Image biomarker standardisation initiative
version 1.4

Alex Zwanenburg   Stefan Leger   Martin Vallières   Steffen Löck

on behalf of the image biomarker standardisation initiative

20th July 2017

The image biomarker standardisation initiative

The image biomarker standardisation initiative (IBSI) is an independent international collaboration which works towards standardisation of image biomarkers. Reproducibility and validation of studies in quantitative image analysis and radiomics is a major challenge for the field (Gillies et al., 2015; Hatt et al., 2016; Yip and Aerts, 2016). The IBSI therefore aims to provide a common nomenclature for image biomarkers, common biomarkers definitions, benchmarks for image processing and feature extraction, as well as reporting guidelines.

Copyright

Contact

Dr. Alex Zwanenburg
alexander.zwanenburg@nct-dresden.de

IBSI collaborators

Mahmoud A. Abdalah	Department of cancer imaging and metabolism, Moffitt Cancer Center, Tampa (FL), USA
Hugo Aerts	Computational Imaging and Bioinformatics Laboratory, Dana-Farber Cancer Institute and Harvard Medical School, Harvard University, Cambridge (MA), USA
Aditya Apte	Department of medical physics, Memorial Sloan Kettering Cancer Center, New York (NY), USA
Saeed Ashrafinia	Department of radiology and radiological science, School of Medicine, Johns Hopkins University, Baltimore (MD), USA
Jorn Beukinga	Department of nuclear medicine and molecular imaging, University of Groningen, University Medical Center Groningen (UMCG), Groningen, the Netherlands
Ronald Boellaard	Department of nuclear medicine and molecular imaging, University of Groningen, University Medical Center Groningen (UMCG), Groningen, the Netherlands

Marta Bogowicz	Department of radiation oncology, University Hospital Zurich, University of Zurich, Switzerland
Luca Boldrini	Department of radiation oncology, Gemelli ART, Università Cattolica del Sacro Cuore, Rome, Italy
Marie-Charlotte Desseroit	Laboratory of medical information processing (LaTIM)–team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, University of Brest, IBSAM, Brest France
Nicola Dinapoli	Department of radiation oncology, Gemelli ART, Università Cattolica del Sacro Cuore, Rome, Italy
Cuong Viet Dinh	Imaging technology for radiation therapy group, the Netherlands Cancer Institute (NKI), Amsterdam, the Netherlands
Issam El Naqa	Medical physics unit, department of oncology, McGill University, Montreal, Canada
Andriy Y. Fedorov	Surgical Planning Laboratory, Brigham and Women's Hospital and Harvard Medical School, Harvard University, Cambridge (MA), USA
Nils Gählert	Medical and biological informatics, German Cancer Research Center (DKFZ), Heidelberg, Germany
Robert Gillies	Department of cancer imaging and metabolism, Moffitt Cancer Center, Tampa (FL), USA
Michael Götz	Medical and biological informatics, German Cancer Research Center (DKFZ), Heidelberg, Germany
Matthias Guckenberger	Department of radiation oncology, University Hospital Zurich, University of Zurich, Switzerland
Mathieu Hatt	Laboratory of medical information processing (LaTIM)–team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, University of Brest, IBSAM, Brest France
Fabian Isensee	Medical and biological informatics, German Cancer Research Center (DKFZ), Heidelberg, Germany
Philippe Lambin	Department of radiation oncology (MAASTRO), GROW–School for Oncology and Developmental Biology, Maastricht University Medical Centre+, Maastricht, the Netherlands
Stefan Leger	OncoRay–National Center for Radiation Research in Oncology, faculty of medicine and university hospital Carl Gustav Carus, Technische Universität Dresden, and Helmholtz-Zentrum Dresden-Rossendorf, Dresden, Germany
Ralph T.H. Leijenaar	Department of radiation oncology (MAASTRO), GROW–School for Oncology and Developmental Biology, Maastricht University Medical Centre+, Maastricht, the Netherlands
Jacopo Lenkowicz	Department of radiation oncology, Gemelli ART, Università Cattolica del Sacro Cuore, Rome, Italy
Fiona Lippert	Section for biomedical physics, department of radiation oncology, Universitätsklinikum Tübingen, Eberhard Karls University Tübingen, Germany
Steffen Löck	OncoRay–National Center for Radiation Research in Oncology, faculty of medicine and university hospital Carl Gustav Carus, Technische Universität Dresden, and Helmholtz-Zentrum Dresden-Rossendorf, Dresden, Germany

Are Losnegård	Department of clinical medicine, University of Bergen, Bergen, Norway
Klaus H. Maier-Hein	Medical and biological informatics, German Cancer Research Center (DKFZ), Heidelberg, Germany
Olivier Morin	Department of radiation oncology, University of California, San Francisco (CA), USA
Arman Rahmim	Department of radiology and radiological science, School of Medicine, Johns Hopkins University, Baltimore (MD), USA
Christian Richter	OncoRay–National Center for Radiation Research in Oncology, faculty of medicine and university hospital Carl Gustav Carus, Technische Universität Dresden, and Helmholtz-Zentrum Dresden-Rossendorf, Dresden, Germany
Nanna M. Sijtsema	Department of radiation oncology, University of Groningen, University Medical Center Groningen (UMCG), Groningen, The Netherlands
Jairo Socarras Fernandez	Section for biomedical physics, department of radiation oncology, Universitätsklinikum Tbingen, Eberhard Karls University Tübingen, Germany
Emiliano Spezi	Biomedical engineering research group, Cardiff School of Engineering, Cardiff University, Cardiff, United Kingdom
Roel J.H.M Steenbakkers	Department of radiation oncology, University of Groningen, University Medical Center Groningen (UMCG), Groningen, The Netherlands
Stephanie Tanadini-Lang	Department of radiation oncology, University Hospital Zurich, University of Zurich, Switzerland
Daniela Thorwarth	Section for biomedical physics, department of radiation oncology, Universitätsklinikum Tübingen, Eberhard Karls University Tübingen, Germany
Esther Troost	OncoRay–National Center for Radiation Research in Oncology, faculty of medicine and university hospital Carl Gustav Carus, Technische Universität Dresden, and Helmholtz-Zentrum Dresden-Rossendorf, Dresden, Germany
Taman Upadhaya	Laboratory of medical information processing (LaTIM)–team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, University of Brest, IBSAM, Brest France
Vincenzo Valentini	Department of radiation oncology, Gemelli ART, Università Cattolica del Sacro Cuore, Rome, Italy
Martin Vallières	Medical physics unit, McGill University, Montreal, Canada
Uulke van der Heide	Imaging technology for radiation therapy group, the Netherlands Cancer Institute (NKI), Amsterdam, the Netherlands
Lisanne V. van Dijk	Department of radiation oncology, University of Groningen, University Medical Center Groningen (UMCG), Groningen, The Netherlands
Floris H.P. van Velden	Department of radiology, Leiden University Medical Center (LUMC), Leiden, the Netherlands
Joost van Griethuysen	Department of radiology, the Netherlands Cancer Institute (NKI), Amsterdam, the Netherlands
Philip Whybra	Biomedical engineering research group, Cardiff School of Engineering, Cardiff University, Cardiff, United Kingdom
Alex Zwanenburg	National Center for Tumor Diseases (NCT), partner site Dresden, Germany

Table 1: Alphabetical list of IBSI collaborators.

Contents

Chapter 1

On this document

While analysis of medical images has practically taken place since the first image was recorded, high throughput analysis of medical images is a more recent phenomenon (Kumar et al., 2012; Lambin et al., 2012; Gillies et al., 2015). The aim of such a radiomics process is to provide decision support based on medical imaging. Part of the radiomics process is the conversion of image data into numerical features which capture different medical image aspects, and can be subsequently correlated as biomarkers to e.g. expected oncological treatment outcome.

With the growth of the radiomics field, it has become clear that results are often difficult to reproduce, that standards for image processing and feature extraction are missing, and that reporting guidelines are absent (Gillies et al., 2015; Hatt et al., 2015; Yip and Aerts, 2016). The image biomarker standardisation initiative (IBSI) seeks to address these issues. The current document provides definitions for a large number of image features, as well as for basic image processing steps.

The definitions presented in the document may furthermore contain references to digital phantoms and other test data. These are used by IBSI to standardise feature implementations and image processing schemes. These data sets and corresponding standardised feature values will be made publicly available at a later time point.

Chapter 2

Document changes

The following is a list of changes to the document from previous versions.

Version 1.4 - current

- Removed spatial filtering from the image processing section. This may be investigated at a later time.

- Swapped position of distance zone matrix and neighbourhood grey tone difference matrix (NGTDM) sections.

- Added note on NGTDM voxels without any valid neighbours.

- Changed font type from *Computer Modern Roman* to *New Century Schoolbook* for better readability.

- Added chapter on image processing from the IBSI internal work document.

- Reintroduced *approximate volume* for consistency with common clinical volume calculation.

- Updated description of intensity-volume histogram feature family.

- Updated section on novel and uncommon imaging features.

Version 1.3

- Updated and revised definitions of morphological features to handle inconsistencies due to mixed voxel representations. Notably *volume* is now calculated from the ROI mesh instead of directly from the ROI voxels. Likewise, *diameter*, *volume density* and *area density* features are (where possible) based on vertices of the ROI mesh, instead of the voxel centre point set. The introductory description of the morphological feature set was updated to introduce nomenclature and definitions used in meshes.

- Added *approximate volume* feature, which captures tumour volume based on voxel counts.

- Added figures to show feature calculation process for texture features.

- Clarified definitions for *Neighbourhood grey tone difference matrix*-based features.

Version 1.2

- Updated the section on *Intensity volume histogram*-based features.

- Updated description for the local intensity feature set.

- Updated the description of the *area density - approximate enclosing ellipsoid* to provide a maximum degree for the Legendre polynomials.

- Updated the description of the *volume density - minimum volume enclosing ellipsoid* to provide a stopping tolerance for Khachiyan's method.

- Feature sets are ordered differently. The morphological features are now presented first, followed by local intensity features and statistical features.

- Fixed various typos.

Version 1.1

- Fixed errors in the definition of *dependence count energy* feature.

- Fixed errors in the definitions of *major axis length*, *minor axis length* and *least axis length* features.

- Updated descriptions of the *volume density - approximate enclosing ellipsoid* and *area density - approximate enclosing ellipsoid* features.

- Added descriptions of the *volume density - minimum volume enclosing ellipsoid* and *area density - minimum volume enclosing ellipsoid* features.

- Fixed errors in layout.

Chapter 3

Image processing

Image processing is the sequence of operations required to derive image biomarkers (features) from images. In this work we focus on the extraction of features from a region of interest (ROI) segmented in an image.

Image processing may be conducted using a wide variety of schemes. We therefore designed a general image processing scheme for image feature calculation based on various schemes used within scientific literature. The image processing scheme is shown in figure 3.1. The processing steps referenced in the figure are described below.

3.1 Image processing elements

The process leading from a medical image to a set or overlay of image features can be viewed as a sequence of operations on the image data, such as segmentation, interpolation and discretisation. Our starting point is the image stack as read from e.g. a series of DICOM or NIfTI files. Various other sources of variability exist, such as acquisition protocol, scanner type, quantitative corrections and image reconstruction algorithms. Since these are often vendor-specific, and/or standardised elsewhere e.g. (Boellaard et al., 2015), these will not be treated here.

3.1.1 Data conversion

Some imaging modalities, notably positron emission tomography (PET), require conversion of image data into a more meaningful presentation.

SUV Activity due to uptake of the PET tracer fluid is converted to a standardised uptake value (SUV) in each voxel. SUV is commonly defined as (Boellaard et al., 2010):

$$SUV_i = \frac{A_i}{A_{adm}/BM}$$

Here, A_i is the activity in kBq/ml in voxel i measured by PET. A_{adm} is the administered activity in MBq, corrected for activity loss due to time between administration and image acquisition and tracer losses during administration. Thus

$$A_{adm} = \left(A_0 - A_{loss} \, (1/2)^{-\frac{\Delta t_{loss}}{t_{1/2}}} \right) (1/2)^{\frac{\Delta t_{acq}}{t_{1/2}}}$$
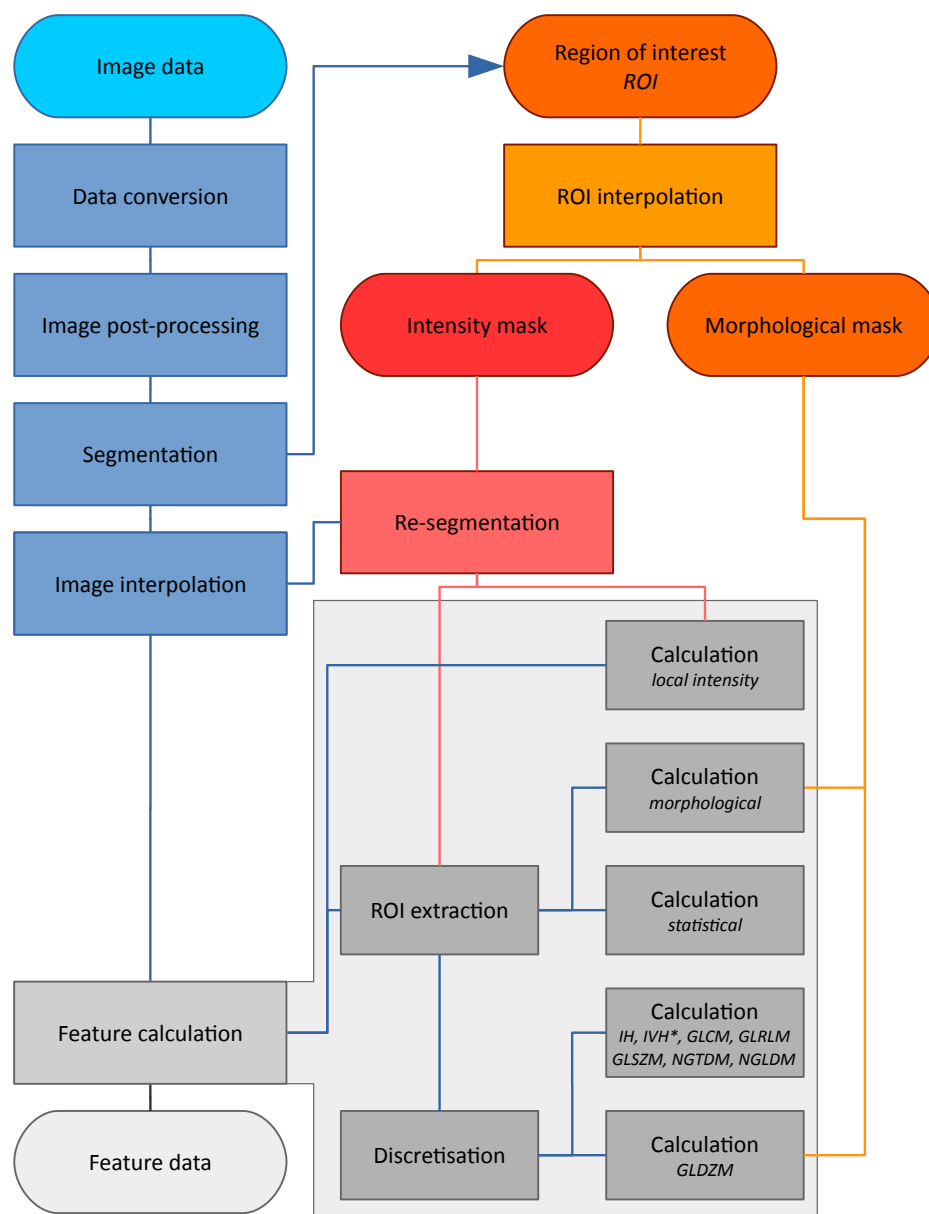
Figure 3.1: Image processing scheme for image biomarker calculation. Depending on the specific imaging modality and purpose, some steps may be omitted. The region of interest (ROI) is explicitly split into two masks, namely an intensity and morphological mask, after interpolation to the same grid as the interpolated image. Feature calculation is expanded to show the different feature families with specific pre-processing. IH: intensity histogram; IVH: intensity volume histogram; GLCM: grey level cooccurrence matrix; GLRLM: grey level run length matrix; GLSZM: grey level size zone matrix; NGTDM: neighbourhood grey tone difference matrix; NGLDM: Neighbouring grey level dependence matrix; GLDZM: grey level distance zone matrix; *Discretisation of IVH differs from IH and texture features, see section 4.5.

A_0 is the initially measured activity in MBq. A_{loss} is the loss in activity due to administration, measured at t_{loss}. Hence the first factor represent the activity of the injected volume at time 0. Δt_{acq} is the time between initial activity measurement and image acquisition and $t_{1/2}$ the half life of the tracer isoptope. BM is the patients body mass in kg.

A plasma glucose correction to SUV may be applied (Boellaard et al., 2010):

$$SUV_{gluc,i} = SUV_i \, \frac{C_{gluc,plasma}}{5.0}$$

Here, $C_{gluc,plasma}$ is the glucose concentration in the plasma in mmol/l.

SUL (Boellaard et al., 2015) recommended additional calculation of standard uptake value normalised to lean body mass (SUL). SUL is defined as:

$$SUL_i = \frac{A_i}{A_{adm}/LBM}$$

Lean body mass (LBM) is defined as (Janmahasatian et al., 2005):

$$LBM = \begin{cases} \dfrac{9270\,BM}{6680 + 216\,(BM/h^2)} & \text{if male} \\[2ex] \dfrac{9270\,BM}{8780 + 244\,(BM/h^2)} & \text{if female} \end{cases}$$

h is the patient height in meters. The factor BM/h^2 is thus equal to the body mass index (BMI) of the patient.

As with SUV, a plasma glucose may be applied:

$$SUL_{gluc,i} = SUL_i \, \frac{C_{gluc,plasma}}{5.0}$$

3.1.2 Image post-processing

Images are post-processed to enhance image quality. For instance, MRI contains both Gaussian and Rician noise (Gudbjartsson and Patz, 1995) and may benefit from denoising. As another example intensities measured using MR may be non-uniform across an image and require correction (Sled et al., 1998). FDG-PET-based may furthermore be corrected for partial volume effects (Soret et al., 2007; Boussion et al., 2009). In CT imaging metal objects, e.g. pacemakers, and tooth implants, introduce artifacts and may require artifact suppression (Gjesteby et al., 2016). Evaluation and standardisation of various post-processing methods falls outside the scope of the current work.

3.1.3 Segmentation

Medical image analysis, within the current paradigm, relies on the definition of regions of interest. These ROI represent anatomical structures, such as a tumour volume or organs at risk, and are used to define areas or volumes from which image features are calculated. ROI can be defined manually by experts or automatically by algorithms. A review of the numerous segmentation methods and their merits is beyond the scope of this work.

From a process point-of-view, segmentation leads to the creation of a ROI mask $\mathbf{R}$, which is defined as:

$$R_j = \begin{cases} 1 & j \text{ in ROI} \\ 0 & \text{otherwise} \end{cases}$$

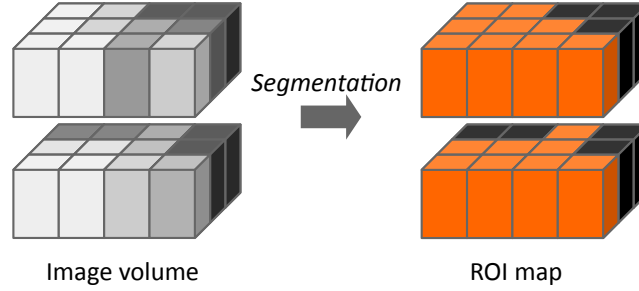An example is shown in figure 3.2.

Segmentation

Image volume   ROI map

Figure 3.2: Segmentation of the image volume leads to the creation of a region of interest (ROI). In this example, segmentation is performed by excluding voxels with $> 50\%$ grey from the ROI. All orange voxels in the ROI are considered to be part of the ROI, while black voxels are excluded.

ROIs are typically saved with the correpsonding image. Some image formats directly store ROI masks as voxels (e.g. NIfTI and NRRD), and generating the ROI mask is conducted by loading the corresponding image. In other cases the ROI is defined by a closed set of (planar) polygons, for example within a DICOM RTSTRUCT file. Defining the ROI mask then consists of determining which voxel centers lie within the space enclosed by the contour polygon in each slice.

A common method to determine whether a point lies in a 2D polygon is the *crossing number* algorithm, for which a number of implementations exist (Schirra, 2008). The main concept of this algorithm is that for any point inside the polygon any line originating outside the polygon will cross the polygon an uneven number of times. An simple example is shown in Figure 3.3. The example implementation makes use of the fact that the ROI mask is a regular grid to scan entire rows or columns at a time. The contour polygon is considered to be composed of a connected set of lines. The simple implementation consist of the following steps:

1. For each row j, find those polygon edges whose y-component of the vertices do not both lie on the same side of the row coordinate y_j. This step is used to limit calculation of intersection points to only those that cross a ray cast from outside the polygon (e.g. ray with origin $(-1, y_j)$ and direction $(1, 0)$). This an optional step.

2. Determine intersection points x_i of the polygon edges with the ray.

3. Iterate over intersection points, and add 1 to the count of each voxel center with $x \geq x_i$.

4. Apply even-odd rule. Voxels with an odd count are inside the polygon, whereas voxels with an even count are outside.

Note that the example represents a relatively naive implementation that will not consistently assign voxel centers positioned on the polygon itself to the interior.
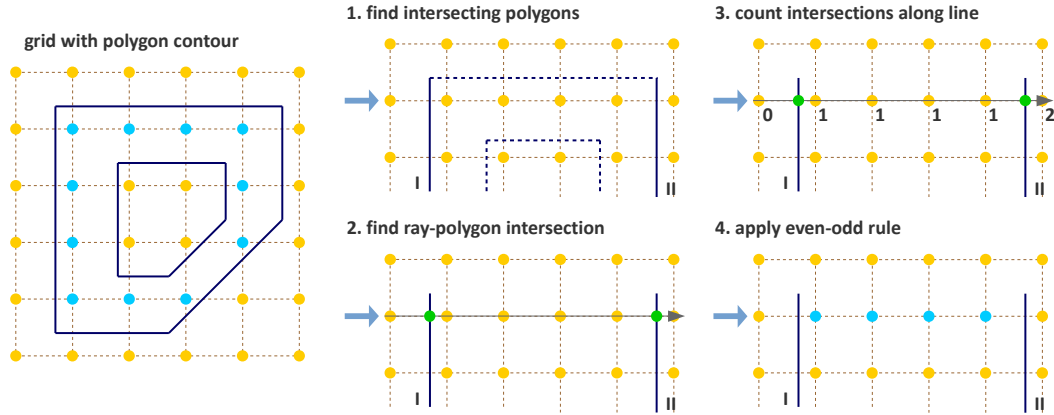
Figure 3.3: Simple algorithm for determining which voxels are inside a polygon. The suggested implementation consists of four steps: (1) Omit edges that will not intersect with the current row of voxel centers. (2) Calculate intersection points of edges I and II with the ray for the current row. (3) Determine the number of intersections crossed from ray origin to the row voxel centers. (4) Apply even-odd rule to determine whether voxel centers are inside the polygon.

3.1.4 Voxel interpolation

Texture feature sets require interpolation to isotropic voxel sizes to be rotationally invariant, and to allow comparison between image data from different patients. Voxel interpolation affects image feature values as many image features are sensitive to changes in voxel size (Yan et al., 2015; Shafiq-ul Hassan et al., 2017). Maintaining consistent isotropic voxel dimensions across different patients and institutions is therefore important for reproducibility. The isotropic voxel dimension depends on the imaging modality used for image acquisition. At the moment there are no clear indications whether up-sampling or down-sampling schemes are preferable. Consider, for example, an image stack of slices with $1.0 \times 1.0 \times 3.0$mm voxels. Up-sampling to the largest dimension ($3.0 \times 3.0 \times 3.0$mm) incurs some information loss, while down-sampling ($1.0 \times 1.0 \times 1.0$mm) requires inference and introduces artificial information.

While in general we consider three-dimensional (3D) interpolation for an image stack, two-dimensional (2D) voxel interpolation within the image slice plane may be recommended in some situations. In 2D voxels are not interpolated between slices. For example, if slice thickness is large compared to in-plane voxel dimensions, a 2D approach is expected to be preferable. Otherwise applying 3D interpolation either requires inferencing a large number of voxels between slices, or the loss of a large fraction of in-plane information.

Interpolation algorithms Interpolation algorithms determine the grey level values in the interpolation grid after interpolation of the original grid. In these grids, voxels are spatially represented by their center. A number of algorithms are available for interpolation. Common choices are *nearest neighbour*, *trilinear*, *tricubic convolution* and *tricubic spline interpolation*. In short, *nearest neighbour interpolation* assigns grey levels from the closest voxels in the input grid to the output grid. *Trilinear interpolation* uses the eight closest voxels in the input grid to calculate the new grey level value using linear interpolation.

Tricubic convolution and *tricubic spline interpolation* draw upon a larger neighbourhood to evaluate a smooth, continuous third-order polynomial at the points of the output grid. The difference between *tricubic convolution* and *tricubic spline interpolation* lies in the implementation. Whereas *tricubic spline interpolation* evaluates the smooth and continuous third-order polynomial at every point, *tricubic convolution* approximates the solution using a convolution filter. Though *tricubic convolution* is faster, with modern hardware and common image sizes, the difference in execution speed is practically meaningless. Both interpolation algorithm produce similar results, and both are often referred to as *tricubic interpolation*.

While no consensus exists concerning the optimal choice of interpolation algorithm, *trilinear interpolation* is a conservative choice. It does not lead to the blockiness produced by *nearest neighbour interpolation*, and does not lead to out-of-range grey values which may occur due to overshoot with *tricubic* and higher order interpolations. The latter problem can occur in acute transitions in grey levels, where the local neighbourhood itself is not sufficiently smooth to evaluate the polynomial within the allowed range. *Tricubic* methods, however, produce smoother images and may produce less interpolation artifacts.

Grey values after interpolation may require rounding, or application of cut-off values. For example, in CT images grey levels represent Hounsfield units, and do not take non-integer values. Such interpolated CT grey levels are then rounded to the nearest integer. Such operations are part of voxel interpolation.

Voxel interpolation also requires interpolation of the ROI mask $\mathbf{R}$ to the same dimensions. Interpolation of ROI mask is best conducted using either the *nearest neighbour* and *trilinear interpolation* methods as these are guaranteed to produce meaningful masks. After *trilinear interpolation* voxels are assigned to the ROI if they contain at least fraction δ of the original ROI:

$$R_j = \begin{cases} 1 & R_{interp,j} \geq \delta \\ 0 & R_{interp,j} < \delta \end{cases}$$

A reasonable choice for the partial volume threshold is $\delta = 0.5$. For *nearest neighbour interpolation* the ROI mask should not contain fractions, and may be used in its entirety.

Interpolation grid   Interpolated voxel centers lie on a regular mesh grid. The choice of mesh grid alignment to the original voxel grid affects feature values due to different voxels being included in the ROI. Three common alignment grids may be identified, and are shown in Figure 3.4:

1. **Fit to original grid** In this case the interpolation mesh grid is deformed so that the voxel centers at the grid corners overlap with the corners of the original grid. For an original 4×4 voxel grid with spacing $(3.00, 3.00)$ and a desired interpolation spacing of $(2.00, 2.00)$ we first calculate the extent of the original voxel grid in world coordinates leading to an extent of $((4-1)\,3.00, ((4-1)\,3.00) = (9.00, 9.00)$. In this case the interpolated grid will not exactly fit the original grid. Therefore we try to find the closest fitting grid, which leads to a 6×6 grid by rounding up $(9.00/2.00, 9.00/2.00)$. The resulting voxel grid has a grid spacing of $(1.80, 1.80)$ in world coordinates and $(0.60, 0.60)$ in grid coordinates.

2. **Align grid origins** A simple approach which keeps the required grid spacing is alignment of the origins of the interpolation and original grids. One drawback is that the definition of the origin may differ between programming languages. Keeping with the

example, the interpolation grid is (6×6). The resulting voxel grid has a grid spacing of $(2.00, 2.00)$ in world coordinates and $(0.67, 0.67)$ in grid coordinates. By definition both grids are aligned at the origin, $(0.00, 0.00)$.

3. **Align grid centers** An implementation-independent solution is to align both grids on the grid center. Again, keeping with the example, the interpolation grid is (6×6). Again, the resulting voxel grid has a grid spacing of $(2.00, 2.00)$ in world coordinates and $(0.67, 0.67)$ in grid coordinates.

Alignment by center is preferred as it is implementation-independent and maintains voxel spacing. Below is a description of how to implement alignment by center.

Determining interpolation grid size The grid size of the interpolation grid can be determined as follows. Let n_a be the number of points in the original grid in a single dimension and $s_{a,w}$ their spacing in world coordinates. $s_{b,w}$ is the desired spacing for the particular dimension. The interpolation grid size is then:

$$n_b = \lceil \frac{n_a s_a}{s_b} \rceil$$

Rounding towards infinity offers the guarantee that the interpolation grid exists even when the original grid contains but few voxels. However, it also means that a part of the grid points falls outside the original grid and is extrapolated. Padding the original grid with the voxel values at the borders is recommended. Some algorithm implementations may perform padding internally.

Determining interpolation grid position For the *align grid centers* method, the position of the interpolation grid points is determined as follows. Let n_a and n_b be the number of points in the original and interpolation grid respectively in a single dimension. Let moreover $s_{a,w}$ be the original spacing and $s_{b,w}$ the desired spacing in the same dimension in world coordinates. When $x_{a,w}$ is the origin of the original grid in world coordinates, the origin of the interpolation grid is located at:

$$x_{b,w} = x_{a,w} + \frac{s_a(n_a - 1) - s_b(n_b - 1)}{2}$$

In the grid coordinate system, the original grid origin is located at $x_{a,g} = 0$. The origin of the interpolation grid is then located at:

$$x_{b,g} = \frac{1}{2} \left(n_a - 1 - \frac{s_{b,w}}{s_{a,w}} (n_b - 1) \right)$$

Here the fraction $s_{b,w}/s_{a,w} = s_{b,g}$ is the desired spacing in grid coordinates. In one dimension the interpolation grid points are located at grid coordinates:

$$x_{b,g}, \ x_{b,g} + s_{b,g}, \ x_{b,g} + 2s_{b,g}, \ \ldots, \ x_{b,g} + (n_b - 1)s_{b,g}$$

The above description extends to every dimension of the grid.

original grid

Original grid
Size: 4x4 points
Spacing: (3.00, 3.00)

fit to original grid

Interpolation grid
Size: 6x6 points
Desired spacing: (2.00, 2.00)
Realised spacing: (1.80, 1.80)

align grid origins

Interpolation grid
Size: 6x6 points
Desired spacing: (2.00, 2.00)
Realised spacing: (2.00, 2.00)

align grid centers

Interpolation grid
Size: 6x6 points
Desired spacing: (2.00, 2.00)
Realised spacing: (2.00, 2.00)

Figure 3.4: Different interpolation mesh grids based on an original 4×4 grid with $(3.00, 3.00)$ spacing. The desired interpolation spacing is $(2.00, 2.00)$. *Fit to original grid* creates an interpolation mesh grid that overlaps with the corners of the original grid. *Align grid origins* creates an interpolation mesh grid that is positioned at the origin of the original grid. *Align grid centers* creates an interpolation grid that is centred on the the center of original and interpolation grids.
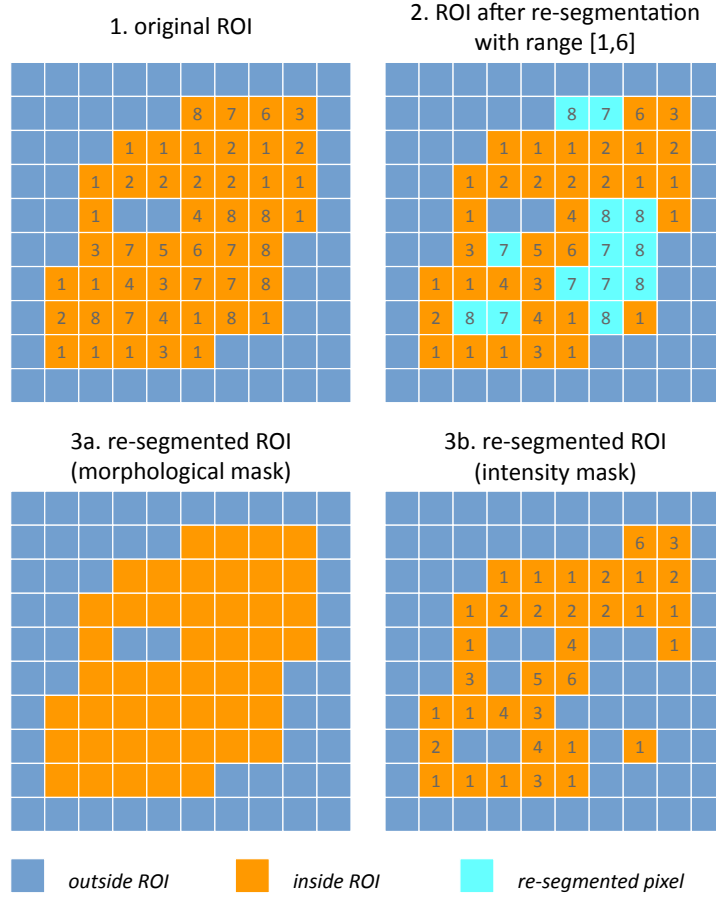
Figure 3.5: Example showing how intensity and morphological masks may differ due to re-segmentation. (1) The original region of interest (ROI) is shown with pixel intensities. (2) Subsequently, the ROI is re-segmented to only contain values in the range $[1, 6]$. Pixels outside this range are marked for removal from the intensity mask. (3a) Resulting morphological mask, which is identical to the original ROI. (3b) Re-segmented intensity mask. Note that due to re-segmentation intensity and morphological masks are different.

3.1.5   Re-segmentation

Re-segmentation entails updating the ROI mask $\mathbf{R}$ based on corresponding image volume voxel grey level intensities $\mathbf{X}_{gl}$. Re-segmentation may be performed to exclude voxels from a previously segmented ROI, and is usually performed after interpolation.

Intensity and morphological masks of a ROI   A ROI consists of an intensity mask and a morphological mask. For many feature families only the intensity mask is important. However, for morphological and grey level distance zone matrix (GLDZM) feature families, both intensity and morphological masks are used. Normally both masks are identical, but re-segmentation updates intensity masks. For example, limiting the ROI to a prescribed

range of voxel intensities may remove voxels from the ROI. The intensity mask is updated to only contain ROI voxels within the prescribed range. The morphological mask remains unchanged by re-segmentation. Any changes to the morphological mask should be made prior to re-segmentation as part of the segmentation process. A two-dimensional example is shown in Figure 3.5.

Range re-segmentation Re-segmentation may be performed to remove voxels from the intensity mask that fall outside of a specified range. An example is the exclusion of voxels containing air or bone tissue in the tumour ROI within CT images, or low activity areas in PET images. Such a range is usually presented as a closed interval $[a, b]$ or half-open interval $[a, \infty)$.

The intensity threshold range should be propagated to feature families that require a specified intensity range (e.g. intensity-volume histogram features), as well as *fixed bin size* discretisation.

Intensity outlier filtering ROI voxels with outlier intensities may be removed from the intensity mask. One method for defining outliers was suggested by Vallières et al. (2015) after Collewet et al. (2004). The mean μ and standard deviation σ of grey levels of voxels assigned to the ROI are calculated. Subsequently voxels outside the range $[\mu - 3\sigma, \mu + 3\sigma]$ are excluded from the intensity mask.

3.1.6 ROI extraction

Many feature families require that the ROI is isolated from the surrounding voxels. The ROI intensity mask is used to extract the image volume to be studied. Excluded voxels are commonly replaced by a placeholder value, often *NaN*. This placeholder value is then used to exclude these voxels from calculations. Voxels included in the ROI mask naturally retain their original grey level value.

3.1.7 Grey level discretisation

Grey level discretisation or quantisation of the ROI is often required to make calculation of textural features tractable (Yip and Aerts, 2016), and has noise-suppressing properties as well. An example of discretisation is shown in Figure 3.6.

Two approaches to discretisation are commonly used. One involves the discretisation to a fixed number of bins, and the other discretisation with a fixed bin width. As we will observe, there is no inherent preference for one or the other method. However both methods have particular characteristics, which are described below, which may make them better suited for some tasks. Note that the lowest bin always has value 1 instead of 0. This ensures consistency for calculations of textural features, where grey level 0 is not allowed for some features.

Fixed bin number In the *fixed bin number* method, grey levels X_{gl} are discretised to a fixed number of N_b bins. It is defined as follows:

$$X_{b,i} = \begin{cases} 1 & X_{gl,i} = X_{gl,min} \\ \lceil N_b \frac{X_{gl,i} - X_{gl,min}}{X_{gl,max} - X_{gl,min}} \rceil & X_{gl,i} > X_{gl,min} \end{cases}$$
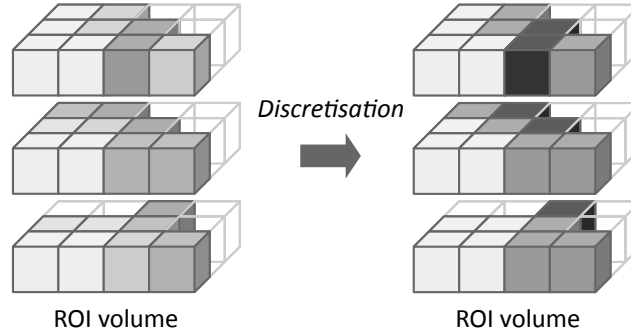
Figure 3.6: The image volume contained in the region of interest (ROI) is discretised. Here, grey levels from the original ROI volume were assigned to 3 bins to create a discretised volume.

In short, the grey level $X_{gl,i}$ of voxel i is corrected by the lowest occurring grey level $X_{gl,min}$ in the ROI and divided by the bin width $(X_{gl,max} - X_{gl,min})/N_b$, and subsequently rounded up to the nearest integer.

The *fixed bin number* method loses the relationship between image intensity and physiological meaning. However, it introduces a normalising effect which may be beneficial when the imaging modality cannot be well-calibrated, but where contrast is important. The *fixed bin number* method maintains contrast, and increases comparability between images recorded in different patients.

The effect of the number of bins for *fixed bin number* discretisation was studied by Hatt et al. (2015), in a large methodological study with 555 pretreatment FDG-PET images covering a range of different tumours. They found that *fixed bin number* discretisation using 64 bins provides the best compromise between differentiation and resolution.

In another methodological study van Velden et al. (2016) studied the effect of *fixed bin number* versus *fixed bin size* methods. They concluded that textural features from FDG-PET images had better repeatability and lower sensitivity to delineation changes using the *fixed bin size* discretisation method.

Fixed bin size   *Fixed bin size* discretisation is conceptually simple. A new bin is assigned every w_b grey levels, starting at a minimum $X_{gl,min}$. The minimum may be user-set value, for example deriving from re-segmentation, or data-driven as $X_{gl,min} = \min(X_{gl})$ for the intensities in the ROI. To maintain consistency between subjects it is recommended that the user provides the minimum grey level whenever possible. Discretisation then proceeds as follows:

$$X_{b,i} = \begin{cases} 1 & X_{gl,i} = X_{gl,min} \\ \lceil \frac{X_{gl,i} - X_{gl,min}}{w_b} \rceil & X_{gl,i} > X_{gl,min} \end{cases}$$

The *fixed bin size* method has the advantage of maintaining a direct relationship with the original scale, which is useful for well-calibrated imaging modalities such as CT.

Leijenaar et al. (2015) compared the effect of *fixed bin size* and *fixed bin number* discretisation methods on textural features from FDG-PET images recorded in a cohort of 35 non-small cell lung cancer patients. They concluded that *fixed bin size* may be more appropriate for inter- and intra-patient comparison of textural feature values in a clinical

setting. In addition, van Velden et al. (2016) concluded that *fixed bin size* demonstrated higher repeatability and less sensitivity to delineation changes, as noted above.

It should also be noted that several studies have used *fixed bin size* for CT images, e.g. (Aerts et al., 2014; van Dijk et al., 2017). Both studies used a bin size of 25 HU, but did not report on minimum grey level.

Other methods Many other methods and variations for discretisation exist. Vallières et al. (2015) describe the use of *intensity histogram equalisation* and *Lloyd-Max* algorithms for discretisation. *Intensity histogram equalisation* involves redistributing grey levels so that the resulting bins contain a similar number of voxels, i.e. the histogram is as flat as possible (Hall et al., 1971). The *Lloyd-Max* algorithm is a clustering method that seeks to minimise discretisation errors (Max, 1960; Lloyd, 1982).

3.1.8 Feature calculation

Feature calculation is the processing step where single features are calculated from a ROI as descriptors for said volume. Feature calculation is handled in full details in the next chapter.

Chapter 4

Image features

In this chapter we will describe a set of quantitative image features. The set of features largely builds upon the feature sets proposed by Aerts et al. (2014) and Hatt et al. (2016), which are themselves largely derived from earlier work. References to earlier work are provided where possible.

The set of features can be divided into a number of families, of which statistical, intensity histogram-based, intensity-volume histogram-based, morphological features, local intensity, and textural features are treated here. These are some of the most commonly used features. Note however that many other features also exist, but are not described here.

Features are calculated on the base image, as well as any image transformations. For the definitions, it assumed that an image segmentation mask, identifying the voxels corresponding to the region of interest (ROI), exists. The ROI itself consists of two masks, an intensity mask and a morphological mask. These masks may be identical, but are not necessarily so. Image processing may lead to the exclusion of voxels from the intensity mask, based on their voxel intensity. The intensity mask is used in the calculation of most features. The morphological mask is specifically used in the calculation of morphological features and distance zone matrices.

The required image processing steps before feature calculated depend on the particular feature family. Most feature families require that voxels outside of the ROI intensity mask are removed from the analysis volume, and replaced by a placeholder value such as NaN. Several feature families also require prior discretisation of grey levels into grey level bins, notably intensity histogram and textural features. Other feature families do not require discretisation before calculations. For more details, see figure 3.1 in the previous chapter on image processing.

4.1   Morphological features

Morphological features describe geometric aspects of a region of interest (ROI), such as area and volume. Morphological features are based on ROI voxel representations of the volume. Three voxel representations of the volume are conceivable. First, the volume can be represented by a stack of voxels with each taking up a certain volume. Second, the voxel point set $\mathbf{X}_c$ consisting of coordinates of the voxel centers may be used. The third option is representation of the outer structure of the volume by mesh vertices and faces. We use the

second representation when the inner structure of the volume is important, and the third representation when only the outer surface structure is important. The first representation is not used because it does handle partial volume effects at the ROI edge well, and also to avoid inconsistencies in feature values introduced by mixing representations in small voxel volumes.

Mesh-based representation The surface of the ROI volume can be translated into a triangle mesh using a meshing algorithm. While there are multiple meshing algorithms avaliable, we suggest the use of the *Marching Cubes* algorithm (Lorensen and Cline, 1987; Lewiner et al., 2003) because of its widespread availability for different programming languages and reasonable approximation of the surface area and volume (Stelldinger et al., 2007). In practice, mesh-derived feature values depend upon the meshing algorithm used and small differences may occur.
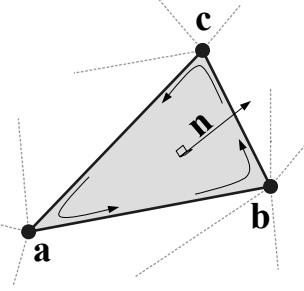
Figure 4.1: Meshing algorithms draw faces and vertices to cover the ROI. One face, spanned by vertices $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$, is highlighted. Moreover, the vertices define the three edges $\mathbf{ab} = \mathbf{b} - \mathbf{a}$, $\mathbf{bc} = \mathbf{c} - \mathbf{b}$ and $\mathbf{ca} = \mathbf{a} - \mathbf{c}$. The face normal $\mathbf{n}$ is determined using the right-hand rule, and calculated as $\mathbf{n} = (\mathbf{ab} \times \mathbf{bc}) / \|\mathbf{ab} \times \mathbf{bc}\|$, i.e. the outer product of edge $\mathbf{ab}$ with edge $\mathbf{bc}$, normalised by its length.

Meshing algorithms use the ROI voxel point set $\mathbf{X}_c$ to create a closed mesh. Dependent on the algorithm, a parameter is required to specify where the mesh should be drawn. A default level of 0.5 times the voxel spacing is used for marching cube algorithms. Alternatively a value of 0.5 can be used for isosurface algorithms, when the ROI mask consists of 0 and 1 values. Depending on implementation, isosurface algorithms may also require padding of the ROI mask with non-ROI voxels to correctly estimate the isosurface contours in places where ROI voxels would otherwise be located at the edge of the mask.

The closed mesh drawn by the meshing algorithm consists of N_{fc} triangle faces spanned by N_{vx} vertex points, see Figure 4.1. The set of vertex points is then $\mathbf{X}_{vx}$.

The calculation of the mesh volume requires that all faces have the same orientation of the face normal. Consistent orientation can be checked by the fact that in a regular, closed mesh, all edges are shared between exactly two faces. Given the edge spanned by vertices $\mathbf{a}$ and $\mathbf{b}$, the edge must be $\mathbf{ab} = \mathbf{b} - \mathbf{a}$ for one face and $\mathbf{ba} = \mathbf{a} - \mathbf{b}$ for the adjacent face. This ensures consistent application of the right-hand rule, and thus consistent orientation of the face normals. Algorithm implementations may return consistently orientated faces by default.

ROI morphological and intensity masks The ROI consists of a morphological and an intensity mask. The intensity mask may differ from the morphological mask due to excluded voxels with out-of-range intensities. The morphological mask is used to calculate many of the morphological features, and is used to generate the voxel point set $\mathbf{X}_c$. Any holes within the morphological map are understood to be the result of segmentation decisions, and thus to be intentional. The intensity mask is used to generate the voxel intensity set $\mathbf{X}_{gl}$ with corresponding point set $\mathbf{X}_{c,gl}$. For the digital phantom both masks are identical. For the cases based on patient data, however, the masks differ due to resegmentation of the intensity mask.

4.1.1 Volume

The volume V is calculated from the ROI mesh as follows (Zhang and Chen, 2001). Each face can be viewed as forming a tetrahedron with the origin as the remaining vertex. By placing the origin at $(0, 0, 0)$, the signed volume of each tetrahedron formed by face k and the origin is:

$$V_k = \frac{\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})}{6}$$

Here $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ are the vertex points of face k. Depending on the orientation of the normal, the signed volume may be positive or negative. Hence, the orientation of face normals should be consistent, e.g. all normals must be either pointing outward or inward. The volume V is then calculated by summing over the face volumes, and taking the absolute value:

$$V = \left| \sum_{k=1}^{N_{fc}} V_k \right|$$

For positron emission tomography *volume* is equivalent to the *metabolically active tumour volume* (MATV).

4.1.2 Approximate volume

In clinical practice, volumes are commonly determined on voxel counts. For larger volumes, the differences between *approximate volume* and mesh-based volume are usually negligible. However for small volumes with a low number of voxels *approximate volume* will overestimate volume compared to mesh-based volume. It is therefore only used as a reference feature, and not in the calculation of other morphological features.

Approximate volume is defined as:

$$F_{morph.approx.vol} = \sum_{j=1}^{N_v} V_j$$

Here N_v is the number of voxels in the morphological mask of the ROI, and V_j the volume of voxel j.

4.1.3 Surface area

The area A is also calculated from the ROI mesh, by summing over the face area surfaces. The area of face k is then:

$$A_k = \frac{|\mathbf{ab} \times \mathbf{ac}|}{2}$$

Here $\mathbf{ab} = \mathbf{b} - \mathbf{a}$ is the vector from vertex $\mathbf{a}$ to vertex $\mathbf{b}$, and $\mathbf{ac} = \mathbf{c} - \mathbf{a}$ the vector from vertex $\mathbf{a}$ to vertex $\mathbf{c}$. The total surface area A is then:

$$A = \sum_{k=1}^{N_{fc}} A_k$$

4.1.4 Surface to volume ratio

The surface to volume ratio is given as:

$$F_{morph.av} = \frac{A}{V}$$

4.1.5 Compactness 1

The following are several features describing the deviation of the ROI volume from a sphere. All these definitions can be derived from one another. As a results these features are are highly correlated and may thus redundant. Compactness 1 is a measure for how compact, or sphere-like the volume is. Compactness 1 is defined as:

$$F_{morph.comp.1} = \frac{V}{\pi^{1/2} A^{3/2}}$$

Some definitions use $A^{2/3}$ instead of $A^{3/2}$, e.g. (Aerts et al., 2014). This is most likely an error, as the feature would no longer be dimensionless.

4.1.6 Compactness 2

Compactness 2 is another measure to describe how sphere-like the volume is:

$$F_{morph.comp.2} = 36\pi \frac{V^2}{A^3}$$

By definition $F_{morph.comp.1} = 1/6\pi \left(F_{morph.comp.2} \right)^{1/2}$.

4.1.7 Spherical disproportion

Spherical disproportion is another measure to describe how sphere-like the volume is:

$$F_{morph.sph.dispr} = \frac{A}{4\pi R^2} = \frac{A}{(36\pi V^2)^{1/3}}$$

By definition $F_{morph.sph.dispr} = \left(F_{morph.comp.2} \right)^{-1/3}$.

4.1.8 Sphericity

Sphericity is a further measure to describe how sphere-like the volume is:

$$F_{morph.sphericity} = \frac{\left(36\pi V^2\right)^{1/3}}{A}$$

By definition $F_{morph.sphericity} = \left(F_{morph.comp.2}\right)^{1/3}$.

4.1.9 Asphericity

Asphericity describes how much the ROI deviates from a perfect sphere. Asphericity is defined as:

$$F_{morph.asphericity} = \left(\frac{1}{36\pi} \frac{A^3}{V^2}\right)^{1/3} - 1$$

By definition $F_{morph.asphericity} = \left(F_{morph.comp.2}\right)^{-1/3} - 1$

4.1.10 Centre of mass shift

The distance between the ROI volume centroid and the intensity-weighted ROI volume centroid measures the placement of high and low intensity regions within the volume. Let $N_{v,m}$ be the number of voxels in the morphological mask. The ROI volume centre of mass is calculated from the ROI voxel point set $\mathbf{X}_c$ as follows:

$$\overrightarrow{CoM}_{geom} = \frac{1}{N_{v,m}} \sum_{i=1}^{N_{v,m}} \vec{X}_{c,i}$$

The intensity-weighted ROI volume is based on the intensity mask. The position of each voxel centre in $\mathbf{X}_{c,gl}$ is weighted by its corresponding intensity $\mathbf{X}_{gl}$, with $N_{v,gl}$ being the number of voxels in the intensity mask:

$$\overrightarrow{CoM}_{gl} = \frac{\sum_{i=1}^{N_{v,gl}} X_{gl,i} \vec{X}_{c,gl,i}}{\sum_{i=1}^{N_{v,gl}} X_{gl,i}}$$

The distance between the two centres of mass is then:

$$F_{morph.com} = ||\overrightarrow{CoM}_{geom} - \overrightarrow{CoM}_{gl}||_2$$

4.1.11 Maximum 3D diameter

The maximum 3D diameter is the distance between the two most distant vertices in the ROI mesh vertex set $\mathbf{X}_{vx}$:.

$$F_{morph.diam} = \max\left(||\vec{X}_{vx,i} - \vec{X}_{xv,j}||_2\right), \qquad i = 1, \dots, N \qquad j = 1, \dots, N$$

A practical way of determining the maximum 3D diameter is to first construct the convex hull of the ROI mesh. The convex hull vertex set $\mathbf{X}_{vx,convex}$ is guaranteed to contain the two most distant vertices of $\mathbf{X}_{vx}$. This significantly reduces the computational cost of calculating distance between all vertices. Despite the remaining $O(n^2)$ cost of calculating distances between different vertices, the size of $\mathbf{X}_{vx,convex}$ is usually considerably smaller than the size of $\mathbf{X}_{vx}$. Moreover, the convex hull is later used for the calculation of other morphological features.

4.1.12 Major axis length

Principal component analysis (PCA) can be used to determine the main orientation of the ROI. On a three dimensional object, PCA yields three orthogonal eigenvectors $\{e_1, e_2, e_3\}$ and three eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$. These eigenvalues and eigenvectors geometrically describe a triaxial ellipsoid. The three eigenvectors determine the orientation of the ellipsoid, whereas the eigenvalues provide a measure of how far the ellipsoid extends along each eigenvector.

The eigenvalues can be ordered so that $\lambda_{major} \geq \lambda_{minor} \geq \lambda_{least}$ correspond to the major, minor and least axes of the ellipsoid respectively. The semi-axes lengths a, b and c for the major, minor and least axes are then $2\sqrt{\lambda_{major}}$, $2\sqrt{\lambda_{minor}}$ and $2\sqrt{\lambda_{least}}$ respectively. The major axis length is twice the semi-axis length a, determined using the largest eigenvalue obtained by PCA on the point set of voxel centers $\mathbf{X}_c$ (Heiberger and Holland, 2015):

$$F_{morph.pca.major} = 2a = 4\sqrt{\lambda_{major}}$$

4.1.13 Minor axis length

The minor axis length of the ROI provides a measure of how far the volume extends along the second largest axis. The minor axis length is twice the semi-axis length b, determined using the second largest eigenvalue obtained by PCA on the point set of voxel centers $\mathbf{X}_c$ (Heiberger and Holland, 2015):

$$F_{morph.pca.minor} = 2b = 4\sqrt{\lambda_{minor}}$$

4.1.14 Least axis length

The least axis is the axis along which the object is least extended. The least axis length is twice the semi-axis length c, determined using the smallest eigenvalue obtained by PCA on the point set of voxel centers $\mathbf{X}_c$ (Heiberger and Holland, 2015):

$$F_{morph.pca.least} = 2c = 4\sqrt{\lambda_{least}}$$

4.1.15 Elongation

The ratio of the major and minor axis lengths could be viewed as the extent to which a volume is longer than it is wide, i.e. is eccentric. For computational reasons, we express elongation as an inverse ratio. 1 is thus completely non-elongated, e.g. a sphere, and smaller values express greater elongation of the ROI volume.

$$F_{morph.pca.elongation} = \sqrt{\frac{\lambda_{minor}}{\lambda_{major}}}$$

4.1.16 Flatness

The ratio of the major and least axis lengths could be viewed as the extent to which a volume is flat relative to its length. For computational reasons, we express flatness as an inverse ratio. 1 is thus completely non-flat, e.g. a sphere, and smaller values express objects which are increasingly flatter.

$$F_{morph.pca.flatness} = \sqrt{\frac{\lambda_{least}}{\lambda_{major}}}$$

4.1.17 Volume density - axis-aligned bounding box

Volume density is the fraction of the ROI volume and a comparison volume. Here the comparison volume is that of the axis-aligned bounding box of the ROI mesh vertex set $\mathbf{X}_{vx}$ or the ROI mesh convex hull vertex set $\mathbf{X}_{vx,convex}$. Both vertex sets generate an identical bounding box, which is the smallest box enclosing the vertex set, and aligned with the axes of the reference frame. Thus:

$$F_{morph.v.dens.aabb} = \frac{V}{V_{aabb}}$$

This feature is also called *extent* (El Naqa et al., 2009).

4.1.18 Area density - axis-aligned bounding box

Conceptually similar to the *volume density - axis-aligned bounding box* feature, *area density* considers the ratio of the ROI surface area and the surface area A_{aabb} of the axis-aligned bounding box enclosing the ROI mesh vertex set $\mathbf{X}_{vx}$ (van Dijk et al., 2017). The bounding box is identical to the one used in the *volume density - axis-aligned bounding box* feature. Thus:

$$F_{morph.a.dens.aabb} = \frac{A}{A_{aabb}}$$

4.1.19 Volume density - oriented minimum bounding box

The volume of an axis-aligned bounding box is generally not the smallest obtainable volume enclosing the ROI. By orienting the box along a different set of axes, a smaller enclosing volume may be attainable. The oriented minimum bounding box of the ROI mesh vertex set $\mathbf{X}_{vx}$ or $\mathbf{X}_{vx,convex}$ encloses the vertex set and has the smallest possible volume. A 3-dimensional rotating callipers technique was devised by O'Rourke (1985) to derive the oriented minimum bounding box. Due to computational complexity of the rotating callipers technique, the oriented minimum bounding box is commonly approximated at lower complexity, see e.g. Barequet and Har-Peled (2001) and Chan and Tan (2001). Thus:

$$F_{morph.v.dens.ombb} = \frac{V}{V_{ombb}}$$

Here V_{ombb} is the volume of the oriented minimum bounding box.

4.1.20 Area density - oriented minimum bounding box

The area density is estimated as:

$$F_{morph.a.dens.ombb} = \frac{A}{A_{ombb}}$$

Here A_{ombb} is the surface area of the same bounding box as calculated for the *volume density - oriented minimum bounding box* feature.

4.1.21 Volume density - approximate enclosing ellipsoid

The eigenvectors and eigenvalues from principal component analysis of the ROI voxel center point set $\mathbf{X}_c$ can be used to describe an ellipsoid approximating the point cloud (Mazurowski et al., 2016). The volume of an ellipsoid is $V_{aee} = 4\pi\,a\,b\,c/3$, with a, b, and c being the length of the ellipsoid's semi-principal axes (Weisstein, 2016), see section 4.1.12. The volume density is then:

$$F_{morph.v.dens.aee} = \frac{3V}{4\pi abc}$$

4.1.22 Area density - approximate enclosing ellipsoid

The surface area of an ellipsoid can generally not be evaluated in an elementary form. However, it is possible to approximate the surface using a infinite series. We use the same semi-principal axes as for the *volume density - approximate ellipsoid* feature and define:

$$A_{aee}\left(a,b,c\right) = 4\pi\,a\,b \sum_{\nu=0}^{\infty} \frac{(\alpha\,\beta)^{\nu}}{1-4\nu^2} P_{\nu}\left(\frac{\alpha^2+\beta^2}{2\alpha\beta}\right)$$

Here $\alpha = \sqrt{1-b^2/a^2}$ and $\beta = \sqrt{1-c^2/a^2}$ are eccentricities of the ellipsoid and P_{ν} is the Legendre polynomial function for degree ν. Though infinite, the series converges, and calculation may be stopped early. Gains in precision past $\nu = 20$ are very limited, and as a default we stop calculations at this polynomial degree.

The area density is then approximately:

$$F_{morph.a.dens.aee} = \frac{A}{A_{aee}}$$

4.1.23 Volume density - minimum volume enclosing ellipsoid

The *approximate ellipsoid* may not enclose the ROI or be the smallest enclosing ellipsoid. The minimum volume enclosing ellipsoid is generally approximated to make calculation more feasible. Various algorithms have been described, e.g. (Todd and Yldrm, 2007; Ahipaaolu, 2015), which are usually elaborations on Khachiyan's barycentric coordinate descent method (Khachiyan, 1996).

The minimum volume enclosing ellipsoid encloses the ROI mesh vertex set $\mathbf{X}_{vx}$ and $\mathbf{X}_{vx,convex}$. Use of the convex mesh set $\mathbf{X}_{vx,convex}$ may be recommended due to its sparsity. The volume of the minimum volume enclosing ellipsoid is defined by its semi-axes lengths $V_{mvee} = 4\pi\,a\,b\,c/3$. Then:

$$F_{morph.v.dens.mvee} = \frac{V}{V_{mvee}}$$

For Khachiyan's barycentric coordinate descent-based methods we use a default tolerance $\tau = 0.001$ as stopping criterion.

4.1.24 Area density - minimum volume enclosing ellipsoid

The surface area of an ellipsoid does not have a general elementary form, but should be approximated, see section 4.1.22. Let the approximated surface area be A_{mvee}. Then:

$$F_{morph.a.dens.mvee} = \frac{A}{A_{mvee}}$$

4.1.25 Volume density - convex hull

The convex hull encloses ROI mesh vertex set $\mathbf{X}_{vx}$ and consists of the vertex set $\mathbf{X}_{vx,convex}$ and corresponding faces. The volume of the ROI mesh convex hull set is calculated as for the *volume* feature. The volume density can then be calculated as follows:

$$F_{morph.v.dens.conv.hull} = \frac{V}{V_{convex}}$$

This feature is also called *solidity* (El Naqa et al., 2009).

4.1.26 Area density - convex hull

The area of the convex hull is the sum of the area of the faces of the convex hull, as in the calculation of the *area* feature. The convex hull is identical to the one used in the *volume density - convex hull* feature. Then:

$$F_{morph.a.dens.conv.hull} = \frac{A}{A_{convex}}$$

4.1.27 Integrated intensity

Integrated intensity is the average grey level multiplied by the volume. In the context of ^{18}F-FDG-PET, this feature is called *total legion glycolysis* (Vaidya et al., 2012). Thus:

$$F_{morph.integ.int} = V \, \frac{1}{N_{v,gl}} \sum_{i=1}^{N_{v,gl}} X_{gl,i}$$

$N_{v,gl}$ is the number of voxels in the intensity mask.

4.1.28 Moran's I index

Moran's *I* index is an indicator of spatial autocorrelation (Moran, 1950; Dale et al., 2002). It is defined as:

$$F_{morph.moran.i} = \frac{N_v}{\sum_{i=1}^{N_v}\sum_{j=1}^{N_v} w_{ij}} \frac{\sum_{i=1}^{N_v}\sum_{j=1}^{N_v} w_{ij}\left(X_{gl,i}-\mu\right)\left(X_{gl,j}-\mu\right)}{\sum_{i=1}^{N_v}\left(X_{gl,i}-\mu\right)^2}, \qquad i \neq j$$

Here $N_v = N_{v,gl}$ is the number of voxel in the ROI intensity mask, μ is the mean of $\mathbf{X}_{gl}$ and w_{ij} is a weight factor, equal to the inverse Euclidean distance between voxels i and j of the point set $\mathbf{X}_{c,gl}$ of the intensity mask (Da Silva et al., 2008). Moran's index values close to 1.0, 0.0 and -1.0 indicate high spatial autocorrelation, no spatial autocorrelation and high spatial anti-autocorrelation respectively.

4.1.29 Geary's C measure

Geary's *C* measures spatial autocorrelation, like Moran's *I* index (Geary, 1954; Dale et al., 2002). Geary's *C* however, directly measures grey level differences between voxels and is more sensitive to local spatial autocorrelation. This measure is defined as:

$$F_{morph.geary.c} = \frac{N_v-1}{2\sum_{i=1}^{N_v}\sum_{j=1}^{N_v} w_{ij}} \frac{\sum_{i=1}^{N_v}\sum_{j=1}^{N_v} w_{ij}\left(X_{gl,i}-X_{gl,j}\right)^2}{\sum_{i=1}^{N_v}\left(X_{gl,i}-\mu\right)^2}, \qquad i \neq j$$

As with Moran's I, $N_v = N_{v,gl}$ is the number of voxel in the ROI intensity mask, μ is the mean of $\mathbf{X}_{gl}$ and w_{ij} is a weight factor, equal to the inverse Euclidean distance between voxels i and j of the ROI voxel point set $\mathbf{X}_{c,gl}$ (Da Silva et al., 2008).

4.2 Local intensity features

Voxel intensities within a defined neighbourhood around a center voxel are considered for local intensity features. Unlike many feature sets, local features do not draw solely on intensities within the ROI. While only voxels within the ROI intensity map are used as a center voxel, the corresponding local neighbourhood draws upon all voxels regardless of being in a ROI.

4.2.1 Local intensity peak

The local intensity peak was originally devised for reducing variance in determining standardised uptake values (Wahl et al., 2009). It is defined as the mean grey level in a 1 cm³ spherical volume, centered on the voxel with the maximum grey level in the ROI intensity mask.

To calculate $F_{loc.peak.loc}$, we first select all the voxels with voxel centers within radius $r = \left(\frac{3}{4\pi}\right)^{1/3} \approx 0.62$ cm of the voxel with the maximum grey level. Subsequently, the mean grey level of the selected voxels, including the center voxel, are calculated.

In case the maximum grey level is found in multiple ROI voxels, *local intensity peak* is calculated for each of these voxels, and the highest local intensity peak chosen.

4.2.2 Global intensity peak

The *global intensity peak* is similar to $F_{loc.peak.loc}$. Instead of calculating the mean intensity for the voxel(s) with the maximum grey level, the mean intensity is calculated within a neighbourhood for every voxel in the ROI intensity mask. The highest intensity peak value is then selected.

4.3 Statistical features

The statistical features describe how grey levels within the region of interest (ROI) are distributed. The features in this set do not require discretisation, and may be used to describe a continuous distribution, such as FDG-PET. Then, let $\mathbf{X}_{gl} = \{X_{gl,1}, X_{gl,2}, \ldots, X_{gl,N_v}\}$ be the set of grey levels of the N_v voxels included in the ROI intensity mask.

4.3.1 Mean

The mean grey level of $\mathbf{X}_{gl}$ is calculated as:

$$F_{stat.mean} = \frac{1}{N_v} \sum_{j=1}^{N_v} X_{gl,j}$$

4.3.2 Variance

The grey level variance of $\mathbf{X}_{gl}$ is defined as:

$$F_{stat.var} = \frac{1}{N_v} \sum_{j=1}^{N_v} \left(X_{gl,j} - \mu\right)^2$$

4.3.3 Skewness

The skewness of the grey level distribution of $\mathbf{X}_{gl}$ is defined as:

$$F_{stat.skew} = \frac{\frac{1}{N_v} \sum_{j=1}^{N_v} \left(X_{gl,j} - \mu\right)^3}{\left(\frac{1}{N_v} \sum_{j=1}^{N_v} \left(X_{gl,j} - \mu\right)^2\right)^{3/2}}$$

Here $\mu = F_{stat.mean}$. If the grey level variance $F_{stat.var} = 0$, $F_{stat.skew} = 0$.

4.3.4 Kurtosis

Kurtosis, or technically excess kurtosis, is calculated as measure of peakedness in the grey level distribution of $\mathbf{X}_{gl}$:

$$F_{stat.kurt} = \frac{\frac{1}{N_v} \sum_{j=1}^{N_v} \left(X_{gl,j} - \mu\right)^4}{\left(\frac{1}{N_v} \sum_{j=1}^{N_v} \left(X_{gl,j} - \mu\right)^2\right)^2} - 3$$

Here $\mu = F_{stat.mean}$. Note that kurtosis is corrected by a Fisher correction of -3 to center kurtosis on 0 for normal distributions. If the grey level variance $F_{stat.var} = 0$, $F_{stat.kurt} = 0$.

4.3.5 Median

The median value $F_{stat.median}$ is the sample median of $\mathbf{X}_{gl}$.

4.3.6 Minimum grey level

The minimum grey level is equal to the lowest grey level present in $\mathbf{X}_{gl}$.

4.3.7 10th percentile

P_{10} is the 10th percentile of $\mathbf{X}_{gl}$. P_{10} is more robust to outliers in grey level then the minimum grey level.

4.3.8 90th percentile

P_{90} is the 90th percentile of $\mathbf{X}_{gl}$. P_{90} is more robust to outliers in grey level then the maximum grey level.

4.3.9 Maximum grey level

The maximum grey level is equal to the highest grey level present in $\mathbf{X}_{gl}$.

4.3.10   Interquartile range

The interquartile range (IQR) of $\mathbf{X}_{gl}$ is defined as:

$$F_{stat.iqr} = P_{75} - P_{25}$$

P_{25} and P_{75} are the 25th and 75th percentile of $\mathbf{X}_{gl}$, respectively.

4.3.11   Range

The range of grey levels is defined as:

$$F_{stat.range} = \max(\mathbf{X}_{gl}) - \min(\mathbf{X}_{gl})$$

4.3.12   Mean absolute deviation

The mean absolute deviation is a measure of dispersion from the mean of $\mathbf{X}_{gl}$:

$$F_{stat.mad} = \frac{1}{N_v} \sum_{j=1}^{N_v} |X_{gl,j} - \mu|$$

Here $\mu = F_{stat.mean}$.

4.3.13   Robust mean absolute deviation

The mean absolute deviation can be influenced by outliers, and mean dispersion from the mean could be affected as a consequence. The set of grey levels can be restricted to grey levels which lie closer to the center of the distribution. Let

$$\mathbf{X}_{10-90} = \{x \in \mathbf{X}_{gl} | P_{10}(\mathbf{X}_{gl}) \leq x \leq P_{90}(\mathbf{X}_{gl})\}$$

This means that $\mathbf{X}_{10-90}$ is the set of $N_{10-90} \leq N$ voxels in $\mathbf{X}_{gl}$ whose grey levels are equal to, or lie between, the values corresponding to the 10th and 90th percentiles of $\mathbf{X}_{gl}$. The robust mean absolute deviation is then:

$$F_{stat.rmad} = \frac{1}{N_{10-90}} \sum_{j=1}^{N_{10-90}} \left| X_{gl,10-90,j} - \overline{X}_{gl,10-90} \right|$$

$\overline{X}_{gl,10-90}$ denotes the sample mean of $\mathbf{X_{gl,10-90}}$.

4.3.14   Median absolute deviation

Median absolute deviation is similar in concept to $F_{stat.mad}$, but measures dispersion from the median instead of mean. Thus

$$F_{stat.medad} = \frac{1}{N_v} \sum_{j=1}^{N_v} |X_{gl,j} - M|$$

Here, median $M = F_{stat.median}$.

4.3.15 Coefficient of variation

The coefficient of variation measures the dispersion of the $\mathbf{X}_{gl}$ distribution. It is defined as

$$F_{stat.cov} = \frac{\sigma}{\mu}$$

Here $\sigma = F_{stat.var}{}^{1/2}$ and $\mu = F_{stat.mean}$ are the standard deviation and mean of the grey level distribution, respectively.

4.3.16 Quartile coefficient of dispersion

The quartile coefficient of dispersion is a robust alternative to coefficient of variance. It is defined as

$$F_{stat.qcod} = \frac{P_{75} - P_{25}}{P_{75} + P_{25}}$$

P_{25} and P_{75} are the 25$^{\text{th}}$ and 75$^{\text{th}}$ percentile of $\mathbf{X}_{gl}$, respectively.

4.3.17 Energy

The grey level energy of $\mathbf{X}_{gl}$ is defined as:

$$F_{stat.energy} = \sum_{j=1}^{N_v} X_{gl,j}^2$$

4.3.18 Root mean square

The root mean square metric, also called the quadratic mean, of $\mathbf{X}_{gl}$ is defined as:

$$F_{stat.rms} = \sqrt{\frac{\sum_{j=1}^{N_v} X_{gl,j}^2}{N_v}}$$

4.4 Intensity histogram features

An intensity histogram is generated by discretising the original set of grey levels $\mathbf{X}_{gl}$ into grey level bins. Approaches to discretisation are described elsewhere in the document. Thus, let $\mathbf{X}_d = \{X_{d,1}, X_{d,2}, \ldots, X_{d,N_v}\}$ be the set of discretised grey levels of the N_v voxels in the ROI intensity mask. Let $\mathbf{H} = \{n_1, n_2, \ldots\}$ be the histogram with frequency count n_i of each discretised grey level i in $\mathbf{X}_d$. The number of grey level bins of the histogram is N_g. The occurrence probability p_i for each grey level bin i is then approximated as $p_i = n_i/N_v$.

4.4.1 Intensity histogram mean

The mean grey level of $\mathbf{X}_d$ is calculated as:

$$F_{ih.mean} = \frac{1}{N_v} \sum_{j=1}^{N_v} X_{d,j}$$

An equivalent formulation is:

$$F_{ih.mean} = \sum_{i=1}^{N_g} i \, p_i$$

4.4.2 Intensity histogram variance

The variance of $\mathbf{X}_d$ is defined as:

$$F_{ih.var} = \frac{1}{N_v} \sum_{j=1}^{N_v} (X_{d,j} - \mu)^2$$

Here $\mu = F_{ih.mean}$. This formulation is equivalent to:

$$F_{ih.var} = \sum_{i=1}^{N_g} (i - \mu)^2 \, p_i$$

4.4.3 Intensity histogram skewness

The skewness of $\mathbf{X}_d$ is defined as:

$$F_{ih.skew} = \frac{\frac{1}{N_v} \sum_{j=1}^{N_v} (X_{d,j} - \mu)^3}{\left(\frac{1}{N_v} \sum_{j=1}^{N_v} (X_{d,j} - \mu)^2 \right)^{3/2}}$$

Here $\mu = F_{ih.mean}$. This formulation is equivalent to:

$$F_{ih.skew} = \frac{\sum_{i=1}^{N_g} (i - \mu)^3 \, p_i}{\left(\sum_{i=1}^{N_g} (i - \mu)^2 \, p_i \right)^{3/2}}$$

If the discretised grey level variance $F_{ih.var} = 0$, $F_{ih.skew} = 0$.

4.4.4 Intensity histogram kurtosis

Kurtosis, or technically excess kurtosis, is calculated as measure of peakedness of the distribution $\mathbf{X}_d$:

$$F_{ih.kurt} = \frac{\frac{1}{N_v} \sum_{j=1}^{N_v} (X_{d,j} - \mu)^4}{\left(\frac{1}{N_v} \sum_{j=1}^{N_v} (X_{d,j} - \mu)^2 \right)^2} - 3$$

Here $\mu = F_{ih.mean}$. The alternative, but equivalent, formulation is:

$$F_{ih.kurt} = \frac{\sum_{i=1}^{N_g} (i - \mu)^4 \, p_i}{\left(\sum_{i=1}^{N_g} (i - \mu)^2 \, p_i \right)^2} - 3$$

Note that kurtosis is corrected by a Fisher correction of -3 to center kurtosis on 0 for normal distributions. If the discretised grey level $F_{ih.var} = 0$, $F_{ih.kurt} = 0$.

4.4.5 Intensity histogram median

The median value $F_{ih.median}$ is the sample median of $\mathbf{X}_d$.

4.4.6 Intensity histogram minimum grey level

The minimum grey level bin is equal to the lowest discretised grey level present in $\mathbf{X}_d$.

4.4.7 Intensity histogram 10th percentile

P_{10} is the 10th percentile of $\mathbf{X}_d$.

4.4.8 Intensity histogram 90th percentile

P_{90} is the 90th percentile of $\mathbf{X}_d$.

4.4.9 Intensity histogram maximum grey level

The maximum grey level is equal to the highest discretised grey level present in $\mathbf{X}_d$.

4.4.10 Intensity histogram mode

The mode of $\mathbf{X}_d$ is the most common discretised grey level present, i.e. i for which count n_i is maximal. The mode may not be uniquely defined. When multiple bins contain the highest grey level count, the bin closest to the histogram mean is chosen as $F_{ih.mode}$. In pathological cases with two such bins equidistant to the mean, the bin to the left of the mean is selected.

4.4.11 Intensity histogram interquartile range

The interquartile range (IQR) of $\mathbf{X}_d$ is defined as:

$$F_{ih.iqr} = P_{75} - P_{25}$$

P_{25} and P_{75} are the 25th and 75th percentile of $\mathbf{X}_d$, respectively. The interquartile range of $\mathbf{X}_d$ is always an integer.

4.4.12 Intensity histogram range

The range of grey levels in the histogram is defined as:

$$F_{ih.range} = \max(\mathbf{X}_d) - \min(\mathbf{X}_d)$$

This equal to the width of the histogram.

4.4.13 Intensity histogram mean absolute deviation

The mean absolute deviation is a measure of dispersion from the mean of $\mathbf{X}_d$:

$$F_{ih.mad} = \frac{1}{N_v} \sum_{i=1}^{N_v} |X_{d,i} - \mu|$$

Here $\mu = F_{ih.mean}$.

4.4.14   Intensity histogram robust mean absolute deviation

The histogram mean absolute deviation can be influenced by outliers, and mean dispersion from the mean could be affected as a consequence. The set of discretised grey levels under consideration can be restricted to those which are closer to the center of the distribution. Let

$$\mathbf{X}_{10-90} = \{x \in \mathbf{X}_d | P_{10}(\mathbf{X}_d) \leq x \leq P_{90}(\mathbf{X}_d)\}$$

Shortly, $\mathbf{X}_{10-90}$ is the set of $N_{10-90} \leq N$ voxels in $\mathbf{X}_d$ whose discretised grey levels are equal to, or lie between, the values corresponding to the 10$^{\text{th}}$ and 90$^{\text{th}}$ percentiles of $\mathbf{X}_d$. The robust mean absolute deviation is then:

$$F_{ih.rmad} = \frac{1}{N_{10-90}} \sum_{j=1}^{N_{10-90}} \left| X_{d,10-90,j} - \overline{X}_{d,10-90} \right|$$

$\overline{X}_{d,10-90}$ denotes the sample mean of $\mathbf{X_{d,10-90}}$.

4.4.15   Intensity histogram median absolute deviation

Histogram median absolute deviation is similar in concept to $F_{ih.mad}$, but measures dispersion from the median instead of mean. Thus:

$$F_{ih.medmad} = \frac{1}{N_v} \sum_{j=1}^{N_v} |X_{d,j} - M|$$

Here, median $M = F_{ih.median}$.

4.4.16   Intensity histogram coefficient of variation

The coefficient of variation measures the dispersion of the histogram. It is defined as:

$$F_{ih.cov} = \frac{\sigma}{\mu}$$

Here $\sigma = F_{ih.var}^{1/2}$ and $\mu = F_{ih.mean}$ are the standard deviation and mean of the discretised grey level distribution, respectively.

4.4.17   Intensity histogram quartile coefficient of dispersion

The quartile coefficient of dispersion is a robust alternative to coefficient of variance. It is defined as:

$$F_{ih.qcod} = \frac{P_{75} - P_{25}}{P_{75} + P_{25}}$$

P_{25} and P_{75} are the 25$^{\text{th}}$ and 75$^{\text{th}}$ percentile of $\mathbf{X}_d$, respectively.

4.4.18   Intensity histogram entropy

Entropy is a information-theoretic concept that gives a metric for the information contained within $\mathbf{X}_d$. The particular metric used is Shannon entropy, which is defined as:

$$F_{ih.entropy} = -\sum_{i=1}^{N_g} p_i \log_2 p_i$$

4.4.19 Intensity histogram uniformity

Uniformity of $\mathbf{X}_d$ is defined as:

$$F_{ih.uniformity} = \sum_{i=1}^{N_g} p_i^2$$

Note that this feature is also referred to as energy.

4.4.20 Maximum histogram gradient

The histogram gradient can be calculated as:

$$\mathbf{H}' = \left\{ H(2) - H(1), \ldots, \frac{H(i+1) - H(i-1)}{2}, \ldots, H(N_g) - H(N_g - 1) \right\}$$

Note that this requires a non-sparse representation, i.e. empty bins should be presented. The histogram gradient can ostensibly be calculated in different ways. The suggested method has the advantage of being simple to implement and producing a gradient with same number of elements as the original histogram. The latter helps avoid ambiguity concerning which discretised grey level corresponds to which bin that would occur in simple bin difference gradients. This feature was then defined by van Dijk et al. (2017) as:

$$F_{ih.max.grad} = \max\left(\mathbf{H}'\right)$$

4.4.21 Maximum histogram gradient grey level

This feature was defined by van Dijk et al. (2017) as the discretised grey level corresponding to the maximum histogram gradient, i.e. i for which $\mathbf{H}'$ was maximal.

4.4.22 Minimum histogram gradient

This feature was defined by van Dijk et al. (2017) as the minimum gradient of the grey level histogram:

$$F_{ih.min.grad} = \min\left(\mathbf{H}'\right)$$

4.4.23 Minimum histogram gradient grey level

This feature was defined by van Dijk et al. (2017) as the discretised grey level corresponding to the minimum histogram gradient, i.e. i for which $\mathbf{H}'$ was minimal.

4.5 Intensity-volume histogram features

The (cumulative) intensity-volume histogram (IVH) of the set of grey levels $\mathbf{X}_{gl}$ of voxels in the ROI intensity mask describes the relationship between grey level i and the volume fraction ν which contains at least grey level i or a higher (El Naqa et al., 2009).

The calculation of IVH features requires a discrete grey level ROI voxel set $\mathbf{X}_d$ which contains discrete grey levels from set $\mathbf{G}$. Depending on the imaging modality, calculation of both $\mathbf{X}_d$ and $\mathbf{G}$ is straightforward, or requires prior discretisation.

Images with discretised intensities Some images by default contain voxels with discrete intensities, e.g. CT with Hounsfield units. In this case, the discretised ROI voxel set $\mathbf{X}_d = \mathbf{X}_{gl}$. The corresponding set of discretised grey levels $\mathbf{G} = \{\min(\mathbf{X}_d), \min(\mathbf{X}_d) + 1, \ldots, \max(\mathbf{X}_d)\}$. In case when a range of intensities in the ROI has been set by the user, the set of discretised $\mathbf{G}$ grey levels should cover the specified range.

Images requiring discretisation Images without discrete intensities require discretisation. Our general recommendation is to use a *fixed bin number* discretisation method with $N_b = 1000$ bins. The bin number for voxel j is then:

$$X_{b,j} = \begin{cases} 1 & X_{gl,j} = X_{gl,min} \\ \lceil N_b \frac{X_{gl,j} - X_{gl,min}}{X_{gl,max} - X_{gl,min}} \rceil & X_{gl,j} > X_{gl,min} \end{cases}$$

Here, $X_{gl,min} = \min(\mathbf{X}_{gl})$ and $X_{gl,max} = \max(\mathbf{X}_{gl})$ are the minimum and maximum grey level in the ROI. The discrete grey level set for IVH is represented by the bin centers, not bin numbers. Therefore, let $w_b = (X_{gl,max} - X_{gl,min})/N_b$ be the bin width, and:

$$X_{d,j} = X_{gl,min} + (X_{b,j} - 0.5) w_b$$

The corresponding discretised grey levels are $\mathbf{G} = \{\min(\mathbf{X}_d), \min(\mathbf{X}_d) + w_b, \ldots, \max(\mathbf{X}_d)\}$.

In some cases, notably SUV PET images, a more meaningful grey level range can be defined. In SUV PET images the minimum SUV is defined at 0.0, and intensity is related to underlying physiological processes. In such cases, a *fixed bin size* discretisation method may be more appropriate. The bin number of voxel j is then:

$$X_{b,j} = \begin{cases} 1 & X_{gl,j} = X_{gl,min} \\ \lceil \frac{X_{gl,j} - X_{gl,min}}{w_b} \rceil & X_{gl,j} > X_{gl,min} \end{cases}$$

Here w_b and $X_{gl,min}$ are the bin width and minimum grey level parameters respectively. For SUV PET, one may use $w_b = 0.10$ and $X_{gl,min} = 0.00$. As above, the discrete grey level of voxel j is represented by the bin centers:

$$X_{d,j} = X_{gl,min} + (X_{b,j} - 0.5) w_b$$

The corresponding discretised grey levels are $\mathbf{G} = \{\min(\mathbf{X}_d), \min(\mathbf{X}_d) + w_b, \ldots, \max(\mathbf{X}_d)\}$. As for images with discretised intensities, the user may specify a closed or half-open intensity range. For modalities where *fixed bin size* binning is used, only the minimum image intensity setting has an effect on IVH feature values.

Calculating the IV histogram From $\mathbf{X}_d$ and $\mathbf{G}$ we calculate fractional volumes and fractional grey levels. As voxels for the same image stack usually all have the same dimensions, we may define fractional volume ν for discrete grey level i in $\mathbf{G}$ as:

$$\nu_i = 1 - \frac{1}{N_v} \sum_{j=1}^{N_v} \delta(X_{d,j} < i)$$

Here δ is the Kronecker delta. In essence, we count the voxels containing a discretised grey level smaller than i, divide by the total number of voxels, and then subtract this volume fraction.

i	γ	ν
1	0.0	1.000
2	0.2	0.324
3	0.4	0.324
4	0.6	0.311
5	0.8	0.095
6	1.0	0.095

Table 4.1: Example intensity-volume histogram evaluated at discrete grey levels i of the digital phantom. γ is the fractional grey level starting at $i = 1$, and with maximum $i = 6$. ν is the corresponding partial volume fraction.

The grey level fraction γ for discrete grey level i in **G** is calculated as:

$$\gamma_i = \frac{i - \min(\mathbf{X}_d)}{\max(\mathbf{X}_d) - \min(\mathbf{X}_d)}$$

An example IVH for the digital phantom is shown in Table 4.1.

4.5.1 Volume at intensity fraction

The *volume at intensity fraction V_x* is the largest volume fraction (ν) that has an intensity fraction γ of least $x\%$. This differs from conceptually similar dose-volume histograms used in radiotherapy planning, where V_{10} would indicate the volume fraction receiving at least 10 Gy planned dose. El Naqa et al. (2009) defined both V_{10} and V_{90} as features.

4.5.2 Intensity at volume fraction

The *intensity at volume fraction I_x* is the minimum grey level i present in at most $x\%$ of the volume. El Naqa et al. (2009) defined both I_{10} and I_{90} as features.

4.5.3 Volume at intensity fraction difference

This feature is the difference between the volume fractions at two different intensity fractions, e.g. $V_{10} - V_{90}$ (El Naqa et al., 2009).

4.5.4 Intensity at volume fraction difference

This feature is the difference between grey levels at two different fractional volumes, e.g. $I_{10} - I_{90}$ (El Naqa et al., 2009).

4.5.5 Area under IVH curve

The area under the IVH curve was defined by van Velden et al. (2011). The area under the IVH curve can be approximated by calculating the Riemann sum using the trapezoidal rule. Note that if there is only one grey level in the ROI, $F_{ivh.auc} = 0$.

4.6 Textural features - Grey level co-occurrence based features

In image analysis, texture is one of the defining sets of features. Texture features were originally designed to assess surface texture in 2D images. Texture is not restricted to 2D slices, but may be extended to 3D objects. Image grey levels are generally discretised before calculation of textural features. Approaches to discretisation are described elsewhere in the document.

The grey level co-occurrence matrix (GLCM) is a matrix that expresses how combinations of discretised grey levels of neighbouring pixels, or voxels in a 3D volume, are distributed along one of the image directions. In a 3 dimensional approach to texture analysis, the direct neighbourhood of a voxel consists of the 26 directly neighbouring voxels. Thus, there are 13 unique direction vectors within a neighbourhood volume for distance 1, i.e. $(0,0,1)$, $(0,1,0)$, $(1,0,0)$, $(0,1,1)$, $(0,1,-1)$, $(1,0,1)$, $(1,0,-1)$, $(1,1,0)$, $(1,-1,0)$, $(1,1,1)$, $(1,1,-1)$, $(1,-1,1)$ and $(1,-1,-1)$.

An alternative approach is to determine the GLCM on image slices, and ignore connections between slices. In this 2 dimensional approach there are only 8 direct neighbours. The corresponding direction vectors at distance 1 are $(1,0,0)$, $(1,1,0)$, $(0,1,0)$ and $(-1,1,0)$. A drawback of the 2 dimensional approach is that the resulting features are not geometrically invariant. However, it should also be considered that for many imaging modalities voxel sizes are not isotropic. The in-plane voxel dimensions are usually smaller than the slice thickness. This difference necessitates interpolation between slices to maintain isotropic voxels. At this point in time it is not clear which of the two approaches produces the most robust and effective imaging biomarkers.

A GLCM is calculated for each direction vector, as follows. Let $\mathbf{M}_\Delta$ be the $N_g \times N_g$ grey level co-occurence matrix, where N_g is the number of discretised grey levels present in the ROI intensity mask, and Δ the particular direction. Element (i,j) is the frequency at which combinations of discretised grey levels i and j occur in neighbouring voxels along direction δ and along direction $-\delta$. Then, $\mathbf{M}_\Delta = \mathbf{M}_\delta + \mathbf{M}_{-\delta} = \mathbf{M}_\delta + \mathbf{M}_\delta^T$ (Haralick et al., 1973).

An example for GLCM calculation is shown in Table 4.2. Corresponding grey level co-occurrence matrices for each direction are shown in Table 4.3.

						j					j		
1	2	2	3		0	3	0	0		0	0	0	2
1	2	3	3		0	1	3	1		3	1	0	1
4	2	4	1	i	0	0	1	0	i	0	3	1	0
4	1	2	3		2	1	0	0		0	1	0	0

(a) Grey levels   **(b)** $\mathbf{M}_{\delta=\rightarrow}$ **(c)** $\mathbf{M}_{\delta=\leftarrow}$

Table 4.2: Grey levels (a) and corresponding grey level co-occurrence matrices for the $0°$ (b) and $180°$ directions (c). In vector notation these directions are $\delta = (1,0)$ and $\delta = (-1,0)$

Grey level co-occurrence matrix features rely on the probability distribution for the elements of the GLCM. Let us consider $\mathbf{M}_{\Delta=(1,0)}$ from the example, as in Table 4.4. We get a probability distribution for grey level co-occurrences, $\mathbf{P}_\Delta$, by normalising $\mathbf{M}_\Delta$ by the sum of its elements. Each element p_{ij} of $\mathbf{P}_\Delta$ is then the joint probability of grey levels i and j occurring in neighbouring voxels along direction Δ. Then $p_{i.} = \sum_{j=1}^{N_g} p_{ij}$ is the row marginal

	j			
i	0	3	0	2
	3	2	3	2
	0	3	2	0
	2	2	0	0

(a) $\mathbf{M}_{\Delta=\rightarrow}$

	j			
i	0	2	0	1
	2	2	1	2
	0	1	2	1
	1	2	1	0

(b) $\mathbf{M}_{\Delta=\nearrow}$

	j			
i	2	1	2	1
	1	4	1	1
	2	1	2	1
	1	1	1	2

(c) $\mathbf{M}_{\Delta=\uparrow}$

	j			
i	0	2	1	1
	2	2	2	1
	1	2	0	1
	1	1	1	0

(d) $\mathbf{M}_{\Delta=\nwarrow}$

Table 4.3: Grey level co-occurrence matrices for the $0°$ (a), $45°$ (b), $90°$ (c) and $135°$ (d) directions. In vector notation these directions are $\Delta = (1,0)$, $\Delta = (1,1)$, $\Delta = (0,1)$ and $\Delta = (-1,1)$

probability, and $p_{.j} = \sum_{i=1}^{N_g} p_{ij}$ is the column marginal probability. As $\mathbf{P}_\Delta$ is by definition symmetric, $p_{i.} = p_{.j}$. Furthermore, let us consider diagonal and cross-diagonal probabilities p_{i-j} and p_{i+j}.

$$p_{i-j}(k) = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p_{ij}\,\delta(k-|i-j|) \qquad k = 0,\ldots,N_g-1$$

$$p_{i+j}(k) = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p_{ij}\,\delta(k-(i+j)) \qquad k = 2,\ldots,2N_g$$

Here, $\delta(x)$ is the Kronecker delta, which equals 1 when $x = 0$ and 0 elsewhere. In calculating diagonal and cross-diagonal probabilities it is used as a filter to select only certain combinations of elements (i,j).

It should be noted that while a distance d of 1 is commonly used for the GLCM, different distances are possible. For example, for $d = 3$ the voxels at $(0,0,3)$, $(0,3,0)$, $(3,0,0)$, $(0,3,3)$, $(0,3,-3)$, $(3,0,3)$, $(3,0,-3)$, $(3,3,0)$, $(3,-3,0)$, $(3,3,3)$, $(3,3,-3)$, $(3,-3,3)$ and $(3,-3,-3)$ from the center voxel are considered.

Summarising features Feature values are calculated after calculating the grey level co-occurrence matrices and corresponding probability distributions from the ROI intensity mask. Five methods can be used to arrive at a single feature value for each volume. A schematic example is shown in Figure 4.2. Three methods involve merging of matrices. By merging the occurrence count for each individual combination of elements (i,j) in the GLCMs is summed. Probability distributions are subsequently calculated using the merged matrix, and features calculated.

		j			$\sum_j$
	0	3	0	2	**5**
	3	2	3	2	**10**
i	0	3	2	0	**5**
	2	2	0	0	**4**
$\sum_i$	**5**	**10**	**5**	**4**	**24**

(a) $\mathbf{M}_{\Delta=(1,0)}$ with margins

		j			$p_{i.}$
	0.00	0.13	0.00	0.08	**0.21**
	0.13	0.08	0.13	0.08	**0.42**
i	0.00	0.13	0.08	0.00	**0.21**
	0.08	0.08	0.00	0.00	**0.17**
$p_{.j}$	**0.21**	**0.42**	**0.21**	**0.17**	**1.00**

(b) $\mathbf{P}_{\Delta=(1,0)}$ with margins

$k = \lvert i - j \rvert$	0	1	2	3
p_{i-j}	0.17	0.50	0.17	0.17

(c) Diagonal probability for $\mathbf{P}_{\Delta=(1,0)}$

$k = i + j$	2	3	4	5	6	7	8
p_{i+j}	0.00	0.25	0.08	0.42	0.25	0.00	0.00

(d) Cross-diagonal probability for $\mathbf{P}_{\Delta=(1,0)}$

Table 4.4: Grey level co-occurrence matrix for the $0°$ direction (a); its corresponding probability matrix $\mathbf{P}_{\Delta=(1,0)}$ with marginal probabilities $p_{i.}$ and $p_{.j}$(b); the diagonal probabilities p_{i-j} (c); and the cross-diagonal probabilities p_{i+j} (d). Discrepancies in panels b, c, and d are due to rounding errors. Note that due the matrix symmetry marginal probabilities $p_{i.}$ and $p_{.j}$ are the same in both row and column margins.

4.6.1 Joint maximum

The joint maximum is the probability corresponding to the most common grey level co-occurrence in the GLCM.

$$F_{cm.joint.max} = \max(p_{ij})$$

4.6.2 Joint average

The joint average is the grey level weighted sum of joint probabilities.

$$F_{cm.joint.avg} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i\, p_{ij}$$

4.6.3 Joint variance

The joint variance, which is also called *sum of squares* (Haralick et al., 1973), is defined as:

$$F_{cm.joint.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu)^2\, p_{ij}$$
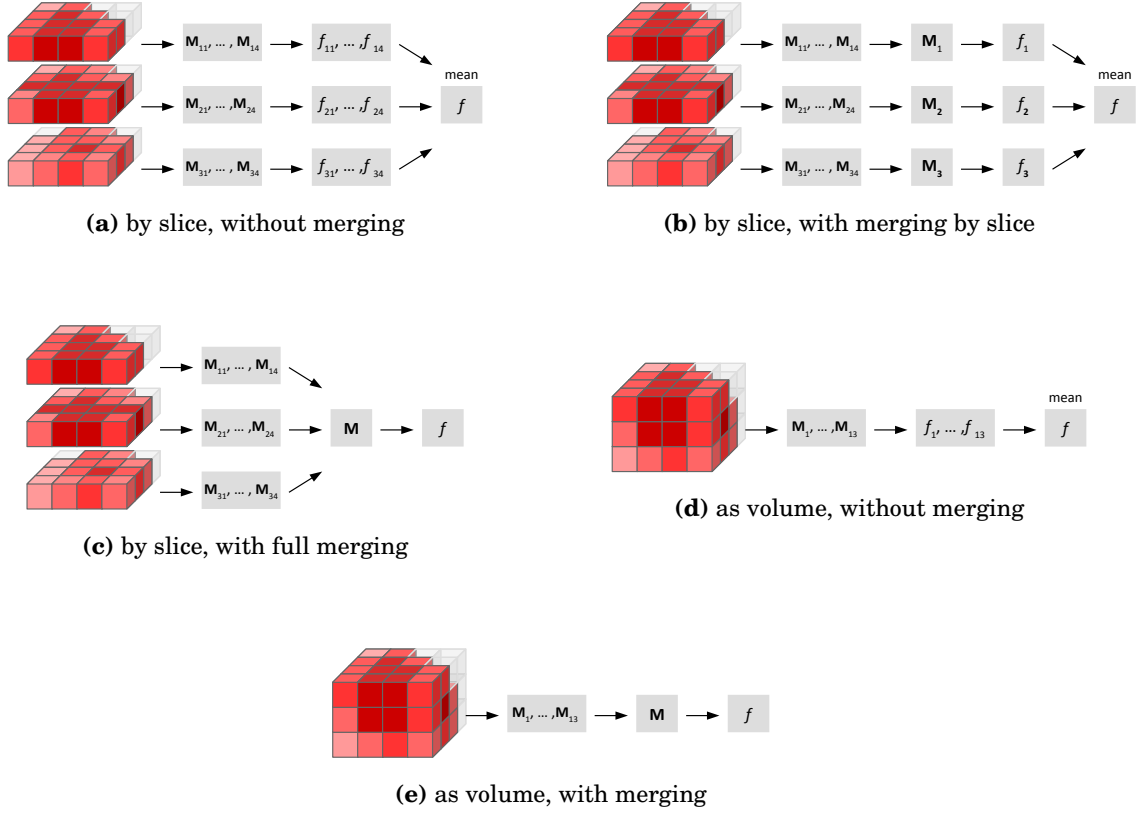
(a) by slice, without merging

(b) by slice, with merging by slice

(c) by slice, with full merging

(d) as volume, without merging

(e) as volume, with merging

Figure 4.2: Approaches to calculating grey level co-occurrence matrix-based features. $M_{\Delta k}$ are texture matrices calculated for direction Δ in slice k (if applicable), and $f_{\Delta k}$ is the corresponding feature value. In (b), (c) and (e) the matrices are merged prior to feature calculation.

Here μ is equal to the value of $F_{cm.joint.avg}$, which was defined previously.

4.6.4 Joint entropy

Joint entropy (Haralick et al., 1973) is defined as:

$$F_{cm.joint.entr} = -\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} p_{ij} \log_2 p_{ij}$$

4.6.5 Difference average

The average for the diagonal probabilities is defined as:

$$F_{cm.diff.avg} = \sum_{k=0}^{N_g-1} k\, p_{i-j}(k)$$

Note that k in $p_{i-j}(k)$ is used as an index for p_{i-j} and does not represent a multiplication.

4.6.6 Difference variance

The variance for the diagonal probabilities (Haralick et al., 1973) is defined as:

$$F_{cm.diff.var} = \sum_{k=0}^{N_g-1} (k - \mu)^2 p_{i-j}(k)$$

Here μ is equal to the value of $F_{cm.diff.avg}$.

4.6.7 Difference entropy

The entropy for the diagonal probabilities (Haralick et al., 1973) is defined as:

$$F_{cm.diff.entr} = - \sum_{k=0}^{N_g-1} p_{i-j}(k) \log_2 p_{i-j}(k)$$

4.6.8 Sum average

The average for the cross-diagonal probabilities (Haralick et al., 1973) is defined as:

$$F_{cm.sum.avg} = \sum_{k=2}^{2N_g} k \, p_{i+j}(k)$$

4.6.9 Sum variance

The variance for the cross-diagonal probabilities (Haralick et al., 1973) is defined as:

$$F_{cm.sum.var} = \sum_{k=2}^{2N_g} (k - \mu)^2 p_{i+j}(k)$$

Here μ is equal to the value of $F_{cm.sum.avg}$.

4.6.10 Sum entropy

The entropy for the cross-diagonal probabilities (Haralick et al., 1973) is defined as:

$$F_{cm.sum.entr} = - \sum_{k=2}^{2N_g} p_{i+j}(k) \log_2 p_{i+j}(k)$$

4.6.11 Angular second moment

The angular second moment (Haralick et al., 1973), which represents the energy of $\mathbf{P}_\Delta$, is defined as:

$$F_{cm.energy} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij}^2$$

4.6.12 Contrast

Contrast assesses grey level variations (Haralick et al., 1973). Hence elements of M_Δ that represent large grey level differences receive greater weight. Contrast is defined as (Clausi, 2002):

$$F_{cm.contrast} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i-j)^2 \, p_{ij}$$

Note that the original definition is seemingly more complex, but simplifying terms leads to the above formulation of contrast.

4.6.13 Dissimilarity

Dissimilarity is conceptually similar to the *Contrast* feature, and is defined as:

$$F_{cm.dissimilarity} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} |i-j| \, p_{ij}$$

4.6.14 Inverse difference

Inverse difference is a measure of homogeneity. Grey level co-occurrences with a large difference in levels are weighed less, thus lowering the total feature score. The feature score is maximal if all grey levels are the same. Inverse difference is defined as:

$$F_{cm.inv.diff} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p_{ij}}{1 + |i-j|}$$

4.6.15 Inverse difference normalised

Clausi (Clausi, 2002) suggests normalising *inverse difference* to improve classification ability of this feature. This feature is then defined as:

$$F_{cm.inv.diff.norm} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p_{ij}}{1 + |i-j|/N_g}$$

Note that in Clausi's definition, $|i-j|^2/N_g^2$ is used instead of $|i-j|/N_g$, which is likely an oversight, as this exactly the same definition as the *inverse difference moment normalised* function.

4.6.16 Inverse difference moment

Inverse difference moment (Haralick et al., 1973) is similar in concept to the *inverse difference* feature, but with lower weights for elements that are further from the diagonal.

$$F_{cm.inv.diff.mom} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p_{ij}}{1 + (i-j)^2}$$

4.6.17 Inverse difference moment normalised

Clausi (Clausi, 2002) suggests normalising *inverse difference moment* to improve classification performance of this feature. This leads to the following definition:

$$F_{cm.inv.diff.mom.norm} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{p_{ij}}{1 + (i-j)^2 / N_g^2}$$

4.6.18 Inverse variance

The inverse variance feature is defined as:

$$F_{cm.inv.var} = 2 \sum_{i=1}^{N_g} \sum_{j>i}^{N_g} \frac{p_{ij}}{(i-j)^2}$$

4.6.19 Correlation

Correlation (Haralick et al., 1973) is defined as:

$$F_{cm.corr} = \frac{1}{\sigma_{i.} \, \sigma_{.j}} \left(-\mu_{i.} \, \mu_{.j} + \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i \, j \, p_{ij} \right)$$

$\mu_{i.} = \sum_{i=1}^{N_g} i \, p_{i.}$ and $\sigma_{i.} = \left(\sum_{i=1}^{N_g} (i - \mu_{i.})^2 p_{i.} \right)^{1/2}$ are the mean and standard deviation of row marginal probability $p_{i.}$, respectively. Likewise, $\mu_{.j}$ and $\sigma_{.j}$ are the mean and standard deviation column marginal probability $p_{.j}$, respectively. The equation for correlation can be simplified since $\mathbf{P}_\Delta$ is symmetrical:

$$F_{cm.corr} = \frac{1}{\sigma_{i.}^2} \left(-\mu_{i.}^2 + \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i \, j \, p_{ij} \right)$$

An equivalent formulation of *correlation* is:

$$F_{cm.corr} = \frac{1}{\sigma_{i.} \, \sigma_{.j}} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu_{i.}) \, (j - \mu_{.j}) \, p_{ij}$$

Again, simplifying due to matrix symmetry yields:

$$F_{cm.corr} = \frac{1}{\sigma_{i.}^2} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i - \mu_{i.}) \, (j - \mu_{i.}) \, p_{ij}$$

4.6.20 Autocorrelation

Aerts et al. (2014) defined autocorrelation as:

$$F_{cm.auto.corr} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i \, j \, p_{ij}$$

4.6.21 Cluster tendency

Cluster tendency is defined as:

$$F_{cm.clust.tend} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - \mu_{i.} - \mu_{.j})^2 \, p_{ij}$$

Here $\mu_{i.} = \sum_{i=1}^{N_g} i \, p_{i.}$ and $\mu_{.j} = \sum_{j=1}^{N_g} j \, p_{.j}$. Because of the symmetric nature of $\mathbf{P}_\Delta$, the feature can also be formulated as:

$$F_{cm.clust.tend} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - 2\mu_{i.})^2 \, p_{ij}$$

4.6.22 Cluster shade

Cluster shade is defined as (Unser, 1986):

$$F_{cm.clust.shade} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - \mu_{i.} - \mu_{.j})^3 \, p_{ij}$$

As with *cluster tendency*, $\mu_{i.} = \sum_{i=1}^{N_g} i \, p_{i.}$ and $\mu_{.j} = \sum_{j=1}^{N_g} j \, p_{.j}$. Because of the symmetric nature of $\mathbf{P}_\Delta$, the feature can also be formulated as:

$$F_{cm.clust.shade} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - 2\mu_{i.})^3 \, p_{ij}$$

4.6.23 Cluster prominence

Cluster prominence is defined as (Unser, 1986):

$$F_{cm.clust.prom} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - \mu_{i.} - \mu_{.j})^4 \, p_{ij}$$

As before, $\mu_{i.} = \sum_{i=1}^{N_g} i \, p_{i.}$ and $\mu_{.j} = \sum_{j=1}^{N_g} j \, p_{.j}$. Because of the symmetric nature of $\mathbf{P}_\Delta$, the feature can also be formulated as:

$$F_{cm.clust.prom} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (i + j - 2\mu_{i.})^4 \, p_{ij}$$

4.6.24 First measure of information correlation

Information theoretic correlation is estimated using two different measures (Haralick et al., 1973). For symmetric $\mathbf{P}_\Delta$ the first measure is defined as:

$$F_{cm.info.corr.1} = \frac{HXY - HXY_1}{HX}$$

$HXY = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij} \log_2 p_{ij}$ is the entropy for the joint probability. $HX = -\sum_{i=1}^{N_g} p_{i.} \log_2 p_{i.}$ is the entropy for the row marginal probability, which due to symmetry is equal to the entropy of the column marginal probability. HXY_1 is a type of entropy that is defined as:

$$HXY_1 = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij} \log_2 (p_{i.} p_{.j})$$

4.6.25 Second measure of information correlation

The second measure of information theoretic correlation is estimated as follows for symmetric $\mathbf{P}_\Delta$:

$$F_{cm.info.corr.2} = \sqrt{1 - \exp\left(-2\left(HXY_2 - HXY\right)\right)}$$

As earlier, $HXY = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij} \log_2 p_{ij}$. HXY_2 is a type of entropy defined as:

$$HXY_2 = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{i.} p_{.j} \log_2 (p_{i.} p_{.j})$$

If $HXY > HXY_2$, $F_{cm.info.corr.2} = 0$, as this would otherwise lead to complex numbers.

4.7 Textural features - Grey level run length based features

The grey level run length matrix (GLRLM) was introduced by Galloway (1975) to define various textural features. Like the grey level co-occurrence matrix, GLRLM also assesses the distribution of discretised grey levels in an image or in a stack of images. However, instead of assessing the combination of levels between neighbouring pixels or voxels, GLRLM assesses grey level run lengths. Run length counts the frequency of consecutive voxels with discretised grey level i along direction Δ.

A complete example for GLRLM construction from a 2D image is shown in Table 4.5. Let $\mathbf{M}_\Delta$ be the $N_g \times N_r$ grey level run length matrix, where N_g is the number of discretised grey levels present in the ROI intensity mask and N_r the maximal possible run length along direction Δ. The directions considered are the same as for GLCM. Matrix element $r_{ij} = r(i,j)$ is the number of occurrences where discretised grey level i appears in j consecutive neighbouring voxels or pixels. Then, let N_v be the total number of voxels in the ROI intensity mask, and $N_s = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} r_{ij}$ the sum over all elements in $\mathbf{M}_\Delta$. Marginal sums can also be defined. Let $r_{i.}$ be the marginal sum of the runs over run lengths j for grey value i, that is $r_{i.} = \sum_{j=1}^{N_r} r_{ij}$. The marginal sum of the runs over the grey values i for run length j is then $r_{.j} = \sum_{i=1}^{N_g} r_{ij}$.

Summarising features Feature values are calculated after calculating the grey level run length matrices within the ROI intensity mask. Five methods can be used to arrive at a single feature value for each volume. A schematic example was shown in Figure 4.2, and applies here as well. Three methods involve merging of matrices. By merging the number of runs for each individual combination of elements (i, j) in the GLRLMs is summed. Features are subsequently calculated from the merged matrix. Note that when matrices are combined, N_v should likewise be summed to retain consistency.

1	2	2	3
1	2	3	3
4	2	4	1
4	1	2	3

(a) Grey levels

		Run length j			
		1	2	3	4
	1	4	0	0	0
i	2	3	1	0	0
	3	2	1	0	0
	4	3	0	0	0

(b) $\mathbf{M}_{\Delta=\rightarrow}$

		Run length j			
		1	2	3	4
	1	4	0	0	0
i	2	3	1	0	0
	3	2	1	0	0
	4	3	0	0	0

(c) $\mathbf{M}_{\Delta=\nearrow}$

		Run length j			
		1	2	3	4
	1	2	1	0	0
i	2	2	0	1	0
	3	2	1	0	0
	4	1	1	0	0

(d) $\mathbf{M}_{\Delta=\uparrow}$

		Run length j			
		1	2	3	4
	1	4	0	0	0
i	2	3	1	0	0
	3	4	0	0	0
	4	3	0	0	0

(e) $\mathbf{M}_{\Delta=\nwarrow}$

Table 4.5: Grey level run length matrices for the $0°$ (a), $45°$ (b), $90°$ (c) and $135°$ (d) directions. In vector notation these directions are $\Delta = (1,0)$, $\Delta = (1,1)$, $\Delta = (0,1)$ and $\Delta = (-1,1)$

4.7.1   Short runs emphasis

This feature emphasises short run lengths (Galloway, 1975). It is defined as:

$$F_{rlm.sre} = \frac{1}{N_s} \sum_{j=1}^{N_r} \frac{r_{.j}}{j^2}$$

4.7.2   Long runs emphasis

This feature emphasises long run lengths (Galloway, 1975). It is defined as:

$$F_{rlm.lre} = \frac{1}{N_s} \sum_{j=1}^{N_r} j^2 r_{.j}$$

4.7.3 Low grey level run emphasis

This feature is a grey level analogue to $F_{rlm.sre}$ (Chu et al., 1990). Instead of low run lengths, low grey levels are emphasised. The feature is defined as:

$$F_{rlm.lgre} = \frac{1}{N_s} \sum_{i=1}^{N_g} \frac{r_{i.}}{i^2}$$

4.7.4 High grey level run emphasis

The high grey level run emphasis feature is a grey level analogue to $F_{rlm.lre}$ (Chu et al., 1990). The feature emphasises high grey levels, and is defined as:

$$F_{rlm.hgre} = \frac{1}{N_s} \sum_{i=1}^{N_g} i^2 r_{i.}$$

4.7.5 Short run low grey level emphasis

This feature emphasises runs in the upper left quadrant of the GLRLM, where short run lengths and low grey levels are located (Dasarathy and Holder, 1991). It is defined as:

$$F_{rlm.srlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} \frac{r_{ij}}{i^2 j^2}$$

4.7.6 Short run high grey level emphasis

This feature emphasises runs in the lower left quadrant of the GLRLM, where short run lengths and high grey levels are located (Dasarathy and Holder, 1991). The feature is defined as:

$$F_{rlm.srhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} \frac{i^2 r_{ij}}{j^2}$$

4.7.7 Long run low grey level emphasis

This feature emphasises runs in the upper right quadrant of the GLRLM, where long run lengths and low grey levels are located (Dasarathy and Holder, 1991). The feature is defined as:

$$F_{rlm.lrlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} \frac{j^2 r_{ij}}{i^2}$$

4.7.8 Long run high grey level emphasis

This feature emphasises runs in the lower right quadrant of the GLRLM, where long run lengths and high grey levels are located (Dasarathy and Holder, 1991). The feature is defined as:

$$F_{rlm.lrhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} i^2 j^2 r_{ij}$$

4.7.9   Grey level non-uniformity

This feature assesses the distribution of runs over the grey values (Galloway, 1975). The feature value is low when runs are equally distributed along grey levels. The feature is defined as:

$$F_{rlm.glnu} = \frac{1}{N_s} \sum_{i=1}^{N_g} r_{i.}^2$$

4.7.10   Grey level non-uniformity normalised

This is a normalised version of the grey level non-uniformity feature. It is defined as:

$$F_{rlm.glnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_g} r_{i.}^2$$

4.7.11   Run length non-uniformity

This features assesses the distribution of runs over the run lengths (Galloway, 1975). The feature value is low when runs are equally distributed along run lengths. It is defined as:

$$F_{rlm.rlnu} = \frac{1}{N_s} \sum_{j=1}^{N_r} r_{.j}^2$$

4.7.12   Run length non-uniformity normalised

This is normalised version of the run length non-uniformity feature. It is defined as:

$$F_{rlm.rlnu.norm} = \frac{1}{N_s^2} \sum_{j=1}^{N_r} r_{.j}^2$$

4.7.13   Run percentage

This feature assesses the fraction of the number of realised runs and the maximum number of potential runs (Galloway, 1975). Strongly linear or highly uniform ROI volumes produce a low run percentage. It is defined as:

$$F_{rlm.r.perc} = \frac{N_s}{N_v}$$

As noted above, when this feature is calculated using a merged GLRLM, N_v should be the sum of the number of voxels of the underlying matrices to allow proper normalisation.

4.7.14   Grey level variance

This feature estimates the variance in runs for the grey levels. Let $p_{ij} = r_{ij}/N_s$ be the joint probability estimate for finding discretised grey level i with run length j. The feature is then defined as:

$$F_{rlm.gl.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} (i - \mu)^2 p_{ij}$$

Here, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} i\, p_{ij}$.

4.7.15 Run length variance

This feature estimates the variance in runs for run lengths. As before let $p_{ij} = r_{ij}/N_s$. The feature is defined as:

$$F_{rlm.rl.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} (j - \mu)^2 p_{ij}$$

Mean run length is defined as $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} j\, p_{ij}$.

4.7.16 Run entropy

Run entropy was investigated by Albregtsen et al. (2000). Again, let $p_{ij} = r_{ij}/N_s$. The entropy is then defined as:

$$F_{rlm.rl.entr} = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p_{ij} \log_2 p_{ij}$$

4.8 Textural features - Grey level size zone based features

The grey level size zone matrix (GLSZM) counts the number of groups of connected voxels with a specific discretised grey level value and size (Thibault et al., 2014). Voxels are connected if the neighbouring voxel has the same discretised grey level value. Whether a voxel classifies as a neighbour depends on its connectedness. In the 3 dimensional approach to texture analysis we consider 26-connectedness, which indicates that a connection exists if any of the 26 neighbouring voxels shares the grey level of the center voxel. In the 2 dimensional approach, 8-connectedness is used. An issue with the 2 dimensional approach may be that voxels may be connected across slices, but disconnected within the plane of the slice. Whether this issue negatively affects predictive performance of GLSZM-based features has not been determined.

Let M be the $N_g \times N_z$ grey level size zone matrix, where N_g is the number of discretised grey levels present in the ROI intensity mask and N_z the maximum zone size of a group, or zone, of connected voxels with the same grey level value. Element $s_{ij} = s(i,j)$ of M is then number of zones with discretised grey level i and size j. Furthermore, let N_v be the number of voxels in the intensity mask and $N_s = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} s_{ij}$ be the total number of zones. Marginal sums can likewise be defined. Let $s_{i.} = \sum_{j=1}^{N_z} s_{ij}$ be the number of zones with discretised grey level i, regardless of size. Likewise, let $s_{.j} = \sum_{i=1}^{N_g} s_{ij}$ be the number of zones with size j, regardless of grey level. A two dimensional example is shown in Table 4.6.

Summarising features GLSZM feature definitions are based on the definitions of GLRLM features (Thibault et al., 2014). Feature values are calculated after calculating the grey level size zone matrices for the ROI intensity mask. Three methods can be used to arrive

at a single feature value for each volume. A schematic example is shown in Figure 4.3. One method involves merging of matrices. By merging the number of zones for each individual combination of elements (i, j) in the GLSZMs is summed. Features are subsequently calculated from the merged matrix. Note that when matrices are combined, N_v should likewise be summed to retain consistency.

	1	2	2	3
	1	2	3	3
	4	2	4	1
	4	1	2	3

(a) Grey levels

		Zone size j				
		1	2	3	4	5
i	1	2	1	0	0	0
	2	0	0	0	0	1
	3	1	0	1	0	0
	4	1	1	0	0	0

(b) Grey level size zone matrix

Table 4.6: Original image with grey levels (a); and corresponding grey level distance zone matrix (GLSZM) under 4-connectedness (b). Element $s(i, j)$ of the GLSZM indicates the number of times a zone of j pixels and grey level i occurs within the image.
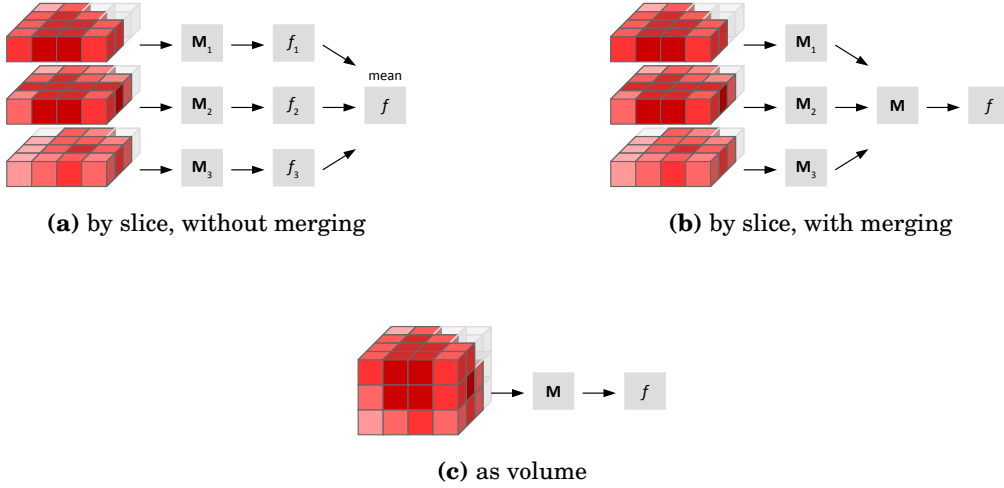

(a) by slice, without merging

(b) by slice, with merging

(c) as volume

Figure 4.3: Approaches to calculating grey level size zone matrix-based features. M_k are texture matrices calculated for slice k (if applicable), and f_k is the corresponding feature value. In (b) the matrices from the different slices are merged prior to feature calculation.

4.8.1 Small zone emphasis

This feature emphasises small zones. It is defined as:

$$F_{szm.sze} = \frac{1}{N_s} \sum_{j=1}^{N_z} \frac{s_{.j}}{j^2}$$

4.8.2 Large zone emphasis

This feature emphasises large zones. It is defined as:

$$F_{szm.lze} = \frac{1}{N_s} \sum_{j=1}^{N_z} j^2 s_{.j}$$

4.8.3 Low grey level zone emphasis

This feature is a grey level analogue to $F_{szm.sze}$. Instead of small zone sizes, low grey levels are emphasised. The feature is defined as:

$$F_{szm.lgze} = \frac{1}{N_s} \sum_{i=1}^{N_g} \frac{s_{i.}}{i^2}$$

4.8.4 High grey level zone emphasis

The high grey level zone emphasis feature is a grey level analogue to $F_{szm.lze}$. The feature emphasises high grey levels, and is defined as:

$$F_{szm.hgze} = \frac{1}{N_s} \sum_{i=1}^{N_g} i^2 s_{i.}$$

4.8.5 Small zone low grey level emphasis

This feature emphasises runs in the upper left quadrant of the GLSZM, where small zone sizes and low grey levels are located. It is defined as:

$$F_{szm.szlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} \frac{s_{ij}}{i^2 j^2}$$

4.8.6 Small zone high grey level emphasis

This feature emphasises runs in the lower left quadrant of the GLSZM, where small zone sizes and high grey levels are located. The feature is defined as:

$$F_{szm.szhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} \frac{i^2 s_{ij}}{j^2}$$

4.8.7 Large zone low grey level emphasis

This feature emphasises runs in the upper right quadrant of the GLSZM, where large zone sizes and low grey levels are located. The feature is defined as:

$$F_{szm.lzlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} \frac{j^2 s_{ij}}{i^2}$$

4.8.8 Large zone high grey level emphasis

This feature emphasises runs in the lower right quadrant of the GLSZM, where large zone sizes and high grey levels are located. The feature is defined as:

$$F_{szm.lzhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} i^2 j^2 s_{ij}$$

4.8.9 Grey level non-uniformity

This feature assesses the distribution of zone counts over the grey values. The feature value is low when zone counts are equally distributed along grey levels. The feature is defined as:

$$F_{szm.glnu} = \frac{1}{N_s} \sum_{i=1}^{N_g} s_{i.}^2$$

4.8.10 Grey level non-uniformity normalised

This is a normalised version of the grey level non-uniformity feature. It is defined as:

$$F_{szm.glnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_g} s_{i.}^2$$

4.8.11 Zone size non-uniformity

This features assesses the distribution of zone counts over the different zone sizes. The feature value is low when zone counts are equally distributed along zone sizes. It is defined as:

$$F_{szm.zsnu} = \frac{1}{N_s} \sum_{j=1}^{N_z} s_{.j}^2$$

4.8.12 Zone size non-uniformity normalised

This is a normalised version of the zone size non-uniformity feature. It is defined as:

$$F_{szm.zsnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_z} s_{.j}^2$$

4.8.13 Zone percentage

This feature assesses the fraction of the number of realised zones and the maximum number of potential zones. Strongly linear or highly uniform ROI volumes produce a low zone percentage. It is defined as:

$$F_{szm.z.perc} = \frac{N_s}{N_v}$$

4.8.14 Grey level variance

This feature estimates the variance in zone counts for the grey levels. Let $p_{ij} = s_{ij}/N_s$ be the joint probability estimate for finding discretised grey level i with zone size j. The feature is then defined as:

$$F_{szm.gl.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} (i - \mu)^2 p_{ij}$$

Here, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} i\, p_{ij}$.

4.8.15 Zone size variance

This feature estimates the variance in zone counts for the different zone sizes. As before let $p_{ij} = s_{ij}/N_s$. The feature is defined as:

$$F_{szm.zs.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} (j - \mu)^2 p_{ij}$$

Mean zone size is defined as $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} j\, p_{ij}$.

4.8.16 Zone size entropy

Let $p_{ij} = s_{ij}/N_s$. Zone size entropy is then defined as:

$$F_{szm.zs.entr} = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_z} p_{ij} \log_2 p_{ij}$$

4.9 Textural features - Grey level distance zone based features

The grey level distance zone matrix (GLDZM) counts the number of groups of connected voxels with a specific discretised grey level value and distance to ROI edge (Thibault et al., 2014). The matrix captures the relation between location and grey level. Two maps are required to calculate the GLDZM. The first is a grey level grouping map, which is identical to the one created for the grey level size zone matrix (GLSZM). The second is a distance map.

Voxels are connected if the neighbouring voxel has the same grey level value. Whether a voxel classifies as a neighbour depends on its connectedness. We consider 26-connectedness for a 3 dimensional approach and 8-connectedness in the 2 dimensional approach for consistency with the GLSZM.

The distance to ROI edge is defined in according to 6 or 4-connectedness. Because of the definition of connectedness that is used, the distance of a voxel to the outer border is equal to the minimum number of voxel edges that needs to be crossed along connected voxels to reach the ROI edge. The distance for a connected group of voxels with the same grey value equals the minimum distance in the respective voxels.

Our definition deviates from the original by Thibault et al. (2014). The original was defined in a rectangular 2D image, whereas ROIs are rarely rectangular cuboids. Approximating distance using Chamfer maps is then no longer a fast and easy solution. Determining distance in 6 or 4-connectedness is a relatively efficient solution, as one can step-wise determine distance by starting at the ROI edge and working inward. A second difference is that the lowest possible distance is 1 instead of 0 for voxels directly on the ROI edge. This prevents division by 0 for some features.

Let $\mathbf{D}$ be the $N_g \times N_d$ grey level size zone matrix, where N_g is the number of discretised grey levels present in the ROI intensity mask and N_d the maximum distance of a group, or zone, of connected voxels with the same discretised grey level value. Element $d_{ij} = d(i,j)$ of $\mathbf{D}$ is then number of zones with discretised grey level i and size j. Furthermore, let N_v be the number of voxels and $N_s = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} d_{ij}$ be the total number of zones. Marginal sums can likewise be defined. Let $d_{i.} = \sum_{j=1}^{N_d} d_{ij}$ be the number of zones with discretised grey level i, regardless of distance. Likewise, let $d_{.j} = \sum_{i=1}^{N_g} d_{ij}$ be the number of zones with size j, regardless of grey level. A two dimensional example is shown in Table 4.7.

Morphological and intensity masks   The morphological mask of the ROI is used to determine the distance map, whereas the intensity mask is used for determining the zones.

Summarising features   Feature values are calculated after calculating the grey level distance zone matrices. Three methods can be used to arrive at a single feature value for each volume. A schematic example was shown in Figure 4.3 and also applies here. One method involve merging of matrices. By merging the number of zones for each individual combination of elements (i,j) in the GLDZMs is summed. Features are subsequently calculated from the merged matrix. Note that when matrices are combined, N_v should likewise be summed to retain consistency.

1	2	2	3
1	2	3	3
4	2	4	1
4	1	2	3

(a) Grey levels

1	1	1	1
1	2	2	1
1	2	2	1
1	1	1	1

(b) Distance map

	d	
	1	2
i 1	3	0
2	2	0
3	2	0
4	1	1

(c) Grey level distance zone matrix

Table 4.7: Original image with grey levels (a); corresponding distance map for distance to border (b); and corresponding grey level distance zone matrix (GLDZM) under 4-connectedness (b). Element $d(i,j)$ of the GLDZM indicates the number of times a zone with grey level i and a minimum distance to border j occurs within the image.

4.9.1 Small distance emphasis

This feature emphasises small distances. It is defined as:

$$F_{dzm.sde} = \frac{1}{N_s} \sum_{j=1}^{N_d} \frac{d_{.j}}{j^2}$$

4.9.2 Large distance emphasis

This feature emphasises large distances. It is defined as:

$$F_{dzm.lde} = \frac{1}{N_s} \sum_{j=1}^{N_d} j^2 d_{.j}$$

4.9.3 Low grey level zone emphasis

This feature is a grey level analogue to $F_{dzm.sde}$. Instead of small zone distances, low grey levels are emphasised. The feature is defined as:

$$F_{dzm.lgze} = \frac{1}{N_s} \sum_{i=1}^{N_g} \frac{d_{i.}}{i^2}$$

4.9.4 High grey level zone emphasis

The high grey level zone emphasis feature is a grey level analogue to $F_{dzm.lde}$. The feature emphasises high grey levels, and is defined as:

$$F_{dzm.hgze} = \frac{1}{N_s} \sum_{i=1}^{N_g} i^2 d_{i.}$$

4.9.5 Small distance low grey level emphasis

This feature emphasises runs in the upper left quadrant of the GLDZM, where small zone distances and low grey levels are located. It is defined as:

$$F_{dzm.sdlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} \frac{d_{ij}}{i^2 j^2}$$

4.9.6 Small distance high grey level emphasis

This feature emphasises runs in the lower left quadrant of the GLDZM, where small zone distances and high grey levels are located. The feature is defined as:

$$F_{dzm.sdhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} \frac{i^2 d_{ij}}{j^2}$$

4.9.7 Large distance low grey level emphasis

This feature emphasises runs in the upper right quadrant of the GLDZM, where large zone distances and low grey levels are located. The feature is defined as:

$$F_{dzm.ldlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} \frac{j^2 d_{ij}}{i^2}$$

4.9.8 Large distance high grey level emphasis

This feature emphasises runs in the lower right quadrant of the GLDZM, where large zone distances and high grey levels are located. The feature is defined as:

$$F_{dzm.ldhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} i^2 j^2 d_{ij}$$

4.9.9 Grey level non-uniformity

This feature assesses the distribution of zone counts over the grey values. The feature value is low when zone counts are equally distributed along grey levels. The feature is defined as:

$$F_{dzm.glnu} = \frac{1}{N_s} \sum_{i=1}^{N_g} d_{i.}^2$$

4.9.10 Grey level non-uniformity normalised

This is a normalised version of the grey level non-uniformity feature. It is defined as:

$$F_{dzm.glnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_g} d_{i.}^2$$

4.9.11 Zone distance non-uniformity

This features assesses the distribution of zone counts over the different zone distances. The feature value is low when zone counts are equally distributed along zone distances. It is defined as:

$$F_{dzm.zdnu} = \frac{1}{N_s} \sum_{j=1}^{N_d} d_{.j}^2$$

4.9.12 Zone distance non-uniformity normalised

This is a normalised version of the zone distance non-uniformity feature. It is defined as:

$$F_{dzm.zdnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_d} d_{.j}^2$$

4.9.13 Zone percentage

This feature assesses the fraction of the number of realised zones and the maximum number of potential zones. Strongly linear or highly uniform ROI volumes produce a low zone percentage. It is defined as:

$$F_{dzm.z.perc} = \frac{N_s}{N_v}$$

4.9.14 Grey level variance

This feature estimates the variance in zone counts for the grey levels. Let $p_{ij} = d_{ij}/N_s$ be the joint probability estimate for finding zones with discretised grey level i at distance j. The feature is then defined as:

$$F_{dzm.gl.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} (i - \mu)^2 p_{ij}$$

Here, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} i\, p_{ij}$.

4.9.15 Zone distance variance

This feature estimates the variance in zone counts for the different zone distances. As before let $p_{ij} = d_{ij}/N_s$. The feature is defined as:

$$F_{dzm.zd.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} (j - \mu)^2 p_{ij}$$

Mean zone size is defined as $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} j\, p_{ij}$.

4.9.16 Zone distance entropy

Let $p_{ij} = d_{ij}/N_s$. Zone distance entropy is then defined as:

$$F_{dzm.zd.entr} = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_d} p_{ij} \log_2 p_{ij}$$

4.10 Textural features - Neighbourhood grey tone difference based features

Amadasun and King (1989) introduced an alternative grey level matrix. The neighbourhood grey tone difference matrix (NGTDM) contains the sum of grey level differences of pixels/voxels with discretised grey level i and the average discretised grey level of neighbouring pixels/voxels within a distance d. For 3D volumes, we can extend the original definition by Amadasun and King. Let $X_{dgl}(j_x, j_y, j_z)$ be the discretised grey level of a

voxel at position (j_x, j_y, j_z). Then the average grey level within a neighbourhood centred at (j_x, j_y, j_z), but excluding (j_x, j_y, j_z) itself is:

$$\overline{A}_i = \overline{A}(j_x, j_y, j_z)$$

$$= \frac{1}{W} \sum_{k_z=-d}^{d} \sum_{k_y=-d}^{d} \sum_{k_x=-d}^{d} X_{dgl}(j_x+k_x, j_y+k_y, j_z+k_z)$$

$$(k_x, k_y, k_z) \neq (0,0,0)$$

$W = (2d + 1)^3 - 1$ is the size of the neighbourhood. Let n_i be the number of voxels with discretised grey level i that have a complete neighbourhood. Then the entry in the grey tone difference matrix for grey level i is:

$$s_i = \begin{cases} \sum^{n_i} |i - \overline{A}_i| & \text{for } n_i > 0 \\ 0 & \text{for } n_i = 0 \end{cases}$$

A 2D example is shown in Table 4.8. $d = 1$ is used in this example, leading to 8 neighbouring pixels. $s_1 = 0$ because there are no valid pixels with grey level 1. Two pixels have grey level 2. The average value of their neighbours are $19/8$ and $21/8$. Thus $s_2 = |2 - 19/8| + |2 - 21/8| = 1$. Similarly $s_3 = |3 - 19/8| = 0.625$ and $s_4 = |4 - 17/8| = 1.825$.

Note that for practical purposes, we do not require a complete neighbourhood of valid voxels in irregularly shaped volumes. Instead, W is equal to the number of valid voxels in the neighbourhood. Missing voxels receive $X_{dgl} = 0$, and are not counted to determine $\overline{A}_i$. n_i is then the total number of voxels with grey level i. Voxels without any neighbours are not counted.

Many NGTDM-based features depend on the N_g grey level probabilities $p_i = n_i/N_v$, where N_g is the number of discretised grey levels in the ROI intensity mask and $N_v = \sum n_i$. Furthermore, let $N_{g,p} \leq N_g$ be the number of discretised grey levels with $p_i > 0$. In the above example, $N_g = 4$ and $N_{g,p} = 3$.

Summarising features Feature values are calculated after calculating the neighbourhood grey tone difference matrices for the ROI intensity mask. Three methods can be used to arrive at a single feature value for each volume. A schematic example was shown in Figure 4.3, and applies here as well. One method involves merging of matrices. By merging the grey tone difference s_i is summed over the different NGTDMs. Features are subsequently calculated from the merged matrix. Note that when matrices are combined, N_v should likewise be summed to retain consistency.

4.10.1 Coarseness

Grey level differences in coarse textures are generally small due to large-scale patterns. Summing differences gives an indication of the level of the spatial rate of change in intensity (Amadasun and King, 1989). Coarseness is defined as:

$$F_{ngt.coarseness} = \frac{1}{\sum_{i=1}^{N_g} p_i \, s_i}$$

Because $\sum_{i=1}^{N_g} p_i \, s_i$ potentially evaluates to 0, the maximum value of $F_{ngt.coarseness}$ is set to an arbitrary number of 10^6. Amadasun and King originally circumvented this issue by adding a unspecified small number ϵ to the denominator, but an explicit, though arbitrary, maximum value should allow for more consistency.

1	2	2	3
1	2	3	3
4	2	4	1
4	1	2	3

(a) Grey levels

i		n_i	p_i	s_i
	1	0	0.00	0.000
	2	2	0.50	1.000
	3	1	0.25	0.625
	4	1	0.25	1.825

(b) Neighbourhood grey tone difference matrix

Table 4.8: Original image with grey levels (a) and corresponding neighbourhood grey tone difference matrix (NGTDM) (b). The N_v pixels with valid neighbours at distance 1 are located within the rectangle in (a). The grey level count n_i, the grey level probability $p_i = n_i/N_v$, and the neighbourhood average grey level s_i for pixels with grey level i. Note that our definition deviates from the original definition of Amadasun and King (1989), which is presented in panels a and b. In our definition fully valid neighbourhood is no longer required. The NGTDM is thus calculated on the entire pixel area, and not solely on those pixels within the rectangle of panel a.

4.10.2 Contrast

Contrast depends on the dynamic range of the grey levels as well as the spatial frequency of intensity changes (Amadasun and King, 1989). Thus, contrast is defined as:

$$F_{ngt.contrast} = \left(\frac{1}{N_{g,p}\left(N_{g,p}-1\right)} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_i p_j \left(i-j\right)^2 \right) \left(\frac{1}{N_v} \sum_{i=1}^{N_g} s_i \right)$$

Grey level probabilities p_i and p_j are copies with different iterators, i.e. $p_i = p_j$ for $i = j$. The first term considers the grey level dynamic range, whereas the second term is a measure for intensity changes within the volume. If $N_{g,p} = 1$, $F_{ngt.contrast} = 0$.

4.10.3 Busyness

Textures with large changes in grey levels between neighbouring voxels are called busy (Amadasun and King, 1989). Busyness was defined as:

$$F_{ngt.busyness} = \frac{\sum_{i=1}^{N_g} p_i \, s_i}{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i p_i - j p_j}, \qquad p_i \neq 0 \text{ and } p_j \neq 0$$

As before, $p_i = p_j$ for $i = j$. The original definition was erroneously formulated as the denominator will always evaluate to 0. Therefore we use a slightly different definition:

$$F_{ngt.busyness} = \frac{\sum_{i=1}^{N_g} p_i \, s_i}{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} |ip_i - jp_j|}, \qquad p_i \neq 0 \text{ and } p_j \neq 0$$

If $N_{g,p} = 1$, $F_{ngt.busyness} = 0$.

4.10.4 Complexity

Complex textures are non-uniform and rapid changes in grey levels are common (Amadasun and King, 1989). Texture complexity is defined as:

$$F_{ntg.complexity} = \frac{1}{N_v} \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} |i - j| \frac{p_i \, s_i + p_j \, s_j}{p_i + p_j}, \qquad p_i \neq 0 \text{ and } p_j \neq 0$$

As before, $p_i = p_j$ for $i = j$, and likewise $s_i = s_j$ for $i = j$.

4.10.5 Strength

Amadasun and King (1989) defined texture strength as:

$$F_{ngt.strength} = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} (p_i + p_j) \, (i - j)^2}{\sum_{i=1}^{N_g} s_i}, \qquad p_i \neq 0 \text{ and } p_j \neq 0$$

As before, $p_i = p_j$ for $i = j$. If $\sum_{i=1}^{N_g} s_i = 0$, $F_{ngt.strength} = 0$.

4.11 Textural features - Neighbouring grey level dependence based features

Sun and Wee (1983) defined the neighbouring grey level dependence matrix (NGLDM) as an alternative to the grey level co-occurrence matrix. The NGLDM captures the coarseness of the overall texture and is rotationally invariant.

To construct the NGLDM a neighbourhood is defined as the voxels located within a distance d around a center voxel. The discretised grey levels of the center voxel c and a neighbouring voxel m are said to be dependent if $|X_{gl,c} - X_{gl,m}| \leq a$, with a being a positive integer coarseness parameter. The number of grey level dependent voxels in the neighbourhood are then counted. This iteratively done for every voxel. M is then the $N_g \times N_n$ neighbouring grey level dependence matrix, where N_g is the number of discretised grey levels present in the ROI intensity mask and N_n the maximum grey level dependence count present. Element $s_{ij} = s(i,j)$ of M is then the number of neighbourhoods with center voxel discretised grey level i and dependence $k = j - 1$. Since a dependence $k = 0$ is possible, iterator j has to be explicitly defined as $j = k + 1$. Furthermore, let N_v be the number of voxels in the ROI intensity mask, and $N_s = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} s_{ij}$ the number of neighbourhoods. Marginal sums can likewise be defined. Let $s_{i.} = \sum_{j=1}^{N_n}$ be the number of neighbourhoods with discretised grey level i, essentially constituting a grey level histogram of the volume

or slice. Let $s_{j.} = \sum_{i=1}^{N_g} s_{ij}$ be the number of neighbourhoods with dependence j, regardless of grey level. A two dimensional example is shown in Table 4.9.

The definition we use deviates from the original by Sun and Wee (1983). Because regions of interest are rarely cuboid, omission of neighbourhoods which contain voxels beyond the ROI edge may lead to inconsistent results, especially for larger distance d. Hence the neighbourhoods of all voxels in the within the ROI intensity mask are considered, and consequently $N_v = N_s$. Neighbourhood voxels located outside the ROI do not add to dependence j. Thus, small and tortuous ROI will be considered coarser as neighbouring grey level dependence j is low.

Note that while $a = 0$ is a typical choice for the coarseness parameter, different a are possible. For consistency it may be beneficial to keep $a = 0$ and change grey levels through a discretisation procedure. Likewise, a typical choice for neighbourhood radius is Chebyshev distance $d = 1$, which is effectively the same as Eucledian distance $d = \sqrt{3}$, but larger values may be useful as well.

The NGLDM-based features are analogous to that of the grey level run length matrix, grey level size zone matrix and grey level distance zone matrix, and expand on the set defined by Sun and Wee (1983).

Summarising features Feature values are calculated after calculating the neighbouring grey level dependence matrices for the ROI intensity mask. Three methods can be used to arrive at a single feature value for each volume. A schematic example was shown in Figure 4.3, and applies here as well. One method involve merging of matrices. By merging the number of neighbourhoods for each individual combination of elements (i, j) in the NGLDMs is summed. Features are subsequently calculated from the merged matrix.

					dependence k			
					0	1	2	3
1	2	2	3	1	0	0	0	0
1	2	3	3	2	0	0	1	1
4	2	4	1	3	0	0	1	0
4	1	2	3	4	1	0	0	0

(a) Grey levels   **(b)** Neighbouring grey level dependence matrix

(the left table has i label on the right block)

Table 4.9: Original image with grey levels and pixels with a complete neighbourhood within the square (a); corresponding neighbouring grey level dependence matrix for distance $d = \sqrt{2}$ and coarseness parameter $a = 0$ (b). Element $s(i, j)$ of the NGLDM indicates the number of neighbourhoods with a center pixel with grey level i and neighbouring grey level dependence k within the image. Note that in practice we no longer require a complete neighbourhood. Thus every voxel is considered as a center voxel with a neighbourhood, instead of being constrained to the voxels within the square in panel (a).

4.11.1 Low dependence emphasis

This feature emphasises low neighbouring grey level dependence counts. Sun and Wee (1983) refer to this feature as *Small number emphasis*. It is defined as:

$$F_{ngl.lde} = \frac{1}{N_s} \sum_{j=1}^{N_n} \frac{s_{.j}}{j^2}$$

4.11.2 High dependence emphasis

This feature emphasises high neighbouring grey level dependence counts. Sun and Wee (1983) refer to this feature as *Large number emphasis*. It is defined as:

$$F_{ngl.hde} = \frac{1}{N_s} \sum_{j=1}^{N_n} j^2 s_{.j}$$

4.11.3 Low grey level count emphasis

This feature is a grey level analogue to $F_{ngl.lde}$. Instead of low neighbouring grey level dependence counts, low grey levels are emphasised. The feature is defined as:

$$F_{ngl.lgce} = \frac{1}{N_s} \sum_{i=1}^{N_g} \frac{s_{i.}}{i^2}$$

4.11.4 High grey level count emphasis

The high grey level count emphasis feature is a grey level analogue to $F_{ngl.hde}$. The feature emphasises high grey levels, and is defined as:

$$F_{ngl.hgce} = \frac{1}{N_s} \sum_{i=1}^{N_g} i^2 s_{i.}$$

4.11.5 Low dependence low grey level emphasis

This feature emphasises neighbouring grey level dependence counts in the upper left quadrant of the NGLDM, where low dependence counts and low grey levels are located. It is defined as:

$$F_{ngl.ldlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} \frac{s_{ij}}{i^2 j^2}$$

4.11.6 Low dependence high grey level emphasis

This feature emphasises neighbouring grey level dependence counts in the lower left quadrant of the NGLDM, where low dependence counts and high grey levels are located. The feature is defined as:

$$F_{ngl.ldhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} \frac{i^2 s_{ij}}{j^2}$$

4.11.7 High dependence low grey level emphasis

This feature emphasises neighbouring grey level dependence counts in the upper right quadrant of the NGLDM, where high dependence counts and low grey levels are located. The feature is defined as:

$$F_{ngl.hdlge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} \frac{j^2 s_{ij}}{i^2}$$

4.11.8 High dependence high grey level emphasis

This feature emphasises neighbouring grey level dependence counts in the lower right quadrant of the NGLDM, where high dependence counts and high grey levels are located. The feature is defined as:

$$F_{ngl.hdhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} i^2 j^2 s_{ij}$$

4.11.9 Grey level non-uniformity

This feature assesses the distribution of neighbouring grey level dependence counts over the grey values. The feature value is low when dependence counts are equally distributed along grey levels. The feature is defined as:

$$F_{ngl.glnu} = \frac{1}{N_s} \sum_{i=1}^{N_g} s_{i.}^2$$

4.11.10 Grey level non-uniformity normalised

This is a normalised version of the grey level non-uniformity feature. It is defined as:

$$F_{ngl.glnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_g} s_{i.}^2$$

4.11.11 Dependence count non-uniformity

This features assesses the distribution of neighbouring grey level dependence counts over the different dependence counts. The feature value is low when dependence counts are equally distributed. Sun and Wee (1983) refer to this feature as *Number nonuniformity*. It is defined as:

$$F_{ngl.dcnu} = \frac{1}{N_s} \sum_{j=1}^{N_n} s_{.j}^2$$

4.11.12 Dependence count non-uniformity normalised

This is a normalised version of the dependence count non-uniformity feature. It is defined as:

$$F_{ngl.dcnu.norm} = \frac{1}{N_s^2} \sum_{i=1}^{N_n} s_{.j}^2$$

4.11.13 Dependence count percentage

This feature assesses the fraction of the number of realised neighbourhoods and the maximum number of potential neighbourhoods. The feature may be omitted as it evaluates to 1 when complete neighbourhoods are not required, which is the case under our definition. It is defined as:

$$F_{ngl.dc.perc} = \frac{N_s}{N_v}$$

4.11.14 Grey level variance

This feature estimates the variance in dependence counts for the grey levels. Let $p_{ij} = s_{ij}/N_s$ be the joint probability estimate for finding discretised grey level i with dependence j. The feature is then defined as:

$$F_{ngl.gl.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} (i - \mu)^2 p_{ij}$$

Here, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} i\, p_{ij}$.

4.11.15 Dependence count variance

This feature estimates the variance in dependence counts for the different dependence counts possible. As before let $p_{ij} = s_{ij}/N_s$. The feature is defined as:

$$F_{ngl.dc.var} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} (j - \mu)^2 p_{ij}$$

Mean dependence count is defined as $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} j\, p_{ij}$.

4.11.16 Dependence count entropy

This feature is referred to as *Entropy* by Sun and Wee (1983). Let $p_{ij} = s_{ij}/N_s$. Dependence count entropy is then defined as:

$$F_{ngl.dc.entr} = -\sum_{i=1}^{N_g} \sum_{j=1}^{N_n} p_{ij} \log_2 p_{ij}$$

This definition remedies an error in the original definition, where the term within the logarithm is dependence count s_{ij} instead of count probability p_{ij}.

4.11.17 Dependence count energy

This feature is called *second moment* by Sun and Wee (1983). Let $p_{ij} = s_{ij}/N_s$. Then dependence count energy is defined as:

$$F_{ngl.dc.energy} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} p_{ij}^2$$

This definition remedies an error in the original definition. There squared dependence count s_{ij}^2 is normalised only by N_s, thus leaving a major volume dependency. Here we effectively normalise s_{ij}^2 by N_s^2.

Chapter 5

Novel and uncommon imaging features

Several improvements to existing methods, and less commonly used image feature methods are described in this section.

5.1 Textural features - Distance weighted texture matrices

Changes to grey level co-occurrence, run length and neighbourhood grey tone difference matrices were suggested by Vallières et al. (2015). The contents of these matrices can be updated based on the distance between center voxels and reference voxels (GLCM, NGTDM) and the length of runs. Different definitions of distance exist and can be used. Let $\mathbf{k} = (k_x, k_y, k_z)$ be the vector from a center voxel at $\mathbf{j} = (j_x, j_y, j_z)$ to a neighbour voxel at $\mathbf{j} + \mathbf{k}$. For GLCM and GLRLM $\mathbf{k}$ is the direction vector. Depending on definition of the norm, distance is defined as:

- ℓ_1 norm or *Manhattan* norm:

$$\delta_1 = |k_x| + |k_y| + |k_z|$$

- ℓ_2 norm or *Euclidian* norm:

$$\delta_2 = \sqrt{k_x^2 + k_y^2 + k_z^2}$$

- ℓ_∞ norm or *Chebyshev* norm:

$$\delta_\infty = \max(|k_x|, |k_y|, |k_z|)$$

Weighting factors may be defined to either correct distance or emphasise local textures: $w(\delta(\mathbf{k}))$. For conventional texture matrices $w = 1$.

Grey level co-occurrence matrix Correcting the GLCM for distance takes place by weighting each GLCM with distance $\delta(\mathbf{k})$ using e.g. the Euclidian or Manhattan norm. Using the Chebyshev norm conforms to the original definition and has no effect. Local

textures may furthermore be emphasised using an inverse distance function to weight each GLCM, e.g. $w = \delta^{-1}$ or $\exp(-\delta^2)$ (van Griethuysen et al., 2017).

Weighting the GLCM is only meaningful if GLCM matrices for all directions are merged into a single GLCM before feature calculation.

Grey level run length matrix The correction to the GLRLM matrix consists of replacing the run length by the actual distance, i.e. run length $j^* = \delta(\mathbf{k})j$. As with GLCM, the Chebyshev norm reverts to the original definition for GLRLM. Weighting is only meaningful if GLRLM matrices for all directions are merged before feature calculation.

Neighbourhood grey tone difference matrix The neighbourhood grey tone difference may be corrected for distance or locally emphasised. The calculation for the average grey level in the neighbourhood centred at $\mathbf{j}$ may be updated:

$$
\begin{aligned}
\overline{A}_i &= \overline{A}(\mathbf{j}) \\
&= \frac{1}{W} \sum_{k_z=-d}^{d} \sum_{k_y=-d}^{d} \sum_{k_x=-d}^{d} w(\delta(\mathbf{k})) X_{gl}(\mathbf{j}+\mathbf{k}) \qquad \mathbf{k} \neq (0,0,0)
\end{aligned}
$$

Here $W = w(\delta(\mathbf{k}))$. For $w = 1$, the NGTDM reverts to its original.

5.2 Textural features - Neighbourhood co-occurrence matrix

The co-occurrence matrix is defined along specific directions. The neighbourhood co-occurrence matrix is a modification which considers the neighbourhood of a voxel to consist of all voxels within distance d. The remaining calculations are performed as for the conventional GLCM.

5.3 Textural features - Extended emphasis features

The emphasis type features in the grey level run length, grey level size zone, grey level distance zone and neighbouring grey level dependence matrix-based feature sets put a particular emphasis on part of the respective matrices. They do so using a combination of squared indices. However, it should be noted that different emphasis can be put using indices with different powers (van Dijk et al., 2017). For example, the *long run high grey level emphasis* feature is defined as:

$$
F_{rlm.lrhge} = \frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} i^2 j^2 r_{ij}
$$

Changes powers to the indices i and j yields new features, i.e. $\frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} i^3 j\, r_{ij}$ and $\frac{1}{N_s} \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} i\, j^3 r_{ij}$ constitute new features with a different emphasis on run lengths and grey levels. We can therefore generate a large number of features by changing powers.

Nomenclature   Many emphasis features will not have existing names. What is proposed therefore is to encode the respective powers in the name as *emphasis*(power of index i, power of index j). This naming scheme, when applied to grey level run length matrix-based features leads to, e.g. *emphasis*(2,2) for *long run high grey level emphasis*, *emphasis*(0,-2) for *short runs emphasis* and *emphasis*(-2,2) for *long run low grey level emphasis*.

Bibliography

Aerts, H. J. W. L., Rios-Velazquez, E., Leijenaar, R. T. H., Parmar, C., Grossmann, P., Cavalho, S., Bussink, J., Monshouwer, R., Haibe-Kains, B., Rietveld, D., Hoebers, F. J. P., Rietbergen, M. M., Leemans, C. R., Dekker, A., Quackenbush, J., Gillies, R. J., and Lambin, P. (2014). Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. *Nature communications*, 5:4006.

Ahipaaolu, S. D. (2015). Fast algorithms for the minimum volume estimator. *Journal of Global Optimization*, 62(2):351–370.

Albregtsen, F., Nielsen, B., and Danielsen, H. (2000). Adaptive gray level run length features from class distance matrices. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, volume 3, pages 738–741. IEEE Comput. Soc.

Amadasun, M. and King, R. (1989). Textural features corresponding to textural properties. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1264–1273.

Barequet, G. and Har-Peled, S. (2001). Efficiently Approximating the Minimum-Volume Bounding Box of a Point Set in Three Dimensions. *Journal of Algorithms*, 38(1):91–109.

Boellaard, R., Delgado-Bolton, R., Oyen, W. J. G., Giammarile, F., Tatsch, K., Eschner, W., Verzijlbergen, F. J., Barrington, S. F., Pike, L. C., Weber, W. A., Stroobants, S. G., Delbeke, D., Donohoe, K. J., Holbrook, S., Graham, M. M., Testanera, G., Hoekstra, O. S., Zijlstra, J. M., Visser, E. P., Hoekstra, C. J., Pruim, J., Willemsen, A. T., Arends, B., Kotzerke, J., Bockisch, A., Beyer, T., Chiti, A., and Krause, B. J. (2015). FDG PET/CT: EANM procedure guidelines for tumour imaging: version 2.0. *European journal of nuclear medicine and molecular imaging*, 42(2):328–54.

Boellaard, R., O'Doherty, M. J., Weber, W. A., Mottaghy, F. M., Lonsdale, M. N., Stroobants, S. G., Oyen, W. J. G., Kotzerke, J., Hoekstra, O. S., Pruim, J., Marsden, P. K., Tatsch, K., Hoekstra, C. J., Visser, E. P., Arends, B., Verzijlbergen, F. J., Zijlstra, J. M., Comans, E. F. I., Lammertsma, A. A., Paans, A. M., Willemsen, A. T., Beyer, T., Bockisch, A., Schaefer-Prokop, C., Delbeke, D., Baum, R. P., Chiti, A., and Krause, B. J. (2010). FDG PET and PET/CT: EANM procedure guidelines for tumour PET imaging: version 1.0. *European Journal of Nuclear Medicine and Molecular Imaging*, 37(1):181–200.

Boussion, N., Le Rest, C. C., Hatt, M., and Visvikis, D. (2009). Incorporation of wavelet-based denoising in iterative deconvolution for partial volume correction in whole-body PET imaging. *European journal of nuclear medicine and molecular imaging*, 36(7):1064–75.

Chan, C. and Tan, S. (2001). Determination of the minimum bounding box of an arbitrary solid: an iterative approach. *Computers and Structures*, 79(15):1433–1449.

Chu, A., Sehgal, C. M., and Greenleaf, J. F. (1990). Use of gray value distribution of run lengths for texture analysis. *Pattern Recognition Letters*, 11(6):415–419.

Clausi, D. A. (2002). An analysis of co-occurrence texture statistics as a function of grey level quantization. *Canadian Journal of Remote Sensing*, 28(1):45–62.

Collewet, G., Strzelecki, M., and Mariette, F. (2004). Influence of MRI acquisition protocols and image intensity normalization methods on texture classification. *Magnetic resonance imaging*, 22(1):81–91.

Da Silva, E. C., Silva, A. C., De Paiva, A. C., and Nunes, R. A. (2008). Diagnosis of lung nodule using Moran's index and Geary's coefficient in computerized tomography images. *Pattern Analysis and Applications*, 11(1):89–99.

Dale, M. R. T., Dixon, P., Fortin, M.-J., Legendre, P., Myers, D. E., and Rosenberg, M. S. (2002). Conceptual and mathematical relationships among methods for spatial analysis. *Ecography*, 25(5):558–577.

Dasarathy, B. V. and Holder, E. B. (1991). Image characterizations based on joint gray levelrun length distributions. *Pattern Recognition Letters*, 12(8):497–502.

El Naqa, I., Grigsby, P. W., Apte, A., Kidd, E., Donnelly, E., Khullar, D., Chaudhari, S., Yang, D., Schmitt, M., Laforest, R., Thorstad, W. L., and Deasy, J. O. (2009). Exploring feature-based approaches in PET images for predicting cancer treatment outcomes. *Pattern recognition*, 42(6):1162–1171.

Galloway, M. M. (1975). Texture analysis using gray level run lengths. *Computer Graphics and Image Processing*, 4(2):172–179.

Geary, R. C. (1954). The Contiguity Ratio and Statistical Mapping. *The Incorporated Statistician*, 5(3):115–145.

Gillies, R. J., Kinahan, P. E., and Hricak, H. (2015). Radiomics: Images Are More than Pictures, They Are Data. *Radiology*, 278(2):151169.

Gjesteby, L., De Man, B., Jin, Y., Paganetti, H., Verburg, J., Giantsoudi, D., and Wang, G. (2016). Metal Artifact Reduction in CT: Where Are We After Four Decades? *IEEE Access*, 4:5826–5849.

Gudbjartsson, H. and Patz, S. (1995). The Rician distribution of noisy MRI data. *Magnetic resonance in medicine*, 34(6):910–4.

Hall, E. L., Kruger, R. P., Samuel, J., Dwyer, D., McLaren, R. W., Hall, D. L., and Lodwick, G. (1971). A Survey of Preprocessing and Feature Extraction Techniques for Radiographic Images. *IEEE Transactions on Computers*, C-20(9):1032–1044.

Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(6):610–621.

Hatt, M., Majdoub, M., Vallières, M., Tixier, F., Le Rest, C. C., Groheux, D., Hindié, E., Martineau, A., Pradier, O., Hustinx, R., Perdrisot, R., Guillevin, R., El Naqa, I., and Visvikis, D. (2015). 18F-FDG PET uptake characterization through texture analysis: investigating the complementary nature of heterogeneity and functional tumor volume in a multi-cancer site patient cohort. *Journal of nuclear medicine*, 56(1):38–44.

Hatt, M., Tixier, F., Pierce, L., Kinahan, P. E., Le Rest, C. C., and Visvikis, D. (2016). Characterization of PET/CT images using texture analysis: the past, the present... any future? *European journal of nuclear medicine and molecular imaging*.

Heiberger, R. M. and Holland, B. (2015). *Statistical Analysis and Data Display*. Springer Texts in Statistics. Springer New York, New York, NY.

Janmahasatian, S., Duffull, S. B., Ash, S., Ward, L. C., Byrne, N. M., and Green, B. (2005). Quantification of lean bodyweight. *Clinical pharmacokinetics*, 44(10):1051–65.

Khachiyan, L. G. (1996). Rounding of Polytopes in the Real Number Model of Computation. *Mathematics of Operations Research*, 21(2):307–320.

Kumar, V., Gu, Y., Basu, S., Berglund, A. E., Eschrich, S. A., Schabath, M. B., Forster, K., Aerts, H. J. W. L., Dekker, A., Fenstermacher, D., Goldgof, D. B., Hall, L. O., Lambin, P., Balagurunathan, Y., Gatenby, R. A., and Gillies, R. J. (2012). Radiomics: the process and the challenges. *Magnetic Resonance Imaging*, 30(9):1234–1248.

Lambin, P., Rios-Velazquez, E., Leijenaar, R. T. H., Carvalho, S., van Stiphout, R. G. P. M., Granton, P., Zegers, C. M. L., Gillies, R. J., Boellard, R., Dekker, A., and Aerts, H. J. W. L. (2012). Radiomics: Extracting more information from medical images using advanced feature analysis. *European Journal of Cancer*, 48(4):441–446.

Leijenaar, R. T. H., Nalbantov, G., Carvalho, S., van Elmpt, W. J. C., Troost, E. G. C., Boellaard, R., Aerts, H. J. W. L., Gillies, R. J., and Lambin, P. (2015). The effect of SUV discretization in quantitative FDG-PET Radiomics: the need for standardized methodology in tumor texture analysis. *Scientific reports*, 5(August):11075.

Lewiner, T., Lopes, H., Vieira, A. W., and Tavares, G. (2003). Efficient Implementation of Marching Cubes' Cases with Topological Guarantees. *Journal of Graphics Tools*, 8(2):1–15.

Lloyd, S. P. (1982). Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.

Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169.

Max, J. (1960). Quantizing for minimum distortion. *IEEE Transactions on Information Theory*, 6(1):7–12.

Mazurowski, M. A., Czarnek, N. M., Collins, L. M., Peters, K. B., and Clark, K. (2016). Predicting outcomes in glioblastoma patients using computerized analysis of tumor shape: preliminary data. In Tourassi, G. D. and Armato, S. G., editors, *SPIE Medical Imaging*, volume 9785, page 97852T.

Moran, P. A. P. (1950). Notes on continuous stochastic phenomena. *Biometrika*, 37:17–23.

O'Rourke, J. (1985). Finding minimal enclosing boxes. *International Journal of Computer and Information Sciences*, 14(3):183–199.

Schirra, S. (2008). How Reliable Are Practical Point-in-Polygon Strategies? In *Algorithms - ESA 2008*, pages 744–755. Springer Berlin Heidelberg, Berlin, Heidelberg.

Shafiq-ul Hassan, M., Zhang, G. G., Latifi, K., Ullah, G., Hunt, D. C., Balagurunathan, Y., Abdalah, M. A., Schabath, M. B., Goldgof, D. G., Mackin, D., Court, L. E., Gillies, R. J., and Moros, E. G. (2017). Intrinsic dependencies of CT radiomic features on voxel size and number of gray levels. *Medical Physics*.

Sled, J. G., Zijdenbos, A. P., and Evans, A. C. (1998). A nonparametric method for automatic correction of intensity nonuniformity in MRI data. *IEEE transactions on medical imaging*, 17(1):87–97.

Soret, M., Bacharach, S. L., and Buvat, I. (2007). Partial-volume effect in PET tumor imaging. *Journal of nuclear medicine*, 48(6):932–45.

Stelldinger, P., Latecki, L. J., and Siqueira, M. (2007). Topological equivalence between a 3D object and the reconstruction of its digital image. *IEEE transactions on pattern analysis and machine intelligence*, 29(1):126–40.

Sun, C. and Wee, W. G. (1983). Neighboring gray level dependence matrix for texture classification. *Computer Vision, Graphics, and Image Processing*, 23(3):341–352.

Thibault, G., Angulo, J., and Meyer, F. (2014). Advanced statistical matrices for texture characterization: application to cell classification. *IEEE transactions on bio-medical engineering*, 61(3):630–7.

Todd, M. J. and Yldrm, E. A. (2007). On Khachiyan's algorithm for the computation of minimum-volume enclosing ellipsoids. *Discrete Applied Mathematics*, 155(13):1731–1744.

Unser, M. (1986). Sum and difference histograms for texture classification. *IEEE transactions on pattern analysis and machine intelligence*, 8(1):118–125.

Vaidya, M., Creach, K. M., Frye, J., Dehdashti, F., Bradley, J. D., and El Naqa, I. (2012). Combined PET/CT image characteristics for radiotherapy tumor response in lung cancer. *Radiotherapy and oncology*, 102(2):239–45.

Vallières, M., Freeman, C. R., Skamene, S. R., and El Naqa, I. (2015). A radiomics model from joint FDG-PET and MRI texture features for the prediction of lung metastases in soft-tissue sarcomas of the extremities. *Physics in medicine and biology*, 60(14):5471–96.

van Dijk, L. V., Brouwer, C. L., van der Schaaf, A., Burgerhof, J. G., Beukinga, R. J., Langendijk, J. A., Sijtsema, N. M., and Steenbakkers, R. J. (2017). CT image biomarkers to improve patient-specific prediction of radiation-induced xerostomia and sticky saliva. *Radiotherapy and Oncology*, 122(2):185–191.

van Griethuysen, J. J. M., Fedorov, A., Parmar, C., Hosny, A., Aucoin, N., Narayan, V., Beets-Tan, R. G. H., Fillion-Robin, J.-C., Pieper, S., and Aerts, H. J. W. L. (2017). Computational Radiomics System to Decode the Radiographic Phenotype. *Submitted for publication*.

van Velden, F. H. P., Cheebsumon, P., Yaqub, M., Smit, E. F., Hoekstra, O. S., Lammertsma, A. A., and Boellaard, R. (2011). Evaluation of a cumulative SUV-volume histogram method for parameterizing heterogeneous intratumoural FDG uptake in non-small cell lung cancer PET studies. *European journal of nuclear medicine and molecular imaging*, 38(9):1636–47.

van Velden, F. H. P., Kramer, G. M., Frings, V., Nissen, I. A., Mulder, E. R., de Langen, A. J., Hoekstra, O. S., Smit, E. F., and Boellaard, R. (2016). Repeatability of Radiomic Features in Non-Small-Cell Lung Cancer [(18)F]FDG-PET/CT Studies: Impact of Reconstruction and Delineation. *Molecular imaging and biology*, 18(5):788–95.

Wahl, R. L., Jacene, H., Kasamon, Y., and Lodge, M. A. (2009). From RECIST to PERCIST: Evolving Considerations for PET response criteria in solid tumors. *Journal of nuclear medicine*, 50 Suppl 1(5):122S–50S.

Weisstein, E. W. (2016). Ellipsoid, http://mathworld.wolfram.com/Ellipsoid.html.

Yan, J., Chu-Shern, J. L., Loi, H. Y., Khor, L. K., Sinha, A. K., Quek, S. T., Tham, I. W. K., and Townsend, D. (2015). Impact of Image Reconstruction Settings on Texture Features in 18F-FDG PET. *Journal of nuclear medicine*, 56(11):1667–73.

Yip, S. S. F. and Aerts, H. J. W. L. (2016). Applications and limitations of radiomics. *Physics in medicine and biology*, 61(13):R150–66.

Zhang, C. and Chen, T. (2001). Efficient feature extraction for 2D/3D objects in mesh representation. In *Proceedings 2001 International Conference on Image Processing*, volume 2, pages 935–938. IEEE.