UCLA
Department of Statistics Papers

Title
A Learning Based Algorithm for Automatic Extraction of the Cortical Sulci

Permalink
https://escholarship.org/uc/item/1q04v6g4

Authors
Zheng, Songfeng
Tu, Zhuowen
Yuille, Alan
et al.

Publication Date
2006-04-09

Peer reviewed

A Learning Based Algorithm for Automatic Extraction of the Cortical Sulci

Songfeng Zheng[1], Zhuowen Tu[2], Alan L. Yuille[1], Allan L. Reiss[3],
Rebecca A. Dutton[2], Agatha D. Lee[2], Albert M. Galaburda[4],
Paul M. Thompson[2], Ivo Dinov[1,2], and Arthur W. Toga[2]

[1] Department of Statistics, UCLA, Los Angeles, CA, USA
`{sfzheng, ztu, yuille}@stat.ucla.edu`
[2] Laboratory of Neuro Imaging, UCLA Medical School, Los Angeles, CA, USA
[3] School of Medicine, Stanford University, Stanford, CA, USA
[4] School of Medical, Harvard University, Cambridge, MA, USA

Abstract. This paper presents a learning based method for automatic extraction of the major cortical sulci from MRI volumes or extracted surfaces. Instead of using a few pre-defined rules such as the mean curvature properties, to detect the major sulci, the algorithm learns a discriminative model by selecting and combining features from a large pool of candidates. We used the Probabilistic Boosting Tree algorithm [16] to learn the model, which implicitly discovers and combines rules based on manually annotated sulci traced by neuroanatomists. The algorithm almost has no parameters to tune and is fast because of the adoption of integral volume and 3D Haar filters. For a given approximately registered MRI volume, the algorithm computes the probability of how likely it is that each voxel lies on a major sulcus curve. Dynamic programming is then applied to extract the curve based on the probability map and a shape prior. Because the algorithm can be applied to MRI volumes directly, there is no need to perform preprocessing such as tissue segmentation or mapping to a canonical space. The learning aspect makes the approach flexible and it also works on extracted cortical surfaces.

1 Introduction

Cortical sulci are important structures of the brain. Reliably extracting major cortical sulci from MR images helps us to better understand the functionalities of the brain [7], facilitates many neuro studies for comparing different subjects[12], and assists other brain mapping tasks such as registration [5]. Fig. 1 shows several major cortical sulci: the Central Sulcus, the Postcentral Sulcus, the Superior Temporal Sulcus, the Intraparietal Sulcus, the Middle Frontal Sulcus and the Intraparietal Sulcus.

Cortical sulci lie on the valleys of cortical folds, and can be characterized by mathematical measures such as the mean curvatures [6, 1]. However, major cortical sulci have complicated geometric and photometric patterns and one often needs to use many protocols and high-level knowledge [9] to guide the manually labelling process. In general, automatic extraction of cortical sulci is

a challenging task [12] due to their large intra-class variation and inter-class similarity across different subjects.

Most existing approaches for automatic cortical sulci detection work on the extracted cortical surfaces, which require a pre-processing stage to segment the tissue. Tao et al. [14] used global and local shape priors of sulcus curves on a canonical unit sphere to guide the extraction of the curves, but the method involved mapping the cortical surface to the unit sphere. Khaneja [6] used a dynamic programming approach to find the curves by minimizing an energy function, but this algorithm was not fully automatic since it required the starting and ending points of the sulci to be specified by hand. In [11], the patterns of different local folds are learned and they form a random graph using neural networks. The work by Rettmann [10] extracted sulcal regions using a watershed transformation method applied to cortical surfaces.
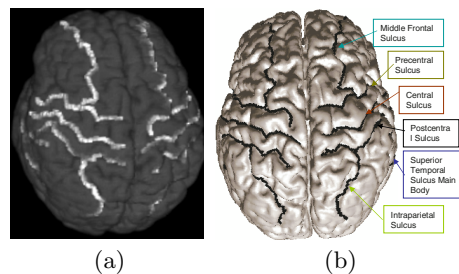

(a)   (b)

Fig. 1. Examples of cortical sulci: (a) shows an MRI volume overlayed with several major cortical sulci, such as the Central Sulcus and the Postcentral Sulcus; (b) illustrates a corresponding extracted cortical surface with the same set of manually labelled sulcus curves

Our method is different from the above methods and it can be directly applied to either MRI volumes or extracted cortical surfaces. Instead of using a number of pre-defined rules/features, we learn/compute the likelihood of each voxel being on a sulcus curve based on a sub-volume ($15 \times 15 \times 15$) centered at this voxel. The *probabilistic boosting tree* (PBT) algorithm [16] is employed to select and combine hundreds of features from a large set of candidates to make an overall decision using a hierarchical structure. The candidate pool consists of around 5,000 features at three scales including intensity, gradients, curvatures, shape indices, locations, and 3D Haar filters etc.. A data-dependent prior is used to put geometric constraints on the curve. Similar to [6], we use a dynamic programming approach to combine the likelihood and shape prior. The algorithm is fully automatic, very general, and has almost no parameter to tune for different major sulcus curves. Moreover, it can be applied in other curve/object detection tasks in medical imaging.

We have a dataset of 40 volumes with several major cortical sulci manually labelled. We split them randomly into 15 training volumes and 25 testing volumes. Training is done both on the MRI volume and extracted cortical surfaces, and we compare the results on the testing images.

2   Problem Formulation

In this section, we give the problem formulation for 3D curve detection. For the rest of this paper, we use $\mathbf{V}$ to represent a 3D MRI volume. When a surface has been extracted, the algorithm works almost the same except for some minor modifications in computing gradients and curvature features.

For an input volume, $\mathbf{V}$, the task of cortical sulcus curve detection is to extract the 3D curve C:

$$C = \{\mathbf{r}_i, i = 0, \cdots, L\}$$

where L is the length of the curve, and $\mathbf{r}_i$ is the coordinates of the ith point on the curve.

We define variable B as the background, i.e. the part of the volume not on the curve C. Clearly, $C \cap B = \emptyset$ and $\Lambda = C \cup B$ is the domain of the whole volume. We define a label y for each voxel $\mathbf{r}$, so that $y = +1$ if the voxel is on the curve C and $y = -1$ if it is on the background B. We then define a discriminative model $\hat{p}(C|\mathbf{V})$ to be of form:

$$-\log \hat{p}(C|\mathbf{V}) = E_1(C) + E_2(C) + E_3$$

where E_3 is a constant which does not depend on C (corresponding to the normalization constant for the distribution $\hat{p}(C|\mathbf{V})$). From the Bayesian point of view, the optimal curve C^* is the one maximizes $\hat{p}(C|\mathbf{V})$.

The term E_1 is given by:

$$E_1(C) = -\sum_{\mathbf{r} \in B} \log p(\mathbf{V}(\mathbf{r}), y = -1|\mathbf{V}(N(\mathbf{r})/\mathbf{r})) - \sum_{\mathbf{r} \in C} \log p(\mathbf{V}(\mathbf{r}), y = +1|\mathbf{V}(N(\mathbf{r})/\mathbf{r}))$$

$$= -\sum_{\mathbf{r} \in \Lambda} \log p(\mathbf{V}(\mathbf{r}), y = -1|\mathbf{V}(N(\mathbf{r})/\mathbf{r})) - \sum_{\mathbf{r} \in C} \log \frac{p(y = +1|\mathbf{V}(N(\mathbf{r})))}{p(y = -1|\mathbf{V}(N(\mathbf{r})))}.$$

where $N(\mathbf{r})$ is the sub-volume centered at voxel $\mathbf{r}$; $\mathbf{V}(\cdot)$ is the intensity value(s) at the given voxel(s); $N(\mathbf{r})/\mathbf{r}$ includes all the voxels in the sub-volume except $\mathbf{r}$; $p(\mathbf{V}(\mathbf{r}), y|\mathbf{V}(N(\mathbf{r})/\mathbf{r}))$ is a conditional joint probability like pseudo-likelihood [2] in spirit. The first term in the above equation does not depend on C, therefore can be ignored. The probabilities $p(y = +1|\mathbf{V}(N(\mathbf{r})))$ and $p(y = -1|\mathbf{V}(N(\mathbf{r})))$ will be learned by PBT from manually labelled data.

The term E_2 is defined as:

$$E_2(C) = -\alpha L + \beta \sum_{i=0}^{L-1} \|\nabla \mathbf{V}(\mathbf{r}_{i+1}) - \nabla \mathbf{V}(\mathbf{r}_i)\|$$

where α and β are positive parameters to balance the importance of the corresponding terms. The motivation of introducing the first term is the observation that the sulcus curves are not very smooth, while the first term favors long curves, therefore, it discourages a curve being too straight. The second term of E_2 accumulates data dependent cues, i.e., we prefer that the intensity along the detected curve does not change too much.

In summary, maximizing the probability $\hat{p}(C|\mathbf{V})$ is equivalent to minimizing the energy function:

$$E(C) = -\sum_{\mathbf{r} \in C} \log \frac{p(y = +1|\mathbf{V}(N(\mathbf{r})))}{p(y = -1|\mathbf{V}(N(\mathbf{r})))} + E_2(C) \tag{1}$$

Here, the models capturing the appearances of foreground and background are combined in the discriminative probability ratio. Note that $p(y = +1|\mathbf{V}(N(\mathbf{r})))$ is the posterior probability of a voxel $\mathbf{r}$ belonging to the foreground (sulcus curve) given the sub-volume centered at $\mathbf{r}$. The second column in Fig. 3 shows such probability maps. The optimal curve C^* is the one that minimizes the above energy function $E(C)$.

We use dynamic programming (DP) to minimize $E(C)$ given by equation (1). DP is guaranteed to find the global minimum of $E(C)$, but requires starting and ending points (we refer to both as end points). We propose an adaptive method for selecting the end points, instead of choosing the end points by hand [6]. Our method uses the training data to measure the mean and covariance of the positions of the end points and then constrains the end points to lie within boxes centered on the means and with sides equal to twice the variance. We further localize the end points by requiring that $p(y = +1|\mathbf{V}(N(\mathbf{r})) > T$ where T is a learned threshold. Of these remaining points, we select the one that has the smallest value of $p(y = +1|\mathbf{V}(N(\mathbf{r}))$.

3   The Discriminative Models

Now the task is to learn and compute the discriminative probability $p(y|\mathbf{V}(N(\mathbf{r})))$ for each voxel $\mathbf{r}$. We adopt a new learning framework, probabilistic boosting tree [16], to learn complex discriminative models based on boosting algorithms [4].

For learning, we design a pool of around 5,000 features at three scales, including intensity, gradients, curvatures, shape indices, locations, and 3D Haar filters etc.. We used integral volume to compute the response of 3D Haar
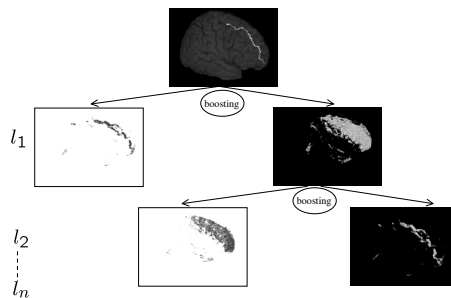

Fig. 2. Illustration of a boosting tree on a training volume: The left branch shows the probability of each voxel is not on the sulcus curve and the right branch shows the probability that each voxel is on the sulcus curve

filters. At each location (x_1, y_1, z_1), the integral volume is computed as $\int_0^{x_1} \int_0^{y_1} \int_0^{z_1} V(x, y, z) dx dy dz$. Thus the computational cost for computing the response of 3D Haar filters is reduced significantly since every time we only need to sum up the values at the corners of the 3D Haar filter in the integral volume.

The tree is trained recursively: At the current node, the empirical distribution $\hat{q}(y)$ of the data is calculated, if the node is not pure (i.e., $\hat{q}(y)$ is not close to 0 or 1), a strong classifier is trained on the data. Each sample is then passed to the left and right subtree, weighted by $q(-1|\mathbf{V}(N(\mathbf{r})))$ and $q(+1|\mathbf{V}(N(\mathbf{r})))$, respectively, where $q(+1|\mathbf{V}(N(\mathbf{r})))$ is the probability that $\mathbf{V}(N(\mathbf{r}))$ is a positive sample, according to the strong classifier. Thus, the strong classifier at each node is not used to return the class of the sample but rather to assign the sample to the left or right subtree. Training proceeds recursively(see [16] for information about the basic algorithm). Fig. 2 shows the first two levels of a tree. PBT does training and testing in a divide-and-conquer manner and outputs the overall discriminative probability as:

$$\tilde{p}(y|\mathbf{V}(N(\mathbf{r}))) = \sum_{l_1} \tilde{p}(y|l_1, \mathbf{V}(N(\mathbf{r})))q(l_1|\mathbf{V}(N(\mathbf{r})))$$

$$= \sum_{l_1, l_2} \tilde{p}(y|l_2, l_1, \mathbf{V}(N(\mathbf{r})))q(l_2|l_1, \mathbf{V}(N(\mathbf{r})))q(l_1|\mathbf{V}(N(\mathbf{r})))$$

$$= \sum_{l_1, \cdots, l_n} \tilde{p}(y|l_n, \cdots, l_1, \mathbf{V}(N(\mathbf{r}))), \cdots, q(l_2|l_1, \mathbf{V}(N(\mathbf{r})))q(l_1|\mathbf{V}(N(\mathbf{r})))$$

where $l_i \in \{+1, -1\}$ is augmented hidden variable as shown in Fig. 2, indicating which branch is for this node: $l_i = -1$ and $l_i = +1$ point to the left and right branch, respectively. $q(l_i|l_{i-1}, \cdots, l_1, \mathbf{V}(N(\mathbf{r})))$ is the discriminative probability computed according to the boosting strong classifier learned at each node, and $\hat{q}(y|l_n, \cdots, l_1, \mathbf{V}(N(\mathbf{r})))$ is the empirical distribution at the leaf node.

4   Experiments

We have tested the proposed approach for extracting the Central and the Middle Frontal Sulci. In both cases, we used ground truth estimates of the Sulci location to train and test the algorithm. We also compared the performance for extracting the Central Sulcus when the cortical surface is available. The MR images, cortical surfaces and manually labelled landmark data are exactly the same as those in [13].

Learning the PBT took approximately eight hours (it is a function of the size of the training dataset); computing the posterior probability by PBT took about one minute per image; running the dynamic programming took around twenty seconds per image. The computer used has 2.4 GHz CPU and 1.0GB memory. Standard code optimization techniques can reduce these times significantly.

We have a dataset of 40 volumes of which we use 15 for training and the remained 25 for testing. In these volumes, the length of the sulci varies from 60-70 voxels. The PBT is learned on the 15 training volumes and the majority of features it selected to use were 3D Haar filters, though some curvature features

were also selected. Fig. 3 shows the results on some of the testing volumes for detecting the Central Sulcus and the Middle Frontal Sulcus on MRI volumes. The first column shows the input with manual labels superimposed and the second column shows the posterior probability of the sulci output by the PBT. Note that the probability maps have large responses around the correct position, but the maps are blurred, sometimes disconnected, and therefore are not sufficient to localize the sulcus directly. We then applied dynamic programming to minimize
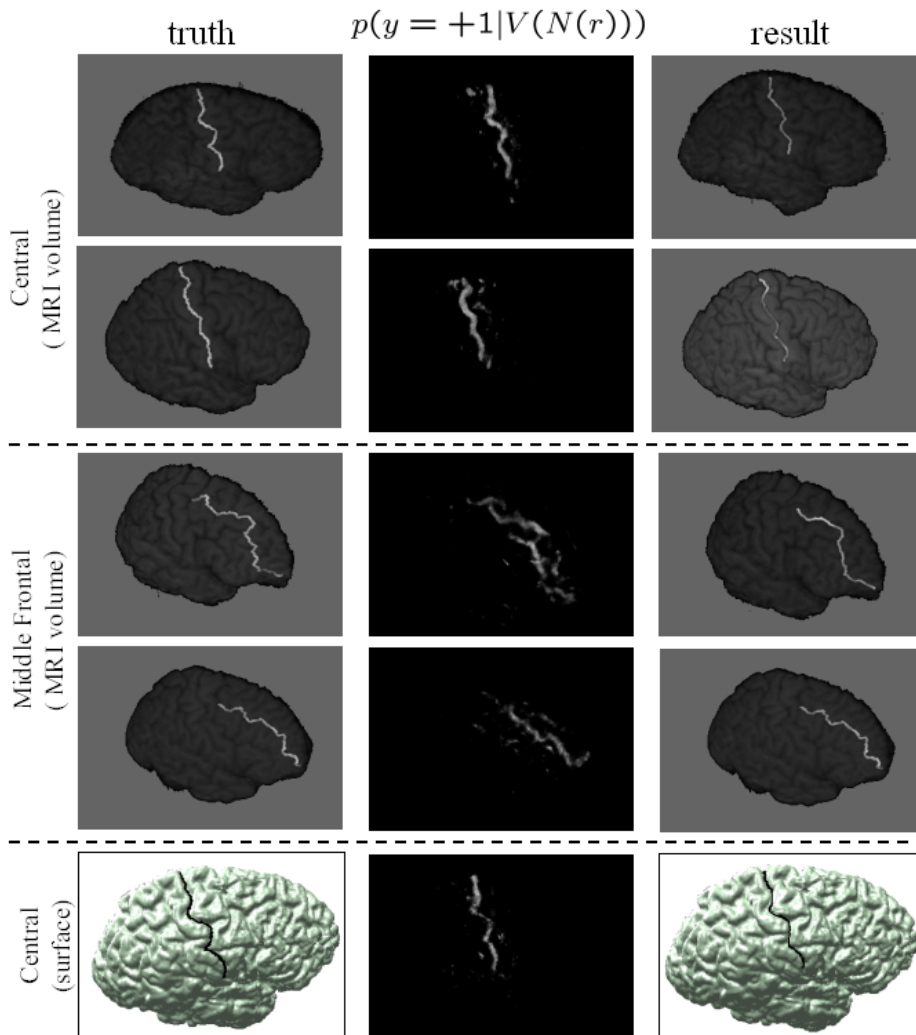

Fig. 3. Detection results on some of the testing images for the Central Sulcus on MRI, the Middle Frontal Sulcus on MRI, and the Central Sulcus on the cortical surfaces. The ground truth (left), the PBT posterior $p(y = +1|\mathbf{V}(N(\mathbf{r})))$ (middle), and the result of DP superimposed on the image (right).

the full energy function, equation (1). This resulted in a very clear estimate of the sulcus location, which are compared to the ground truth in Fig. 3.

Our method can also be applied to cases where only the cortical surface is available. This requires removing certain features from those considered in the case of volume data (i.e. we keep the 3D Haar filters, but remove intensity). We repeated the experiment on the Central Sulcus with the same training and testing dataset as before, and the result is also shown in Fig. 3.

To quantitatively evaluate the performance of our approach, we measured the distances between the estimated curves and the ground truth. We used the following distance measures:

$$H_{wor}(C,G) = \max_{c \in C} \min_{g \in G} |c - g|, \quad H_{av}(C,G) = \frac{1}{|C|} \sum_{c \in C} \min_{g \in G} |c - g|. \quad (2)$$

Here C represents the curve found by DP and G is the ground truth. $H_{av}(C,G)$ gives the average distances from curve C to their closest points on curve G. By contrast, $H_{wor}(C,G)$ measures the worst case fit from curve C to curve G. For symmetry, we also consider $H_{av}(G,C)$.

Our evaluation results, as shown in table 1, show very good performance. In the table, $< \cdot >$ denotes the average over the dataset, for example, $< H_{av}(C,G) >= (1/N) \sum_{i=1}^{N} H_{av}(C_i,G_i)$, where N is the number of examples in the dataset. All the values of H_{av} are in the range of 3-5 voxels for both training and testing datasets. Observe that although the worst case measures are relatively big, the average distance is small, which suggests that some points occasionally have big offset while the overall curves still can be detected fairly accurately. The biggest errors occurred at the starting and ending points of the sulci. The testing errors are only slightly bigger than the training errors, which implies that our algorithm generalizes well.

Table 1. Error measures on 15 training and 25 testing images for the extracting of the Central Sulcus on volume, the Middle Frontal Sulcus on volume, and the Central Sulcus on surfaces. See text for notations. The unit of distance is the voxel.

Dataset	$< H_{av}(C,G) >$	$< H_{av}(G,C) >$	$< H_{wor}(C,G) >$
Testing (Central on volume)	2.74	3.46	7.53
Central (Central on volume)	3.76	4.22	8.56
Testing (Middle Frontal on volume)	4.26	4.59	12.04
Training (Middle Frontal on volume)	4.09	4.46	8.99
Testing (Central on surface)	2.79	3.07	9.48
Training (Central on surface)	2.44	2.88	8.44

5   Conclusions and Future Work

We presented a method for extracting cortical sulci, which uses a discriminative model including a discriminative term and a data dependent prior. The discriminative term was learned in a supervised way by the probabilistic boosting

tree algorithm, and dynamic programming was used to find the optimal solution by minimizing the discriminative energy function. A procedure was given for adaptively selecting the end points for dynamic programming.

The method was applied to extracting the Central Sulcus and the Middle Frontal Sulcus from raw MRI volume data. In both cases, the performance was very good when evaluated on both the training and testing data. The method was also applied to extracting the Central Sulcus when the cortical surface has already been extracted and the results were similar to those obtained directly from raw volume data. The biggest errors occurred at the end points of the sulci and might be due to ambiguity in the precise positions of these points (i.e. errors arising in the ground truth), or the procedure for estimating the starting and ending points. This is a topic for future research.

In this work, we designed a feature pool consists of about 5,000 candidates, most of which are 3D Haar filters. If effective features are used, we expect that the algorithm will achieve similar performance with fewer features. Our next step is to explore more sophisticated features for this particular medical image task.

Acknowledgement

References

1. A. Bartesaghi and G. Sapiro, "A System for the Generation of Curves on 3D Brain Images", *Human Brain Map.* 14:1-15 2001.
2. J. Besag, "Efficiency of pseudo-likelihood estimation for simple Gaussian fields", *Biometrika* 64: 616-618, 1977.
3. A. Caunce and C.J. Taylor, "Building 3D Sulcal Models Using Local geometry", *Medical Image Analysis*, 5 69-80, 2001.
4. Y. Freund and R. Schapire, "A Decision-Theoretic Generalization of On-line Learning and An Application to Boosting", *J. of Computer and Sys. Sci.*, 1997.
5. P. Hellier and C. Barillot, "Coupling Dense and Landmark-Based Approaches for Nonrigid Registration", *IEEE Trans. Med. Imaging*, vol. 22, no. 2, Feb. 2003
6. N. Khaneja, M.I. Miller, and U. Grenander, "Dynamic Programming Generation of Curves on Brain Surfaces", *PAMI*, vo.20, no. 11, 1998.
7. G. LeGoualher, E. Procyk, D.L. Collins, R. Venugopal, C. Barillot, A.C. Evans, "Automated extraction and variability analysis of sulcal neuroanatomy", *IEEE Trans. Med. Imaging* vo.18, no. 3, March 1999.
8. G. Lohmann, "Extracting Line Representations of Sulcal and Gyral Patterns in MR Images of the Hunam Brain", *IEEE Trans. on Medical Imag.*, vol.17, no.6, Dec. 1998.
9. M. Ono, S. Kubik, and S.D. Abernathey, "Atlas of the Cerebral Sulci", *New York:Thieme Medical*, 1990.

10. M.E. Rettmann, X. Han, C. Xu, J.L. Prince, "Automated sulcal segmentation using watersheds on the cortical surface", *Neuroimage*, 15(2):329-44, Feb. 2002.

11. D. Riviere, J.F. Mangin, D. Papadopoulos-Orfanos, J.M. Martinez, V. Frouin, and J. Regis, "Automatic Recognition of Cortical Sulci of the Human Brain Using A Congregation of Neural Networks", *Medical Image Analysis*, 77-92, 2002.

12. P.M. Thompson et al. "Mapping Cortical Change in Alzheimer's Disease, Brain Development, and Schizophrenia", *NeuroImage*, 23 Suppl 1:S2-18, September 2004.

13. P.M. Thompson, A.D. Lee, R.A. Dutton, J.A. Geaga,K.M. Hayashi KM, U. Bellugi, A.M. Galaburda, J.R. Korenberg, D.L. Mills, A.W. Toga, A.L. Reiss, "Abnormal Cortical Complexity and Thickness Profiles Mapped in Williams Syndrome", *J. of Neuroscience*, 25(18):, April 2005.

14. X. Tao, J.L. Prince, and C. Davatzikos, "Using a Statistical Shape Model to Extract Sulcal Curves on the Outer Cortex of the Human Brain", *IEEE Tran. on Medical Imag.*, vol.21, no. 5, May 2002.

15. J.P. Thirion and A. Gourdon, "Computing the Differential Characteristics of Isointensity Surfaces", *Computer Vision and Image Understanding*, vol. 61, no. 2, 1995.

16. Z. Tu, "Probabilistic Boosting-Tree: Learning Discriminative Models for Classification, Recognition, and Clustering", *Proc. of ICCV*, 2005.