Segmentation of 2D and 3D Images through a Hierarchical Clustering Based on Region Modelling

Xinquan Shen[†,*] Michael Spann[†] and Peter Nacken[‡]
† School of Electronic and Electrical Engineering
The University of Birmingham, Birmingham B15 2TT, UK
‡ Shell KSEPL, Rijswijk, The Netherlands

Abstract

This paper presents an unsupervised segmentation method applicable to both 2D and 3D images. The segmentation is achieved by a bottom-up hierarchical analysis to progressively agglomerate pixels/voxels in the image into non-overlapped homogeneous regions characterised by a linear signal model. A hierarchy of adjacency graphs is used to describe agglomeration results from the hierarchical analysis, and is constructed by successively performing a clustering operation which produces an optimal classification by merging each region with its nearest neighbours. The nearest neighbour of a region is determined by a merge condition derived under the framework of a statistical inference and a dissimilarity function based on the error produced by fitting the region model to pixels/voxels in two adjacent regions. The top level of the hierarchy then describes the segmentation result.

Image Processing **Segmentation** **Hierarchical Analysis**
Linear Signal Models **Nearest Neighbour**

1 Introduction

An important task in computer vision is image segmentation which aims to partition the image into physically meaningful regions. In this paper, we present an unsupervised segmentation method applicable to both 2D and 3D images.

The segmentation of 3D images is often required by applications such as seismic and medical data analysis. Limited work has been reported on this area. Most of them achieve the segmentation through fitting a 3D deformable model to the object in the image[1, 2, 3, 4]. In general, the type of

* Author to whom correspondence should be addressed. E-mail: shenx@eee.bham.ac.uk; Fax: 44 121 414 4291

the object shape and/or the location of the object in space are assumed to be known. To achieve an unsupervised segmentation is thus difficult. On the other hand, there is a huge body of work on the segmentation of 2D images[5, 6]. Hierarchical approaches are often used among such segmentation methods. The approach centres on the use of a hierarchical data structure representing the image at a number of spatial resolutions. Each level in the hierarchy corresponds to a condensed version of the image at the previous level. The condensation can be achieved by reducing the resolution of the image, leading to a regular hierarchical structure (pyramid)[7], or by clustering primitives (non-squared regions) in the image, leading to an irregular hierarchical structure[8, 9, 10]. Compared to the regular hierarchical structure, the irregular structure is more flexible in describing image content and often leads to a better result.

Segmentation using a hierarchical approach is normally achieved in two ways: top-down analysis and bottom-up analysis. Top-down analysis segments the image by successively refining the result from the previous coarse level[11, 12, 13, 14]. Basically, a signal model is assumed for the image regions. Processing at each level aims to obtain an optimal classification by computing a maximum a posteriori probability (MAP) which is often implemented by an iterative procedure. In general, a priori information about the number of regions in the image is needed, although an unsupervised segmentation could be achieved by estimating this information beforehand or during the computation of the MAP segmentation[12, 13]. The method is computationally expensive, especially when the number of regions in the image is large.

In bottom-up methods, the segmentation is achieved by successively (from bottom to top) clustering regions at each level until no further change can be produced from this operation or some conditions are satisfied. Clustering at each level involves merging several small regions into a larger region, and thus produces a condensed representation of the image at the next higher level. In general, the segmentation is unsupervised. Burt *et al.*[15] developed a linked pyramid structure in which each cell represents a region in the image. Segmentation is implemented by iteratively updating parent-child links based on the region property assigned to each cell until no change occurs from one iteration to the next. The method is applied to extracting object from background. Montanvert *et al.*[9] proposed a hierarchical representation based on irregular tessellations of the image. Each irregular tessellation is described by an adjacency graph in which each node corresponds to a region in the image and two nodes are linked by an edge if their corresponding regions are spatially adjacent. The structure is thus adapted to the image content while all other properties of a regular pyramid are preserved. The hierarchy is constructed bottom-up by applying a stochastic decimation at each level. The method is essentially for finding connected components of labelled images. Segmentation of gray level images requires an empirical threshold and the result is not unique due to the stochastic decimation. Based on a similar hierarchical structure, Nacken[10] achieves the segmentation by also applying a relinking operation[15, 16, 17, 10] after clustering at each level. The relinking procedure iteratively updates the assignment of pixels on the boundary of regions until an optimal classification is reached. Again, an empirical threshold is required.

In this paper, we achieve an unsupervised segmentation for both 2D and 3D images through a bottom-up hierarchical analysis. A hierarchy of adjacency graphs is used to describe irregular partitions of the image. The hierarchy is constructed bottom-up, by successively applying a clustering operation to each level of the hierarchy to agglomerate regions in the current graph into regions in the graph at the next higher level, until no further agglomeration can be made. The top level of the hierarchy then describes the result of the segmentation while the bottom level corresponds to the original input image. Clustering at each level of the hierarchy is performed based on the criterion in which two adjacent regions are classified into one cluster if and only if they are mutually nearest neighbours. Homogeneous regions in the image are characterised by a linear model which is expressed as a linear polynomial of the image coordinates corrupted with an additive noise described by a Gaussian distribution. The nearest neighbour of a given region here refers to a region which is spatially adjacent and most similar to the given region. In this work, this region is determined by a merge condition and a dissimilarity function defined on two adjacent regions. By modelling the region graylevel variation with the linear signal model, the merge condition is derived under the framework of a statistical inference, and the dissimilarity function is determined based on the fitting error produced by fitting the region model to pixels/voxels in two adjacent regions. The classification at each level is thus optimal with respect to the implied definition of the nearest neighbour and no relinking operations are needed.

The paper is organised as follows. Section 2 presents a linear signal model for representing homogeneous regions. Section 3 describes the hierarchical structure and clustering method for achieving the segmentation. Implementation details and experimental results are given in Section 4 and 5 respectively. Some further discussion on the proposed segmentation method is given in Section 6. Finally, Section 7 is for conclusions.

2 Linear Signal Model

The linear signal model has been widely used in image analysis[18, 19, 20, 21, 22]. The idea is to assume that the image luminance is linearly dependent on spatial position. The gray level of the image is then expressed as a linear combination of image coordinates, parameterised by a set of coefficients (called model parameters) which can be estimated from a set of measurements (known position and gray level pairs). By fitting the model to image data, a strategy can be devised for applications such as edge detection[18], classification[23, 20], and image coding[21]. In the following sections, all the discussion is based on 3D images. Results applied to 2D images can be directly obtained as a special case.

Assume that a homogeneous region in a 3D gray level image can be described by the 4th degree linear polynomial corrupted by additive noise. The gray level $g(x, y, z)$ at the voxel (x, y, z) in the image then obeys the following linear model:

$$g(x, y, z) = ax + by + cz + d + \eta(x, y, z) \tag{1}$$

where, a, b, c, and d are parameters of the model. $\eta(x, y, z)$ is additive random noise. We assume $\eta(x, y, z)$ can be described by a Gaussian distribution with mean 0 and variance σ^2. The region described by this model has a sloped gray level surface and parameters a, b and c determine the slope of the surface in each coordinate direction. For the convenience of the discussion, we use homogeneous coordinates $(x, y, z, 1)$ to represent the image point (x, y, z), and write (1) in matrix notation:

$$g(x, y, z) = \boldsymbol{x}^T \boldsymbol{\xi} + \eta(x, y, z) \tag{2}$$

where, $\boldsymbol{x} = [\, x\, y\, z\, 1\,]^T$ is a vector of image coordinates, $\boldsymbol{\xi} = [\, a\, b\, c\, d\,]^T$ is a vector of model parameters, and T denotes transpose.

Given a set of voxels in the region, the model parameters can be estimated in various ways[24, 25]. The linear least-squares method is most frequently used. It gives a closed form solution for the estimation, and supports a hierarchical analysis of images (see Section 3.1).

Let $(\boldsymbol{x}_i; g_i), i = 1, 2, \ldots, N$, be N ($N \geq 4$) voxels in the region (g_i is the gray level at the position $\boldsymbol{x}_i$), and $\hat{\boldsymbol{\xi}} = [\, \hat{a}\, \hat{b}\, \hat{c}\, \hat{d}\,]^T$ be the estimation of parameter vector $\boldsymbol{\xi}$. The least-squares method determines a $\hat{\boldsymbol{\xi}}$ which minimises the following sum of squared residuals:

$$r^2 = \sum_{i=1}^{N} \left[g_i - \boldsymbol{x}_i^T \hat{\boldsymbol{\xi}} \right]^2 \tag{3}$$

By vanishing the derivative $\dfrac{\partial r^2}{\partial \hat{\boldsymbol{\xi}}}$, $\hat{\boldsymbol{\xi}}$ can be solved from the following matrix equation[24],

$$\boldsymbol{A} \hat{\boldsymbol{\xi}} = \boldsymbol{B} \tag{4}$$

where, $\boldsymbol{A}$ and $\boldsymbol{B}$ have the following forms:

$$\boldsymbol{A} = \begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i z_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i z_i & \sum y_i \\ \sum x_i z_i & \sum y_i z_i & \sum z_i^2 & \sum z_i \\ \sum x_i & \sum y_i & \sum z_i & N \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} \sum x_i g_i \\ \sum y_i g_i \\ \sum z_i g_i \\ \sum g_i \end{bmatrix}$$

When $\boldsymbol{A}$ (called design matrix) is full ranked, a unique solution of $\hat{\boldsymbol{\xi}}$ can be obtained (see Section 4 for the treatment of the case when $\boldsymbol{A}$ is not full ranked). Incorporating (4), (3) can be written as

$$r^2 = S - \hat{\boldsymbol{\xi}}^T \boldsymbol{B} \tag{5}$$

where, $S = \sum_{i=1}^{N} g_i^2$.

$\hat{\boldsymbol{\xi}}$ has the following properties:

$$\mathrm{E}(\hat{\boldsymbol{\xi}}) = \boldsymbol{\xi}, \quad \mathrm{Cov}(\hat{\boldsymbol{\xi}}) = \sigma^2 \boldsymbol{A}^{-1} \tag{6}$$

where $\mathrm{E}(\cdot)$ and $\mathrm{Cov}(\cdot)$ denote the computation of expectation value and covariance respectively. It can be shown that $\hat{\boldsymbol{\xi}}$ is the unbiased estimate of minimum variance of $\boldsymbol{\xi}$, and $\hat{\boldsymbol{\xi}}$ is independent of the sum of squared residuals r^2 [24].

3 Hierarchical Clustering

3.1 Hierarchy of adjacency graphs

In this work, we use an adjacency graph of regions to describe the partition of an image. An adjacency graph of regions is a non-directed graph[26] G in which each node corresponds to a region and two nodes are linked by an edge if and only if the corresponding two regions are spatially adjacent. A hierarchy of adjacency graphs is a stack of graphs, $G_0, G_1, \ldots, G_L$, each of which is an adjacency graph of regions (see Fig. 1). G_0, called the base level, corresponds to the original input image, while G_L describes the final result of the segmentation. Graph G_l, $l > 0$, is generated by applying a clustering operation to the graph G_{l-1}. Each node in G_l is a parent of nodes in a connected sub-graph of G_{l-1} and corresponds to a region generated by merging its child regions. The hierarchy of adjacency graphs is thus constructed by successively applying a clustering operation to the adjacency graph at the top level of the current hierarchy. In the following discussion, we do not distinguish the node and its corresponding region when the meaning in the context is clear.

By modelling regions with the linear model (1), each region can be uniquely described by $\hat{\boldsymbol{\xi}}$ and r^2 which, in the least-squares sense, can be computed from $\boldsymbol{A}$, $\boldsymbol{B}$ and S (see Section 2). $\boldsymbol{A}$, $\boldsymbol{B}$ and S can be computed hierarchically if they are maintained at each node in the graph. Let nodes $v_i^{(l)}(\boldsymbol{A}_i^{(l)}, \boldsymbol{B}_i^{(l)}, S_i^{(l)})$, $i = 1, 2, \ldots, K$, in the graph G_l be K children of the node $v^{(l+1)}(\boldsymbol{A}^{(l+1)},$

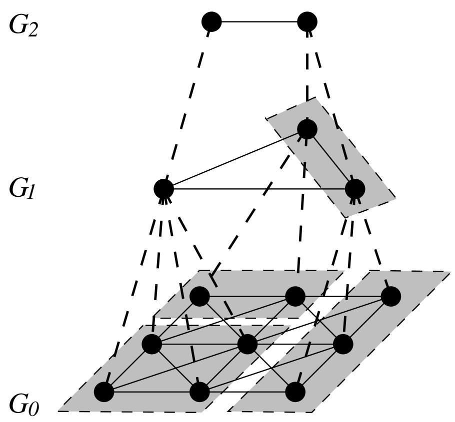


Figure 1: A hierarchy of adjacency graphs

$\boldsymbol{B}^{(l+1)}$, $S^{(l+1)}$) in G_{l+1}. We have

$$\begin{aligned}\boldsymbol{A}^{(l+1)} &= \sum_{i=1}^{K} \boldsymbol{A}_i^{(l)} \\ \boldsymbol{B}^{(l+1)} &= \sum_{i=1}^{K} \boldsymbol{B}_i^{(l)} \\ S^{(l+1)} &= \sum_{i=1}^{K} S_i^{(l)}\end{aligned} \tag{7}$$

Thus, $\hat{\boldsymbol{\xi}}$ and r^2 can be computed hierarchically.

3.2 Clustering an adjacency graph

The clustering algorithm applied to an adjacency graph needs to partition nodes in the graph into a number of non-intersected sets (clusters). Nodes in each cluster are within a connected sub-graph and are merged to generate a node in the graph at the next higher level of the hierarchy. Montanvert *et al.*(9) proposed a clustering method based on stochastic decimation of a graph. Each cluster consists of a central node and a maximal subset of its neighbourhood both of which are determined by an iterative stochastic process. Nacken(10) modified the method by selecting the maximal subset based on a dissimilarity measurement. With this method, clustering at each level can be achieved by an iterative procedure. However, nodes on the boundary of a cluster may not be optimally classified with respect to the implied clustering criterion. An iterative relinking procedure is often applied to obtain an optimal parent-children configuration. When the adjacency graph has a large number of nodes, computations at each relinking iteration is expensive.

In this work, we present a clustering method based on the nearest neighbour criterion(27). With respect to this criterion, classification of each level is optimal, and no relinking operations are needed.

The method only requires a single visit to each node and its neighbours.

3.2.1 Nearest neighbour

Let $v_i, i = 1, 2, \ldots, p$, be p nodes spatially adjacent to a node v_0. Node $v_k, 1 \leq k \leq p$, is the nearest neighbour of v_0 if

$$\varphi(v_0, v_k) = \min_{1 \leq j \leq p} \{\varphi(v_0, v_j) | C(v_0, v_j) = \text{True}\} \tag{8}$$

where, C is a predicate function and φ is a dissimilarity function.

Predicate C controls a condition for merging two nodes. When $C(v_0, v_k)$ is true, v_k becomes a candidate node to be merged with v_0. The nearest neighbour is thus a candidate node which has the smallest dissimilarity with v_0. The dissimilarity function φ can be of any form, as long as its value is proportional to the dissimilarity between two nodes, i.e. the more dissimilar the two nodes, the larger the value of the function. It should be noted that the 'nearest' here is measured on the similarity of two spatially adjacent nodes instead of the spatial distance between two nodes as is the usual meaning of the word. Under the above definition, a given node may have more than one nearest neighbour. Also, the fact that v_k is the nearest neighbour of v_0 does not necessarily imply that v_0 is the nearest neighbour of v_k. In our clustering algorithm (see Section3.2.4), two nodes are merged if and only if they are mutually the nearest neighbours.

3.2.2 Dissimilarity function

Let v_1 and v_2 be two adjacent nodes in an adjacency graph. At the base level, v_1 and v_2 correspond to a single voxel respectively. The difference between gray levels of two voxels is thus a reliable value for measuring the dissimilarity of two voxels. The dissimilarity function is simply defined as follows:

$$\varphi(v_1, v_2) = |g(v_1) - g(v_2)| \tag{9}$$

where, $g(v_i), i = 1, 2$, is the gray level of the voxel v_i.

At non-base levels, v_1 and v_2 correspond to a region, respectively, whose gray level surface is described by the region model (1). We consider $r_{1,2}^2$, the sum of squared residuals produced by fitting the model to the region $v = v_1 \cup v_2$ ("$\cup$" denotes a union operation). Clearly, $r_{1,2}^2$ approaches zero when v_1 and v_2 have the same gray level surface, while the increase in the difference between v_1 and v_2 will lead to a large value of $r_{1,2}^2$. The dissimilarity function at non-base levels is thus defined as follows:

$$\varphi(v_1, v_2) = \frac{r_{1,2}^2}{N_1 + N_2 - 2m} \tag{10}$$

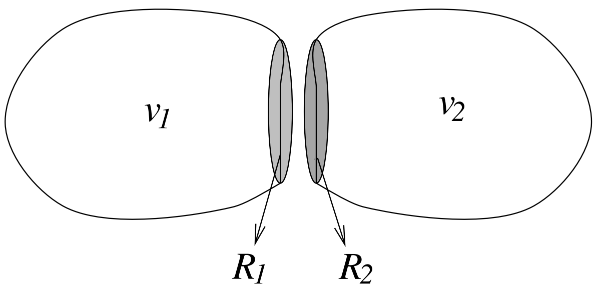


Figure 2: Adjacent border of two 2D regions

where, N_1 and N_2 are the number of voxels in nodes v_1 and v_2 respectively, and m is the number of parameters in the region model (here $m = 4$). Statistically, $\varphi(v_1, v_2)$ is an unbiased estimation of the noisy variance σ^2 on the region v.

3.2.3 Predicate function

A. At the base level

At the base level of the hierarchy, voxels are clustered based on the difference between gray levels of two voxels. For any two adjacent nodes v_1 and v_2, function C is simply defined as follows:

$$C(v_1, v_2) = \begin{cases} \text{True} & \text{if } |g(v_1) - g(v_2)| < \delta \\ \text{False} & \text{otherwise} \end{cases} \tag{11}$$

where, $\delta > 0$ is a threshold which defines a maximum value allowing for the gray level difference between two voxels to be merged. In our implementation, a fixed value of δ is used for all experiments (see Section 5).

B. At non-base levels

Let v_1 and v_2 be two adjacent nodes with estimated parameters $\hat{\boldsymbol{\xi}}_1 = [\,\hat{a}_1\,\hat{b}_1\,\hat{c}_1\,\hat{d}_1\,]^T$ and $\hat{\boldsymbol{\xi}}_2 = [\,\hat{a}_2\,\hat{b}_2\,\hat{c}_2\,\hat{d}_2\,]^T$ and the sum of squared residuals r_1^2 and r_2^2, respectively. Intuitively, two nodes should be merged if their corresponding gray level surfaces are part of the same surface. Notice that using $\hat{\boldsymbol{\xi}}_1 = \hat{\boldsymbol{\xi}}_2$ as a necessary condition for merging v_1 and v_2 is inappropriate, since a small deflection in slopes may lead to large change in intercepts ($\hat{d}_1$ and $\hat{d}_2$). Instead, we merge two nodes if they have the same slopes and there is a continuous gray level transition on their adjacent border consisting of those voxels each of which is spatially adjacent to a voxel in the opposite node. A merging condition can be derived under the framework of statistical inference.

Let $R_1(R_2)$ be the set containing all the voxels in $v_1(v_2)$ on the adjacent border of two nodes v_1 and v_2 (see Fig. 2 for an example in the 2D case.). A continuous gray level transition on the adjacent border of v_1 and v_2 means that voxels in R_1 and R_2 are on both of the surfaces determined by nodes v_1 and v_2. For each voxel in R_1, we can apply a statistical test to examine whether that voxel is also on the surface of v_2, and vice versa for voxels in R_2.

Let $(\boldsymbol{x}; g) \in R_2$, and

$$h = g - \boldsymbol{x}^T \hat{\boldsymbol{\xi}}_1 \tag{12}$$

h is then the difference between the gray level of the given voxel and the prediction of the region model (for node v_1) at that voxel. Normally distributed noise implies that h is also normally distributed. Independence of the noise implies that g is independent of gray levels of voxels in v_1. Under the hypothesis that $(\boldsymbol{x}; g)$ is on the surface of v_1, we have

$$E(h) = 0 \tag{13}$$

$$D(h) = \sigma^2 \left(1 + \boldsymbol{x}^T \boldsymbol{A}_1^{-1} \boldsymbol{x}\right) \tag{14}$$

where, $D(\cdot)$ denotes the computation of variance, and $\boldsymbol{A}_1$ is the design matrix computed from v_1.

Since $\hat{\boldsymbol{\xi}}_1$ is independent of r_1^2, h is independent of r_1^2. Therefore, under the above hypothesis, the statistic

$$\lambda_1 = \frac{h}{\sqrt{1 + \boldsymbol{x}^T \boldsymbol{A}_1^{-1} \boldsymbol{x}}} \cdot \frac{\sqrt{N_1 - m}}{r_1} \tag{15}$$

has a t-distribution with $(N_1 - m)$ degrees of freedom, where N_1 is the number of voxels in v_1. We then accept the hypothesis if

$$P(t > |\lambda_1|) \geq \alpha \tag{16}$$

where, α is a given significance level. A significance level $\alpha = 0.01$ means that the probability of taking the voxel $(\boldsymbol{x}; g)$ as not on the surface of v_1 when it is in fact on the surface is less than 1%.

A similar t-test can be derived for each voxel in R_1. For noisy images, when more than half of the voxels in R_1 and R_2 are on both of surfaces determined by v_1 and v_2, we have a strong reason to assume the existence of a continuous gray level transition from R_1 to R_2. Let k_1 (k_2) be the number of voxels in R_1 (R_2) which are also on the surface of v_2 (v_1) according to the above t-test. The gray level transition on the adjacent border of v_1 and v_2 is considered as continuous, if

$$\mu = \frac{k_1 + k_2}{K_1 + K_2} > 0.5 \tag{17}$$

where, K_1 and K_2 are the number of voxels in R_1 and R_2 respectively.

To check whether v_1 and v_2 have the same slopes, we consider $\hat{a}_1 - \hat{a}_2$, $\hat{b}_1 - \hat{b}_2$, and $\hat{c}_1 - \hat{c}_2$. Under the hypothesis that v_1 and v_2 have the same slopes, $\hat{a}_1 - \hat{a}_2$, $\hat{b}_1 - \hat{b}_2$ and $\hat{c}_1 - \hat{c}_2$ each have a normal distribution with mean 0 and variance $\sigma^2(\boldsymbol{A}_1^{-1}(0,0) + \boldsymbol{A}_2^{-1}(0,0))$, $\sigma^2(\boldsymbol{A}_1^{-1}(1,1) +$

$\boldsymbol{A}_2^{-1}(1,1))$, and $\sigma^2(\boldsymbol{A}_1^{-1}(2,2)+\boldsymbol{A}_2^{-1}(2,2))$ respectively*. Thus, statistics:

$$\begin{aligned}\lambda_2 &= \frac{\hat{a}_1-\hat{a}_2}{\sqrt{\boldsymbol{A}_1^{-1}(0,0)+\boldsymbol{A}_2^{-1}(0,0)}}\cdot\frac{\sqrt{N_1+N_2-2m}}{\sqrt{r_1^2+r_2^2}}\\ \lambda_3 &= \frac{\hat{b}_1-\hat{b}_2}{\sqrt{\boldsymbol{A}_1^{-1}(1,1)+\boldsymbol{A}_2^{-1}(1,1)}}\cdot\frac{\sqrt{N_1+N_2-2m}}{\sqrt{r_1^2+r_2^2}}\\ \lambda_4 &= \frac{\hat{c}_1-\hat{c}_2}{\sqrt{\boldsymbol{A}_1^{-1}(2,2)+\boldsymbol{A}_2^{-1}(2,2)}}\cdot\frac{\sqrt{N_1+N_2-2m}}{\sqrt{r_1^2+r_2^2}}\end{aligned} \tag{18}$$

each have a t-distribution with N_1+N_2-2m degrees of freedom†, where N_1 and N_2 are the number of voxels in v_1 and v_2 respectively. Given a significance level α, we consider v_1 and v_2 have the same slopes, if the following condition is satisfied:

$$P(t>|\lambda_2|)\geq\alpha \text{ and } P(t>|\lambda_3|)\geq\alpha \text{ and } P(t>|\lambda_4|)\geq\alpha \tag{19}$$

With conditions (17) and (19), a predicate function for region merging can be defined as follows:

$$C_1(v_1,v_2)=\begin{cases}\text{True} & \begin{array}{l}\text{if } \mu>0.5 \text{ and } P(t>|\lambda_2|)\geq\alpha \text{ and}\\ P(t>|\lambda_3|)\geq\alpha \text{ and } P(t>|\lambda_4|)\geq\alpha\end{array}\\ \text{False} & \text{otherwise}\end{cases} \tag{20}$$

Predicate $C_1(v_1,v_2)$ is in fact equivalent to an operator checking whether there exists an edge between nodes v_1 and v_2. This can be shown from the fact that a continuous gray level transition on the adjacent border ensures that there is no step edge there, while roof edges can be detected by testing the hypothesis that v_1 and v_2 have the same slopes. However, this edge operator is sensitive to noise on the adjacent border, especially when there are only a few voxels on the adjacent border. As a result, C_1 is more likely to report the existence of an edge. To compensate for this, we also consider the average difference between two nodes.

Let $(\boldsymbol{x}_i;g'_i), i=1,2,\ldots,N_1$, be N_1 voxels in node v_1, and $(\boldsymbol{y}_i;g''_i), i=1,2,\ldots,N_2$, be N_2 voxels in node v_2, and

$$\begin{aligned}\overline{\boldsymbol{x}} &= \frac{1}{N_1}\sum_{i=1}^{N_1}\boldsymbol{x}_i\\ \overline{\boldsymbol{y}} &= \frac{1}{N_2}\sum_{i=1}^{N_2}\boldsymbol{y}_i\end{aligned} \tag{21}$$

Consider

$$h_1=\frac{1}{N_2}\sum_{i=1}^{N_2}\left|g''_i-\boldsymbol{y}_i^T\hat{\boldsymbol{\xi}}_1\right| \tag{22}$$

*$\mathrm{A}_k^{-1}(i,j), k=1,2$, is the element at the ith row and the jth column of the matrix A_k^{-1}

†Since regions here are generally not symmetrical about coordinates, $\hat{a}_1-\hat{a}_2$, $\hat{b}_1-\hat{b}_2$ and $\hat{c}_1-\hat{c}_2$ are not necessarily independent. We have little hope to combine them in one statistical test.

$$h_2 = \frac{1}{N_1} \sum_{i=1}^{N_1} \left| g_i' - \boldsymbol{x}_i^T \hat{\boldsymbol{\xi}}_2 \right| \tag{23}$$

We take h_1 as the prediction error of the surface $\hat{\boldsymbol{\xi}}_1$ at the point $\overline{\boldsymbol{y}}$, and h_2 as the prediction error of the surface $\hat{\boldsymbol{\xi}}_2$ at the point $\overline{\boldsymbol{x}}$. Two nodes are then merged if both of these prediction errors are small enough. Similarly, we can derive a t-test for each of hypotheses $h_1 = 0$ and $h_2 = 0$, using statistics:

$$\lambda_5 = \frac{h_1}{\sqrt{1 + \overline{\boldsymbol{y}}^T \boldsymbol{A}_1^{-1} \overline{\boldsymbol{y}}}} \cdot \frac{\sqrt{N_1 - m}}{r_1} \tag{24}$$

$$\lambda_6 = \frac{h_2}{\sqrt{1 + \overline{\boldsymbol{x}}^T \boldsymbol{A}_2^{-1} \overline{\boldsymbol{x}}}} \cdot \frac{\sqrt{N_2 - m}}{r_2} \tag{25}$$

Given a significance level α, a predicate function associated with nodes v_1 and v_2 can be defined as follows:

$$C_2(v_1, v_2) = \begin{cases} \text{True} & \text{if } P(t > |\lambda_5|) \geq \alpha \text{ and } P(t > |\lambda_6|) \geq \alpha \\ \text{False} & \text{otherwise} \end{cases} \tag{26}$$

The final predicate function for merging two nodes is thus defined as follows:

$$C(v_1, v_2) = \begin{cases} \text{True} & \text{if } C_1(v_1, v_2) \text{ or } C_2(v_1, v_2) \\ \text{False} & \text{otherwise} \end{cases} \tag{27}$$

3.2.4 Clustering algorithm

Under the nearest neighbour criterion, two adjacent nodes are classified into the same class if they are mutually the nearest neighbours. The algorithm for clustering an adjacency graph G_l in the hierarchy is as follows:

1. Create a new graph G_{l+1}.
2. For each un-classified node v_i in G_l, do 3-6.
3. Find a node v_j which is the nearest neighbour of v_i. If v_j is not found, go to 6. Otherwise, continue.
4. If v_i is not the nearest neighbour of v_j, go to 6.
5. If v_j is un-classified
   (a) create a new node v;

(b) mark v_i and v_j as the children of v;

(c) mark v_i and v_j as classified;

(d) add v to G_{l+1};

(e) go to 2.

else

(a) assign the parent of v_j to v;

(b) mark v_i as the child of v;

(c) mark v_i as classified;

(d) go to 2.

6. (a) create a new node v;

(b) mark v_i as the child of v;

(c) mark v_i as classified;

(d) add v to G_{l+1};

(e) go to 2.

4 Implementation

The segmentation method presented in previous sections is for 3D images. Similar results can be obtained for 2D images, by simply dropping the parameter c in (1). For a linear model with m parameters, at least m pixels/voxels are needed to estimate these parameters. Here, a region with less than m pixels/voxels is considered as noise and merged to a neighbouring region which gives the smallest difference between their mean gray levels. On the other hand, the design matrix $\boldsymbol{A}$ will not be full ranked when pixels/voxels in the region are co-linear (for 2D images) or co-planar (for 3D images). A line of 1 pixel/voxel width and a slice of 1 voxel width is unlikely to be a separate region, and again is taken as noise regions. Thus, clustering at the base level is applied iteratively until all regions have more than m pixels/voxels and pixels/voxels in each region are not co-linear (for 2D images) or co-planar (for 3D images).

The method normally creates a high hierarchy. This is due to the fact that the merge is allowed only for nodes which are mutually the nearest neighbours, leading to a low condensation rate at each level. Maintaining the whole hierarchy in the entire procedure of the segmentation operation is not economic. Instead, we only keep two successive levels at any time by storing in each

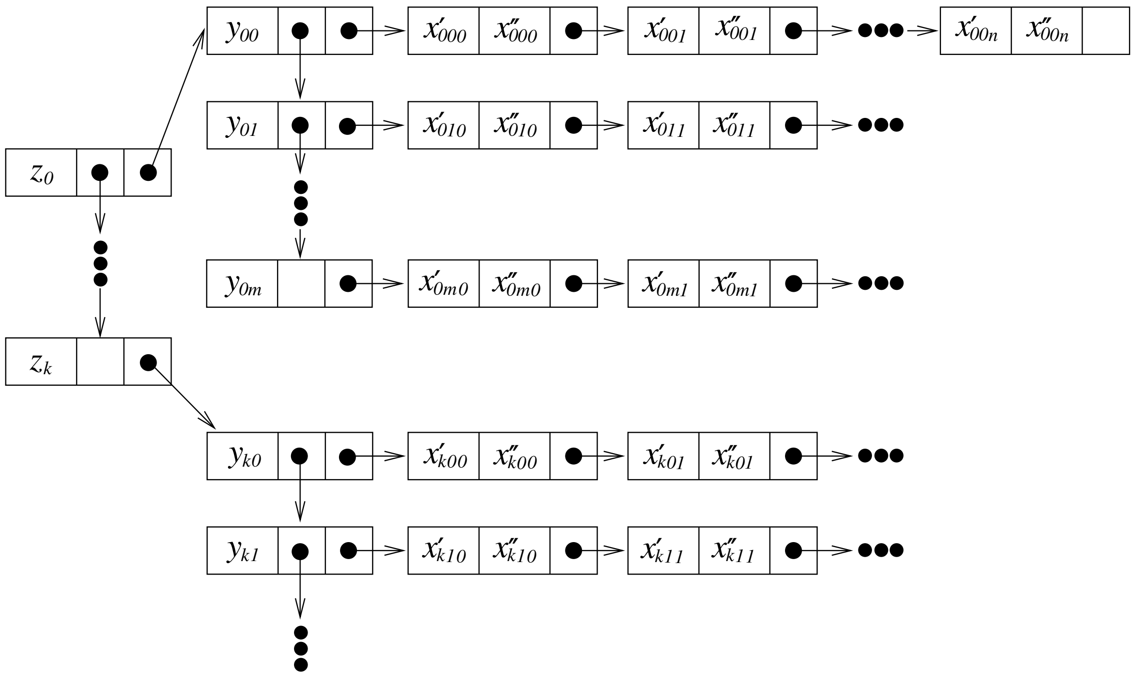


Figure 3: A representation scheme for 3D regions

node the corresponding segmented region. Clustering at each level of the hierarchy involves region operations such as merging two regions, accessing each pixel/voxel in the region, and finding pixels/voxels on the adjacent border of two regions. It is thus important to represent the segmented regions efficiently both in storage and region operations.

For 2D images, a run-length representation[28] is used to describe regions. For 3D images, we use an extended version of the 2D run-length representation[29, 30]. The idea is to create a run-length representation for each slice. Fig. 3 illustrates the corresponding data structure for representing a 3D region. Within the image, slices which are parallel to the X-Y coordinate plane (called X-Y slices) and contain voxels in the region are recorded by their corresponding Z coordinates. Within each X-Y slice, rows which are parallel to the X axis (called X-rows) and contain the voxels in the region are recorded by their corresponding Y coordinates. For each X-row, voxels in the region are recorded by run-lengths which are described by X coordinates (x' and x'') of the end-points of runs. Algorithms for efficient region operations based on this representation scheme have been devised [30].

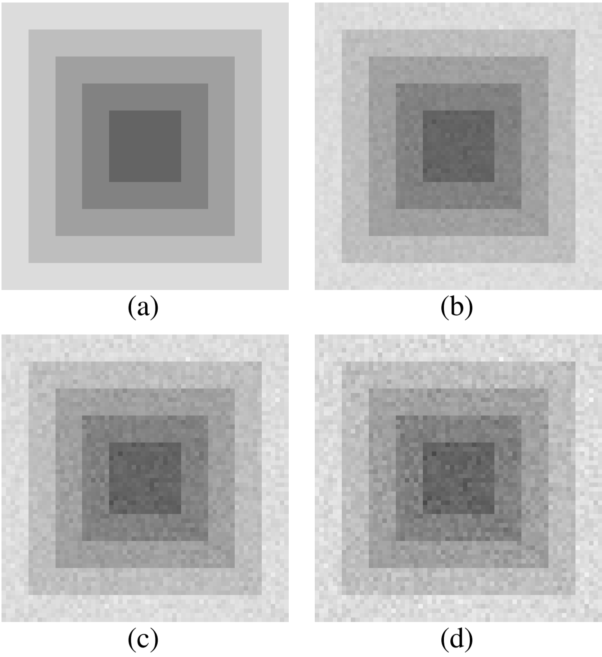


Figure 4: A $64 \times 64 \times 16$ synthetic image (a), and noisy images produced by adding white Gaussian noises with mean 0 and variances 16 (b), 36 (c), and 100 (d) respectively. For each image, only the X-Y slice at $Z = 0$ is shown.

5 Experimental results

Experiments are applied to both 2D and 3D images. In all of the experiments, regions with less than 27 voxels (9 pixels for 2D images) are considered as noise regions. A fixed threshold value $\delta = 5.0$ is used for defining the predicate function $C(v_1, v_2)$ at the base level.

5.1 Synthetic data

A $64 \times 64 \times 16$ synthetic 3D image is created. The image contains a $64 \times 64 \times 16$ block recursively embedded by a $(64 - 12 * k) \times (64 - 12 * k) \times 16$ block, $k = 1, 2, 3, 4$. Each X-Y slice has the same image. An X-Y slice at $Z = 0$ is shown in Fig. 4(a). From dark to bright, the gray level for each region is 100, 130, 160, 190, and 220 respectively. White Gaussian noise with mean 0 and different variances are added to the image. Fig. 4(b), Fig. 4(c), and Fig. 4(d) show the noisy images with variances of Gaussian noise 16, 36, and 64, respectively, by giving the X-Y slice at $Z = 0$ in each image. All these 4 images are processed using the significance level $\alpha = 0.001$. In each case, the correct number of regions are produced. The misclassification is mainly located

		original image	image b	image c	image d
region 1	P	4096	4095	4094	4124
	M	100	99.58	99.62	99.72
	S	0.0	3.99	5.96	7.96
region 2	P	8448	8449	8460	8384
	M	130	129.52	129.55	129.61
	S	0.0	4.03	6.02	8.0
region 3	P	13056	13054	13054	13153
	M	160	159.47	159.47	159.44
	S	0.0	3.98	5.97	8.11
region 4	P	17664	17666	17645	19711
	M	190	189.45	189.42	191.82
	S	0.0	4.01	6.0	10.33
region 5	P	22272	22272	22283	20164
	M	220	219.48	219.47	220.29
	S	0.0	4.03	6.03	7.83
% of misclassification		–	0.005	0.05	3.41

Table 1: Segmentation results after processing the images shown in Fig. 4. P – region population; M – mean gray level; S – standard deviation.

around borders of adjacent regions. Instead of showing every slice in the output image to see the segmentation result, we summarise the result in Table 1. It can be seen that the method produces a satisfactory result for images with reasonable amounts of noise.

5.2 Real data

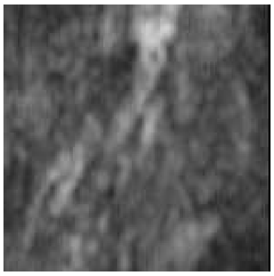
Both 2D and 3D real images are used for testing the algorithm. Fig. 5 is a 3D ($150 \times 150 \times 15$) seismic (variance filtered) image shown with each X-Y slice (150×150). The image contains two channels (with bright gray levels) which were eroded in the topography during a period of low sea level, and filled with sediment, different from the surroundings, at a later period in geological time. Their 3D structures are most important to geologists. Fig. 6 shows the segmentation result after processing the image with $\alpha = 0.001$. It can be seen that two channels are found fairly accurately (shown with bright regions). A 3D visualisation of these two channels is given in Fig. 7. Fig. 8 is a $150 \times 160 \times 15$ block from a 3D MRI brain image shown with each X-Y slice. Fig. 9 shows the segmentation result after processing the image with $\alpha = 0.005$. Details corresponding to the skull, structures surrounding the cerebrum, and gray and white matters of the cerebrum have been picked up. Fig. 10 shows the result by applying the method to a 2D outdoor image with $\alpha = 0.01$. It can be seen that important details like the shadow of the boat have been successfully segmented. Fig. 11 gives another result of segmenting a 2D aerial image.

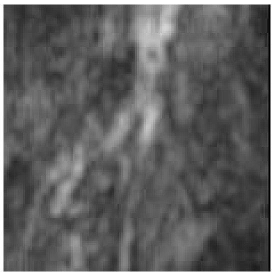
6 Discussion

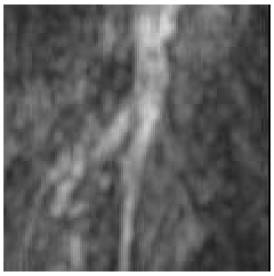
Image segmentation is an ill-defined problem, since what is really a physically meaningful region is often context dependent. It is difficult to construct a general region model based on the image information such as gray level and texture. Consequently, techniques of segmentation are often specific to a class of images where regions associated with the object in the image can be described with a mathematical model.

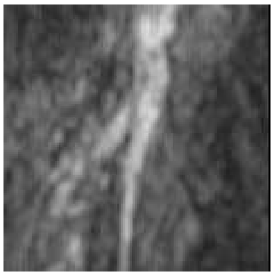
In this work, a physically meaningful region in an image is assumed to be a homogeneous region in which the gray level variation can be described by a linear polynomial corrupted with Gaussian noise. This assumption presents two kinds of problems: (1) not all object regions can be characterised by the homogeneity in gray levels; and (2) not all homogeneous regions are physically meaningful. Mathematically, this region model only characterises regions where the image illuminance is linearly dependent on spatial positions. It does not reflect any structured texture information (non-linear) associated with the object in the image. Thus, this work is mainly applicable to noisy images without containing strong textures. To tolerate the non-linear gray level variation, a higher order region model is required[(13)]. On the other hand, without priori knowledge about the object in the image, we never know whether a homogeneous region physically corresponds to an independent object region, a part of an object region, or simply a noisy area. It is generally agreed that a large homogeneous region is often a physically meaningful region, while a very small homogeneous region is possibly a result of the effect of noise. However, the determination of the noise region is still subjective. In this work, a homogeneous region with less than 27 voxels (9 pixels for 2D images) is taken as noise, and is merged with its nearest neighbour.

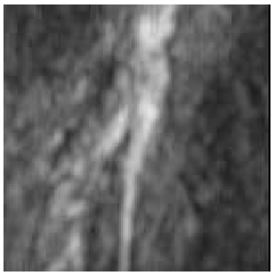
The linear model defined in (1) allows us to cluster the image under the framework of a hypothesis inference. Statistically-based methods are often more robust in dealing with noise than other methods based on thresholds. The main tuning parameter here is the significance level α which quantitatively relates the segmentation result to a probability for allowed misclassification. In this work, smaller values of α enforce the clustering operation to be more tolerant to the differences within the region in order to reduce the chance of excluding a voxel/pixel (or a sub-region) which belongs to the region. Thus, a small α encourages the merge, and leads to a smaller number of regions.

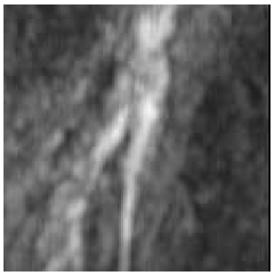

Slice at $Z = 0$

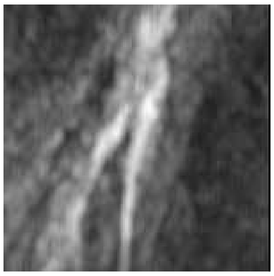

Slice at $Z = 1$


Slice at $Z = 2$

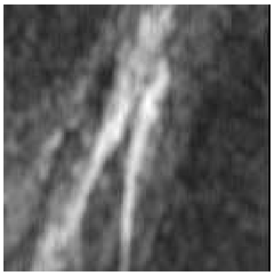

Slice at $Z = 3$

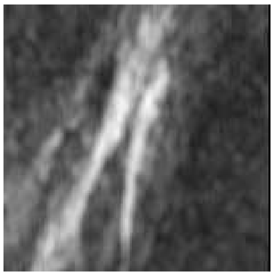

Slice at $Z = 4$

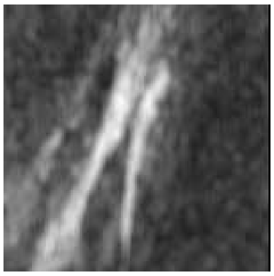

Slice at $Z = 5$

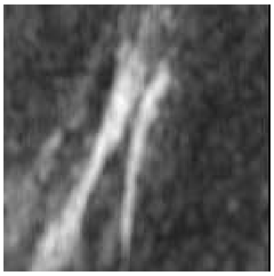

Slice at $Z = 6$

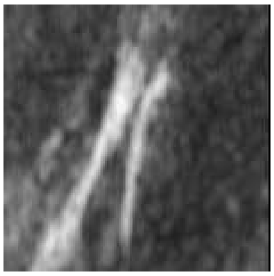

Slice at $Z = 7$

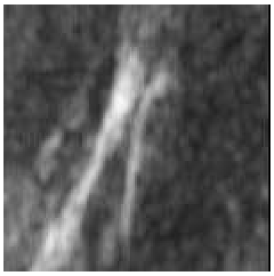

Slice at $Z = 8$


Slice at $Z = 9$


Slice at $Z = 10$


Slice at $Z = 11$


Slice at $Z = 12$


Slice at $Z = 13$

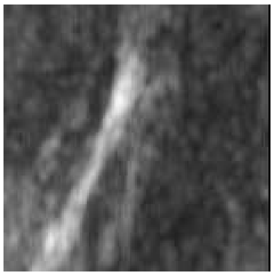
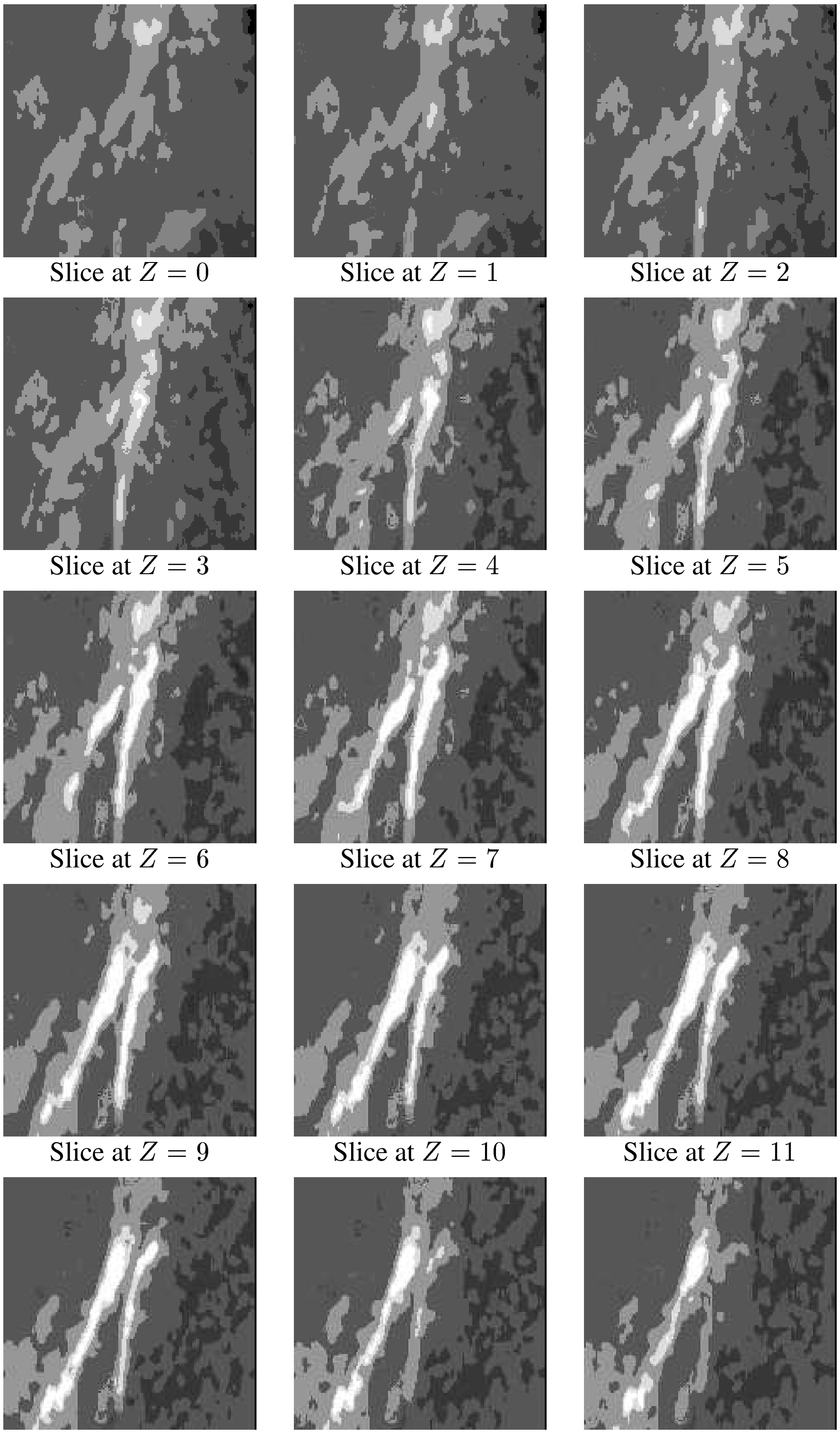

Slice at $Z = 14$

Figure 5: A $150 \times 150 \times 15$ 3D seismic (variance filtered) image shown with each X-Y slice.



Slice at $Z = 0$ | Slice at $Z = 1$ | Slice at $Z = 2$

Slice at $Z = 3$ | Slice at $Z = 4$ | Slice at $Z = 5$

Slice at $Z = 6$ | Slice at $Z = 7$ | Slice at $Z = 8$

Slice at $Z = 9$ | Slice at $Z = 10$ | Slice at $Z = 11$

Slice at $Z = 12$ | Slice at $Z = 13$ | Slice at $Z = 14$

Figure 6: Segmentation result after processing the image shown in Fig. 5 with $\alpha = 0.001$. The corresponding X-Y slices are shown.

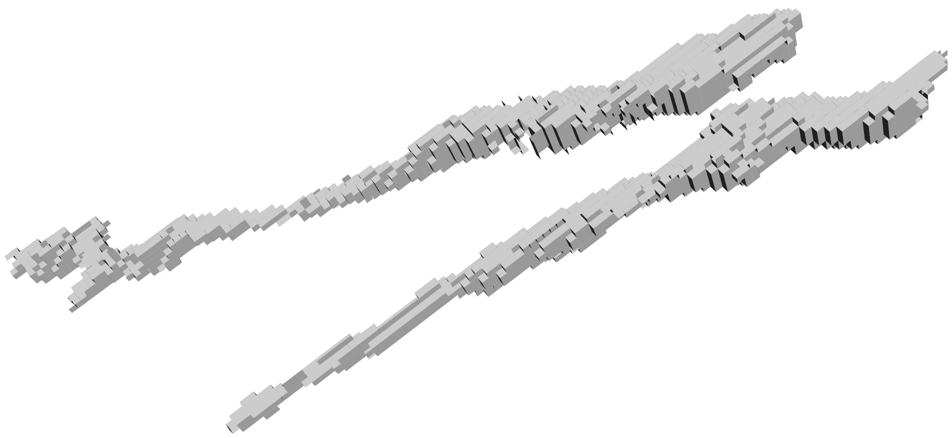


Figure 7: A 3D visualisation of two channel regions in the image shown in Fig. 6 (see text).

We achieve the segmentation through a bottom-up hierarchical analysis using a classification criterion based on the nearest neighbour defined in Section 3.2. Pure homogeneous (constant gray level) regions are always found first. This gives us a chance to perform some extra operations on these regions before any further merge operations. For example, if prior knowledge about the object regions is available, a knowledge-based analysis can be applied to these regions to determine whether a small region is physically meaningful in order to reduce possible errors due to the subjective treatment of noisy regions.

The nearest neighbour criterion for classification is widely used in many applications. Classifications so obtained are optimal with respect to the implied definition of the nearest neighbour. In this work, the nearest neighbour of a region is determined based on the difference between gray level surfaces of two regions, and the gray level transition on their adjacent border. However, other information such as region size and distance between centroids of regions may also be used. For example, to prevent a small region from being swallowed by a large region, we may incorporate the region size information so that the nearest neighbour of a region gives the smallest difference in region sizes. Embedding this kind of classification into a bottom-up hierarchical analysis, we can free the relinking operation which is normally required in other hierarchical methods in order to achieve a local optimisation of the classification. However, a higher hierarchy may be required to complete the segmentation.

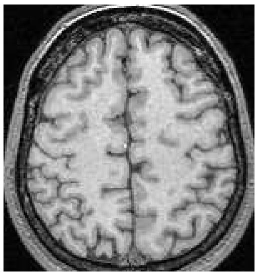
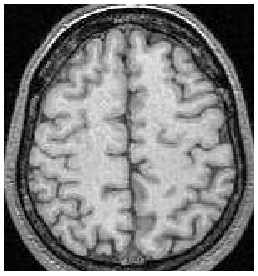
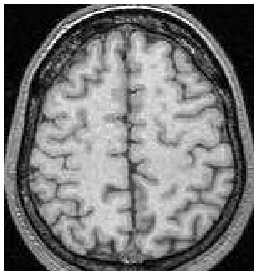
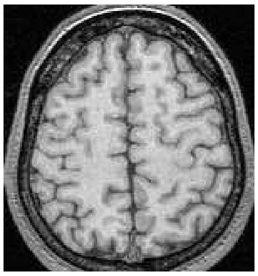
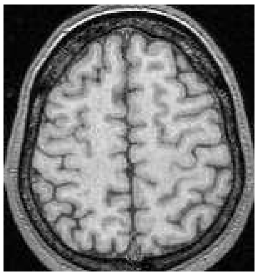
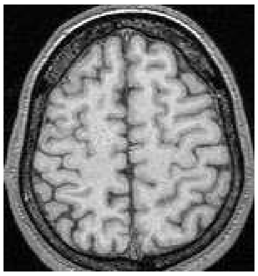
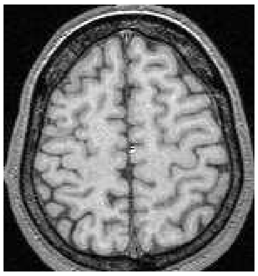
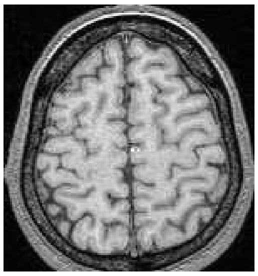
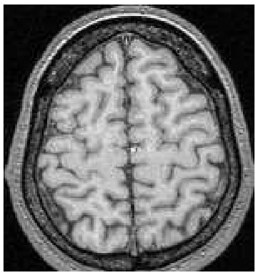
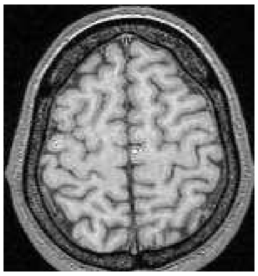
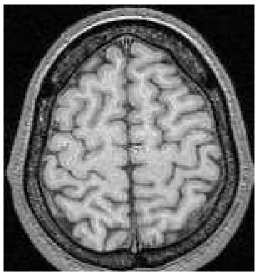
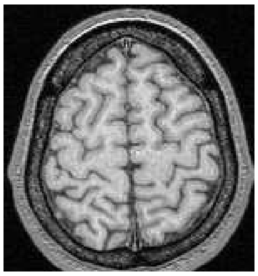
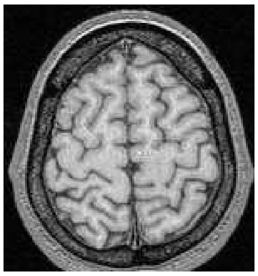
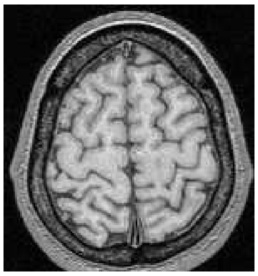
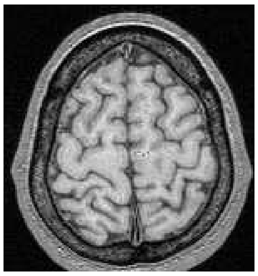
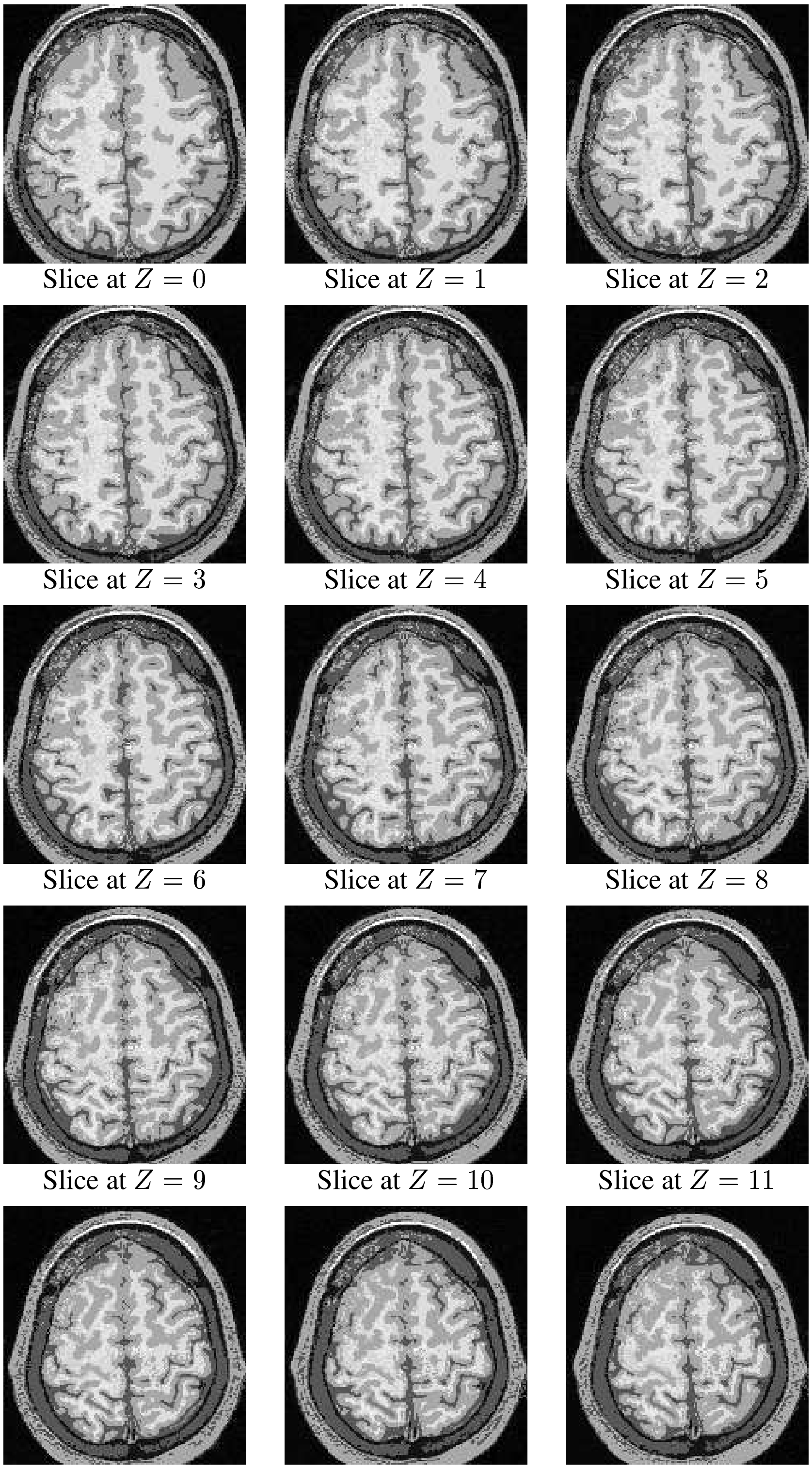

Slice at $Z = 0$

Slice at $Z = 1$

Slice at $Z = 2$

Slice at $Z = 3$

Slice at $Z = 4$

Slice at $Z = 5$

Slice at $Z = 6$

Slice at $Z = 7$

Slice at $Z = 8$

Slice at $Z = 9$

Slice at $Z = 10$

Slice at $Z = 11$

Slice at $Z = 12$

Slice at $Z = 13$

Slice at $Z = 14$

Figure 8: A $150 \times 160 \times 15$ 3D MRI brain image shown with each X-Y slice.



Slice at $Z = 0$ Slice at $Z = 1$ Slice at $Z = 2$

Slice at $Z = 3$ Slice at $Z = 4$ Slice at $Z = 5$

Slice at $Z = 6$ Slice at $Z = 7$ Slice at $Z = 8$

Slice at $Z = 9$ Slice at $Z = 10$ Slice at $Z = 11$

Slice at $Z = 12$ Slice at $Z = 13$ Slice at $Z = 14$

Figure 9: Segmentation result after processing the image shown in Fig. 8 with $\alpha = 0.005$. The corresponding X-Y slices are shown.

Figure 10: Experimental result on a 2D image. On the left is the original image. On the right is the result after processing the image with $\alpha = 0.01$.

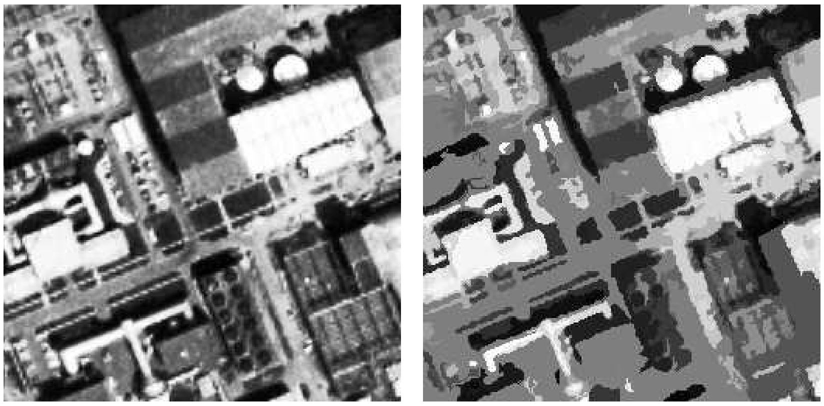


Figure 11: Experimental result on a 2D image. On the left is the original image. On the right is the result after processing the image with $\alpha = 0.01$.

7 Conclusion

We have proposed an unsupervised segmentation method applicable to both 2D and 3D images. Using a hierarchy of adjacency graphs to describe the irregular partitions of the image, the method achieves the segmentation by successively clustering regions at each level until no further region merging can be made. Regions in the image are characterised with a linear model whose parameters are estimated hierarchically using the least-squares method. By fitting the model to regions, a merge condition and a dissimilarity function defined on two adjacent regions are derived to define the nearest neighbour of a given region. Controlled by the nearest neighbour criterion, clustering at each level produces a classification which is optimal with respect to the definition of the nearest neighbour. Since the nearest neighbour is determined based on statistical inferences, the segmentation is, in principle, controlled by a significance level which can be selected in accordance with the statistical meaning without requiring any prior information. Experiments have been applied to both 2D and 3D images, and satisfactory results have been obtained.

Acknowledgement

This work was funded by the EPSRC IMV initiative (grant of GR/K 95987) and carried out in collaboration with Shell KSEPL.

References

1. R. Szeliski and D. Tonnesen. Surface Modelling with Oriented Particle System. *Computer Graphics*, 26(2):185–194, July 1992.

2. I. Cohen, L. D. Cohen, and N. Ayache. Using Deformable Surfaces to Segment 3-D Images and Infer Differential Structures. *Computer Vision, Graphics and Image Processing: Image Understanding*, 56(2):242–263, September 1992.

3. B. C. Vemuri, A. Radisavljevic, and C. M. Leonard. Multi-resolution Stochastic 3D Shape Models for Image Segmentation. In *13th International Conference on Information Processing in Medical Imaging*, pages 62–76, 1993.

4. R. Bajcsy and S. Kovacic. Multiresolution Elastic Matching. *Computer Graphics and Image Processing*, 46:1–21, 1989.

5. R. Haralick and L. Shapiro. Image Segmentation Techniques. *Computer Vision, Graphics and Image Processing*, 29:100–132, 1985.

6. N. R. Pal and S. K. Pal. A Review on Image Segmentation Techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

7. S. L. Tanimoto and T. Pavlidis. A Hierarchical Data Structure for Picture Processing. *Computer Graphics and Image Processing*, 4:104–119, 1975.

8. W. Kropatsch and A. Montanvert. Regular versus Irregular Pyramid Structures. In U. Eckhardt, A. Hubler, W. Nagel, and G. Werner, editors, *Geometrical Problems of Image Processing*, pages 11–22, Georgenthal, Germany, 1991.

9. A. Montanvert, P. Meer, and A. Rosenfeld. Hierarchical Image Analysis Using Irregular Tessellations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):307–316, April 1991.

10. P. Nacken. Image Segmentation by Connectivity Preserving Relinking in Hierarchical Graph Structures. *Pattern Recognition*, 28(6):907–920, 1995.

11. C. A. Bouman and B. Liu. Multiple Resolution Segmentation of Textured Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):99–113, February 1991.

12. C. S. Won and H. Derin. Unsupervised Segmentation of Noisy and Textured Images Using Markov Random Fields. *Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing*, 54(4):308–328, July 1992.

13. M. Spann and A. Grace. Adaptive Segmentation of Noisy and Textured Images. *Pattern Recognition*, 27(12):1717–1733, 1994.

14. C. A. Bouman and M. Shapiro. A Multiscale Random Field Model for Bayesian Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(2):163–177, March 1994.

15. P. Burt, T. Hong, and A. Rosenfeld. Segmentation and Estimation of Image Region Properties through Cooperative Hierarchical Computation. *IEEE Transactions on Systems, Man and Cybernetics*, 11(12):802–809, December 1981.

16. J. Cibulskis and C. R. Dyer. Node Linking Strategies in Pyramids for Image Segmentation. In A. Rosenfeld, editor, *Multiresolution Image Processing and Analysis*, chapter 8, pages 109–120. Springer Verlag, 1984.

17. M. Spann. Figure/Ground Separation Using Stochastic Pyramid Relinking. *Pattern Recognition*, 24(10):993–1002, 1991.

18. R. Haralick. Edge and Region Analysis for Digital Image Data. *Computer Graphics and Image Processing*, 12:60–73, 1980.

19. A. K. Jain. Advances in Mathematical Models for Image Processing. *Proceedings of IEEE*, 69:502–528, 1981.

20. P. Meer, D. Mintz, and A. Rosenfeld. Least Median of Squares Based Robust Analysis of Image Structure. In *Proceedings of DARPA Image Understanding Workshop*, pages 231–254, Pittsburgh PA, 1990.

21. P. Willemin, T. R. Reed, and M. Kunt. Image Sequence Coding by Split and Merge. *IEEE Transactions on Communications*, 39(12):1845–1855, 1991.

22. M. Spann and E. P. Ong. A Least-Median-of-Squares Based Region Growing Algorithm for Greylevel and Motion Segmentation, 1997. Submitted to *Computer Vision and Image Understanding*.

23. R. H. Laprade. Split-and-merge Segmentation of Aerial Photographs. *Computer Vision, Graphics and Image Processing*, 44:77–86, 1988.

24. C. R. Rao. *Linear Statistical Inference and Its Applications*. John Wiley & Sons, Inc., 2nd edition, 1973.

25. P. J. Huber. *Robust Statistics*. John Wiley & Sons, Inc., 1981.

26. F. Harary. *Graph Theory*. Addison-Wesley Publishing Company, 1969.

27. V. B. Dasarathy. *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, 1991.

28. R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*. Addison-Wesley Publishing Company, 1992.

29. W. N. Martin and J. K. Aggarwal. Volumetric Descriptions of Objects from Multiple Views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):150–158, March 1983.

30. X. Shen and M. Spann. 3D Region Representation Based on Run-lengths: Operations and Efficiency. *Pattern Recognition*. Accepted.