There are several commercial and academic software tools that support different segmentation algorithms. In general, commercial software packages have better implementation (with a user-friendly interface for manual and semi-automatic segmentation methods), but they often lag behind the latest developments in the field. In contrast, academic software packages, such as ITK,[15] BioImage Suite,[16] MIPAV[17] and ImageJ,[18] tend to be more oriented toward single-modality applications and less friendly in handling multi-modality images, as proposed here.

Most automatic algorithms attempt to utilize image intensity variations or image gradient information. However, for low-contrast images, many of these algorithms tend to provide suboptimal solutions that are not clinically acceptable. For such cases, it has been demonstrated that if multiple images are available for the same object (same image modality or different image modalities), all of the available complementary information can be fed into the segmentation algorithms to define the so-called biophysical target.[19] Thus, the segmentation algorithms would benefit from the complementary information from different images, and consequently the accuracy of the final segmentation results could be improved. Similar approaches have been applied to detect the blood-wall interface of heart ventricles from CT, MRI, and ultrasound images using a snake deformable model;[20] to classify coronary artery plaque composition from multiple contrast MR images using the K-means clustering algorithm;[21] and to define tumor target volumes using PET/CT/MR images for radiotherapy treatment planning by using a multivalued, deformable level set approach, as in our previous work. Mathematically, such an approach is a framework that could be thought of as a mapping from the imaging space to the "perception" space identified by radiologists:[19]

$$\text{Biophysical target} = f(CT, PET, MRI, \ldots; \lambda), \tag{3.1}$$

where $f(\cdot)$ is the mapping function from the different imaging modalities to the target space parameterized by λ, which represents the users' defined set of parameters representing prior knowledge. This framework is highlighted in Fig. 3.1.

Despite the opportunities presented by this framework for streamlining the integration of multiple imaging modalities for better tissue classification or target definition, there are several challenges that should be addressed before clinical implementation can be achieved. First, image misalignment is an issue when dealing with images acquired from different scanners. This is partially resolved for PET/CT but not for many other image modalities. Therefore, methods for image registration should be incorporated into the framework. The second and more-challenging issue is the characterization of the mapping in Eq. (3.1) because it relies on translating higher-level human expertise into cues that computer algorithms

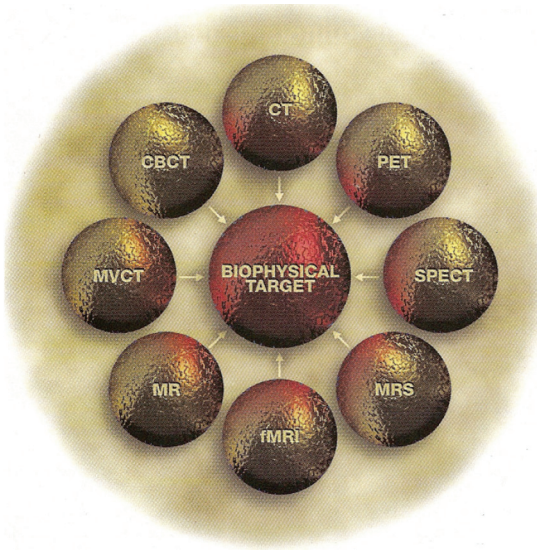

Figure 3.1 Biophysical target generated from multimodality imaging by combining anatomical and functional information.

can understand. These challenges have motivated us to develop a software tool to support such a multimodality image segmentation framework that can potentially learn information from the user's interactions. In our recent work, we attempted to resolve some of these problems by developing a dedicated software tool for multimodality image analysis called MIASYS.[10] This software is the first tool to offer a dedicated and comprehensive framework to cope with the emerging needs of therapeutic and diagnostic radiological applications.

3.3 Methods for Multimodality Image Segmentation

There are several methods that have been proposed to integrate multimodality imaging information, primarily by extending automated and semi-automated single- or monomodality segmentation methods into an interactive multimodality framework in which the available complementary information can be fed into the segmentation algorithms to define the biophysical target (as described earlier). Thus, the segmentation algorithms would benefit from the complementary information provided by different images, and consequently the accuracy of the final segmentation results could be improved. Similar approaches have been applied to detect the blood-wall interface of heart ventricles from CT, MRI, and ultrasound images using a snake deformable model (Sebbahi et al., 1997); to classify coronary artery plaque composition from multiple contrast MR images using the K-means clustering algorithm;[22] and to define tumor

target volumes using PET/CT/MR images for radiotherapy treatment planning by using a multivalued deformable level set approach.[19] This approach could be applied to several segmentation algorithms that are amenable to such generalization,[10] as discussed in the following sections.

3.3.1 Multiple-image thresholding

Thresholding is one of the most basic image segmentation methods; it is commonly used to delineate objects with high contrast with respect to the surrounding image background. Threshold values can be selected experimentally, e.g., in detecting tumors in PET based on cutoff values of the standardized uptake value (SUV), a threshold value is usually selected as SUV > 2.5 or 40% of maximum SUV.[23] In another example, an optimal threshold image-intensity value could be selected iteratively to separate the lungs from the body and chest wall structures.[24]

One way to expand the thresholding method to support hybrid images involves applying different threshold values to the different images and combining the thresholding results for different images in logical ways to form the final result. Previous work by Yang et al.[10] used the "thresholding conditions" notion to describe such multiple-image thresholding operations. For example, a thresholding condition could be "Im1 $<$ 100 & Im2 $>$ 50 | 20 $<$ Im3 $<$ 150," where Im1, Im2, and Im3 denote the intensity values of images 1, 2, and 3. MIASYS, for instance, is able to interpret the meaning of such a thresholding condition expression and carry out all of the mathematical and logical computations to yield the final combined result. The software tool is implemented with MATLAB, and it accepts any valid MATLAB expression as a thresholding condition. The expressions can contain any arithmetic, logical operators, and parentheses. The new thresholding method, which accepts such mathematical expressions, is flexible, easy to use, and very powerful.

3.3.2 Clustering algorithms

Clustering algorithms are frequently used for different image analysis problems such as image segmentation, object recognition, and image retrieval.[25] For image segmentation, they are used to automatically discriminate different tissue types based on primitive image features such as image intensity. One of the most commonly used algorithms is the K-means algorithm. Another algorithm found to be more robust is the fuzzy C-means (FCM) algorithm.

3.3.2.1 Fuzzy C-means algorithm

Similar to other clustering algorithms, the goal of the FCM algorithm[26] is to divide the image histogram into a few clusters and to iteratively find

the center of each cluster by minimizing the following system energy function:

$$J(x,c)=\sum_{i=1}^{N}\sum_{k=1}^{K}\|\mathbf{x}_i-\mathbf{c}_k\|^2, \tag{3.2}$$

where $\mathbf{x}_i$ is the image intensity for pixel i, N is the total number of pixels in the image, K is the total number of clusters, and $\mathbf{c}_k$ is the cluster center intensity value for cluster k.

In the FCM algorithm, a fuzzy membership function is defined and computed as Eq. 3.3, and the cluster center $\mathbf{c}_k^n$ is updated according to Eq. (3.4):

$$\mathbf{u}_{ik}^n=\frac{\|\mathbf{x}_i-\mathbf{c}_k^n\|^{-2}}{\sum_{k=1}^{K}\|\mathbf{x}_i-\mathbf{c}_k^n\|^{-2}}, \tag{3.3}$$

$$\mathbf{c}_k^{n+1}=\frac{\sum_{i=1}^{N}(\mathbf{u}_{ik}^n)^b\mathbf{x}_i}{\sum_{i=1}^{N}(\mathbf{u}_{ik}^n)^b}, \tag{3.4}$$

where $\mathbf{u}_{ik}^n$ is the fuzzy membership probability that image pixel $\mathbf{x}_i$ belongs to cluster k at iteration n, $\mathbf{c}_k^n$ is the updated cluster center intensity value for cluster k at iteration n, and b is a user defined parameter, where $0<b<1$.

The user starts the FCM method by setting the value of K, after which the $\mathbf{c}_k^0$ can be automatically and randomly initialized, and the iterations are repeated by computing Eqs. (3.3) and (3.4), respectively. The iterations stop when $\mathbf{c}_k^n$ is stabilized.

3.3.2.2 Extending the fuzzy C-means algorithm to multiple images

The FCM method can be naturally expanded to support multiple images by defining $\mathbf{x}_i$ and $\mathbf{c}_k^n$ as vectors instead of scalar values:

$$\mathbf{x}_i=\langle x_{i,1},x_{i,2},\ldots,x_{i,M}\rangle, \tag{3.5}$$

$$\mathbf{c}_k^n=\langle c_{k,1}^n,c_{k,2}^n,\ldots,c_{k,M}^n\rangle, \tag{3.6}$$

where M is the total number of images in the multimodality image data set, $x_{i,M}$ is the image pixel intensity value for the pixel i in the image m ($1<m<M$), and $\mathbf{c}_{k,m}^n$ is the cluster center intensity value for the cluster k in the image m for iteration n.

With $\mathbf{x}_i$ and $\mathbf{c}_k^n$ defined as vectors, Eqs. (3.3) and (3.4) can be rewritten as

$$\mathbf{u}_{ik}^n=\frac{\left(\sum_{m=1}^{M}\alpha_m(x_{i,m}-c_{k,m}^n)^2\right)^{-2}}{\sum_{k=1}^{K}\left(\sum_{m=1}^{M}\alpha_m(x_{i,m}-c_{k,m}^n)^2\right)^{-2}}, \tag{3.7}$$

$$\mathbf{c}_{k,m}^{n+1} = \frac{\sum_{i=1}^{N} (\mathbf{u}_{ik}^{n})^{b} \mathbf{x}_{i,m}}{\sum_{i=1}^{N} (\mathbf{u}_{ik}^{n})^{b}}, \quad m = 1, M, \tag{3.8}$$

where α_m is the user-defined weighting parameter for image m.

In this case, the computation of the fuzzy membership value $\mathbf{u}_{ik}^{n}$ is contributed by all of the images in the multimodality image dataset. The new weighting parameters α_m control the contribution from the different images according to the users' prior knowledge.

The FCM algorithm works in the image intensity histogram domain; image pixel spatial information is not considered in the algorithm, and there is also no difference in applying this algorithm to 2D images or to 3D volume images. Better FCM algorithms that consider image pixel spatial information, which may be added to the software tool in the future, have been reported in the literature.[27,28]

3.3.2.3 K-means clustering algorithm

The K-means algorithm[29] is actually a precursor of the FCM algorithm. It uses the hard membership function instead of the fuzzy membership function. The performance of the K-means algorithm is generally comparable to but less robust than the performance of the FCM algorithm. However, both algorithms may suffer from lacking a proper spatial neighborhood definition, which is addressed in the active contour algorithms.

3.3.3 Active contour algorithms

Deformable models are geometric representations of curves (in 2D) or surfaces (in 3D) that are defined explicitly or implicitly in the imaging domain. These models deform under the influence of force-like equations that are computed from the image data.[30,31] Contours of structures in the images are characterized by sharp variations in the image intensity, and therefore the deformable models can be warped to match the contours by means of energy minimization.[31–33]

So-called "snake" algorithms were among the first deformable models developed.[34] Snakes use an explicit parametric representation of the object boundary that deforms by means of energy minimization (or dynamic force equilibrium). Mathematically, if the deformable model is represented by

$$C(s) = \{x(s), y(s), z(s)\}, \ s \in [0, 1], \tag{3.9}$$

then its movement is governed by the following functional:

$$J(C(t)) = \int_0^1 \left(\alpha(s) \left| \frac{\partial C(s,t)}{\partial s} \right|^2 + \beta(s) \left| \frac{\partial^2 C(s,t)}{\partial s^2} \right|^2 \right) ds + \gamma \int_0^1 P(C(s,t)) ds, \tag{3.10}$$

where the first term corresponds to the internal energy and controls the tension and rigidity of the model.† The second term corresponds to the external energy (P represents the potential energy) that could be given as $g(|\nabla I|)$, where g is selected to be a monotonically decreasing function of the gradient of image intensity I. Other examples could include using pressure or balloon representations to represent an expanding object or other diffusing functions. Using calculus of variation techniques, the solution to the equation is obtained by solving the associated Euler–Lagrange PDE:[31,33]

$$\frac{\partial}{\partial s}\left(\alpha\frac{\partial C}{\partial s}\right)+\frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 C}{\partial s^2}\right)+\nabla P(C(s,t))=0. \tag{3.11}$$

However, the formulation in Eq. (3.11) is nonconvex and suffers from several drawbacks such as sensitivity to contour initialization, dependency on parameterization, and an inability to account for topological adaptation (e.g., delineation of a necrotic tumor). To solve the sensitivity problem, the geodesic active contour model was proposed,[35] which in principle is equivalent to Eq. (3.10) if the smoothness constraint is eliminated (i.e., by setting $\beta = 0$). This has led to the development of the flow or curve evolution concept:

$$\frac{\partial C}{\partial t}=\vec{V}(\kappa), \tag{3.12}$$

where $\vec{V}$ is the velocity function (of magnitude V) in the normal direction ($\vec{N}$), and κ is the local contour curvature. However, to resolve the main problem of parameterization and topological adaptation, the level set approach was proposed.[30]

In the level set approach, the curve [in Eq. (3.12)] is embedded in an implicit level set function ϕ. This function defines sets of contour values and positions, including the target boundary at the zero level [$\phi(C) = 0$], as illustrated in Fig. 3.2. In this case, the evolution equation is rewritten as

$$\frac{\partial \phi}{\partial t}=V(\kappa)|\nabla\phi|+F(\Theta), \tag{3.13}$$

where V is defined to be proportional to the curvature and inversely proportional to the image gradient, and $F(\Theta)$ is the external force constraint with vector parameter Θ that could be used to add context-knowledge information such as shape priors. The level set function ϕ is typically selected as a signed distance function. Efficient solutions were developed for Eq. (3.13) by using finite difference and fast marching methods.[30]

† The first-order derivative suppresses stretching and makes the contour behave like an elastic string. The second-order derivative suppresses bending and makes the model behave like a rigid rod.

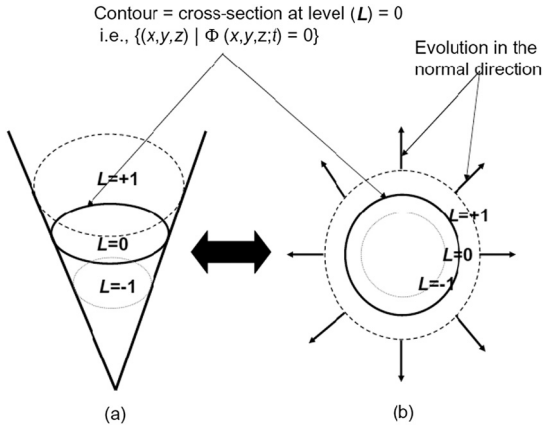

Figure 3.2 Deformable image segmentation by the level set method. (a) Representation of the level set surface at time t of the evolving function ϕ. (b) A projected view showing the evolution direction. Typically, the function ϕ evolves at a velocity proportional to the curvature of the contour and inversely proportional to the image gradient. In the present example, ϕ is represented by a signed Euclidean distance transform of value L. The contour is extracted at $L = 0$, with negative values inside the contour representing the volume of interest and positive values outside representing the background.

3.3.3.1 "Active-contour-without-edge" algorithm

The "active-contour-without-edge" algorithm is one of the frequently used active contour image segmentation algorithms, originally reported by Vese and Chan.[36] Unlike many other algorithms, it is based on regional average intensity value instead of image gradient because gradients tend to be more sensitive to noise and therefore less reliable.

To separate an image into two mutually exclusive partitions, the following system energy equation is to be minimized:

$$F(c_1, c_2, C) = \int_{\Omega_1} (I - c_1)^2 d\Omega_1 + \int_{\Omega_2} (I - c_2)^2 d\Omega_2 + \mu|C| + \nu \cdot \Gamma(C), \quad (3.14)$$

where I is the image intensity; Ω_1 and Ω_2 are the two partitions (Ω_1 is inside the contour defined by the level set, and Ω_2 is outside the contour); c_1 and c_2 are the average image intensity in partition Ω_1 and Ω_2, respectively; C is the partition boundary and $|C|$ is the total length of C for 2D images (or the total surface area of C for 3D images); and ν and μ are user-selected weighting constants. $\Gamma(C)$ is the area inside C for 2D images or the volume inside C for 3D images.

The level set formation for Eq. (3.14) is

$$F(c_1, c_2, C) = \int_{\Omega} \Big[(I - c_1)^2 H(\phi) + (I - c_2)^2 (1 - H(\phi)) + \mu \cdot \delta(\phi)|\nabla\phi| + \nu \cdot H(\phi) \Big] d\Omega, \quad (3.15)$$