bundle embeddings were computed on the tractograms of the single subject data, by averaging all the streamlines embeddings belonging to each bundle type. During validation, streamlines from all the testing subjects were embedded as latent vectors, and classified as the bundle type that was closest to one of the 72 bundle embedding vectors. The classification results were examined by comparing with their ground truth types in the dataset.

The classification accuracy was assessed by computing the top-k metrics, which were defined as the percentage of correctly assigned streamlines based on the Euclidean distances to the embedded bundles. The top-1, top-3 and top-5

were recorded to examine the performance of the learnt latent representation model.

The model and training process were implemented using Python 3.7 and pytorch (version 1.9.1). The tract files were loaded and processed with nibabel (version 3.1.0). The k-means clustering and TSNE dimension reduction were performed using sklearn (version 0.23.2) and numpy (version 1.19.5). The training and validation were performed on both a 32-thread CPU (Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40 GHz) and GPU (Nvidia Quadro P5000). The GPU has 2560 CUDA cores and 16 GB on-chip memory, and the CUDA version 10.2 was used with the driver version of 440.82.

Results

The model converged after training for typically two epochs, without pre-training from other types of tractogram data. Each iteration to train a 1-layer seq2seq model took $0.919 (\pm 0.201)$ seconds on CPU and $0.089 (\pm 0.016)$ seconds on GPU respectively. Training the 2-layer seq2seq model needed $1.637 (\pm 0.380)$ seconds on CPU and $0.107 (\pm 0.014)$ seconds on GPU. There were significant speedups when using the GPU to train models. The best performing model with the lowest loss on the

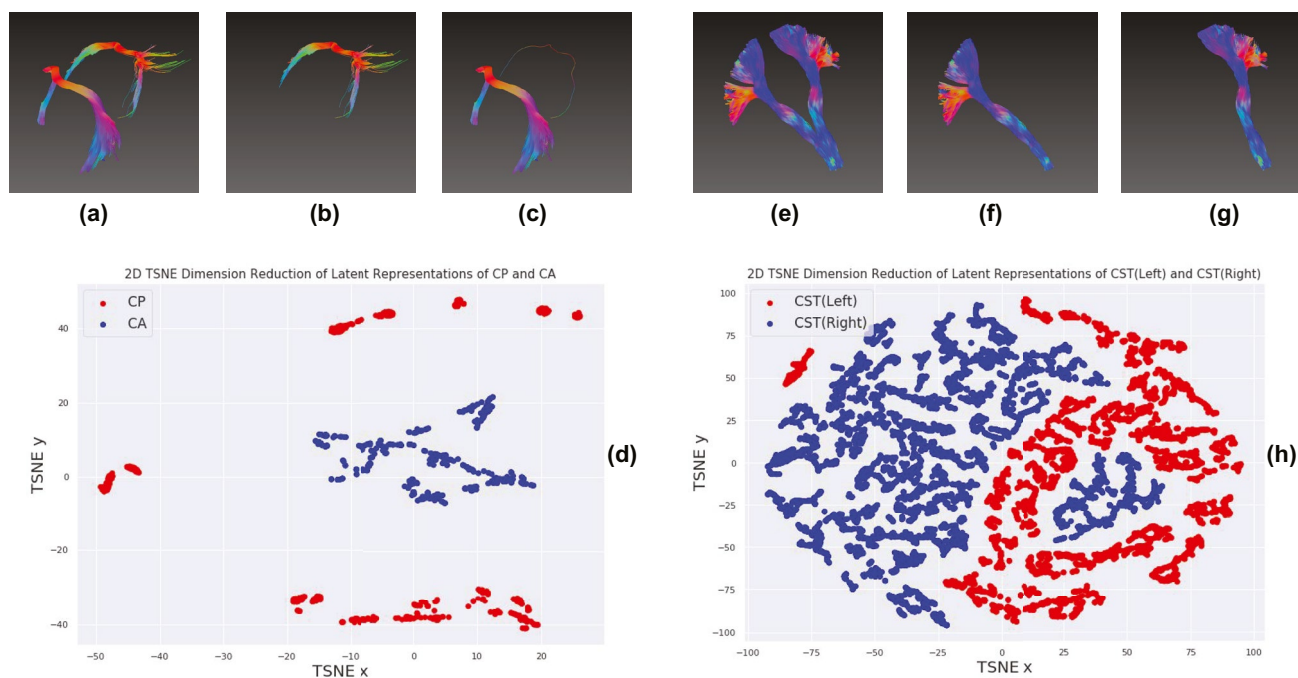


Fig. 3 Streamline clustering of CA/CP and left/right CST, and their t-SNE 2D feature distributions. **a–c** Visual inspection of the clustering results for CA and CP; **d** the t-SNE 2D projection of the latent vectors of all streamlines in CA and CP showing a clearly separation;

e–g clustering results of the left and right CST; and **h** the t-SNE 2D projection of the latent vectors of all streamlines for the left and right CST

validation data was chosen for reporting the results. In the inference phase, only the encoder was used for embedding streamlines.

Streamline Clustering

The latent features can distinguish spatially symmetric structures with non-parametric clustering algorithms. As shown in Figs. 3a–c, e–g and 4, the streamline bundles that have symmetric parts (left and right), e.g. uncinate fasciculus (UF), inferior longitudinal fasciculus (ILF), parieto-occipital pontine tract (POPT), superior longitudinal fasciculus (SLF), corticospinal tract (CST), optic radiation (OR) and cingulum were perfectly clustered, while the results for the fronto-pontine tracts (FPT) were correct with minor errors. However, the left and right components of the inferior cerebellar peduncle (ICP) and superior cerebellar peduncle (SCP) were not separated. Figure 3d and h show the feature distributions of the anterior commissure (CA) and posterior commissure (CP) and the left/right CST after dimension reduction from 128 to 2. These bundles were clearly separable indicating that the learned latent representations captured the necessary spatial information of the streamlines.

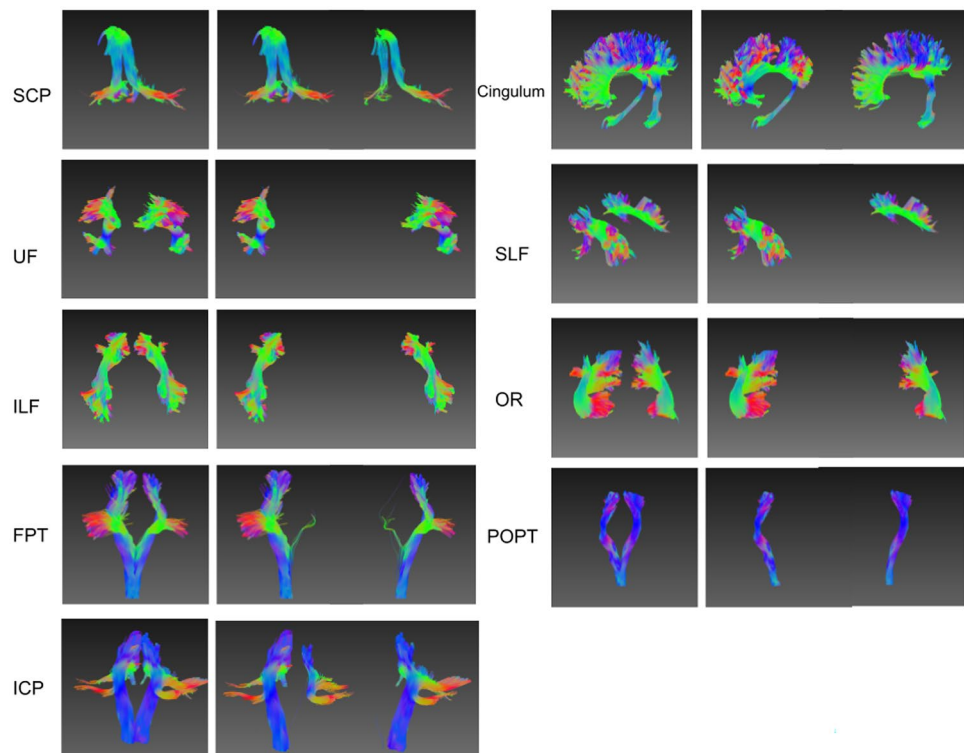
A more challenging task was decomposition of the streamlines in the entire left hemisphere into meaningful bundles. In this case, the corpus callosum (CC), fornix and middle cerebellar peduncle (MCP) were excluded.

As shown in Fig. 5, after clustering into ten classes, the model produced major bundles with minor errors, including for the UF, CST, OR, SLF and cingulum. Some of the bundles overlapped, including the SCP and ICP, POPT and FPT, and ILF and OR. Figure 6 shows the 2D feature embedding of the ten bundles (i.e. 72,944 streamlines) from the left hemisphere after dimension reduction with t-distributed stochastic neighbor embedding (t-SNE) (Van der Maaten & Hinton, 2008). The original feature vectors of 128 dimensions were reduced to two dimensions for visualization purposes. Interestingly, streamlines of the same bundle type were visualized closer to each other and formed separable clusters, even in a two dimensional space representation. There were some major clusters, e.g. the left ILF bundle, the left UF bundle and the left Cingulum bundle, while some small disjointed samples spreaded away from the corresponding large clusters, which were due to either embedding errors or anomalous streamlines in the dataset. Furthermore, it was clear that there was an overlap between the left ILF bundle and the left OR bundle in the 2D visualization, because they were geometrically similar and the model might not perform well in these complicated cases.

Streamline Classification

Using the synthetic dataset we measured the classification accuracy based on the sorted Euclidean distances between the streamlines and the 25 bundles. The results were reported

Fig. 4 Streamline clustering for left/right symmetric bundles. The symmetric bundles are embedded into latent representations and clustered into two groups. UF, ILF, POPT, SLF, CST and OR are perfectly clustered, while the ICP, cingulum and SCP failed to discriminate between symmetric parts



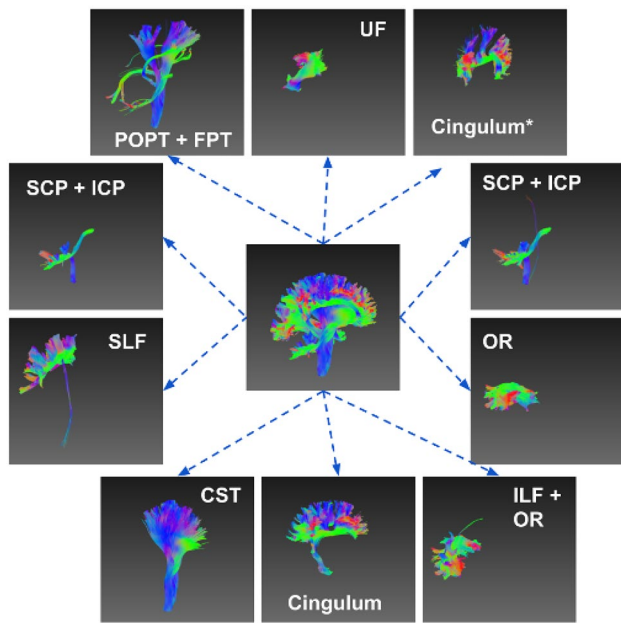
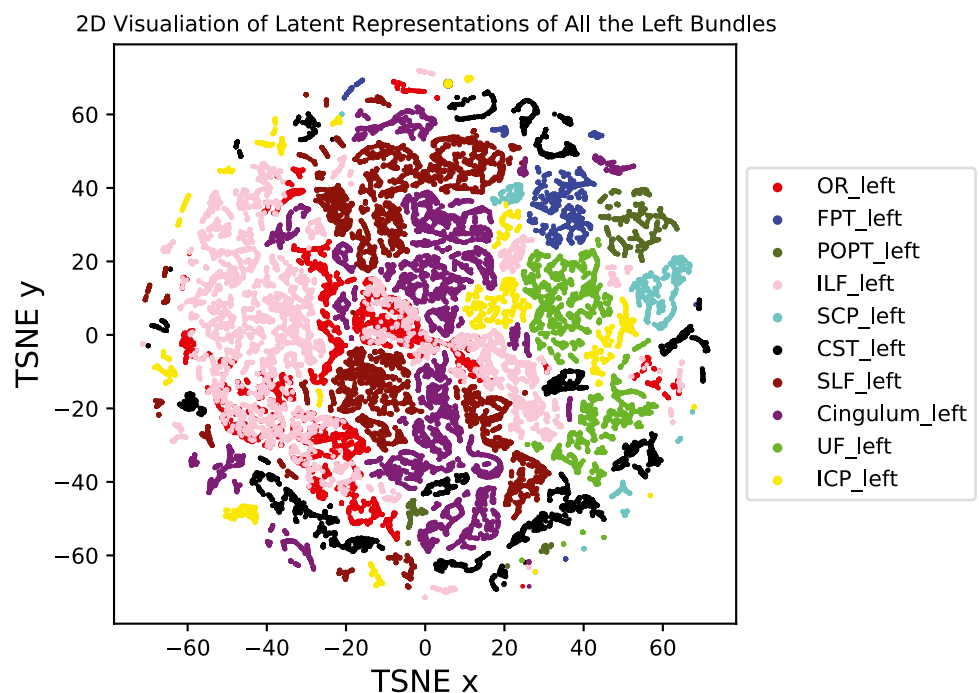


Fig. 5 Decomposition of the left hemisphere and the clustered bundles. All the streamlines in the left hemisphere were converted into latent vectors and clustered into ten groups. Most of the streamline bundles can be extracted, while some groups contain parts from two bundles

for the top-1, top-3 and top-5, i.e. the correct label in the k nearest, where $k = \{1, 3, 5\}$. The overall top-1, top-3 and top-5 accuracy were 81.1%, 97.0% and 98.7%, i.e. around 81% of the time, the nearest bundle type is the correct type.

Fig. 6 t-SNE 2D feature distributions of the ten selected bundles on the left hemisphere. The original feature vectors are 128 dimensions, and the dimensions are reduced to 2D by using t-SNE. Visual inspection of the feature distributions indicates that the streamlines from the same bundle types are closer to each other



The details of the classification accuracy for each bundle type can be seen in Table 1. Seven out of 25 bundles had classification accuracy of $>98\%$ and mostly $>99\%$, including for the left and right SLF, UF, and ICP and for the left CST. However, the accuracy was poorer for the bundles in the top-1 metric, e.g. CC (54%), fornix (61%), left POPT (55%), left SCP (43%), right FPT (55%) and left cingulum (69%). For the top-3 metric all of the bundle types, excluding the CC and left FPT, achieved $>90\%$ accuracy and twelve bundle types (out of 25) had a top-3 metric beyond 99%.

Streamline Querying

The results of querying similar streamlines based on different distance thresholds is shown in Fig. 7. A streamline was randomly selected from the whole brain tractograms and treated as the seeding streamline. In Fig. 7, the seeding streamline was from the right CST bundle. By measuring the distances between the seeding streamline and all other streamlines in the tractogram enables streamlines to be queried within different distance ranges. In this case, small distance thresholds, e.g. 0.125 and 0.25 led to small streamline bundles which were highly similar to the seeding streamline and were a subset of the right CST bundle. Increasing the distance thresholds resulted in the inclusion of more streamlines, with a threshold of 0.75 including the entire right CST bundle. Larger thresholds did not query additional streamlines since there were no streamlines within the distance range of 0.75 and 1. In practice, much smaller step sizes

Table 1 Top K accuracy (out of 25) based on distance measurements for each bundle type

Bundle	Top1	Top3	Top5	Bundle	Top1	Top3	Top5
OR_left	0.8875	0.9998	0.9998	CST_left	0.9979	0.9982	0.9982
MCP	0.9447	0.9999	0.9999	SLF_left	0.9989	0.9995	0.9995
FPT_left	0.6982	0.9652	0.9710	SCP_right	0.7699	0.9878	0.9994
Fornix	0.6197	0.8932	0.9862	FPT_right	0.5571	0.8254	0.9146
UF_right	0.9976	0.9981	0.9997	Cingulum_right	0.8730	0.9836	0.9949
CC	0.5440	0.8598	0.9290	Cingulum_left	0.6906	0.9450	0.9710
ILF_right	0.7613	0.9851	0.9934	CST_right	0.9629	0.9773	0.9979
POPT_left	0.5522	0.9706	0.9848	UF_left	0.9919	0.9919	1.0000
ILF_left	0.7310	0.9982	1.0000	CA	0.9002	0.9954	1.0000
POPT_right	0.8140	0.9152	0.9380	SLF_right	0.9977	0.9999	1.0000
ICP_right	0.9997	0.9997	0.9997	CP	0.7452	0.9781	0.9973
SCP_left	0.4368	0.9733	1.0000	ICP_left	0.9879	1.0000	1.0000
OR_right	0.8090	0.9999	0.9999				

can be used to progressively explore the sub-structure of the white matter bundles.

Quantitatively, selecting the threshold can be guided by the distance measurements between the seeding streamline to the embedded bundles. In Fig. 8, it shows the streamline-bundle distances calculated for the same seeding streamline used in Fig. 7. The distance between the seeding streamline and the bundle embedding of the right CST was 0.719, consistent with the observation that the threshold of 0.75 returns the streamlines from the entire right CST bundle. The distance to the second closest bundle type, the right UF bundle was 1.95. Hence, any threshold selection below that value did not include streamlines from other bundle types.

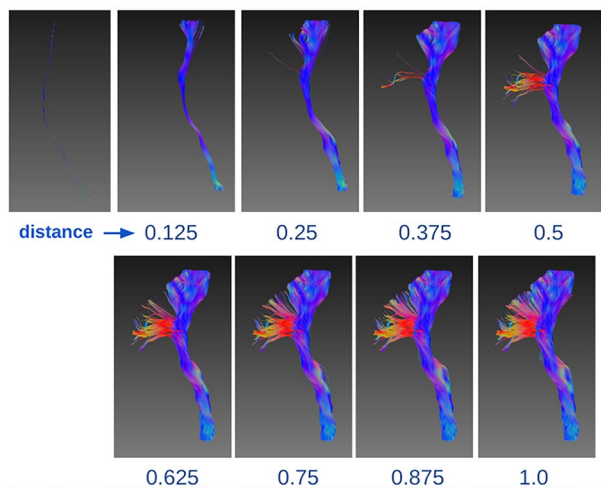


Fig. 7 Query similar streamlines by various distances. A streamline is randomly selected from left CST and used as the seeding streamline to query similar streamlines based on various distance thresholds. Use of a smaller threshold results in sub-bundles that are very similar to the seeding streamline, while a larger threshold leads to the extraction of the entire left CST bundle

Bundle-Wise Distance Analysis

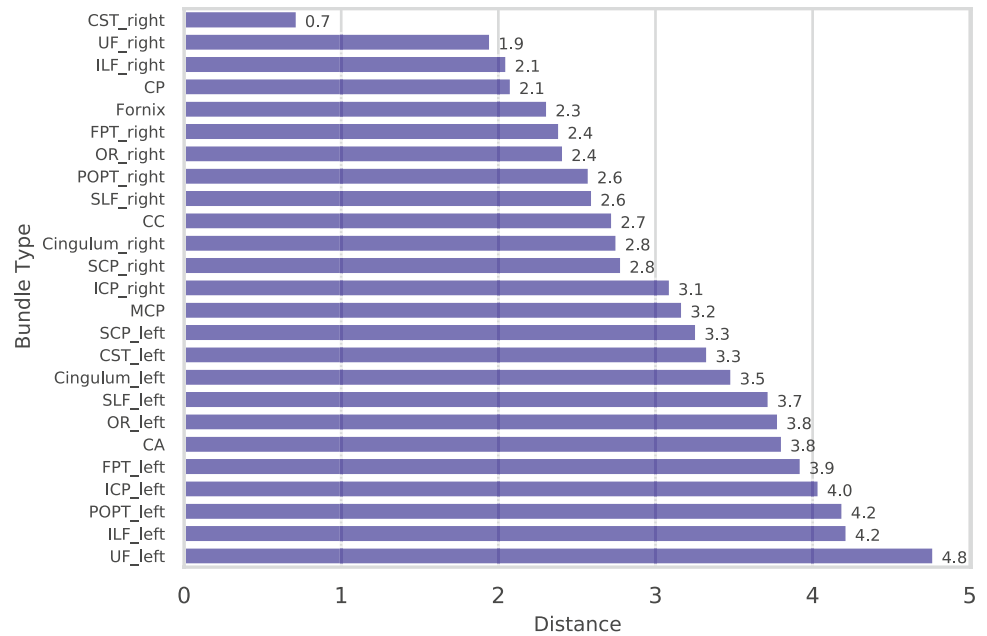
Group-wise similarity between the embedded bundles can be analyzed by the distance matrix. Figure 9 shows the distances between the 25 predefined bundles in the synthetic dataset, using bundle embedding. Symmetric bundles, such as left and right ICP were closer to each other in the latent space, and the same property was observed for SCP and the cingulum indicating that the shape information was encoded by the neural network. SCP and ICP bundles were closer to each other, as part of the streamlines have similar spatial properties. Groups of bundles are spatially closer to each other in the top-left part of the distance matrix, including the CA, CC, CP, left and right CST, left and right cingulum, left and right FPT and the fornix.

Hierarchical Streamline Clustering and Streamline Filtering

Figure 10 shows a hierarchical dissection of the streamlines in the corpus callosum using non-parametric clustering of the streamline embedding. It is similar to a binary tree and each non-leaf node is clustered into two groups. It can be seen that the dissection was along the anterior and posterior direction and the iterative process segmented the CC bundle into two, four and eight clusters on level 1, 2 and 3 respectively. In this case, we used two clusters for each non-leaf node although any number of clusters can be set, and different numbers of clusters can be used for non-leaf nodes on the same level. For example, on level 1, the first sub-bundle can be further segmented into three sub-parts, while the second sub-bundle can be segmented into four sub-parts.

Clustering of streamline embedding can be used to identify and remove false streamlines. In Fig. 10, the first sub-bundle on level 2 was further segmented into two parts (i.e.

Fig. 8 Sorted embedding distances of a randomly sampled streamline



the sub-bundle 1 and 2 on level 3), with the streamlines in the second sub-bundle on level 3 identified as false connections (in this case, the streamlines seeded from cortical region in the left hemisphere, had a sharp turn in the past in the right hemisphere and ended in the seeding region). This finding highlights that the proposed technique can filter false streamlines generated in the tractography process as embedded streamline are spatially separable in the latent space.

Model Generalization Validation on Large Scale Non-synthetic Data

The validation results on tractograms from HCP dataset shows that the model learnt from the training data and the bundle embedding generated from the single training subject can be generalized to a large population without the need of extra processing steps. We measured the top1, top3 and top3 classification accuracies for each bundle type over all the 101 testing subjects. A 2-layer LSTM network for both encoder and decoder were used to increase the model capability. Overall, the proposed method achieved an average top-1 accuracy of $0.591(\pm 0.243)$ over all the bundles on the 101 testing subjects, $0.853(\pm 0.153)$ of top-3 and $0.923(\pm 0.111)$ of top-5, compared to top-1, $0.780(\pm 0.120)$, top-3, $0.953(\pm 0.070)$ and top-5, $0.980(\pm 0.041)$ on the training subject. In Fig. 11, it shows the accuracy distributions for all the bundles across all the testing subjects. There were 19 out of 72 bundles that had an average top-1 accuracy beyond 80%, and 9 bundles achieved

accuracy above 90%. Particularly, the top-1 accuracies for MCP, the right Inferior occipito-frontal fascicle (IFO) and the right CST were $99.9\%(\pm 0.001)$, $94.8\%(\pm 0.055)$ and $93.5\%(\pm 0.43)$. In terms of the top-3 measurement, classification accuracies were above 80% for 53 bundles and above 90% for 33 bundles. Furthermore, there were 54 and 64 bundles out of all the 72 bundles having top-5 accuracies above 80% and 90% respectively.

Inference Throughput

To measure the inference time of the proposed methods, 10,000 streamlines were randomly sampled from the HCP tractogram dataset, and were used to measure the time of embedding them into latent vectors using 1-layer and 2-layer LSTM encoders. For each test case, 20 runs were conducted and the means and standard deviations were calculated and shown in Table 2. It can be seen that the model running on the CPU, can process 10,000 streamlines uni-directionally within 0.6 s using 1 layer of LSTM and 0.72 s for encoding streamlines from two directions. Doubling the number of LSTM layers in the model on the CPU took twice as long. On the other hand, using the GPU achieved roughly $20\times$ speedup, and completed the same uni-directional tasks within 0.026 s and 0.053 s using 1-layer and 2-layer LSTM encoders. Besides, it was worth mentioning that the inference times for encoding streamlines from bi-directions were very similar for models with 1 or 2 LSTM layers.

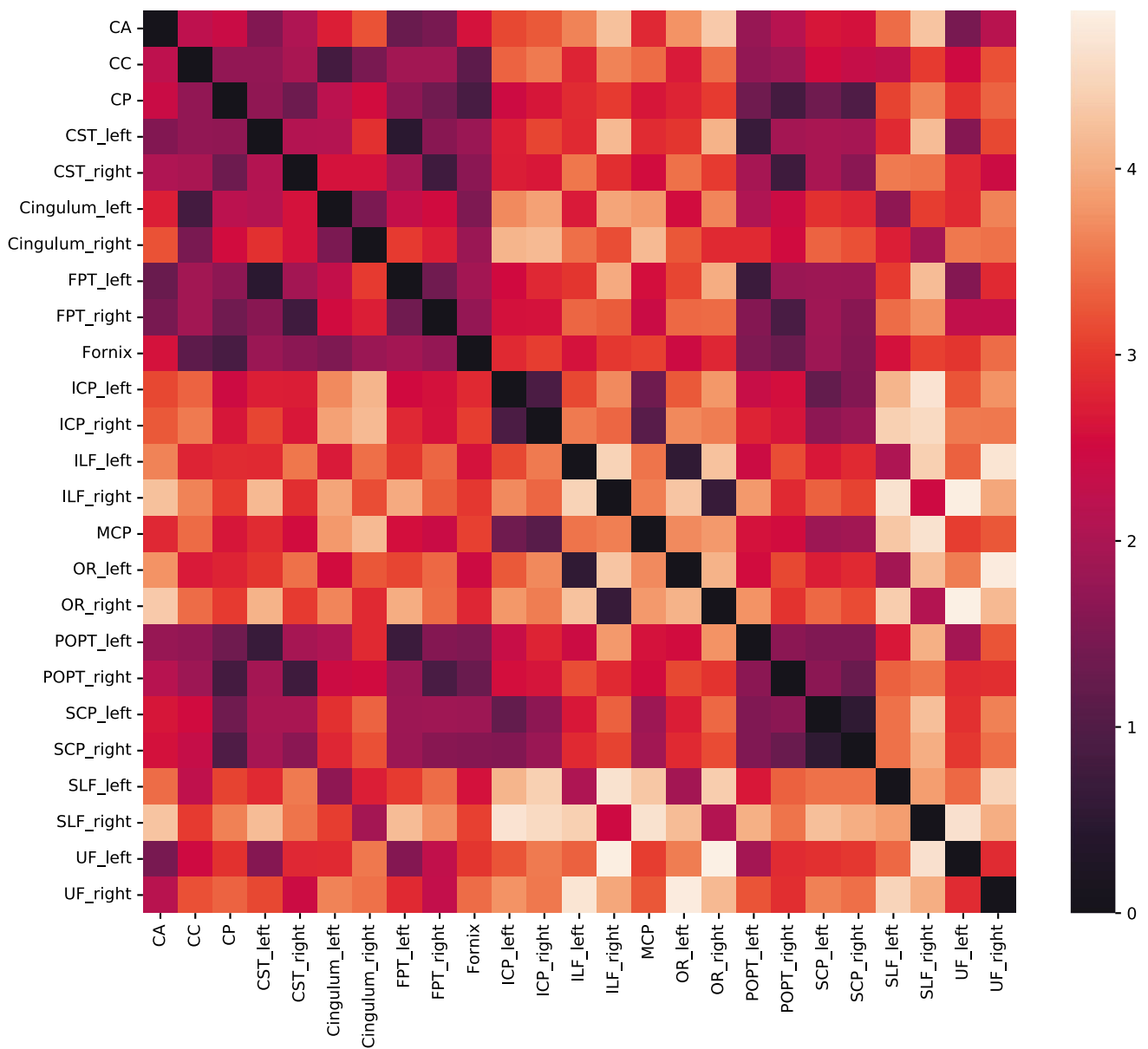


Fig. 9 Bundle-wise distance matrix. Each bundle is represented by its bundle embedding which is the average of all the streamlines in this bundle. The bundle-wise distance matrix is constructed by computed the pairwise distances among all the bundle types

Discussion

We present a recurrent neural network to encode streamlines with variable lengths into fixed-size latent vector representations and quantify similarities by measuring the Euclidean distance between the streamlines. Streamline bundles were represented as the average of the streamline latent vectors. With both embedded streamlines and bundles in the same vector length, the method enables quantitative analysis of streamlines at any granularity of streamline bundles. We validated the method using the ISMRM 2015 tractography challenge dataset and tractograms data generated from 102 subjects in HCP dataset, and showed that embedding

streamlines to a learnable latent space enables efficient similarity measurement and streamline clustering.

The method is flexible such that any streamline length can be embedded into a latent vector of variable size. Existing methods generally resample streamlines to ensure they have the same length as required for streamline-wise distance measurements (Garyfallidis et al., 2012) and as inputs into neural networks for classification (Colon-Perez et al., 2016; Liu et al., 2019). CNN-based auto-encoders suffer from the same length limitation due to the fixed input size of CNN (Lam et al., 2018). However, the resampling process results in the loss of detailed information of the streamline and treats long and short streamlines differently. Our

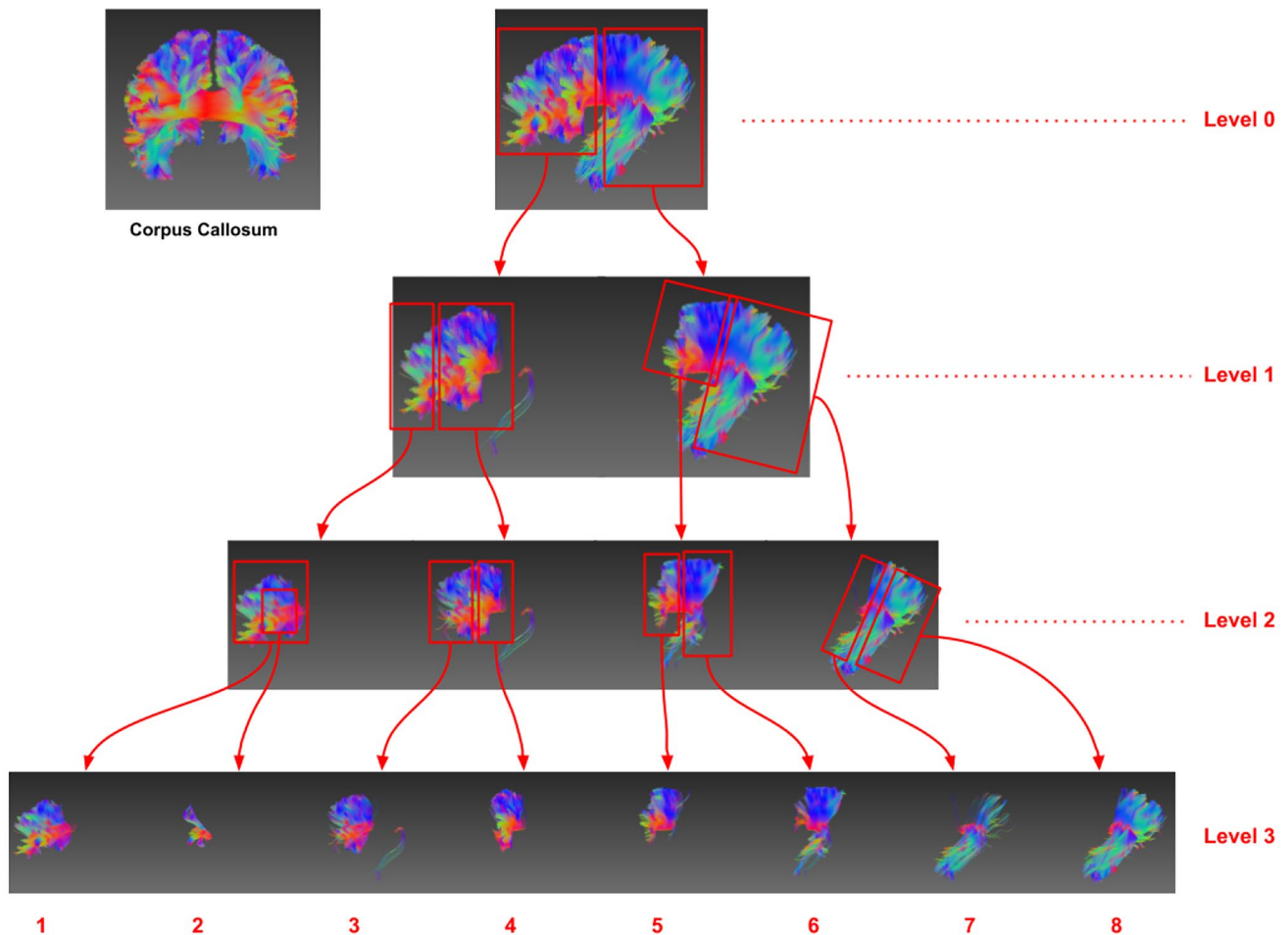


Fig. 10 Hierarchical dissection of streamlines in the corpus callosum using streamline embedding clustering. On each level, the sub-bundles are further clustered into two groups iteratively. On level 3, the CC bun-

dle can be dissected into eight sub-bundles along the anterior–posterior direction. Among the sub-bundles on level 3, the second leaf sub-bundles are identified as false streamlines

RNN-based method naturally supports variable lengths of sequence input without the need to truncate or resample the original streamlines. Unlike manually designed features (Yendiki et al., 2011; Bertò et al., 2021), the latent space dimension in the proposed framework can be customized by adjusting the RNN hidden dimension. RNN networks with larger hidden dimension, i.e. higher dimension of the latent space, will have better capability of representing longer and more complex streamline structures, model larger numbers of streamlines, while smaller latent spaces require less computation. Hence, the size of the latent vectors is flexible and can be tailored in a case-by-case manner. In our work, a 128 dimension latent space was used, which was effective for modelling the streamlines in the ISMRM 2015 tractography dataset.

Sequential information is important when encoding streamlines into latent representations. A streamline is essentially a sequence of 3D spatial points that represent its location and shape in the brain. Treating a streamline as

a collection of points may lose its sequential information during embedding. Both distance-based and even recent deep learning methods that are based on CNN networks fail to capture the sequential components. CNN network architectures are designed to be translationally invariant, and do not view the streamlines as an ordered sequence but instead process it, for the case of auto-encoding streamlines, as a 1D image with three channels. CNNs are sensitive to streamline segment patterns but ignore positions in the streamlines. Unlike CNNs, RNNs not only treat sequences naturally but provide a better representation capability for long sequences, i.e. streamlines with long length. Adding position encoding techniques to CNNs may compensate for the weakness of dealing with sequential data, similar to recently introduced techniques such as the Transformer architecture that could potentially be used for encoding sequential data types with attention mechanisms (Wolf et al., 2020). However, such approaches are not the focus of the current paper and will be explored in future work.

Fig. 11 Classification accuracy distributions of top-1, top-3 and top-5 measurements on all the 72 bundle types for the 101 testing subjects

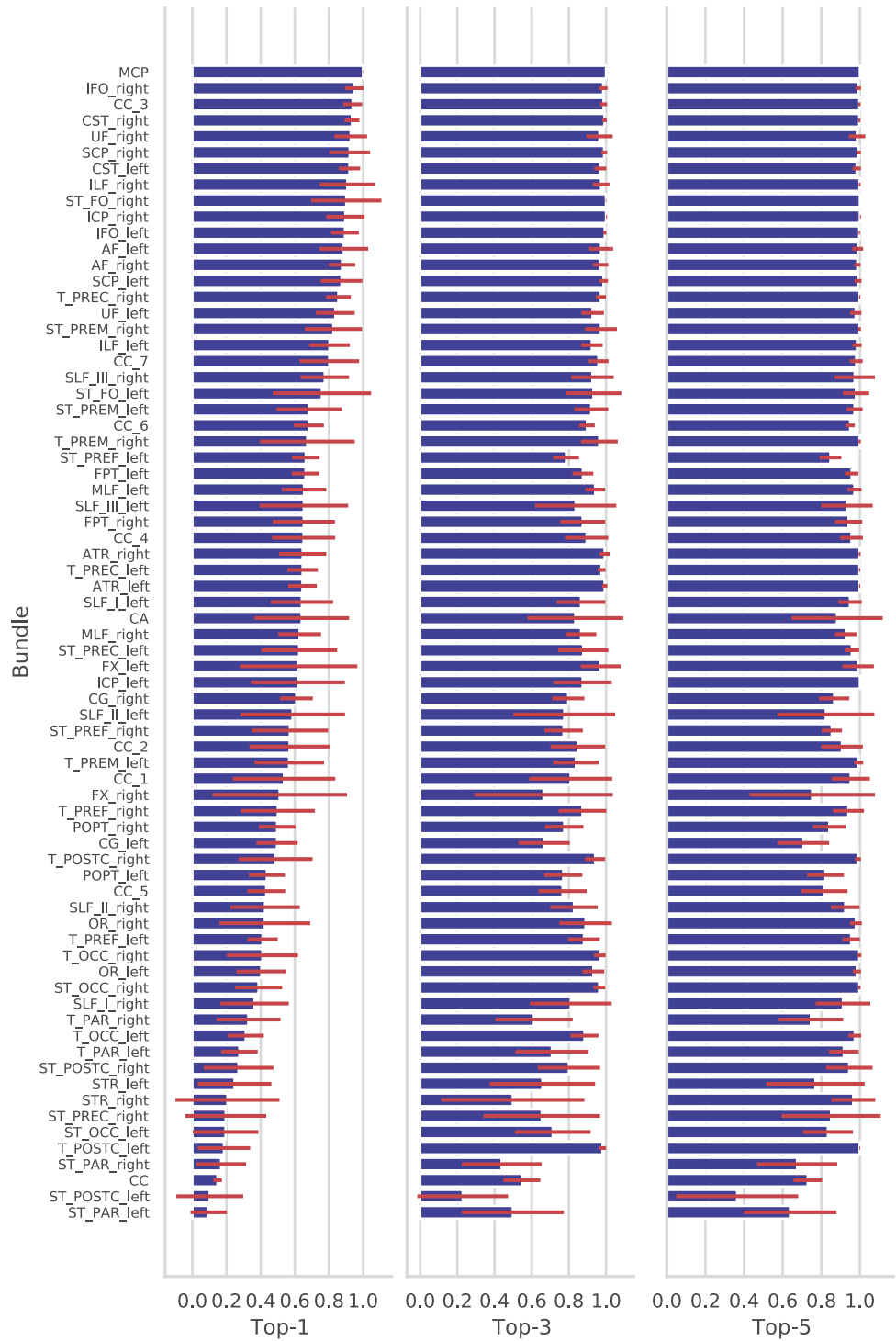


Table 2 Inference time in seconds for embedding 10,000 streamlines with 1 and 2 LSTM layer(s) encoder on CPU and GPU

	1 LSTM Layer Encoder		2 LSTM Layers Encoder	
	Uni-directional	Bi-directional	Uni-directional	Bi-directional
CPU (Intel Xeon E5-2630)	0.588 (±0.029)	0.722 (±0.018)	1.209 (±0.046)	1.344 (±0.041)
GPU (Nvidia Quadro P5000)	0.026 (±0.002)	0.144 (±0.001)	0.053(±0.001)	0.146(±0.001)

The latent space learnt with RNN auto-encoding provides an efficient way to form streamline bundle representations using data-driven clustering algorithms. Any granularities of the streamline bundle can be represented using the average of the embedded streamlines in the bundle in a computationally efficient manner. Distances can be measured between streamlines and also between a streamline and a bundle, since they are in the same closed latent space. The framework represents each streamline and bundle as tabular data, opening the possibility of analysis with generic data mining algorithms (e.g. anomaly detection) capable of dealing with high dimensional data. Bundle embedding can also be used to identify group-level landmarks (Gori et al., 2016) and create a multi-subject atlas (Guevara et al., 2012) in a data-driven and unsupervised manner. Finally, using the trained decoder the embedded bundle can be projected back to the 3D space for visual inspection purposes.

Of crucial importance is whether the streamline groups or bundles generated from clustering algorithms in a latent space are anatomically plausible. We validated the streamline groups obtained using k-means clustering on the left brain hemisphere and manually examined each group, identified their similarity to the ground truth bundle types and found that the major bundles were extracted with minimal bias. Our motivation was not to re-discover well defined bundles, but rather to explore smaller structures, streamline filtering and hierarchical dissection. Qualitatively, the method requires a human expert to visually inspect the identified sub-bundle structures and determine their anatomical validity. The framework provides a data-driven streamline parcellation tool to study sub-bundle structure as an alternative to traditional cortical atlas based methods (Bullmore & Sporns, 2009; Sporns et al., 2005). Quantitatively, the method has the potential to be used with graph theory analysis over populations (Bassett & Bullmore, 2017; Yeh et al., 2016) to study connectivity patterns at various granularities of streamline parcellation.

The performance of the method can be improved by encoding streamlines from both directions and further validated using both simulated and non-synthetic datasets. In the current work, the encoding was initiated from one direction or both directions using uni-directional LSTM modules whereas bidirectional LSTM (biLSTM) views of streamline in the auto-encoding process may improve the robustness of the embedding model. Alternatively, the effectiveness of using the Transformer architecture could be explored to learn a latent space to represent streamlines. Also, due to the population variation, improving model generalization is another future research direction. Furthermore, other clustering methods can also be used as a replacement for the k-means clustering algorithm in this case to validate the representation accuracy of the proposed embedding method.

Conclusion

In this work, we proposed a novel recurrent neural network auto-encoder to embed any length of white matter streamlines and at any level of bundle granularity, to a fixed-size latent representation for parcellation of whole-brain tractograms. The embedding process preserves the position, shape and sequential information of the streamline and bundle. The method was validated on both synthetic dataset and tractograms from the HCP project, and it was effective to distinguish symmetric bundle structures, produce bundle groups with clustering algorithms, and achieved good classification accuracy based on distance measurements on an annotated dataset. The method can be applied to query streamlines according to similarity, hierarchically dissect streamline bundles for studying sub-structures, and identify and remove biased streamlines. The method enables the possibility of quantitative distance analysis using data mining and unsupervised learning techniques.

Information Sharing Statements The source code of the proposed work is freely available for non-commercial use from <https://github.com/BioMedAnalysis/latenttrack>.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Declarations

Conflict of Interest The authors have no relevant financial interests in this article and no potential conflicts of interest to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Alexandroni, G., Podolsky, Y., Greenspan, H., Remez, T., Litany, O., Bronstein, A., & Giryas, R. (2017, September). White matter fiber representation using continuous dictionary learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 566–574). Springer, Cham.
- Bassett, D. S., & Bullmore, E. T. (2017). Small-world brain networks revisited. *The Neuroscientist*, 23(5), 499–516.

- Bertò, G., Bullock, D., Astolfi, P., Hayashi, S., Zigiotta, L., Annicchiarico, L., & Olivetti, E. (2021). Classifyber, a robust streamline-based linear classifier for white matter bundle segmentation. *NeuroImage*, 224, 117402.
- Batchelor, P. G., Calamante, F., Tournier, J. D., Atkinson, D., Hill, D. L. G., & Connelly, A. (2006). Quantification of the shape of fiber tracts. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 55(4), 894–903.
- Brun, A., Knutsson, H., Park, H. J., Shenton, M. E., & Westin, C. F. (2004, September). Clustering fiber traces using normalized cuts. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 368–375). Springer, Berlin, Heidelberg.
- Bullmore, E., & Sporns, O. (2009). Complex brain networks: Graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3), 186–198.
- Bin, Y., Yang, Y., Shen, F., Xie, N., Shen, H. T., & Li, X. (2018). Describing video with attention-based bidirectional LSTM. *IEEE Transactions on Cybernetics*, 49(7), 2631–2641.
- Cabeen, R. P., Toga, A. W., & Laidlaw, D. H. (2021). Tractography Processing with the Sparse Closest Point Transform. *Neuroinformatics*, 19(2), 367–378.
- Cammoun, L., Gigandet, X., Meskaldji, D., Thiran, J. P., Sporns, O., Do, K. Q., & Hagmann, P. (2012). Mapping the human connectome at multiple scales with diffusion spectrum MRI. *Journal of Neuroscience Methods*, 203(2), 386–397.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint [arXiv:1406.1078](https://arxiv.org/abs/1406.1078).
- Chung, M. K., Adluru, N., Lee, J. E., Lazar, M., Lainhart, J. E., & Alexander, A. L. (2010). Cosine series representation of 3D curves and its application to white matter fiber bundles in diffusion tensor imaging. *Statistics and Its Interface*, 3(1), 69.
- Clayden, J. D., Storkey, A. J., & Bastin, M. E. (2007). A probabilistic model-based approach to consistent white matter tract segmentation. *IEEE Transactions on Medical Imaging*, 26(11), 1555–1561.
- Colon-Perez, L. M., Triplett, W., Bohsali, A., Corti, M., Nguyen, P. T., Patten, C., & Price, C. C. (2016). A majority rule approach for region-of-interest-guided streamline fiber tractography. *Brain Imaging and Behavior*, 10(4), 1137–1147.
- Corouge, I., Gouttard, S., & Gerig, G. (2004, April). Towards a shape model of white matter fiber bundles using diffusion tensor MRI. In 2004 2nd IEEE International Symposium on Biomedical Imaging: Nano to Macro (IEEE Cat No. 04EX821) (pp. 344–347). IEEE.
- Gao, L., Guo, Z., Zhang, H., Xu, X., & Shen, H. T. (2017). Video captioning with attention-based LSTM and semantic consistency. *IEEE Transactions on Multimedia*, 19(9), 2045–2055.
- Garyfallidis, E., Brett, M., Correia, M. M., Williams, G. B., & Nimmo-Smith, I. (2012). Quickbundles, a method for tractography simplification. *Frontiers in Neuroscience*, 6, 175.
- Gori, P., Colliot, O., Marrakchi-Kacem, L., Worbe, Y., Fallani, F. D. V., Chavez, M., & Durrleman, S. (2016). Parsimonious approximation of streamline trajectories in white matter fiber bundles. *IEEE Transactions on Medical Imaging*, 35(12), 2609–2619.
- Guevara, P., Duclap, D., Poupon, C., Marrakchi-Kacem, L., Fillard, P., Le Bihan, D., & Mangin, J. F. (2012). Automatic fiber bundle segmentation in massive tractography datasets using a multi-subject bundle atlas. *NeuroImage*, 61(4), 1083–1099.
- Gupta, V., Thomopoulos, S. I., Rashid, C. K., Rashid, F., & Thompson, P. M. (2018, April). Fibernet 2.0: an automatic neural network based tool for clustering white matter fibers in the brain. In 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018) (pp. 708–711). IEEE.
- Gupta, V., Thomopoulos, S. I., Rashid, F. M., & Thompson, P. M. (2017, September). FiberNET: An ensemble deep learning framework for clustering white matter fibers. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 548–555). Springer, Cham.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Jianu, R., Demiralp, C., & Laidlaw, D. (2009). Exploring 3D DTI fiber tracts with linked 2D representations. *IEEE Transactions on Visualization and Computer Graphics*, 15(6), 1449–1456.
- Jones, D. K. (2010). *Diffusion mri*. Oxford University Press.
- Jones, D. K., & Pierpaoli, C. (2005). Confidence mapping in diffusion tensor magnetic resonance imaging tractography using a bootstrap approach. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 53(5), 1143–1149.
- Kumar, K., Siddiqi, K., & Desrosiers, C. (2019). White matter fiber analysis using kernel dictionary learning and sparsity priors. *Pattern Recognition*, 95, 83–95.
- Labra, N., Guevara, P., Duclap, D., Houenou, J., Poupon, C., Mangin, J. F., & Figueroa, M. (2017). Fast automatic segmentation of white matter streamlines based on a multi-subject bundle atlas. *Neuroinformatics*, 15(1), 71–86.
- Lam, P. D. N., Belhomme, G., Ferrall, J., Patterson, B., Styner, M., & Prieto, J. C. (2018, March). TRAFIC: fiber tract classification using deep learning. In Medical Imaging 2018: Image Processing (Vol. 10574, p. 1057412). International Society for Optics and Photonics.
- Le, Q., & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188–1196). PMLR.
- Legarreta, J. H., Petit, L., Rheault, F., Theaud, G., Lemaire, C., Descoteaux, M., & Jodoin, P. M. (2021). Filtering in Tractography using Autoencoders (FINTA). *Medical Image Analysis*, 102126.
- Liu, F., Feng, J., Chen, G., Wu, Y., Hong, Y., Yap, P. T., & Shen, D. (2019, October). DeepBundle: fiber bundle parcellation with graph convolution neural networks. In International Workshop on Graph Learning in Medical Imaging (pp. 88–95). Springer, Cham.
- Maddah, M., Crimson, W. E. L., & Warfield, S. K. (2006, April). Statistical modeling and EM clustering of white matter fiber tracts. In 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2006. (pp. 53–56). IEEE.
- Maddah, M., Mewes, A. U., Haker, S., Grimson, W. E. L., & Warfield, S. K. (2005, October). Automated atlas-based clustering of white matter fiber tracts from DTMRI. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 188–195). Springer, Berlin, Heidelberg.
- Maier-Hein, K. H. et al. (2015). Tractography Challenge ISMRM 2015 Data. <https://doi.org/10.5281/zenodo.572345>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111–3119).
- Olivetti, E., Nguyen, T. B., & Garyfallidis, E. (2012, July). The approximation of the dissimilarity projection. In 2012 Second International Workshop on Pattern Recognition in NeuroImaging (pp. 85–88). IEEE.
- Oota, S. R., Rowtula, V., Gupta, M., & Bapi, R. S. (2019, July). StepEncog: A convolutional LSTM autoencoder for near-perfect

- fMRI encoding. In 2019 International Joint Conference on Neural Networks (IJCNN) (pp. 1–8). IEEE.
- Sporns, O., Tononi, G., & Kötter, R. (2005). The human connectome: A structural description of the human brain. *PLoS Computational Biology*, 1(4), e42.
- Sundermeyer, M., Schlüter, R., & Ney, H. (2012). LSTM neural networks for language modeling. In Thirteenth annual conference of the international speech communication association.
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Advances in neural information processing systems (pp. 3104–3112).
- Tournier, J. D., Yeh, C. H., Calamante, F., Cho, K. H., Connolly, A., & Lin, C. P. (2008). Resolving crossing fibres using constrained spherical deconvolution: Validation using diffusion-weighted imaging phantom data. *NeuroImage*, 42(2), 617–625.
- Ugurlu, D., Firat, Z., Türe, U., & Unal, G. (2018). Neighborhood resolved fiber orientation distributions (NRFOD) in automatic labeling of white matter fiber pathways. *Medical Image Analysis*, 46, 130–145.
- Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).
- Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E., Yacoub, E., Ugurbil, K., & Wu-Minn HCP Consortium. (2013). The WU-Minn human connectome project: An overview. *NeuroImage*, 80, 62–79.
- Wakana, S., Caprihan, A., Panzenboeck, M. M., Fallon, J. H., Perry, M., Gollub, R. L., & Mori, S. (2007). Reproducibility of quantitative tractography methods applied to cerebral white matter. *NeuroImage*, 36(3), 630–644.
- Wassermann, D., Makris, N., Rathi, Y., Shenton, M., Kikinis, R., Kubicki, M., & Westin, C. F. (2016). The white matter query language: A novel approach for describing human white matter anatomy. *Brain Structure and Function*, 221(9), 4705–4721.
- Wasserthal, J., Neher, P., & Maier-Hein, K. H. (2018). TractSeg-Fast and accurate white matter tract segmentation. *NeuroImage*, 183, 239–253.
- Wolf, T., Chaumond, J., Debut, L., Sanh, V., Delangue, C., Moi, A., ... & Rush, A. M. (2020, October). Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 38–45).
- Wu, Y., Hong, Y., Ahmad, S., Lin, W., Shen, D., Yap, P. T., & UNC/UMN Baby Connectome Project Consortium. (2020, October). Tract dictionary learning for fast and robust recognition of fiber bundles. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 251–259). Springer, Cham.
- Yeh, F. C., Badre, D., & Verstynen, T. (2016). Connectometry: A statistical approach harnessing the analytical potential of the local connectome. *NeuroImage*, 125, 162–171.
- Yendiki, A., Panneck, P., Srinivasan, P., Stevens, A., Zöllei, L., Augustinack, J., & Fischl, B. (2011). Automated probabilistic reconstruction of white-matter pathways in health and disease using an atlas of the underlying anatomy. *Frontiers in Neuroinformatics*, 5, 23.
- Zhang, S., Correia, S., & Laidlaw, D. H. (2008). Identifying white-matter fiber bundles in DTI data using an automated proximity-based fiber-clustering method. *IEEE Transactions on Visualization and Computer Graphics*, 14(5), 1044–1053.
- Zhang, F., Karayumak, S. C., Hoffmann, N., Rathi, Y., Golby, A. J., & O'Donnell, L. J. (2020). Deep white matter analysis (DeepWMA): Fast and consistent tractography segmentation. *Medical Image Analysis*, 65, 101761.
- Zhong, S., Chen, Z., & Egan, G. (2020). Auto-encoded latent representations of white matter streamlines. In 28th Virtual Conference & Exhibition of the International Society for Magnetic Resonance in Medicine (ISMRM). International Society for Magnetic Resonance in Medicine (Vol. 8).
- Ziyan, U., Sabuncu, M. R., Grimson, W. E. L., & Westin, C. F. (2009). Consistency clustering: A robust algorithm for group-wise registration, segmentation and automatic atlas construction in diffusion MRI. *International Journal of Computer Vision*, 85(3), 279–290.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.