Segmentation of Radiological Images

Nigel W. John

4.1 Introduction

Today’s typical hospital environment is well-equipped with medical scanners that routinely provide valuable information to aid with the diagnosis or treatment planning for a particular patient. Computerised tomography (CT), magnetic resonance imaging (MRI), ultrasound, and positron emission tomography (PET) are examples of imaging modalities that are frequently used. An experienced radiologist can gain much insight by viewing the individual images that a scanner produces. However, segmentation of the radiological images to extract or classify a specific region or volume of interest is often required to partition the image data into its constituent components. Image segmentation provides quantitative information about relevant anatomy, for example to determine the size or volume. It also enables an accurate three-dimensional (3D) visualisation of a particular structure using surface triangulation, isosurfacing or volume rendering. There is no single approach that can generally solve the problem of segmentation, and different methods will be more effective depending on the image modality being processed. Pham et al. (2000) and Preim and Bartz (2007) provide detailed reviews of the classic segmentation algorithms, many of which are now implemented within the radiological software supplied by the scanner manufacturer. There are also many excellent software packages freely available in the public domain; however, a software review is outside the scope of this chapter. The interested reader should refer to the Insight Toolkit (ITK), which is an open-source software system that supports a comprehensive collection of the latest segmentation techniques (Yoo 2004). Most of the figures in this chapter have been generated using either the ImageJ (Abramoff et al. 2004) or ITK-SNAP (Yushkevich et al. 2006) software applications.

The remainder of this chapter provides an overview of the most common segmentation techniques currently in use, and highlights their specific advantages and disadvantages.

4.2 Manual Segmentation

The simplest approach for implementing a segmentation tool is to allow for manual tracing or drawing onto the radiological images. Geometric shapes can be used for boundaries, but most approaches record vertices at each mouse click and then draw a line seg-
ment or spline curve between consecutive vertices. Alternatively, the mouse position can be sampled continuously, following the arc of the mouse movements across the screen. The mouse click method is demonstrated in Figure 4.1, which has been created using the open source ITK-SNAP software tool. Each slice on which the structure of interest occurs has to be processed in turn, resulting in a stack of contours. Typically, editing functions are also provided to adjust the position of the contour on a slice. Points on the boundary and inside the contour are considered as belonging to the target structure. Points outside of the contour do not belong to the structure. A surface triangulation algorithm can then be applied to convert the contour stack into a surface mesh that can be visualized in 3D. There are several well-known algorithms in computer graphics that can achieve this task; typically they are based on Delaunay triangulation (Delaunay 1934)\(^1\). Boissonnat (1988), for example, showed how to efficiently compute the Delaunay triangulation between two adjacent contours, two by two, and using only 2D operations. Other segmentation methods described below also produce a contour stack and will require such a surface triangulation technique (see Fig. 4.5d).

\(^1\) Delaunay triangulation for a set \(P\) of points in the plane is the triangulation \(DT(P)\) of \(P\) such that no point in \(P\) is inside the circumcircle of any triangle in \(DT(P)\). Delaunay triangulations maximise the minimum angle of all the angles of the triangles in the triangulation.

Manual segmentation is time-consuming and prone to the inaccuracies introduced by the end user’s skill with the mouse. Care must also be taken when there are contours within contours, defining “holes” in the target structure.

### 4.3 Thresholding

Another straightforward but fast image segmentation method is thresholding. Individual pixels in the image are marked as target pixels if their value is greater than some threshold value; otherwise they are marked as background pixels. A tolerance value is typically used so that values close to the selected threshold will also be selected. The resulting segmentation is a binary image with all target pixels given a value of 1, and all other (background) pixels given a value of 0.

A common technique for choosing a threshold value is to select a local minimum from a histogram of the image pixel intensities. Figure 4.2 is an example of the results that can be obtained; in this case the isodata algorithm (Ridler and Calvard 1978) has been used to determine a threshold. The problem is that the boundary between two tissue types often overlaps, and is often referred to as the partial volume effect. In such cases a local minimum may not be detectable in the histogram. More sophisticated techniques have been developed; for example, Otsu (1979) maximizes the separation between different threshold classes in the data, based on an initial guess of the thresholds.

Thresholding is often combined with a connected component analysis (CCA) step. The binary image is scanned pixel-by-pixel from top to bottom and left to right to identify and label connected pixel regions. Once a tissue type has been labelled, the process can be repeated by generating a new binary image using a threshold value for another tissue type.

In 3D, an isosurface algorithm can be applied to the raw image data, or preferably to segmented data. Using a particular threshold value, a surface can be extracted using, for example, the well-known marching cubes algorithm (Lorensen and Cline 1987). Isosurfacing is usually carried out after one of the more sophisticated segmentation techniques described below has been applied.