

# GCA: A Coclustering Algorithm for Thalamo-Cortico-Thalamic Connectivity Analysis

Cui Lin, Shiyong Lu, Xuwei Liang, and Jing Hua  
Department of Computer Science  
Wayne State University

{ cuilin, shiyong, george.liang, jinghua } @wayne.edu

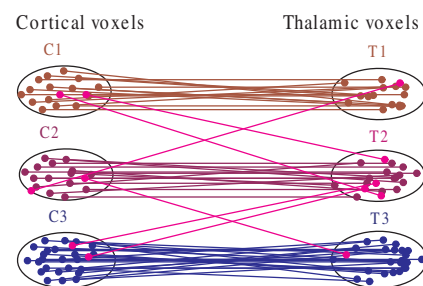
## Abstract

The reciprocal connectivity between the cerebral cortex and the thalamus in a human brain is involved in consciousness and related to various brain disorders, thus, in-vivo analysis of this connectivity is critically important for brain diagnosis and surgery planning. While existing work either focuses on fiber tracking analysis or on thalamic nuclei segmentation, to our best knowledge, no techniques yet exist for performing in-vivo analysis of thalamo-cortico-thalamic connectivity. In this paper, (i) we propose a new partitioning paradigm, called coclustering, to model this problem. In contrast to the traditional clustering paradigm, a coclustering procedure not only simultaneously partitions cortical voxels and thalamic voxels into groups, but also identifies the corresponding strong connectivities between the two classes of groups; (ii) we develop the first coclustering algorithm, Genetic Coclustering Algorithm (GCA), to solve the coclustering problem; and (iii) we apply GCA to perform in-vivo analysis of the thalamo-cortico-thalamic connectivity and produce a strikingly clear 3-D visualization of the seven thalamic nuclei groups as well as their connectivities to the corresponding cortical regions of a human brain.

## 1 Introduction

The reciprocal connectivity between the cerebral cortex and the thalamus in a human brain is involved in human consciousness and related to various brain disorders [13, 10, 1, 2]. Served as the “gate to consciousness”, each thalamus is subdivided into different neuronal aggregations or functionally specific clusters that are referred to as thalamic nuclei [16]. Each nuclei is in contact with different portions of the cerebrum, which transmits to the cerebral cortex all the impulses in order to enter human awareness. Since the insight to this connectivity is of importance for

understanding the mechanisms of brain functionalities and various brain disorders, in-vivo analysis of such connectivity is crucial for neuroscience in general and brain diagnosis and brain surgery planning in particular.



Strong connectivity between a thalamic nuclei and its corresponding cortical region

Unfortunately, to our best knowledge, no techniques yet exist for performing in-vivo analysis of the thalamo-cortico-thalamic connectivity. On one hand, existing brain analysis techniques either focuses on fiber tracking, which groups fibers into bundles [5, 17, 4, 3], or thalamus nuclei segmentation [11, 6, 12, 16]. On the other hand, traditional clustering algorithms are not applicable to this problem. As illustrated in Figure 1, each thalamic nuclei is strongly connected to a particular region of the cortex. For example, most voxels in thalamic nuclei  $T_2$  are connected to voxels of cortical region  $C_2$ , with only few connections to other cortical regions, and vice versa. Therefore, in order to perform an accurate in-vivo analysis of the thalamo-cortico-thalamic connectivity, a partitioning procedure is required that not only simultaneously partitions cortical voxels and thalamic voxels into groups, but also identifies the corresponding strong connectivities between the two classes of groups. However, traditional clustering algorithms, such as K-means [8], hierarchical clustering [9], and SVM [7], can

only perform partitioning on one class of objects, either on cortical voxels, on fibers, or on thalamic voxels, without the consideration of the above anatomical connectivity constraints. As a result, they will fail to identify accurately the corresponding strong connectivity between a thalamic nuclei and its corresponding cortical region.

The main contributions of this paper are:

1. We propose a new partitioning paradigm, called *co-clustering*, to model the thalamo-cortico-thalamic connectivity analysis problem. In contrast to the traditional clustering paradigm, a coclustering procedure not only simultaneously partitions cortical voxels and thalamic voxels into groups, but also identifies the corresponding strong connectivities between the two classes of groups.
2. We develop the first coclustering algorithm, Genetic Coclustering Algorithm (GCA), to solve the coclustering problem.
3. We apply GCA to perform in-vivo analysis of the thalamo-cortico-thalamic connectivity and produce a strikingly clear 3-D visualization of the seven thalamic nuclei groups as well as their connectivities to the corresponding cortical regions of a human brain. To our best knowledge, this is the first technique for the in-vivo analysis of the thalamo-cortico-thalamic connectivity.

*Organization.* The rest of the paper is organized as follows: Section 2 formalizes the coclustering model for the thalamo-cortico-thalamic connectivity analysis. Section 3 proposes our first coclustering algorithm, Genetic Coclustering Algorithm (GCA), to solve the coclustering problem. Section 4 presents sample pictures of the 3-D visualization of the analysis results. Finally, Section 5 concludes the paper and comments on future directions of this research.

## 2 The Coclustering Model

In this section, we present our coclustering model, which models the thalamo-cortico-thalamic analysis problem. In this model, the structure of the cortex and thalamus is represented as a bipartite graph  $G = (C, T, F)$  as illustrated in Figure 1, where  $C$  is a set of cortical voxels,  $T$  is a set of thalamic voxels, and  $F$  is the set of fibers connecting  $C$  and  $T$ . Although not required by our model, the working hypothesis is that most thalamic voxels within one thalamic nuclei usually connect to a specific cortical region, and the connectivities to other cortical regions are relatively weaker, and vice versa. The goal of a coclustering procedure is to group the two classes of objects simultaneously while minimizing the cross-connectivity cost between them.

More specifically, a coclustering procedure will partition both classes of objects into  $K$  groups so that (1) for each class, similar objects are within the same group, while dissimilar objects are in different groups, (2) there is a one-to-one correspondence between the clusters in the first class (Class of Cortical voxels) and the second class (Class of Thalamic voxels); the corresponding cluster to a cluster is called its *spouse cluster*, and (3) the total cross-connectivity cost which results from those edges between a cluster in one class and a non-spouse cluster in the other class is minimized.

To achieve the above goals, we define several notions. First, we define the centroid of a cluster and its Within-Cluster Variation (*WCV*) to quantify the similarity of objects within one cluster.

The centroid of a cortical cluster  $C_K$  is defined as:

$$\vec{\mu}_K = \frac{\sum_{\vec{X}_n \in C_K} \vec{X}_n}{|C_K|}$$

where  $|C_K|$  represents the number of cortical voxels in cluster  $C_K$ . Similarly, the centroid of a thalamic cluster  $T_K$  is defined as:

$$\vec{\nu}_K = \frac{\sum_{\vec{Y}_n \in T_K} \vec{Y}_n}{|T_K|}$$

where  $|T_K|$  represents the number of thalamic voxels in cluster  $T_K$ .

The Within-Cluster Variation of cortical cluster  $C_K$  is defined as:

$$WCV(C_K) = \sum_{\vec{X}_n \in C_K} d(\vec{X}_n, \vec{\mu}_K)$$

where  $d(\vec{X}_n, \vec{\mu}_K)$  is the Euclidean distance between the cortical voxel  $\vec{X}_n$  and the centroid  $\vec{\mu}_K$  of cortical cluster  $C_K$ . Similarly, the Within-Cluster Variation of thalamic cluster  $T_K$  is defined as:

$$WCV(T_K) = \sum_{\vec{Y}_n \in T_K} d(\vec{Y}_n, \vec{\nu}_K)$$

where  $d(\vec{Y}_n, \vec{\nu}_K)$  is the Euclidean distance between the thalamic voxel  $\vec{Y}_n$  and the centroid  $\vec{\nu}_K$  of the  $K$ th thalamic cluster.

Second, we define the Total Within-Cluster Variation to quantify the quantity of a particular partitioning. The Total Within-Cluster Variation (*TWCV*) of a cortical partition  $(C_1, \dots, C_K)$  is defined as

$$\begin{aligned} TWCV(C_1, \dots, C_K) &= \sum_{k=1}^K WCV(C_k) \\ &= \sum_{k=1}^K \sum_{\vec{X}_n \in C_k} \sum_{d=1}^D (X_{n,d} - \mu_{k,d})^2 \end{aligned}$$

$$= \sum_{k=1}^K \sum_{d=1}^D X_{n_d}^2 - \sum_{k=1}^K \frac{1}{|C_k|} \sum_{d=1}^D (SCF_{k_d})^2$$

where  $SCF_{k_d}$  is the sum of the  $d$ th feature of all voxels in  $C_k$ . Similarly, the Total Within-Cluster Variation of a thalamic partition  $(T_1, \dots, T_K)$  is defined as

$$\begin{aligned} & TWCV(T_1, \dots, T_K) \\ &= \sum_{k=1}^K \sum_{d=1}^D Y_{n_d}^2 - \sum_{k=1}^K \frac{1}{|T_k|} \sum_{d=1}^D (STF_{k_d})^2 \end{aligned}$$

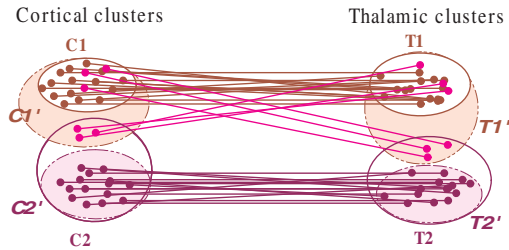
where  $STF_{k_d}$  is the sum of the  $d$ th feature of all voxels in  $T_k$ .

Third, in order to minimize the cross-connectivity cost, for each thalamic cluster, we define the set of cortical voxels that are connected to it as its *shaded thalamic cluster*. More formally, given a thalamic partition  $(T_1, \dots, T_K)$ , the shaded cortical cluster  $C'_K$  ( $k = 1, \dots, K$ ) is defined as:

$$C'_K = \{c | c \in C, \exists t \in T_K, (c, t) \in F\}$$

A shaded thalamic cluster can be defined similarly.

For example, in Figure 2, all the cortical voxels that are connected to voxels in thalamic cluster  $T_1$  forms the shaded cortical cluster  $C'_1$ , while all thalamic voxels that are connected to the voxels in cortical cluster  $C_1$  forms the shaded thalamic cluster  $T'_1$ . In an ideal coclustering, as  $C'_2$  and  $T'_2$ , a shaded cluster should *coincide* with the corresponding spouse cluster. However, this is not always the case in general. The cross-connectivity cost can be characterized by the disagreement between shaded clusters and spouse clusters and quantified by the total within cluster variance of shaded clusters with respect to their corresponding spouse clusters, called  $STWCV$ , that is defined as follows.



Shaded cortical and thalamic clusters

The Shaded Within-Cluster Variation ( $SWCV$ ) of cortical cluster  $C'_K$  is defined as:

$$SWCV(C'_K) = \sum_{\vec{X}'_n \in C'_K} d(\vec{X}'_n, \vec{\mu}'_K)$$

Note that, instead of using the centroid of  $C'_K$ , the centroid of the  $C_K$  is used to calculate  $SWCV(C'_K)$ . The intuition is that, in an ideal partitioning, the shaded partition  $C'_1 \dots C'_K$  should mostly coincide with  $C_1 \dots C_K$ .

Similarly to the shaded thalamic partition, the centroid of the  $T_K$  is used to calculate  $SWCV(T'_K)$  for finding the optimal solution. The  $SWCV$  of thalamic cluster  $T'_K$  is defined as:

$$SWCV(T'_K) = \sum_{\vec{Y}'_n \in T'_K} d(\vec{Y}'_n, \vec{\nu}'_K)$$

The Shaded Total Within-Cluster Variation ( $STWCV$ ) of cortical partition  $(C'_1, \dots, C'_K)$  is defined as:

$$\begin{aligned} STWCV(C'_1, \dots, C'_K) &= \sum_{k=1}^K SWCV(C'_k) \\ &= \sum_{k=1}^K \sum_{\vec{X}'_n \in C'_k} \sum_{d=1}^D (X'_{n_d} - \mu_{k_d})^2 \end{aligned}$$

Similarly, the Shaded Total Within-Cluster Variation of thalamic partition  $(T'_1, \dots, T'_K)$  is defined as:

$$\begin{aligned} STWCV(T'_1, \dots, T'_K) &= \sum_{k=1}^K SWCV(T'_k) \\ &= \sum_{k=1}^K \sum_{\vec{Y}'_n \in T'_k} \sum_{d=1}^D (Y'_{n_d} - \nu_{k_d})^2 \end{aligned}$$

**Statement of the problem.** Finally, the coclustering problem can be formally stated as follows: given a cluster number  $K$ , a bipartite  $G = (C, T, F)$ , and a distance metric  $d$  for nodes in  $C$  and  $T$ , partition  $(C, T)$  into  $K$  cluster pairs  $(C_1, T_1), (C_2, T_2), \dots, (C_K, T_K)$ , such that the following objective function  $OTWCV$  is minimized:

$$\begin{aligned} OTWCV &= TWCV(C_1, \dots, C_K) + TWCV(T_1, \dots, T_K) \\ &\quad + STWCV(C'_1, \dots, C'_K) + STWCV(T'_1, \dots, T'_K) \end{aligned}$$

### 3 Our Proposed GCA Algorithm

In this section, we propose the first coclustering algorithm, *Genetic Coclustering Algorithm (GCA)*, to solve the coclustering problem. It is based on the genetic algorithm approach [15] by working on a coding of the solution space over which the search has to be performed. These encoded solutions are called *chromosomes* and the objective function value  $OTWCV$  can be calculated for each solution according to its definition in Section 2. Each solution is encoded by a string of symbols and GCA evolves solutions over generations. During each generation, GCA produces a new population from the current population by applying genetic operators viz., selection, mutation, and K-means operator. Each solution in the population is associated with a figure of merit (fitness value) depending on  $OTWCV$ . The selection operator selects a solution from the current population

for the next population with a probability proportional to its fitness value. The mutation operator toggles each position in a string with a probability, called the *mutation probability* ( $MP$ ). Finally the K-means operator is introduced to speed up GCA's convergence to the global optimum. The rationale that we use a genetic algorithm approach is that it has been shown that genetic algorithms maintaining the best discovered solution either before or after the selection operator asymptotically converge to the global optimum.

More specifically, GCA maintains a population (set) of  $Z$  coded solutions, where  $Z$  is a parameter specified by the user. Each solution is coded by a string  $(c_1 \cdots c_N, t_1 \cdots t_N)$  for  $N$  cortical voxels and  $N$  thalamic voxels, where each  $c_i$  or  $t_i$ , called an *allele*, denotes a cortical voxel or a corresponding thalamic voxel. Each allele takes a value from  $1, \dots, K$ , representing the cluster number to which the voxel belongs.

GCA starts with the initialization phase, which generates the initial population  $P_0$ , and then run the selection operator, the mutation operator, and the K-means operator sequentially on the current population  $P_i$  to obtain the next generation  $P_{i+1}$ . This sequence of operators are run iteratively to produce one generation after another until a termination condition is reached.

During evolution, some solutions in which some cortical clusters or thalamic clusters are empty might be produced. These solutions are called *illegal* solutions. To deal with illegal solutions, we define the notion of *legality ratio*. Given a solution  $S_z$  that is encoded by  $(c_1 \cdots c_N, t_1 \cdots t_N)$ , let  $k_1$  be the number of nonempty cortical clusters and  $k_2$  be the number of nonempty thalamic clusters in  $S_z$ , the *legality ratio* of  $S_z$  is defined as:

$$e(S_z) = (k_1 + k_2) / 2 * K$$

A solution  $S_z$  is *legal* if  $e(S_z) = 1$  and *illegal* otherwise. Although illegal solutions are not needed eventually, they are helpful for GCA's convergence to the global optimum.

### 3.1 Phase 1: Initialization Operator

GCA starts with the initialization phase, which randomly generates the initial population  $P_0$  of  $Z$  solutions, where  $Z$  is a parameter specified by the user. Each allele  $c_i$  in a solution  $(c_1 \cdots c_N, t_1 \cdots t_N)$  is initialized to a cluster number randomly selected from the uniform distribution over the set  $\{1, 2, \dots, K\}$ ;  $c_i$  is initialized to a value equal to its corresponding  $t_i$ .

Illegal solutions are permitted but are considered as the most undesirable solutions by defining their OTWCVs as  $+\infty$  and assigning them with low fitness values (will be defined in the next subsection). With a low fitness value, an illegal solution will have a lower probability for survival.

This flexibility of allowing illegal solutions in the evolution process not only avoids the overhead of illegal solution elimination and thus improves the time performance of the algorithm, but also provides the chances for illegal solutions to mutate to legal solutions during the mutation phase.

### 3.2 Phase 2: the Selection Operator

We use proportional selection for the selection operator in which, the population of the next generation is determined by  $Z$  independent random experiments. Each experiment randomly selects a solution from the current population  $(S_1, S_2, \dots, S_z)$  according to the probability distribution  $(p_1, p_2, \dots, p_z)$  defined by

$$p_z = \frac{F(S_z)}{\sum_{z=1}^Z F(S_z)}$$

where  $F(S_z)$  denotes the fitness value of solution  $S_z$  with respect to the current population and is defined as follows.

$$F(S_z) = (OTWCV_{max} - OTWCV_{S_z}) * e(S_z)$$

where  $OTWCV_{max}$  is the maximal value of  $OTWCV$  that has been encountered till the present generation.

The intuition behind this fitness function is that each solution will have a probability to survive by being assigned with a positive fitness value, but a solution with a smaller  $OTWCV$  has a greater fitness value and hence has a higher probability to survive. Illegal solutions are also allowed to survive but with lower fitness values than all legal solutions in the current population. Finally, illegal strings that have more empty clusters are assigned with smaller fitness values and hence have lower probabilities for survival.

### 3.3 Phase 3: the Mutation Operator

The mutation operator is very useful for GCA to reach better solutions based on the evolutionary theory that offsprings produced by mutations might be superior to their parents. More importantly, the mutation operator performs the functionality of shaking the algorithm out of a local optimum and of moving it towards the global optimum [14].

Given a solution  $S_z = (c_1 \cdots c_N, t_1 \cdots t_N)$ , the mutation operator mutates each allele  $c_i$  or  $t_i$  to new values  $k_1$  and  $k_2$  simultaneously (might be equal to  $c_i t_i$ ), where  $k_1$  and  $k_2$  are numbers randomly selected from  $(1, 2, \dots, K)$  with probability  $MP$  respectively and independently where  $0 < MP < 1$  is a parameter called the *mutation probability* that is specified by the user. To define the probability mass function of mutation, we first define the similarity measurement for both cortical voxels and thalamic voxels.

The similarity measurement between cortical elements is defined as:

$$CS(\vec{X}_n, \vec{C}_{k_1}) = \max_{k=1}^K \{d(\vec{X}_n, \vec{C}_k)\} - d(\vec{X}_n, \vec{C}_{k_1})$$

where  $d(\vec{X}_n, \vec{C}_{k_1})$  is the Euclidean distance between cortical voxel  $\vec{X}_n$  and the centroid  $\vec{C}_{k_1}$  of the  $k_1$ th cortical cluster.

The similarity measurement between thalamic elements is defined as:

$$TS(\vec{Y}_n, \vec{T}_{k_2}) = \max_{k=1}^K \{d(\vec{Y}_n, \vec{T}_k)\} - d(\vec{Y}_n, \vec{T}_{k_2})$$

where  $d(\vec{Y}_n, \vec{T}_{k_2})$  is the Euclidean distance between thalamic voxel  $\vec{Y}_n$  and the centroid  $\vec{T}_{k_2}$  of the  $k_2$ th cluster on thalamus.

During mutation, we replace each allele  $c_i t_i$  by  $k_1 k_2$  for  $i = (1, \dots, N)$  simultaneously where  $k_1$  and  $k_2$  are selected from  $(1, \dots, K)$  with the probability distribution  $(p_{11}, p_{12}, \dots, p_{k_1 k_2}, \dots, p_{KK})$  in which

$$p_{k_1 k_2} = \frac{CS(\vec{X}_n, \vec{C}_{k_1}) + CS(\vec{X}_n, \vec{C}_{k_2}) + TS(\vec{Y}_n, \vec{T}_{k_2}) + TS(\vec{Y}_n, \vec{T}_{k_1})}{\sum_{k_1=1}^K \sum_{k_2=1}^K (CS(\vec{X}_n, \vec{C}_{k_1}) + CS(\vec{X}_n, \vec{C}_{k_2}) + TS(\vec{Y}_n, \vec{T}_{k_2}) + TS(\vec{Y}_n, \vec{T}_{k_1}))}$$

The distance between a voxel and an empty cluster is defined to be 0 to increase the chance of converting an illegal solution to a legal one. The above mutation operator is defined such that (1)  $\vec{X}_n$  and  $\vec{Y}_n$  might be reassigned randomly to each cluster with a positive probability; (2) the probability of changing allele value  $c_i t_i$  to a cluster number  $k_1 k_2$  is greater if  $\vec{X}_n$  and  $\vec{Y}_n$  are closer to the centroid of the  $k_1$ th cortical cluster and the  $k_2$ th thalamic cluster. The first property ensures that an arbitrary solution, including the global optimum, might be generated by the mutation from the current solution with a positive probability; the second property encourages that each  $\vec{X}_n$  and  $\vec{Y}_n$  are moving towards a closer cluster with a higher probability.

### 3.4 Phase 4: the K-means Operator

In order to speed up the convergence process, we introduce the following K-means operator based on the idea of the classical K-means algorithm [8]. We treat illegal solutions and legal solutions separately.

For an illegal solution  $S_z = (c_1 \dots c_N, t_1 \dots t_N)$ , we replace each  $c_i t_i$  by new values  $k_1 k_2$  for  $i = 1, \dots, N$  simultaneously, where  $k_1$  and  $k_2$  are numbers selected from  $(1, 2, \dots, K)$  such that the value returned by the following  $\tau$  function is minimized.

$$\tau(k_1, k_2) = d(\vec{X}_n, \vec{C}_{k_1}) + d(\vec{X}_n, \vec{C}_{k_2}) + d(\vec{Y}_n, \vec{T}_{k_2}) + d(\vec{Y}_n, \vec{T}_{k_1})$$

The distance between a voxel and an empty cluster centroid is defined to be 0 with the effort to convert an illegal solution to a legal one. For a legal solution  $S_z = (c_1 \dots c_N, t_1 \dots t_N)$ , we replace each  $c_i t_i$  by new values

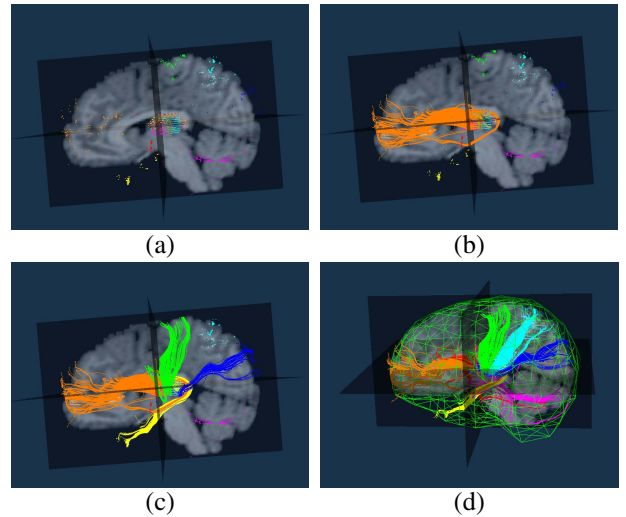
$k_1 k_2$  for  $i = 1, \dots, N$  simultaneously, where  $k_1$  is a number selected from  $(1, 2, \dots, K)$  such that the value returned by the following  $\lambda$  function is minimized.

$$\lambda(k) = d(\vec{X}_n, \vec{C}_k) + d(\vec{X}_n, \vec{C}'_k) + d(\vec{Y}_n, \vec{T}_k) + d(\vec{Y}_n, \vec{T}'_k)$$

Here, we replace  $c_i t_i$  to the same new cluster number to reduce cross-connectivity cost.

## 4 3-D Visualization of the GCA results

By virtue of DTI techniques, our GCA results can be visualized and analyzed in different levels of details: Figure 3-(a) shows the clustered thalamic voxels and cortical voxels; two local views show the specific thalamo-cortico-thalamic clusters as well as their fiber tracts, as shown in Figure 3-(b) and (c). The global view, in Figure 3-(d), presents the whole picture of all the connectivity between cortical and thalamic voxels. From our visualized results,



(a) Clustered thalamic voxels and cortical voxels; (b) A specific thalamo-cortico-thalamic cluster; (c) Four thalamo-cortico-thalamic clusters; (d) Global view of clustered thalamo-cortico-thalamic connectivity

each spouse cluster is defined by one color, so there are totally seven colored clusters on both the thalamus and cortex. Each voxel within one cluster is rendered by its belonging cluster's color. As shown in Figure 3-(b) and (c), The groups of tiny colorful points with varied colors represent the voxels in different clusters with respect to their 3D space position. We use the same color to render the fibers if their starting voxels on thalamus and ending voxels on cortex belong to one spouse cluster. For example, in Figure

3-(b), yellow fibers connect the yellow thalamic voxels and their yellow spouse cluster on cortex.

The visualized results show that totally 2301 connections linked the cortical areas with 7 thalamic nuclei. Each thalamic nuclei contains the varied number of connectional associations - fiber tracts, from the highest 545 to the lowest 110. Furthermore, our results have shed light on the gross connectional organization of thalamo-cortico-thalamic connectivity and localization of the “place” of individual structures with the overall scheme.

## 5 Conclusions and Future Work

In this paper, we defined our coclustering problem and we applied our first coclustering algorithm, Generic Coclustering Algorithm, to the in-vivo analysis of the thalamo-cortico-thalamic connectivity.

Our visualized results show that our algorithm can not only segment the thalamic voxels into seven nuclei groups, but also their anatomic connectivity to their seven corresponding regions on the surface of the brain. Our ability to identify cortico-thalamocortical connectivity by DTI noninvasive imaging makes it possible to perform a more comprehensive quantitative analysis of thalamo-cortico-thalamic connections, a more accurate neurosurgical planning, and an improved anatomical localization of functional activation.

Although the coclustering problem is motivated by the need of thalamo-cortico-thalamic connectivity analysis, we expect that it will have a wide range of applications. In the future, we will apply our GCA to social science to analyze the interactions between male and female communities and to bioinformatics to analyze the relationships between transcription factors and their binding sites.

## Acknowledgment

This work was supported in part by the Michigan Technology Tri-Corridor basic research grant MTTC05-135/GR686. The authors would like to thank Dr. Otto Muzik at the PET center of the school of Medicine, Wayne State University, for providing the human brain DTI datasets.

## References

- [1] M. S. Bagary, J. Foong, M. Maier, G. duBoulay, G. J. Barker, D. H. Miller, F.R.C.P., and M. A. Ron. A magnetization transfer analysis of the thalamus in schizophrenia. *The Journal of Neuropsychiatry and Clinical Neurosciences*, 14:443–448, November 2002.
- [2] R. B. Banati, J. Newcombe, A. Gunn, F. Turkheimer, F. Hepner, G. Price, F. Wegner, G. Giovannoni, D. H. Miller, G. D. Perkin, T. Smith, A. K. Hewson, G. Bydder, G. W. Kreutzberg, T. Jones, M. L. Cuzner, and R. Myers. The peripheral benzodiazepine binding site in the brain in multiple sclerosis. *Brain*, 123:2321–2337, November 2000.
- [3] A. Brun, H. Knutsson, H.-J. Park, M. E. Shenton, and C.-F. Westin. Clustering fiber traces using normalized cuts. *Medical Image Computing and Computer-Assisted Intervention*, 1:368 – 375, 2004.
- [4] A. Brun, H.-J. Park, H. Knutsson, and C.-F. Westin. Coloring of DT-MRI fiber traces using laplacian eigenmaps. *Lecture Notes in Computer Science 2809*, 1:564, 2003.
- [5] I. Corouge, S. Gouttard, and G. Gerig. Towards a shape model of white matter fiber bundles using diffusion tensor MRI. *International Symposium on Biomedical Imaging, Conf. Proc.*, pages 344–347, 2004.
- [6] S. C. Deoni, M. J. Josseau, B. K. Rutt, and T. M. Peters. Visualization of thalamic nuclei on high resolution, multi-averaged t1 and t2 maps acquired at 1.5 t. *Human Brain Mapping*, Jul;25(3):353–9, 2005.
- [7] D. T.-N. Do and F. Poulet. Incremental SVM and visualization tools for bio-medical data mining. 2003.
- [8] J. Han and M. Kamber. *Data Mining*, pages 349–353. I-55860-489-8. Morgan Kaufmann Publishers, 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205, USA, 2001.
- [9] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*, pages 308–314. 0-262-08290-X. Massachusetts Institute of Technology, San Diego, USA, 2001.
- [10] J. M. Henderson, K. Carpenter, H. Cartwright, and G. M. Halliday. Loss of thalamic intralaminar nuclei in progressive supranuclear palsy and parkinson’s disease: clinical and therapeutic implications. *Brain*, 123:No. 7, 1410–1421, July 2000.
- [11] T. J. Imig and A. Morel. Tonotopic organization in lateral part of posterior group of thalamic nuclei in the cat. *Journal of Neurophysiology*, 53:836–851, 1985.
- [12] L. Jonasson, P. Hagmann, C. Pollo, X. Bresson, C. R. Wilson, R. Meuli, and J.-P. Thiran. A level set method for segmentation of the thalamus and its nuclei in DT-MRI. *Elsevier Science*, 2005.
- [13] C. Juhasz, D. C. Chugani, O. Muzik, A. Shah, J. Shah, C. Watson, A. Canady, and H. T. Chugani. Relationship of flumazenil and glucose pet abnormalities to neocortical epilepsy surgery outcome. *American Academy of Neurology*, 56:1650–1658, 2001.
- [14] G. Rudolph. Convergence analysis of canonical genetic algorithms. *Neural Networks, IEEE Transactions*, 5:96 – 101, 1994.
- [15] M. U and B. S. Genetic algorithm based clustering technique. *IEEE Trans on Evolutionary Computation*, 3:103C112, 1999.
- [16] M. Wiegell, D. T. H. Larsson, and V. Wedeen. Automatic segmentation of thalamic nuclei from diffusion tensor magnetic resonance imaging. *NeuroImage*, 19:391–401, 2003.
- [17] Z. Ding, J.C. Gore, and A.W. Anderson. Classification and quantification of neuronal fiber pathways using diffusion tensor MRI. *Magn. Res. Med.*, 49:716C 721, 2003.