Hierarchical Temporal Memory Network for Medical Image Processing

Tao ZHOU¹, Zhen-zhen ZHANG², Yuan-yuan CHEN³,*

¹University of Electronic Science and Technology of China, Chengdu, China
²College of Electrical & Information Engineering, Southwest Minzu University, Chengdu, China
³College of Computer Science, Sichuan University, Chengdu, China

*Corresponding author

Keywords: Hierarchical temporal memory; medical image classification; medical image sequences.

Abstract. Medical image segmentation is a basic step in medical image analysis, especially for medical image sequences such as CT sequences. Automated segmentation of different objects in the medical image sequences is of great significance to the 3D reconstruction of medical images. A novel image recognition method which can be implemented in automated medical image segmentation is introduced. In contrast with other algorithm, HTM (hierarchical temporal memory) is a network using a spatio-temporary hierarchy that works as our neocortex. The algorithm referred in this paper consists of three main steps. Firstly, a four level hierarchical structure is established. Secondly, create frames by animating gray images to train the HTM network. During the learning phase, the nodes in HTM network build its representations spatial pooler and temporal pooler for inputs. Thirdly, test with dataset to get the inference result for classification. The results show that the proposed method can recognize the “middle slice” for different given objects when process the medical image sequences.

Introduction

Medical image sequences segmentation [1-4] is the first and fundamental step in many computer-assisted hepatectomy, such as tumor detection in CT image sequences, or 3D reconstruction for some certain organs by medical image sequences. Region-based [5-6] and contour-based [7] segmentation methods are the two main groups in this research topic, as well as prior knowledge based methods and statistical based methods. Most of these methods are semi–automatic. Take liver in the CT image sequences for example, some classical algorithms are only available for the middle slice image, in which the liver area is near to the largest size. We could more easily find the liver boundary in the middle slice than others because of its similar contour in different slice groups. If we start from one of the middle slices and get the segmentation result, then process the other slices in two batches that one move in the direction towards the head and another in the direction towards the feet, all the slices could be processed automatically. Therefore, how to determine the middle slice takes an important role in automated segmentation of medical image sequences.

Segmentation for single image usually partitions an image into distinct regions containing each pixels with similar attributes, which is an important research area in image processing, and has broad applications. A number of results have been achieved in the segmentation of natural images. Different from natural image segmentation tasks, object segmentation in medical images, such as liver in CT images, is still a difficult problem for its noise, ambiguous boundaries, low intensity contrast, and large variability inter-patient and intra-patient in geometric properties of objects. The classical segmentation methods usually needs an initial seed point or contour. Take region-growing algorithm for example. The process of region-growing algorithm includes two steps: firstly, select a seed point which locates in the region that of interest; secondly, using the algorithm to obtains the whole region.

Therefore, the selection of the initial point or contour affects the segmentation results greatly. When processing the medical image sequences, we cannot do an automated segmentation using
region-growing algorithm since it is hard to determine the seed point in each slice automatically. An idea is that, if we can find out the middle slice for the organ, that is, in which the area of the organ is relative large. The problem is, before segmentation, it is difficult to know the area of the organ. However, when thinking about our brain, a common student could easily find the middle slice after learning large amount of slices images. Therefore, we intend to find a novel algorithm about this open problem which meets the way that our brain works. There are lots of method based on the principle of brain work [8-9]. Among these theorems, Jeff Hawkin’s theorem on intelligence [10] proposed that, all the intelligence comes from neocortex. According to the structure of neocortex, Hierarchical Temporal Memory (HTM) is proposed. HTM is a theory that is considered that it builds a model of the world using a spatio-temporal hierarchy.

In this paper, an image recognition based on HTM framework which can identify the middle slice in the CT slice group is proposed. In Section 2, we describe the HTM network, including the HTM hierarchy, the learning method and the inference method. The details of the learning and inference algorithm can be found in [11]. In section 3, we describe the procedure and results of the experiments of applying the proposed algorithm on real CT data as well as the related performance discussion. Finally in section 4, we conclude by summarizing the approach and pointing out possible future pursuits in medical image processing and research on HTM algorithm.

Hierarchical Temporal Memory (HTM) Network
The HTM network creates a system for visual pattern recognition. This algorithm can be thought of as a memory system that exploits the hierarchical structure of the visual world. In this section, we introduce the HTM network in detail, including its architecture, the process of learning and inference.

![Figure 1. The architecture of HTM network.](image)

Architecture of HTM Network
The architecture of an HTM network is show in Fig. 1. As can be seen in fig. 1, the network builds a hierarchical structure to get invariant representations of complex patterns which are learned by invariant representation of simpler patterns.

The basic element in HTM network is node. The nodes that are closest to the raw inputs are at the lowest level of the hierarchy and the nodes that are furthest away from the raw inputs are at the top level of the network. In Fig. 2, the input images go directly to level-1 nodes. The top-level of the hierarchy has a single node.

Furthermore, the nodes in the HTM network are also the basic algorithm and memory modules of the network. Each node has its inputs and outputs and contains memory which is used to store
information about this node. A node learns invariant representations of its input using some learning algorithm. Usually, all the nodes in an HTM network share the same learning algorithm. The learning algorithms within a node observes the input space and abstracts some characteristics about it to be stored in the nodes memory.

As most machine learning algorithms that based on the principle of brain work, the HTM network is operating in two distinct phase: learning and inference. Each node in the hierarchical structure uses the same learning and inference algorithm, that is as the same as neocortex. Learning phase entails storing spatial patterns and groups of the spatial patterns. Inference phase do recognition according to the memorized patterns. Bayesian belief propagation is used in this stage, and which might model the cortical computations.

Learning

We train the network layer by layer. During the learning stage, the network (in figure 1) is fed with movies of liver motions. The input to the network is a frame from the training movie. Learning phase entails storing spatial patterns and groups of the spatial patterns. All the nodes in this network form its invariant representations for inputs using two kinds of pooling mechanisms: spatial pooling and temporal pooling.

Spatial Pooling. As mentioned before, each node has a memory that stores patterns in its receptive field and it stores patterns and then gives each pattern a distinct label or pattern index.

Spatial pooling is used to remove noise from patterns and to form a finite number of quantization centers, or can be considered as a clustering process, which can reduce the number of the training data.

Any input pattern that fed to the node will be compared to the patterns that are already stored in the node. In noiseless situation, if an identical pattern is not in the memory, the new input pattern is added to the memory. Considering the noise, the input pattern is compared to the stored patterns according to some similarity metrics. If no similar pattern is found in the memory, the new pattern is added. The label or index of each pattern does not react any property of the pattern. All the patterns stored in a node are represented by a matrix, where each row of the matrix is a different pattern. Therefore, the number of rows in each node can be different. Take the nodes in the lowest for example, after training the spatial poolers in this layer, all the nodes store a matrix.

Temporal Pooling. Temporal pooler groups the quantization centers according to their temporal information. Two patterns that frequently occurred one after another would like to be grouped together because they may belong to the same pattern in fact.

In this period, HTM network learns transition probabilities and then do temporal grouping. Each node constructs and maintains a Markov graph. The vertices of this Markov graph correspond to the stored patterns. The link between two vertices is used to represent the number of transition events between the patterns corresponding to those vertices. After the Markov graph has been constructed, each node use Agglomerative Hierarchical Clustering (AHC) [13] to partition the vertices of the Markov graph into a set of temporal groups. The probability of transition between two patterns is used as a measure of the similarity between those patterns for the AHC algorithm. Each temporal group is a subset of the set of vertices of the Markov graph. After partitioning, the vertices of the same temporal group are highly likely to follow one another.

In conclusion, learning process in object recognition problem can be described in details as follow steps:

Step1: generating the training data by animating images to create movies.

Step2: Learning spatial pooler in all level one node: form the quantization centers by calculating the distance between input pattern and quantization centers memorized in the node.

Step3: After spatial pooler finished learning; it switched into inference mode to learn the temporal pooler. Each node computes the time-adjacency matrix, and discovers the Markov Chains by partitioning the matrix using an unsupervised algorithm.

Step4: Once temporal pooler finished learning; it switched into inference mode to learn the spatial pooler in its parent node until reaching the top level.
Step5: On the top level, a supervised signal is implemented to categorize the quantization centers.

Inference

After learning, the nodes memorized the invariant representations of the training images. Then, it can be used to do inference for other inputs. Inference phase do recognition according to the memorized patterns and Markov Chains in the node. Bayesian belief propagation is used in this stage, and which might model the cortical computations.

Experiments

Settings

**Dataset.** We experiment the data collected from SLIVER07 [14], which is considered as standard database used for developing segmentation algorithms of the liver. Each data has about 100 slices and the relevant segmentation result. We randomly select half of image sequences in this dataset for training, and the rest are used for testing. For more rapid computation, slice images which are 512×512 pixels were resized to 256×256 in preprocessing.

**Structure of the HTM Network.** We organize the HTM network which is used to model the image classification in a 4 level hierarchy. Level one of the network has 32×32 nodes arranged in a 32×32 grid. The input to a level 2 node comes from the outputs of 16 level 1 nodes. As a result, level two of the network has 64 nodes arrange in an 8×8 node. The input to a level 3 node comes from the outputs of 16 level 2 nodes. Then, level 3 of the network has 4 nodes, which will all go to a single level 4 node. Each level 1 node receive the information of input image that is divided into adjoining patches of 8×8 patches.

Training

In the training period, in order to teach HTM network the temporal information that for determine whether two different patterns belong to the same object or not, we create movies by simulating movies of smooth translation as shown in Figure 2.

![Figure 2. Some frames create by an image that contains the segmentation result.](image)

**Learning Result.** During learning stage, all nodes use the same algorithm. Movies are fed to the node repeatedly to learn the spatial pooler and the temporal pooler until all the nodes finished learning. Figure 2 shows some patterns memorized by spatial pooler in a node. Each coincidence in Figure 2 is only a small portion of the input image.

![Figure 3. Some patterns memorized by spatial pooler in a node.](image)

Some learned coincidences memorized in a level 2 node are shown in Figure 3. The white point stands for 1, and black point stands for 0. A coincidence in higher level is the co-occurrence of output of the temporal pooler of its child nodes.
Figure 4 shows some learned groups in a level one node. Those grouped coincidences would like to be considered as the same object.

**The Top Level.** Different from the common multi-object recognition problem, our aim is to identify whether the image is a middle slice or not. In other words, we get the similarity of the input image with a middle slice. It may be thought as a single-label categorization problem or find the best-match. In the top level, we let all patterns in spatial pooler into one group.

**Inference.** Some sets of slice images were fed to the system during inference stage. We get the output of temporal pooler on the top level. 32 slices images in a slice sequence is shown in Figure 5. We input those test images to learned HTM network. Three groups of the inference result of each slice image is calculated as Figure 6.
Discussion

The setting of the parameters that are used in this algorithm is still a problem exists in my current work. Furthermore, when and how to do the normalization is another question. In Bayesian belief propagation, each node would like to normalize its output. However, in gray image processing, normalization seems to change the result which is used to determine the similarity.

Conclusion

An HTM structure for medical image recognition is established is this paper. A middle slice can be discovered in the sequence using the HTM algorithm to some extent. For object-recognition problem, only the feed-forward message propagation in network is considered. In order to get the segmentation of the object, how to implement the top-down message will be researched next.

References


