Chapter 1
Interchanging geometry conventions in 3DEM: mathematical context for the development of standards

C.O.S. Sorzano, R. Marabini, J. Vargas, J. Otón, J. Cuenca-Alba
A. Quintana, J.M. de la Rosa-Trevín, J.M. Carazo

Abstract The specification of the information on the three-dimensional orientation of an image with respect to a given coordinate system is at the heart of our ability to reconstruct a three-dimensional object from sets of its two-dimensional projection images. Transferring this information from one package to another is important to structural biologists wanting to get the best from each software suite. In this chapter, we review in depth the main considerations and implications associated with the unambiguous specification of geometrical specifications, in this way paving the way to the future specifications of standards in the field of three-dimensional electron microscopy. This is the case of EMX in which affine transformations have been adopted as the means to communicate geometrical information.

1.1 Introduction

Transformation matrices are normally used by Three-Dimensional Electron Microscopy (3DEM) analysis programs to describe the relative spatial relationship of

C.O.S. Sorzano, J. Vargas, J. Otón, J. Cuenca-Alba, A. Quintana, J.M. de la Rosa-Trevín, J.M. Carazo
Biocomputing Unit, National Center for Biotechnology (CSIC), c/Darwin, 3, Campus Universidad Autónoma, 28049 Cantoblanco, Madrid, Spain.

C.O.S. Sorzano
Bioengineering Lab., Universidad CEU San Pablo, Campus Urb. Montepríncipe s/n, 28668 Boadilla del Monte, Madrid, Spain.

R. Marabini
Escuela Politécnica Superior, Universidad Autónoma de Madrid, Campus Universidad Autónoma, 28049 Cantoblanco, Madrid, Spain.
e-mail: coss@cnb.csic.es

the set of experimental projections obtained by the microscope or the relative spatial relationship of a set of volumes.

There are multiple ways to specify orientations (and, more broadly, general geometrical transformations) that are in use in the three-dimensional electron microscopy field. Indeed, because of the importance of this angular information and the diversity of ways it is presented, the interchange of angular information among different image processing suites is complicated, leading to serious interoperability issues. It is in this context in which we will review in the following sections the main issues associated with the unambiguous specification of geometrical properties, so that they could serve as the basis for future definitions of a standard in the field.

Recently, the EMX format (Electron Microscopy eXchange, http://i2pc.cnb.csic.es/emx) has been put forward as a way to interchange data and metadata information among different 3DEM software packages. EMX considers the interchange of geometrical information through an affine transformation matrix. This approach allows communicating rotations, translations, mirrors, scalings and shearings through a homogeneous matrix $\tilde{A}$ (Jain, 1989). The use of a general affine matrix was already introduced in X-ray crystallography by Rossmann and Blow (1962).

The affine matrix representation has a number of advantages over other ways of representing geometrical transformations:

- Transformation versatility: Euler angles are frequently used within the 3DEM field to represent rotations, quaternions to a lesser extent. However, Euler angles and quaternions can only be used to communicate rotations while affine transformations can also represent mirrors (used by 2D Classification algorithms such as ML2D (Scheres et al, 2005) or CL2D (Sorzano et al, 2010)), shrink and shearing (used in Electron Tomography (Cantele et al, 2007)), and different scales (needed to represent small differences in magnification). All these transformations can be represented in a unified framework through the affine transformation. In fact, affine transformations are already used by IMOD (Kremer et al, 1996) to stitch together different tomograms.
- Intuitiveness: The possibility of interpreting the affine transformation matrix as a change of basis between two coordinate systems (see Section 1.7) provides a rather intuitive meaning to this representation. This intuition is not so clear for the Euler angles or quaternions, even less when common geometrical operations are performed at the projection level. Additionally, linear algebra and its matrix representation is well known by 3DEM practitioners. This also makes the affine transformation representation more accessible.
- Mathematical properties: Affine transformations matrices are unique (Euler angles, for example, are not) and they do not suffer from the Gimbal Lock problem (encountered in some Euler angle definitions as will be pointed out further along this chapter).

In this chapter we provide an example of all the elements needed to define a standard for the interchange of geometrical information and discuss its meaning and implications in real and Fourier space. We propose that defining a standard amounts to:

- Defining a common nomenclature so that the terminology used in the standard is well understood.
- Defining a coordinate system so that the origin and relative position of the axes are well defined.
- Defining a way of encoding geometrical transformations.
- Defining the way in which the geometrical transformations are applied to volumes and images.
- Defining a way of composing several geometrical transformations.

In this chapter, we review all these points and discuss about the meaning of the geometrical transformations defined. In the appendixes we review:

- the most common transformations needed in 3DEM (translations, mirrors, shears, rotations, and scalings).
- the way in which Euler angles are used to define rotations in volumes and projection orientations.

1.2 Standard nomenclature

Before going further, it is good to agree on a number of standard geometrical concepts. The transformations defined in this chapter are referred to as 3D points, which are normally written in Cartesian coordinates $(x,y,z) \in \mathbb{R}^3$. Note that we distinguish between a 3D point and a 3D vector . Both can be written as a collection of three numbers (x,y,z). However, a 3D point belongs to an affine space, while a 3D vector belongs to a vector space. In fact, an affine space is composed by a set A (in our case $A = \mathbb{R}^3$), a vector space V (in our case $V = \mathbb{R}^3$), and a faithful and transitive group action of V on A. Conceptually, this means that we have points (in the affine space) and vectors (in the vector space), and that we know how to add a vector to a point to produce a new point. The subtraction of two points produces a vector.

The geometrical transformations that we define below are applied on points. A point is used to define the location of a given Coulomb potential within a macromolecular structure. The function $V(\mathbf{r}) : \mathbb{R}^3 \to \mathbb{R}$ is used to represent the Coulomb potential, the field acting on the electrons forming the image in an electron microscope, at a point $\mathbf{r}$ in space. The estimate of this function is normally referred to as a volume or a map. Similarly, we define an image as a function $I(\mathbf{s}) : \mathbb{R}^2 \to \mathbb{R}$.

While transforming points, it is customary to express the point in its homogeneous coordinates , which are attained simply by adding a 1 to the end of the list of coordinate values. We will refer to this point with $\tilde{\mathbf{r}} = (x,y,z,1)$ to distinguish it from its non-homogeneous expression $\mathbf{r} = (x,y,z)$. Note that $\tilde{\mathbf{r}} \in \mathbb{R}^3 \times \{1\}$. Depending on the context we may use a point $\mathbf{r}$ or its homogeneous coordinate $\tilde{\mathbf{r}}$ as the argument of a volume. In both cases the intensity value associated to that point is the same.

1.3 A standard coordinate system

Agreeing on geometrical issues must start with agreeing on the coordinate system that defines the context in which the different geometrical elements are expressed. Heymann et al (2005) established a right-handed coordinate system as depicted in Fig. 1.1. A right-handed coordinate system is characterized by the following relationships among the vectors defining the coordinate system

$$\begin{aligned} X \times Y &= Z \\ Y \times Z &= X \\ Z \times X &= Y. \end{aligned} \tag{1.1}$$

This coordinate system is called right-handed because when considering any cross-product of the form $A \times B = C$, if we make the fingers of the right hand to turn from A to B (finger ends pointing to B), then the right thumb is pointing in the direction of C.

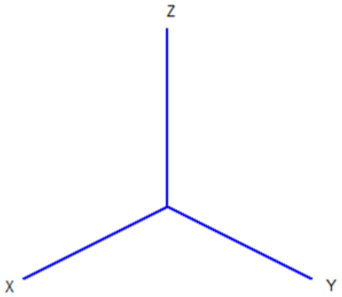

Fig. 1.1 Right-handed coordinate system established as a convention in Heymann et al (2005). Z is vertical, in the paper plane, and X and Y stand out of the paper, toward the reader.

In 3DEM, it is sometimes useful to depict the coordinate system attached to the electron microscope (see Fig. 1.2). The origin of the coordinate system is forced to coincide with the center of the macromolecule being reconstructed, which is supposed to be at the level of the sample holder. Electrons travel from negative Z to positive Z (Z is aligned with the microscope column). Y is pointing towards the microscopist and X is pointing to the right of the microscopist. Note that an observer within the coordinate system cannot see any difference between the situation depicted in Fig. 1.1 and in Fig. 1.2. Depending on the operation to be performed it might be more intuitive to work with one or the other depiction.

Let us consider a volume of size $N_x \times N_y \times N_z$ voxels ($N_x \times N_y$ for images). Programming languages normally assign indexes that go from 0 (C, C++, Python, Java) to $N_i - 1$ ($i = x,y,z$; see Fig. 1.3), or from 1 (Fortran, MATLAB) to N_i. In the fol-

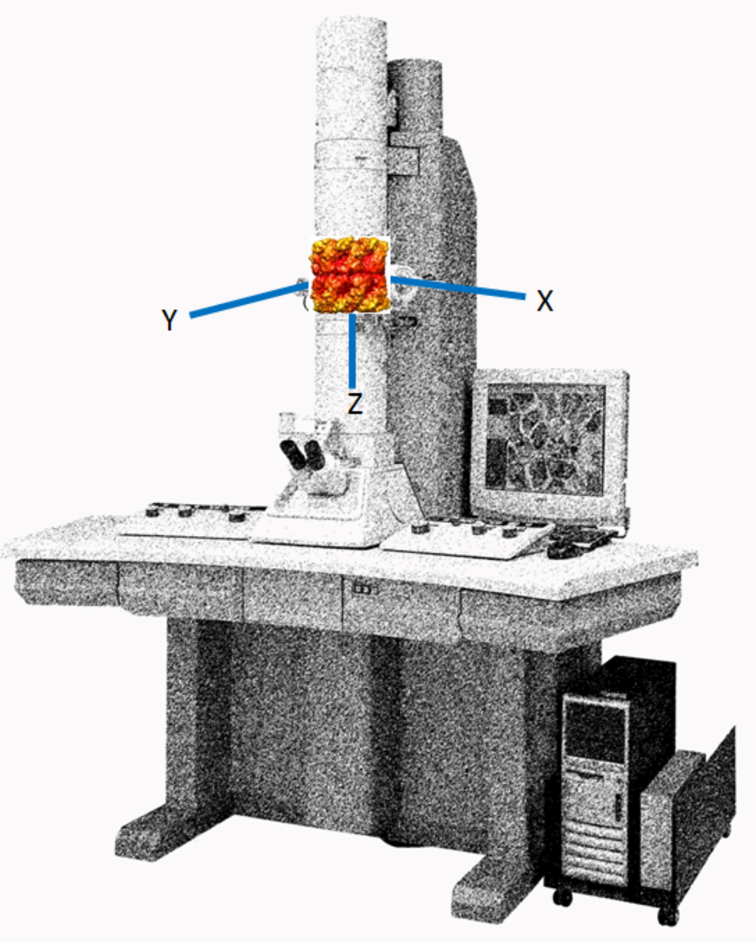

Fig. 1.2 Same coordinate system of Fig. 1.1 in a different orientation for visualization.

lowing, we will use the C indexing scheme. However, to define geometrical transformations it is normally preferred to set the origin of the coordinate system in the middle of the volume rather than in one corner. To fully define a standard we need to specify where the origin is within a volume. A possibility is to set it at the center of the voxel whose index is $(\lfloor \frac{N_x}{2} \rfloor, \lfloor \frac{N_y}{2} \rfloor, \lfloor \frac{N_z}{2} \rfloor)$, where $\lfloor x \rfloor$ is the rounding-down operator (see Fig. 1.3). Now the pixel coordinates at the center of the pixel range from $-\lfloor \frac{N_i}{2} \rfloor$ to $\lfloor \frac{N_i}{2} \rfloor$ if N_i is odd, or to $\lfloor \frac{N_i}{2} \rfloor - 1$ if N_i is even. All transformations described below are using the so-called logical coordinates.

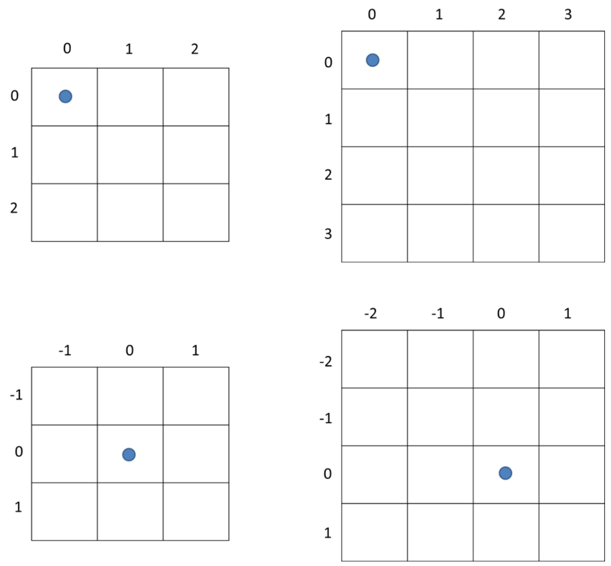

Fig. 1.3 Top row: coordinates of the pixel center for 3×3 and 4×4 images when the image origin is in one corner of the image (physical coordinates). Bottom row: coordinates of the pixel center of the same images when the image origin is in the middle of the image (logical coordinates).

1.4 Standard definition of geometrical transformations

Geometrical transformations can be represented by matrix operations between homogeneous coordinates:

$$\tilde{\mathbf{r}}_{\tilde{A}} = \tilde{A}\tilde{\mathbf{r}}, \tag{1.2}$$

where $\tilde{\mathbf{r}} \in \mathbb{R}^3 \times \{1\}$ is the homogeneous coordinate of the point to transform, $\tilde{\mathbf{r}}_{\tilde{A}} \in \mathbb{R}^3 \times \{1\}$ is its transformed point in homogeneous coordinates, and $\tilde{A}$ is a 4×4 invertible, real matrix of the form

$$\tilde{A} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}. \tag{1.3}$$

We use the tilde on the matrix A to remind us that it operates on homogeneous coordinates. Matrix $\tilde{A}$ comprises translations (given by the vector $\mathbf{t}$) as well as rotations, mirrors, shearing, and scaling (encoded in the matrix R, see Fig. 1.4 for an example of these transformations applied to a unit cube). The effect of Eq. (1.3) on any point is

$$\mathbf{r}_{\tilde{A}} = R\mathbf{r} + \mathbf{t}, \tag{1.4}$$

that is, a linear transformation of its coordinates plus a translation.

Altogether, matrix $\tilde{A}$ is called an affine transformation and it has the property that it preserves straight lines (if a set of points lying on a straight line are transformed, then the transformed points also lie on a straight line), ratios of distances of points lying on a straight line (e.g., the transformation of the mid-point between two points is also the mid-point between the two transformed points), and parallelism of straight lines (if two lines are parallel, the transformed lines are also parallel). It does not preserve segment lengths and the angle between two adjacent segments.

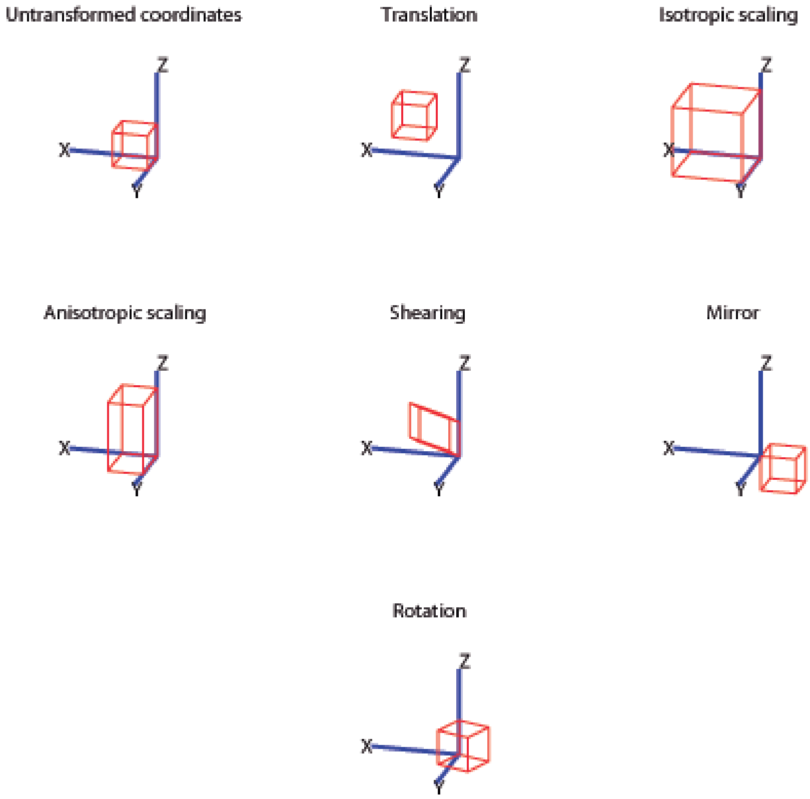

Fig. 1.4 Sample transformations applied to a unit cube.

Appendix 1 provides a detailed description of the different options to represent geometrical transformations. Given the importance of rotations and Euler angles for the 3DEM community, they are given a special treatment in the Appendix.

1.5 Standard use of the geometrical transformation

Encoding orientational information (through the full transformation matrix, Eq. (1.3)) is not enough; the way of using this information must also be agreed on. A standard option is to define a transformed volume as

$$V_{\tilde{A}}(\tilde{\mathbf{r}}) = V(\tilde{A}^{-1}\tilde{\mathbf{r}}), \tag{1.5}$$

where $\tilde{\mathbf{r}} \in \mathbb{R}^3 \times \{1\}$ represents a homogeneous 3D coordinate (i.e., $\tilde{\mathbf{r}} = (x,y,z,1)^T$), $V(\tilde{\mathbf{r}}) : \mathbb{R}^3 \times \{1\} \to \mathbb{R}$ is a volume, and $V_{\tilde{A}}(\tilde{\mathbf{r}})$ its transformed version after applying the transformation defined by $\tilde{A}$. Note that we have used discrete coordinates in $\mathbf{r}$, i.e., their units are pixels as defined in Fig. 1.3. If we want to express $\mathbf{r}$ coordinates in continuous coordinates (whose units are Angstroms), then we only need to substitute in all equations $\mathbf{r}$ by $\frac{1}{T_s}\mathbf{r}$, where T_s is the sampling rate in Angstroms per pixel.

In 3DEM, we must analogously define the projection according to the orientation encoded by the matrix $\tilde{A}$. Let $\tilde{\mathbf{s}}$ be the homogeneous coordinate of a pixel location, and $I_{\tilde{A}}(\tilde{\mathbf{s}}) : \mathbb{R}^2 \times \{1\} \to \mathbb{R}$ be the projection associated with it defined by

$$I_{\tilde{A}}(\tilde{\mathbf{s}}) = \int_{-\infty}^{\infty} V_{\tilde{A}}(\tilde{H}^T\tilde{\mathbf{s}})dt \tag{1.6}$$

where

$$\tilde{H}^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & t \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.7}$$

Note that t is a dummy integration variable . As shown below this matrix definition of the projection has operative advantages.

Although Eqs. (1.3), (1.5) and (1.6) are enough to fully specify the transfer of geometrical information, in the following sections we explictly discuss three of the most common usages.

1.5.1 Alignment of volumes

Let us assume we want to exchange a set of N volumes so that all of them are aligned to a common reference $V_{ref}(\tilde{\mathbf{r}})$. This information can be communicated by transferring the matrices $\tilde{A}_i$ such that each volume $V_i(\tilde{\mathbf{r}})$ can be transformed as $V_{\tilde{A}_i}(\tilde{\mathbf{r}}) = V_i(\tilde{A}_i^{-1}\tilde{\mathbf{r}})$ and $V_{\tilde{A}_i}(\tilde{\mathbf{r}})$ is aligned with the reference $V_{ref}(\tilde{\mathbf{r}})$. All values of $\tilde{\mathbf{A}}$ in Eq. (1.3) have to be specified. The average aligned volume can be computed as

$$V_{avg}(\tilde{\mathbf{r}}) = \frac{1}{N}\sum_{i=1}^{N} V_{\tilde{A}_i}(\tilde{\mathbf{r}}). \tag{1.8}$$

1.5.2 2D Alignment of images

The problem to solve in this case is the exchange of angular information on sets of images so that they can be brought together to the same reference system. The affine transformation can be simplified to

$$\tilde{A} = \begin{pmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix}. \tag{1.9}$$

Let us assume we want to exchange a set of N images so that all of them are aligned to a common reference $I_{ref}(\tilde{\mathbf{s}})$. We need to communicate the matrices $\tilde{A}_i$ such that each image $I_i(\tilde{\mathbf{s}})$ can be transformed as $I_{\tilde{A}_i}(\tilde{\mathbf{s}}) = I_i(\tilde{A}_i^{-1}\tilde{\mathbf{s}})$ and $I_{\tilde{A}_i}(\tilde{\mathbf{s}})$ is aligned with the reference $I_{ref}(\tilde{\mathbf{s}})$. All values in Eq. (1.9) have to be specified. The average aligned image can be computed as

$$I_{avg}(\tilde{\mathbf{s}}) = \frac{1}{N}\sum_{i=1}^{N} I_{\tilde{A}_i}(\tilde{\mathbf{s}}). \tag{1.10}$$

1.5.3 Alignment of image relative to volume

In this case, the task is to communicate the relative orientation of a set of images given a volume. Given a reference volume $V_{ref}(\tilde{\mathbf{r}})$ and a set of N projection images $I_i(\tilde{\mathbf{s}})$, we need to communicate the transformation matrices $\tilde{A}_i$ such that

$$I_i(\tilde{\mathbf{s}}) = \int_{-\infty}^{\infty} V_{ref,\tilde{A}_i}(\tilde{H}^T\tilde{\mathbf{s}})dt = \int_{-\infty}^{\infty} V_{ref}(\tilde{A}_i^{-1}\tilde{H}^T\tilde{\mathbf{s}})dt. \tag{1.11}$$

All values in Eq. (1.3) have to be specified.

1.6 Specifying a sequence of transformations

1.6.1 Volume transformations

Let us presume that we apply a transformation matrix $\tilde{A}_1$ to a volume $V(\tilde{\mathbf{r}})$ in order to produce a transformed volume $V_{\tilde{A}_1}(\tilde{\mathbf{r}})$ as $V_{\tilde{A}_1}(\tilde{\mathbf{r}}) = V(\tilde{A}_1^{-1}\tilde{\mathbf{r}})$. Then, we further transform the new volume with a second transformation matrix $\tilde{A}_2$ as $V_{\tilde{A}_2}(\tilde{\mathbf{r}}) = V_{\tilde{A}_1}(\tilde{A}_2^{-1}\tilde{\mathbf{r}})$. We can combine the two transformations in a single matrix by considering that

$$V_{\tilde{A}_2}(\tilde{\mathbf{r}}) = V_{\tilde{A}_1}(\tilde{A}_2^{-1}\tilde{\mathbf{r}}) = V(\tilde{A}_1^{-1}\tilde{A}_2^{-1}\tilde{\mathbf{r}}) = V((\tilde{A}_2\tilde{A}_1)^{-1}\tilde{\mathbf{r}}). \tag{1.12}$$

In this way, the overall transformation is given by $\tilde{A} = \tilde{A}_2\tilde{A}_1$.

It must be noted that we can decompose the matrix $\tilde{A}$ as the multiplication of other two matrices:

$$\tilde{A} = \begin{pmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} = \begin{pmatrix} I & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} R & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} = \tilde{T}\tilde{R}, \tag{1.13}$$

where $\tilde{T}$ represents a pure translation and $\tilde{R}$ a combination of rotations, scalings, mirrors and shearings. Then, $\tilde{A}$ implies first the application of $\tilde{R}$ and then a translation. Note that matrix multiplication is not commutative and, therefore, $\tilde{T}\tilde{R} \neq \tilde{R}\tilde{T}$.

1.7 Meaning of the geometrical transformation

1.7.1 In real space

Given a transformation matrix $\tilde{A}$ and a point in 3D space $\tilde{\mathbf{r}}$, we can find its location after transforming it as $\tilde{\mathbf{r}}_{\tilde{A}} = \tilde{A}\tilde{\mathbf{r}}$. In the same way, we could transform a whole object $V_{\tilde{A}}(\tilde{\mathbf{r}})$ as in Fig. 1.5. The blue coordinate system (X,Y,Z) is a coordinate system that stays fixed during the rotations (we will refer to it as the *Universal coordinate system*). However, we could have only rotated (in the opposite direction) the coordinate system without actually rotating the volume (see Fig. 1.7). The red coordinate system (X',Y',Z') is a coordinate system that reflects the transformation (we will refer to it as the *Transformed coordinate system*). This is an important source of misunderstandings: there are at least two ways of interpreting the transformation of an object (either we transform the object, or we keep the object fixed and transform only the coordinate system). In tensor analysis, there is a similar situation and the terms covariance and contravariance have been coined, depending on whether the physical magnitude (in this case, the volume) is transformed bound to the transformed axes or not. It has been proven (Dalarsson and Dalarsson, 2005) that when the original coordinate system is orthonormal (our case), covariant transformations and contravariant transformations are indistinguishable, that is, we cannot know whether we have transformed the axes and left the volume fixed, or we have transformed the volume together with the axes.

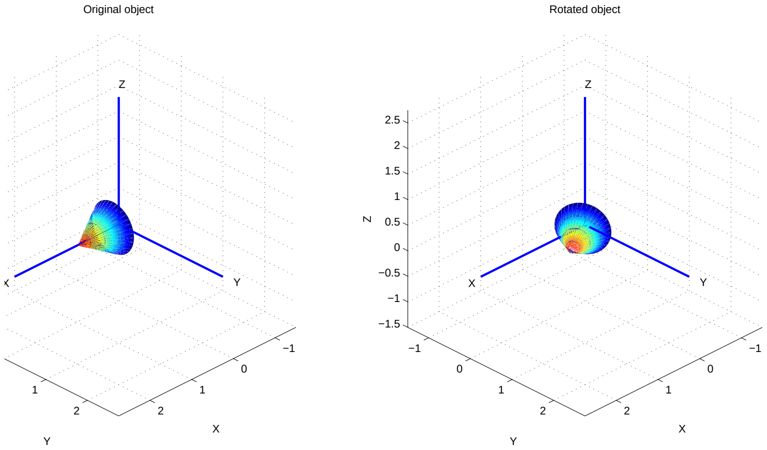

Fig. 1.5 Rotation of a volume applying a transformation matrix. We show in blue a universal coordinate system.

The first interpretation of Eq. (1.5) follows the idea of Fig. 1.5, i.e., $V_{\tilde{A}}(\tilde{\mathbf{r}})$ is the expression of the rotated volume in the fixed coordinate system ($\tilde{\mathbf{r}}$ is a point in the universal coordinate system). The second interpretation of Eq. (1.5) follows the idea of Fig. 1.7, i.e., $V_{\tilde{A}}(\tilde{\mathbf{r}})$ is the expression of the object (which has not moved) in the transformed coordinate system, which has moved (but in the opposite direction). Note that the result of Eq. (1.5) is the same regardless of our interpretation.

In the first interpretation, the multiplication $\tilde{\mathbf{r}}_{\tilde{A}} = \tilde{A}\tilde{\mathbf{r}}$ gives us the coordinates (in the universal coordinate system) of the point $\tilde{\mathbf{r}}$ (in the universal coordinate system) after transforming. In the second interpretation, $\tilde{A}$ provides the transformation matrix between the two coordinate systems (transformed and universal); i.e., the coordinate $\tilde{\mathbf{r}}_O$ in the transformed coordinate system is expressed in the universal coordinate system as $\tilde{\mathbf{r}}_U = \tilde{A}^{-1}\tilde{\mathbf{r}}_O$. Conversely, $\tilde{\mathbf{r}}_O = \tilde{A}\tilde{\mathbf{r}}_U$ (remember that both $\tilde{\mathbf{r}}_O$ and $\tilde{\mathbf{r}}_U$ must be expressed in homogeneous coordinates). In the absence of translations, the columns of R (a submatrix of $\tilde{A}$) represent the expression of the universal coordinate system axes in the transformed coordinate system. If $\tilde{A}$ is a rotation matrix (i.e., $\mathbf{t} = \mathbf{0}$ and $R^T R = RR^T = I$), then its rows represent the expression of the transformed coordinate system axes in the universal coordinate system. The double interpretation of Eq. (1.5) has animated quite a number of discussions in the field. In Appendix 2 we explore the consequences that these two possible interpretations

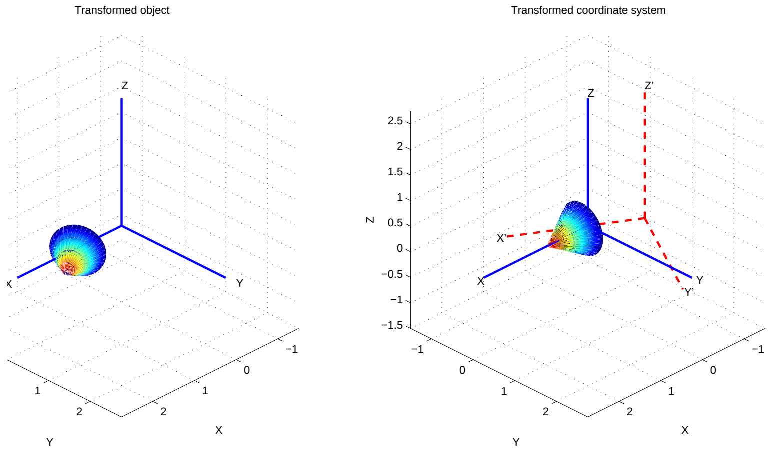

Fig. 1.6 Transformed object and coordinate system using an affine matrix composed by a rotation followed by a translation.

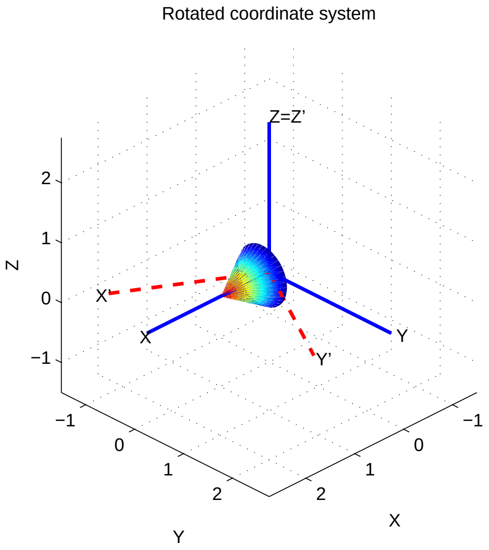

Fig. 1.7 Rotation of the transformed coordinate system (represented in red) using the same affine matrix as in Fig. 1.5. Note that the rotated object in Fig. 1.5 (right) seen from the universal coordinate system looks the same as the fixed object in this figure seen from the transformed coordinate system.

have on the sequential application of Euler angles (a common way of representing rotations in 3DEM).

The simplest interpretation of Eq. (1.6) is bound to the rotation of the transformed coordinate system leaving the object fixed. The projection image $I_{\tilde{A}}(\tilde{\mathbf{s}})$ can be obtained in two steps: 1) we transform the transformed coordinate system leaving the object fixed as in Fig. 1.6; 2) we integrate the input (unmoved) volume along the Z' direction. The X and Y axes of the resulting image are aligned with the X' and Y' axes of the transformed coordinate system (see Fig. 1.8). In the absence of translations, we may easily relate the transformed to the universal coordinate systems. In particular, the third row of matrix R gives the orientation of the projection direction, while the first and second rows indicate the orientation of the projection image with respect to the volume.

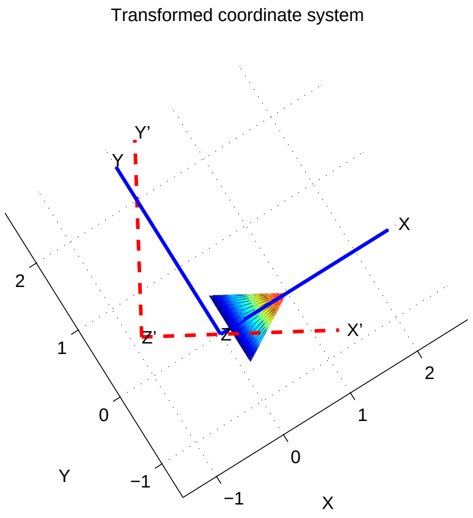

Fig. 1.8 Projection in the X' and Y' plane of the object transformed according to the matrix of Fig. 1.6. The X' and Y' axes have been oriented in their normalized orientation (Heymann et al, 2005).

1.7.2 In Fourier space

In the following we present how the situation presented in the previous section translates into Fourier space, paying particular attention to some unique properties pertaining to Fourier space that are broadly used in some 3DEM approaches.

Let us define the Fourier transform of a volume (similarly for an image) in homogeneous spatial frequency coordinates as

$$\widehat{V}(\tilde{\mathbf{R}}) = \mathscr{F}\mathscr{T}\{V(\tilde{\mathbf{r}})\} = \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}) e^{-i\left\langle \tilde{\mathbf{R}},\tilde{\mathbf{r}}\right\rangle} d\tilde{\mathbf{r}}. \tag{1.14}$$

We may transform both sides of Eq. (1.5) to obtain

$$\begin{aligned} \widehat{V}_{\tilde{A}}(\tilde{\mathbf{R}}) &= \int_{\mathbb{R}^3} V(\tilde{A}^{-1}\tilde{\mathbf{r}}) e^{-i\left\langle \tilde{\mathbf{R}},\tilde{\mathbf{r}}\right\rangle} d\tilde{\mathbf{r}} \\ &= \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}\tilde{\mathbf{r}}'\right\rangle} |\det(\tilde{A})| d\tilde{\mathbf{r}}'. \end{aligned} \tag{1.15}$$

If we now decompose $\tilde{A}$ into its translational and non-translational factors as in Eq. (1.62), then we have

$$\begin{aligned} \widehat{V}_{\tilde{A}}(\tilde{\mathbf{R}}) &= \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{\mathbf{R}},\tilde{T}\tilde{A}'\tilde{\mathbf{r}}'\right\rangle} |\det(\tilde{A})| d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})| \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}'(\tilde{\mathbf{r}}'+\tilde{\mathbf{t}})\right\rangle} d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})| e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}'\tilde{\mathbf{t}}\right\rangle} \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}'\tilde{\mathbf{r}}'\right\rangle} d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})| e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}'\tilde{\mathbf{t}}\right\rangle} \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{A}'^T\tilde{\mathbf{R}},\tilde{\mathbf{r}}'\right\rangle} d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})| e^{-i\left\langle \tilde{\mathbf{R}},\tilde{A}'\tilde{\mathbf{t}}\right\rangle} \widehat{V}(\tilde{A}'^T\tilde{\mathbf{R}}), \end{aligned} \tag{1.16}$$

where, as shown in Eq. (1.74), we have made use of the fact that the determinant of $\tilde{A}$ is the determinant of its scaling component, that is, the product of the eigenvalues of the scaling matrix. The equation above states that the value of the Fourier transform of $V_{\tilde{A}}$ at the spatial frequency $\tilde{\mathbf{R}}$ is the same, except for a scale factor and a phase factor related to the translation, as the one of the Fourier transform of V evaluated at the spatial frequency $\tilde{A}'^T\tilde{\mathbf{R}}$ (note that the relationship between the two spatial frequencies uses only the non-translational part of the affine transformation).

If we repeat the same exercise with the projection equation (Eq. (1.6)), we get the so-called Central Slice Theorem

$$\begin{aligned} \widehat{I}_{\tilde{A}}(\tilde{\mathbf{S}}) &= \int_{\mathbb{R}^2} \left(\int_{-\infty}^{\infty} V_{\tilde{A}}(\tilde{H}^T\tilde{\mathbf{s}}) dt \right) e^{-i\left\langle \tilde{\mathbf{S}},\tilde{\mathbf{s}}\right\rangle} d\tilde{\mathbf{s}} \\ &= \int_{\mathbb{R}^2} \left(\int_{-\infty}^{\infty} V(\tilde{A}^{-1}\tilde{H}^T\tilde{\mathbf{s}}) dt \right) e^{-i\left\langle \tilde{\mathbf{S}},\tilde{\mathbf{s}}\right\rangle} d\tilde{\mathbf{s}} \\ &= \int_{\mathbb{R}^3} V(\tilde{A}^{-1}\tilde{H}^T\tilde{\mathbf{s}}) e^{-i\left\langle \tilde{\mathbf{S}},\tilde{\mathbf{s}}\right\rangle} d\tilde{\mathbf{s}}dt \\ &= \int_{\mathbb{R}^3} V(\tilde{A}^{-1}\tilde{\mathbf{r}}) e^{-i\left\langle \tilde{\mathbf{S}},\tilde{H}_0\tilde{\mathbf{r}}\right\rangle} d\tilde{\mathbf{r}} \\ &= \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}') e^{-i\left\langle \tilde{\mathbf{S}},\tilde{H}_0\tilde{A}\tilde{\mathbf{r}}'\right\rangle} |\det(\tilde{A})| d\tilde{\mathbf{r}}', \end{aligned} \tag{1.17}$$

where H_0 is the projection matrix defined in Eq. (1.7) with $t = 0$ (this matrix takes a 3D point and projects it into the XY plane). If we decompose $\tilde{A}$ as in Eq. (1.62), then we have

$$\begin{aligned}\widehat{I}_{\tilde{A}}(\tilde{\mathbf{S}}) &= \int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}')e^{-i\left\langle \tilde{\mathbf{S}}, \tilde{H}_0\tilde{T}\tilde{A}'\tilde{\mathbf{r}}'\right\rangle}|\det(\tilde{A})|d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})|\int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}')e^{-i\left\langle \tilde{\mathbf{S}}, \tilde{H}_0\tilde{A}'(\tilde{\mathbf{r}}'+\tilde{\mathbf{t}})\right\rangle}d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})|e^{-i\left\langle \tilde{\mathbf{S}}, \tilde{H}_0\tilde{A}'\tilde{\mathbf{t}}\right\rangle}\int_{\mathbb{R}^3} V(\tilde{\mathbf{r}}')e^{-i\left\langle \tilde{\mathbf{S}}, \tilde{H}_0\tilde{A}'\tilde{\mathbf{r}}'\right\rangle}d\tilde{\mathbf{r}}' \\ &= |\det(\tilde{A})|e^{-i\left\langle \tilde{\mathbf{S}}, \tilde{H}_0\tilde{A}'\tilde{\mathbf{t}}\right\rangle}\widehat{V}(\tilde{A}'^T\tilde{H}_0^T\tilde{\mathbf{S}}).\end{aligned} \tag{1.18}$$

This equation provides a way of relating the 2D images frequencies $\tilde{\mathbf{S}}$ with the 3D location of the corresponding value in 3D Fourier space $\tilde{\mathbf{R}} = \tilde{A}'^T\tilde{H}_0^T\tilde{\mathbf{S}}$, or in non-homogeneous coordinates

$$\mathbf{R} = \mathbf{a}_1 S_x + \mathbf{a}_2 S_y, \tag{1.19}$$

where $\mathbf{a}_i$ is the i-th column of the matrix A_3. These spatial frequency points form a plane passing through the origin, whose implicit equation is

$$\langle \mathbf{R}, \mathbf{a}_3 \rangle = 0, \tag{1.20}$$

that is, $\mathbf{a}_3$ is the normal to this plane.

1.7.2.1 Common line

Let us assume that we have two independent projections ($I_{\tilde{A}}$ and $I_{\tilde{B}}$) whose projecting transformations $\tilde{A}$ and $\tilde{B}$ are plain rotations (this is the most common assumption when processing single particles). Let us also assume that the projections are translationally aligned so that $\mathbf{t}_A = \mathbf{t}_B = \mathbf{0}$. The Central Slice Theorem in this case would state that

$$\begin{aligned}\widehat{I}_{\tilde{A}}(\tilde{\mathbf{S}}) &= \widehat{V}(\tilde{A}'^T\tilde{H}_0^T\tilde{\mathbf{S}}) \\ \widehat{I}_{\tilde{B}}(\tilde{\mathbf{S}}) &= \widehat{V}(\tilde{B}'^T\tilde{H}_0^T\tilde{\mathbf{S}}).\end{aligned} \tag{1.21}$$

The two corresponding planes in Fourier space are given by the equations

$$\begin{aligned}\langle \mathbf{R}, \mathbf{a}_3 \rangle &= 0 \\ \langle \mathbf{R}, \mathbf{b}_3 \rangle &= 0.\end{aligned} \tag{1.22}$$

The intersection of these two planes is the line (called common line because the values of $\widehat{I}_{\tilde{A}}$ and $\widehat{I}_{\tilde{B}}$ are the same) that passes through the origin and whose direction is defined by

$$\mathbf{u} = \mathbf{a}_3 \times \mathbf{b}_3. \tag{1.23}$$

The spatial frequencies belonging to this line are of the form

$$\mathbf{R} = S\mathbf{u}, \tag{1.24}$$

where S is the absolute spatial frequency. If we want to find these spatial frequencies in the Fourier transform of the images, we have to solve the equations

$$\begin{aligned}\tilde{A}'^T\tilde{H}_0^T\tilde{\mathbf{S}}_A = \widetilde{S\mathbf{u}} &\Rightarrow \tilde{\mathbf{S}}_A = \tilde{H}_0\tilde{A}'\widetilde{S\mathbf{u}} \\ \tilde{B}'^T\tilde{H}_0^T\tilde{\mathbf{S}}_B = \widetilde{S\mathbf{u}} &\Rightarrow \tilde{\mathbf{S}}_B = \tilde{H}_0\tilde{B}'\widetilde{S\mathbf{u}}.\end{aligned} \tag{1.25}$$

Finally, we can state that thanks to the Central Slice Theorem, the two projections share a line such that

$$\widehat{I}_{\tilde{A}}(\tilde{\mathbf{S}}_A) = \widehat{I}_{\tilde{B}}(\tilde{\mathbf{S}}_B). \tag{1.26}$$

1.7.3 Projection transformations

We may also consider the sequential application of transformations for producing projections (see Eq. (1.6)). This is useful, for instance, for the purpose of relating the geometrical transformation needed to produce an image to its mirrored version (this is required, for example, when the 2D classification algorithm assigns to the same class some mirrored versions of experimental projections as is done by ML2D (Scheres et al, 2005) or CL2D (Sorzano et al, 2010)). As we shall see below, mirroring allows efficient computation of views from opposite projection directions.

Let us illustrate the sequential application of transformations to produce an image through the particular example of mirroring over the X axis (i.e., X does not change its sign, but Y does). However, the same kind of operations are needed, for instance, to correctly align Random Conical Tilt pairs of tilted and untilted projections.

Let us consider the projection generated through a rotation matrix $\tilde{R}$

$$I_{\tilde{R}}(\mathbf{s}) = \int_{-\infty}^{\infty} V(\tilde{R}^T \tilde{H}^T \tilde{\mathbf{s}}) dt, \tag{1.27}$$

where we have made use of the fact that for a rotation matrix $\tilde{R}^{-1} = \tilde{R}^T$.

A mirrored version of the image $I_{\tilde{R}}(\mathbf{s})$ can be described as

$$I_{mirror}(\tilde{\mathbf{s}}) = I(\tilde{M}_{2D}^T \tilde{\mathbf{s}}), \tag{1.28}$$

where $\tilde{M}_{2D}^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$. Substituting Eq. (1.27) into Eq. (1.28) we have

$$I_{mirror}(\tilde{\mathbf{s}}) = \int_{-\infty}^{\infty} V(\tilde{R}^T \tilde{H}^T \tilde{M}_{2D}^T \tilde{\mathbf{s}}) dt = \int_{-\infty}^{\infty} V(\tilde{R}^T \tilde{M}_{3D}^T \tilde{H}^T \tilde{\mathbf{s}}) dt, \tag{1.29}$$

being

$$\tilde{M}_{3D}^T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{1.30}$$

We can combine the two matrices into a single one as

$$I_{mirror}(\tilde{\mathbf{s}}) = I_{\tilde{M}_{3D}\tilde{R}}(\tilde{\mathbf{s}}), \tag{1.31}$$

i.e., the combined transformation matrix is $\tilde{R}' = \tilde{M}_{3D}\tilde{R}$, which is the same as $\tilde{R}$ but changing the direction of the Y and Z axes. The interpretation of this transformation matrix as a change of basis provides an important insight into the way the mirroring is performed. Particularly, we see that Z has changed its sign, meaning that the two projections are seen from opposite projection directions. The fact that X does not change and Y changes its sign is exactly the desired effect (mirroring over X).

In Appendix 3 we repeat this exercise with Euler angles and quaternions.

1.8 Conclusions

In this chapter we have studied in detail the issues needed to define a standard that allows interchange of geometrical information among 3DEM software packages. In particular, we have provided specific suggestions on how to:

- Define a common nomenclature: see Section 1.2.
- Define a coordinate system and its origin: see Section 1.3.
- Define a way of encoding geometrical transformations: see Section 1.4 and Eqs. (1.2) and (1.3).
- Define the way in which the geometrical transformations are applied to volumes and images: see Section 1.5 and Eqs. (1.5) and (1.6).
- Define a way of composing several geometrical transformations: see Section 1.6 and Eq. (1.12).

We have explored their meaning and implications as well as compared the elements suggested here to other more common ways of encoding geometrical information currently in use in 3DEM. We have provided mechanisms to change from any convention to the standard and back. Initiatives like EMX (Electron Microscopy eXchange) put forward the importance of defining such interchange standards so that software interoperability is guaranteed.

Acknowledgements

The authors would like to acknowledge economical support from the Spanish Ministry of Economy and Competitiveness through Grants AIC-A-2011-0638, BFU2009-09331, BIO2010-16566, ACI2009-1022, ACI2010-1088, CAM(S2010/BMD-2305) and NSF Grant 1114901. As well as postdoctoral Juan de la Cierva Grants with references JCI-2011-10185 and JCI-2010-07594. C.O.S. Sorzano is recipient of a RamÃşn y Cajal fellowship. This work was funded by Instruct, part of the European Strategy Forum on Research Infrastructures (ESFRI) and supported by national member subscriptions.

Appendix 1

We can establish a hierarchy of transformations. The most simple (translations, rotations and mirrors) are referred to as Euclidean transformations (beside preserving the above mentioned properties, they also preserve angles and distances). These transformations, together with shearing and scaling, are generalized by the affine transformations . These can be further generalized into the projective transformations (the matrix $\tilde{\mathbf{A}}$ is full) although these latter are normally not needed in electron microscopy because the electron beam is assumed to be a plane wavefront and the image recording is performed by orthographic projection.

Depending on the nature of R we have different transformations ranging from plain translations, scaling, mirrors, shears, and rotations to the full affine transformation. In the following sections we will analyze all these possibilities through examples. In general, we can analyze the nature of the transformations performed by the matrix R through its eigenvalue decomposition. The identity transformation is characterized by an eigenvalue 1 with multiplicity 3 (i.e., its eigenvectors span a subspace of dimension 3, that is $\mathbb{R}^3$). Shears are characterized by an eigenvalue 1 with multiplicity 3 (but the eigenvectors span a subspace of dimension 1 or 2). Isotropic scaling is characterized by an eigenvalue s (the scaling factor) with multiplicity 3 (eigenspace of dimension 3). Anisotropic scaling is characterized by several positive, real eigenvalues, each one with multiplicity 1 (corresponding eigenspaces of dimension 1). Mirrors have -1 as eigenvalue (with the dimension of the corresponding eigenspace equal to the multiplicity of -1 as eigenvalue). Finally, rotations are characterized by a pair of conjugate complex eigenvalues, with unit norm and multiplicity 1 (the corresponding subspaces spanned are of dimension 1). Note that we can perform anisotropic scaling, mirroring and rotation in a single plane by controlling the norm of the complex eigenvalues and their phases.

Translations

Matrix $\tilde{A}$ represents a translation if the matrix R is the identity matrix, that is

$$\tilde{A} = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{1.32}$$

The new point $\mathbf{r}_A$ becomes

$$\tilde{\mathbf{r}}_A = (r_x + t_x, r_y + t_y, r_z + t_z, 1)^T \Rightarrow \mathbf{r}_A = \mathbf{r} + \mathbf{t}. \tag{1.33}$$

Note that $\mathbf{r}_A$ and $\mathbf{r}$ are points, while $\mathbf{t}$ is a vector. The corresponding matrix R has only one eigenvalue (1) with multiplicity 3. The eigenspace associated to this eigenvalue is of dimension 3.

Scaling

We can scale any of the coordinates of the input point by setting the R matrix to be

$$R = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix}, \tag{1.34}$$

with $s_i \in (0, \infty)$. If $s_x = s_y = s_z$, then the scaling is called isotropic, otherwise each direction is scaled in a different way and the scaling is called anisotropic. The transformed coordinates, assuming no translation ($\mathbf{t} = \mathbf{0}$), are

$$\tilde{\mathbf{r}}_A = (s_x r_x, s_y r_y, s_z r_z, 1)^T. \tag{1.35}$$

Whether the scaling is a contraction or expansion depends on the way it is applied to the volume. If Eq. (1.5) (see below) is used, then the volume is expanded if $s_i > 1$, and the volume is contracted if $s_i < 1$. Matrix R above scales the volume along the basis axes (X, Y, Z). We could compress along any other orthogonal directions by applying any orthogonal matrix, O, as in

$$R = O \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{pmatrix} O^T. \tag{1.36}$$

Remember that a square matrix is orthogonal if $OO^T = O^T O = I$, in fact, as we will see below, an orthogonal matrix is a rotation.

The eigenvalues of the matrix R (even with the orthogonal matrix) are s_x, s_y and s_z (each one with multiplicity 1), and the eigenspace associated with each eigenvalue is of dimension 1.

Shears

Shearing can be understood as the result of compressing each axis with a different strength and a different direction causing the deformation of the volume. This is a common situation in a number of Electron Tomography (ET) applications due to the cutting of the sample, and less so in Single-Particle Analysis. Suppose we deform the volume by compressing the X axis in the direction of Y, the corresponding transformation matrix would be

$$R_{sh_1} = \begin{pmatrix} 1 & h_{xy} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{1.37}$$

and the new coordinates

$$\tilde{\mathbf{r}}_A = (r_x + h_{xy} r_y, r_y, r_z, 1)^T. \tag{1.38}$$

R_{sh_1} has 1 as its eigenvalue with multiplicity 3, but the eigenspace spanned $(\{(1,0,0)^T, (0,0,1)^T)\})$ has dimension only 2. We could also deform X in the direction of Z with the matrix

$$R_{sh_2} = \begin{pmatrix} 1 & h_{xy} & h_{xz} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.39}$$

In this case, the eigenvalues of R_{sh_2} are still 1 (3 times), but the eigenspace is now of dimension 1 $(\{(1,0,0)^T)\})$. Finally, we could also deform Y in the direction of Z with the matrix

$$R_{sh_3} = \begin{pmatrix} 1 & h_{xy} & h_{xz} \\ 0 & 1 & h_{yz} \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.40}$$

The eigenvalue and eigenspace structure of this matrix is the same as in the previous case.

It can be proven that any other shearing matrix can be expressed as a function of one of the R_{sh} matrices above by applying the appropriate orthogonal matrix:

$$R = OR_{sh_i}O^T. \tag{1.41}$$

Mirrors

We can mirror with respect to a plane by simply inverting one coordinate. For instance, the mirror with respect to the YZ plane is given by the matrix

$$R_{plane} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.42}$$

The eigenvalues of this matrix are -1 (with multiplicity 1 and dimension of the associated eigenspace 1) and 1 (with multiplicity 2 and dimension of the associated eigenspace 2).

We can also mirror with respect to a line. For instance, the mirror with respect to the Z line is given by

$$R_{line} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.43}$$

The eigenvalues of this matrix are -1 (with multiplicity 2 and dimension of the associated eigenspace 1) and 1 (with multiplicity 1 and dimension of the associated eigenspace 1).

Finally, we can mirror with respect to a point (the origin) in the direction with the matrix

$$R_{point} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}. \tag{1.44}$$

The eigenvalue of this matrix is -1 (with multiplicity 3 and dimension of the associated eigenspace 3).

As in the previous transformations, we can mirror with respect to any arbitrary plane or line by simply applying the appropriate orthogonal matrix

$$R = OR_iO^T. \tag{1.45}$$

Rotations

Among all affine transformations that can be represented with the matrix R, rotations play a prominent role in 3DEM because they are used to relate reconstructed volumes to experimental projections. R is a rotation matrix if it belongs to $SO(3)$ (the Special Orthogonal Group of degree 3, that is, the set of 3×3 orthogonal matrices with real coefficients and whose determinant is 1; remind that R is orthogonal if $R^TR = RR^T = I$).

Rotations about the standard X, Y, Z axes are particularly simple, and interestingly (as explained later) any rotation can be explained as the composition of three rotations around these axes. The rotation matrices normally used in 3DEM around each one of these axes are left-hand rotations (the left-hand thumb points along the rotation axis, and the rest of the fingers give the sense of positive rotations; looking at the left-hand, positive rotations are clockwise):

$$R_X(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{pmatrix} \tag{1.46}$$

$$R_Y(\alpha) = \begin{pmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{pmatrix} \tag{1.47}$$

$$R_Z(\alpha) = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{1.48}$$

The eigenvalues of any of these rotation matrices are $e^{i\alpha}$, $e^{-i\alpha}$, and 1 ($i=\sqrt{-1}$), each one with multiplicity 1 and the dimension of the corresponding eigenspace, 1.

As always, we can rotate around any axis by using the appropriate orthogonal matrix

$$R = OR_iO^T. \tag{1.49}$$

However, in the case of rotations, there are more compact ways of expressing any arbitrary rotation in terms of the so-called Euler angles, or quaternions and view vectors (see below).

Euler angles

Euler angles is the most common way of expressing rotations in 3DEM. They are normally described as a first rotation around a given coordinate axis, so that a new set of rotated coordinate system is formed. Then a second rotation around one of the transformed axes, to end with a third rotation around a twice transformed axis. Mathematically, we can say that $R=R_3R_2R_1$. It is indeed a very compact representation since with only 3 numbers (the three Euler angles) we can represent the full rotation matrix (with 3×3 parameters). In 3DEM the most widely used convention is the ZYZ (used by Spider (Frank et al, 1996), Xmipp (Sorzano et al, 2004), Imagic (van Heel et al, 1996), MRC (Crowther et al, 1996), and Frealign (Grigorieff, 2007)): first rotation around Z (this is called the rotational angle, ϕ), second rotation around Y (azimuthal angle, θ), and third rotation around Z (in-plane rotation, ψ). The corresponding Euler matrix is

$$\begin{aligned} R &= R_Z(\psi)R_Y(\theta)R_Z(\phi) \\ &= \begin{pmatrix} \cos\psi\cos\theta\cos\phi-\sin\psi\sin\phi & \cos\psi\cos\theta\sin\phi+\sin\psi\cos\phi & -\cos\psi\sin\theta \\ -\sin\psi\cos\theta\cos\phi-\cos\psi\sin\phi & -\sin\psi\cos\theta\sin\phi+\cos\psi\cos\phi & \sin\psi\sin\theta \\ \sin\theta\cos\phi & \sin\theta\sin\phi & \cos\theta \end{pmatrix}. \end{aligned} \tag{1.50}$$

In Imagic the rotation matrices are right-handed (counter clockwise) (Baldwin and Penczek, 2007), so the same matrix is obtained by using the angles $(-\phi,-\theta,-\psi)$, and the MRC obtains the same rotation matrix with the angles $(\phi,\theta,-\psi)$ (Baldwin and Penczek, 2007).

Given the rotation matrix, we can easily compute the Euler angles with the following algorithm

```
$|\sin\theta| = \sqrt{r_{13}^2 + r_{23}^2}$;
if $|\sin\theta| > 0$ then
    $\phi$ = atan2($r_{32}$,$r_{31}$);
    $\psi$ = atan2($r_{23}$,$-r_{13}$);
    if sin($\psi$)=0 then
        $s$ = sign($\frac{-r_{13}}{\cos(\psi)}$);
    else
        $s$ = sign($\frac{r_{23}}{\sin(\psi)}$);
    end
    $\theta$ = atan2($s|\sin\theta|$, $r_{33}$);
else
    $\phi = 0$;
    if sign($r_{33}$)>0 then
        $\theta = 0$;
        $\psi$ = atan2($-r_{21}$,$r_{11}$);
    else
        $\theta = \pi$;
        $\psi$ = atan2($r_{21}$,$-r_{11}$);
    end
end
```

where sign(x) is the sign of x (1 or -1) and atan2(y,x) is the arc tangent function with 2 arguments (i.e., that explicitly considers the angular quadrants).

However, ZYZ is not the only possible decomposition of matrix R. There are numerous ways to choose the axes and the decomposition of R as a product of three simpler rotation matrices is not unique. Different conventions exist, up to 12: ZYZ, ZXZ, XZX, XYX, YXY, YZY, XYZ, XZY, YXZ, YZX, ZXY and ZYX (Shoemake, 1994). Indeed, these are in use in 3DEM: EMAN (Ludtke et al, 1999) uses ZXZ. Baldwin and Penczek (2007) provides the algorithm to convert from EMAN ZXZ convention to the standard ZYZ convention. EMAN uses the angles ϕ, azimuth (az) and altitude (alt) in the following combination

$$R_{EMAN} = R_Z(\phi_{EMAN})R_X(alt)R_Z(az). \tag{1.51}$$

We can convert between the two systems using

$$\begin{array}{l} az = \phi + \frac{\pi}{2} \\ alt = \theta \\ \phi_{EMAN} = \psi - \frac{\pi}{2} \end{array} \tag{1.52}$$

Non-uniqueness of the Euler angles

Even if a single angular decomposition is agreed on (e.g., ZYZ), there is a second source of non-uniqueness. It can be easily proven that $R(\phi,\theta,\psi) = R(\phi +$

$\pi, -\theta, \psi - \pi)$, i.e., we can express one rotation with two totally different sets of angles using the same Eulerian convention (ZYZ). A third source of non-uniqueness comes from the so-called gimbal lock problem (Koks, 2006). Let us assume that $\theta = 0$, then the rotation matrix becomes $R(\phi, \theta, \psi) = R_Z(\psi)R_Y(0)R_Z(\phi) = R_Z(\psi + \phi)$. We have a single degree of freedom (a single rotation matrix), despite the fact that we have fixed only a single angle out of the possible three. This problem is shared by all Euler angle conventions with two rotations around the same axis (in the ZYZ convention, the first and third rotation are around the same axis). On top of this loss of degrees of freedom, the set of possible ways to describe the same rotation becomes infinite. It is obvious that $R(\phi, 0, \psi) = R(\phi + \alpha, 0, \psi - \alpha)$ for any value of α (again, different sets of Euler angles represent the same rotation).

The immediate consequence of this non-uniqueness problem for 3DEM is that to determine if two different projections are close to each other in their projection directions, it does not suffice comparing their two Euler angle sets (note that the projection direction is determined only by two angles, ϕ and θ). Instead, we will have to use the rotation matrix R and check whether the two projection directions (i.e., the third row of the corresponding rotation matrices) are close to each other.

Quaternions and View vectors

Quaternions were introduced in 1843 by Hamilton as an extension to complex numbers (in a simple way we can think of them as complex numbers that instead of having an imaginary part, they have a 3D vector as imaginary part). Hamilton defined a normed division algebra upon this set (i.e., he specified the way of summing, subtracting, multiplying, dividing and defining a norm). Quaternions can be written in different notations. The one most similar to complex numbers is to write a quaternion as $q = a + bi + cj + dk$ where a is the equivalent of the real part, and b, c, d are the equivalent of the imaginary part (now, three "imaginary parts"), with i, j, k being the equivalent of the imaginary number i. We can also write the quaternion as a 4D vector, $q = (a, b, c, d)$, or as a "sum" of a number and a 3D vector, $q = a + (b, c, d)$.

Algebra with quaternions is similar to the algebra with complex numbers. Addition is defined in the standard way

$$(a,b,c,d) + (a',b',c',d') = (a + a', b + b', c + c', d + d'). \tag{1.53}$$

However, multiplication is more tricky. Let us first review the multiplication rules with complex numbers. If we think of a complex number $z = a + bi$, we may think of it as $z = a \cdot 1 + b \cdot i$. Multiplying two such numbers would amount to

$$(a \cdot 1 + b \cdot i) \cdot (a' \cdot 1 + b' \cdot i) = aa'(1 \cdot 1) + ba'(i \cdot 1) + ab'(1 \cdot i) + bb'(i \cdot i). \tag{1.54}$$

The following table describes the multiplication rules for 1 and i

·	1	i
1	1	i
i	i	-1

Using these rules we can write:

$$\begin{aligned}(a\cdot 1+b\cdot i)\cdot(a'\cdot 1+b'\cdot i) &= aa'(1\cdot 1)+ba'(i\cdot 1)+ab'(1\cdot i)+bb'(i\cdot i)\\ &= (aa'-bb')\cdot 1+(ba'+ab')\cdot i,\end{aligned} \tag{1.55}$$

which gives the standard multiplication rule for complex numbers. Multiplication with quaternions is similar only that now, the multiplication rules are

·	1	i	j	k
1	1	i	j	k
i	1	-1	k	-j
j	j	-k	-1	i
k	k	j	-i	-1

Interestingly, quaternions provide a framework in which a division algebra (an algebra in which division is available) can be defined on 3D vectors (Alsina and Bayer Isant, 2004), although this digression falls out of the scope of this chapter.

Quaternions are much less known to the EM community, but they can also be used to describe rotations and they are used in the internal representation of rotations by Bsoft (Harauz, 1990; Heymann and Belnap, 2007). In particular, a rotation with an α angle around a given 3D axis $\mathbf{u}$ can be encoded into a quaternion as

$$q_{\mathbf{u},\alpha}=\cos(\tfrac{\alpha}{2})+\sin(\tfrac{\alpha}{2})\frac{\mathbf{u}}{\|\mathbf{u}\|}. \tag{1.56}$$

If the quaternion is represented as $q_{\mathbf{u},\alpha}=a+(b,c,d)$, then the corresponding rotation matrix is

$$R=\begin{pmatrix} a^2+b^2-c^2-d^2 & 2bc-ad & 2bd+2ac\\ 2bc+2ad & a^2-b^2+c^2-d^2 & 2cd-2ab\\ 2bd-2ac & 2cd+2ab & a^2-b^2-c^2+d^2\end{pmatrix} \tag{1.57}$$

Any time that a rotation has to be applied on an image or volume, the quaternion is translated into its corresponding rotation matrix, and then this is applied using a formula similar to Eq. (2) in the main text. Conversely, it has been shown that for each rotation matrix, there is a unique quaternion whose norm is 1 (unitary quaternion) (Kuipers, 1999). We can recover the quaternion from the rotation matrix diagonal by solving the following equation system:

$$\begin{pmatrix}1&1&-1&-1\\1&-1&1&-1\\1&-1&-1&1\\1&1&1&1\end{pmatrix}\begin{pmatrix}a^2\\b^2\\c^2\\d^2\end{pmatrix}=\begin{pmatrix}r_{11}\\r_{22}\\r_{33}\\1\end{pmatrix}. \tag{1.58}$$

The last equation of this equation system simply forces the quaternion to be unitary. The signs of the quaternion components can be calculated as

sign(a)=1;
sign(b)=sign($r_{32}-r_{23}$);
sign(c)=sign($r_{13}-r_{31}$);
sign(d)=sign($r_{21}-r_{12}$);

Uniqueness of the representation (in this case the quaternion) is important because we can compare two geometrical specifications by simply comparing their corresponding representations.

Heymann et al (2005) introduces the use of view vectors as another way of representing rotations. A rotation is defined by an axis (x,y,z) and a rotation α around this axis. Its representation as a quaternion is obvious and Heymann et al (2005) and Baldwin and Penczek (2007) present formulas on how to convert from view vectors to the Euler ZYZ angular convention. In particular, given the Euler angles we can compute the view vector parameters as

$$\begin{aligned} x &= \cos\phi\sin\theta \\ y &= \sin\phi\sin\theta \\ z &= \cos\theta \\ \alpha &= \phi+\psi. \end{aligned} \tag{1.59}$$

Conversely, given the view vector we can recover the Euler angles as

```
**if** $x = y = 0$ **then**
    $\phi = \theta = 0$;
**else**
    $\phi = \arctan\left(\frac{y}{x}\right)$ ;
    $\theta = \arccos(z)$;
**end**
$\psi = \alpha - \phi$;
```

Affine transformations: a composition of several transformations

All the previous transformations are generalized by the affine transformation whose matrix has already been presented in Eq. (1.3). The affine transformation can adopt any of the previous forms for translations, scalings, shearing, mirrors, and rotations, and even combine any number of them. Let's say we concatenate a finite number, N, of transformations as

$$\begin{aligned} \tilde{\mathbf{r}}_{\tilde{A}_1} &= \tilde{A}_1\tilde{\mathbf{r}} \\ \tilde{\mathbf{r}}_{\tilde{A}_2} &= \tilde{A}_2\tilde{\mathbf{r}}_{\tilde{A}_1} \\ &\dots \\ \tilde{\mathbf{r}}_{\tilde{A}_N} &= \tilde{A}_N\tilde{\mathbf{r}}_{\tilde{A}_{N-1}} \end{aligned} \tag{1.60}$$

We can combine all these transformations into a single affine transformation given by

$$\tilde{\mathbf{r}}_{\tilde{A}_N} = \left(\tilde{A}_N ... \tilde{A}_2 \tilde{A}_1\right) \tilde{\mathbf{r}} \tag{1.61}$$

Recovering basic transformations from the affine transformation

We have seen in the previous section, that given a sequence of basic transformations (shifts, rotations, shears, mirrors, ...) we can trivially combine them into a single affine transformation that performs all the steps one after the other. Going in the other direction is not so easy, in general, as we will show in this section.

Given the affine transformation $\tilde{A}$ we first decompose it as

$$\tilde{A} = \tilde{T}\tilde{A}' = \begin{pmatrix} I_3 & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} A_3 & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix}, \tag{1.62}$$

where I_3 is the 3×3 identity matrix, and A_3 is the 3×3 top-left submatrix of $\tilde{A}$.

We now apply a polar decomposition to the matrix A_3 to factorize it as

$$A_3 = U_1 S. \tag{1.63}$$

The polar decomposition breaks the original matrix A_3 into a unitary matrix U and a positive semi-definite Hermitian matrix S. Since we required the affine transformation to be invertible, then S is a symmetric, positive definite matrix. The polar decomposition is computed through the Singular Value Decomposition of the input matrix:

$$A_3 = W \Sigma V^T. \tag{1.64}$$

Then, the matrices U_1 and S are computed as

$$\begin{aligned} U_1 &= W V^T \\ S &= V \Sigma V^T \end{aligned} \tag{1.65}$$

Since V is an orthogonal matrix it is a rotation. Since the eigenvalues of Σ are positive (because S is positive definite), then S is simply a scaling transformation along some arbitrary axes given by the rotation V.

Matrix U_1 contains all the transformations whose eigenvalues have unit module (shears, mirrors, and rotations). We now apply a QR factorization to U_1

$$U_1 = U_2 U_3 \tag{1.66}$$

As a result of the QR factorization, matrix U_2 is unitary and U_3 is upper triangular. We will further factorize these matrices separating their mirror components (M_1 and M_2) from the rotation (R) and shearing (Sh)

$$\begin{aligned} U_2 &= R M_1 \\ U_3 &= M_2 Sh \end{aligned} \tag{1.67}$$

We can do so by analyzing the eigendecomposition of U_2

$$U_2 = PDP^{-1} \tag{1.68}$$

If there are no complex eigenvalues, then $R = I_3$ and $M_1 = U_2$. If there are, then we set the non-complex eigenvalue to 1 (in a separate matrix D') and compute the rotation as

$$R = PD'P^{-1}. \tag{1.69}$$

The mirroring component of U_2 can be computed as

$$M_1 = R^T U_2. \tag{1.70}$$

We can estimate M_2 to be the main diagonal of U_3

$$M_2 = \text{diag}(U_3), \tag{1.71}$$

and Sh to be

$$Sh = U_3 M_2^{-1} \tag{1.72}$$

We note that we have decomposed U_1 as

$$U_1 = U_2 U_3 = RM_1 M_2 Sh = RMSh \tag{1.73}$$

where R is a rotation matrix, M is a mirror matrix, and Sh is a shearing matrix. Putting it all together, in homogeneous coordinates, we have

$$\tilde{A} = \tilde{T}\tilde{R}\tilde{M}\tilde{Sh}\tilde{S}. \tag{1.74}$$

Appendix 2

Orienting volumes

Euler angles have been traditionally "explained" in the field as "The three Euler angles are defined as three rotations ... The first rotation is a rotation about the standard Z-axis by an angle ϕ ... The second rotation is a rotation about the [new] Y' axis by an angle θ ... The third rotation is a rotation about the [new] Z'' axis by an angle ψ" (see, for example, Heymann et al (2005)).

Despite this seemingly clear definition of Euler angles, its application is ambiguous since it does not specify whether we should keep the object bound to the rotating coordinate system or not. This controversy has animated not few geometrical discussions within the EM community. In this section we show that there are several ways of sequentially applying the Euler angles, all of them correct.

In the following we consider an affine transformation matrix that is composed only by a rotation matrix. We will refer to it as $\tilde{R}$. Combining our rotation equation

$V_{\tilde{R}}(\tilde{\mathbf{r}}) = V(\tilde{R}^{-1}\tilde{\mathbf{r}})$ with the Euler decomposition of the rotation we get

$$V_{\tilde{R}}(\tilde{\mathbf{r}}) = V\left((\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi))^{-1}\tilde{\mathbf{r}}\right).$$

This is a simple "operational" recipe, easy to implement (we have added a subindex to the rotation matrices to remember the order in which they are applied). If we follow the first interpretation (see main text) of the rotation formula $V_{\tilde{R}}(\tilde{\mathbf{r}}) = V(\tilde{R}^{-1}\tilde{\mathbf{r}})$, then we produce the rotated object in the universal coordinate system. We can fill the rotated object at universal locations $\tilde{\mathbf{r}}$ by simply looking at the original volume at the location $\tilde{R}^{-1}\tilde{\mathbf{r}}$. Note that if $\tilde{R}$ is a rotation matrix, we have to use $\tilde{R}^{-1}$ that is rotating in the opposite direction.

If we want to follow, step by step, the simple rotations given by the Euler matrix, we can do so in three different ways:

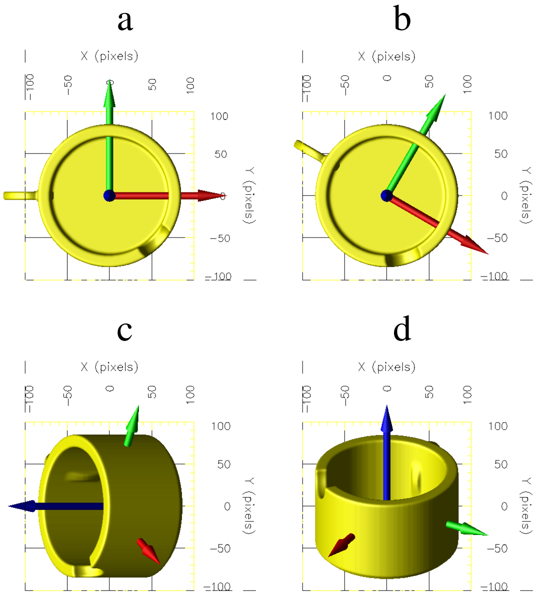

Fig. 1.9 Example of rotation about the universal axes. a) Original, unrotated volume. b) Volume after rotating 30° about Z. c) Volume after rotating 60° about Y. d) Volume after rotating 90° about Z. A movie showing the transformation from (a) to (d) is available at the URL http://i2pc.cnb.csic.es/rotationsResources/rotateVolumeUniversal.html.

1. Rotate the volume about the universal coordinate axes. The simplest way of reproducing the $V\left((\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi))^{-1}\tilde{\mathbf{r}}\right)$ step by step is by simply following the instructions "encoded" in $\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi)$ (the whole procedure is illustrated in Fig. 1.9). An important note, especially in relationship to the next

procedure, is that in our figure we have represented the universal coordinate system by the grid, and the transformed coordinate system by colored arrows. By convention, we establish that in a given coordinate system the standard way of looking at an object is by placing X pointing right, Y pointing up, and Z pointing to the observer (this is easily seen in Fig. 1.9a).

Following the "encoded instructions", we must first rotate the volume by an angle ϕ about Z. This produces a new volume $V_{Z1}(\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{\mathbf{r}})$. Note (Fig. 1.9b) that $\tilde{R}_{Z1}$ is a left-hand rotation matrix and, correspondingly, the volume is rotating clock-wise. Now, we rotate this volume about the universal Y by θ. The new volume is then $V_{Y2}(\tilde{\mathbf{r}}) = V_{Z1}(\tilde{R}_{Y2}^{-1}(\theta)\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{R}_{Y2}^{-1}(\theta)\tilde{\mathbf{r}})$. Finally, the third rotation is again about the universal Z by ψ. The final volume is $V_{Z3}(\tilde{\mathbf{r}}) = V_{Y2}(\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{R}_{Z2}^{-1}(\theta)\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}}) = V\left((\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi))^{-1}\tilde{\mathbf{r}}\right)$, as required by the rotation equation.

2. Rotate the transformed coordinate system while keeping the object fixed. The second interpretation (see Section 1.7.1) of the rotation equation $V_{\tilde{R}}(\tilde{\mathbf{r}}) = V(\tilde{R}^{-1}\tilde{\mathbf{r}})$ tells us that $V_{\tilde{R}}(\tilde{\mathbf{r}})$ is the expression of the fixed object in the rotated transformed coordinate system. The first rotation is about Z by ϕ. As in the previous case, this produces the volume $V_{Z_1}(\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{\mathbf{r}})$. However, the interpretation of the $V_{Z_1}(\tilde{\mathbf{r}})$ is quite different: $V_{Z_1}(\tilde{\mathbf{r}})$ in our previous procedure was the value of the rotated volume at the universal location $\tilde{\mathbf{r}}$ after one rotation; now $V_{Z_1}(\tilde{\mathbf{r}})$ is the value of the unrotated volume at the transformed location $\tilde{\mathbf{r}}$ after one rotation. Now note (Fig. 1.10b) that the transformed coordinate system is rotating counterclock-wise. For the second rotation we have to rotate the new coordinate axes about the new Y' producing a new coordinate system (X'',Y'',Z''). From this new coordinate system, the unrotated object looks as $V_{Y_2}(\tilde{\mathbf{r}}) = V_{Z_1}(\tilde{R}_{Y2}^{-1}(\theta)\tilde{\mathbf{r}})$. Note that we are using the rotation matrix $\tilde{R}_Y$ and not $\tilde{R}_Y''$ because in the coordinate system (X',Y',Z'), the expression of Y' is $(0,1,0)$ and, therefore, the appropriate rotation matrix is the one we usually associate with Y. So far we have expressed our fixed object in the coordinate system (X'',Y'',Z'') as a function of how it is seen in the coordinate system (X',Y',Z'). If we want to refer it to the original object, we simply substitute $V_{Z_1}(\tilde{\mathbf{r}})$ by its value, to get $V_{Y_2}(\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{R}_{Y2}^{-1}(\theta)\tilde{\mathbf{r}})$, which is exactly the same functional relationship as the one we obtained in our previous procedure although its interpretation is quite different. The last rotation is performed around the new Z'', but in the coordinate system of (X'',Y'',Z'') this is seen as a standard rotation about Z: $V_{Z_3}(\tilde{\mathbf{r}}) = V_{Y_2}(\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}})$. Again, substituting $V_{Y_2}(\tilde{\mathbf{r}})$ by its value, we get $V_{Z_3}(\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{R}_{Y2}^{-1}(\theta)\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}}) = V\left((\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi))^{-1}\tilde{\mathbf{r}}\right)$, that is, the functional relationship of the rotation equation. We see that the tracking of the Euler rotations from this interpretation of the rotation equation is not as straightforward as in the previous procedure. Moreover, this second interpretation has an important consequence as shown in Fig. 1.10. During the procedure we have to keep the object fixed and rotate the axes, but $V_{Z_3}(\tilde{\mathbf{r}})$ gives us the expression of the fixed object in the transformed coordinate system. If we want to see the rotated object, we have to be an observer in the transformed coordinate system, i.e., we

have to align the new axes in the standard way an observer at that coordinate system would look at the object.

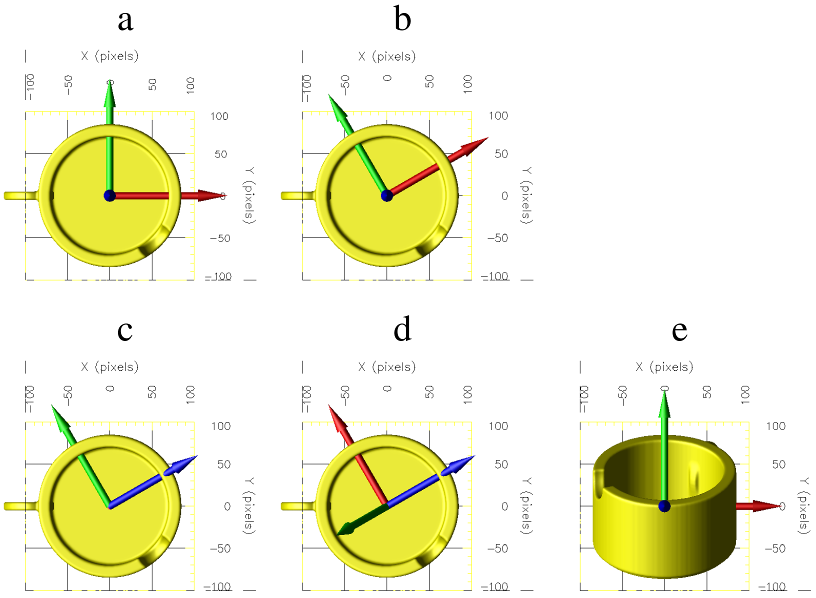

Fig. 1.10 Example of rotation keeping the object fixed and rotating only the coordinate axes. Note that in this approach the system of coordinates formed by the three arrows is not attached to the volume, that is, the volume does not rotate. a) Original, unrotated system of coordinates. b) System of coordinates after rotating 30° about Z. c) System of coordinates after rotating 60° about Y. d) System of coordinates after rotating 90° about Z. e) View of the volume after the system of coordinates formed by the three arrows is used to determine the standard display orientation (X pointing right, Y pointing up, and Z point to the observer; compare this figure to Figure 1.9d). A movie showing the transformation from (a) to (e) is available at URL http://i2pc.cnb.csic.es/rotationsResources/rotateIntrinsicCoordinate.html.

3. Rotate the volume about the transformed coordinate axes. The traditional way of describing the Euler angles is *"First rotate about Z by ϕ. Then, rotate θ about the new Y. Finally, rotate ψ about the new Z"*. We have seen that this description corresponds to the second interpretation of the rotation equation. However, if we misunderstand this point and rotate the object by ϕ and then try to rotate by θ about the new Y, we will be lead to a rotation that is not the same as the one obtained by the previous procedures. If we still want to rotate the volume about its new transformed axes, we can get the same rotation as before simply by inverting the order of the rotations as shown in the following derivation (the whole procedure can be visually followed in Fig. 1.11). The first rotation is around Z but using the angle ψ (!). So we use the standard rotation matrix $\tilde{R}_{Z3}(\psi)$. This will give us the new object $V_{Z_1}(\tilde{\mathbf{r}}) = V(\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}})$ and a new set of coordinate axes

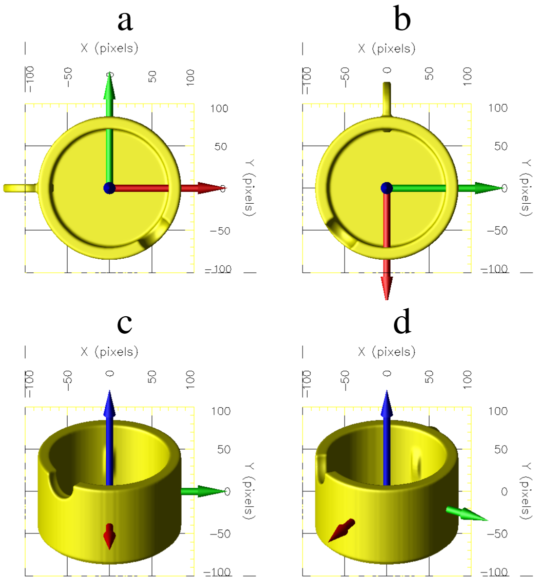

Fig. 1.11 Example of rotation about the transformed axes: a) Original, unrotated volume. b) Volume after rotating 90° about Z (blue arrow in (a)). c) Volume after rotating 60° about the new Y (Y', green arrow in (b)). d) Volume after rotating 30° about the new Z (Z'', blue arrow in (c); compare to Figure 1.10e). A movie showing the transformation from (a) to (d) is available at http://i2pc.cnb.csic.es/rotationsResources/rotateVolumeIntrinsic.html.

(X',Y',Z') whose expression in the universal coordinate system is given by the rows of the matrix $\tilde{R}_{Z3}(\psi)$. The second rotation should be performed about the new Y, i.e. Y'. The rotation matrix of θ radians around Y' is given by $\tilde{R}_{Y'}(\theta) = \tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z3}^{-1}(\psi)$. This produces a new volume $V_{Y_2}(\tilde{\mathbf{r}}) = V_{Z_1}(\tilde{R}_{Y'}^{-1}(\theta)\tilde{\mathbf{r}}) = V(\tilde{R}_{Y2}^{-1}(\theta)\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}})$ and a new set of axes (X'',Y'',Z'') whose expression in the universal coordinate system is given by the rows of the matrix $\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)$. Finally, the third rotation must be performed about the new Z'' but with angle ϕ (!), whose rotation matrix is $\tilde{R}_{Z''}(\phi) = \tilde{R}_{Y'}(\theta)\tilde{R}_{Z1}(\phi)\tilde{R}_{Y'}^{-1}(\theta) = \ldots = \tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi)\tilde{R}_{Y2}^{-1}(\theta)\tilde{R}_{Z3}^{-1}(\psi)$. This last rotation produces the volume $V_{Z_3}(\tilde{\mathbf{r}}) = V_{Y_2}(\tilde{R}_{Z''}^{-1}(\phi)\tilde{\mathbf{r}}) = V(\tilde{R}_{Z1}^{-1}(\phi)\tilde{R}_{Y2}^{-1}(\theta)\tilde{R}_{Z3}^{-1}(\psi)\tilde{\mathbf{r}}) = V\left((\tilde{R}_{Z3}(\psi)\tilde{R}_{Y2}(\theta)\tilde{R}_{Z1}(\phi))^{-1}\tilde{\mathbf{r}}\right)$.

Orienting projections

On many ocasions it is useful to have an intuitive idea of the projection orientation given some Euler angles. The formal definition has already been given in Eq. (1.6). In the following, we provide a "manual" rule of how to determine the projection direction and its in-plane rotation given (ϕ,θ,ψ). An easy way to do this is depicted in Fig. 1.12. First, the tilt angle is applied, moving the projection coordinate system and the projection ray (in red) along the “meridian” that passes through X. Then, we move ϕ degrees around the “parallel” in which we previously stopped. Finally, you rotate the projection, in-plane, according to ψ. These movements define the “camera” coordinate system, and the projection will be computed as line integrals in the direction parallel to Z'.

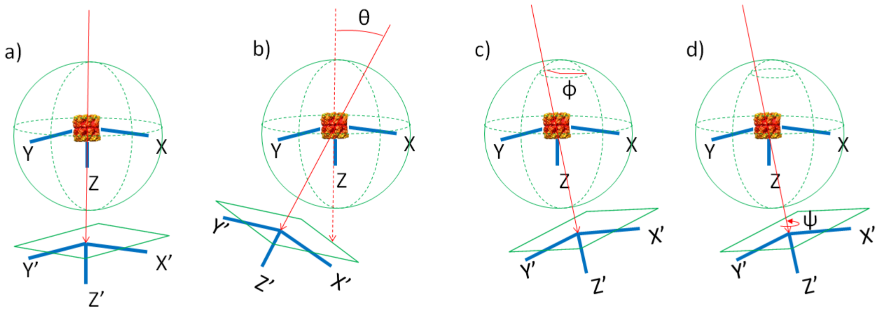

Fig. 1.12 a) Original projection orientation with $\phi = \theta = \psi = 0$. b) Apply the tilt angle θ with the projection ray (in red) bound to the projection coordinate system. c) Apply the rotational angle ϕ. d) Fianlly, apply the in-plane rotation.

Appendix 3

Mirroring using Euler angles

Let us assume that the rotation matrix $\tilde{R}$ can be represented by the ZYZ Euler angles (ϕ, θ, ψ). Now, we are interested in finding some angles (ϕ', θ', ψ') such that

$$\tilde{R}(\phi', \theta', \psi') = \tilde{M}_{3D}\tilde{R}(\phi, \theta, \psi). \tag{1.75}$$

We first note that $\tilde{M}_{3D} = \tilde{R}_X(\pi) = \tilde{R}_Y(\pi)\tilde{R}_Z(\pi)$, then expanding the definition of each side of the equation, we have

$$\begin{aligned}\tilde{R}_Z(\psi')\tilde{R}_Y(\theta')\tilde{R}_Z(\phi') &= \tilde{R}_Y(\pi)\tilde{R}_Z(\pi)\tilde{R}_Z(\psi)\tilde{R}_Y(\theta)\tilde{R}_Z(\phi) \\ &= \tilde{R}_Y(\pi)\tilde{R}_Z(\psi+\pi)\tilde{R}_Y(\theta)\tilde{R}_Z(\phi) \\ &= \tilde{R}_Z(-(\psi+\pi))\tilde{R}_Y(\theta+\pi)\tilde{R}_Z(\phi),\end{aligned} \tag{1.76}$$

where we have made use of the fact that $\tilde{R}_Y(\pi)\tilde{R}_Z(\alpha) = \tilde{R}_Z(-\alpha)\tilde{R}_Y(\pi)$. Thus, we see that the angles of the mirrored version of $I_{\tilde{R}}(\tilde{\mathbf{s}})$ are $(\phi, \theta+\pi, -(\psi+\pi))$. This is a relatively trivial operation, and it provides some intuitive insight on how the two projection directions are related (the azimuthal angle is the same (ϕ), and the tilt angle (θ) is just the opposite in the projection sphere).

Mirroring using quaternions

If we perform the same exercise using quaternions, $\tilde{M}_{3D}$ is represented by the quaternion $q_{(1,0,0),\pi}$. Let us assume that matrix $\tilde{R}$ is represented by some unitary quaternion $q_{\mathbf{u},\alpha} = (a,b,c,d)$. The composition of rotations $\tilde{M}_{3D}\tilde{R}$ is represented by the multiplication of quaternions

$$\begin{aligned} q_{\mathbf{u}',\alpha'} = q_{(1,0,0),\pi} q_{\mathbf{u},\alpha} &= (-a,d,-c,b) \\ &= \sin(\tfrac{\alpha}{2}) + (\cos(\tfrac{\alpha}{2})\mathbf{e}_x + \sin(\tfrac{\alpha}{2})\mathbf{e}_x \times \mathbf{u}), \end{aligned} \tag{1.77}$$

where $\mathbf{e}_x^T = (1,0,0)$. Although the operation at the quaternion component level is trivial (some rearrangements and changes of signs), we have totally lost any intuition about how the two projection directions (mirrored and not mirrored) are related.

References

Alsina M, Bayer Isant P (2004) Quaternion Orders, Quadratic Forms, and Shimura Curves. American Mathematical Society

Baldwin PR, Penczek PA (2007) The transform class in sparx and eman2. J Structural Biology 157(1):250–261

Cantele F, Zampighi L, Radermacher M, Zampighi G, Lanzavecchia S (2007) Local refinement: an attempt to correct for shrinkage and distortion in electron tomography. J Structural Biology 158:59–70

Crowther RA, Henderson R, Smith JM (1996) MRC image processing programs. J Structural Biology 116:9–16

Dalarsson M, Dalarsson N (2005) Tensors, relativity and cosmology. Elsevier Academic Press

Frank J, Radermacher M, Penczek P, Zhu J, Li Y, Ladjadj M, Leith A (1996) SPIDER and WEB: Processing and visualization of images in 3D electron microscopy and related fields. J Structural Biology 116:190–9

Grigorieff N (2007) Frealign: High-resolution refinement of single particle structures. J Structural Biology 157:117–125

Harauz G (1990) Representation of rotations by unit quaternions. Ultramicroscopy 33:209–213

van Heel M, Harauz G, Orlova EV, Schmidt R, Schatz M (1996) A new generation of the IMAGIC image processing system. J Structural Biology 116:17–24

Heymann B, Belnap D (2007) Bsoft: Image processing and molecular modeling for electron microscopy. J Structural Biology 157:3–18

Heymann JB, Chagoyen M, Belnap DM (2005) Common conventions for interchange and archiving of three-dimensional electron microscopy information in structural biology. J Structural Biology 151:196–207

Jain AK (1989) Fundamentals of digital image processing. Prentice-Hall

Koks D (2006) Explorations in Mathematical Physics. Springer

Kremer JR, Mastronarde DN, McIntosh JR (1996) Computer visualization of three-dimensional image data using imod. J Structural Biology pp 71–76

Kuipers JB (1999) Quaternions and rotation sequences. Princeton University Press

Ludtke SJ, Baldwin PR, Chiu W (1999) EMAN: Semiautomated software for high-resolution single-particle reconstructions. J Structural Biology 128:82–97

Rossmann MG, Blow DM (1962) The detection of sub-units within the crystallographic asymmetric unit. Acta Crystallographica 15:24–31

Scheres SHW, Valle M, Núñez R, Sorzano COS, Marabini R, Herman GT, Carazo JM (2005) Maximum-likelihood multi-reference refinement for electron microscopy images. J Molecular Biology 348:139–149

Shoemake K (1994) Graphic Gems IV, AP Professional (Academic Press), chap Euler Angle Conversion, pp 222–229

Sorzano COS, Marabini R, , VelÃązquez-Muriel J, Bilbao-Castro JR, Scheres SHW, Carazo JM, Pascual-Montano A (2004) XMIPP: A new generation of an open-source image processing package for electron microscopy. J Structural Biology 148:194–204

Sorzano COS, Bilbao-Castro JR, Shkolnisky Y, Alcorlo M, Melero R, Caffarena-FernÃąndez G, Li M, Xu G, Marabini R, Carazo JM (2010) A clustering approach to multireference alignment of single-particle projections in electron microscopy. J Structural Biology 171:197–206

Index