6

Processing tools for acoustic 3D images

V. Murino, M. Palmese and A. Trucco

6.1 INTRODUCTION

To investigate man-made objects lying on the seafloor or embedded in the bottom, acoustic imaging techniques and subsequent data processing can be applied. In this chapter, we address the use of acoustic image processing to design a complete system for segmentation, reconstruction and final virtual rendering of underwater sub-bottom objects.

The analysis of volumetric acoustic images is a research field that is evolving: conventional approaches have not been yet devised. At present, different applications and different sensing configurations address distinct problems; thus, *ad hoc* solutions have been developed to address the SITAR objectives. Nevertheless, in general, there is the need to remove noise from raw data by performing preliminary processing. Image processing techniques to improve amplitude/intensity image quality are quite commonly utilized, mainly at speckle reduction and contrast enhancement (Alexandrou and De Moustier, 1988; Malinverno *et al.*, 1990; Sauter and Parson, 1994).

After image filtering, more structured post-processing methods can be applied, especially segmentation and reconstruction techniques for high-level tasks, like classification and object recognition. Depending on the type of original data to be analyzed, a variety of methodologies have been proposed to tackle the aforesaid issues; so it is not possible to identify an algorithm or even a specific methodology as "the standard" (Caiti *et al.*, 2003), even if statistical approaches can be recognized as the most commonly used in the literature, especially for segmentation problems. After the identification of image regions corresponding to objects present in the ensonified volume, methods able to extract certain features aimed at a better understanding of the acoustic image can be applied. Actually, such methods differ according to the various cases considered, and are often adapted to the specific image and to the kind of image representation (Murino and Trucco, 2000). Finally, techniques aimed at improving scene comprehension resulting in a 3D augmented/virtual reality

representation can be very useful to assist a human operator to investigate the acoustic scene from multiple points of view (Murino and Trucco, 2000).

The processing chain presented in this chapter includes blocks devoted to noise reduction, statistical 3D segmentation, semi-automatic surface fitting, extraction of measurements and VRML (Virtual Reality Modeling Language) rendering. In addition, a multi-resolution data representation based on an octree approach can be applied at different steps of the chain, if necessary. In particular, the 3D segmentation method presented here is based on a volume growing approach, basically a 3D extension to traditional 2D region growing. Algorithm initialization consists in the choice of usually small volumes belonging to the different regions to be segmented. The volume growing operation is guided by a statistical approach based on optimal decision theory: at each step the algorithm estimates the parameters of a probability density function that adequately fits the current volume histogram. Concerning the surface fitting block, it is based on predefined geometric models: one of them is tentatively selected by the user after observation of the segmented object and is automatically or partially–manually adapted to the segmented data. Model invariance with respect to the pose and orientation of the object has been stressed and a metric to measure the goodness of the obtained results has been defined. The processing chain described in this chapter can be seen as a comprehensive tool that represents a novel and relevant achievement in the processing of underwater 3D images.

6.2 IMAGE PROCESSING CHAIN

The developed image analysis tool can be divided into four main phases (see Figure 6.1). Starting with the 3D raw image an optional filtering stage can be applied to reduce noise effects, thus preparing data for the segmentation task that allows one to partition the original image into regions representing the objects contained in the scene. The segmentation result is the input to the third phase, the surface reconstruction and parameter extraction stage, that based on geometrical models enables fitting a chosen shape to the cloud of points resulting from the previous step. Finally, the fourth stage is devoted to three-dimensional visualization of the obtained results. Moreover, an additional phase to generate a multi-resolution structure can be performed at any level of the processing chain, and hence it is described in Figure 6.1 as a transversal step to the typical scheme.

6.3 DATA REPRESENTATION

When a volume is organized as a 3D regular grid of voxels[1] it can be very efficient to build a multi-level structure. The octree is a multi-resolution pyramid for representation of three-dimensional spaces (Rodriguez *et al.*, 2000) where each level corresponds

[1] A voxel is a volume element, representing a value on a regular grid in 3D space (similar to the pixel, or picture element, in 2D space).

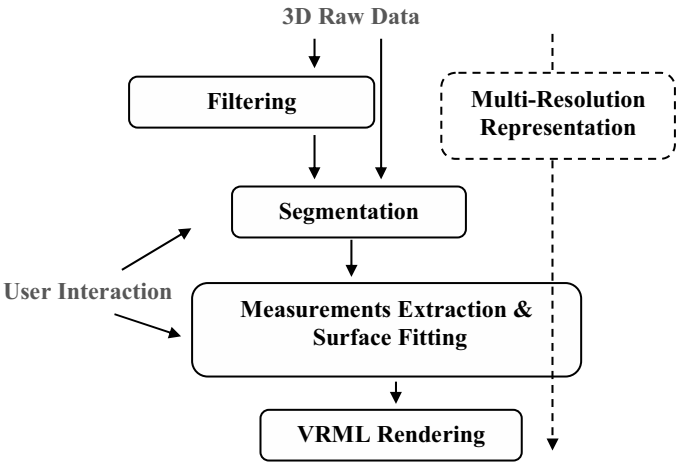

Figure 6.1. Image analysis processing scheme.

to a different resolution degree of the original image. The octree is built starting with the base of the pyramid – that is, from the finest-resolution level – and the resolution is iteratively reduced by spatially smoothing the available data. Considering the $(l-1)$-level, the corresponding image is divided in $2 \times 2 \times 2$ non-overlapping blocks; for each block the intensity levels of the non-empty voxels are averaged and will result in a single voxel in the upper level (l) (see Figure 6.2). By this procedure, it is possible to decrease data noise and to eliminate gaps in the data.

Hierarchical data representations make it possible to decide at which level of spatial resolution an algorithm can be applied. The same operation can work at the first level of the structure when the finest resolution is required, or it can be eased and sped up at a lower resolution level. A 3D image having a one-degree lower resolution in an octree scheme contains eight times fewer data, so it can be processed approximately eight times more quickly. Considering our processing scheme the octree structure can be applied at different levels of the chain depending on the current application:

- if gaps are present in raw data, the octree structure can be applied at the beginning of processing;

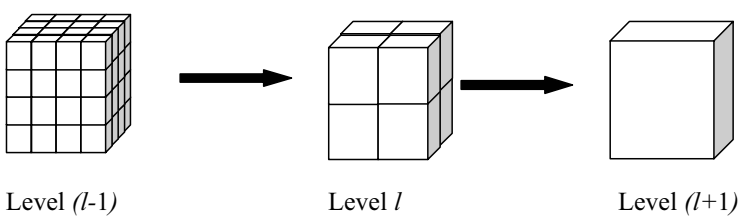

Figure 6.2. Octree structure generation procedure.

- when it is necessary to lighten the segmentation algorithm or the 3D rendering task, the operations can be performed at a poorer resolution level in the octree pyramid.

6.4 FILTERING

In addition to the spatial smoothing made possible by octree data representation, a 3D recursive median filter has also been implemented, in order to reduce speckle noise, which is characteristic of acoustic data. The median filter has been performed by a sliding three-dimensional mask with size $N \times N \times N$. The side dimension N must be an odd number and can be decided by the user according to the specific data. Appropriate size (from $3 \times 3 \times 3$ to $7 \times 7 \times 7$) must be chosen balancing the tradeoff between noise removal and image blurring. Moreover, as the median filtering is computationally intensive (the number of operations grows exponentially with the mask size) and to avoid an excessive loss in the data it is possible to apply a recursive implementation of the 3D filter choosing a small value for N. In this way, regions in which the signal period is lower than one-half the mask width will be continually altered by each successive pass until the resultant period is greater than one-half the window width, minimizing signal suppression. Also, the number of iterations can be adapted depending on the specific data needs.

6.5 VOLUMETRIC SEGMENTATION: STATE OF THE ART

To analyse 3D acoustic images of the sub-bottom and extract salient characteristics of buried objects a segmentation process is a fundamental step. The literature on the segmentation of 3D acoustic sub-bottom images is very limited, and the segmentation of 3D images is generally still considered a challenging problem for computer analysis. Therefore, it is necessary to also take into account 3D segmentation methods proposed for different application fields (e.g., medical imaging). Analysis of the state of the art has emphasized that the existing algorithms are difficult to classify in separate categories. Nevertheless, it is possible to subdivide them according to the following approaches: deformable surface models (balloons, level sets, bubbles), octrees, Markov random fields and volume growing.

In the balloon model, after initialization of a surface within the object of interest (manual procedure), the deformation process is based on minimization of the energy of a surface (Bowden *et al.*, 1997). Also, in the level set method (Baillard *et al.*, 2001) user interaction is required for the initialization task, but here the deformation surface is embedded at the zero level of a hyper-surface, which is the solution of a partial differential equation. With this formulation fast numerical methods can be applied to solve the problem. The issue of manual initialization can be resolved by the use of bubbles (Tek and Kimia, 1997). In the deformation process (reaction–diffusion process), after the random initialization of bubbles in the volumetric image, the bubbles grow, shrink, merge, split and disappear. For 3D acoustic images of the

sub-bottom a fast 3D segmentation method based on octrees was proposed (Rodriguez *et al.*, 2000). The method is highly robust to noise and gaps in the data by performing the data processing in a multi-resolution pyramid – an octree. The tree is built by spatially smoothing the available data. This reduces noise and the number of gaps facilitating classification at a low spatial resolution. The main drawback of this method is the low spatial resolution. A way to incorporate spatial correlation and *a priori* information in the segmentation process is by use of a Markov random field (MRF) (Choi *et al.*, 1997). The MRF itself is a *conditional* probability model, where the probability of a given voxel status depends on its neighbourhood; the segmentation procedure is formulated as a maximum *a posteriori* estimation problem. Finally, a volume growing approach, 3D extension of 2D region growing, can be applied to segment a 3D image (Umesh and Chaudhuri, 2000). The segmentation task starts with a voxel or a set of voxels which belongs to the object of interest, examines its neighbourhood and decides for each voxel whether it belongs to the same object or not, according to a given criterion (stochastic approach – that is, MRF – watershed segmentation, dilation–erosion process).

6.6 THE VOLUME GROWING APPROACH

3D segmentation is the basic activity, while the pre-processing operations (e.g., filtering and enhancement) are aimed at preparing data to obtain good segmentation results, and can be arranged after definition of the 3D segmentation strategy. Image segmentation consists in identifying, inside a volumetric image, regions that are considered homogeneous in terms of some criterion. In our case segmentation is achieved by a "seeded" volume growing procedure. The choice of this method has been motivated by the fact that the volume growing approach combines different information about the image voxels: it considers both the spatial connectivity between voxels belonging to the same region and the similarity of the voxel intensity levels. This procedure needs to be initialized by a human operator, whose task is to select a (usually small) number of voxels belonging to different classes present in the image, according to the operator's judgement. These initialization voxels are called "seeds" (Adams and Bischof, 1994). Starting with the seeds, at each step of the algorithm the regions grow by addition of connected voxels in accordance with a measure of distance. The output of the volume growing procedure is a set of labelled volumes, where a label indicates the membership of a voxel in a segmented object. It is worth highlighting that initialization of the developed algorithm has been simplified in order to set the user free from the hard task of choosing complex parameters. The selection of the seeds is fast and user-friendly: starting with a 2D slice usually placed in the middle of the original volume the operator has to identify different regions corresponding to the different classes to be segmented. This first step of the segmentation procedure is the only one that requires user interaction and in fact provides all the information needed by the algorithm to go on automatically until the end of processing.

Let us suppose that n seeds have been selected, each corresponding to a class H_i, $i = 1, \ldots, n$. At each step of the algorithm, new voxels are added to some of the classes H_i by the following procedure:

(*i*) the unallocated voxels that border at least one of the regions are examined and labelled according to a measure of distance;
(*ii*) if the considered voxel q has more than one neighbouring region, we have to decide to which region voxel q is to be added. We calculate the distances from all its neighbouring regions, and add q to the closest region;
(*iii*) the algorithm ends when all the voxels have been allocated to the classes or the classes do not grow any more.

Let us describe the segmentation iterative procedure in detail. Let Q be the set of all the unallocated voxels q which border at least one of the regions $H_1, \ldots, H_i, \ldots, H_n$ already classified after a generic number of iterations. Remember that at the beginning of the process the sets $H_1, \ldots, H_i, \ldots, H_n$ are equal to the seeds selected by the operator. Thus, the set Q is defined as follows:

$$Q = \left\{ q \;\middle|\; \left(q \notin \bigcup_{1}^{n} H_i \right) \wedge (q \in N(a)), \forall a \in H = \bigcup_{1}^{n} H_i \right\} \tag{6.1}$$

where $N(a)$ is the set of immediate neighbours of voxel a. In particular, our choice for the set $N(a)$ is composed of the six voxels that are connected to voxel a.

At each step of the algorithm all the voxels of the Q set are labelled according to a measure of distance. If voxel q is adjacent to a different H_i it will be assigned to the region that satisfies the aggregation criterion. If the voxel is connected to only one region but such a region does not satisfy the condition of association, it will be temporarily labelled and included in a set of uncertain voxels; at the end of the iterative procedure this set will be reconsidered on the basis of the updated measure of distance. It is emphasized that a voxel is associated to a region if and only if it is connected to that region, as we have assumed connected components are to be segmented. At the end of each iteration the parameters of the aggregation condition are updated on the basis of the new voxels added to the classes. The algorithm is repeated until all the voxels have been allocated to classes or the classes do not grow any more. The segmentation algorithm can be outlined as shown in Figures 6.3, 6.4 and 6.5 where the schemes of the initialization step, the recursive phase and the final processing are respectively presented.

6.7 MEAN INTENSITY AGGREGATION CRITERION

The choice of aggregation criterion is critical to the success of the segmentation task. Two different criteria have been used in the segmentation procedure: the first is based on the intensity mean value of current regions; the second is based on the fitting of

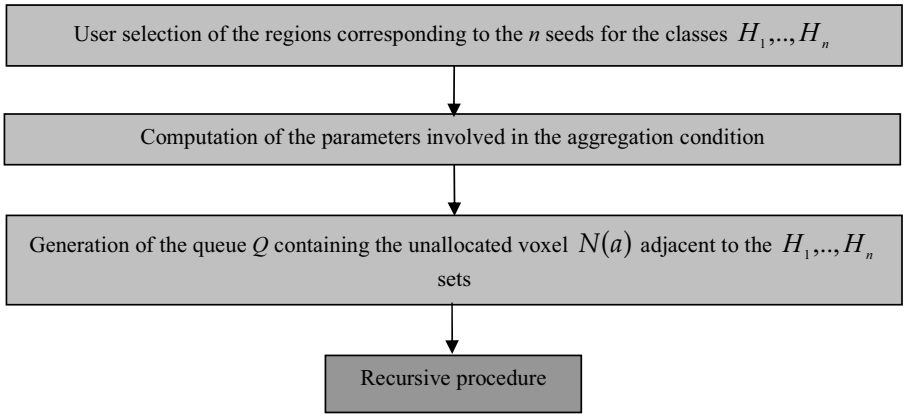

Figure 6.3. Scheme of the initialization step of the segmentation algorithm.

current volume histograms to an adequate probability density function (PDF). In the first case, the voxel is considered homogeneous to the region with the closest intensity mean value in terms of a measure of distance. The distance d is a simple quantity that shows how far the intensity (i.e., the scattering strength) of the considered voxel q is from the intensity mean value of the current region; it is defined as:

$$d_i(q) = \left| g(q) - \frac{\int_{H_i} g(y)\, dy}{\int_{H_i} dy} \right| \tag{6.2}$$

where $g(q)$ is the intensity level of voxel q. According to this criterion the algorithm chooses the kth region if the following expression is satisfied:

$$d_k(q) = \min_i \{d_i(q)\} \tag{6.3}$$

Actually, the effective assignment of voxel q to the chosen region must rely on the condition of spatial connectivity. This simple criterion is based on the hypothesis that assumes to work with image regions affected by Gaussian noise with zero mean and equal standard deviations. Even if for many images the assumption of equal variance noise is reasonable, we can modify distance d by taking into account the standard deviation SD of each growing region:

$$d_i(q) = \left| \frac{g(q) - \dfrac{\int_{H_i} g(y)\, dy}{\int_{H_i} dy}}{\underset{y \in H_i}{\mathrm{SD}}\,[g(y)]} \right| \tag{6.4}$$

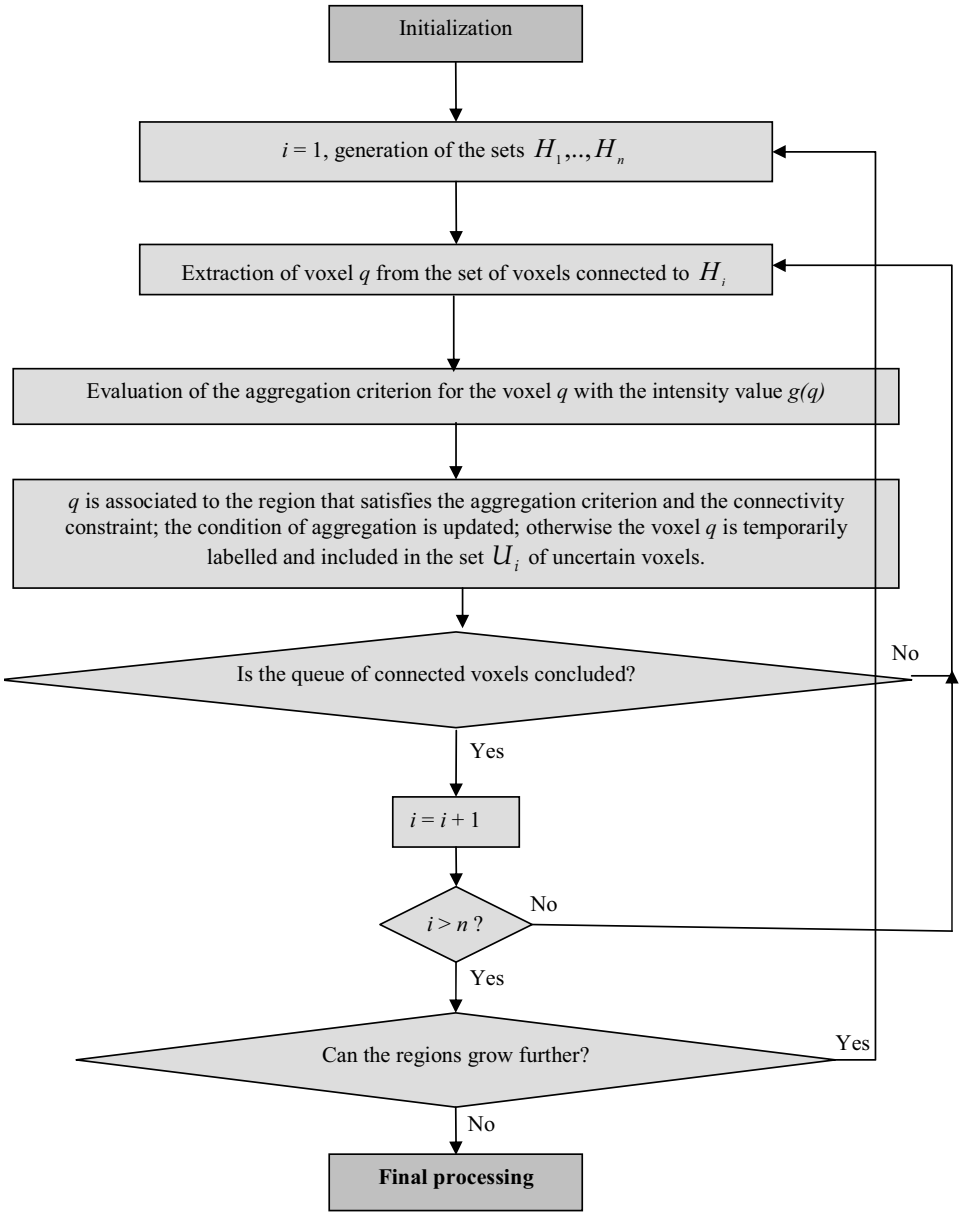

Figure 6.4. Scheme of the iterative phase of the segmentation algorithm.

However, this implementation of seeded volume growing is more computationally expensive without improving in an evident way the performances of the algorithm. Thus, when the assumption of Gaussian noise is not reasonable or *a priori* information is available a different formulation of the aggregation condition is necessary.

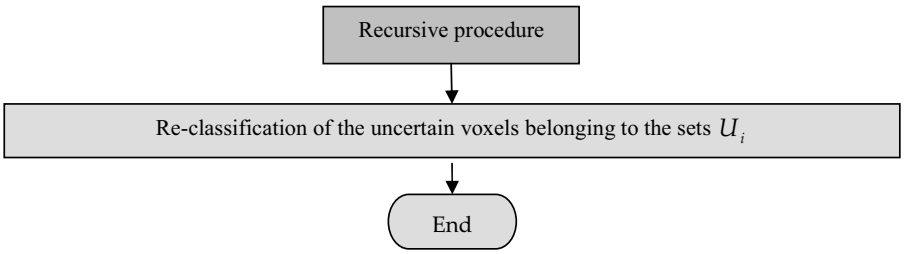

Figure 6.5. Scheme of the final step of the segmentation algorithm.

6.8 STATISTICAL AGGREGATION CRITERION

If *a priori* knowledge is available about the intensity distributions of the regions of the scene, it is possible to formulate a more accurate aggregation criterion taking into account second-order statistical information. This second solution (statistical volume growing) is based on the optimal decision theory: at each step of the algorithm and for each region, we calculate the current histogram and compute the parameters of the chosen density function (e.g., Gaussian, Rayleigh, Poisson density, etc.) that "best" fits the histogram. Each voxel connected to a region is added to that region if its intensity value satisfies the threshold condition based on the intersections of the current densities.

Voxel q, with an intensity level g and connected to a region H_i according to the estimated densities $p(g \mid H_i)$, will be assigned to that region if:

$$p(g \mid H_i) \geq p(g \mid H_k), \quad \forall k \neq i \tag{6.5}$$

This condition relies on the Bayesian approach based on the following hypothesis: if $\{H_1, \ldots, H_n\}$ are the available classes, the corresponding probability density functions $p(g \mid H_i)$ are known. Let us call $\{P_i\}$ the probabilities of the classes $\{H_i\}$ and R the Bayesian risk:

$$R = \sum_{i=1}^{n} \int_{Z_i} \left[\sum_{j=1}^{n} P_j \cdot C_{ij} \cdot p(g \mid H_j) \right] \cdot dg \tag{6.6}$$

where we have used the following notation: g is the voxel intensity level (feature) and is a stochastic variable, Z_i denotes the set of intensity levels belonging to the ith decision region, C_{ij} represents the cost we pay when voxel q is associated to H_i when it belongs to H_j. It is important to note that the probability $p(g \mid H_i)$ can assume the expression of any probability density function – for example, Gaussian or Rayleigh density – on the basis of statistical information about the specific component present in the 3D image. Moreover, we want to highlight that this criterion only takes into account the feature associated to each voxel; the spatial connectivity between the voxels belonging to the same region is considered through the volume growing process. In the case of non-sensitive cost classification the Bayes problem can be formulated as a minimization of

the probability of error. Starting with Equation (6.6), defining the costs C_{ij} as follows:

$$\left.\begin{aligned} C_{ii} &= 0 \quad \forall i \\ C_{ij} &= 1 \quad \forall j \neq i, \forall i \end{aligned}\right\} \tag{6.7}$$

we obtain the consequent expression for the Bayesian risk:

$$r = \left\{ \sum_{i=1}^{n} \int_{Z_i} \left[\sum_{j=1, j\neq i}^{n} P_j \cdot p(g \mid H_j) \right] \cdot dg \right\} \tag{6.8}$$

In addition if we assume equal *a priori* P_i, where

$$P_i = \frac{1}{n} \qquad i = 1, \ldots, n \tag{6.9}$$

the quantity to be minimized becomes:

$$R = \frac{1}{n} \cdot \left\{ \sum_{i=1}^{n} \int_{Z_i} \left[\sum_{j=1, j\neq i}^{n} p(g \mid H_j) \right] \cdot dg \right\} \tag{6.10}$$

Let us denote by Z the set of all possible values of g

$$Z = \bigcup_{i=1}^{n} Z_i \tag{6.11}$$

and let us select the kth decision region Z_k:

$$Z_k = Z - \bigcup_{i=1, i\neq k}^{n} Z_i \tag{6.12}$$

By extracting the kth element from the external sum of Equation (6.10) and by including Equation (6.12) in Equation (6.10), the formulation for the risk is given by:

$$R = \frac{1}{n} \cdot \left\{ \sum_{i=1, i\neq k}^{n} \int_{Z_i} \left[\sum_{j=1, j\neq i}^{n} p(g \mid H_j) - \sum_{j=1, j\neq k}^{n} p(g \mid H_j) \right] \cdot dg + \int_{Z} \left[\sum_{j=1, j\neq k}^{n} p(g \mid H_j) \right] \cdot dg \right\} \tag{6.13}$$

where it should be recalled that a probability density function satisfies the condition of unitary area:

$$\int_{Z} p(g \mid H_i) \cdot dg = 1 \quad \forall i \tag{6.14}$$

Taking into account Equation (6.14), Equation (6.13) can be rewritten as:

$$R = \frac{1}{n} \cdot \left\{ \sum_{i=1, i\neq k}^{n} \int_{Z_i} [p(g \mid H_k) - p(g \mid H_i)] \cdot dg + (n-1) \right\} \tag{6.15}$$

Let us focus our attention on the integrals contained in Equation (6.15): the minimization of these terms coincides with the choice of decision regions that allow one to

satisfy this condition:

$$\int_{Z_i} [p(g \mid H_k) - p(g \mid H_i)] \cdot dg < 0 \quad \forall k \neq i \tag{6.16}$$

namely, in all possible expressions of R in which the decision region Z_i appears, such a region will produce a negative contribution to the sum, thus reducing the Bayesian risk.

In conclusion, according to the Bayes criterion, we associate the voxel with intensity value g to the ith region if such a region satisfies condition (6.5): $p(g \mid H_i) \geq p(g \mid H_k)$, $\forall k \neq i$. It is worth observing that such a criterion follows the maximum likelihood (ML) rule, as we have assumed we are working with equal *a priori* probabilities. Actually, when this hypothesis is verified the ML method provides an optimal decision. Moreover, it is very favourable to adopt this method in terms of computational lightness. Furthermore, to lessen the computational load of the segmentation algorithm, it can be useful to update the parameters of the probability density functions of the regions only at the end of each iteration of the volume growing procedure. When there is lack of *a priori* information about the distributions of the intensity levels of the regions we are going to segment, the selection of a PDF with a single density shape will limit the performance of the segmentation approach. To overcome this problem it is possible to consider a density that allows us to control its kernel shape by selecting some parameters. The Weibull PDF is a family of densities of which the Rayleigh, Gaussian and exponential densities are special cases, corresponding to specific configurations of its parameters. The Weibull PDF of the intensity level g is given by:

$$f(g) = abg^{b-1} - e^{-ag^b} \quad g \geq 0, a, b > 0 \tag{6.17}$$

where a is a scale parameter and b is a shape parameter. Several methods, both graphical and analytical, are proposed in the literature to estimate the Weibull parameters (Al-Fawzan, 2005). Due to the high probability of error in using graphical methods and thanks to the availability of high-speed computers, we have preferred to use analytical methods –for example, method of moments (MOM), maximum likelihood estimator (MLE) and least squares method (LSM). The choice of the method depends on whether one needs a quick or an accurate estimate. In our case, to limit the computation time of the segmentation procedure, we have applied the last two methods, MLE and LSM, as – even though they yield less accurate results – they involve fewer calculations and require less computation time than MOM. In particular, LSM provides satisfactory estimates in a very short computation time. Examples of estimates are shown in Figure 6.6: the solid curve represents the histogram of the region we are considering; the dotted and the dashed lines indicate the Weibull PDFs estimated with LSM and MLE, respectively. It is possible to note that MLE is more accurate in fitting the original histogram (in the last part of the curve there is a total overlapping of the histogram with the estimated curve); nevertheless, LSM provides a good approximation in a shorter time without compromising the segmentation results.

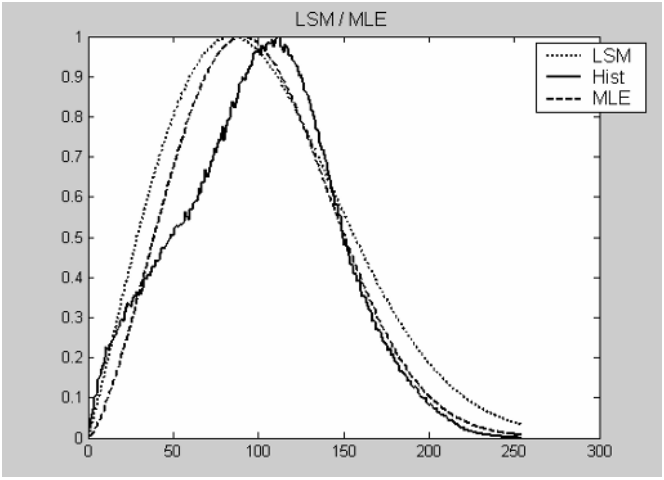

Figure 6.6. Examples of estimates of the Weibull PDF parameters by LSM and MLE.

6.9 PARAMETER EXTRACTION

Analysis of the segmentation results has the objective of extracting parameters useful to identify the spatial pose and the dimensions of objects eventually present in the scene of interest. To estimate such features a technique able to work with a wide assortment of object geometrical configurations has been developed.

The input to this step is a point distribution coming from the segmentation stage representing the natural or artificial objects we want to analyse. Generally, segmented data do not include the object in its wholeness, thus the algorithm must face the availability of partial data. The set of parameters we want to estimate concerns the pose of the object in space; namely, starting with an inertial coordinate system (ICS), defined by the vectors $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$, we want to obtain the object coordinate system (OCS), defined by the vectors $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3$, fixed with respect to the object and taking into account eventual object symmetry. The first parameter we intend to find is the barycentre of the target. Unfortunately, the typical shortage of available data makes it impossible to give a correct measure of barycentre coordinates without an accurate morphological investigation. Generally, acoustical systems allow one to image only the surface of the objects within the scene, as acoustical energy is almost totally backscattered and only in small part is refracted inside the object. Therefore, initially we will limit ourselves to computing the barycentre coordinates $\mathbf{b}$ of the cloud of points arising from the segmentation process. The origin of the OCS is placed coincident with the distribution barycentre. To evaluate the inclination of the object, the rotation of OCS with respect to ICS has been expressed by means of the Euler angles (φ, θ, ψ). Starting with the inertial coordinate system the first rotation is by angle φ about the $\mathbf{v}_3$ axis (obtaining the vectors $\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \bar{\mathbf{v}}_3$), the second rotation is by angle $\theta \in [0, \pi]$ about the current $\bar{\mathbf{v}}_1$ axis (obtaining the vectors $\check{\mathbf{v}}_1, \check{\mathbf{v}}_2, \check{\mathbf{v}}_3$), and the third rotation is by an angle ψ about the current $\check{\mathbf{v}}_3$ axis (again), obtaining in this way the OCS (see Figure 6.7).

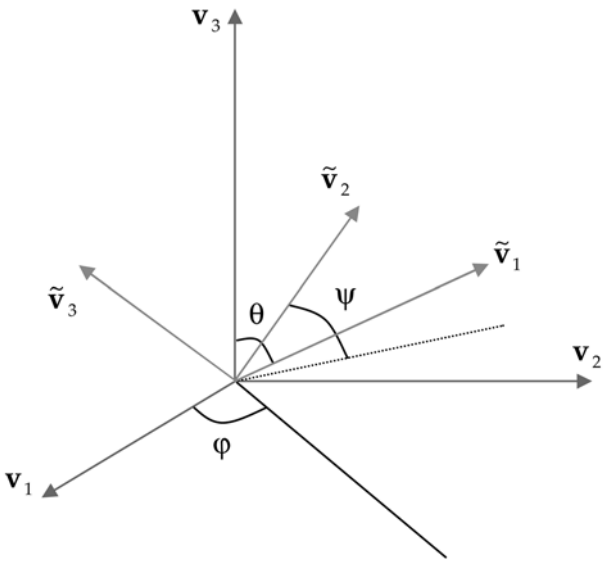

Figure 6.7. Adopted notation for the Euler angles, the ICS and the OCS.

Commonly, artificial objects can be assumed as rotational solids – namely, generated by a rotation of a surface about an axis. If we are in this condition we can neglect the ψ angle, as it does not provide any further information about the OCS. Therefore, for a rotational solid we can impose the following relation:

$$\psi = 0 \tag{6.18}$$

Let us assume we are working under this condition; to estimate the inclination of the object it is sufficient to compute the vector $\mathbf{t}$ defined as:

$$\mathbf{t} = [\theta \quad \varphi] \tag{6.19}$$

To obtain information on the orientation of a 3D distribution of points, it is possible to use a particular tensor associated to such a distribution, the inertial tensor. We assume that the point distribution has the same axes of symmetry of the object we want to reconstruct. Let us define the 3D distribution as the set of points in $\mathbf{R}^3$ with coordinates expressed in the ICS:

$$\{\mathbf{r}_i\} = \{x_i, y_i, z_i\}, \quad i = 1, \ldots, N \tag{6.20}$$

Let us denote by m_i the mass corresponding to the ith point of the distribution. It is known that the barycentre $\mathbf{b}$ of a discrete distribution of N points is given by the following expression:

$$\mathbf{b} = \frac{\sum_{i=1}^{N} m_i \cdot \mathbf{r}_i}{M} \tag{6.21}$$

where M is the total mass of the distribution, equal to $M = \sum_{i=1}^{N} m_i$. As all the distribution points have been classified as belonging to the same object by the segmentation process, they are assigned the same weight in the barycentre formula. Thus, we set

$$m_i = 1 \quad \forall i \tag{6.22}$$

By including Equation (6.22) in Equation (6.21) it is possible to obtain the final expression for the distribution barycentre:

$$\mathbf{b} = \frac{\sum_{i=1}^{N} \mathbf{r}_i}{N} \tag{6.23}$$

As previously mentioned, the origin of the OCS has been set in $\mathbf{b}$. Let us consider a temporary coordinate system TCS, defined by vectors $\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \hat{\mathbf{v}}_3$, generated by a rigid translation of ICS in $\mathbf{b}$. Let us denote by $\hat{\mathbf{r}}_i$ the position vectors of the N points with respect to this new coordinate system and by $(\hat{x}_i, \hat{y}_i, \hat{z}_i)$ the respective coordinates.

After computing the new position vectors, $\hat{\mathbf{r}}_i = \mathbf{r}_i - \mathbf{b}$, we proceed by estimating the axes of symmetry of the distribution (see Figure 6.8). If $\rho(\hat{x}, \hat{y}, \hat{z})$ is a continuous distribution in $\mathbf{R}^3$, we define the inertial tensor **IT** of ρ as a 3×3 matrix given by the following formula:

$$\mathbf{IT} = \begin{bmatrix} \int \rho(\hat{x},\hat{y},\hat{z}) \cdot (\hat{y}^2 + \hat{z}^2) \cdot dV & -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{x} \cdot \hat{y} \cdot dV & -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{x} \cdot \hat{z} \cdot dV \\ -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{x} \cdot \hat{y} \cdot dV & \int \rho(\hat{x},\hat{y},\hat{z}) \cdot (\hat{x}^2 + \hat{z}^2) \cdot dV & -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{y} \cdot \hat{z} \cdot dV \\ -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{x} \cdot \hat{z} \cdot dV & -\int \rho(\hat{x},\hat{y},\hat{z}) \cdot \hat{y} \cdot \hat{z} \cdot dV & \int \rho(\hat{x},\hat{y},\hat{z}) \cdot (\hat{x}^2 + \hat{y}^2) \cdot dV \end{bmatrix} \tag{6.24}$$

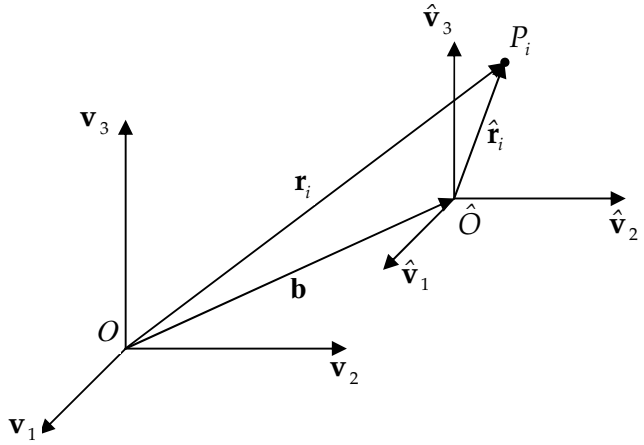

Figure 6.8. Rigid translation of the ICS in the distribution barycentre.

where dV is the infinitesimal volume element defined as:

$$dV = d\hat{x} \cdot d\hat{y} \cdot d\hat{z} \tag{6.25}$$

To obtain the same expression for a discrete distribution of N points we have to replace all the integrals with finite sums, and – taking into account the condition expressed by Equation (6.22) – we finally obtain:

$$\mathbf{IT} = \sum_{i=1}^{N} \begin{bmatrix} (\hat{y}_i^2 + \hat{z}_i^2) & -\hat{x}_i \cdot \hat{y}_i & -\hat{x}_i \cdot \hat{z}_i \\ -\hat{x}_i \cdot \hat{y}_i & (\hat{x}_i^2 + \hat{z}_i^2) & -\hat{y}_i \cdot \hat{z}_i \\ -\hat{x}_i \cdot \hat{z}_i & -\hat{y}_i \cdot \hat{z}_i & (\hat{x}_i^2 + \hat{y}_i^2) \end{bmatrix} \tag{6.26}$$

As **IT** is a real and symmetric matrix, it is always possible to perform a matrix diagonalization. This means that **IT** has non-degenerate eigenvalues e_1, e_2, e_3 and corresponding linearly independent eigenvectors. The eigenvalues of this tensor correspond to the rotation inertia of the distribution around the principal axes defined by the directions of the corresponding eigenvectors of **IT** (Giannitrapani *et al.*, 1999). In the case of a symmetric distribution, the eigenvectors of such a matrix, being the principal inertial axes, are symmetry axes too. Now, starting with the symmetry axes it is possible to get the Euler angles θ and φ necessary to describe the pose of the analysed object. We recall that angle ψ is neglected because of the hypothesis to work with a rotational object.

In Figure 6.9 the TCS denoted by vectors $\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \hat{\mathbf{v}}_3$ and the OCS denoted by $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3$ are shown; it is possible to note the θ and φ angles describing the rotation between the two systems.

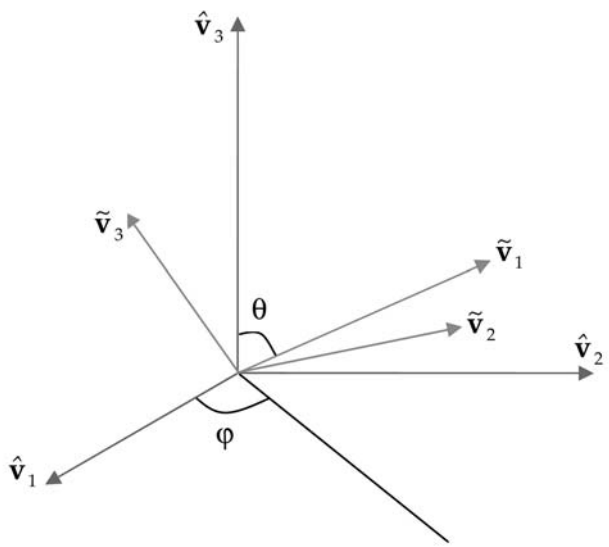

Figure 6.9. Rotation between the coordinate system denoted by vectors $\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \hat{\mathbf{v}}_3$ and the OCS expressed by the Euler angles.

To extract the coordinates of the OCS vectors with respect to the system TCS, it is necessary to compute the following scalar products:

$$\left.\begin{aligned}
\tilde{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_1 &= \sin(\theta)\cos(\varphi) \\
\tilde{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_2 &= \sin(\theta)\sin(\varphi) \\
\tilde{\mathbf{v}}_1 \cdot \hat{\mathbf{v}}_3 &= \cos(\theta) \\
\tilde{\mathbf{v}}_2 \cdot \hat{\mathbf{v}}_1 &= -\sin(\varphi) \\
\tilde{\mathbf{v}}_2 \cdot \hat{\mathbf{v}}_2 &= \cos(\varphi) \\
\tilde{\mathbf{v}}_2 \cdot \hat{\mathbf{v}}_3 &= 0 \\
\tilde{\mathbf{v}}_3 \cdot \hat{\mathbf{v}}_1 &= -\cos(\theta)\cos(\varphi) \\
\tilde{\mathbf{v}}_3 \cdot \hat{\mathbf{v}}_2 &= -\cos(\theta)\sin(\varphi) \\
\tilde{\mathbf{v}}_3 \cdot \hat{\mathbf{v}}_3 &= \sin(\theta)
\end{aligned}\right\} \tag{6.27}$$

The coordinates of the three vectors $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3$ with respect to the TCS are equal to:

$$\left.\begin{aligned}
\tilde{\mathbf{v}}_1 &= \begin{bmatrix} \sin(\theta)\cdot\cos(\varphi) \\ \sin(\theta)\cdot\sin(\varphi) \\ \cos(\theta) \end{bmatrix} \\
\tilde{\mathbf{v}}_2 &= \begin{bmatrix} -\sin(\varphi) \\ \cos(\varphi) \\ 0 \end{bmatrix} \\
\tilde{\mathbf{v}}_3 &= \begin{bmatrix} -\cos(\theta)\cdot\cos(\varphi) \\ -\cos(\theta)\cdot\sin(\varphi) \\ \sin(\theta) \end{bmatrix}
\end{aligned}\right\} \tag{6.28}$$

Therefore, the rotation can be expressed by the following 3×3 matrix:

$$\mathbf{R} = \begin{bmatrix} \sin(\theta)\cdot\cos(\varphi) & -\sin(\varphi) & -\cos(\theta)\cdot\cos(\varphi) \\ \sin(\theta)\cdot\sin(\varphi) & \cos(\varphi) & -\cos(\theta)\cdot\sin(\varphi) \\ \cos(\theta) & 0 & \sin(\theta) \end{bmatrix} \tag{6.29}$$

It is now clear that the θ and φ angles can be easily computed starting with the components of the generic vector $\tilde{\mathbf{v}}_k$ – that is, we can consider the vector $\tilde{\mathbf{v}}_1$ and find the parameters performing the following procedure. Let us denote by v_{kj} the jth component of the kth vector $\tilde{\mathbf{v}}_k$. A possible expression for the θ angle is:

$$\theta = \arccos(v_{13}) \tag{6.30}$$

In this way we obtain an angle belonging to the interval $[0, \pi]$ according to the Euler

convention. If $\sin(\theta) \neq 0$, the φ angle can be estimated starting with the φ' angle resulting from the expression:

$$\varphi' = \arccos\left[\frac{v_{11}}{\sin(\theta)}\right] \tag{6.31}$$

In this way we obtain again an angle belonging to the interval $[0, \pi]$, but to describe all the possible poses of an object it is necessary that

$$\varphi \in [0, 2\pi] \tag{6.32}$$

thus, the final φ is computed by taking into account the signs of the $\tilde{\mathbf{v}}_1$ components too:

$$\varphi = \begin{cases} \varphi' & \sin v_{12} \geq 0 \\ 2\pi - \varphi' & \sin v_{12} < 0 \end{cases} \tag{6.33}$$

Alternatively, if $\sin(\theta) = 0$, φ can be neglected as we are considering a rotational solid; thus, we can simply set $\varphi = 0$. In this way we have computed the angles θ and φ that identify univocally the direction of $\tilde{\mathbf{v}}_1$, which is a symmetry axis of the analysed object; we do not know yet which is the eigenvector that corresponds to such $\tilde{\mathbf{v}}_1$. As we are considering rotational solids, the computed angles make sense only if $\tilde{\mathbf{v}}_1$ is the rotation axis of the object. To identify the rotation axis among the three eigenvectors, user interaction is required. 3D data are projected onto the planes normal to the eigenvectors and the operator must choose the projection orthogonal to the axis of symmetry, identifying in such a projection an object with a circular (or partially circular) symmetry.

In conclusion, the developed method allows one to find the symmetry axes of an object starting with the corresponding point distribution. The proposed technique presents some limits when the point distribution resulting from the segmentation process represents only a fraction of the interested object, and thus is not characterized by the same properties of symmetry. This shortage of data produces two main effects:

- the barycentre position results up-shifted;
- the point distribution loses the symmetries of the original object, compromising computation of the θ angle.

To tackle this problem an *ad hoc* technique has been developed. The aim is to extract from the available data a portion that preserves the symmetry properties of the object. This operation works as a sort of further stage of segmentation to separate different regions of the same object and take out the region useful for the successive step. The first processing step concerns the extraction of the skeleton (Giannitrapani *et al.*, 1999; Murino and Giannitrapani, 1999). Before describing the applied procedure, a rough definition of skeleton is proposed – that is, a skeleton is a distribution of 3D points that:

(1) must be thinner than the original one;
(2) must be located in the neighbours of the median lines of the original point set;
(3) must have the same homotopy group; and
(4) have to be invariant to 3D rotation.

The third point requires, essentially, that the operations performed on the distribution to extract the skeleton preserve such topological information as the number of "holes" in the distribution and their relative position. In short, skeleton extraction is performed along the following lines. Let us define the 3D image as the ensemble of points in $\mathbf{R}^3$:

$$I = \{\mathbf{x}_i\} = \{x_i, y_i, z_i\}, \quad i = 1, \ldots, N \tag{6.34}$$

We also define for each point i and each R in $\mathbf{R}^+$ (ray of a sphere about the point i) the subset O_i^R of I and the 3D point $\mathbf{b}_i^R$ defined in the following way:

$$O_i^R \equiv \{\mathbf{x}_j \in I : |\mathbf{x}_i - \mathbf{x}_j| < R\} \tag{6.35}$$

$$\mathbf{b}_i^R \equiv \frac{\sum_{\mathbf{x}_j \in O_i^R} \mathbf{x}_j}{\dim\{O_i^R\}} \tag{6.36}$$

where $\dim\{O_i^R\}$ is the cardinality of O_i^R (i.e., $\mathbf{b}_i^R$ is the centroid of O_i^R). Let us define the *interior* λ_i^R for the point i in the following way:

$$\lambda_i^R \equiv \frac{R - |\mathbf{x}_i - \mathbf{b}_i^R|}{R} \tag{6.37}$$

One can easily verify that:

$$0 \leq \lambda_i^R \leq 1 \quad \forall i \tag{6.38}$$

The interior is a measure of how much a point is "inside" the object specified by I. If $\lambda_i^R \approx 1$, the point is inside a homogeneous zone of radius at least equal to R; in contrast, for $\lambda_i^R \approx 0$, the point is very near to a border of the three-dimensional distribution of points. Defining the following image transformation as:

$$I \to I^R = \{\mathbf{b}_i^R\} \tag{6.39}$$

and indicating by the symbol $I^{R,n}$ the iterative application of it for n times, our skeleton extraction is simply the construction of the image $I^{R,n}$ for a suitable choice of R and n. The overall effect of this transformation is to shift points on the border – that is, points with a low λ^R – toward the centre while leaving points well inside an object – that is, points with an almost unitary value of λ^R – unaltered. The iterative application of such a procedure tends to shift all the points of the distribution towards the skeleton. Such an algorithm may be seen from a physical point of view as a short-range interaction between physical points.

The choice of free parameter R is very important. If it is too small, with respect to the average mutual distances of the points, the distribution would likely collapse in a certain number of disconnected punctual regions, preventing extraction of the skeleton. If it is too large the value of λ may also be small for points well inside the object and again the skeleton will not be properly extracted. The parameter R plays a similar role to that of the dimension of the structuring element in classical mathematical morphology. It is straightforward to verify that the skeleton thus extracted is invariant

for three-dimensional rotation (a condition that is never exactly satisfied for skeletons of a 2D bitmap image). Using *a priori* knowledge of the structure represented by the 3D image and of sensor resolution, it is possible to estimate proper values for R and n aimed at improving the extraction of a good skeleton. Thus, the choice of radius R has been formulated as:

$$R = \frac{ris}{2} \tag{6.40}$$

where *ris* is sonar system resolution. Also, the number of iterations n is relevant to obtain a good skeleton extraction. Unfortunately, it is difficult to find an objective measure of the goodness of a skeleton without *a priori* information about the real object shape. Thus, the stop criterion we have selected is based on local properties of the skeleton itself, in particular on the average number of points $\bar{n}_v$ which is contained in a neighbourhood. If the relative variation of $\bar{n}_v$ with respect to the previous value is little this means that points density is going to stabilize. In this situation it can be convenient to stop the iterative procedure. In particular, considering the kth iteration, the relative variation $\bar{n}_r(k)$ is given by the following formula:

$$\bar{n}_r(k) = \frac{\bar{n}_v(k) - \bar{n}_v(k-1)}{\bar{n}_v(k)} \tag{6.41}$$

where $\bar{n}_v(k)$ and $\bar{n}_v(k-1)$ are the average numbers of points in a neighbourhood during the kth and $(k-1)$th iteration, respectively. Such $\bar{n}_r(k)$ should be compared with an adequate threshold, chosen *ad hoc* for the specific case.

In Figure 6.10 the algorithm of skeleton extraction is outlined. In particular, we have highlighted the three successive phases:

(1) initialization (dark grey);
(2) recursive procedure (light grey); and
(3) final step (darker grey).

Because of the recursive nature of the algorithm and the possibility of working with a large number of distribution points, skeleton extraction could result in an onerous operation. To limit computational time, it is necessary to choose a threshold that minimizes the number of iterations and, at the same time, allows one to obtain a good contraction of the distribution. Figures 6.11 and 6.12 show a point distribution arising from the segmentation process and the relative extracted skeleton, respectively. The considered object is a cylinder, whose inclination is defined by non-null θ and φ; it is possible to note that the part of the cylinder imaged by the sonar system – and thus segmented – coincides with the upper surface of the whole target.

This is an example of partial data that requires the step of skeleton extraction to separate an object region needed to estimate the inclination of the whole object. In this specific case we can set apart the upper base of the cylinder and then perform the inertial tensor method to compute the Euler angles. We suppose that we have an estimation of the φ angle, provided by preliminary application of the inertial tensor

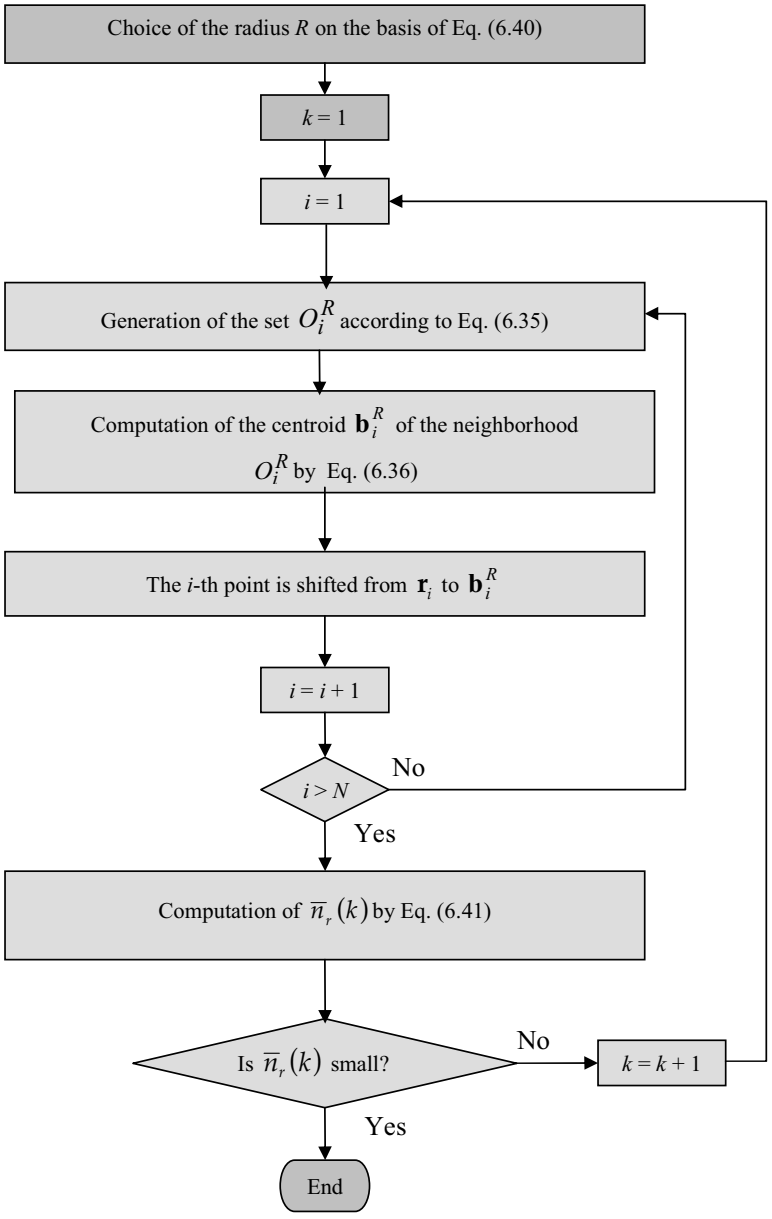

Figure 6.10. Scheme of the skeleton extraction procedure.

method. Our aim is to obtain an accurate measure of the θ angle, which provides the inclination to the normal to the upper cylinder base with respect to the vertical direction. The plane containing the upper base of the cylinder can be easily selected within the object skeleton: it can be identified as the plane that satisfies the following

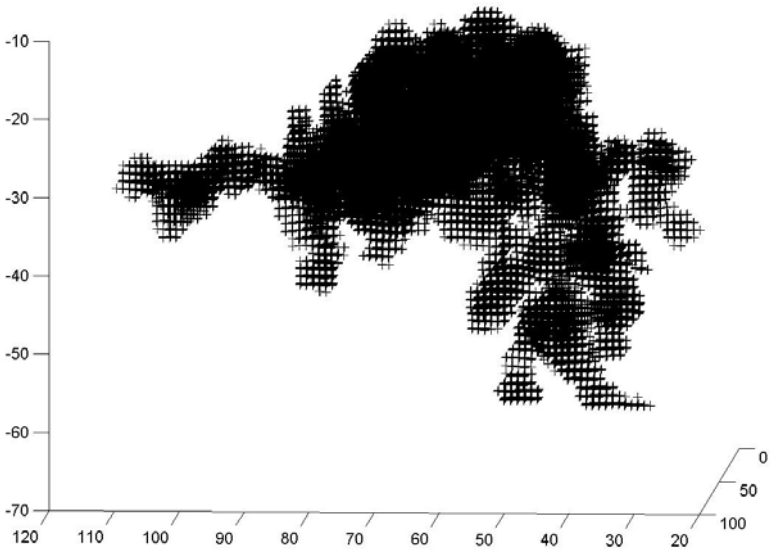

Figure 6.11. Segmented region corresponding to a leaning cylinder object.

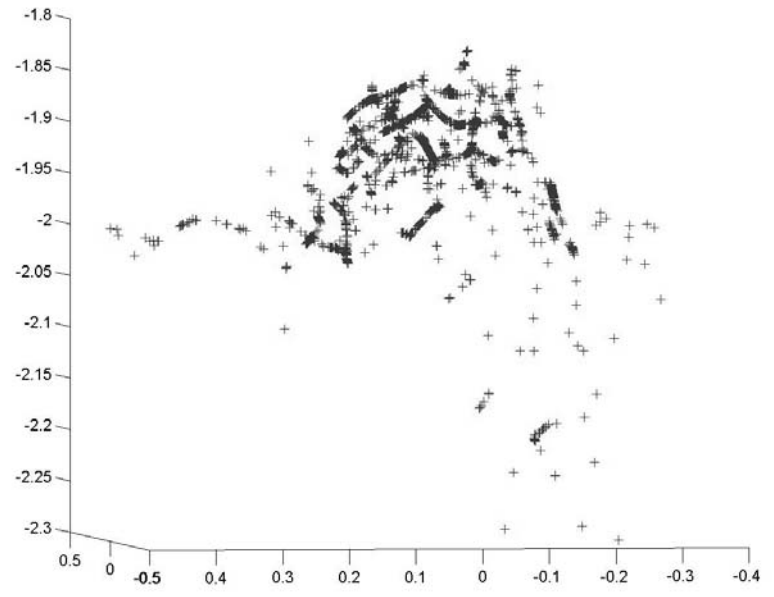

Figure 6.12. Skeleton extracted by the distribution of points of Figure 6.11.

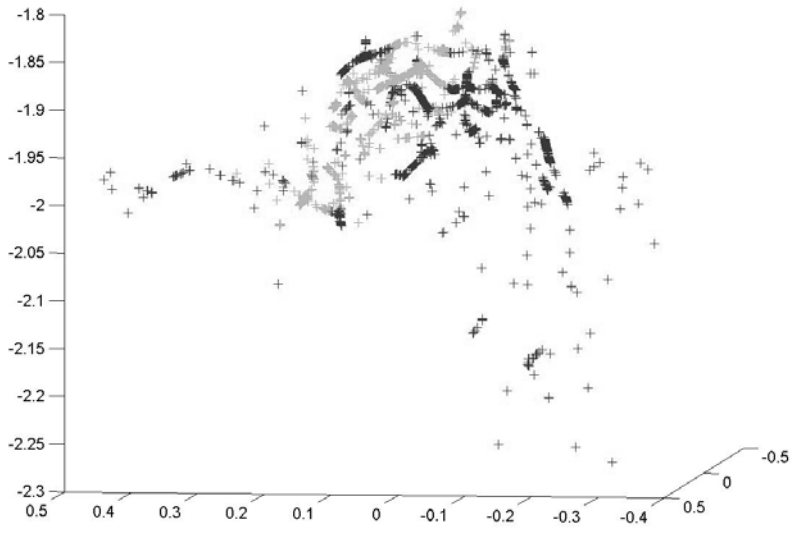

Figure 6.13. The upper base (light grey) of the cylinder has been selected starting with the skeleton of Figure 6.12.

conditions:

- it contains the highest point of the distribution, $P_S = (x_s, y_s, z_s)$;
- it presents a thickness which is related to the system resolution, *ris*, and thus, it is composed of the points that are placed between the planes:

$$\left.\begin{aligned}\sin(\theta)\cdot\cos(\varphi)\cdot[x-x_S]+\sin(\theta)\cdot\sin(\varphi)\cdot[y-y_S]+\cos(\theta)\cdot[z-(z_S+ris/2)]=0\\ \sin(\theta)\cdot\cos(\varphi)\cdot[x-x_S]+\sin(\theta)\cdot\sin(\varphi)\cdot[y-y_S]+\cos(\theta)\cdot[z-(z_S-ris/2)]=0\end{aligned}\right\} \tag{6.42}$$

where φ is supposed known, whereas θ is the parameter we want to determine. By assigning different θ we find different planes; the "best" θ is the one that allows us to include the greatest number of points of the distribution between the considered planes. It is then possible to improve the measures of the Euler angles by applying the inertial tensor method to the extracted region. Figure 6.13 shows the same skeleton as Figure 6.12 where the extracted base has been painted light grey.

6.10 CYLINDER MODEL

Once the parameters defining the pose of the analysed object have been computed, the successive step concerns the estimation of features related to the physical dimensions of the target. This task is strongly dependent on the geometrical shape of the considered object; thus, we have developed a surface reconstruction method *ad hoc* for

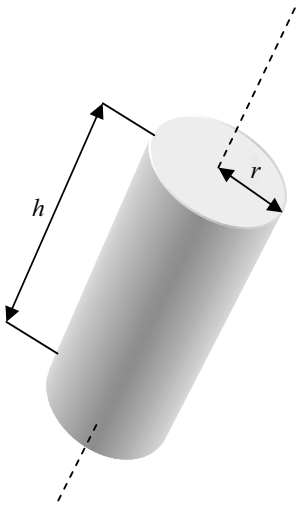

Figure 6.14. Cylinder model parameterized by radius r and height h.

specific object models. We briefly describe the technique for the cylinder model; it is straightforward to extend the procedure to any other shape that can be expressed in an analytical form.

The model refers to a circular right cylinder of height h and radius r. This phase is performed considering the points of the distribution related to the coordinate system placed on the object (OCS), denoted by vectors, $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3$. It is recalled that the OCS has the origin placed in the barycentre of the distribution that generally does not coincide with the true barycentre of the target; in this phase it is thus necessary to update the barycentre estimation by taking advantage of further analyses. In Figure 6.14 the adopted cylinder model is represented.

As in underwater acoustics the scattered energy is totally produced by the object's surfaces, the assumed model is based on the assumption that all the points of the distribution are placed at the same distance (equal to r) from the rotation axis, except for the points located on the eventual upper base of the cylinder that will be contained in a circle of radius r. To obtain an estimation of the radius r, we can project all the points onto a plane normal to the cylinder axis and interpolate the projections with a circumference. Actually, if the projected points are too spread out, it can be useful for an operator to interact by mouse-clicking to select the points belonging to such a circumference. At the end of this procedure we obtain a set of coordinates (x_i^L, y_i^L) corresponding to points placed at an average distance r from the axis of the cylinder. It is now possible to estimate the radius r and the centre coordinates (x_C, y_C) of the circumference that best interpolates the selected points.

The generic equation of a circumference with radius r and centre (x_C, y_C) is the following:

$$(x - x_C)^2 + (y - y_C)^2 = r^2 \tag{6.43}$$

that can be written also in this way:

$$x^2 + y^2 + a \cdot x + b \cdot y + c = 0 \tag{6.44}$$

where the coefficients a, b, c are equal to:

$$\left.\begin{aligned} a &= -2 \cdot x_C \\ b &= -2 \cdot y_C \\ c &= x_C^2 + y_C^2 - r^2 \end{aligned}\right\} \tag{6.45}$$

We can apply Equation (6.44) to the M points denoted by (x_i^L, y_i^L), thus obtaining a linear system of M equations and three unknown variables a, b, c, where $M \gg 3$. In general, such a system is over-determined and there is no solution. In this case it is possible to find a least squares solution, obtaining an estimation for the unknown parameters a, b and c.

Finally, the coordinates of the barycentre projected onto the plane orthogonal to the cylinder axis are equal to:

$$(x_C, y_C) = \left(-\frac{a}{2}, -\frac{b}{2}\right) \tag{6.46}$$

whereas the radius of the cylinder is determined by the following expression:

$$r = \sqrt{\left(-\frac{a}{2}\right)^2 + \left(-\frac{b}{2}\right)^2 - c} \tag{6.47}$$

We have completed the analysis on the plane normal to the cylinder axis; now we must get some information about the axis itself. To estimate the height of the cylinder the simplest method consists in computing the maximum and minimum heights (q_{max} and q_{min}) of the points along the cylinder axis. With respect to the OCS such heights can be expressed by:

$$\left.\begin{aligned} q_{max} &= \max_i\{\tilde{x}_i\} \\ q_{min} &= \max_i\{\tilde{x}_i\} \end{aligned}\right\} \tag{6.48}$$

Therefore, h is equal to:

$$h = q_{max} - q_{min} \tag{6.49}$$

In an analogous way, the height of the barycentre along the symmetry axis can be computed as the average value between q_{min} and q_{max}:

$$q_C = \frac{q_{min} + q_{max}}{2} \tag{6.50}$$

Starting with the coordinates of the barycentre $\mathbf{b}_C = (q_C, x_C, y_C)$ with respect to the OCS – considering the vector inclination $\mathbf{t}$ (6.19) – and the radius and height estimations r and h, we are able to unambiguously represent the cylinder.

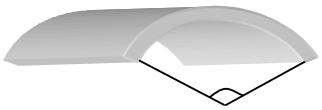

Figure 6.15. Cylinder slice model.

Actually, the developed method provides for creating a model which is also a portion of the cylinder. For example, in the event the segmented region corresponds to a longitudinal slice of a cylinder surface (see Figure 6.15), the user can choose to visualize the whole reconstructed cylinder or only the relative portion.

6.11 VRML RENDERING

The development of an efficient technique of three-dimensional visualization can be a useful step in the human interpretation of both segmentation results and the respective model reconstructions. The operator can explore the ensonified scene and take decisions supported by a better understanding of the objects present in the area of interest. To this aim we have performed a 3D rendering by means of the Virtual Reality Modeling Language, VRML. The VRML format allows one to build a file standardized by ISO/IEC to describe volumetric interactive scenes and worlds. This standard is especially good at visualizing point distributions as segmented data and synthetic representation as reconstructed data, allowing us to explore the obtained results in an accurate way.

By means of VRML a virtual world can be easily described using a classic editing application, by adopting semantics independent of the physical devices or any other implementation-dependent concepts. VRML is also intended to be a universal interchange format for integrated 3D graphics and multimedia. VRML browsers, as well as authoring tools for the creation of VRML files, are widely available for many different platforms. The separation of the descriptive and implementative aspects allows a VRML file containing the information relative to a complex virtual world to be very compact and easily shared on the web. Furthermore, the rendering quality achievable with VRML is very high.

Concerning the visualization of segmented data and object models, VRML offers further advantages. In a VRML file it is possible to define an inertial coordinate system and thus refer all the object local systems with respect to it. This allows one to describe in a simple way the translations and the rotations of the segmented objects. Besides, the VRML semantic already provides the definitions of some geometric primitives – as cylinders, cones, boxes and spheres – allowing one to describe directly some object models. Moreover, it is possible to add new types (called prototypes) which are not provided by the standard semantic and useful to describe more complex shapes, such as truncated cones and rings. Both the standard primitives and the prototypes are parameterized and thus can be adapted to the characteristics of the objects (Figure 6.16).

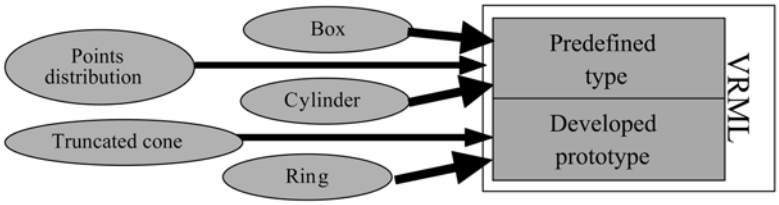

Figure 6.16. VRML standard primitives and prototypes.

6.12 CONCLUSIONS

In this chapter, a complete processing tool of use in the examination of 3D acoustic sub-bottom images has been described. The processing chain allows one to separate – from raw data – the image regions representing natural or artificial objects buried beneath the seafloor. In particular, the devised segmentation process is based on a semi-automatic ''seeded'' volume growing approach. The voxel classification is guided by a statistical criterion by fitting current volume histograms with an adequate probability density function. The segmented object is then analysed to extract measurements about its shape and pose and to obtain a 3D virtual representation by VRML modellling. In addition, a pre-processing noise reduction stage and a multi-resolution data representation based on an octree approach can be applied, if necessary.