

Chapter 2

Basic Terms and Notation

Contents

2.1	Images, Segmentations, and Surface Meshes	18
2.1.1	Three-dimensional Medical Images	18
2.1.2	Segmentations of Three-dimensional Medical Images	18
2.1.3	Triangle Surface Meshes	19
2.1.4	From Segmentations to Surface Meshes and Back	21
2.2	Deformable Surface Meshes	21
2.2.1	Displacement Fields and Sets of Candidate Displacements	22
2.2.2	Appearance Cost	22

This chapter introduces basic terms and notations which we use throughout the following chapters.

Lists $A := \{a_i\}_{i=1}^{n_A}$ appear in different contexts throughout this thesis. We denote the index set of a list A as $N_A := \{1, \dots, n_A\}$. We shortly denote that an element a appears in a list, i.e. $\exists i \in N_A : a = a_i$, as $a \in A$. We denote list B appended to list A as $A \cup B$. Whenever it adds to clarity of notation, we denote a function f whose domain is the index set N_A sloppily also as a function whose “domain” is A .

For a finite list $A := \{a_i\}_{i=1}^{n_A}$ and a function $y : N_A \rightarrow \mathbf{R}$, in case y is not guaranteed to have a unique minimizer, with $i := \min \{\operatorname{argmin}_{j \in N_A} \{y(j)\}\}$ we denote the respective element of the list, a_i , sloppily as $\operatorname{argmini}_{a \in A} \{y(a)\}$. We define $\operatorname{argmaxi}$ analogously. This definition is convenient in case an objective y is not guaranteed to have a unique minimizer, yet we need to decide for one single minimizer without having a sophisticated criterion for this decision at hand. In this case $\operatorname{argmini}$ helps us denote one minimizer distinguished by means of a straightforward criterion,

namely the minimizer with minimal index in the list.

2.1 Images, Segmentations, and Surface Meshes

2.1.1 Three-dimensional Medical Images

Three-dimensional medical images are *digital* images (see e.g. Handels (2009)). Intensities are acquired only at a finite set of locations $X \subset \mathbf{R}^3$. A location $x \in X$ together with an intensity $I(x) \in \mathbf{R}$ as acquired for this location is called *voxel* (see e.g. Smith (1995)). Locations X are organized as a regular grid. One commonly used grid is

$$X = \left\{ \mathbf{x} \in \mathbf{R}^3 : \forall c \in \{1, 2, 3\} \exists i \in \{0, \dots, n_c - 1\} : x_c = x_c^{(0)} + \delta_c \cdot i \right\}, \quad (2.1)$$

where $\delta_c \in \mathbf{R}^+$ denotes the distance between locations for which intensities are acquired in direction of coordinate c , and $n_c \in \mathbf{N}^+$ denotes the number of such locations. The location $\mathbf{x}^{(0)} \in \mathbf{R}^3$ is the *origin* of the image. Due to the details of image acquisition, there is often one distinguished plane with isotropic resolution, i.e. $\delta_i = \delta_j$ for $i, j \in \{1, 2, 3\}, i \neq j$, while resolution is lower in direction orthogonal to this plane, i.e. $\delta_k \neq \delta_i$ for $k \in \{1, 2, 3\} \setminus \{i, j\}$. In consequence, grids are in general anisotropic.

The convex hull of X yields an interval $\Omega \subset \mathbf{R}^3$. Image intensities for locations $y \in \Omega \setminus X$ can be derived by means of *interpolation*. One widely used interpolation method assigns to y a convex combination of intensities as given at the eight closest locations contained in X , computed via subsequent linear interpolations in each coordinate direction. This approach is called *trilinear interpolation*. Another approach assigns to y the intensity given at the closest location $x \in X$. This is referred to as *nearest neighbor interpolation*. For more details on grids and interpolation see e.g. Modersitzki (2004).

A digital image together with an interpolation method yields a function

$$I : \Omega \rightarrow \mathbf{R}.$$

In the following such a function I derived from a grid (2.1) and trilinear interpolation is our mathematical model for a three-dimensional (3d) medical image. It assigns intensities (gray-values) to an interval $\Omega \subset \mathbf{R}^3$ that contains (parts of) the space that a human body occupies within some reference coordinate system.

2.1.2 Segmentations of Three-dimensional Medical Images

A *segmentation* of an image is a finite partition of the image. It assigns a region-ID or *label* to each location in the image domain, $\Omega \rightarrow \{0, 1, \dots, n\}$, with n the number of labels. The “classical” definition of a segmentation requires locations

equipped with the same label to be *similar* in terms of image characteristics (see e.g. Pham et al. (2000)). This definition is suitable to characterize image partitions as generated by automatic, low-level segmentation methods. In this thesis, we use the term “segmentation” to also refer to image partitions that do not necessarily comply with this definition. Particularly, we call a partition “segmentation” if its intended purpose is to assign the same label to locations that share *semantic information* such as *belonging* to the same organ or other anatomical structure (see e.g. Heimann (2003)).

A *binary* segmentation partitions the image domain into two subsets called foreground (0) and background (1): $\Omega \rightarrow \{0, 1\}$. A segmentation of a voxel image that assigns a unique label to each voxel is called *hard segmentation* (see e.g. Pham et al. (2000)). As opposed to *soft* or *probabilistic* segmentations, hard segmentations do not allow for sub-voxel accuracy of segmentations, nor do they capture information about segmentation uncertainty. Although sub-voxel accuracy and segmentation uncertainty are highly interesting and valuable concepts, this thesis does not pursue this path of research, but sticks to conventional, hard segmentations. In the following, we use the terms *hard segmentation*, *voxel segmentation* and *segmentation* synonymously.

A hard segmentation implies a voxel labeling $X \rightarrow \{0, 1, \dots, n_R\}$. A labeling of Ω is commonly derived from a voxel labeling by means of nearest neighbor interpolation. In this sense, a *segmentation of an anatomical structure in a voxel image* is a binary partition that assigns “foreground” (label 1) to each voxel of which $> 50\%$ volume belong to the anatomical structure, and “background” (label 0) to all others. A segmentation of multiple anatomical structures in a voxel image assigns label l to each voxel of which $> 50\%$ volume belong to structure l , and label 0 to all others.

Note that the term *segmentation* refers to both the task and the result of partitioning, i.e. *segmenting*, an image.

2.1.3 Triangle Surface Meshes

Following Schneider and Eberly (2003, Chapter 9.3), we call *triangle mesh* a finite collection of *vertices*, *edges*, and *triangles* that satisfies certain conditions as described in the following. A vertex is a point in space, $v \in \mathbf{R}^3$. We refer to the list of vertices of a triangle mesh as $V := \{v_i \in \mathbf{R}^3\}_{i=1}^{n_V}$, where n_V is the number of vertices. We refer to the index set of V as $N_V := \{1, \dots, n_V\}$. We call the concatenation of all vertices of a mesh *shape vector*, $\mathbf{v} := (v_1^T, \dots, v_{n_V}^T)^T \in \mathbf{R}^{3n_V}$. An edge is a line segment that connects two different vertices. An edge is identified by a tuple formed by the respective vertices’ indices, (j, k) , $j, k \in N_V$, $j \neq k$. We refer to the list of edges $E \subset N_V \times N_V$ as $E := \{(j, k)_i\}_{i=1}^{n_E}$, where n_E is the number of edges. A triangle is the convex hull of three different vertices. A triangle is identified by a

triple formed by the respective vertices' indices, $(j, k, l), j, k, l \in N_V, j \neq k \neq l \neq j$. We refer to the list of triangles (*faces*) $F \subset N_V \times N_V \times N_V$ as $F := \{(j, k, l)_i\}_{i=1}^{n_F}$, where n_F is the number of triangles.

The conditions that the collection (V, E, F) has to satisfy to be a *mesh* are that each vertex belongs to at least one edge, and each edge belongs to at least one triangle. If any two triangles of a mesh are connected by a path that leads from triangle to triangle over *shared edges* (i.e. edges that belong to multiple triangles), the mesh is called *connected*. A connected mesh is called *manifold* if each edge is shared by at most two triangles. A manifold mesh is called *orientable* if triangle triples can be ordered such that each edge (j, k) of the mesh that is shared by two triangles appears in order j, k in one triple and in reverse order, k, j , in the other triple. Informally speaking this implies that the mesh has “inside” and “outside”. A mesh is called *closed* if any edge is shared by exactly two triangles. Closedness implies the mesh to be manifold.

If not noted otherwise, in the following we use the term *triangle surface mesh* or just *surface mesh* to refer to a closed orientable mesh $M := (V, E, F)$. Note that we deviate from Schneider and Eberly (2003) only by not considering *self-intersections* of meshes in the above definitions. A mesh has self-intersections if it has faces, edges or vertices that interpenetrate each other. Self-intersecting meshes appear throughout this thesis – we call them *meshes* for ease of terminology, and discuss the impact of self-intersections if it is important in a certain situation.

Any point that *lies on a surface mesh* $M := (V, E, F)$ is defined by means of a triangle $t := (j, k, l) \in F$ it lies on and by its *barycentric coordinates* (α, β, γ) on this triangle (see e.g. Coxeter (1969)). Its location $x \in \mathbf{R}^3$ is

$$x = \alpha v_j + \beta v_k + \gamma v_l \text{ with } \alpha, \beta, \gamma \in [0, 1], \alpha + \beta + \gamma = 1 .$$

In other words, as $\gamma = 1 - \alpha - \beta$, every point on M is described by a triple (t, α, β) with $t \in F$ and $\alpha, \beta \in [0, 1]$. In reverse every such triple describes a point on M . In case of self-intersections of the mesh, there are point positions $x \in \mathbf{R}^3$ on M that are described by multiple triples, yet one triple always describes exactly one point. For a mesh $M := (V, E, F)$ we denote

$$\mathcal{F} := \{(t, \alpha, \beta) : t \in F, \alpha, \beta \in [0, 1]\} .$$

We denote the respective set of point position in $\mathbf{R}^3$ as

$$\mathfrak{M} := \{\alpha v_j + \beta v_k + \gamma v_l : ((j, k, l), \alpha, \beta) \in \mathcal{F}, \gamma := 1 - \alpha - \beta\} .$$

There is a bijection between $\mathcal{F}$ and $\mathfrak{M}$ if and only if the respective mesh M is free of self-intersections.

The surface normal at vertex v of a surface mesh, denoted as n_v , can be estimated via the set of edges of the mesh that contain the vertex. Similarly, the principal

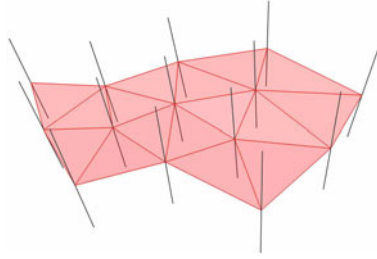


Figure 2.1 Detail of a triangular surface mesh (red/gray). Vertex normals indicated by black line segments.

curvatures at a vertex can be estimated from a vertex neighborhood (Hildebrandt et al., 2005). Figure 2.1 shows a detail of a triangle mesh.

2.1.4 From Segmentations to Surface Meshes and Back

A binary label image can be converted into a surface mesh that represents the boundary of the volume labeled “foreground” by means of the *Marching Cubes* or *Generalized Marching Cubes* algorithms (Lorensen and Cline, 1987; Hege et al., 1997). In reverse, a surface mesh can be converted into a binary label image by means of *Scan Conversion* (Kaufman, 1987). With the same algorithms, multi-label images can be converted into meshes (which are in general non-manifold), and vice-versa.

2.2 Deformable Surface Meshes

Deformable surface meshes are surface meshes which are deformed by means of vertex re-locations with the goal of yielding a geometric representation of a target structure sought in an image. For each vertex of a deformable surface mesh, a set of *candidate locations*, i.e. potential new vertex positions, are tested for *appearance match*. This is done by comparing actual image appearance at candidate locations to an *appearance model*. Therefore, at each candidate location, image information is assessed in a certain neighborhood, as required for comparison with the particular appearance model. Given this analysis of appearance match, a new shape is computed by *displacing* the vertices of the mesh to suitable locations, following a trade-off between appearance match and anatomically plausible deformation.

2.2.1 Displacement Fields and Sets of Candidate Displacements

A surface mesh is *deformed* by moving each of its vertices to a new position, i.e. by *displacing* each vertex. A *displacement field* assigns a *displacement* $s \in S$ to each vertex of a mesh, where $S \subset \mathbf{R}^3$ is a set of *candidate displacements*. Assigning a displacement to each vertex of a mesh yields a list of displacements $\{d_i \in S\}_{i=1}^{nv}$. If it adds to clarity of notation, we also sloppily denote a displacement field as a “function” on the list of vertices V , $d : V \rightarrow S, v_i \mapsto d(v_i) := d_i$. We denote an “overall” mesh displacement $(d_1^T, \dots, d_{nv}^T) \in \mathbf{R}^{3nv}$ as $d\mathbf{v}$. We refer to the set $v + S$ as *set of candidate locations* of v . Sets of candidate displacements can also be defined as vertex-individual sets: In this case we denote the set of candidate displacements for vertex v_i as $L_i \subset \mathbf{R}^3$, or, if it adds to clarity, also as $L(v_i)$.

Whenever we denote sets of candidate displacements in contexts that do not require an explicit specification as to whether we refer to global sets S or vertex-individual sets L_i , we denote sets of candidate displacements as S for ease of notation.

Discrete sets of candidate displacements are also denoted as *lists*, $S := \{s_i\}_{i=1}^n$, where n is the number of candidate displacements. For a discrete set of candidate displacements, we denote the minimum Euclidean distance between unequal displacements $s_i, s_j \in S$ as *sampling distance* $\delta_S := \min_{s_i \neq s_j} \|s_i - s_j\|$. The sampling distance δ_{L_i} of a vertex-individual set L_i is defined analogously. In case δ_{L_i} is equal for all $v_i \in V$, we refer to it as δ_L . Discrete sets of candidate displacements yield discrete sets of candidate locations per vertex, $v + S$. In this case we refer to a candidate location $v + s \in v + S$ also as *sample point*.

A displacement field induces a deformation of all points on a mesh $M := (V, E, F)$ by means of their barycentric coordinates. This deformation can be described as a well-defined function on $\mathcal{F}$, namely $\text{id}_{\mathcal{F}}$. For an intersection-free mesh M , we can denote a mesh deformation as a well-defined function m on $\mathfrak{M}$: With $\hat{v}_i := v_i + d_i$ denoting the displaced vertex positions, $\hat{V} := \{\hat{v}_i\}$, and $\hat{M} := (\hat{V}, E, F)$,

$$m : \mathfrak{M} \rightarrow \hat{\mathfrak{M}}, \alpha v_j + \beta v_k + \gamma v_l \mapsto \alpha \hat{v}_j + \beta \hat{v}_k + \gamma \hat{v}_l.$$

2.2.2 Appearance Cost

We refer to a function $\phi : V \times S \rightarrow \mathbf{R}_0^+$ that assigns scalar *appearance costs* to pairs of a vertex and a candidate displacement as *appearance cost function*. Appearance costs reflect the dissimilarity between actual image appearance in the vicinity of candidate locations on the one hand, and expected intensities as captured by an appearance model on the other hand. For vertex-individual sets of candidate displacements L_i the respective cost function is defined as $\phi : \{(v_i, l) : v_i \in V, l \in L_i\} \rightarrow \mathbf{R}_0^+$. In case only the “current” vertex positions of a mesh are of interest, i.e. $S = \{\mathbf{0}\}$, we denote appearance cost functions as $\phi : V \rightarrow \mathbf{R}_0^+$.

A *global* appearance cost function $\Phi : \mathbf{R}^{3n_V} \rightarrow \mathbf{R}_0^+$ measures the dissimilarity between actual image appearance in the vicinity of a surface mesh depending on its shape vector $\mathbf{v} \in \mathbf{R}^{3n_V}$ and expected intensities as captured by an appearance model. In this thesis we employ global appearance cost functions which are sums of individual vertex-wise costs (cf. e.g. Sec. 3.3, and 4.2.2). Formally, this means $\Phi(\mathbf{v}) := \sum_{v \in V} \phi(v)$. This is a simple yet common definition (see e.g. Cootes et al. (1995); Khoshelham (2007); Yin et al. (2010)). While global appearance cost functions may be defined differently, e.g. by means of mutual information as is popular for multi-modal image registration (see e.g. Modersitzki (2004)), a respective discussion lies out of the scope of this thesis.

Deformable Meshes for Medical Image Segmentation
Accurate Automatic Segmentation of Anatomical
Structures

Kainmueller, D.

2015, XVIII, 180 p. 52 illus., 30 illus. in color., Softcover

ISBN: 978-3-658-07014-4