arXiv:1612.07003v8 [cs.CV] 28 Feb 2019

Image biomarker standardisation initiative

Reference manual

The image biomarker standardisation initiative

The image biomarker standardisation initiative (IBSI) is an independent international collaboration which works towards standardising the extraction of image biomarkers from acquired imaging for the purpose of high-throughput quantitative image analysis (radiomics). Lack of reproducibility and validation of high-throughput quantitative image analysis studies is considered to be a major challenge for the field[31,38,84]. Part of this challenge lies in the scantiness of consensus-based guidelines and definitions for the process of translating acquired imaging into high-throughput image biomarkers. The IBSI therefore seeks to provide image biomarker nomenclature and definitions, benchmark data sets, and benchmark values to verify image processing and image biomarker calculations, as well as reporting guidelines, for high-throughput image analysis.

Permanent identifiers

The IBSI uses permanent identifiers for image biomarker definitions and important related concepts such as image processing. These consist of four-character codes and may be used for reference. Please do not use page numbers or section numbers as references, as these are subject to change.

Copyright

This document is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

The digital phantom (see section 5.1) is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/ or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

The radiomics phantom (see section 5.2), which is based on a human lung cancer computed tomography image and published by cancerdata.org (DOI:10.17195/candat.2016.08.1), is licensed under the Creative Commons Attribution-NonCommercial 3.0 Unported Licence. To view a copy of this license, visit https://creativecommons.org/licenses/by-nc/3.0/ or send a letter to Creative Commons, PO Box 1866, Mount View, CA 94042, USA. This license pertains to both the original `DICOM` set, as well as the same data in `NifTI` format released by the IBSI.

Citation information

To cite the document or the digital phantom, please use the following citation:

1. Zwanenburg A, Leger S, Vallières M, Löck S. Image biomarker standardisation initiative. arXiv preprint arXiv:1612.07003.

Additionally, when using the radiomics phantom originally published on cancerdata.org, please include the following citation:

1. Lambin P, Leijenaar RT, Deist TM, Peerlings J, de Jong EE, van Timmeren J, Sanduleanu S, Larue RT, Even AJ, Jochems A, van Wijk Y. Radiomics: the bridge between medical imaging and personalized medicine. Nature Reviews Clinical Oncology. 2017 Dec;14(12):749.
2. Lambin P. Radiomics Digital Phantom, CancerData (2016), DOI:10.17195/candat.2016.08.1

Contact

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

Dr. Martin Vallières
mart.vallieres@gmail.com

IBSI contributors

Mahmoud A. Abdalah	Department of Cancer Imaging and Metabolism, Moffitt Cancer Center, Tampa (FL), USA
Hugo J.W.L. Aerts	Dana-Farber Cancer Institute, Brigham and Women's Hospital, and Harvard Medical School, Harvard University, Boston (MA), USA
Vincent Andrearczyk	Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), Switzerland
Aditya Apte	Department of Medical Physics, Memorial Sloan Kettering Cancer Center, New York (NY), USA
Saeed Ashrafinia	Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore (MD), USA *and* Department of Radiology and Radiological Science, Johns Hopkins University, Baltimore (MD), USA
Spyridon Bakas	Center for Biomedical image Computing and Analytics (CBICA), University of Pennsylvania, Philadelphia (PA), USA *and* Department of Radiology, Hospital of the University of Pennsylvania, Philadelphia (PA), USA
Roelof J. 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 *and* Radiology and Nuclear Medicine, VU University Medical Centre (VUMC), Amsterdam, The Netherlands
Marta Bogowicz	Department of Radiation Oncology, University Hospital Zurich, University of Zurich, Switzerland
Luca Boldrini	Polo Scienze Oncologiche ed Ematologiche, Istituto di Radiologia, Università Cattolica del Sacro Cuore, Fondazione Policlinico Universitario Agostino Gemelli, Roma, Italia
Irène Buvat	Imagerie Moléculaire In Vivo, CEA, Inserm, Univ Paris Sud, CNRS, Universit Paris Saclay, Orsay, France
Gary J.R. Cook	Cancer Imaging Dept, School of Biomedical Engineering and Imaging Sciences, Kings College London, London, United Kingdom

continued on next page

Christos Davatzikos	Center for Biomedical image Computing and Analytics (CBICA), University of Pennsylvania, Philadelphia (PA), USA *and* Department of Radiology, Hospital of the University of Pennsylvania, Philadelphia (PA), USA
Adrien Depeursinge	Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), Switzerland *and* Biomedical Imaging Group, Ecole polytechnique federale de Lausanne (EPFL), Switzerland
Marie-Charlotte Desseroit	Laboratory of Medical Information Processing (LaTIM) - team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, IBSAM, UBO, UBL, Brest, France
Nicola Dinapoli	Polo Scienze Oncologiche ed Ematologiche, Università Cattolica del Sacro Cuore, Fondazione Policlinico Universitario Agostino Gemelli, Roma, Italia
Cuong Viet Dinh	Department of Radiation Oncology, the Netherlands Cancer Institute (NKI), Amsterdam, the Netherlands
Sebastian Echegaray	Department of Radiology, Stanford University School of Medicine, Stanford (CA), USA
Issam El Naqa	Department of Radiation Oncology, Physics Division, University of Michigan *and* Medical Physics Unit, McGill University, Montréal, Qubec, Canada
Hesham Elhalawani	Division of Radiation Oncology, MD Anderson Cancer Center, Houston (TX), USA
Andriy Y. Fedorov	Surgical Planning Laboratory, Brigham and Women's Hospital and Harvard Medical School, Harvard University, Boston (MA), USA
Roberto Gatta	Polo Scienze Oncologiche ed Ematologiche, Università Cattolica del Sacro Cuore, Fondazione Policlinico Universitario Agostino Gemelli, Roma, Italia
Robert J. Gillies	Department of Cancer Imaging and Metabolism, Moffitt Cancer Center, Tampa (FL), USA
Vicky Goh	Cancer Imaging Dept, School of Biomedical Engineering and Imaging Sciences, Kings College London, London, United Kingdom
Matthias Guckenberger	Department of Radiation Oncology, University Hospital Zurich, University of Zurich, Switzerland
Nils Gählert	Department of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany
Michael Götz	Department of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany
Sung Min Ha	Center for Biomedical image Computing and Analytics (CBICA), University of Pennsylvania, Philadelphia (PA), USA *and* Department of Radiology, Hospital of the University of Pennsylvania, Philadelphia (PA), USA
Mathieu Hatt	Laboratory of Medical Information Processing (LaTIM) - team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, IBSAM, UBO, UBL, Brest, France
Fabian Isensee	Department of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany

continued on next page

Jayashree Kalpathy-Cramer	Athinoula A. Martinos Center for Biomedical Imaging, Massachusetts General Hospital (MGH) and Harvard Medical School, Harvard University, Cambridge (MA), USA
Philippe Lambin	Department of Radiation Oncology (The D-Lab), 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, Helmholtz-Zentrum Dresden - Rossendorf, Dresden, Germany *and* National Center for Tumor Diseases (NCT), Partner Site Dresden, Germany: German Cancer Research Center (DKFZ), Heidelberg, Germany; Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universitt Dresden, Dresden, Germany, and; Helmholtz Association / Helmholtz-Zentrum Dresden - Rossendorf (HZDR), Dresden, Germany *and* German Cancer Consortium (DKTK), Partner Site Dresden, and German Cancer Research Center (DKFZ), Heidelberg, Germany
Ralph T.H. Leijenaar	Department of Radiation Oncology (The D-Lab), GROW-School for Oncology and Developmental Biology, Maastricht University Medical Centre+, Maastricht, The Netherlands
Jacopo Lenkowicz	Polo Scienze Oncologiche ed Ematologiche, Istituto di Radiologia, Università Cattolica del Sacro Cuore, Fondazione Policlinico Universitario Agostino Gemelli, Roma, Italia
Fiona Lippert	Section for Biomedical Physics, Department of Radiation Oncology, University of Tübingen, Germany
Are Losnegård	Department of Clinical Medicine, University of Bergen, Bergen, Norway
Steffen Löck	OncoRay – National Center for Radiation Research in Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universität Dresden, Helmholtz-Zentrum Dresden - Rossendorf, Dresden, Germany *and* German Cancer Consortium (DKTK), Partner Site Dresden, and German Cancer Research Center (DKFZ), Heidelberg, Germany *and* Department of Radiotherapy and Radiation Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universitt Dresden, Dresden, Germany
Dennis S. Mackin	Department of Radiation Physics, University of Texas MD Anderson Cancer Center, Houston (TX), USA
Klaus H. Maier-Hein	Department of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany
Todd McNutt	Department of Radiation Oncology, Johns Hopkins University, Baltimore (MD), USA
Olivier Morin	Department of Radiation Oncology, University of California, San Francisco (CA), USA
Henning Müller	Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), Switzerland *and* University of Geneva, Switzerland

continued on next page

Sandy Napel	Department of Radiology, Stanford University School of Medicine, Stanford (CA), USA *and* Department of Electrical Engineering, Stanford University, Stanford (CA), USA *and* Department of Medicine (Biomedical Informatics Research), Stanford University School of Medicine, Stanford (CA), USA
Christophe Nioche	Imagerie Moléculaire In Vivo, CEA, Inserm, Univ Paris Sud, CNRS, Universit Paris Saclay, Orsay, France
Fanny Orlhac	Imagerie Moléculaire In Vivo, CEA, Inserm, Univ Paris Sud, CNRS, Universit Paris Saclay, Orsay, France
Sarthak Pati	Center for Biomedical image Computing and Analytics (CBICA), University of Pennsylvania, Philadelphia (PA), USA *and* Department of Radiology, Hospital of the University of Pennsylvania, Philadelphia (PA), USA
Elisabeth A.G. Pfaehler	Department of Nuclear Medicine and Molecular Imaging, University of Groningen, University Medical Center Groningen (UMCG), Groningen, the Netherlands
Arman Rahmim	Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore (MD), USA *and* Department of Radiology and Radiological Science, Johns Hopkins University, Baltimore (MD), USA
Arvind U.K. Rao	Department of Computational Medicine and Bioinformatics and Department of Radiation Oncology, University of Michigan, Ann Arbor (MI), USA
Christian Richter	OncoRay – National Center for Radiation Research in Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universität Dresden, Helmholtz-Zentrum Dresden - Rossendorf, Dresden, Germany *and* German Cancer Consortium (DKTK), Partner Site Dresden, and German Cancer Research Center (DKFZ), Heidelberg, Germany *and* Helmholtz-Zentrum Dresden - Rossendorf, Institute of Radiooncology OncoRay, Dresden, Germany
Muhammad Musib Siddique	Cancer Imaging Dept, School of Biomedical Engineering and Imaging Sciences, Kings College London, London, United Kingdom
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, University of Tübingen, Germany
Emiliano Spezi	School of Engineering, Cardiff University, Cardiff, United Kingdom *and* Department of Medical Physics, Velindre Cancer Centre, Cardiff, UK
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, University of Tübingen, Germany

continued on next page

Esther G.C. Troost	OncoRay – National Center for Radiation Research in Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universität Dresden, Helmholtz-Zentrum Dresden - Rossendorf, Dresden, Germany *and* National Center for Tumor Diseases (NCT), Partner Site Dresden, Germany: German Cancer Research Center (DKFZ), Heidelberg, Germany; Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universitt Dresden, Dresden, Germany, and; Helmholtz Association / Helmholtz-Zentrum Dresden - Rossendorf (HZDR), Dresden, Germany *and* German Cancer Consortium (DKTK), Partner Site Dresden, and German Cancer Research Center (DKFZ), Heidelberg, Germany *and* Department of Radiotherapy and Radiation Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universitt Dresden, Dresden, Germany *and* Helmholtz-Zentrum Dresden - Rossendorf, Institute of Radiooncology OncoRay, Dresden, Germany
Taman Upadhaya	Department of Nuclear Medicine, CHU Milétrie, Poitiers, France *and* Laboratory of Medical Information Processing (LaTIM) - team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, IBSAM, UBO, UBL, Brest, France
Vincenzo Valentini	Polo Scienze Oncologiche ed Ematologiche, Istituto di Radiologia, Università Cattolica del Sacro Cuore, Fondazione Policlinico Universitario Agostino Gemelli, Roma, Italia
Martin Vallières	Laboratory of Medical Information Processing (LaTIM) - team ACTION (image-guided therapeutic action in oncology), INSERM, UMR 1101, IBSAM, UBO, UBL, Brest, France *and* Medical Physics Unit, McGill University, Montréal, Qubec, Canada
Uulke A. van der Heide	Department of Radiation Oncology, 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
Joost van Griethuysen	Department of Radiology, the Netherlands Cancer Institute (NKI), Amsterdam, the Netherlands *and* GROW-School for Oncology and Developmental Biology, Maastricht University Medical Center, Maastricht, The Netherlands *and* Department of Radiation Oncology, Dana-Farber Cancer Institute, Brigham and Womens Hospital, Harvard Medical School, Boston, MA
Floris H.P. van Velden	Department of Radiology, Leiden University Medical Center (LUMC), Leiden, the Netherlands
Philip Whybra	School of Engineering, Cardiff University, Cardiff, United Kingdom

continued on next page

Alex Zwanenburg	OncoRay – National Center for Radiation Research in Oncology, Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universität Dresden, Helmholtz-Zentrum Dresden - Rossendorf, Dresden, Germany *and* National Center for Tumor Diseases (NCT), Partner Site Dresden, Germany: German Cancer Research Center (DKFZ), Heidelberg, Germany; Faculty of Medicine and University Hospital Carl Gustav Carus, Technische Universitt Dresden, Dresden, Germany, and; Helmholtz Association / Helmholtz-Zentrum Dresden - Rossendorf (HZDR), Dresden, Germany *and* German Cancer Consortium (DKTK), Partner Site Dresden, and German Cancer Research Center (DKFZ), Heidelberg, Germany

Table 1 — Alphabetical list of IBSI contributors.

Contents

Chapter 1

Introduction

A biomarker is "*a characteristic that is objectively measured and evaluated as an indicator of normal biological processes, pathogenic processes, or pharmacologic responses to a therapeutic intervention*" [7]. Biomarkers may be measured from a wide variety of sources, such as tissue samples, cell plating, and imaging. The latter are often referred to as imaging biomarkers. Imaging biomarkers consist of both qualitative biomarkers, which require expert interpretation, and quantitative biomarkers which are based on mathematical definitions. Calculation of quantitative imaging biomarkers can be automated, which enables high-throughput analyses. We refer to such (high-throughput) quantitative biomarkers as image biomarkers to differentiate them from qualitative imaging biomarkers. Image biomarkers characterise the contents of (regions of) an image, such as *volume* or *mean intensity.* Because of the historically close relationship with the computer vision field, image biomarkers are also referred to as image features. The term *features*, instead of biomarkers, will be used throughout the remainder of the reference manual, as the contents are generally applicable and not limited to life sciences and medicine only.

This work focuses specifically on the (high-throughput) extraction of image biomarkers from acquired, reconstructed and stored imaging. High-throughput quantitative image analysis (radiomics) has shown considerable growth in e.g. cancer research [41], but the scarceness of consensus guidelines and definitions has led to it being described as a "wild frontier" [13]. This reference manual therefore presents an effort to chart a course through part of this frontier by presenting consensus-based recommendations, guidelines, benchmarks and definitions for image biomarker extraction, and thus increase the reproducibility of studies involving radiomics.

We opted for a specific focus on image biomarker extraction from acquired imaging. Thus, imaging biomarker validation, viewed in a broader framework such as the one presented by O'Connor et al. [54], or in smaller-scope workflows such as those presented by Caicedo et al. [13] and by Lambin et al. [41], falls beyond the scope of this work. Notably, the question of standardising imaging biomarker acquisition and analysis is being addressed in a more comprehensive manner by groups such as the Quantitative Imaging Biomarker Alliance [52,66], the Quantitative Imaging Network [16,53], and task groups and committees of the American Association of Physicists in Medicine, the European Association for Nuclear Medicine [11], the European Society of Radiology (ESR) [27], and the European Organisation for Research and Treatment of Cancer (EORTC) [54,82], among others. Where overlap exists, the reference manual refers to existing recommendations and guidelines.

This reference manual is divided into several chapters that describe processing of acquired imaging for high-throughput image biomarker extraction (**Chapter 2**); define a diverse set of image biomarkers (**Chapter 3**); describe guidelines for reporting on high-throughput image biomarker extraction and an image biomarker nomenclature (**Chapter 4**); describe the benchmark data sets (**Chapter 5**); and the associated benchmark values for software verification (**Chapter 6**).

Chapter 2

Image processing

Image processing is the sequence of operations required to derive image biomarkers (features) from acquired images. In the context of this work an image is defined as a three-dimensional (3D) stack of two-dimensional (2D) digital image slices. Image slices are stacked along the z-axis. This stack is furthermore assumed to possess the same coordinate system, i.e. image slices are not rotated or translated (in the xy-plane) with regards to each other. Moreover, digital images typically possess a finite resolution. Intensities in an image are thus located at regular intervals, or spacing. In 2D such regular positions are called *pixels*, whereas in 3D the term *voxels* is used. Pixels and voxels are thus represented as the intersections on a regularly spaced grid. Alternatively, pixels and voxels may be represented as rectangles and rectangular cuboids. The centers of the pixels and voxels then coincide with the intersections of the regularly spaced grid. Both representations are used in the document.

Pixels and voxels contain an intensity value for each channel of the image. The number of channels depends on the imaging modality. Most medical imaging generates single-channel images, whereas the number of channels in microscopy may be greater, e.g. due to different stainings. In such multi-channel cases, features may be extracted for each separate channel, a subset of channels, or alternatively, channels may be combined and converted to a single-channel representation. In the remainder of the document we consider an image as if it only possesses a single channel.

The intensity of a pixel or voxel is also called a *grey level* or *grey tone*, particularly in single-channel images. Though practically there is no difference, the terms *grey level* or *grey tone* are more commonly used to refer to discrete intensities, including discretised intensities.

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 schemes used within scientific literature[38]. The image processing scheme is shown in figure 2.1. The processing steps referenced in the figure are described in detail within this chapter.

2.1 Data conversion 23XZ

Some imaging modalities require conversion of raw image data into a more meaningful presentation, e.g. standardised uptake values (SUV)[11]. This is performed during the data conversion step. Assessment of data conversion methods falls outside the scope of the current work.

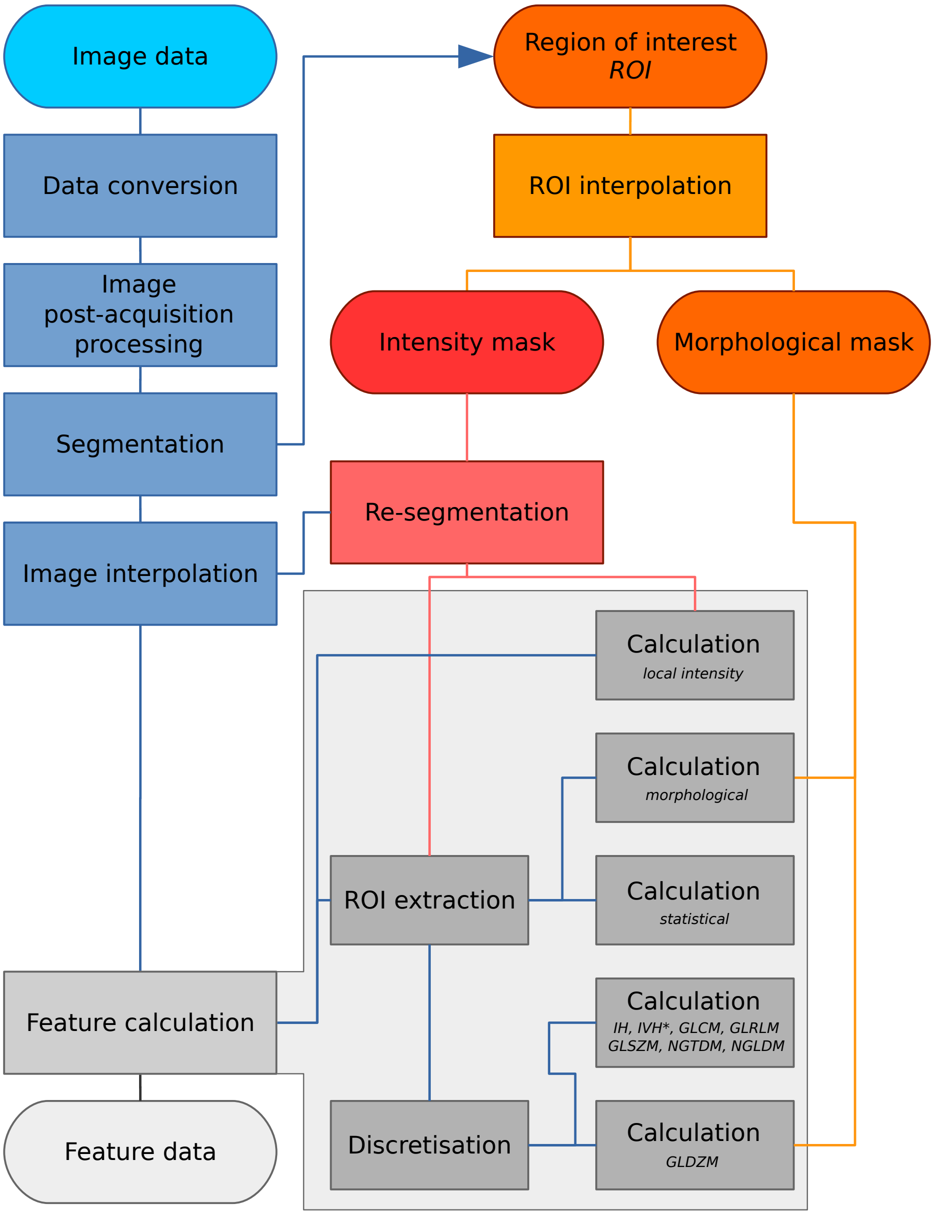


Figure 2.1 — Image processing scheme for image feature calculation. Depending on the specific imaging modality and purpose, some steps may be omitted. The region of interest (ROI) is explicitly split into two masks, namely an intensity and morphological mask, after interpolation to the same grid as the 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 3.5.

2.2 Image post-acquisition processing PCDE

Images are post-processed to enhance image quality. For instance, magnetic resonance imaging (MRI) contains both Gaussian and Rician noise[33] and may benefit from denoising. As another example, intensities measured using MR may be non-uniform across an image and could require correction[9,59,80]. FDG-PET-based may furthermore be corrected for partial volume effects[12,64] and noise[25,43]. In CT imaging, metal objects, e.g. pacemakers and tooth implants, introduce artifacts and may require artifinterpact suppression[32]. Microscopy images generally benefit from field-of-view illumination correction as illumination is usually inhomogeneous due to the light-source or the optical path[13,60].

Evaluation and standardisation of various image post-acquisition processing methods falls outside the scope of the current work. Note that vendors may provide or implement software to perform noise reduction and other post-processing during image reconstruction. In such cases, additional post-acquisition processing may not be required.

2.3 Segmentation OQYT

High-throughput image analysis, within the feature-based paradigm, relies on the definition of regions of interest (ROI). ROIs are used to define the region in which features are calculated. What constitutes an ROI depends on the imaging and the study objective. For example, in 3D microscopy of cell plates, cells are natural ROIs. In medical imaging of cancer patients, the tumour volume is a common ROI. ROIs can be defined manually by experts or (semi-)automatically using algorithms.

From a process point-of-view, segmentation leads to the creation of an ROI mask $\mathbf{R}$, for which every voxel $j \in \mathbf{R}$ (R_j) is defined as:

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

ROIs are typically stored with the accompanying image. Some image formats directly store ROI masks as voxels (e.g. `NIfTI`, `NRRD` and `DICOM Segmentation`), and generating the ROI mask is conducted by loading the corresponding image. In other cases the ROI is saved as a set of (x, y, z) points that define closed loops of (planar) polygons, for example within `DICOM RTSTRUCT` or `DICOM SR` files. In such cases, we should determine which voxel centers lie within the space enclosed by the contour polygon in each slice to generate the ROI mask.

A common method to determine whether a point in an image slice lies inside a 2D polygon is the *crossing number* algorithm, for which several implementations exist[56]. The main concept behind 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. A simple example is shown in figure 2.2. The implementation in the example makes use of the fact that the ROI mask is a regular grid to scan entire rows at a time. The example implementation consists of the following steps:

1. (optional) A ray is cast horizontally from outside the polygon for each of the n image rows. As we iterate over the rows, it is computationally beneficial to exclude polygon edges that will not be crossed by the ray for the current row j. If the current row has y-coordinate y_j, and edge k has two vertices with y-coordinates y_{k1} and y_{k2}, the ray will not cross the edge if both vertices lie either above or below y_j, i.e. $y_j < y_{k1}, y_{k2}$ or $y_j > y_{k1}, y_{k2}$. 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 (remaining) polygon edges with the ray.
3. Iterate over intersection points and add 1 to the count of each pixel center with $x \geq x_i$.
4. Apply the *even-odd* rule. Pixels with an odd count are inside the polygon, whereas pixels 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.

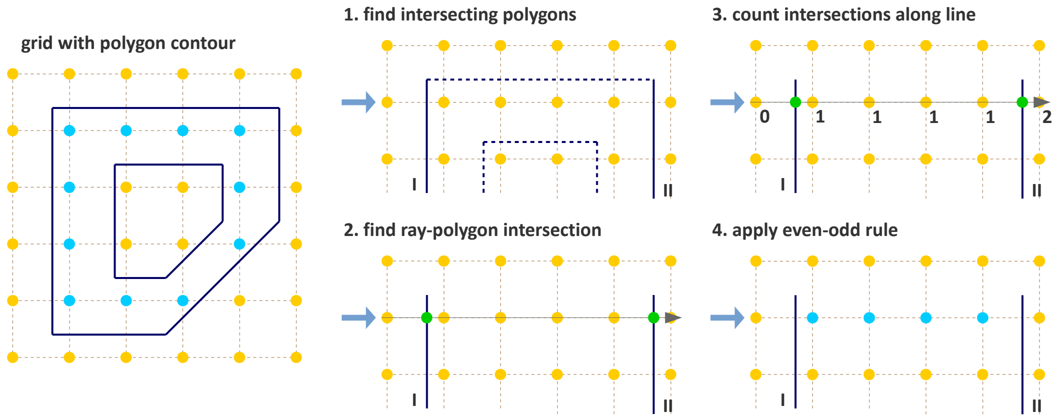

grid with polygon contour
1. find intersecting polygons
I
II
2. find ray-polygon intersection
I
II
3. count intersections along line
0
1
1
1
1
2
I
II
4. apply even-odd rule
I
II

Figure 2.2 — Simple algorithm to determine which pixels are inside a 2D 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.

2.4 Interpolation VTM2

Texture feature sets require interpolation to isotropic voxel spacing to be rotationally invariant, and to allow comparison between image data from different samples, cohorts or batches. Voxel interpolation affects image feature values as many image features are sensitive to changes in voxel size[4,8,57,58,83]. Maintaining consistent isotropic voxel spacing across different measurements and devices is therefore important for reproducibility. At the moment there are no clear indications whether upsampling or downsampling schemes are preferable. Consider, for example, an image stack of slices with $1.0 \times 1.0 \times 3.0$ mm^3 voxel spacing. Down-sampling ($1.0 \times 1.0 \times 1.0$ mm^3) requires inference and introduces artificial information, while conversely upsampling to the largest dimension ($3.0 \times 3.0 \times 3.0$ mm^3) incurs information loss. Multiple-scaling strategies potentially offer a good trade-off[74]. Note that upsampling may introduce image aliasing artifacts that require anti-aliasing filters prior to filtering[48,86].

While in general 3D interpolation algorithms are used to interpolate 3D images, 2D interpolation within the image slice plane may be recommended in some situations. In 2D interpolation voxels are not interpolated between slices. This may be beneficial if, for example, the spacing between slices is large compared to the desired voxel size, and/or compared to the in-plane spacing. Applying 3D interpolation would either require inferencing a large number of voxels between slices (upsampling), or the loss of a large fraction of in-plane information (downsampling). The disadvantage of 2D interpolation is that voxel spacing is no longer isotropic, and as a consequence texture features can only be calculated in-plane.

Interpolation algorithms

Interpolation algorithms translate image intensities from the original image grid to an interpolation grid. In such grids, voxels are spatially represented by their center. Several algorithms are commonly used for interpolation, such as *nearest neighbour*, *trilinear*, *tricubic convolution* and *tricubic spline interpolation*. In short, *nearest neighbour interpolation* assigns the intensity of the most nearby voxel in the original grid to each voxel in the interpolation grid. *Trilinear interpolation* uses the intensities of the eight most nearby voxels in the original grid to calculate a new interpolated intensity using linear interpolation. *Tricubic convolution* and *tricubic spline interpolation* draw upon a larger neighbourhood to evaluate a smooth, continuous third-order polynomial at the voxel centers in the interpolation 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 voxel center, *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 algorithms 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 usually seen as a conservative choice. It does not lead to the blockiness produced by *nearest neighbour interpolation* that introduces bias in local textures[38]. Nor does it lead to out-of-range intensities which may occur due to overshoot with *tricubic* and higher order interpolations. The latter problem can occur in acute intensity transitions, where the local neighbourhood itself is not sufficiently smooth to evaluate the polynomial within the allowed range. *Tricubic* methods, however, may retain tissue contrast differences better. Particularly when upsampling, *trilinear* interpolation may act as a low-pass filter which suppresses higher spatial frequencies and cause artefacts in high-pass spatial filters. Interpolation algorithms and their advantages and disadvantages are treated in more detail elsewhere, e.g. Thévenaz et al.[68].

In a phantom study, Larue et al.[42] compared *nearest neighbour*, *trilinear* and *tricubic* interpolation and indicated that feature reproducibility is dependent on the selected interpolation algorithm, i.e. some features were more reproducible using one particular algorithm.

Rounding image intensities after interpolation 68QD

Image intensities may require rounding after interpolation, or the application of cut-off values. For example, in CT images intensities represent Hounsfield Units, and these do not take non-integer values. Following voxel interpolation, interpolated CT intensities are thus rounded to the nearest integer.

Partial volume effects in the ROI mask E8H9

If the image on which the ROI mask was defined, is interpolated after the ROI was segmented, the ROI mask **R** should likewise be interpolated to the same dimensions. Interpolation of the ROI mask is best conducted using either the *nearest neighbour* or *trilinear interpolation* methods, as these are guaranteed to produce meaningful masks. *Trilinear interpolation* of the ROI mask leads to partial volume effects, with some voxels containing fractions of the original voxels. Since a ROI mask is a binary mask, such fractions need to be binarised by setting a partial volume threshold δ:

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

A common choice for the partial volume threshold is $\delta = 0.5$. For *nearest neighbour interpolation* the ROI mask does not contain partial volume fractions, and may be used directly.

Interpolation results depend on the floating point representation used for the image and ROI masks. Floating point representations should at least be full precision (`32-bit`) to avoid rounding errors.

Interpolation grid UMPJ

Interpolated voxel centers lie on the intersections of a regularly spaced grid. Grid intersections are represented by two coordinate systems. The first coordinate system is the grid coordinate system, with origin at $(0.0, 0.0, 0.0)$ and distance between directly neighbouring voxel centers (spacing) of 1.0. The grid coordinate system is the coordinate system typically used by computers, and consequentially, by interpolation algorithms. The second coordinate system is the world coordinate system, which is typically found in (medical) imaging and provides an image scale. As the desired isotropic spacing is commonly defined in world coordinate dimensions, conversions between world coordinates and grid coordinates are necessary, and are treated in more detail after assessing grid alignment methods.

Grid alignment affects feature values and is non-trivial. Three common grid alignments may be identified, and are shown in figure 2.3:

1. **Fit to original grid** (58MB). In this case the interpolation grid is deformed so that the voxel centers at the grid intersections overlap with the original grid vertices. For an original 4×4 voxel grid with spacing $(3.00, 3.00)$ mm and a desired interpolation spacing of $(2.00, 2.00)$ mm we first calculate the extent of the original voxel grid in world coordinates leading to an extent of $((4 - 1)\,3.00, ((4 - 1)\,3.00) = (9.00, 9.00)$ mm. 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 grid has a grid spacing of $(1.80, 1.80)$ mm in world coordinates, which differs from the desired grid spacing of $(2.00, 2.00)$ mm.
2. **Align grid origins** (SBKJ). A simple approach which conserves the desired grid spacing is the alignment of the origins of the interpolation and original grids. Keeping with the same example, the interpolation grid is (6×6). The resulting voxel grid has a grid spacing of $(2.00, 2.00)$ mm in world coordinates. By definition both grids are aligned at the origin, $(0.00, 0.00)$.
3. **Align grid centers** (3WE3). The position of the origin may depend on image meta-data defining image orientation. Not all software implementations may process this meta-data the same way. An implementation-independent solution is to align both grids on the grid center. Again, keeping with the same example, the interpolation grid is (6×6). Thus, the resulting voxel grid has a grid spacing of $(2.00, 2.00)$ mm in world coordinates.

Align grid centers is recommended as it is implementation-independent and achieves the desired voxel spacing. Technical details of implementing *align grid centers* are described below.

Interpolation grid dimensions 026Q

The dimensions of the interpolation grid are determined as follows. Let n_a be the number of points along one axis of the original grid and $s_{a,w}$ their spacing in world coordinates. Then, let

$s_{b,w}$ be the desired spacing after interpolation. The axial dimension of the interpolated mesh grid is then:

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

Rounding towards infinity guarantees that the interpolation grid exists even when the original grid contains few voxels. However, it also means that the interpolation mesh grid is partially located outside of the original grid. Extrapolation is thus required. Padding the original grid with the intensities at the boundary is recommended. Some implementations of interpolation algorithms may perform this padding internally.

Interpolation grid position QCY4

For the *align grid centers* method, the positions of the interpolation grid points are determined as follows. As before, let n_a and n_b be the dimensions of one axis in the original and interpolation grid, respectively. Moreover, let $s_{a,w}$ be the original spacing and $s_{b,w}$ the desired spacing for the same axis in world coordinates. Then, with $x_{a,w}$ the origin of the original grid in world coordinates, the origin of the interpolation grid is located at:

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

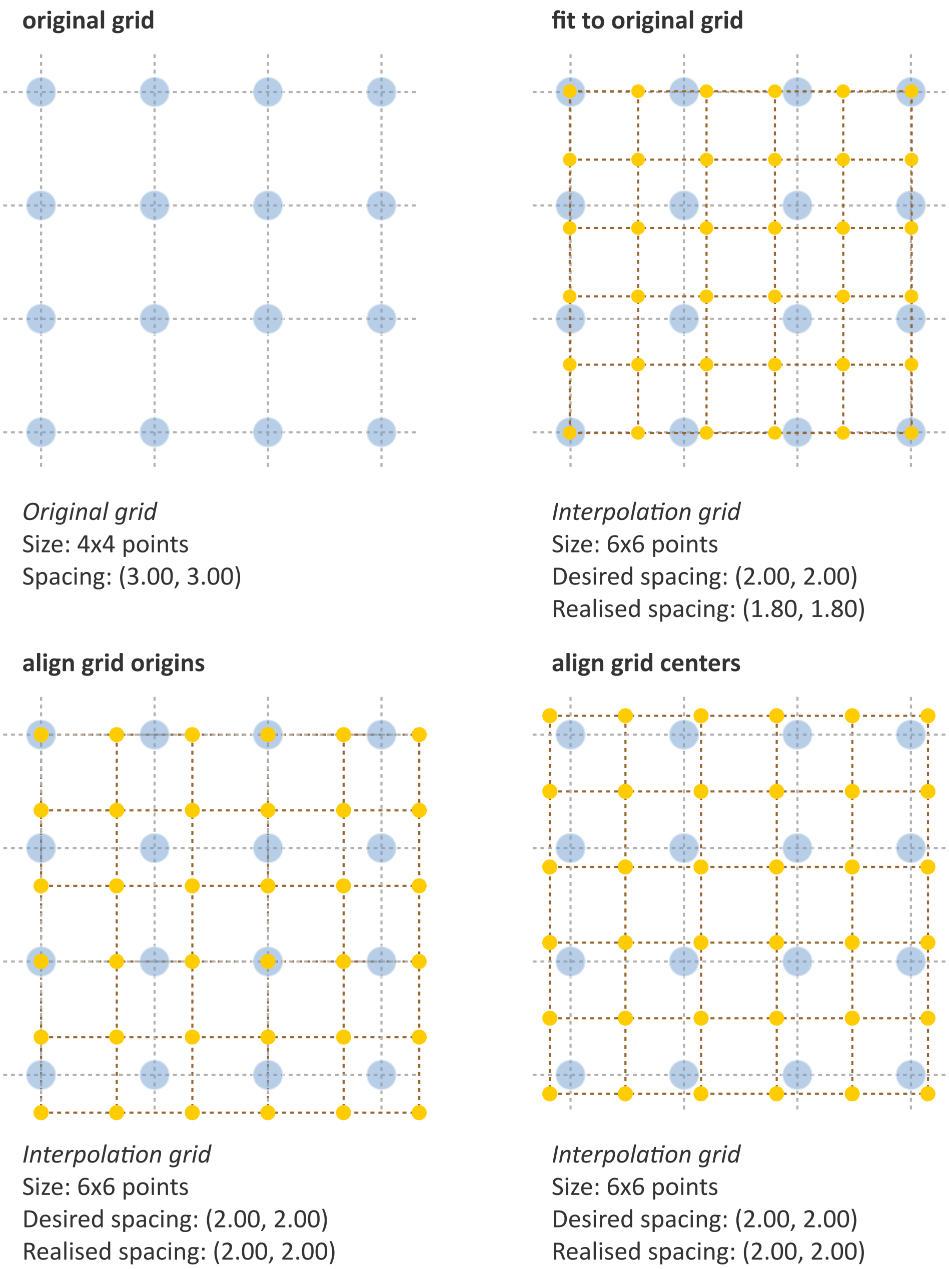
In the grid coordinate system, the original grid origin is located at $x_{a,g} = 0$. The origin of the interpolation grid is then located at:

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

Here the fraction $s_{b,w}/s_{a,w} = s_{b,g}$ is the desired spacing in grid coordinates. Thus, the interpolation grid points along the considered axis are located at grid coordinates:

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

Naturally, the above description applies to each grid axis.



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 2.3 — Different interpolation mesh grids based on an original 4×4 grid with $(3.00, 3.00)$ mm spacing. The desired interpolation spacing is $(2.00, 2.00)$ mm. *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 centered on the center of original and interpolation grids.

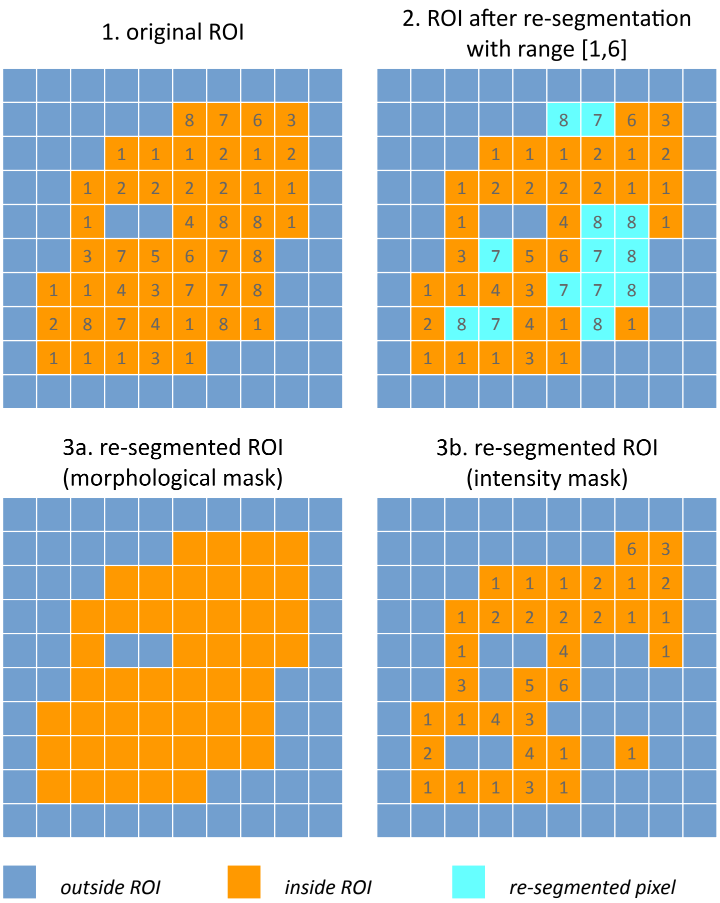

1. original ROI
2. ROI after re-segmentation with range [1,6]
3a. re-segmented ROI (morphological mask)
3b. re-segmented ROI (intensity mask)
outside ROI
inside ROI
re-segmented pixel

Figure 2.4 — 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.

2.5 Re-segmentation IF9H

Re-segmentation entails updating the ROI mask $\mathbf{R}$ based on corresponding voxel intensities $\mathbf{X}_{gl}$. Re-segmentation may be performed to exclude voxels from a previously segmented ROI, and is performed after interpolation. An example use would be the exclusion of air or bone voxels from an ROI defined on CT imaging. Two common re-segmentation methods are described in this section. Combining multiple re-segmentation methods is possible. In this case, the intersection of the intensity ranges defined by the re-segmentation methods is used.

Intensity and morphological masks of an ROI ECJF

Conventionally, an ROI consists of a single mask. However, re-segmentation may lead to exclusion of internal voxels, or divide the ROI into sub-volumes. To avoid undue complexity by again updating the re-segmented ROI for a more plausible morphology, we define two separate ROI masks.

The morphological mask (G5KJ) is not re-segmented and maintains the original morphology as defined by an expert and/or (semi-)automatic segmentation algorithms.

The intensity mask (`SEFI`) can be re-segmented and will contain only the selected voxels. For many feature families, only this is important. However, for morphological and grey level distance zone matrix (GLDZM) feature families, both intensity and morphological masks are used. A two-dimensional example is shown in figure 2.4.

Range re-segmentation USB3

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 with Hounsfield Units indicating air and bone tissue in the tumour ROI within CT images, or low activity areas in PET images. Such ranges of intensities of included voxels are usually presented as a closed interval $[a, b]$ or half-open interval $[a, \infty)$, respectively. For arbitrary intensity units (found in e.g. raw MRI data, uncalibrated microscopy images, and many spatial filters), no re-segmentation range can be provided.

When a re-segmentation range is defined by the user, it needs to be propagated and used for the calculation of features that require a specified intensity range (e.g. intensity-volume histogram features) and/or that employs *fixed bin size* discretisation. Recommendations for the possible combinations of different imaging intensity definitions, re-segmentation ranges and discretisation algorithms are provided in Table 2.1.

Intensity outlier filtering 7ACA

ROI voxels with outlier intensities may be removed from the intensity mask. One method for defining outliers was suggested by Vallières et al.[73] after Collewet et al.[18]. The mean μ and standard deviation σ of grey levels of voxels assigned to the ROI are calculated. Voxels outside the range $[\mu - 3\sigma, \mu + 3\sigma]$ are subsequently excluded from the intensity mask.

2.6 ROI extraction 1OBP

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 may then used to exclude these voxels from calculations. Voxels included in the ROI mask retain their original intensity.

2.7 Intensity discretisation 4R0B

Discretisation or quantisation of image intensities inside the ROI is often required to make calculation of texture features tractable[84], and possesses noise-suppressing properties as well. An example of discretisation is shown in figure 2.5.

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 (as described below) that may make them better suited for specific purposes. Note that the lowest bin always has value 1, and not 0. This ensures consistency for calculations of texture features, where for some features grey level 0 is not allowed .

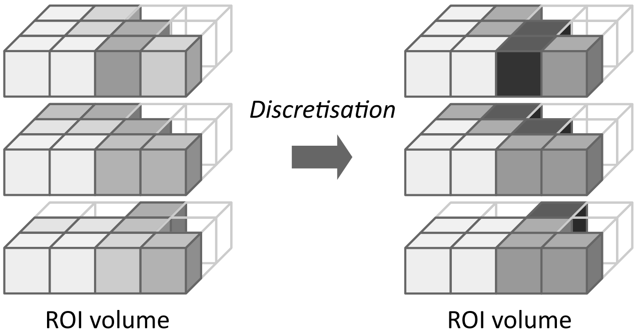


Figure 2.5 — The image volume contained in the region of interest (ROI) is discretised. Here, intensities from the original ROI volume were assigned to 3 intensity bins to create a discretised volume.

Fixed bin number K15C

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

$$X_{d,k} = \begin{cases} \left\lfloor N_g \frac{X_{gl,k}-X_{gl,min}}{X_{gl,max}-X_{gl,min}} \right\rfloor + 1 & X_{gl,k} < X_{gl,max} \\ N_g & X_{gl,k} = X_{gl,max} \end{cases}$$

In short, the intensity $X_{gl,k}$ of voxel k is corrected by the lowest occurring intensity $X_{gl,min}$ in the ROI, divided by the bin width $\left(X_{gl,max} - X_{gl,min}\right)/N_g$, and subsequently rounded down to the nearest integer (floor function).

The *fixed bin number* method breaks the relationship between image intensity and physiological meaning (if any). However, it introduces a normalising effect which may be beneficial when intensity units are arbitrary (e.g. raw MRI data and many spatial filters), and where contrast is considered important. Furthermore, as values of many features depend on the number of grey levels found within a given ROI, the use of a *fixed bin number* discretisation algorithm allows for a direct comparison of feature values across multiple analysed ROIs (e.g. across different samples).

Fixed bin size Q3RU

Fixed bin size discretisation is conceptually simple. A new bin is assigned for every intensity interval with width w_b; i.e. w_b is the bin width, starting at a minimum $X_{gl,min}$. The minimum intensity may be a user-set value as defined by the lower bound of the re-segmentation range, or data-driven as defined by the minimum intensity in the ROI $X_{gl,min} = \min\left(X_{gl}\right)$. In all cases, the method used and/or set minimum value must be clearly reported. However, to maintain consistency between samples, we strongly recommend to always set the same minimum value for all samples as defined by the lower bound of the re-segmentation range (e.g. HU of -500 for CT, SUV of 0 for PET, etc.). In the case that no re-segmentation range may be defined due to arbitrary intensity units (e.g. raw MRI data and many spatial filters), the use of the *fixed bin size* discretisation algorithm is not recommended.

The *fixed bin size* method has the advantage of maintaining a direct relationship with the original intensity scale, which could be useful for functional imaging modalities such as PET.

Discretised intensities are computed as follows:

$$X_{d,k} = \left\lfloor \frac{X_{gl,k} - X_{gl,min}}{w_b} \right\rfloor + 1$$

In short, the minimum intensity $X_{gl,min}$ is subtracted from intensity $X_{gl,k}$ in voxel k, and then divided by the bin width w_b. The resulting value is subsequently rounded down to the nearest

Imaging intensity units[1]	**Re-segmentation range**	**FBN**[2]	**FBS**[3]
definite	$[a, b]$	✔	✔
	$[a, \infty)$	✔	✔
	none	✔	✕
arbitrary	none	✔	✕

Table 2.1 — Recommendations for the possible combinations of different imaging intensity definitions, re-segmentation ranges and discretisation algorithms. Checkmarks (✔) represent recommended combinations of re-segmentation range and discretisation algorithm, whereas crossmarks (✕) represent non-recommended combinations.

[1] PET and CT are examples of imaging modalities with *definite* intensity units (e.g. SUV and HU, respectively), and raw MRI data of arbitrary intensity units.

[2] *Fixed bin number* (FBN) discretisation uses the actual range of intensities in the analysed ROI (re-segmented or not), and not the re-segmentation range itself (when defined).

[3] *Fixed bin size* (FBS) discretisation uses the lower bound of the re-segmentation range as the minimum set value. When the re-segmentation range is not or cannot be defined (e.g. arbitrary intensity units), the use of the FBS algorithm is not recommended.

integer (floor function), and 1 is added to arrive at the discretised intensity.

Other methods

Many other methods and variations for discretisation exist, but are not described in detail here. Vallières et al.[73] described the use of *intensity histogram equalisation* and *Lloyd-Max* algorithms for discretisation. *Intensity histogram equalisation* involves redistributing intensities so that the resulting bins contain a similar number of voxels, i.e. contrast is increased by flattening the histogram as much as possible[34]. Histogram equalisation of the ROI imaging intensities can be performed before any other discretisation algorithm (e.g. FBN, FSB, etc.), and it also requires the definition of a given number of bins in the histogram to be equalised. The *Lloyd-Max* algorithm is an iterative clustering method that seeks to minimise mean squared discretisation errors[46,49].

Recommendations

The discretisation method that leads to optimal feature inter- and intra-sample reproducibility is modality-dependent. Usage recommendations for the possible combinations of different imaging intensity definitions, re-segmentation ranges and discretisation algorithms are provided in Table 2.1. Overall, the discretisation choice has a substantial impact on intensity distributions, feature values and reproducibility[4,24,37,38,44,57,79].

2.8 Feature calculation

Feature calculation is the final processing step where feature descriptors are used to quantify characteristics of the ROI. After calculation such features may be used as image biomarkers by relating them to physiological and medical outcomes of interest. Feature calculation is handled in full details in the next chapter.

Let us recall that the image processing steps leading to image biomarker calculations can be performed in many different ways, notably in terms of spatial filtering, segmentation, interpolation and discretisation parameters. Furthermore, it is plausible that different texture features will better quantify the characteristics of the ROI when computed using different image processing parameters. For example, a lower number of grey levels in the discretisation process (e.g. 8 or

16) may allow to better characterize the sub-regions of the ROI using *grey level size zone matrix* (GLSZM) features, whereas *grey level co-occurence matrix* (GLCM) features may be better modeled with a higher number of grey levels (e.g. 32 or 64). Overall, these possible differences opens the door to the optimization of image processing parameters for each different feature in terms of a specific objective. For the specific case of the optimization of image interpolation and discretisation prior to texture analysis, Vallières *et al.*[73] have named this process *texture optimization*. The authors notably suggested that the *texture optimization* process could have significant influence of the prognostic capability of subsequent features. In another study[74], the authors constructed predictive models using textures calculated from all possible combinations of PET and CT images interpolated at four isotropic resolutions and discretised with two different algorithms and four numbers of grey levels.

Chapter 3

Image features

In this chapter we will describe a set of quantitative image features. The feature set presented here largely builds upon the feature sets proposed by Aerts et al.[1] and Hatt et al.[38], which are themselves largely derived from earlier works. References to earlier work are provided whenever they could be identified.

The set of features can be divided into a number of families, of which intensity-based statistical, intensity histogram-based, intensity-volume histogram-based, morphological features, local intensity, and texture matrix-based features are treated here. All texture matrices are rotationally and translationally invariant. Illumination invariance of texture matrices may be achieved by particular discretisation schemes, e.g. *histogram matching.* None of the texture matrices are scale invariant, a property which can be useful in many (biomedical) applications for scale optimization. What the presented texture matrices lack, however, is directionality in combination with rotation invariance. These may be achieved by local binary patterns and steerable filters, which however fall beyond the scope of the current work. For these and other texture features, see Depeursinge et al.[23].

Features are calculated on the base image, as well as images transformed using wavelet or Gabor filters). To calculate features, it is assumed that an image segmentation mask exists, which identifies the voxels located within a region of interest (ROI). The ROI itself consists of two masks, an intensity mask and a morphological mask. These masks may be identical, but not necessarily so, as described in Section 2.5.

Several feature families require additional image processing steps before feature calculation. Notably intensity histogram and texture feature families require prior discretisation of intensities into grey level bins. Other feature families do not require discretisation before calculations. For more details on image processing, see figure 2.1 in the previous chapter.

Below is an overview table that summarises image processing requirements for the different feature families.

		ROI mask		
Feature family	**count**	**morph.**	**int.**	**discr.**
morphology	29	✔	✔	×
local intensity	2	×	✔[a]	×
intensity-based statistics	18	×	✔	×
intensity histogram	23	×	✔	✔
intensity-volume histogram	5	×	✔	✔[b]
grey level co-occurrence matrix	25	×	✔	✔
grey level run length matrix	16	×	✔	✔
grey level size zone matrix	16	×	✔	✔
grey level distance zone matrix	16	✔	✔	✔
neighbourhood grey tone difference matrix	5	×	✔	✔
neighbouring grey level dependence matrix	17	×	✔	✔

Table 3.1 — Feature families and required image processing. For each feature family, the number of features in the document, the required input of a morphological (morph.) and/or intensity (int.) ROI mask, as well as the requirement of image discretisation (discr.) is provided.
[a] The entire image volume should be available when computing local intensity features.
[b] Image discretisation for the intensity-volume histogram is performed with finer discretisation than required for e.g. textural features.

Aside from image processing requirements there are two other concepts which were not explicitly introduced, but which play an important role for many features: distance and feature aggregation. In addition, distance weighting for texture features is described. All three are defined below.

Grid distances MPUJ

Grid distance is an important concept that is used by several feature families, particularly texture features. Grid distances can be measured in several ways. Let $\mathbf{m} = (m_x, m_y, m_z)$ be the vector from a center voxel at $\mathbf{k} = (k_x, k_y, k_z)$ to a neighbour voxel at $\mathbf{k} + \mathbf{m}$. The following norms (distances) are used:

- ℓ_1 norm or *Manhattan* norm (LIFZ):

$$\|\mathbf{m}\|_1 = |m_x| + |m_y| + |m_z|$$

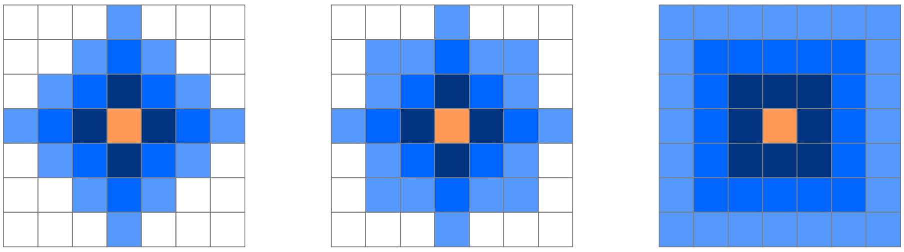
- ℓ_2 norm or *Euclidean* norm (G9EV):

$$\|\mathbf{m}\|_2 = \sqrt{m_x^2 + m_y^2 + m_z^2}$$

- ℓ_∞ norm or *Chebyshev* norm (PVMT):

$$\|\mathbf{m}\|_\infty = \max(|m_x|, |m_y|, |m_z|)$$

An example of how the above norms differ in practice is shown in figure 3.1.



(a) Manhattan norm **(b)** Euclidean norm **(c)** Chebyshev norm

Figure 3.1 — Grid neighbourhoods for distances up to 3 according to Manhattan, Euclidean and Chebyshev norms. The orange pixel is considered the center pixel. Dark blue pixels have distance $\delta = 1$, blue pixels $\delta \leq 2$ and light blue pixels $\delta \leq 3$ for the corresponding norm.

Feature aggregation 5QB6

Features from some families may be calculated from, e.g. slices. As a consequence, multip le values for the same feature may be computed. These different values should be combined into a single value for many common purposes. This process is referred to as feature aggregation. Feature aggregation methods depend on the family, and are detailed in the family description.

Distance weighting 6CK8

Distance weighting is not a default operation for any of the texture families, but is implemented in software such as PyRadiomics[77]. It may for example be used to put more emphasis on local intensities.

3.1 Morphological features HCUG

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:

1. The volume is represented by a collection of voxels with each voxel taking up a certain volume (LQD8).
2. The volume is represented by a voxel point set $\mathbf{X}_c$ that consists of coordinates of the voxel centers (4KW8).
3. The volume is represented by a surface mesh (WRJH).

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 outside volume approximations because it does not 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 WRJH

A mesh-based representation of the outer surface allows consistent evaluation of the surface volume and area independent of size. Voxel-based representations lead to partial volume effects and over-estimation of surface area. The surface of the ROI volume is translated into a triangle mesh using a meshing algorithm. While multiple meshing algorithms exist, we suggest the use of the *Marching Cubes* algorithm[45,47] because of its widespread availability in different programming languages and reasonable approximation of the surface area and volume[65]. In practice, mesh-derived feature values depend upon the meshing algorithm and small differences may occur between meshing implementations.

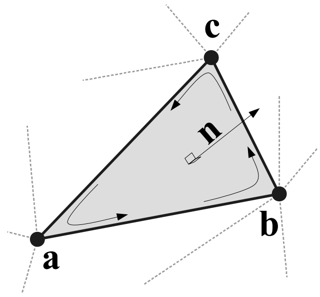


Figure 3.2 — Meshing algorithms draw faces and vertices to cover the ROI. One face, spanned by vertices $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$, is highlighted. Moreover, the vertices define the three edges $\mathbf{ab} = \mathbf{b} - \mathbf{a}$, $\mathbf{bc} = \mathbf{c} - \mathbf{b}$ and $\mathbf{ca} = \mathbf{a} - \mathbf{c}$. 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. Other algorithms require a so-called *isovalue*, for which a value of 0.5 can be used since the ROI mask consists of 0 and 1 values, and we want to roughly draw the mesh half-way between voxel centers. Depending on implementation, algorithms may also require padding of the ROI mask with non-ROI (0) voxels to correctly estimate the mesh 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. An example triangle face is drawn in Figure 3.2. 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 morphological mask is used to calculate many of the morphological features and to generate the voxel point set $\mathbf{X}_c$. Any holes within the morphological mask 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 $\mathbf{X}_{c,gl}$. In the benchmark data sets (Chapter 5), the masks are identical for the digital phantom, but differ due to re-segmentation of the intensity mask.

Aggregating features

By definition, morphological features are calculated in 3D (DHQ4), and not per slice.

Units of measurement

By definition, morphological features are computed using the unit of length as defined in the DICOM standard, i.e. millimeter for most medical imaging modalities[1].

If the unit of length is not defined by a standard, but is explicitly defined as meta data, this definition should be used. In this case, care should be taken that this definition is consistent across all data in the cohort.

If a feature value should be expressed as a different unit of length, e.g. cm instead of mm, such conversions should take place after computing the value using the standard units.

3.1.1 Volume (mesh) RNU0

The mesh-based *volume* V is calculated from the ROI mesh as follows[85]. A tetrahedron is formed by each face k and the origin. By placing the origin vertex of each tetrahedron at $(0,0,0)$, the signed volume of the tetrahedron 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:

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

[1] DICOM PS3.3 2019a - Information Object Definitions, Section 10.7.1.3

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

3.1.2 Volume (voxel counting) YEKZ

In clinical practice, volumes are commonly determined by counting voxels. For volumes consisting of a large number of voxels (1000s), the differences between *voxel counting* and *mesh-based* approaches are usually negligible. However for volumes with a low number of voxels (10s to 100s), *voxel counting* will overestimate volume compared to the *mesh-based* approach. It is therefore only used as a reference feature, and not in the calculation of other morphological features.

Voxel counting volume is defined as:

$$F_{morph.approx.vol} = \sum_{k=1}^{N_v} V_k$$

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

3.1.3 Surface area (mesh) COJK

The *surface area* A is also calculated from the ROI mesh by summing over the triangular face surface areas[1]. By definition, the area of face k is:

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

As in Figure 3.2, edge $\mathbf{ab} = \mathbf{b} - \mathbf{a}$ is the vector from vertex $\mathbf{a}$ to vertex $\mathbf{b}$, and edge $\mathbf{ac} = \mathbf{c} - \mathbf{a}$ the vector from vertex $\mathbf{a}$ to vertex $\mathbf{c}$. The total *surface area* A is then:

$$F_{morph.area} = A = \sum_{k=1}^{N_{fc}} A_k$$

3.1.4 Surface to volume ratio 2PR5

The *surface to volume ratio* is given as[1]:

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

3.1.5 Compactness 1 SKGS

Several features (*compactness 1* and *2*, *spherical disproportion*, *sphericity* and *asphericity*) quantify the deviation of the ROI volume from a representative spheroid. All these definitions can be derived from one another. As a results these features are are highly correlated and may thus be redundant. *Compactness 1* [1] is a measure for how compact, or sphere-like the volume is. It is defined as:

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

Some definitions use $A^{2/3}$ instead of $A^{3/2}$ [1], but this does not lead to dimensionless quantity.

3.1.6 Compactness 2 BQWJ

Compactness 2[1] also quantifies how sphere-like the volume is:

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

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

3.1.7 Spherical disproportion KRCK

Spherical disproportion[1] likewise describes how sphere-like the volume is:

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

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

3.1.8 Sphericity QCFX

Sphericity[1] is a further measure to describe how sphere-like the volume is:

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

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

3.1.9 Asphericity 25C7

Asphericity[6] also describes how much the ROI deviates from a perfect sphere, with perfectly spherical volumes having an asphericity of 0. Asphericity is defined as:

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

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

3.1.10 Centre of mass shift KLMA

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

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

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

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

The distance between the two centres of mass is then:

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

3.1.11 Maximum 3D diameter `L0JK`

The *maximum 3D diameter*[1] is the distance between the two most distant vertices in the ROI mesh vertex set $\mathbf{X}_{vx}$:

$$F_{morph.diam} = \max\left(||\vec{X}_{vx,k_1} - \vec{X}_{vx,k_2}||_2\right), \qquad k_1 = 1,\ldots,N \qquad k_2 = 1,\ldots,N$$

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 distances between all vertices. Despite the remaining $O(n^2)$ cost of calculating distances between different vertices, $\mathbf{X}_{vx,convex}$ is usually considerably smaller in size than $\mathbf{X}_{vx}$. Moreover, the convex hull is later used for the calculation of other morphological features (3.1.25-3.1.26).

3.1.12 Major axis length `TDIC`

Principal component analysis (PCA) can be used to determine the main orientation of the ROI[63]. 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. Several features make use of principal component analysis, namely *major*, *minor* and *least axis length*, *elongation*, *flatness*, and *approximate enclosing ellipsoid volume* and area density.

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 $\mathbf{X}_c$[39]:

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

3.1.13 Minor axis length `P9VJ`

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, as described in Section 3.1.12:

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

3.1.14 Least axis length 7J51

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, as described in Section 3.1.12:

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

3.1.15 Elongation Q3CK

The ratio of the major and minor principal 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.

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

3.1.16 Flatness N17B

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.

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

3.1.17 Volume density - axis-aligned bounding box PBX1

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 (AABB) 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.

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

This feature is also called *extent*[26,63].

3.1.18 Area density - axis-aligned bounding box R59B

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 $\mathbf{X}_{vx}$[76]. The bounding box is identical to the one used in the *volume density - axis-aligned bounding box* feature. Thus:

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

3.1.19 Volume density - oriented minimum bounding box ZH1A

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 3D rotating callipers technique was devised by O'Rourke[55] 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[10] and Chan and Tan[14]. Thus:

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

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

3.1.20 Area density - oriented minimum bounding box IQYR

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.

3.1.21 Volume density - approximate enclosing ellipsoid 6BDE

The eigenvectors and eigenvalues from PCA of the ROI voxel center point set $\mathbf{X}_c$ can be used to describe an ellipsoid approximating the point cloud[50]. The volume of an ellipsoid is $V_{aee} = 4\pi\, a\, b\, c/3$, with a, b, and c being the lengths of the ellipsoid's semi-principal axes, see Section 3.1.12. The *volume density* is then:

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

3.1.22 Area density - approximate enclosing ellipsoid RDD2

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

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

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 limited, and as a default we stop calculations at this polynomial degree.

The *area density* is then approximated as:

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

3.1.23 Volume density - minimum volume enclosing ellipsoid SWZ1

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.[2,70], which are usually elaborations on Khachiyan's barycentric coordinate descent method[40].

The minimum volume enclosing ellipsoid (MVEE) encloses the ROI mesh vertex set $\mathbf{X}_{vx}$, and by definition $\mathbf{X}_{vx,convex}$ as well. Use of the convex mesh set $\mathbf{X}_{vx,convex}$ is recommended due to its sparsity compared to the full vertex set. The volume of the MVEE 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.

3.1.24 Area density - minimum volume enclosing ellipsoid BRI8

The surface area of an ellipsoid does not have a general elementary form, but should be approximated as noted in Section 3.1.22. Let the approximated surface area of the MVEE be A_{mvee}. Then:

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

3.1.25 Volume density - convex hull R3ER

The convex hull encloses ROI mesh vertex set $\mathbf{X}_{vx}$ and consists of the vertex set $\mathbf{X}_{vx,convex}$ and corresponding faces, see section 3.1.11. The volume of the ROI mesh convex hull set V_{convex} is calculated as for the *volume* feature (3.1.1). 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*[26,63].

3.1.26 Area density - convex hull 7T7F

The area of the convex hull A_{convex} is the sum of the areas of the faces of the convex hull, as in the calculation of the *area* feature (section 3.1.3). The convex hull is identical to the one used in the *volume density - convex hull* feature. Then:

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

3.1.27 Integrated intensity 99N0

Integrated intensity is the average grey level multiplied by the volume. In the context of ^{18}F-FDG-PET, this feature is called *total lesion glycolysis*[72]. Thus:

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

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

3.1.28 Moran's I index N365

Moran's I index is an indicator of spatial autocorrelation[20,51]. It is defined as:

$$F_{morph.moran.i} = \frac{N_{v,gl}}{\sum_{k_1=1}^{N_{v,gl}} \sum_{k_2=1}^{N_{v,gl}} w_{k_1 k_2}} \frac{\sum_{k_1=1}^{N_{v,gl}} \sum_{k_2=1}^{N_{v,gl}} w_{k_1 k_2} \left(X_{gl,k_1} - \mu\right) \left(X_{gl,k_2} - \mu\right)}{\sum_{k=1}^{N_{v,gl}} \left(X_{gl,k} - \mu\right)^2}, \qquad k_1 \neq k_2$$

As before $N_{v,gl}$ is the number of voxels in the ROI intensity mask, μ is the mean of $\mathbf{X}_{gl}$ and $w_{k_1 k_2}$ is a weight factor, equal to the inverse Euclidean distance between voxels k_1 and k_2 of the point set $\mathbf{X}_{c,gl}$ of the ROI intensity mask[19]. Values of Moran's I close to 1.0, 0.0 and -1.0 indicate high spatial autocorrelation, no spatial autocorrelation and high spatial anti-autocorrelation, respectively.

Note that for an ROI containing many voxels, calculating Moran's I index may be computationally expensive due to $O(n^2)$ behaviour. Approximation by repeated subsampling of the ROI may be required to make the calculation tractable, at the cost of accuracy.

3.1.29 Geary's C measure NPT7

Geary's C measure assesses spatial autocorrelation, similar to Moran's I index[20,30]. In contrast to Moran's I index, Geary's C measure directly assesses grey level differences between voxels and is more sensitive to local spatial autocorrelation. This measure is defined as:

$$F_{morph.geary.c} = \frac{N_{v,gl} - 1}{2 \sum_{k_1=1}^{N_{v,gl}} \sum_{k_2=1}^{N_{v,gl}} w_{k_1 k_2}} \frac{\sum_{k_1=1}^{N_{v,gl}} \sum_{k_2=1}^{N_{v,gl}} w_{k_1 k_2} \left(X_{gl,k_1} - X_{gl,k_2}\right)^2}{\sum_{k=1}^{N_{v,gl}} \left(X_{gl,k} - \mu\right)^2}, \qquad k_1 \neq k_2$$

As with Moran's I, $N_{v,gl}$ is the number of voxels in the ROI intensity mask, μ is the mean of $\mathbf{X}_{gl}$ and $w_{k_1 k_2}$ is a weight factor, equal to the inverse Euclidean distance between voxels k_1 and k_2 of the ROI voxel point set $\mathbf{X}_{c,gl}$[19].

Just as Moran's I, Geary's C measure exhibits $O(n^2)$ behaviour and an approximation scheme may be required to make calculation feasible for large ROIs.

3.2 Local intensity features 9ST6

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

Aggregating features

By definition, local intensity features are calculated in 3D (DHQ4), and not per slice.

3.2.1 Local intensity peak VJGA

The *local intensity peak* was originally devised for reducing variance in determining standardised uptake values[81]. It is defined as the mean intensity in a 1 cm^3 spherical volume (in world coordinates), which is centered on the voxel with the maximum intensity level in the ROI intensity mask[28].

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

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

3.2.2 Global intensity peak 0F91

The *global intensity peak* feature $F_{loc.peak.global}$ is similar to the *local intensity peak*[28]. However, instead of calculating the mean intensity for the voxel(s) with the maximum intensity, the mean intensity is calculated within a 1 cm^3 neighbourhood for every voxel in the ROI intensity mask. The highest intensity peak value is then selected.

Calculation of the *global intensity peak* feature may be accelerated by construction and application of an appropriate spatial spherical mean convolution filter, due to the convolution theorem. In this case one would first construct an empty 3D filter that will fit a 1 cm^3 sphere. Within this context, the filter voxels may be represented by a point set, akin to $\mathbf{X}_c$ in section 3.1. Euclidean distances in world spacing between the central voxel of the filter and every remaining voxel are computed. If this distance lies within radius $r = \left(\frac{3}{4\pi}\right)^{1/3} \approx 0.62$ the corresponding voxel receives a label 1, and 0 otherwise. Subsequent summation of the voxel labels yields N_s, the number of voxels within the 1 cm^3 sphere. The filter then becomes a spherical mean filter by dividing the labels by N_s.

3.3 Intensity-based statistical features UHIW

The intensity-based 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 intensity distribution. Intensity-based statistical features are not meaningful if the intensity scale is arbitrary.

The set of intensities of the N_v voxels included in the ROI intensity mask is denoted as $\mathbf{X}_{gl} = \{X_{gl,1}, X_{gl,2}, \ldots, X_{gl,N_v}\}$.

Aggregating features

We recommend calculating intensity-based statistical features using the 3D volume (DHQ4). Computing features per slice and subsequently averaging (3IDG) is not recommended.

3.3.1 Mean Q4LE

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

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

3.3.2 Variance ECT3

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

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

3.3.3 Skewness KE2A

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

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

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

3.3.4 (Excess) kurtosis IPH6

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

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

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

3.3.5 Median Y12H

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

3.3.6 Minimum grey level 1GSF

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

3.3.7 10th percentile QG58

P_{10} is the 10th percentile of $\mathbf{X}_{gl}$. P_{10} is more robust to grey level outliers than the *minimum grey level* and is defined as $F_{stat.P10}$.

3.3.8 90th percentile 8DWT

P_{90} is the 90th percentile of $\mathbf{X}_{gl}$. P_{90} is more robust to grey level outliers than the *maximum grey level* and is defined as $F_{stat.P90}$.

3.3.9 Maximum grey level 84IY

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

3.3.10 Interquartile range SALO

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 percentiles of $\mathbf{X}_{gl}$, respectively.

3.3.11 Range 2OJQ

The *range* of grey levels is defined as:

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

3.3.12 Mean absolute deviation 4FUA

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

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

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

3.3.13 Robust mean absolute deviation 1128

The *mean absolute deviation* feature may be influenced by outliers. To increase robustness, the set of grey levels can be restricted to those which lie closer to the center of the distribution. Let

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

Thus $\mathbf{X}_{gl,10-90}$ is the set of $N_{v,10-90} \leq N_v$ 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:

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

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

3.3.14 Median absolute deviation N72L

Median absolute deviation is similar in concept to *mean absolute deviation*, but measures dispersion from the median instead of mean. Thus:

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

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

3.3.15 Coefficient of variation 7TET

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.

3.3.16 Quartile coefficient of dispersion 9S40

The *quartile coefficient of dispersion* is a more 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 25th and 75th percentile of $\mathbf{X}_{gl}$, respectively.

3.3.17 Energy N8CA

Energy[1] of $\mathbf{X}_{gl}$ is defined as:

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

3.3.18 Root mean square 5ZWQ

The *root mean square* feature[1], which also called the *quadratic mean*, of $\mathbf{X}_{gl}$ is defined as:

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

3.4 Intensity histogram features ZVCW

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 in Section 2.7.

Let $\mathbf{X}_d = \{X_{d,1}, X_{d,2}, \ldots, X_{d,N_v}\}$ be the set of N_g discretised grey levels of the N_v voxels in the ROI intensity mask. Let $\mathbf{H} = \left\{n_1, n_2, \ldots, n_{N_g}\right\}$ be the histogram with frequency count n_i of each discretised grey level i in $\mathbf{X}_d$. The occurrence probability p_i for each grey level bin i is then approximated as $p_i = n_i/N_v$.

Aggregating features

We recommend calculating intensity histogram features using the 3D volume (`DHQ4`). Computing features per slice and subsequently averaging (`3IDG`) is not recommended.

3.4.1 Intensity histogram mean X6K6

The *mean*^{1} of $\mathbf{X}_d$ is calculated as:

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

An equivalent formulation is:

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

3.4.2 Intensity histogram variance CH89

The *variance*^{1} of $\mathbf{X}_d$ is defined as:

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

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

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

3.4.3 Intensity histogram skewness 88K1

The *skewness*^{1} of $\mathbf{X}_d$ is defined as:

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

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

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

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

3.4.4 Intensity histogram (excess) kurtosis C3I7

Kurtosis[1], or technically excess kurtosis, is calculated as measure of peakedness of the distribution $\mathbf{X}_d$:

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

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

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

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

3.4.5 Intensity histogram median WIFQ

The *median* $F_{\mathit{ih.median}}$ is the sample median of $\mathbf{X}_d$[1].

3.4.6 Intensity histogram minimum grey level 1PR8

The *minimum grey level*[1] $F_{\mathit{ih.min}}$ is equal to the lowest discretised grey level present in $\mathbf{X}_d$. For *fixed bin number* discretisation $F_{\mathit{ih.min}} = 1$ by definition, but it may deviate for *fixed bin size* discretisation.

3.4.7 Intensity histogram 10th percentile GPMT

P_{10} is the 10th percentile of $\mathbf{X}_d$ and is defined as $F_{\mathit{ih.P10}}$.

3.4.8 Intensity histogram 90th percentile OZOC

P_{90} is the 90th percentile of $\mathbf{X}_d$ and is defined as $F_{\mathit{ih.P90}}$.

3.4.9 Intensity histogram maximum grey level 3NCY

The *maximum grey level*[1] $F_{\mathit{ih.max}}$ is equal to the highest discretised grey level present in $\mathbf{X}_d$. $F_{\mathit{ih.max}} = N_g$ by definition.

3.4.10 Intensity histogram mode AMMC

The *mode* of $\mathbf{X}_d$ $F_{\mathit{ih.mode}}$ 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 $F_{\mathit{ih.mode}}$. In pathological cases with two such bins equidistant to the mean, the bin to the left of the mean is selected.

3.4.11 Intensity histogram interquartile range WR00

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.

3.4.12 Intensity histogram range 5Z3W

The *range* of grey levels[1] in the histogram is defined as:

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

The *intensity histogram range* is therefore equal to the width of the histogram. For *fixed bin number* discretisation $F_{ih.range} = N_g$ by definition.

3.4.13 Intensity histogram mean absolute deviation D2ZX

The *mean absolute deviation*[1] 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 $\mu = F_{ih.mean}$.

3.4.14 Intensity histogram robust mean absolute deviation WRZB

Intensity histogram mean absolute deviation may be affected by outliers. To increase robustness, the set of discretised grey levels under consideration can be restricted to those which are closer to the center of the distribution. Let

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

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

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

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

3.4.15 Intensity histogram median absolute deviation 4RNL

Histogram median absolute deviation is conceptually similar to *histogram mean absolute deviation*, but measures dispersion from the median instead of mean. Thus:

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

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

3.4.16 Intensity histogram coefficient of variation CWYJ

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.

3.4.17 Intensity histogram quartile coefficient of dispersion SLWD

The *quartile coefficient of dispersion* is a more 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 25th and 75th percentile of $\mathbf{X}_d$, respectively.

3.4.18 Intensity histogram entropy TLU2

Entropy[1] is an 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:

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

3.4.19 Intensity histogram uniformity BJ5W

Uniformity[1] of $\mathbf{X}_d$ is defined as:

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

Note that this feature is sometimes also referred to as *energy*.

3.4.20 Maximum histogram gradient 12CE

The histogram gradient $\mathbf{H}'$ of intensity histogram $\mathbf{H}$ can be calculated as:

$$H'_i = \begin{cases} n_2 - n_1 & i = 1 \\ \left(n_{i+1} - n_{i-1}\right)/2 & 1 < i < N_g \\ n_{N_g} - n_{N_g - 1} & i = N_g \end{cases}$$

Histogram $\mathbf{H}$ should be non-sparse, i.e. bins where $n_i = 0$ should not be omitted. Ostensibly, the histogram gradient can be calculated in different ways. The method above has the advantages of being easy to implement and leading to a gradient $\mathbf{H}'$ with same size as $\mathbf{H}$. This helps avoid ambiguity concerning correspondence between the discretised grey level and the bin. The *maximum histogram gradient*[76] is:

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

3.4.21 Maximum histogram gradient grey level 8E6O

The *maximum histogram gradient grey level*[76] $F_{ih.max.grad.gl}$ is the discretised grey level corresponding to the *maximum histogram gradient*, i.e. i for which $\mathbf{H}'$ was maximal.

3.4.22 Minimum histogram gradient VQB3

The *minimum histogram gradient*[76] is:

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

3.4.23 Minimum histogram gradient grey level RHQZ

The *minimum histogram gradient grey level*[76] $F_{ih.min.grad.gl}$ is the discretised grey level corresponding to the *minimum histogram gradient*, i.e. i for which $\mathbf{H}'$ was minimal.

3.5 Intensity-volume histogram features P88C

The (cumulative) intensity-volume histogram (IVH) of the voxel grey level set $\mathbf{X}_{gl}$ of the ROI intensity mask describes the relationship between discretised grey level i and the fraction of the volume containing at least grey level i, ν[26]. Dependent on the imaging modality, the calculation of IVH features requires discretising $\mathbf{X}_{gl}$ to generate a discretised grey level voxel set $\mathbf{X}_{d,gl}$. Moreover, the total range $\mathbf{G}$ of discretised grey level values with discretisation interval w_d should be provided or determined. The total range determines the range of discretised grey level values to be included in the IVH, whereas the discretisation interval determines the difference between adjacent discretised grey levels in the IVH. For images with definite intensity units, the discretisation interval matches the bin width for discretisation. However, it differs for images with arbitrary intensity units. For consistency of IVH metric comparisons, it is recommended to use a range $\mathbf{G}$ as defined by the re-segmentation range whenever it is possible for imaging modalities with definite intensity units (both discrete and continuous cases).

Definite intensity units – discrete case

Some imaging modalities by default generate voxels with calibrated, discrete intensities – for example CT. In this case, the discretised ROI voxel set $\mathbf{X}_{d,gl} = \mathbf{X}_{gl}$ (i.e. no discretisation required). If a re-segmentation range is provided (see Section 2.5), the total range $\mathbf{G}$ is equal to the re-segmentation range. In the case of a half-open re-segmentation range, the upper limit of the range is $\max(\mathbf{X}_{gl})$. When no range is provided, $\mathbf{G} = [\min(\mathbf{X}_{gl}), \max(\mathbf{X}_{gl})]$. The discretisation interval is $w_d = 1$.

Definite intensity units – continuous case

Imaging with calibrated, continuous intensities such as PET requires discretisation to determine the IVH, while preserving the quantitative intensity information. The use of a *fixed bin size* discretisation method is thus recommended, see Section 2.7. Proper use of this method requires to set the minimum grey level $X_{gl,min}$, the maximum grey level $X_{gl,max}$ and the bin width w_b prior to discretisation. If a re-segmentation range is defined (see Section 2.5), $X_{gl,min}$ is set to the lower bound of the re-segmentation range and $X_{gl,max}$ to the upper bound; otherwise $X_{gl,min} = \min(\mathbf{X}_{gl})$ and $X_{gl,max} = \max(\mathbf{X}_{gl})$ (i.e. the minimum and maximum grey levels in the imaging volume prior to discretisation). The bin width w_b is modality dependent, but should be small relative to the intensity range, e.g. 0.10 SUV for ^{18}F-FDG-PET.

Next, *fixed bin size* discretisation produces the voxel set $\mathbf{X}_d$ of bin numbers, which needs to be converted to bin centers in order to maintain a direct relationship with the original intensities. We thus replace bin numbers $\mathbf{X}_d$ with the intensity corresponding to the bin center:

$$\mathbf{X}_{d,gl} = X_{gl,min} + (\mathbf{X}_d - 0.5)\, w_b$$

The total range is then $\mathbf{G} = [X_{gl,min} + 0.5w_b, X_{gl,max} - 0.5w_b]$. In this case, the discretisation interval matches the bin width, i.e. $w_d = w_b$.

Arbitrary intensity units

Some imaging modalities such as raw MRI data have arbitrary intensities. In such cases, a *fixed bin number* discretisation method with $N_g = 1000$ bins is recommended, see Section 2.7. The discretisation bin width is $w_b = (X_{gl,max} - X_{gl,min})/N_g$, with $X_{gl,max} = \max(\mathbf{X}_{gl})$ and $X_{gl,min} = \min(\mathbf{X}_{gl})$, as re-segmentation ranges generally cannot be provided for non-calibrated intensities. The *fixed bin number* discretisation produces the voxel set $\mathbf{X}_d \in \{1, 2, \ldots, N_g\}$. Because of the lack of calibration, $\mathbf{X}_{d,gl} = \mathbf{X}_d$, and consequentially the discretisation interval is $w_d = 1$ and the total range is $\mathbf{G} = [1, N_g]$

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 3.2 — Example intensity-volume histogram evaluated at discrete grey levels i of the digital phantom. The total range $\mathbf{G} = [1, 6]$, with discretisation interval $w = 1$. Thus γ is the fractional grey level and ν is the corresponding volume fraction that contains grey level i or greater.

Calculating the IV histogram

We use $\mathbf{X}_{d,gl}$ to calculate fractional volumes and fractional grey levels.

As voxels for the same image stack generally all have the same dimensions, we may define fractional volume ν for discrete grey level i in the range $\mathbf{G}$ with discretisation interval w_d as:

$$\nu_i = 1 - \frac{1}{N_v} \sum_{k=1}^{N_v} [X_{d,gl,k} < i]$$

Here $[\ldots]$ is an Iverson bracket, yielding 1 if the condition is true and 0 otherwise. 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 to find ν_i.

The grey level fraction γ for discrete grey level i in the range $\mathbf{G}$ with discretisation interval w_d is calculated as:

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

Note that we evaluate grey levels that may actually be absent in $\mathbf{X}_{d,gl}$. For the digital phantom of the benchmark data sets (Chapter 5) grey levels 2 and 5 are absent, but still evaluated to determine both the fractional volume and the grey level fraction. An example IVH for the digital phantom is shown in Table 3.2.

Aggregating features

We recommend calculating intensity-volume histogram features using the 3D volume (`DHQ4`). Computing features per slice and subsequently averaging (`3IDG`) is not recommended.

3.5.1 Volume at intensity fraction `BC2M`

The *volume at intensity fraction* V_x is the largest volume fraction ν that has an intensity fraction γ of at 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.[26] defined both V_{10} and V_{90} as features. In the context of this work, these two features are defined as $F_{\mathit{ivh.V10}}$ and $F_{\mathit{ivh.V90}}$, respectively.

3.5.2 Intensity at volume fraction GBPN

The *intensity at volume fraction* I_x is the minimum discretised grey level i present in at most $x\%$ of the volume. El Naqa et al.[26] defined both I_{10} and I_{90} as features. In the context of this work, these two features are defined as $F_{\mathit{ivh.I10}}$ and $F_{\mathit{ivh.I90}}$, respectively.

3.5.3 Volume fraction difference between intensity fractions DDTU

This feature is the difference between the volume fractions at two different intensity fractions, e.g. $V_{10} - V_{90}$[26]. In the context of this work, this feature is defined as $F_{\mathit{ivh.V10minusV90}}$.

3.5.4 Intensity fraction difference between volume fractions CNV2

This feature is the difference between discretised grey levels at two different fractional volumes, e.g. $I_{10} - I_{90}$[26]. In the context of this work, this feature is defined as $F_{\mathit{ivh.I10minusI90}}$.

3.5.5 Area under the IVH curve 9CMM

The *area under the IVH curve* $F_{\mathit{ivh.auc}}$ was defined by van Velden et al.[78]. 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, the *area under the IVH curve* $F_{\mathit{ivh.auc}} = 0$.

3.6 Grey level co-occurrence based features LFYI

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 analysis is however not restricted to 2D slices and can be extended to 3D objects. Image intensities are generally discretised before calculation of texture features, see Section 2.7.

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. Generally, the neighbourhood for GLCM is a 26-connected neighbourhood in 3D and a 8-connected neighbourhood in 2D. Thus, in 3D there are 13 unique direction vectors within the neighbourhood for Chebyshev distance $\delta = 1$, 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 $(1,-1,-1)$, whereas in 2D the direction vectors are $(1,0,0)$, $(1,1,0)$, $(0,1,0)$ and $(-1,1,0)$.

A GLCM is calculated for each direction vector, as follows. Let $\mathbf{M}_\mathbf{m}$ be the $N_g \times N_g$ grey level co-occurrence matrix, with N_g the number of discretised grey levels present in the ROI intensity mask, and $\mathbf{m}$ the particular direction vector. Element (i,j) of the GLCM contains the frequency at which combinations of discretised grey levels i and j occur in neighbouring voxels along direction $\mathbf{m}_+ = \mathbf{m}$ and along direction $\mathbf{m}_- = -\mathbf{m}$. Then, $\mathbf{M}_\mathbf{m} = \mathbf{M}_{\mathbf{m}_+} + \mathbf{M}_{\mathbf{m}_-} = \mathbf{M}_{\mathbf{m}_+} + \mathbf{M}_{\mathbf{m}_+}^T$ [36]. As a consequence the GLCM matrix $\mathbf{M}_\mathbf{m}$ is symmetric. An example of the calculation of a GLCM is shown in Table 3.3. Corresponding grey level co-occurrence matrices for each direction are shown in Table 3.4.

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

(a) Grey levels

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

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

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

(c) $\mathbf{M}_{\mathbf{m}_- = \leftarrow}$

Table 3.3 — Grey levels (a) and corresponding grey level co-occurrence matrices for the $0°$ (b) and $180°$ directions (c). In vector notation these directions are $\mathbf{m}_+ = (1,0)$ and $\mathbf{m}_- = (-1,0)$. To calculate the symmetrical co-occurrence matrix $\mathbf{M}_\mathbf{m}$ both matrices are summed by element.

GLCM features rely on the probability distribution for the elements of the GLCM. Let us consider $\mathbf{M}_{\mathbf{m}=(1,0)}$ from the example, as shown in Table 3.5. We derive a probability distribution for grey level co-occurrences, $\mathbf{P}_\mathbf{m}$, by normalising $\mathbf{M}_\mathbf{m}$ by the sum of its elements. Each element p_{ij} of $\mathbf{P}_\mathbf{m}$ is then the joint probability of grey levels i and j occurring in neighbouring voxels along direction $\mathbf{m}$. Then $p_{i.} = \sum_{j=1}^{N_g} p_{ij}$ is the row marginal probability, and $p_{.j} = \sum_{i=1}^{N_g} p_{ij}$ is the column marginal probability. As $\mathbf{P}_\mathbf{m}$ is by definition symmetric, $p_{i.} = p_{.j}$. Furthermore, let us consider diagonal and cross-diagonal probabilities p_{i-j} and p_{i+j}[36,71]:

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

$$p_{i+j,k} = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p_{ij}\ [k = i+j] \qquad k = 2, \ldots, 2N_g$$

Here, $[\ldots]$ is an Iverson bracket, which equals 1 when the condition within the brackets is true and 0 otherwise. In effect we select only combinations of elements (i,j) for which the condition holds.

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

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

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

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

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

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

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

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

Table 3.4 — Grey level co-occurrence matrices for the $0°$ (a), $45°$ (b), $90°$ (c) and $135°$ (d) directions. In vector notation these directions are $\mathbf{m} = (1,0)$, $\mathbf{m} = (1,1)$, $\mathbf{m} = (0,1)$ and $\mathbf{m} = (-1,1)$, respectively.

It should be noted that while a distance $\delta = 1$ is commonly used for GLCM, other distances are possible. However, this does not change the number of For example, for $\delta = 3$ (in 3D) 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.

Aggregating features

To improve rotational invariance, GLCM feature values are computed by aggregating information from the different underlying directional matrices[22]. Five methods can be used to aggregate GLCMs and arrive at a single feature value. A schematic example is shown in Figure 3.3. A feature may be aggregated as follows:

1. Features are computed from each 2D directional matrix and averaged over 2D directions and slices (BTW3).
2. Features are computed from a single matrix after merging 2D directional matrices per slice, and then averaged over slices (SUJT).
3. Features are computed from a single matrix after merging 2D directional matrices per direction, and then averaged over directions (JJUI).
4. The feature is computed from a single matrix after merging all 2D directional matrices (ZW7Z).
5. Features are computed from each 3D directional matrix and averaged over the 3D directions (ITBB).
6. The feature is computed from a single matrix after merging all 3D directional matrices (IAZD).

In methods 2,3,4 and 6, matrices are merged by summing the co-occurrence counts in each matrix element (i,j) over the different matrices. Probability distributions are subsequently calculated for the merged GLCM, which is then used to calculate GLCM features. Feature values may dependent strongly on the aggregation method.

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

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

	j				$p_{i.}$
i	0.00	0.13	0.00	0.08	**0.21**
	0.13	0.08	0.13	0.08	**0.42**
	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) $\mathbf{P}_{\mathbf{m}=(1,0)}$ 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 $\mathbf{P}_{\mathbf{m}=(1,0)}$

$k = i+j$	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 $\mathbf{P}_{\mathbf{m}=(1,0)}$

Table 3.5 — Grey level co-occurrence matrix for the $0°$ direction (a); its corresponding probability matrix $\mathbf{P}_{\mathbf{m}=(1,0)}$ 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 caused by showing only two decimal places. Also, note that due to GLCM symmetry marginal probabilities $p_{i.}$ and $p_{.j}$ are the same in both row and column margins of panel b.

Distances and distance weighting

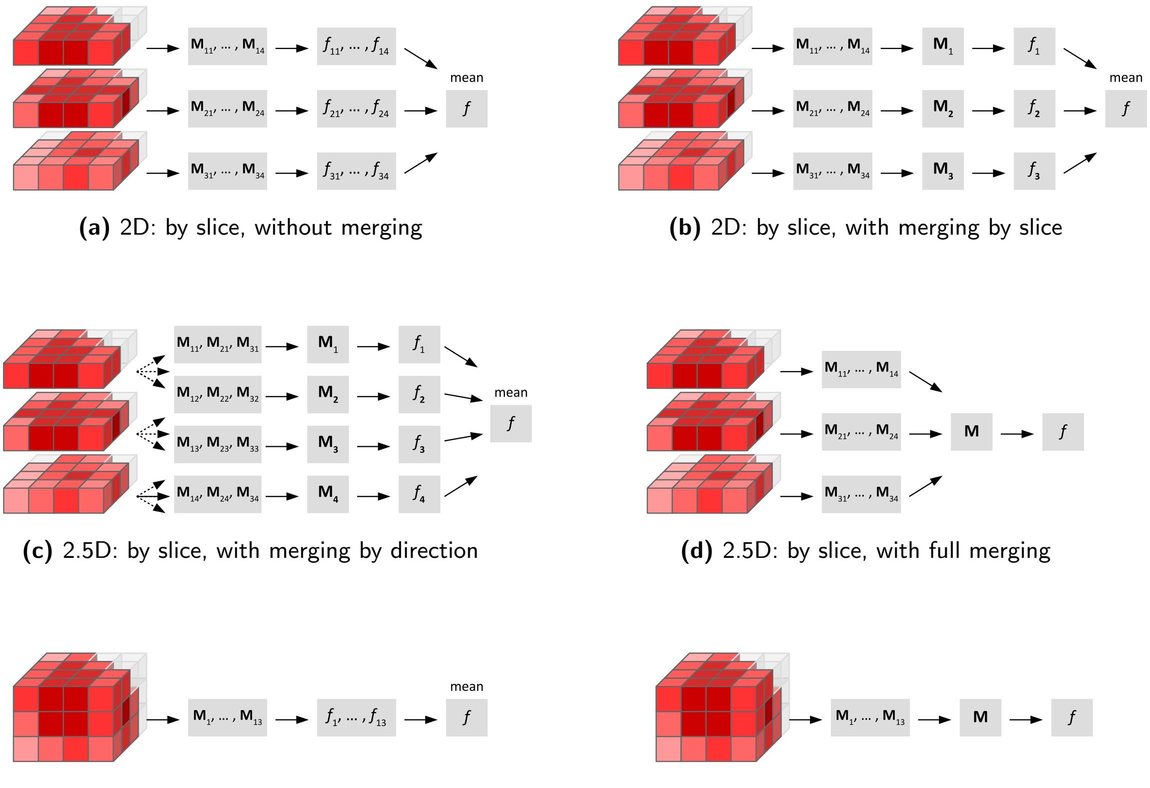
The default neighbourhood includes all voxels within Chebyshev distance 1. The corresponding direction vectors are multiplied by the desired distance δ. From a technical point-of-view, direction vectors may also be determined differently, using any distance norm. In this case, direction vectors are the vectors to the voxels at δ, or between δ and $\delta - 1$ for the Euclidean norm. Such usage is however rare and we caution against it due to potential reproducibility issues.

GLCMs may be weighted for distance by multiplying $\mathbf{M}$ with a weighting factor w. By default $w = 1$, but w may also be an inverse distance function to weight each GLCM, e.g. $w = \|\mathbf{m}\|^{-1}$ or $w = \exp(-\|\mathbf{m}\|^2)$[77], with $\|\mathbf{m}\|$ the length of direction vector m. Whether distance weighting yields different feature values depends on several factors. When aggregating the feature values, matrices have to be merged first, otherwise weighting has no effect. Also, it has no effect if the default neighbourhood is used and the Chebyshev norm is using for weighting. Nor does weighting have an effect if either Manhattan or Chebyshev norms are used both for constructing a non-default neighbourhood and for weighting. Weighting may furthermore have no effect for distance $\delta = 1$, dependent on distance norms. Because of these exceptions, we recommend against using distance weighting for GLCM.

3.6.1 Joint maximum GYBY

Joint maximum[35] is the probability corresponding to the most common grey level co-occurrence in the GLCM:

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



(a) 2D: by slice, without merging

(b) 2D: by slice, with merging by slice

(c) 2.5D: by slice, with merging by direction

(d) 2.5D: by slice, with full merging

(e) 3D: as volume, without merging

(f) 3D: as volume, with full merging

Figure 3.3 — Approaches to calculating grey level co-occurrence matrix-based features. $\mathbf{M}_{\Delta k}$ are texture matrices calculated for direction Δ in slice k (if applicable), and $f_{\Delta k}$ is the corresponding feature value. In (b-d) and (e) the matrices are merged prior to feature calculation.

3.6.2 Joint average

60VM

Joint average[71] is the grey level weighted sum of joint probabilities:

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

3.6.3 Joint variance

UR99

The *joint variance*[71], which is also called *sum of squares*[36], is defined as:

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

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

3.6.4 Joint entropy

TU9B

Joint entropy[36] is defined as:

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

3.6.5 Difference average TF7R

The *difference average*[71] for the diagonal probabilities is defined as:

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

By definition *difference average* is equivalent to the *dissimilarity* feature[77].

3.6.6 Difference variance D3YU

The *difference variance* for the diagonal probabilities[36] 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 *difference average*.

3.6.7 Difference entropy NTRS

The *difference entropy* for the diagonal probabilities[36] is defined as:

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

3.6.8 Sum average ZGXS

The *sum average* for the cross-diagonal probabilities[36] is defined as:

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

By definition, $F_{cm.sum.avg} = 2F_{cm.joint.avg}$[77].

3.6.9 Sum variance OEEB

The *sum variance* for the cross-diagonal probabilities[36] 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 *sum average*. *Sum variance* is mathematically identical to the *cluster tendency* feature[77].

3.6.10 Sum entropy P6QZ

The *sum entropy* for the cross-diagonal probabilities[36] is defined as:

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

3.6.11 Angular second moment 8ZQL

The *angular second moment*[36], which represents the energy of $\mathbf{P}_\Delta$, is defined as:

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

This feature is also called *energy*[1,71] and *uniformity*[17].

3.6.12 Contrast ACUI

Contrast assesses grey level variations[36]. Hence elements of $\mathbf{M}_\Delta$ that represent large grey level differences receive greater weight. *Contrast* is defined as[17]:

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

Note that the original definition by Haralick et al.[36] is seemingly more complex, but rearranging and simplifying terms leads to the above formulation of *contrast*.

3.6.13 Dissimilarity 8S9J

Dissimilarity[17] is conceptually similar to the *contrast* feature, and is defined as:

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

By definition *dissimilarity* is equivalent to the *difference average* feature[77].

3.6.14 Inverse difference IB1Z

Inverse difference is a measure of homogeneity[17]. Grey level co-occurrences with a large difference in levels are weighed less, thus lowering the total feature value. The feature score is maximal if all grey levels are the same. Inverse difference is defined as:

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

The equation above may also be expressed in terms of diagonal probabilities[77]:

$$F_{cm.inv.diff} = \sum_{k=0}^{N_g-1} \frac{p_{i-j,k}}{1+k}$$

3.6.15 Normalised inverse difference NDRX

Clausi[17] suggested normalising *inverse difference* to improve classification ability. The normalised 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 matches the definition of the *normalised inverse difference moment* feature.

The equation may also be expressed in terms of diagonal probabilities[77]:

$$F_{cm.inv.diff.norm} = \sum_{k=0}^{N_g-1} \frac{p_{i-j,k}}{1+k/N_g}$$

3.6.16 Inverse difference moment WFOZ

Inverse difference moment[36] is similar in concept to the *inverse difference* feature, but with lower weights for elements that are further from the diagonal:

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

The equation above may also be expressed in terms of diagonal probabilities[77]:

$$F_{cm.inv.diff.mom} = \sum_{k=0}^{N_g-1} \frac{p_{i-j,k}}{1+k^2}$$

This feature is also called *homogeneity*[71].

3.6.17 Normalised inverse difference moment 1QCO

Clausi[17] suggested normalising *inverse difference moment* to improve classification performance. This leads to the following definition:

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

The equation above may also be expressed in terms of diagonal probabilities[77]:

$$F_{cm.inv.diff.mom.norm} = \sum_{k=0}^{N_g-1} \frac{p_{i-j,k}}{1+(k/N_g)^2}$$

3.6.18 Inverse variance E8JP

The *inverse variance*[1] feature is defined as:

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

The equation above may also be expressed in terms of diagonal probabilities. Note that in this case, summation starts at $k=1$ instead of $k=0$[77]:

$$F_{cm.inv.var} = \sum_{k=1}^{N_g-1} \frac{p_{i-j,k}}{k^2}$$

3.6.19 Correlation NI2N

Correlation[36] 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 of the column marginal probability $p_{.j}$, respectively. The calculation of *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:

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

3.6.20 Autocorrelation QWBO

Soh and Tsatsoulis[61] defined *autocorrelation* as:

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

3.6.21 Cluster tendency DG8W

Cluster tendency[1] is defined as:

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

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:

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

Cluster tendency is mathematically equal to the *sum variance* feature[77].

3.6.22 Cluster shade 7NFM

Cluster shade[71] is defined as:

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

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:

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

3.6.23 Cluster prominence AE86

Cluster prominence[71] is defined as:

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

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:

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

3.6.24 First measure of information correlation R8DG

Information theoretic correlation is estimated using two different measures[36]. 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:

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

3.6.25 Second measure of information correlation JN9H

The *second measure of information theoretic correlation*[36] is estimated as follows for symmetric $\mathbf{P}_\Delta$:

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

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:

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

3.7 Grey level run length based features TP0I

The grey level run length matrix (GLRLM) was introduced by Galloway[29] to define various texture 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, whereas GLCM assesses co-occurrence of grey levels within neighbouring pixels or voxels, GLRLM assesses run lengths. A run length is defined as the length of a consecutive sequence of pixels or voxels with the same grey level along direction $\mathbf{m}$, which was previously defined in Section 3.6. The GLRLM then contains the occurrences of runs with length j for a discretised grey level i.

A complete example for GLRLM construction from a 2D image is shown in Table 3.6. Let $\mathbf{M_m}$ 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 $\mathbf{m}$. Matrix element r_{ij} of the GLRLM is the occurrence of grev level i with run length j. 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 $\mathbf{M_m}$. Marginal sums are also defined. Let $r_{i.}$ be the marginal sum of the runs over run lengths j for grey value i, that is $r_{i.} = \sum_{j=1}^{N_r} r_{ij}$. Similarly, the marginal sum of the runs over the grey values i for run length j is $r_{.j} = \sum_{i=1}^{N_g} r_{ij}$.

Aggregating features

To improve rotational invariance, GLRLM feature values are computed by aggregating information from the different underlying directional matrices[22]. Five methods can be used to aggregate GLRLMs and arrive at a single feature value. A schematic example was previously shown in Figure 3.3. A feature may be aggregated as follows:

1. Features are computed from each 2D directional matrix and averaged over 2D directions and slices (BTW3).
2. Features are computed from a single matrix after merging 2D directional matrices per slice, and then averaged over slices (SUJT).
3. Features are computed from a single matrix after merging 2D directional matrices per direction, and then averaged over directions (JJUI).
4. The feature is computed from a single matrix after merging all 2D directional matrices (ZW7Z).
5. Features are computed from each 3D directional matrix and averaged over the 3D directions (ITBB).
6. The feature is computed from a single matrix after merging all 3D directional matrices (IAZD).

In methods 2,3,4 and 6 matrices are merged by summing the run counts of each matrix element (i, j) over the different matrices. Note that when matrices are merged, N_v should likewise be summed to retain consistency. Feature values may dependent strongly on the aggregation method.

Distance weighting

GLRLMs may be weighted for distance by multiplying the run lengths with a weighting factor w. By default $w = 1$, but w may also be an inverse distance function, e.g. $w = \|\mathbf{m}\|^{-1}$ or $w = \exp(-\|\mathbf{m}\|^2)$[77], with $\|\mathbf{m}\|$ the length of direction vector m. Whether distance weighting yields different feature values depends on several factors. When aggregating the feature values, matrices have to be merged first, otherwise weighting has no effect. It also has no effect if the Chebyshev norm is used for weighting. Distance weighting is non-standard use, and we caution against it due to potential reproducibility issues.

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

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

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

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

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

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

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

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

Table 3.6 — Grey level run length matrices for the $0°$ (a), $45°$ (b), $90°$ (c) and $135°$ (d) directions. In vector notation these directions are $\mathbf{m} = (1, 0)$, $\mathbf{m} = (1, 1)$, $\mathbf{m} = (0, 1)$ and $\mathbf{m} = (-1, 1)$, respectively.

3.7.1 Short runs emphasis 220V

This feature emphasises short run lengths[29]. It is defined as:

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

3.7.2 Long runs emphasis W4KF

This feature emphasises long run lengths[29]. It is defined as:

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

3.7.3 Low grey level run emphasis V3SW

This feature is a grey level analogue to *short runs emphasis*[15]. Instead of short run lengths, low grey levels are emphasised. The feature is defined as:

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

3.7.4 High grey level run emphasis G3QZ

The *high grey level run emphasis* feature is a grey level analogue to *long runs emphasis*[15]. The feature emphasises high grey levels, and is defined as:

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

3.7.5 Short run low grey level emphasis HTZT

This feature emphasises runs in the upper left quadrant of the GLRLM, where short run lengths and low grey levels are located[21]. It is defined as:

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

3.7.6 Short run high grey level emphasis GD3A

This feature emphasises runs in the lower left quadrant of the GLRLM, where short run lengths and high grey levels are located[21]. The feature is defined as:

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

3.7.7 Long run low grey level emphasis IVPO

This feature emphasises runs in the upper right quadrant of the GLRLM, where long run lengths and low grey levels are located[21]. The feature is defined as:

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

3.7.8 Long run high grey level emphasis 3KUM

This feature emphasises runs in the lower right quadrant of the GLRLM, where long run lengths and high grey levels are located[21]. The feature is defined as:

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

3.7.9 Grey level non-uniformity R5YN

This feature assesses the distribution of runs over the grey values[29]. The feature value is low when runs are equally distributed along grey levels. The feature is defined as:

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

3.7.10 Normalised grey level non-uniformity 0VBL

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

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

3.7.11 Run length non-uniformity W92Y

This features assesses the distribution of runs over the run lengths[29]. The feature value is low when runs are equally distributed along run lengths. It is defined as:

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

3.7.12 Normalised run length non-uniformity IC23

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

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

3.7.13 Run percentage 9ZK5

This feature measures the fraction of the number of realised runs and the maximum number of potential runs[29]. 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 before, 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.

3.7.14 Grey level variance 8CE5

This feature estimates the variance in runs over 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. *Grey level variance* is then defined as:

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

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

3.7.15 Run length variance SXLW

This feature estimates the variance in runs over the 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 $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} j\, p_{ij}$.

3.7.16 Run entropy HJ90

Run entropy was investigated by Albregtsen et al.[3]. Again, let $p_{ij} = r_{ij}/N_s$. The entropy is then defined as:

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

3.8 Grey level size zone based features 9SAK

The grey level size zone matrix (GLSZM) counts the number of groups (or zones) of linked voxels[69]. Voxels are linked if the neighbouring voxel has an identical discretised grey level. Whether a voxel classifies as a neighbour depends on its connectedness. In a 3D approach to texture analysis we consider 26-connectedness, which indicates that a center voxel is linked to all of the 26 neighbouring voxels with the same grey level. In the 2 dimensional approach, 8-connectedness is used. A potential issue for the 2D approach is that voxels which may otherwise be considered to belong to the same zone by linking across slices, are now two or more separate zones within the slice plane. Whether this issue negatively affects predictive performance of GLSZM-based features or their reproducibility has not been determined.

Let $\mathbf{M}$ 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 any group of linked voxels. Element s_{ij} of $\mathbf{M}$ is then the 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 3.7.

Aggregating features

Three methods can be used to aggregate GLSZMs and arrive at a single feature value. A schematic example is shown in Figure 3.4. A feature may be aggregated as follows:

1. Features are computed from 2D matrices and averaged over slices (8QNN).
2. The feature is computed from a single matrix after merging all 2D matrices (62GR).
3. The feature is computed from a 3D matrix (KOBO).

Method 2 involves merging GLSZMs by summing the number of zones s_{ij} over the GLSZM for the different slices. Note that when matrices are merged, N_v should likewise be summed to retain consistency. Feature values may dependent strongly on the aggregation method.

Distances

The default neighbourhood for GLSZM is constructed using Chebyshev distance $\delta = 1$. Manhattan or Euclidean norms may also be used to construct a neighbourhood, and both lead to a 6-connected (3D) and 4-connected (2D) neighbourhoods. Larger distances are also technically possible, but will occasionally cause separate zones with the same intensity to be considered as belonging to the same zone. Using different neighbourhoods for determining voxel linkage is non-standard use, and we caution against it due to potential reproducibility issues.

Note on feature references

GLSZM feature definitions are based on the definitions of GLRLM features[69]. Hence, references may be found in the section on GLRLM (3.7).

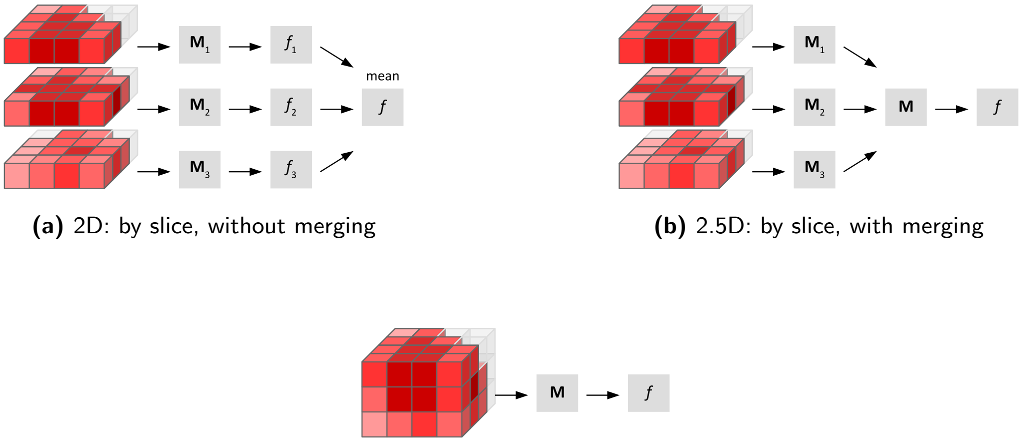
1	2	2	3
1	2	3	3
4	2	4	1
4	1	2	3

(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 3.7 — Original image with grey levels (a); and corresponding grey level size zone matrix (GLSZM) under 8-connectedness (b). Element $s(i,j)$ of the GLSZM indicates the number of times a zone of j linked pixels and grey level i occurs within the image.



(a) 2D: by slice, without merging

(b) 2.5D: by slice, with merging

(c) 3D: as volume

Figure 3.4 — Approaches to calculating grey level size zone matrix-based features. $\mathbf{M}_k$ 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.

3.8.1 Small zone emphasis 5QRC

This feature emphasises small zones. It is defined as:

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

3.8.2 Large zone emphasis 48P8

This feature emphasises large zones. It is defined as:

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

3.8.3 Low grey level zone emphasis XMSY

This feature is a grey level analogue to *small zone emphasis.* Instead of small zone sizes, low grey levels are emphasised. The feature is defined as:

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

3.8.4 High grey level zone emphasis 5GN9

The *high grey level zone emphasis* feature is a grey level analogue to *large zone emphasis.* The feature emphasises high grey levels, and is defined as:

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

3.8.5 Small zone low grey level emphasis 5RAI

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

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

3.8.6 Small zone high grey level emphasis HW1V

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

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

3.8.7 Large zone low grey level emphasis YH51

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

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

3.8.8 Large zone high grey level emphasis J17V

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

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

3.8.9 Grey level non-uniformity JNSA

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:

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

3.8.10 Normalised grey level non-uniformity Y1RO

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

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

3.8.11 Zone size non-uniformity 4JP3

This features assesses the distribution of zone counts over the different zone sizes. *Zone size non-uniformity* is low when zone counts are equally distributed along zone sizes. It is defined as:

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

3.8.12 Normalised zone size non-uniformity VB3A

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

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

3.8.13 Zone percentage P30P

This feature measures the fraction of the number of realised zones and the maximum number of potential zones. Highly uniform ROIs produce a low *zone percentage.* It is defined as:

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

3.8.14 Grey level variance BYLV

This feature estimates the variance in zone counts over the grey levels. Let $p_{ij} = s_{ij}/N_s$ be the joint probability estimate for finding zones with discretised grey level i and 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, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_z} i\, p_{ij}$.

3.8.15 Zone size variance 3NSA

This feature estimates the variance in zone counts over 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 $\mu = \sum_{i=1}^{N_g}\sum_{j=1}^{N_z} j\, p_{ij}$.

3.8.16 Zone size entropy GU8N

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

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

3.9 Grey level distance zone based features VMDZ

The grey level distance zone matrix (GLDZM) counts the number of groups (or zones) of linked voxels which share a specific discretised grey level value and possess the same distance to ROI edge[69]. The GLDZM thus captures the relation between location and grey level. Two maps are required to calculate the GLDZM. The first is a grey level zone map, which is identical to the one created for the grey level size zone matrix (GLSZM), see Section 3.8. The second is a distance map, which will be described in detail later.

As with GSLZM, neighbouring voxels are linked if they share the same grey level value. Whether a voxel classifies as a neighbour depends on its connectedness. We consider 26-connectedness for a 3D approach and 8-connectedness in the 2D approach.

The distance to the ROI edge is defined according to 6 and 4-connectedness for 3D and 2D, respectively. Because of the connectedness definition used, the distance of a voxel to the outer border is equal to the minimum number edges of neighbouring voxels that need to be crossed to reach the ROI edge. The distance for a linked group of voxels with the same grey value is equal to the minimum distance for the respective voxels in the distance map.

Our definition deviates from the original by Thibault et al.[69]. 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 iteratively in 6 or 4-connectedness is a relatively efficient solution, implemented as follows:

1. The ROI mask is morphologically eroded using the appropriate (6 or 4-connected) structure element.
2. All eroded ROI voxels are updated in the distance map by adding 1.
3. The above steps are performed iteratively until the ROI mask is empty.

A second difference with the original definition 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 $\mathbf{M}$ 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 largest distance of any zone. Element $d_{ij} = d(i,j)$ of $\mathbf{M}$ is then number of zones with discretised grey level i and distance 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 zone count. 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 distance j, regardless of grey level. A two dimensional example is shown in Table 3.8.

Morphological and intensity masks.

The GLDZM is special in that it uses both ROI masks. The distance map is determined using the morphological ROI mask, whereas the intensity mask is used for determining the zones, as with the GLSZM.

Aggregating features

Three methods can be used to aggregate GLDZMs and arrive at a single feature value. A schematic example was previously shown in Figure 3.4. A feature may be aggregated as follows:

1. Features are computed from 2D matrices and averaged over slices (8QNN).
2. The feature is computed from a single matrix after merging all 2D matrices (62GR).

3. The feature is computed from a 3D matrix (KOBO).

Method 2 involves merging GLDZMs by summing the number of zones d_{ij} over the GLDZM for the different slices. Note that when matrices are merged, N_v should likewise be summed to retain consistency. Feature values may dependent strongly on the aggregation method.

Distances

In addition to the use of different distance norms to determine voxel linkage, as described in section 3.8, different distance norms may be used to determine distance of zones to the boundary. The default is to use the Manhattan norm which allows for a computationally efficient implementation, as described above. A similar implementation is possible using the Chebyshev norm, as it merely changes connectedness of the structure element. Implementations using an Euclidean distance norm are less efficient as this demands searching for the nearest non-ROI voxel for each of the N_v voxels in the ROI. An added issue is that Euclidean norms may lead to a wide range of different distances j that require rounding before constructing the grey level distance zone matrix $\mathbf{M}$. Using different distance norms is non-standard use, and we caution against it due to potential reproducibility issues.

Note on feature references

GLDZM feature definitions are based on the definitions of GLRLM features[69]. Hence, references may be found in the section on GLRLM (3.7).

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

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

(c) Grey level distance zone matrix

Table 3.8 — 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 (c). 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.

3.9.1 Small distance emphasis OGBI

This feature emphasises small distances. It is defined as:

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

3.9.2 Large distance emphasis MB4I

This feature emphasises large distances. It is defined as:

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

3.9.3 Low grey level zone emphasis S1RA

This feature is a grey level analogue to *small distance emphasis*. Instead of small zone distances, low grey levels are emphasised. The feature is defined as:

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

3.9.4 High grey level zone emphasis K26C

The *high grey level zone emphasis* feature is a grey level analogue to *large distance emphasis*. The feature emphasises high grey levels, and is defined as:

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

3.9.5 Small distance low grey level emphasis RUVG

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:

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

3.9.6 Small distance high grey level emphasis DKNJ

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

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

3.9.7 Large distance low grey level emphasis A7WM

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:

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

3.9.8 Large distance high grey level emphasis KLTH

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

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

3.9.9 Grey level non-uniformity VFT7

This feature measures the distribution of zone counts over the grey values. *Grey level non-uniformity* is low when zone counts are equally distributed along grey levels. The feature is defined as:

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

3.9.10 Normalised grey level non-uniformity 7HP3

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

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

3.9.11 Zone distance non-uniformity V294

Zone distance non-uniformity measures the distribution of zone counts over the different zone distances. *Zone distance non-uniformity* is low when zone counts are equally distributed along zone distances. It is defined as:

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

3.9.12 Zone distance non-uniformity normalised IATH

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

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

3.9.13 Zone percentage VIWW

This feature measures the fraction of the number of realised zones and the maximum number of potential zones. Highly uniform ROIs produce a low *zone percentage*. It is defined as:

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

3.9.14 Grey level variance QK93

This feature estimates the variance in zone counts over 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, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_d} i\, p_{ij}$.

3.9.15 Zone distance variance 7WT1

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

3.9.16 Zone distance entropy GBDU

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

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

3.10 Neighbourhood grey tone difference based featuresIPET

Amadasun and King[5] introduced an alternative to the grey level co-occurrence 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 Chebyshev distance δ. For 3D volumes, we can extend the original definition by Amadasun and King. Let $X_{d,k}$ be the discretised grey level of a voxel at position $\mathbf{k} = (k_x, k_y, k_z)$. Then the average grey level within a neighbourhood centred at (k_x, k_y, k_z), but excluding (k_x, k_y, k_z) itself is:

$$\overline{X}_k = \frac{1}{W} \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} X_d(k_x{+}m_x, k_y{+}m_y, k_z{+}m_z)$$

$$(m_x, m_y, m_z) \neq (0,0,0)$$

$W = (2\delta+1)^3 - 1$ is the size of the 3D neighbourhood. For 2D $W = (2\delta+1)^2 - 1$, and averages are not calculated between different slices. Neighbourhood grey tone difference s_i for discretised grey level i is then:

$$s_i = \sum_{k}^{N_v} |i - \overline{X}_k| \left[X_d(\mathbf{k}) = i \text{ and } k\text{has a valid neighbourhood}\right]$$

Here, $[\ldots]$ is an Iverson bracket, which is 1 if the conditions that the grey level $X_{d,k}$ of voxel k is equal to i and the voxel has a valid neighbourhood are both true; it is 0 otherwise. N_v is the number of voxels in the ROI intensity mask.

A 2D example is shown in Table 3.9. A distance of $\delta = 1$ is used in this example, leading to 8 neighbouring pixels. Entry $s_1 = 0$ 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 $s_4 = |4 - 17/8| = 1.825$.

We deviate from the original definition by Amadasun and King[5] as we do not demand that valid neighbourhoods are completely inside the ROI. In an irregular ROI mask, valid neighbourhoods may simply not exist for a distance δ. Instead, we consider a valid neighbourhood to exist if there is at least one neighbouring voxel included in the ROI mask. The average grey level for voxel k within a valid neighbourhood is then:

$$\overline{X}_k = \frac{1}{W_k} \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} X_d(\mathbf{k}+\mathbf{m})[\mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

The neighbourhood size W_k for this voxel is equal to the number of voxels in the neighbourhood that are part of the ROI mask:

$$W_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [\mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

Under our definition, neighbourhood grey tone difference s_i for discretised grey level i can be directly expressed using neighbourhood size W_k of voxel k:

$$s_i = \sum_{k}^{N_v} |i - \overline{X}_k| \left[X_d(\mathbf{k}) = i \text{ and } W_k \neq 0\right]$$

Consequentially, n_i is the total number of voxels with grey level i which have a non-zero neighbourhood size.

Many NGTDM-based features depend on the N_g grey level probabilities $p_i = n_i/N_{v,c}$, where N_g is the number of discretised grey levels in the ROI intensity mask and $N_{v,c} = \sum n_i$ is total number of voxels that have at least one neighbour. If all voxels have at least one neighbour $N_{v,c} = N_v$. 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 $N_{g,p} = 3$.

Aggregating features

Three methods can be used to aggregate NGTDMs and arrive at a single feature value. A schematic example was previously shown in Figure 3.4. A feature may be aggregated as follows:

1. Features are computed from 2D matrices and averaged over slices (`8QNN`).
2. The feature is computed from a single matrix after merging all 2D matrices (`62GR`).
3. The feature is computed from a 3D matrix (`KOBO`).

Method 2 involves merging NGTDMs by summing the neighbourhood grey tone difference s_i and the number of voxels with a valid neighbourhood n_i and grey level i for NGTDMs of the different slices. Note that when NGTDMs are merged, $N_{v,c}$ and p_i should be updated based on the merged NGTDM. Feature values may dependent strongly on the aggregation method.

Distances and distance weighting

The default neighbourhood is defined using the Chebyshev norm. Manhattan or Euclidean norms may be used as well. This requires a more general definition for the average grey level $\overline{X}_k$:

$$\overline{X}_k = \frac{1}{W_k} \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} X_d(\mathbf{k}+\mathbf{m})[\|\mathbf{m}\| \leq \delta \text{ and } \mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

The neighbourhood size W_k is:

$$W_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [\|\mathbf{m}\| \leq \delta \text{ and } \mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

As before, $[\ldots]$ is an Iverson bracket.

Distance weighting for NGTDM is relatively straightforward. Let w be a weight dependent on $\mathbf{m}$, e.g. $w = \|\mathbf{m}\|^{-1}$ or $w = \exp(-\|\mathbf{m}\|^2)$. The average grey level is then:

$$\overline{X}_k = \frac{1}{W_k} \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} w(\mathbf{m}) X_d(\mathbf{k}+\mathbf{m})[\|\mathbf{m}\| \leq \delta \text{ and } \mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

The neighbourhood size W_k becomes a general weight:

$$W_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} w(\mathbf{m})[\|\mathbf{m}\| \leq \delta \text{ and } \mathbf{m} \neq \mathbf{0} \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

Employing different distance norms and distance weighting is considered non-standard use, and we caution against them due to potential reproducibility issues.

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

(a) Grey levels

		n_i	p_i	s_i
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 3.9 — Original image with grey levels (a) and corresponding neighbourhood grey tone difference matrix (NGTDM) (b). The $N_{v,c}$ pixels with valid neighbours at distance 1 are located within the rectangle in (a). The grey level voxel count n_i, the grey level probability $p_i = n_i/N_{v,c}$, and the neighbourhood grey level difference s_i for pixels with grey level i are included in the NGTDM. Note that our actual definition deviates from the original definition of Amadasun and King[5], which is used here. In our definition complete neighbourhood are no longer required. In our definition the NGTDM would be calculated on the entire pixel area, and not solely on those pixels within the rectangle of panel (a).

3.10.1 Coarseness QCDE

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[5]. *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 *coarseness* value 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.

3.10.2 Contrast 65HE

Contrast depends on the dynamic range of the grey levels as well as the spatial frequency of intensity changes[5]. Thus, *contrast* is defined as:

$$F_{ngt.contrast} = \left(\frac{1}{N_{g,p}\left(N_{g,p}-1\right)} \sum_{i_1=1}^{N_g}\sum_{i_2=1}^{N_g} p_{i_1} p_{i_2}\left(i_1 - i_2\right)^2\right)\left(\frac{1}{N_{v,c}}\sum_{i=1}^{N_g} s_i\right)$$

Grey level probabilities p_{i_1} and p_{i_2} are copies of p_i with different iterators, i.e. $p_{i_1} = p_{i_2}$ for $i_1 = i_2$. The first term considers the grey level dynamic range, whereas the second term is a measure for intensity changes within the volume. If $N_{g,p} = 1$, $F_{ngt.contrast} = 0$.

3.10.3 Busyness NQ30

Textures with large changes in grey levels between neighbouring voxels are said to be busy[5]. *Busyness* was defined as:

$$F_{ngt.busyness} = \frac{\sum_{i=1}^{N_g} p_i\, s_i}{\sum_{i_1=1}^{N_g}\sum_{i_2=1}^{N_g} i_1\, p_{i_1} - i_2\, p_{i_2}}, \qquad p_{i_1} \neq 0 \text{ and } p_{i_2} \neq 0$$

As before, $p_{i_1} = p_{i_2}$ for $i_1 = i_2$. The original definition was erroneously formulated as the denominator will always evaluate to 0. Therefore we use a slightly different definition[38]:

$$F_{ngt.busyness} = \frac{\sum_{i=1}^{N_g} p_i\, s_i}{\sum_{i_1=1}^{N_g}\sum_{i_2=1}^{N_g} |i_1\, p_{i_1} - i_2\, p_{i_2}|}, \qquad p_{i_1} \neq 0 \text{ and } p_{i_2} \neq 0$$

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

3.10.4 Complexity HDEZ

Complex textures are non-uniform and rapid changes in grey levels are common[5]. Texture *complexity* is defined as:

$$F_{ntg.complexity} = \frac{1}{N_{v,c}} \sum_{i_1=1}^{N_g}\sum_{i_2=1}^{N_g} |i_1 - i_2| \frac{p_{i_1}\, s_{i_1} + p_{i_2}\, s_{i_2}}{p_{i_1} + p_{i_2}}, \qquad p_{i_1} \neq 0 \text{ and } p_{i_2} \neq 0$$

As before, $p_{i_1} = p_{i_2}$ for $i_1 = i_2$, and likewise $s_{i_1} = s_{i_2}$ for $i_1 = i_2$.

3.10.5 Strength 1X9X

Amadasun and King[5] defined texture *strength* as:

$$F_{ngt.strength} = \frac{\sum_{i_1=1}^{N_g}\sum_{i_2=1}^{N_g} (p_{i_1} + p_{i_2})\,(i_1 - i_2)^2}{\sum_{i=1}^{N_g} s_i}, \qquad p_{i_1} \neq 0 \text{ and } p_{i_2} \neq 0$$

As before, $p_{i_1} = p_{i_2}$ for $i_1 = i_2$. If $\sum_{i=1}^{N_g} s_i = 0$, $F_{ngt.strength} = 0$.

3.11 Neighbouring grey level dependence based featuresREKO

Sun and Wee [67] defined the neighbouring grey level dependence matrix (NGLDM) as an alternative to the grey level co-occurrence matrix. The NGLDM aims to capture the coarseness of the overall texture and is rotationally invariant.

NGLDM also involves the concept of a neighbourhood around a central voxel. All voxels within Chebyshev distance δ are considered to belong to the neighbourhood of the center voxel. The discretised grey levels of the center voxel k at position $\mathbf{k}$ and a neighbouring voxel m at $\mathbf{k}+\mathbf{m}$ are said to be dependent if $|X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha$, with α being a non-negative integer coarseness parameter. The number of grey level dependent voxels j within the neighbourhood is then counted as:

$$j_k = 1 + \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [|X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha \text{ and } \mathbf{m} \neq \mathbf{0}]$$

Here, $[\ldots]$ is an Iverson bracket, which is 1 if the aforementioned condition is fulfilled, and 0 otherwise. Note that the minimum dependence $j_k = 1$ and not $j_k = 0$. This is done because some feature definitions require a minimum dependence of 1 or are undefined otherwise. One may therefore also simplify the expression for j_k by including the center voxel:

$$j_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [|X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha]$$

Dependence j_k is iteratively determined for each voxel k in the ROI intensity mask. $\mathbf{M}$ 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 $N_n = \max(j_k)$ the maximum grey level dependence count found. Element s_{ij} of $\mathbf{M}$ is then the number of neighbourhoods with a center voxel with discretised grey level i and a neighbouring voxel dependence j. 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, and 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 3.10.

The definition we actually use deviates from the original by Sun and Wee [67]. Because regions of interest are rarely cuboid, omission of neighbourhoods which contain voxels outside the ROI mask may lead to inconsistent results, especially for larger distance δ. 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:

$$j_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [|X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

Note that while $\alpha = 0$ is a typical choice for the coarseness parameter, different α are possible. Likewise, a typical choice for neighbourhood radius δ is Chebyshev distance $\delta = 1$ but larger values are possible as well.

Aggregating features

Three methods can be used to aggregate NGLDMs and arrive at a single feature value. A schematic example was previously shown in Figure 3.4. A feature may be aggregated as follows:

1. Features are computed from 2D matrices and averaged over slices (8QNN).
2. The feature is computed from a single matrix after merging all 2D matrices (62GR).
3. The feature is computed from a 3D matrix (KOBO).

Method 2 involves merging NGLDMs by summing the dependence count s_{ij} by element over the NGLDM of the different slices. Note that when NGLDMs are merged, N_v and N_s should likewise be summed to retain consistency. Feature values may dependent strongly on the aggregation method.

Distances and distance weighting

Default neighbourhoods are constructed using the Chebyshev norm, but other norms can be used as well. For this purpose it is useful to generalise the dependence count equation to:

$$j_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} [\|\mathbf{m}\| \leq \delta \text{ and } |X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

with $\mathbf{m}$ the vector between voxels k and m and $\|\mathbf{m}\|$ its length according to the particular norm.

In addition, dependence may be weighted by distance. Let w be a weight dependent on $\mathbf{m}$, e.g. $w = \|\mathbf{m}\|^{-1}$ or $w = \exp(-\|\mathbf{m}\|^2)$. The dependence of voxel k is then:

$$j_k = \sum_{m_z=-\delta}^{\delta} \sum_{m_y=-\delta}^{\delta} \sum_{m_x=-\delta}^{\delta} w(\mathbf{m})[\|\mathbf{m}\| \leq \delta \text{ and } |X_d(\mathbf{k}) - X_d(\mathbf{k}+\mathbf{m})| \leq \alpha \text{ and } \mathbf{k}+\mathbf{m} \text{ in ROI}]$$

Employing different distance norms and distance weighting is considered non-standard use, and we caution against them due to potential reproducibility issues.

Note on feature references

The NGLDM is structured similarly to the GLRLM, GLSZM and GLDZM. NGLDM feature definitions are therefore based on the definitions of GLRLM features, and references may be found in Section 3.7, except for the features originally defined by Sun and Wee[67].

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

(a) Grey levels

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

(b) Neighbouring grey level dependence matrix

Table 3.10 — 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 our definition a complete neighbourhood is no longer required. 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).

3.11.1 Low dependence emphasis SODN

This feature emphasises low neighbouring grey level dependence counts. Sun and Wee[67] refer to this feature as *small number emphasis*. It is defined as:

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

3.11.2 High dependence emphasis IMOQ

This feature emphasises high neighbouring grey level dependence counts. Sun and Wee[67] refer to this feature as *large number emphasis*. It is defined as:

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

3.11.3 Low grey level count emphasis TL9H

This feature is a grey level analogue to *low dependence emphasis*. Instead of low neighbouring grey level dependence counts, low grey levels are emphasised. The feature is defined as:

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

3.11.4 High grey level count emphasis OAE7

The *high grey level count emphasis* feature is a grey level analogue to *high dependence emphasis*. The feature emphasises high grey levels, and is defined as:

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

3.11.5 Low dependence low grey level emphasis EQ3F

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:

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

3.11.6 Low dependence high grey level emphasis JA6D

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:

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

3.11.7 High dependence low grey level emphasis NBZI

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:

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

3.11.8 High dependence high grey level emphasis 9QMG

The *high dependence high grey level emphasis* 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:

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

3.11.9 Grey level non-uniformity FP8K

Grey level non-uniformity 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:

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

3.11.10 Normalised grey level non-uniformity 5SPA

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

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

When calculating *grey level non-uniformity normalised* using a single 3D NGLDM matrix, it is equivalent to the *intensity histogram uniformity* feature[77].

3.11.11 Dependence count non-uniformity Z87G

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[67] refer to this feature as *number non-uniformity*. It is defined as:

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

3.11.12 Dependence count non-uniformity normalised OKJI

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

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

3.11.13 Dependence count percentage 6XV8

This feature measures the fraction of the number of realised neighbourhoods and the maximum number of potential neighbourhoods. *Dependence count percentage* may be completely omitted as it evaluates to 1 when complete neighbourhoods are not required, as is the case under our definition. It is defined as:

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

3.11.14 Grey level variance 1PFV

This feature estimates the variance in dependence counts over 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, $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} i\, p_{ij}$.

3.11.15 Dependence count variance DNX2

This feature estimates the variance in dependence counts over the different possible dependence counts. 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 $\mu = \sum_{i=1}^{N_g} \sum_{j=1}^{N_n} j\, p_{ij}$.

3.11.16 Dependence count entropy FCBV

This feature is referred to as *entropy* by Sun and Wee[67]. 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 definition of Sun and Wee[67], where the term within the logarithm is dependence count s_{ij} instead of count probability p_{ij}.

3.11.17 Dependence count energy CAS9

This feature is called *second moment* by Sun and Wee[67]. 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 also remedies an error in the original definition, where squared dependence count s_{ij}^2 is divided by N_s only, thus leaving a major volume dependency. In the definition given here, s_{ij}^2 is normalised by N_s^2 through the use of count probability p_{ij}.

Chapter 4

Image biomarker reporting guidelines

Reliable and complete reporting is necessary to ensure reproducibility and validation of results. To help provide a complete report on image processing and image biomarker extraction, we present the guidelines below, as well as a nomenclature system to uniquely features.

4.1 Reporting guidelines

These guidelines are partially based on the work of Sollini et al.[62]. Additionally, guidelines are derived from the image processing and feature calculation steps described within this document. An earlier version was reported elsewhere Vallieres et al.[75].

General	
imaging	Describe which type of imaging was acquired, e.g. CT, PET.
acquisition	Describe image acquisition details, i.e. scanner vendor, method of image acquisition (e.g. dynamic, contrast-enhanced), contrast agents, molecular labels and stainings, as well as image acquisition parameters (e.g. MR TE times, CT tube voltage and current).
reconstruction	Describe how images were reconstructed from the acquired data, e.g. (iterative) reconstruction algorithm parameters and voxel dimensions.
approach	Describe whether the image volume is analysed slice-by-slice (2D), or as a volume (3D).
process workflow	Describe the sequence of image processing steps used for the analysis, preferably schematically.
software	Describe which software (if any) or in-house code was used to perform image processing and feature extraction.
data availability	Describe whether and where (image) data, ROIs and software code is made available.
Data conversion	

continued on next page

procedure	Describe how data was converted from reconstructed image data, e.g. by calculation of standard uptake values for PET. References to algorithms should be provided, or otherwise described in detail.
Image post-acquisition processing	
procedure	Describe post-processing steps and parameters, e.g. field-of-view illumination correction in microscopy, noise correction in MRI and partial volume effect correction in PET. References to algorithms should be provided, or otherwise described in detail.
Segmentation	
ROI	Describe which regions of interest were delineated and used for analysis, e.g. specific organs, specific cells, primary tumours.
procedure	Describe how regions of interest were delineated in the image. Specify if segmentation was performed manually, semi-automated or fully automated, by how many users/experts, and how consensus was formed (if applicable). Algorithms and settings used should be listed.
Interpolation	
voxel dimensions	Specify original and interpolated voxel dimensions.
image interpolation method	Specify interpolation method used, e.g. *trilinear interpolation*. Specify how original and interpolation grids were aligned.
image intensity rounding	Specify rounding procedures for non-integer interpolated grey levels (if applicable).
ROI interpolation method	Specify interpolation method used to interpolate ROI morphological and intensity masks.
ROI partial volume	Specify minimum partial volume fraction required to include an interpolated ROI mask voxel in the interpolated ROI (if applicable).
Re-segmentation	
ROI mask criteria	Specify which criteria were used to include or exclude voxels from an ROI mask (if applicable).
Discretisation	
discretisation method	Specify which method is being used for discretising image intensities. References to the method should be provided, or otherwise described in detail.
discretisation parameters	Specify discretisation parameters for discretisation, e.g. number of bins or bin width.
Feature calculation	
feature set	Report which features were calculated. References to feature definitions should be provided, or otherwise described in detail.
feature parameters	Describe specific settings used for feature calculation, e.g. the distance setting and matrix merging method for the grey level co-occurrence matrix.

continued on next page

standardisation	State whether image processing and feature calculation were tested on the digital phantom and the radiomics phantom data. State whether the feature values calculated match the IBSI benchmark values; otherwise state which ones do not match benchmark values, and indicate why.

Table 4.1 — Guidelines for reporting on image biomarker extraction.

4.2 Feature nomenclature

Image features may be extracted using a variety of different settings, and may even share the same name. A feature nomenclature is thus required. Let us take the example of differentiating the following features: *i*) intensity histogram-based entropy, discretised using a *fixed bin size* algorithm with 25 HU bins, extracted from a CT image; and *ii*) grey level run length matrix entropy, discretised using a *fixed bin number* algorithm with 32 bins, extracted from a PET image. To refer to both as *entropy* would be ambiguous, whereas to add a full textual description would be cumbersome. In the nomenclature proposed below, the features would be called $\mathit{entropy}_{\mathit{IH,\ CT,\ FBS:25HU}}$ and $\mathit{entropy}_{\mathit{RLM,\ PET,\ FBN:32}}$, respectively.

Features are thus indicated by a feature name and a subscript. As the nomenclature is designed to both concise and complete, only details for which ambiguity may exist are to be explicitly incorporated in the subscript. The subscript of a feature name may contain the following items to address ambiguous naming:

1. An abbreviation of the feature family (required).
2. The aggregation method of a feature (optional).
3. A descriptor describing the modality the feature is based on, the specific channel (for microscopy images), the specific imaging data (in the case of repeat imaging or delta-features) sets, conversions (such as SUV and SUL), and/or the specific ROI. For example, one could write *PET:SUV* to separate it from *CT* and *PET:SUL* features (optional).
4. Spatial filters and settings (optional).
5. The interpolation algorithm and uniform interpolation grid spacing (optional).
6. The re-segmentation range and outlier filtering (optional).
7. The discretisation method and relevant discretisation parameters, i.e. number of bins or bin size (optional).
8. Feature specific parameters, such as distance for some texture features (optional).

Optional descriptors are only added to the subscript if there are multiple possibilities. For example, if only CT data is used, adding the modality to the subscript is not required. Nonetheless, such details must be reported as well (see section 4.1).

The sections below have tables with permanent IBSI identifiers for concepts that were defined within this document.

4.2.1 Abbreviating feature families

The following is a list of the feature families in the document and their suggested abbreviations:

feature family	abbreviation	
morphology	MORPH	HCUG
local intensity	LI	9ST6
intensity-based statistics	IS, STAT	UHIW
intensity histogram	IH	ZVCW
intensity-volume histogram	IVH	P88C
grey level co-occurrence matrix	GLCM, CM	LFYI
grey level run length matrix	GLRLM, RLM	TP0I
grey level size zone matrix	GLSZM, SZM	9SAK
grey level distance zone matrix	GLDZM, DZM	VMDZ
neighbourhood grey tone difference matrix	NGTDM	IPET
neighbouring grey level dependence matrix	NGLDM	REK0

4.2.2 Abbreviating feature aggregation

The following is a list of feature families and the possible aggregation methods:

morphology, LI		
–	features are 3D by definition	DHQ4
IS, IH, IVH		
2D	averaged over slices (rare)	3IDG
–, 3D	calculated over the volume (default)	DHQ4
GLCM, GLRLM		
2D:avg	averaged over slices and directions	BTW3
2D:mrg, 2D:smrg	merged directions per slice and averaged	SUJT
2.5D:avg, 2.5D:dmrg	merged per direction and averaged	JJUI
2.5D:mrg, 2.5D:vmrg	merged over all slices	ZW7Z
3D:avg	averaged over 3D directions	ITBB
3D:mrg	merged 3D directions	IAZD
GLSZM, GLDZM, NGTDM, NGLDM		
2D	averaged over slices	8QNN
2.5D	merged over all slices	62GR
3D	calculated from single 3D matrix	K0B0

In the list above, ’–’ signifies an empty entry which does not need to be added to the subscript. The following examples highlight the nomenclature used above:

- joint maximum$_{\text{CM, 2D:avg}}$: GLCM-based *joint maximum* feature, calculated by averaging the feature for every in-slice GLCM.

- short runs emphasis$_{\text{RLM, 3D:mrg}}$: RLM-based *short runs emphasis* feature, calculated from an RLM that was aggregated by merging the RLM of each 3D direction.
- mean$_{\text{IS}}$: intensity statistical *mean* feature, calculated over the 3D ROI volume.
- grey level variance$_{\text{SZM, 2D}}$: SZM-based *grey level variance* feature, calculated by averaging the feature value from the SZM in each slice over all the slices.

4.2.3 Abbreviating interpolation

The following is a list of interpolation methods and the suggested notation. Note that # is the interpolation spacing, including units, and *dim* is 2D for interpolation with the slice plane and 3D for volumetric interpolation.

interpolation method	**notation**
none	INT:–
nearest neighbour interpolation	NNB:*dim*:#
linear interpolation	LIN:*dim*:#
cubic convolution interpolation	CCI:*dim*:#
cubic spline interpolation	CSI:*dim*:#, SI3:*dim*:#

The dimension attribute and interpolation spacing may be omitted if this is clear from the context. The following examples highlight the nomenclature introduced above:

- mean$_{\text{IS, LIN:2D:2mm}}$: intensity statistical *mean* feature, calculated after *bilinear* interpolation with the slice planes to uniform voxel sizes of 2mm.
- mean$_{\text{IH, NNB:3D:1mm}}$: intensity histogram *mean* feature, calculated after *trilinear* interpolation to uniform voxel sizes of 1mm.
- joint maximum$_{\text{CM, 2D:mrg, CSI:2D:2mm}}$: GLCM-based *joint maximum* feature, calculated by first merging all GLCM within a slice to single GLCM, calculating the feature and then averaging the feature values over the slices. GLCMs were determined in the image interpolated within the slice plane to 2 × 2mm voxels using *cubic spline* interpolation.

4.2.4 Describing re-segmentation

Re-segmentation can be noted as follows:

re-segmentation method	**notation**	
none	RS:–	
range	RS:[#,#]	`USB3`
outlier filtering	RS:#σ	`7ACA`

In the table above # signify numbers. A re-segmentation range can be half-open, i.e. RS:[#,∞). Re-segmentation methods may be combined, i.e. both range and outlier filtering methods may be used. This is noted as RS:[#,#]+#σ or RS:#σ+[#,#]. The following are examples of the application of the above notation:

- mean$_{\text{IS, CT, RS:[-200,150]}}$: intensity statistical *mean* feature, based on an ROI in a CT image that was re-segmented within a [-200,150] HU range.
- mean$_{\text{IS, PET:SUV, RS:}[3,\infty)}$: intensity statistical *mean* feature, based on an ROI in a PET image with SUV values, that was re-segmented to contain only SUV of 3 and above.
- mean$_{\text{IS, MRI:T1, RS:}3\sigma}$: intensity statistical *mean* feature, based on an ROI in a T1-weighted MR image where the ROI was re-segmented by removing voxels with an intensity outside a $\mu \pm 3\sigma$ range.

4.2.5 Abbreviating discretisation

The following is a list of discretisation methods and the suggested notation. Note that # is the value of the relevant discretisation parameter, e.g. number of bins or bin size, including units.

discretisation method	**notation**	
none	DIS:–	
fixed bin size	FBS:#	Q3RU
fixed bin number	FBN:#	K15C
histogram equalisation	EQ:#	
Lloyd-Max, minimum mean squared	LM:#, MMS:#	

In the table above, # signify numbers such as the number of bins or their width. Histogram equalisation of the ROI intensities can be performed before the "none", "fixed bin size", "fixed bin number" or "Lloyd-Max, minimum mean squared" algorithms defined above, with # specifying the number of bins in the histogram to be equalised. The following are examples of the application of the above notation:

- mean$_{\text{IH,PET:SUV,RS}[0,\infty]\text{,FBS:0.2}}$: intensity histogram *mean* feature, based on an ROI in a SUV-PET image, with bin-width of 0.2 SUV, and binning from 0.0 SUV.
- grey level variance$_{\text{SZM,MR:T1,RS:}3\sigma\text{,FBN:64}}$: size zone matrix-based *grey level variance* feature, based on an ROI in a T1-weighted MR image, with 3σ re-segmentation and subsequent binning into 64 bins.

4.2.6 Abbreviating feature-specific parameters

Some features and feature families require additional parameters, which may be varied. These are the following:

grey level co-occurrence matrix		
co-occurrence matrix symmetry		
–, SYM	symmetrical co-occurrence matrices	
ASYM	asymmetrical co-occurrence matrices (not recommended)	
distance		
δ:#, δ-∞:#	Chebyshev (ℓ_∞) norm with distance # (default)	PVMT

continued on next page

δ-2:#	Euclidean (ℓ_2) norm with distance #	G9EV
δ-1:#	Manhattan (ℓ_1) norm with distance #	LIFZ
distance weighting		
–, w:1	no weighting (default)	
w:f	weighting with function f	
grey level run length matrix		
distance weighting		
–, w:1	no weighting (default)	
w:f	weighting with function f	
grey level size zone matrix		
linkage distance		
δ:#, δ-∞:#	Chebyshev (ℓ_∞) norm with distance (default) #	PVMT
δ-2:#	Euclidean (ℓ_2) norm with distance #	G9EV
δ-1:#	Manhattan (ℓ_1) norm with distance #	LIFZ
grey level distance zone matrix		
linkage distance		
δ:#, δ-∞:#	Chebyshev (ℓ_∞) norm with distance (default) #	PVMT
δ-2:#	Euclidean (ℓ_2) norm with distance #	G9EV
δ-1:#	Manhattan (ℓ_1) norm with distance #	LIFZ
zone distance norm		
–, l-∞:#	Chebyshev (ℓ_∞) norm	PVMT
l-2:#	Euclidean (ℓ_2) norm	G9EV
l-1:#	Manhattan (ℓ_1) norm (default)	LIFZ
neighbourhood grey tone difference matrix		
distance		
δ:#, δ-∞:#	Chebyshev (ℓ_∞) norm with distance # (default)	PVMT
δ-2:#	Euclidean (ℓ_2) norm with distance #	G9EV
δ-1:#	Manhattan (ℓ_1) norm with distance #	LIFZ
distance weighting		
–, w:1	no weighting (default)	
w:f	weighting with function f	
neighbouring grey level dependence matrix		
dependence coarseness		
α:#	dependence coarseness parameter with value #	
distance		
δ:#, δ-∞:#	Chebyshev (ℓ_∞) norm with distance # (default)	PVMT

continued on next page

δ-2:#	Euclidean (ℓ_2) norm with distance #	G9EV
δ-1:#	Manhattan (ℓ_1) norm with distance #	LIFZ
distance weighting		
–, w:1	no weighting (default)	
w:f	weighting with function f	

In the above table, # represents numbers.

Chapter 5

Benchmark data sets

Image features and image processing were benchmarked using a digital image phantom and the CT image of a lung cancer patient, which are described below.

5.1 Digital phantom

A small digital phantom was developed to compare image features. The phantom is shown in figure 5.1. The phantom has the following characteristics:

- The phantom consists of $5 \times 4 \times 4$ (x, y, z) voxels.
- A slice consists of the voxels in (x, y) plane for a particular slice at position z. Therefore slices are stacked in the z direction.
- Voxels are $2.0 \times 2.0 \times 2.0$ mm in size.
- Not all voxels are included in the region of interest. Several excluded voxels are located on the outside of the ROI, and one internal voxel was excluded as well. Voxels excluded from the ROI are shown in blue in figure 5.1.
- Some intensities are not present in the phantom. Notably, grey levels 2 and 5 are absent. 1 is the lowest grey level present in the ROI, and 6 the highest.

5.1.1 Calculating image features

The digital phantom does not require the additional image processing that conventional images require before feature calculation. Thus, feature calculation is done directly on the phantom itself. The following should be taken into account for calculating image features:

- Discretisation is not required. All features are to be calculated using the phantom as it is. Alternatively, one could use a *fixed bin size* discretisation of 1 or *fixed bin number* discretisation of 6 bins, which does not alter the contents of the phantom.
- Grey level co-occurrence matrices are symmetrical and calculated for (Chebyshev) distance $\delta = 1$.
- Neighbouring grey level dependence and neighbourhood grey tone difference matrices are likewise calculated for (Chebyshev) distance $\delta = 1$. Additionally, the neighbouring grey level dependence coarseness parameter has the value $\alpha = 0$.

- Because discretisation is lacking, most intensity-based statistical features will match their intensity histogram-based analogues in value.
- The ROI morphological and intensity masks are identical for the digital phantom, due to lack of re-segmentation.

5.2 Radiomics phantom data

A small dataset of CT images from four non-small-cell lung carcinoma patients was made publicly available to serve as radiomics phantoms (DOI:10.17195/candat.2016.08.1). We use the image for the first patient (`PAT1`) to provide benchmarks for different image processing steps.

The radiomics phantom data is stored as a stack of slices in `DICOM` format. The image slices can be identified by the `DCM_IMG` prefix. The gross tumour volume (GTV) was delineated and is used as the region of interest (ROI). Contour information is stored as an RT structure set in the DICOM file starting with `DCM_RS`. For broader use, both the `DICOM` set and segmentation mask have been converted to the `NifTI` format. When using the data in `NifTI` format, both image stacks should be converted to (at least) 32-bit floating point and rounded to the nearest integer before further processing.

Five image processing configurations are defined to test different image processing algorithms, see Table 5.1. While most settings are self-explanatory, there are several aspects that require some attention. Configurations are divided in 2D and 3D approaches. For the 2D configurations (A, B), image interpolation is conducted within the slice, and likewise texture features are extracted from the in-slice plane, and not volumetrically (3D). For the 3D configurations (C-E) interpolation is conducted in three dimensions, and features are likewise extracted volumetrically. Discretisation is moreover required for texture, intensity histogram and intensity-volume histogram features, and both *fixed bin number* and *fixed bin size* algorithms are tested.

5.2.1 Notes on interpolation

Interpolation has a major influence on feature values. Different algorithm implementations of the same interpolation method may ostensibly provide the same functionality, but lead to different interpolation grids. It is therefore recommended to read the documentation of the particular implementation to assess if the implementation allows or implements the following guidelines:

- The spatial origin of the original grid in world coordinates matches the `DICOM` origin by definition.
- The size of the interpolation grid is determined by rounding the fractional grid size towards infinity, i.e. a ceiling operation. This prevents the interpolation grid from disappearing for very small images, but is otherwise an arbitrary choice.
- The centers of the interpolation and original image grids should be identical, i.e. the interpolation grid is centered on the center of the original image grid. This prevents spacing inconsistencies in the interpolation grid and avoids potential indexing issues.
- The extent of the interpolation grid is, by definition, always equal or larger than that of the original grid. This means that intensities at the grid boundary are extrapolated. To facilitate this process, the image should be sufficiently padded with voxels that take on the nearest boundary intensity.

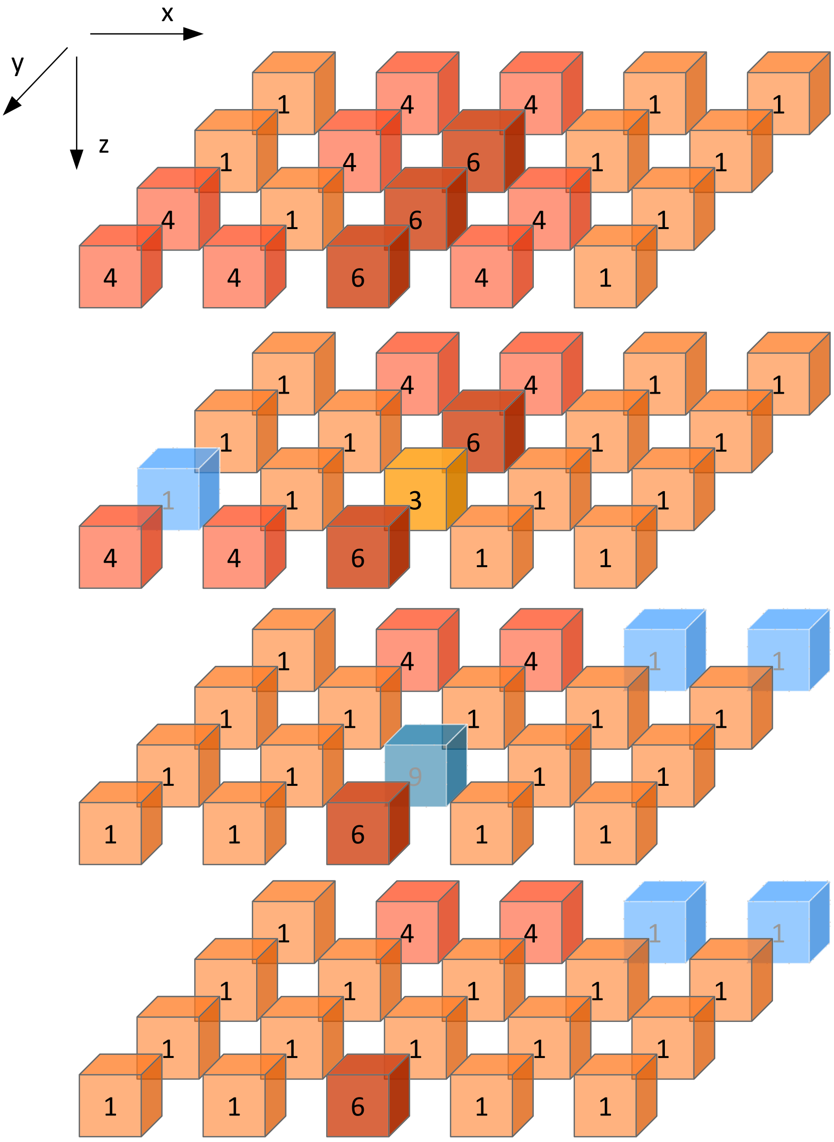


Figure 5.1 — Exploded view of the test volume. The number in each voxel corresponds with its grey level. Blue voxels are excluded from the region of interest. The coordinate system is so that x increases from left to right, y increases from back to front and z increases from top to bottom, as is indicated by the axis definition in the top-left.

- The floating point representation of the image and the ROI masks affects interpolation precision, and consequentially feature values. Image and ROI masks should at least be represented at full precision (`32-bit`) to avoid rounding errors. One example is the unintended exclusion of voxels from the interpolated ROI mask, which occurs when interpolation yields 0.4999... instead of 0.5. When images and ROI masks are converted from lower precision (e.g. `16-bit`), they may require rounding if the original data were integer values, such as Hounsfield Units or the ROI mask labels.

More details are provided in Section 2.4.

5.2.2 Diagnostic features

Identifying issues with an implementation of the image processing sequence may be challenging. Multiple steps follow one another and differences propagate. Hence we define a small number of diagnostic features that describe how the image and ROI masks change with each image processing step.

Benchmark values of diagnostic features for the five different configurations are shown in chapter B of the appendix.

Initial image stack. The following features may be used to describe the initial image stack (i.e. after loading image data for processing):

- *Image dimensions.* This describes the image dimensions in voxels along the different image axes.
- *Voxel dimensions.* This describes the voxel dimensions in mm. The dimension along the z-axis is equal to the distance between the origin voxels of two adjacent slices, and is generally equal to the slice thickness.
- *Mean intensity.* This is the average intensity within the entire image.
- *Minimum intensity.* This is the lowest intensity within the entire image.
- *Maximum intensity.* This is the highest intensity within the entire image.

Interpolated image stack. The above features may also be used to describe the image stack after image interpolation.

Initial region of interest. The following descriptors are used to describe the region of interest (ROI) directly after segmentation of the image:

- *ROI intensity mask dimensions.* This describes the dimensions, in voxels, of the ROI intensity mask.
- *ROI intensity mask bounding box dimensions.* This describes the dimensions, in voxels, of the bounding box of the ROI intensity mask.
- *ROI morphological mask bounding box dimensions.* This describes the dimensions, in voxels, of the bounding box of the ROI morphological mask.
- *Number of voxels in the ROI intensity mask.* This describes the number of voxels included in the ROI intensity mask.

- *Number of voxels in the ROI morphological mask.* This describes the number of voxels included in the ROI intensity mask.
- *Mean ROI intensity.* This is the mean intensity of image voxels within the ROI intensity mask.
- *Minimum ROI intensity.* This is the lowest intensity of image voxels within the ROI intensity mask.
- *Maximum ROI intensity.* This is the highest intensity of image voxels within the ROI intensity mask.

Interpolated region of interest. The same features can be used to describe the ROI after interpolation of the ROI mask.

Re-segmented region of interest. Again, the same features as above can be used to describe the ROI after re-segmentation.

5.2.3 Calculating image features

Unlike the digital phantom, the radiomics phantom does require additional image processing, which is done according to the processing configurations described in Table 5.1. The following should be taken into account when calculating image features:

- Grey level co-occurrence matrices are symmetrical and calculated for (Chebyshev) distance $\delta = 1$.
- Neighbouring grey level dependence and neighbourhood grey tone difference matrices are likewise calculated for (Chebyshev) distance $\delta = 1$. Additionally, the neighbouring grey level dependence coarseness parameter $\alpha = 0$.
- Intensity-based statistical features and their intensity histogram-based analogues will differ in value due to discretisation, in contrast to the same features for the digital phantom.
- Due to re-segmentation, the ROI morphological and intensity masks are not identical.
- Calculation of IVH feature: since by default CT contains definite and discrete intensities, no separate discretisation prior to the calculation of intensity-volume histogram features is required. This is the case for configurations A, B and D (i.e. 'definite intensity units – discrete case'). However, for configurations C and E, we re-discretise the ROI intensities prior to calculation of intensity-volume histogram features to provide better verification of processing algorithms. Configuration C simulates the 'definite intensity units – continuous case', while configuration E simulates the 'arbitrary intensity units' case where re-segmentation range is not used. For details, please consult section 3.5.

	config. A	config. B	config. C	config. D	config. E
Sample	`PAT1`	`PAT1`	`PAT1`	`PAT1`	`PAT1`
ROI	`GTV-1`	`GTV-1`	`GTV-1`	`GTV-1`	`GTV-1`
Approach	2D	2D	3D	3D	3D
Interpolation	no	yes	yes	yes	yes
Voxel dimension (mm)		2×2	$2 \times 2 \times 2$	$2 \times 2 \times 2$	$2 \times 2 \times 2$
Interpolation method		bilinear	trilinear	trilinear	tricubic spline
Grey level rounding		nearest integer	nearest integer	nearest integer	nearest integer
ROI interp. method		bilinear	trilinear	trilinear	trilinear
ROI partial volume		0.5	0.5	0.5	0.5
Re-segmentation					
Range (HU)	$[-500, 400]$	$[-500, 400]$	$[-1000, 400]$	no	$[-1000, 400]$
Outliers	no	no	no	3σ	3σ
Discretisation					
Texture and IH	FBS: 25 HU	FBN: 32 bins	FBS: 25 HU	FBN: 32 bins	FBN: 32 bins
IVH	no	no	FBS: 2.5 HU	no	FBN: 1000 bins

Table 5.1 — Different configurations for image processing. For details, refer to the corresponding sections in chapter 2. ROI: region of interest; HU: Hounsfield Unit; IH: intensity histogram; FBS: fixed bin size; FBN: fixed bin number; IVH: intensity-volume histogram; NA: not applicable.

Chapter 6

Benchmarks

This chapter presents the feature benchmark values for the digital phantom and radiomics phantom. Features based on texture matrices determined by slice and then fully merged (Figures 3.3c and 3.4b) were not benchmarked, as this approach was not used. The list of benchmark values is also available as a separate *.csv* table.

A tolerance was determined for the benchmark values in the radiomics phantom, as minor differences introduced during image processing may lead to different feature values. For this purpose the image data and the mask were rotated (from $-15°$ to $15°$ in $5°$ steps) and translated (0.0, 0.25, 0.50 and 0.75 times the voxel spacing) in the xy-plane, and the ROI mask was eroded (2mm), kept the same, and dilated (2mm). This lead to 336 values for a single feature. The tolerance shown in the tables in this chapter is equal 5% of the interquartile range of these values.

Additionally, it should be noted that all benchmarks are values actually produced by teams involved in the IBSI, rather than averages. All contributed values were rounded to 3 significant digits before being processed and analysed to determine consensus.

Benchmark results are removed until stronger consensus has been established.

Appendix A

Digital phantom texture matrices

This section contains the texture matrices extracted from the digital phantom for reference and benchmark purposes.

A.1 Grey level co-occurrence matrix (2D)

i	j	n
1.0	1.0	10
1.0	4.0	4
4.0	1.0	4
4.0	4.0	6
4.0	6.0	1
6.0	4.0	1
6.0	6.0	4

(a) **x**: (0,1,0)
slice: 1 of 4

i	j	n
1.0	1.0	16
1.0	4.0	2
3.0	6.0	2
4.0	1.0	2
4.0	6.0	1
6.0	3.0	2
6.0	4.0	1

(b) **x**: (0,1,0)
slice: 2 of 4

i	j	n
1.0	1.0	18
1.0	4.0	2
4.0	1.0	2

(c) **x**: (0,1,0)
slice: 3 of 4

i	j	n
1.0	1.0	20
1.0	4.0	2
1.0	6.0	1
4.0	1.0	2
6.0	1.0	1

(d) **x**: (0,1,0)
slice: 4 of 4

i	j	n
1.0	1.0	2
1.0	4.0	4
1.0	6.0	3
4.0	1.0	4
4.0	4.0	4
4.0	6.0	2
6.0	1.0	3
6.0	4.0	2

(e) **x**: (1,-1,0)
slice: 1 of 4

i	j	n
1.0	1.0	6
1.0	3.0	1
1.0	4.0	3
1.0	6.0	3
3.0	1.0	1
3.0	4.0	1
4.0	1.0	3
4.0	3.0	1
6.0	1.0	3

(f) **x**: (1,-1,0)
slice: 2 of 4

i	j	n
1.0	1.0	10
1.0	4.0	2
1.0	6.0	1
4.0	1.0	2
6.0	1.0	1

(g) **x**: (1,-1,0)
slice: 3 of 4

i	j	n
1.0	1.0	14
1.0	4.0	2
1.0	6.0	1
4.0	1.0	2
6.0	1.0	1

(h) **x**: (1,-1,0)
slice: 4 of 4

i	j	n
1.0	1.0	4
1.0	4.0	6
1.0	6.0	2
4.0	1.0	6
4.0	4.0	4
4.0	6.0	4
6.0	1.0	2
6.0	4.0	4

(i) d: (1,0,0)
slice: 1 of 4

i	j	n
1.0	1.0	10
1.0	3.0	2
1.0	4.0	2
1.0	6.0	3
3.0	1.0	2
4.0	1.0	2
4.0	4.0	4
4.0	6.0	1
6.0	1.0	3
6.0	4.0	1

(j) d: (1,0,0)
slice: 2 of 4

i	j	n
1.0	1.0	16
1.0	4.0	1
1.0	6.0	2
4.0	1.0	1
4.0	4.0	2
6.0	1.0	2

(k) d: (1,0,0)
slice: 3 of 4

i	j	n
1.0	1.0	20
1.0	4.0	1
1.0	6.0	2
4.0	1.0	1
4.0	4.0	2
6.0	1.0	2

(l) d: (1,0,0)
slice: 4 of 4

i	j	n
1.0	1.0	6
1.0	4.0	3
1.0	6.0	1
4.0	1.0	3
4.0	4.0	2
4.0	6.0	4
6.0	1.0	1
6.0	4.0	4

(m) d: (1,1,0)
slice: 1 of 4

i	j	n
1.0	1.0	10
1.0	3.0	2
1.0	4.0	1
1.0	6.0	2
3.0	1.0	2
4.0	1.0	1
4.0	6.0	1
6.0	1.0	2
6.0	4.0	1

(n) d: (1,1,0)
slice: 2 of 4

i	j	n
1.0	1.0	12
1.0	4.0	2
1.0	6.0	1
4.0	1.0	2
6.0	1.0	1

(o) d: (1,1,0)
slice: 3 of 4

i	j	n
1.0	1.0	16
1.0	4.0	2
1.0	6.0	1
4.0	1.0	2
6.0	1.0	1

(p) d: (1,1,0)
slice: 4 of 4

Table A.1 — Grey-level co-occurrence matrices extracted from the xy plane (2D) of the digital phantom using Chebyshev distance 1. **x** indicates the direction in (x, y, z) coordinates.

A.2 Grey level co-occurrence matrix (2D, merged)

i	j	n
1.0	1.0	22
1.0	4.0	17
1.0	6.0	6
4.0	1.0	17
4.0	4.0	16
4.0	6.0	11
6.0	1.0	6
6.0	4.0	11
6.0	6.0	4

(a) slice: 1 of 4

i	j	n
1.0	1.0	42
1.0	3.0	5
1.0	4.0	8
1.0	6.0	8
3.0	1.0	5
3.0	4.0	1
3.0	6.0	2
4.0	1.0	8
4.0	3.0	1
4.0	4.0	4
4.0	6.0	3
6.0	1.0	8
6.0	3.0	2
6.0	4.0	3

(b) slice: 2 of 4

i	j	n
1.0	1.0	56
1.0	4.0	7
1.0	6.0	4
4.0	1.0	7
4.0	4.0	2
6.0	1.0	4

(c) slice: 3 of 4

i	j	n
1.0	1.0	70
1.0	4.0	7
1.0	6.0	5
4.0	1.0	7
4.0	4.0	2
6.0	1.0	5

(d) slice: 4 of 4

Table A.2 — Merged grey-level co-occurrence matrices extracted from the xy plane (2D) of the digital phantom using Chebyshev distance 1.

A.3 Grey level co-occurrence matrix (3D)

i	j	n
1.0	1.0	66
1.0	4.0	5
1.0	6.0	1
3.0	6.0	1
4.0	1.0	5
4.0	4.0	16
6.0	1.0	1
6.0	3.0	1
6.0	6.0	8

(a) **x**: (0,0,1)

i	j	n
1.0	1.0	42
1.0	3.0	1
1.0	4.0	9
1.0	6.0	1
3.0	1.0	1
3.0	6.0	1
4.0	1.0	9
4.0	4.0	2
4.0	6.0	2
6.0	1.0	1
6.0	3.0	1
6.0	4.0	2
6.0	6.0	2

(b) **x**: (0,1,-1)

i	j	n
1.0	1.0	64
1.0	4.0	10
1.0	6.0	1
3.0	6.0	2
4.0	1.0	10
4.0	4.0	6
4.0	6.0	2
6.0	1.0	1
6.0	3.0	2
6.0	4.0	2
6.0	6.0	4

(c) **x**: (0,1,0)

i	j	n
1.0	1.0	52
1.0	4.0	8
3.0	6.0	2
4.0	1.0	8
4.0	4.0	2
4.0	6.0	1
6.0	3.0	2
6.0	4.0	1
6.0	6.0	2

(d) **x**: (0,1,1)

i	j	n
1.0	1.0	30
1.0	3.0	2
1.0	4.0	7
1.0	6.0	5
3.0	1.0	2
4.0	1.0	7
4.0	6.0	2
6.0	1.0	5
6.0	4.0	2

(e) **x**: (1,-1,-1)

i	j	n
1.0	1.0	32
1.0	3.0	1
1.0	4.0	11
1.0	6.0	8
3.0	1.0	1
3.0	4.0	1
4.0	1.0	11
4.0	3.0	1
4.0	4.0	4
4.0	6.0	2
6.0	1.0	8
6.0	4.0	2

(f) **x**: (1,-1,0)

i	j	n
1.0	1.0	20
1.0	3.0	1
1.0	4.0	10
1.0	6.0	6
3.0	1.0	1
3.0	4.0	1
4.0	1.0	10
4.0	3.0	1
4.0	4.0	2
6.0	1.0	6

(g) **x**: (1,-1,1)

i	j	n
1.0	1.0	38
1.0	3.0	1
1.0	4.0	7
1.0	6.0	8
3.0	1.0	1
3.0	4.0	1
4.0	1.0	7
4.0	3.0	1
4.0	4.0	8
4.0	6.0	2
6.0	1.0	8
6.0	4.0	2

(h) **x**: (1,0,-1)

i	j	n
1.0	1.0	50
1.0	3.0	2
1.0	4.0	10
1.0	6.0	9
3.0	1.0	2
4.0	1.0	10
4.0	4.0	12
4.0	6.0	5
6.0	1.0	9
6.0	4.0	5

(i) **x**: (1,0,0)

i	j	n
1.0	1.0	34
1.0	3.0	2
1.0	4.0	8
1.0	6.0	7
3.0	1.0	2
4.0	1.0	8
4.0	4.0	8
4.0	6.0	3
6.0	1.0	7
6.0	4.0	3

(j) **x**: (1,0,1)

i	j	n
1.0	1.0	32
1.0	3.0	1
1.0	4.0	6
1.0	6.0	4
3.0	1.0	1
3.0	4.0	1
4.0	1.0	6
4.0	3.0	1
4.0	6.0	3
6.0	1.0	4
6.0	4.0	3

(k) **x**: (1,1,-1)

i	j	n
1.0	1.0	44
1.0	3.0	2
1.0	4.0	8
1.0	6.0	5
3.0	1.0	2
4.0	1.0	8
4.0	4.0	2
4.0	6.0	5
6.0	1.0	5
6.0	4.0	5

(l) **x**: (1,1,0)

i	j	n
1.0	1.0	32
1.0	3.0	1
1.0	4.0	6
1.0	6.0	6
3.0	1.0	1
3.0	4.0	1
4.0	1.0	6
4.0	3.0	1
4.0	4.0	2
4.0	6.0	1
6.0	1.0	6
6.0	4.0	1

(m) **x**: (1,1,1)

Table A.3 — Grey-level co-occurrence matrices extracted volumetrically (3D) from the digital phantom using Chebyshev distance 1. **x** indicates the direction in (x, y, z) coordinates.

A.4 Grey level co-occurrence matrix (3D, merged)

i	j	n
1.0	1.0	536
1.0	3.0	14
1.0	4.0	105
1.0	6.0	61
3.0	1.0	14
3.0	4.0	5
3.0	6.0	6
4.0	1.0	105
4.0	3.0	5
4.0	4.0	64
4.0	6.0	28
6.0	1.0	61
6.0	3.0	6
6.0	4.0	28
6.0	6.0	16

Table A.4 — Merged grey-level co-occurrence matrix extracted volumetrically (3D) from the digital phantom using Chebyshev distance 1.

A.5 Grey level run length matrix (2D)

i	r	n
1.0	1.0	1.0
1.0	2.0	2.0
1.0	4.0	1.0
4.0	1.0	2.0
4.0	2.0	3.0
6.0	3.0	1.0

(a) **x**: (0,1,0)
slice: 1 of 4

i	r	n
1.0	2.0	2.0
1.0	4.0	2.0
3.0	1.0	1.0
4.0	1.0	4.0
6.0	1.0	2.0

(b) **x**: (0,1,0)
slice: 2 of 4

i	r	n
1.0	1.0	1.0
1.0	3.0	3.0
1.0	4.0	1.0
4.0	1.0	2.0
6.0	1.0	1.0

(c) **x**: (0,1,0)
slice: 3 of 4

i	r	n
1.0	2.0	1.0
1.0	3.0	3.0
1.0	4.0	1.0
4.0	1.0	2.0
6.0	1.0	1.0

(d) **x**: (0,1,0)
slice: 4 of 4

i	r	n
1.0	1.0	7.0
1.0	2.0	1.0
4.0	1.0	5.0
4.0	3.0	1.0
6.0	1.0	3.0

(e) **x**: (1,-1,0)
slice: 1 of 4

i	r	n
1.0	1.0	6.0
1.0	2.0	3.0
3.0	1.0	1.0
4.0	1.0	4.0
6.0	1.0	2.0

(f) **x**: (1,-1,0)
slice: 2 of 4

i	r	n
1.0	1.0	5.0
1.0	2.0	3.0
1.0	3.0	1.0
4.0	1.0	2.0
6.0	1.0	1.0

(g) **x**: (1,-1,0)
slice: 3 of 4

i	r	n
1.0	1.0	3.0
1.0	2.0	3.0
1.0	3.0	2.0
4.0	1.0	2.0
6.0	1.0	1.0

(h) **x**: (1,-1,0)
slice: 4 of 4

i	r	n
1.0	1.0	5.0
1.0	2.0	2.0
4.0	1.0	4.0
4.0	2.0	2.0
6.0	1.0	3.0

(i) **x**: (1,0,0)
slice: 1 of 4

i	r	n
1.0	1.0	2.0
1.0	2.0	5.0
3.0	1.0	1.0
4.0	2.0	2.0
6.0	1.0	2.0

(j) **x**: (1,0,0)
slice: 2 of 4

i	r	n
1.0	1.0	1.0
1.0	2.0	4.0
1.0	5.0	1.0
4.0	2.0	1.0
6.0	1.0	1.0

(k) **x**: (1,0,0)
slice: 3 of 4

i	r	n
1.0	1.0	1.0
1.0	2.0	2.0
1.0	5.0	2.0
4.0	2.0	1.0
6.0	1.0	1.0

(l) **x**: (1,0,0)
slice: 4 of 4

i	r	n
1.0	1.0	3.0
1.0	2.0	3.0
4.0	1.0	6.0
4.0	2.0	1.0
6.0	1.0	3.0

(m) **x**: (1,1,0)
slice: 1 of 4

i	r	n
1.0	1.0	2.0
1.0	2.0	5.0
3.0	1.0	1.0
4.0	1.0	4.0
6.0	1.0	2.0

(n) **x**: (1,1,0)
slice: 2 of 4

i	r	n
1.0	1.0	3.0
1.0	2.0	4.0
1.0	3.0	1.0
4.0	1.0	2.0
6.0	1.0	1.0

(o) **x**: (1,1,0)
slice: 3 of 4

i	r	n
1.0	1.0	2.0
1.0	2.0	3.0
1.0	3.0	1.0
1.0	4.0	1.0
4.0	1.0	2.0
6.0	1.0	1.0

(p) **x**: (1,1,0)
slice: 4 of 4

Table A.5 — Grey-level run length matrices extracted from the xy plane (2D) of the digital phantom. **x** indicates the direction in (x, y, z) coordinates.

A.6 Grey level run length matrix (2D, merged)

i	r	n
1.0	1.0	16.0
1.0	2.0	8.0
1.0	4.0	1.0
4.0	1.0	17.0
4.0	2.0	6.0
4.0	3.0	1.0
6.0	1.0	9.0
6.0	3.0	1.0

(a) slice: 1 of 4

i	r	n
1.0	1.0	10.0
1.0	2.0	15.0
1.0	4.0	2.0
3.0	1.0	4.0
4.0	1.0	12.0
4.0	2.0	2.0
6.0	1.0	8.0

(b) slice: 2 of 4

i	r	n
1.0	1.0	10.0
1.0	2.0	11.0
1.0	3.0	5.0
1.0	4.0	1.0
1.0	5.0	1.0
4.0	1.0	6.0
4.0	2.0	1.0
6.0	1.0	4.0

(c) slice: 3 of 4

i	r	n
1.0	1.0	6.0
1.0	2.0	9.0
1.0	3.0	6.0
1.0	4.0	2.0
1.0	5.0	2.0
4.0	1.0	6.0
4.0	2.0	1.0
6.0	1.0	4.0

(d) slice: 4 of 4

Table A.6 — Merged grey-level run length matrices extracted from the xy plane (2D) of the digital phantom.

A.7 Grey level run length matrix (3D)

i	r	n
1.0	1.0	1.0
1.0	2.0	6.0
1.0	3.0	3.0
1.0	4.0	7.0
3.0	1.0	1.0
4.0	1.0	4.0
4.0	2.0	2.0
4.0	4.0	2.0
6.0	1.0	1.0
6.0	2.0	1.0
6.0	4.0	1.0

(a) **x**: (0,0,1)

i	r	n
1.0	1.0	11.0
1.0	2.0	15.0
1.0	3.0	3.0
3.0	1.0	1.0
4.0	1.0	14.0
4.0	2.0	1.0
6.0	1.0	5.0
6.0	2.0	1.0

(b) **x**: (0,1,-1)

i	r	n
1.0	1.0	2.0
1.0	2.0	5.0
1.0	3.0	6.0
1.0	4.0	5.0
3.0	1.0	1.0
4.0	1.0	10.0
4.0	2.0	3.0
6.0	1.0	4.0
6.0	3.0	1.0

(c) **x**: (0,1,0)

i	r	n
1.0	1.0	10.0
1.0	2.0	5.0
1.0	3.0	6.0
1.0	4.0	3.0
3.0	1.0	1.0
4.0	1.0	14.0
4.0	2.0	1.0
6.0	1.0	5.0
6.0	2.0	1.0

(d) **x**: (0,1,1)

i	r	n
1.0	1.0	22.0
1.0	2.0	11.0
1.0	3.0	2.0
3.0	1.0	1.0
4.0	1.0	16.0
6.0	1.0	7.0

(e) **x**: (1,-1,-1)

i	r	n
1.0	1.0	21.0
1.0	2.0	10.0
1.0	3.0	3.0
3.0	1.0	1.0
4.0	1.0	13.0
4.0	3.0	1.0
6.0	1.0	7.0

(f) **x**: (1,-1,0)

i	r	n
1.0	1.0	30.0
1.0	2.0	10.0
3.0	1.0	1.0
4.0	1.0	14.0
4.0	2.0	1.0
6.0	1.0	7.0

(g) **x**: (1,-1,1)

i	r	n
1.0	1.0	16.0
1.0	2.0	12.0
1.0	3.0	2.0
1.0	4.0	1.0
3.0	1.0	1.0
4.0	1.0	8.0
4.0	2.0	4.0
6.0	1.0	7.0

(h) **x**: (1,0,-1)

i	r	n
1.0	1.0	9.0
1.0	2.0	13.0
1.0	5.0	3.0
3.0	1.0	1.0
4.0	1.0	4.0
4.0	2.0	6.0
6.0	1.0	7.0

(i) **x**: (1,0,0)

i	r	n
1.0	1.0	19.0
1.0	2.0	12.0
1.0	3.0	1.0
1.0	4.0	1.0
3.0	1.0	1.0
4.0	1.0	8.0
4.0	2.0	4.0
6.0	1.0	7.0

(j) **x**: (1,0,1)

i	r	n
1.0	1.0	20.0
1.0	2.0	12.0
1.0	3.0	2.0
3.0	1.0	1.0
4.0	1.0	16.0
6.0	1.0	7.0

(k) **x**: (1,1,-1)

i	r	n
1.0	1.0	10.0
1.0	2.0	15.0
1.0	3.0	2.0
1.0	4.0	1.0
3.0	1.0	1.0
4.0	1.0	14.0
4.0	2.0	1.0
6.0	1.0	7.0

(l) **x**: (1,1,0)

i	r	n
1.0	1.0	19.0
1.0	2.0	14.0
1.0	3.0	1.0
3.0	1.0	1.0
4.0	1.0	14.0
4.0	2.0	1.0
6.0	1.0	7.0

(m) **x**: (1,1,1)

Table A.7 — Grey-level run length matrices extracted volumetrically (3D) from the digital phantom. **x** indicates the direction in (x, y, z) coordinates.

A.8 Grey level run length matrix (3D, merged)

i	r	n
1.0	1.0	190.0
1.0	2.0	140.0
1.0	3.0	31.0
1.0	4.0	18.0
1.0	5.0	3.0
3.0	1.0	13.0
4.0	1.0	149.0
4.0	2.0	24.0
4.0	3.0	1.0
4.0	4.0	2.0
6.0	1.0	78.0
6.0	2.0	3.0
6.0	3.0	1.0
6.0	4.0	1.0

Table A.8 — Merged grey-level run length matrix extracted volumetrically (3D) from the digital phantom.

A.9 Grey level size zone matrix (2D)

i	s	n
1.0	3	1
1.0	6	1
4.0	2	1
4.0	6	1
6.0	3	1

(a) slice: 1 of 4

i	s	n
1.0	4	1
1.0	8	1
3.0	1	1
4.0	2	2
6.0	1	2

(b) slice: 2 of 4

i	s	n
1.0	14	1
4.0	2	1
6.0	1	1

(c) slice: 3 of 4

i	s	n
1.0	15	1
4.0	2	1
6.0	1	1

(d) slice: 4 of 4

Table A.9 — Grey level size zone matrices extracted from the xy plane (2D) of the digital phantom.

A.10 Grey level size zone matrix (3D)

i	s	n
1.0	50	1
3.0	1	1
4.0	2	1
4.0	14	1
6.0	7	1

Table A.10 — Grey level size zone matrix extracted volumetrically (3D) from the digital phantom.

A.11 Grey level distance zone matrix (2D)

i	d	n
1.0	1.0	2
4.0	1.0	2
6.0	1.0	1

(a) slice: 1 of 4

i	d	n
1.0	1.0	2
3.0	2.0	1
4.0	1.0	2
6.0	1.0	1
6.0	2.0	1

(b) slice: 2 of 4

i	d	n
1.0	1.0	1
4.0	1.0	1
6.0	1.0	1

(c) slice: 3 of 4

i	d	n
1.0	1.0	1
4.0	1.0	1
6.0	1.0	1

(d) slice: 4 of 4

Table A.11 — Grey level distance zone matrices extracted from the xy plane (2D) of the digital phantom.

A.12 Grey level distance zone matrix (3D)

i	d	n
1.0	1.0	1
3.0	1.0	1
4.0	1.0	2
6.0	1.0	1

Table A.12 — Grey level distance zone matrix extracted volumetrically (3D) from the digital phantom.

A.13 Neighbourhood grey tone difference matrix (2D)

i	s	n
1.0	14.575	9
4.0	5.775	8
6.0	7.325	3

(a) slice: 1 of 4

i	s	n
1.0	11.928571	12
3.0	0.375000	1
4.0	4.800000	4
6.0	8.000000	2

(b) slice: 2 of 4

i	s	n
1.0	7.985714	14
4.0	4.650000	2
6.0	5.000000	1

(c) slice: 3 of 4

i	s	n
1.0	7.582143	15
4.0	4.650000	2
6.0	5.000000	1

(d) slice: 4 of 4

Table A.13 — Neighbourhood grey tone difference matrices extracted from the xy plane (2D) of the digital phantom using Chebyshev distance 1.

A.14 Neighbourhood grey tone difference matrix (3D)

i	s	n
1.0	39.946954	50
3.0	0.200000	1
4.0	20.825401	16
6.0	24.127005	7

Table A.14 — Neighbourhood grey tone difference matrix extracted volumetrically (3D) from the digital phantom using Chebyshev distance 1.

A.15 Neighbouring grey level dependence matrix (2D)

i	j	s
1.0	2.0	3
1.0	3.0	1
1.0	4.0	3
1.0	5.0	2
4.0	2.0	2
4.0	3.0	4
4.0	4.0	2
6.0	2.0	2
6.0	3.0	1

(a) slice: 1 of 4

i	j	s
1.0	3.0	2
1.0	4.0	6
1.0	6.0	4
3.0	1.0	1
4.0	2.0	4
6.0	1.0	2

(b) slice: 2 of 4

i	j	s
1.0	3.0	1
1.0	4.0	5
1.0	5.0	3
1.0	6.0	3
1.0	7.0	2
4.0	2.0	2
6.0	1.0	1

(c) slice: 3 of 4

i	j	s
1.0	3.0	1
1.0	4.0	3
1.0	5.0	3
1.0	6.0	4
1.0	7.0	1
1.0	8.0	3
4.0	2.0	2
6.0	1.0	1

(d) slice: 4 of 4

Table A.15 — Neighbouring grey level dependence matrices extracted from the xy plane (2D) of the digital phantom using Chebyshev distance 1 and coarseness 0.

A.16 Neighbouring grey level dependence matrix (3D)

i	j	s
1.0	5.0	2
1.0	6.0	2
1.0	7.0	1
1.0	8.0	6
1.0	9.0	4
1.0	10.0	6
1.0	11.0	5
1.0	12.0	5
1.0	13.0	3
1.0	14.0	2
1.0	15.0	5
1.0	16.0	3
1.0	17.0	3
1.0	18.0	2
1.0	21.0	1
3.0	1.0	1
4.0	2.0	2
4.0	4.0	2
4.0	5.0	6
4.0	6.0	4
4.0	7.0	2
6.0	2.0	1
6.0	3.0	4
6.0	4.0	1
6.0	5.0	1

Table A.16 — Neighbouring grey level dependence matrix extracted volumetrically (3D) from the digital phantom using Chebyshev distance 1 and coarseness 0.

Appendix B

Radiomics phantom diagnostic features

Assessing the diagnostic features for the radiomics phantom may assist in identifying issues with regards to image processing. Minor deviations to the values presented in this chapter may occur due to rounding errors, or slightly different algorithmic implementations.

Characteristics of the original image The characteristics of the original image, by definition, are the same for all configurations.

feature	**config. A**	**config. B**	**config. C**	**config. D**	**config. E**
image dimension x	204	204	204	204	204
image dimension y	201	201	201	201	201
image dimension z	60	60	60	60	60
voxel dimension x	0.977	0.977	0.977	0.977	0.977
voxel dimension y	0.977	0.977	0.977	0.977	0.977
voxel dimension z	3	3	3	3	3
mean intensity	-266	-266	-266	-266	-266
minimum intensity	-1000	-1000	-1000	-1000	-1000
maximum intensity	3065	3065	3065	3065	3065

Table B.1 — Values of characteristics of the original image. *Mean intensity* is rounded to the nearest integer.

Characteristics of the interpolated image *Mean*, *minimum* and *maximum* intensity may deviate due to differences in how the original image is padded to determine the value of voxels at the edge of the interpolation grid. As we are dealing with a CT data set, it is also important to round intensities to the nearest integer after interpolation.

feature	config. A	config. B	config. C	config. D	config. E
image dimension x	–	100	100	100	100
image dimension y	–	99	99	99	99
image dimension z	–	60	90	90	90
voxel dimension x	–	2	2	2	2
voxel dimension y	–	2	2	2	2
voxel dimension z	–	3	2	2	2
mean intensity	–	-270	-270	-270	-270
minimum intensity	–	-1000	-1000	-1000	-1111
maximum intensity	–	2257	1854	1854	2637

Table B.2 — Values of characteristics of the interpolated image. *Mean intensity* is rounded to the nearest integer. Note that in configuration A no interpolation takes place.

Characteristics of the original ROI mask Voxel counts of the ROI mask may deviate slightly due to differences in algorithm implementation. Different IBSI participants reported 1 voxel more or less than the numbers provided below.

feature	config. A	config. B	config. C	config. D	config. E
int. mask dimension x	204	204	204	204	204
int. mask dimension y	201	201	201	201	201
int. mask dimension z	60	60	60	60	60
int. mask bounding box dim. x	100	100	100	100	100
int. mask bounding box dim. y	99	99	99	99	99
int. mask bounding box dim. z	26	26	26	26	26
morph. mask bounding box dim. x	100	100	100	100	100
morph. mask bounding box dim. y	99	99	99	99	99
morph. mask bounding box dim. z	26	26	26	26	26
int. mask voxel count	125256	125256	125256	125256	125256
morph. mask voxel count	125256	125256	125256	125256	125256
int. mask mean intensity	-47	-47	-47	-47	-47
int. mask minimum intensity	-1000	-1000	-1000	-1000	-1000
int. mask maximum intensity	723	723	723	723	723

Table B.3 — Values of characteristics of the original ROI mask. Intensity and morphological masks are identical. *Mean intensity* is rounded to the nearest integer.

Characteristics of the interpolated ROI mask Interpolation is a critical step in the image processing scheme. When performing interpolation, it is important to keep the points mentioned in section 5.2.1 in mind. Deviations in the ROI mask voxel count are likely to occur if interpolation is performed differently.

feature	config. A	config. B	config. C	config. D	config. E
int. mask dimension x	–	100	100	100	100
int. mask dimension y	–	99	99	99	99
int. mask dimension z	–	60	90	90	90

continued on next page

feature	config. A	config. B	config. C	config. D	config. E
int. mask bounding box dim. x	–	49	49	49	49
int. mask bounding box dim. y	–	49	49	49	49
int. mask bounding box dim. z	–	26	40	40	40
morph. mask bounding box dim. x	–	49	49	49	49
morph. mask bounding box dim. y	–	49	49	49	49
morph. mask bounding box dim. z	–	26	40	40	40
int. mask voxel count	–	29842	45985	45985	45985
morph. mask voxel count	–	29842	45985	45985	45985
int. mask mean intensity	–	-47	-49	-49	-48
int. mask minimum intensity	–	-956	-939	-939	-966
int. mask maximum intensity	–	525	521	521	627

Table B.4 — Values of characteristics of the interpolated ROI mask. After interpolation, intensity and morphological masks are still identical. *Mean intensity* is rounded to the nearest integer. Note that in configuration A no interpolation takes place.

Characteristics of the re-segmented ROI mask These are the characteristics of the ROI masks within which the features are subsequently calculated.

feature	config. A	config. B	config. C	config. D	config. E
int. mask dimension x	204	100	100	100	100
int. mask dimension y	201	99	99	99	99
int. mask dimension z	60	60	90	90	90
int. mask bounding box dim. x	100	49	49	49	49
int. mask bounding box dim. y	99	49	49	49	49
int. mask bounding box dim. z	26	26	40	40	40
morph. mask bounding box dim. x	100	49	49	49	49
morph. mask bounding box dim. y	99	49	49	49	49
morph. mask bounding box dim. z	26	26	40	40	40
int. mask voxel count	114596	27359	45981	44465	40398
morph. mask voxel count	125256	29842	45985	45985	45985
int. mask mean intensity	13	12	-49	-24	30
int. mask minimum intensity	-500	-500	-939	-724	-345
int. mask maximum intensity	377	391	393	521	345

Table B.5 — Values of characteristics of the re-segmented ROI mask. After re-segmentation, intensity and morphological masks are no longer identical. *Mean intensity* is rounded to the nearest integer.

Bibliography

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

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

[3] F. Albregtsen, B. Nielsen, and H. Danielsen. 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, 2000.

[4] B. A. Altazi, G. G. Zhang, D. C. Fernandez, M. E. Montejo, D. Hunt, J. Werner, M. C. Biagioli, and E. G. Moros. Reproducibility of F18-FDG PET radiomic features for different cervical tumor segmentation methods, gray-level discretization, and reconstruction algorithms. *Journal of applied clinical medical physics*, 18(6):32–48, nov 2017.

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

[6] I. Apostolova, I. G. Steffen, F. Wedel, A. Lougovski, S. Marnitz, T. Derlin, H. Amthauer, R. Buchert, F. Hofheinz, and W. Brenner. Asphericity of pretherapeutic tumour FDG uptake provides independent prognostic value in head-and-neck cancer. *European radiology*, 24(9):2077–87, sep 2014.

[7] J. Atkinson A.J., W. Colburn, V. DeGruttola, D. DeMets, G. Downing, D. Hoth, J. Oates, C. Peck, R. Schooley, B. Spilker, J. Woodcock, and S. Zeger. Biomarkers and surrogate endpoints: Preferred definitions and conceptual framework. *Clinical Pharmacology and Therapeutics*, 69(3):89–95, 2001.

[8] C. Bailly, C. Bodet-Milin, S. Couespel, H. Necib, F. Kraeber-Bodéré, C. Ansquer, and T. Carlier. Revisiting the Robustness of PET-Based Textural Features in the Context of Multi-Centric Trials. *PloS one*, 11(7):e0159984, 2016.

[9] M. A. Balafar, A. R. Ramli, M. I. Saripan, and S. Mashohor. Review of brain MRI image segmentation methods. *Artificial Intelligence Review*, 33(3):261–274, mar 2010.

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

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

[12] N. Boussion, C. C. Le Rest, M. Hatt, and D. Visvikis. 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, jul 2009.

[13] J. C. Caicedo, S. Cooper, F. Heigwer, S. Warchal, P. Qiu, C. Molnar, A. S. Vasilevich, J. D. Barry, H. S. Bansal, O. Kraus, M. Wawer, L. Paavolainen, M. D. Herrmann, M. Rohban, J. Hung, H. Hennig, J. Concannon, I. Smith, P. A. Clemons, S. Singh, P. Rees, P. Horvath, R. G. Linington, and A. E. Carpenter. Data-analysis strategies for image-based cell profiling. *Nature Methods*, 14(9): 849–863, 2017.

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

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

[16] L. P. Clarke, R. J. Nordstrom, H. Zhang, P. Tandon, Y. Zhang, G. Redmond, K. Farahani, G. Kelloff, L. Henderson, L. Shankar, J. Deye, J. Capala, and P. Jacobs. The Quantitative Imaging Network: NCI's Historical Perspective and Planned Goals. *Translational oncology*, 7(1):1–4, feb 2014.

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

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

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

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

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

[22] A. Depeursinge and J. Fageot. Biomedical Texture Operators and Aggregation Functions. In A. Depeursinge, J. Fageot, and O. Al-Kadi, editors, *Biomedical texture analysis*, chapter 3, pages 63–101. Academic Press, London, UK, 1st edition, 2017.

[23] A. Depeursinge, A. Foncubierta-Rodriguez, D. Van De Ville, and H. Müller. Three-dimensional solid texture analysis in biomedical imaging: review and opportunities. *Medical image analysis*, 18(1):176–96, jan 2014.

[24] M.-C. Desseroit, F. Tixier, W. A. Weber, B. A. Siegel, C. Cheze Le Rest, D. Visvikis, and M. Hatt. Reliability of PET/CT Shape and Heterogeneity Features in Functional and Morphologic Components of Non-Small Cell Lung Cancer Tumors: A Repeatability Analysis in a Prospective Multicenter Cohort. *Journal of nuclear medicine*, 58(3):406–411, mar 2017.

[25] I. El Naqa. *Image Processing and Analysis of PET and Hybrid PET Imaging*, pages 285–301. Springer International Publishing, Cham, 2017.

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

[27] European Society of Radiology (ESR). ESR statement on the stepwise development of imaging biomarkers. *Insights into imaging*, 4(2):147–52, apr 2013.

[28] V. Frings, F. H. P. van Velden, L. M. Velasquez, W. Hayes, P. M. van de Ven, O. S. Hoekstra, and R. Boellaard. Repeatability of metabolically active tumor volume measurements with FDG PET/CT in advanced gastrointestinal malignancies: a multicenter study. *Radiology*, 273(2):539–48, nov 2014.

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

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

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

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

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

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

[35] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

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

[37] M. Hatt, M. Majdoub, M. Vallières, F. Tixier, C. C. Le Rest, D. Groheux, E. Hindié, A. Martineau, O. Pradier, R. Hustinx, R. Perdrisot, R. Guillevin, I. El Naqa, and D. Visvikis. 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, jan 2015.

[38] M. Hatt, F. Tixier, L. Pierce, P. E. Kinahan, C. C. Le Rest, and D. Visvikis. Characterization of PET/CT images using texture analysis: the past, the present... any future? *European journal of nuclear medicine and molecular imaging*, 44(1):151–165, jan 2017.

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

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

[41] P. Lambin, R. T. H. Leijenaar, T. M. Deist, J. Peerlings, E. E. C. de Jong, J. van Timmeren, S. Sanduleanu, R. T. H. M. Larue, A. J. G. Even, A. Jochems, Y. van Wijk, H. Woodruff, J. van Soest, T. Lustberg, E. Roelofs, W. J. C. van Elmpt, A. L. A. J. Dekker, F. M. Mottaghy, J. E. Wildberger, and S. Walsh. Radiomics: the bridge between medical imaging and personalized medicine. *Nature reviews. Clinical oncology*, 14(12):749–762, dec 2017.

[42] R. T. H. M. Larue, J. E. van Timmeren, E. E. C. de Jong, G. Feliciani, R. T. H. Leijenaar, W. M. J. Schreurs, M. N. Sosef, F. H. P. J. Raat, F. H. R. van der Zande, M. Das, W. J. C. van Elmpt, and P. Lambin. Influence of gray level discretization on radiomic feature stability for different CT scanners, tube currents and slice thicknesses: a comprehensive phantom study. *Acta oncologica*, pages 1–10, sep 2017.

[43] A. Le Pogam, H. Hanzouli, M. Hatt, C. Cheze Le Rest, and D. Visvikis. Denoising of PET images by combining wavelets and curvelets for improved preservation of resolution and quantitation. *Medical image analysis*, 17(8):877–91, dec 2013.

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

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

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

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

[48] D. Mackin, X. Fave, L. Zhang, J. Yang, A. K. Jones, C. S. Ng, and L. Court. Harmonizing the pixel size in retrospective computed tomography radiomics studies. *PLOS ONE*, 12(9):e0178524, sep 2017.

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

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

[51] P. A. P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37: 17–23, 1950.

[52] J. L. Mulshine, D. S. Gierada, S. G. Armato, R. S. Avila, D. F. Yankelevitz, E. A. Kazerooni, M. F. McNitt-Gray, A. J. Buckler, and D. C. Sullivan. Role of the Quantitative Imaging Biomarker Alliance in optimizing CT for the evaluation of lung cancer screen-detected nodules. *Journal of the American College of Radiology*, 12(4):390–5, apr 2015.

[53] R. J. Nordstrom. The quantitative imaging network in precision medicine. *Tomography*, 2(4):239, 2016.

[54] J. P. B. O'Connor, E. O. Aboagye, J. E. Adams, H. J. W. L. Aerts, S. F. Barrington, A. J. Beer, R. Boellaard, S. E. Bohndiek, M. Brady, G. Brown, D. L. Buckley, T. L. Chenevert, L. P. Clarke, S. Collette, G. J. Cook, N. M. DeSouza, J. C. Dickson, C. Dive, J. L. Evelhoch, C. Faivre-Finn, F. A. Gallagher,

F. J. Gilbert, R. J. Gillies, V. Goh, J. R. Griffiths, A. M. Groves, S. Halligan, A. L. Harris, D. J. Hawkes, O. S. Hoekstra, E. P. Huang, B. F. Hutton, E. F. Jackson, G. C. Jayson, A. Jones, D.-M. Koh, D. Lacombe, P. Lambin, N. Lassau, M. O. Leach, T.-Y. Lee, E. L. Leen, J. S. Lewis, Y. Liu, M. F. Lythgoe, P. Manoharan, R. J. Maxwell, K. A. Miles, B. Morgan, S. Morris, T. Ng, A. R. Padhani, G. J. M. Parker, M. Partridge, A. P. Pathak, A. C. Peet, S. Punwani, A. R. Reynolds, S. P. Robinson, L. K. Shankar, R. A. Sharma, D. Soloviev, S. Stroobants, D. C. Sullivan, S. A. Taylor, P. S. Tofts, G. M. Tozer, M. van Herk, S. Walker-Samuel, J. Wason, K. J. Williams, P. Workman, T. E. Yankeelov, K. M. Brindle, L. M. McShane, A. Jackson, and J. C. Waterton. Imaging biomarker roadmap for cancer studies. *Nature Reviews Clinical Oncology*, 14 (3):169–186, mar 2017.

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

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

[57] M. Shafiq-Ul-Hassan, G. G. Zhang, K. Latifi, G. Ullah, D. C. Hunt, Y. Balagurunathan, M. A. Abdalah, M. B. Schabath, D. G. Goldgof, D. Mackin, L. E. Court, R. J. Gillies, and E. G. Moros. Intrinsic dependencies of CT radiomic features on voxel size and number of gray levels. *Medical physics*, 44(3):1050–1062, mar 2017.

[58] I. Shiri, A. Rahmim, P. Ghaffarian, P. Geramifar, H. Abdollahi, and A. Bitarafan-Rajabi. The impact of image reconstruction settings on 18F-FDG PET radiomic features: multi-scanner phantom and patient studies. *European Radiology*, 27(11):4498–4509, nov 2017.

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

[60] K. Smith, Y. Li, F. Piccinini, G. Csucs, C. Balazs, A. Bevilacqua, and P. Horvath. CIDRE: An illumination-correction method for optical microscopy. *Nature Methods*, 12(5):404–406, 2015.

[61] L.-K. Soh and C. Tsatsoulis. Texture analysis of sar sea ice imagery using gray level co-occurrence matrices. *IEEE Transactions on Geoscience and Remote Sensing*, 37(2):780–795, 1999.

[62] M. Sollini, L. Cozzi, L. Antunovic, A. Chiti, and M. Kirienko. PET Radiomics in NSCLC: state of the art and a proposal for harmonization of methodology. *Scientific reports*, 7(1):358, mar 2017.

[63] C. Solomon and T. Breckon. Features. In *Fundamentals of Digital Image Processing*, chapter 9, pages 235–262. John Wiley & Sons, Ltd, Chichester, UK, jan 2011.

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

[65] P. Stelldinger, L. J. Latecki, and M. Siqueira. 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, jan 2007.

[66] D. C. Sullivan, N. A. Obuchowski, L. G. Kessler, D. L. Raunig, C. Gatsonis, E. P. Huang, M. Kondratovich, L. M. McShane, A. P. Reeves, D. P. Barboriak, A. R. Guimaraes, R. L. Wahl, and RSNA-QIBA Metrology Working Group. Metrology Standards for Quantitative Imaging Biomarkers. *Radiology*, 277(3): 813–25, dec 2015.

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

[68] P. Thévenaz, T. Blu, and M. Unser. Image interpolation and resampling. In *Handbook of medical imaging*, pages 393–420. Academic Press, Inc., 2000.

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

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

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

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

[73] M. Vallières, C. R. Freeman, S. R. Skamene, and I. El Naqa. 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, jul 2015.

[74] M. Vallières, E. Kay-Rivest, L. J. Perrin, X. Liem, C. Furstoss, H. J. W. L. Aerts, N. Khaouam, P. F. Nguyen-Tan, C.-S. Wang, K. Sultanem, J. Seuntjens, and I. El Naqa. Radiomics strategies for risk assessment of tumour failure in head-and-neck cancer. *Scientific reports*, 7:10117, aug 2017.

[75] M. Vallieres, A. Zwanenburg, B. Badic, C. Cheze-Le Rest, D. Visvikis, and M. Hatt. Responsible radiomics research for faster clinical translation, 2017.

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

[77] J. J. van Griethuysen, A. Fedorov, C. Parmar, A. Hosny, N. Aucoin, V. Narayan, R. G. Beets-Tan, J.-C. Fillion-Robin, S. Pieper, and H. J. Aerts. Computational radiomics system to decode the radiographic phenotype. *Cancer research*, 77 (21):e104–e107, 2017.

[78] F. H. P. van Velden, P. Cheebsumon, M. Yaqub, E. F. Smit, O. S. Hoekstra, A. A. Lammertsma, and R. Boellaard. 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, sep 2011.

[79] F. H. P. van Velden, G. M. Kramer, V. Frings, I. A. Nissen, E. R. Mulder, A. J. de Langen, O. S. Hoekstra, E. F. Smit, and R. Boellaard. 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, oct 2016.

[80] U. Vovk, F. Pernus, and B. Likar. A review of methods for correction of intensity inhomogeneity in MRI. *IEEE transactions on medical imaging*, 26(3):405–21, mar 2007.

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

[82] J. C. Waterton and L. Pylkkanen. Qualification of imaging biomarkers for oncology drug development. *European journal of cancer*, 48(4):409–15, mar 2012.

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

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

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

[86] A. Zwanenburg, S. Leger, L. Agolli, K. Pilz, E. G. C. Troost, C. Richter, and S. Löck. Assessing robustness of radiomic features by image perturbation. *eprint arXiv:1806.06719 [cs.CV]*, jun 2018.