



Spatial logics and model checking for medical imaging

Fabrizio Banci Buonamici¹ · Gina Belmonte¹ · Vincenzo Ciancia² · Diego Latella² · Mieke Massink²

Published online: 22 February 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

Recent research on spatial and spatio-temporal model checking provides novel image analysis methodologies, rooted in logical methods for topological spaces. Medical imaging (MI) is a field where such methods show potential for ground-breaking innovation. Our starting point is SLCS, the *Spatial Logic for Closure Spaces*—closure spaces being a generalisation of topological spaces, covering also discrete space structures—and `topochecker`, a model checker for SLCS (and extensions thereof). We introduce the logical language `ImgQL` (“Image Query Language”). `ImgQL` extends SLCS with logical operators describing *distance* and *region similarity*. The spatio-temporal model checker `topochecker` is correspondingly enhanced with state-of-the-art algorithms, borrowed from computational image processing, for efficient implementation of distance-based operators, namely *distance transforms*. Similarity between regions is defined by means of a *statistical similarity* operator, based on notions from *statistical texture analysis*. We illustrate our approach by means of an example of analysis of Magnetic Resonance images: segmentation of *glioblastoma* and its *oedema*.

Keywords Spatial logics · Closure spaces · Model checking · Medical imaging · Segmentation · Magnetic resonance imaging · Distance transform · Statistical texture analysis

1 Introduction

Computer Science plays a fundamental role in the field of medical image analysis. Computational methods are currently in use for several different purposes, such as: *computer-aided diagnosis* (CAD), aiming at the classification of areas in images, based on the presence of signs of specific diseases [34]; *Image Segmentation*, tailored to iden-

tify areas that exhibit specific features or functions (such as organs or sub-structures) [43]; *Automatic contouring* of Organs at Risk or target volumes for radiotherapy applications [12]; Identification of *indicators*, computed from the acquired images, enabling early diagnosis, or understanding of microscopic characteristics of specific diseases, or helping in the identification of prognostic factors to predict a treatment output [18,77] (examples of indicators are the *mean diffusivity* and the *fractional anisotropy* obtained from Magnetic Resonance (MR) Diffusion-Weighted Images, or the *magnetisation transfer ratio* maps obtained from a Magnetisation Transfer acquisition [32,55]).

Such kinds of analyses are strictly tied to the spatial features of images.

In this paper, we focus on image segmentation, in particular to identify glioblastomas which are the most common malignant intracranial tumours. For the treatment of glioblastomas, neuroimaging protocols are used before and after treatment to evaluate the effect of treatment strategies and to monitor the evolution of the disease. In clinical studies and routine treatment, magnetic resonance images (MRI) are evaluated based mostly on qualitative criteria such as the presence of hyperintense tissue appearing in the images [60]. The study and development of automatic and semi-

This work is partially supported by Azienda Ospedaliera Universitaria Senese.

✉ Vincenzo Ciancia
Vincenzo.Ciancia@cnr.it

Fabrizio Banci Buonamici
F.Banci@ao-siena.toscana.it

Gina Belmonte
G.Belmonte@ao-siena.toscana.it

Diego Latella
Diego.Latella@cnr.it

Mieke Massink
Mieke.Massink@cnr.it

¹ Azienda Ospedaliera Universitaria Senese, Siena, Italy

² CNR - Ist. di Scienza e Tecnologie dell’Informazione “A. Faedo”, Pisa, Italy

automatic segmentation algorithms are aiming at overcoming the current time-consuming practise of manual delineation of such tumours and at providing an accurate, reliable and reproducible method of segmentation of the tumour area and related tissues [35].

Segmentation of medical images, and brain segmentation in particular, is nowadays an important topic on its own in many applications in neuroimaging; several automatic and semi-automatic methods have been proposed, constituting a very active research area (see for example [33,35,38,54,72,80]). Segmentation methods are often divided into two broad categories [60]: *generative models*, which rely on domain-specific prior knowledge about the appearance of specific brain tissues and anatomy, and *discriminative models*, which exploit little prior knowledge and instead rely mostly on the extraction of a large number of low-level image features such as local histograms, texture features and raw input pixel values. Among the main principles of segmentation algorithms is the exploitation of the variation of grey-scale intensity and texture of images. These features are then used in a wide range of fully automatic and semi-automatic techniques. Among them, we find techniques such as region growing, deformable templates, thresholding and pattern recognition techniques like adaptive fuzzy clustering, artificial neural networks, particle swarm optimisation and random forest. There are also hybrid techniques in which several such techniques are combined. Semi-automatic techniques include fuzzy cognitive maps, support vector machines and neural networks. For further details on the enormous variety of proposed techniques and their performance in terms of accuracy and time complexity, we refer to the following surveys [8,60,62]. At the time of writing, the state of the art in automated segmentation of brain tumours, which is our running example, is almost entirely constituted by various machine learning methods¹. One of the technical challenges of the development of automated (brain) tumour segmentation is that lesion areas are only defined through changes in the intensity (luminosity) in the (black & white) images that are *relative* to surrounding normal tissue. Even manual segmentation by experts shows significant variations when intensity gradients between adjacent tissue structures are smooth or partially obscured [60]. Furthermore, there is a considerable variation across images from different patients and images obtained with different MRI scanners.

In this paper, we propose a novel approach to image segmentation, namely an interactive, logic-based method,

supported by *spatial model checking*, tailored to loosely identify a region of interest in MRI on which to focus the analysis of glioblastoma or other types of tumours. This approach is particularly suitable to exploit the *relative* spatial relations between tissues of interest mentioned earlier. With respect to the aforementioned categories, our approach is that of a hybrid methodology in which the different segmentation methods (such as texture features, local histogram processing, and prior knowledge) can be freely combined and nested, since they are mapped to operators of a domain-specific language for image analysis.

Spatial and Spatio-temporal logics and model checking are the subject of a recent trend in Computer Science (see for instance [22–24,31,45,46,64]) that uses specifications written in logical languages describing *space*—called *spatial logics*—to automatically identify patterns and structures of interest in a variety of domains, ranging from Collective Adaptive Systems [20,25,26] to signals [64] and images [23,46].

The research presented in the present paper stems from the *topological* approach to spatial logics, dating back to the work of Alfred Tarski, who first recognised the possibility of reasoning on space using topology as a mathematical framework for the interpretation of modal logic (see [78] for a thorough introduction). In this context, formulas are interpreted as sets of points of a topological space, and in particular the modal operator \diamond is usually interpreted as the (logical representation of the) topological *closure* operator. A standard reference is the *Handbook of Spatial Logics* [3]. Therein, several spatial logics are described, with applications far beyond topological spaces; such logics treat not only aspects of morphology, geometry and distance, but also advanced topics such as dynamic systems, and discrete structures, that are particularly difficult to deal with from a topological perspective (see, for example [39]); in addition, the idea of investigating a *spatial*, topological, interpretation of the temporal logic *until* operator is discussed (see [78]). In recent work [22,23], Ciancia et al. pushed such theoretical developments further to encompass arbitrary graphs as models of space, by choosing *Closure spaces*, a generalisation of topological spaces [39,40], as underlying model for the logic. This resulted in the definition of the *Spatial Logic for Closure Spaces* (SLCS), and a related model checking algorithm. SLCS includes both the closure modality \diamond —denoted by \mathcal{N} (to be read for “near”) for avoiding confusion—and a spatial until operator—denoted by \mathcal{S} (to be read “surrounded”); a point satisfies $\Phi_1 \mathcal{S} \Phi_2$ if it satisfies Φ_1 and there is no way for moving away to a point not satisfying Φ_1 without first passing by a point satisfying Φ_2 , i.e. the points satisfying Φ_1 are surrounded by points satisfying Φ_2 . Subsequently, in [21], a spatio-temporal logic, combining *Computation Tree Logic* and the newly defined spatial operators, was introduced; the (extended) model checking algorithm has been

¹ This is witnessed by current results that are publicly evaluated in the latest editions of the *multimodal brain tumour image segmentation benchmark (BraTS)*, see https://www.cbica.upenn.edu/sbia/Spyridon.Bakas/MICCAI_BraTS/MICCAI_BraTS_2017_proceedings_shortPapers.pdf.

implemented in the prototype *spatio-temporal model checker topochecker*.²

The broader aim of our research in the context of medical imaging is to enable the *declarative description* and *automatic* or *semi-automatic*, efficient identification of regions in images using *spatial logic* formulas. This is possible by considering such images as instances of (quasi-discrete) closure spaces. The tools and methods we introduce can be used both for two-dimensional (2D) and three-dimensional (3D) MI; we remark that modern MRI machines can usually provide 3D data for analysis; however, in standard practice, 3D information is often discarded in favour of 2D (slice by slice) analysis, due to the lack of well-established methods for 3D analysis. Using 3D information may lead to improved accuracy, and it is therefore of high interest, in current research, to identify techniques for this purpose.

1.1 Original contributions

This paper details and extends the ideas outlined in [9], providing several further original contributions:

- extension of the spatial logic SLCS to *ImgQL*, introducing distance-based operators and showing their formal relation to the other spatial logic operators of SLCS. A novel approach to model checking of distance-based operators is provided based on so-called *distance transforms*, that forms the basis for the definition an efficient algorithm to solve the model checking problem. Asymptotic time complexity of the procedure we propose is linear or quasi-linear, depending on the kind of distance used. This result makes such procedure suitable for the analysis of higher resolution or 3D images;
- introduction of a novel logical connective aimed at estimating similarity between regions. This operator is based on *statistical texture analysis* and is able to classify points of the space based on the similarity between the area where they are localised, and a target region, expressed in logical terms. The connective is specific for medical image analysis. Its embedding shows how such connectives can be integrated into the spatial logic. This provides an example of how other specialised existing algorithms could be introduced and exploited within the spatial logic model checking framework;
- enhancement of the results in the glioblastoma case study first introduced in [9], providing the relevant technical details on the logical specification;
- development of efficient model checking algorithms, that are competitive in computational efficiency w.r.t. state-of-the-art (semi-)automatic segmentation approaches.

A major advantage of our formal verification approach, with respect to the state of the art in (semi-)automated segmentation, is that logical specifications are transparent, reproducible, accurate, human readable, and applicable to both 2D and 3D images.

Texture analysis, distance, and reachability in space can be freely combined as high-level logical operators with a clear and well-defined topological semantics. The interplay of these aspects is the key to obtain our experimental results. The work in [9] constituted a first *proof-of-concept* study. In that study, *topochecker* was used for the declarative specification of regions in medical images. The model checker was used to automatically and efficiently identify and colour *glioblastoma* and the surrounding *oedema* in MRI scans, on the basis of a declarative definition of the two *regions of interest*, given in terms of their visual appearance. The latter is defined by image features such as proximity, interconnection, and texture similarity. The input to the model checker consists of a precise, declarative, unambiguous logical specification, that besides being fairly close to the level of abstraction of an expert description of the process, is also remarkably concise, human readable, robust and reproducible.

1.1.1 Related work

The idea of using model checking, and in particular spatial or spatio-temporal model checking, for the analysis of medical images is relatively recent and there are only a few articles exploring this field so far. In particular, [74] uses spatio-temporal model checking techniques inspired by [45]—pursuing machine learning of the logical structure of image features—for the detection of tumours. In contrast, our approach is more focused on human-intelligible logical descriptions that can be reused and extended. Other interesting work is that in [66] where spatio-temporal meta model checking is used for the analysis of biological processes, with an interesting focus on *multi-scale* aspects.

Among the fully automated approaches that recently are gaining interest are those based on machine learning and, in particular, *deep learning* (see for example [4] for a recent review). Although manual segmentation is still the standard for *in vivo* images, this method is expensive and time-consuming, difficult to reproduce and possibly inaccurate due to human error. Machine learning and deep learning approaches have shown promising results in pattern recognition in areas where large, reliable datasets are available and are currently being developed for application in MRI based brain segmentation with the aim to obtain reliable automatic segmentation methods. Deep learning is based on the use of artificial neural networks, consisting of several layers, that can extract a hierarchy of features from raw input data. These methods depend heavily on the availability of large training datasets. Furthermore, some machine learning approaches

² Topochecker: a topological model checker, see <http://topochecker.isti.cnr.it>, <https://github.com/vincenzoml/topochecker>.

require the generation of manual *ground truth* labels, i.e. data sets in which segments of interest are indicated by experts manually in a standard way. This is a complicated task not only because it is very laborious, but also because of the relatively high intra-expert and inter-expert variability of $20\pm 15\%$ and $28\pm 12\%$, respectively, for manual segmentations of brain tumour images [59]. Interactive approaches based on spatial model checking may therefore also be of help to improve the generation of manual ground truth labels in a more efficient, transparent, and reproducible way.

Spatial (and spatio-temporal) model checking is a relatively new area of research, so that there are not that many proposals available in the literature. In [63], SSTL, a spatial extension of the linear time, time-bounded, Signal Temporal Logic (STL), is presented which provides a logical framework for reasoning about (stochastic) population models where agents live—and move around—in a discrete patch-based representation of space, i.e. a finite (cost-)weighted graph. The logic has a single spatial operator, namely the bounded *somewhere* operator, $\diamond_I \Phi$ —reminiscent of the *somewhere* operator of [67]—which requires Φ to hold in a location reachable from the current one with a total cost falling in interval I . A monitoring—rather than model checking—algorithm is also provided, extending the similar algorithm for STL with the spatial component. In [64,65], some of the co-authors of the present paper, jointly with Nenzi et al. proposed an extension of SSTL with a *bounded surrounded* operator \mathcal{S}_I which, in turn, extends the surrounded operator of [22,23] with the additional requirement that, in order for a point x to satisfy $\Phi_1 \mathcal{S}_I \Phi_2$, it must also be the case that the length of the shortest path from x to the “boundary” Φ_2 must fall in I . Finally, in [5], an alternative characterisation of the bounded surrounded operator is proposed, which defines it as derived from a (basic) *reachability* operator and a new operator, called *escape*. Note that in [22,23], reachability is available in turn as a derived operator—namely, the *dual* of \mathcal{S} .

1.1.2 Outline

A technical introduction to spatial logics and distance-based operators is provided in Sect. 2. Syntax and semantics of the fragment of SLCS we will use in this paper are recalled, as well as the main notions of spatial model checking for the fragment. The definition of a distance operator for *ImgQL* is presented in Sect. 3 together with the related extension of the model checking algorithm. In Sect. 4, the logic framework we propose for statistical texture analysis is presented. Section 5 briefly describes the tool *topochecker*, and provides the results of some benchmarks of its spatial analysis capabilities. In Sect. 6, the glioblastoma case study is presented in detail, including a first assessment of validation. Some concluding remarks are provided in Sect. 7.

2 Logics for closure spaces

In this section, we discuss the background knowledge that we use in the technical developments of the paper. In particular, we briefly introduce the notion of closure spaces, the fragment of SLCS [22,23] we use in this paper, and the related model checking algorithm.

In the sequel, we will often make explicit reference to 2D images and their pixels; here, we point out that this is done only for the sake of simplicity and that all notions, notations, definitions, and results equally apply to 3D images and their voxels (i.e. *volumetric picture elements*, the 3D counterpart of pixels).

2.1 Closure spaces, spatial logics, and model checking

In *spatial logics*, modal operators are interpreted using the concept of *neighbourhood* in a topological space, enabling one to reason about *points* of the space using familiar concepts such as proximity, distance, or reachability. A comprehensive reference for these theoretical developments is [3]. Transferring the results in the field to applications, and in particular to model checking, requires one to use *finite* models. However, finite topological spaces are not satisfactory in this respect; for instance, they cannot encode *arbitrary* graphs, including e.g. those with a non-transitive/non-symmetric edge relation, that may be the object of spatial reasoning in several applications (for instance, consider the graph of roads in a town, including the one-way streets). Extending topological spaces to *closure spaces* (see [39]) is the key to generalise these results. In this paper, we use a fragment of SLCS comprising an operator, called *near*, interpreted as proximity, and the *surrounded* connective, which is a spatial variant of the classical temporal *weak until* operator, able to characterise unbounded areas of space, based on their boundary properties. The *surrounded* connective is similar in spirit to the spatial *until* operator for topological spaces discussed by Aiello and van Benthem in [2,78], although it is interpreted in closure spaces. Several derived operators may be defined, among which, notably, variants of the notion of *reachability* in space. The combination of SLCS with temporal operators from the well-known branching time logic CTL (Computation Tree Logic) [28] has been explored in [21]. Some related case studies have been analysed in [19,26] where the logic caters for spatio-temporal reasoning and model checking. In the present paper, we focus on spatial properties; therefore, we restrict our attention to spatial aspects of our framework.

2.2 A fragment of SLCS

SLCS is a logic for *space*, where the latter is modelled by means of *closure spaces*. Before introducing the fragment

of SLCS we use in the present paper, we recall some basic notions of closure spaces [39,40].

Definition 1 A *closure space* is a pair (X, \mathcal{C}) where X is a non-empty set (of *points*) and $\mathcal{C} : 2^X \rightarrow 2^X$ is a function satisfying the following three axioms:

1. $\mathcal{C}(\emptyset) = \emptyset$;
2. $Y \subseteq \mathcal{C}(Y)$ for all $Y \subseteq X$;
3. $\mathcal{C}(Y_1 \cup Y_2) = \mathcal{C}(Y_1) \cup \mathcal{C}(Y_2)$ for all $Y_1, Y_2 \subseteq X$. □

According to the well-known Kuratowski definition, adding the *idempotence axiom* $\mathcal{C}(\mathcal{C}(Y)) = \mathcal{C}(Y)$ for all $Y \subseteq X$ in Definition 1 makes it a definition of topological spaces [40]. Consequently, the latter are a subclass of closure spaces.

Given any relation $R \subseteq X \times X$, function $\mathcal{C}_R : 2^X \rightarrow 2^X$ with $\mathcal{C}_R(Y) = Y \cup \{x \mid \exists y \in Y. y R x\}$ satisfies the axioms of Definition 1 thus making (X, \mathcal{C}_R) a closure space. The class of closure spaces generated by binary relations on the set of points represent a very interesting subclass of closure spaces, known as *quasi-discrete* closure spaces. Quasi-discrete closure spaces include discrete structures like graphs, where each graph (X, R) with set of nodes X and set of the edges R is in one-to-one correspondence with closure space (X, \mathcal{C}_R) . Clearly, *finite* closure spaces are quasi-discrete closure spaces.

The following definition is instrumental for the definition of *paths* over quasi-discrete closure spaces.

Definition 2 A *continuous function* $f : (X_1, \mathcal{C}_1) \rightarrow (X_2, \mathcal{C}_2)$ is a function $f : X_1 \rightarrow X_2$ such that, for all $Y \subseteq X_1$, we have $f(\mathcal{C}_1(Y)) \subseteq \mathcal{C}_2(f(Y))$. □

In the definition below, $(\mathbb{N}, \mathcal{C}_{Succ})$ is the closure space of natural numbers with the *successor* relation: $(n, m) \in Succ \Leftrightarrow m = n + 1$.

Definition 3 A *path* π in (X, \mathcal{C}_R) is a continuous function $\pi : (\mathbb{N}, \mathcal{C}_{Succ}) \rightarrow (X, \mathcal{C}_R)$. □

The elements of \mathbb{N} will be called *indexes* in the context of paths.

A quasi-discrete closure space (X, \mathcal{C}_R) can be used as the basis for a mathematical model of a 2D digital image; X represents the finite set of *pixels*, and R is the reflexive and symmetric *adjacency* relation between pixels [41]. We note in passing that several different adjacency relations can be used. For instance, in the *orthogonal* adjacency relation (sometimes called von Neumann adjacency) only pixels which share an edge count as adjacent, so that each pixel is adjacent to (itself and) four other pixels; on the other hand, in the *orthodiagonal* adjacency relation pixels are adjacent as long as they share at least either an edge or a corner, so that

$$\begin{array}{l|l}
 \Phi ::= p & [\text{ATOMIC PREDICATE}] \\
 | \neg \Phi & [\text{NEGATION}] \\
 | \Phi_1 \wedge \Phi_2 & [\text{CONJUNCTION}] \\
 | \mathcal{N} \Phi & [\text{NEAR}] \\
 | \Phi_1 \mathcal{S} \Phi_2 & [\text{SURROUNDED}]
 \end{array}$$

Fig. 1 Syntax of the fragment of SLCS

each pixel is adjacent to (itself and) eight other pixels. Pixels are usually associated with specific attributes, such as colours and/or colour-intensity. We model this by assuming that a set A of point *attribute names* is given and by enriching (X, \mathcal{C}_R) with an *attribute evaluation* function $\mathcal{A} : X \times A \rightarrow V$ from points and their attributes to some value set V such that $\mathcal{A}(x, a) \in V$ is the *value* of attribute a of point x . For given set P of *atomic predicates* p , the syntax of the fragment of SLCS we use in this paper is given in Fig. 1.

Informally, it is assumed that space is modelled by a set of points; each atomic predicate $p \in P$ models a specific *feature of points* and is thus associated with the set of points which have this feature. A point x satisfies $\mathcal{N} \Phi$ if a point satisfying Φ can be reached from x in at most one (closure) step, i.e. if x is *near* (or *close*) to a point satisfying Φ . A point x satisfies $\Phi_1 \mathcal{S} \Phi_2$ if it satisfies Φ_1 and in any path π rooted in x (i.e. such that $\pi(0) = x$) and passing through a point $\pi(\ell)$ *not* satisfying Φ_1 , there is a point $\pi(j)$ before or at ℓ (i.e. $0 < j \leq \ell$) that satisfies Φ_2 . In other words, x belongs to an area satisfying Φ_1 and one cannot *escape* from such an area without hitting a point Φ_2 , i.e. x is *surrounded* by Φ_2 . Finally, the fragment includes logical negation (\neg) and conjunction (\wedge).

The above description is formalised by the definition of *model* and *satisfaction relation*:

Definition 4 A *closure model* is a tuple $((X, \mathcal{C}), \mathcal{A}, \mathcal{V})$ consisting of a closure space (X, \mathcal{C}) , a valuation $\mathcal{A} : X \times A \rightarrow V$, assigning to each point and attribute the value of the attribute at that point, and a valuation $\mathcal{V} : P \rightarrow 2^X$ assigning to each atomic predicate the set of points where it holds. □

In the sequel, we assume that an atomic predicate p can be bound to an assertion α , the latter stating a property of attributes, and we use the syntax $p := \alpha$ for atomic predicate definitions, to this purpose. Assertions are standard Boolean expressions, e.g. comparisons of the form $a \geq c$, for $c \in V$, and compositions thereof; we refrain from specifying the actual syntax of assertions, and we assume valuation \mathcal{A} be extended in the obvious way in order to evaluate assertions, e.g. $\mathcal{A}(x, a \geq c) = \mathcal{A}(x, a) \geq c$.

Definition 5 *Satisfaction* $\mathcal{M}, x \models \Phi$ at point $x \in X$ in model $\mathcal{M} = ((X, \mathcal{C}), \mathcal{A}, \mathcal{V})$ is defined by induction on the structure of formulas, as in Fig. 2. □

$\mathcal{M}, x \models p \in P \iff x \in \mathcal{V}(p)$
 $\mathcal{M}, x \models \neg \Phi \iff \mathcal{M}, x \models \Phi$ does not hold
 $\mathcal{M}, x \models \Phi_1 \wedge \Phi_2 \iff \mathcal{M}, x \models \Phi_1$ and $\mathcal{M}, x \models \Phi_2$
 $\mathcal{M}, x \models \mathcal{N} \Phi \iff x \in \mathcal{C}(\{y \mid \mathcal{M}, y \models \Phi\})$
 $\mathcal{M}, x \models \Phi_1 \mathcal{S} \Phi_2 \iff \mathcal{M}, x \models \Phi_1$ and
 for all paths π and indexes ℓ :
 $\pi(0) = x$ and $\mathcal{M}, \pi(\ell) \models \neg \Phi_1$
 implies
 there exists index j such that:
 $0 < j \leq \ell$ and $\mathcal{M}, \pi(j) \models \Phi_2$

Fig. 2 Satisfaction relation for the fragment of SLCS; whenever $p := \alpha$ is a definition for p , we assume $x \in \mathcal{V}(p)$ if and only if $\mathcal{A}(x, \alpha)$ yields the truth-value *true*

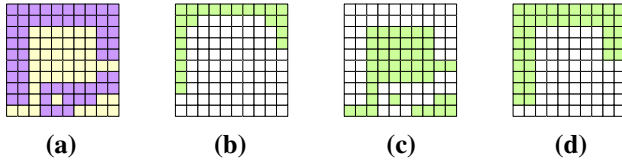


Fig. 3 An example model: **a** the points satisfying atomic predicate p are shown in violet, those satisfying q are shown in yellow; the points satisfying $\neg \mathcal{N}q$ (**b**), those satisfying $q \mathcal{S} p$ (**c**), and those satisfying $p \mathcal{T} \neg(\mathcal{N} q)$ (**d**) are shown in green (colour figure online)

\perp	$\triangleq p \wedge \neg p$	[FALSE]
\top	$\triangleq \neg \perp$	[TRUE]
$\Phi_1 \vee \Phi_2$	$\triangleq \neg(\neg \Phi_1 \wedge \neg \Phi_2)$	[DISJUNCTION]
$\mathcal{I} \Phi$	$\triangleq \neg(\mathcal{N} \neg \Phi)$	[INTERIOR]
$\Phi_1 \mathcal{R} \Phi_2$	$\triangleq \neg(\neg \Phi_2 \mathcal{S} \neg \Phi_1)$	[REACHABILITY]
$\Phi_1 \mathcal{T} \Phi_2$	$\triangleq \Phi_1 \wedge ((\Phi_1 \vee \Phi_2) \mathcal{R} \Phi_2)$	[FROM-TO]
$\mathcal{E} \Phi$	$\triangleq \Phi \mathcal{S} \perp$	[EVERYWHERE]
$\mathcal{F} \Phi$	$\triangleq \neg(\mathcal{E} \neg \Phi)$	[SOMEWHERE]
$\Phi_1 \mathcal{S}_! \Phi_2$	$\triangleq (\Phi_1 \mathcal{S} \Phi_2) \wedge \neg \mathcal{E} \Phi_1$	[STRONG SURROUNDED]

Fig. 4 Some derived operators

In Fig. 3a, a simple finite closure model is shown for which the orthogonal adjacency relation is assumed. All the points satisfying atomic predicate p are shown in violet, whereas those satisfying q are shown in yellow (no point satisfies $p \wedge q$ in this example). Figure 3b shows in green the points that satisfy $\neg \mathcal{N}q$, while Fig. 3c shows in green the points satisfying $q \mathcal{S} p$ (i.e. all q -points that are surrounded by p -points; note that, in the example, these are exactly *all* q -points).

A number of useful *derived operators* are defined in Fig. 4. A few words of explanation are worth for the \mathcal{T} operator, while we refer the reader to [23] for a general discussion on SLCS derived operators. A point satisfies $\Phi_1 \mathcal{T} \Phi_2$ if and only if it lays in an area $Y_1 \subseteq X$ the points of which satisfy Φ_1 and Y_1 “touches” a non-empty area Y_2 , the points of which satisfy Φ_2 ; for this reason, sometimes we call the FROM-TO operator “touches”. Figure 3d shows in green the points satisfying $p \mathcal{T} \neg(\mathcal{N} q)$. A useful pattern, that may be used for filtering noise in images, is formula $\mathcal{N} \mathcal{I} \Phi$. The effect of such a formula is to capture the *regular region* [52] included

Input:
 A finite model $\mathcal{M} = ((X, \mathcal{C}_R), \mathcal{A}, \mathcal{V})$
 and a formula Φ ;
Output:
 The set of points in X satisfying Φ ;

Function $\text{Sat}(\mathcal{M}, \Phi) ::=$

Case Φ **of:**

- p : return $\mathcal{V}(p)$;
- $\neg \Phi$: return $X \setminus \text{Sat}(\mathcal{M}, \Phi)$;
- $\mathcal{N} \Phi$: return $\mathcal{C}_R(\text{Sat}(\mathcal{M}, \Phi))$;
- $\Phi_1 \wedge \Phi_2$: return $\text{Sat}(\mathcal{M}, \Phi_1) \cap \text{Sat}(\mathcal{M}, \Phi_2)$;
- $\Phi_1 \mathcal{S} \Phi_2$: $\text{TempBad} := \text{Sat}(\mathcal{M}, \neg(\Phi_1 \vee \Phi_2))$;
 repeat
 $\text{Bad} := \text{TempBad}$;
 $\text{TempBad} := \text{Bad} \cup (\text{Sat}(\mathcal{M}, \Phi_1) \cap \mathcal{C}_R(\text{Bad}))$;
 until $\text{TempBad} = \text{Bad}$;
 return $\text{Sat}(\mathcal{M}, \Phi_1) \setminus \text{Bad}$.

Fig. 5 Sketch of the model checking algorithm for computing the set $\text{Sat}(\mathcal{M}, \Phi)$ of points in X satisfying Φ

in the set of points satisfying Φ ; point x satisfies $\mathcal{N} \mathcal{I} \Phi$ if and only if it is adjacent to at least one point y satisfying Φ which, in turn, is not adjacent to points satisfying $\neg \Phi$. The effect of such a filter is to eliminate small regions, e.g. those consisting of a single point, when these are considered noise or artefacts.

2.3 Model checking SLCS

In this section, we will briefly recall model checking of SLCS [22,23] over finite models. Note that, in the context of the present paper, we are concerned with so-called *global* model checking, i.e. a procedure that, given a finite model and a logic formula, returns the set of *all* points in the model satisfying the formula [29]. More precisely, the model checking algorithm computes a function Sat such that, for every finite closure model $\mathcal{M} = ((X, \mathcal{C}_R), \mathcal{A}, \mathcal{V})$ and formula Φ , we have $\text{Sat}(\Phi) = \{x \in X \mid \mathcal{M}, x \models \Phi\}$. The algorithm uses recursion on the structure of the formulas and is sketched in Fig. 5. In the present paper, we will focus on the surrounded operator only and we will describe the related section of the model checking algorithm by means of an example. The details of model checking for the other operators are a matter of standard routine and can be found in [22,23].

For a formula $\Phi_1 \mathcal{S} \Phi_2$ the algorithm, roughly speaking, first identifies areas of *bad* points, that is points that can reach a point satisfying $\neg \Phi_1$ without passing by a point satisfying Φ_2 ; then returns the points that satisfy Φ_1 and that are not bad.

Below, we give a brief description of how the algorithm works, using the graphs shown in Fig. 6. Let us consider the model of Fig. 6a as input model, where points are represented by coloured squares and the adjacency relation is the orthogonal one. In this example, we assume that the set of atomic predicates is the set $\{\text{pink}, \text{yellow}, \text{white}\}$ —

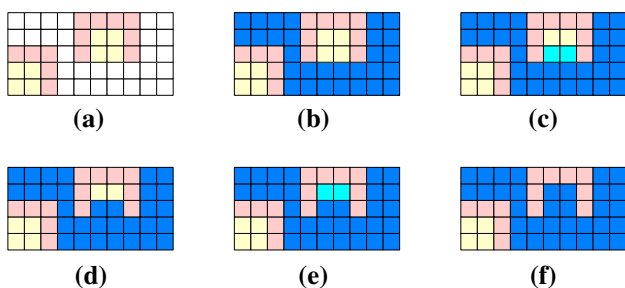


Fig. 6 Model checking *yellow S pink*: **a** input model; **b** set $\{x|\mathcal{M}, x \models \neg(\text{yellow} \vee \text{pink})\}$; **c** set $\{x|\mathcal{M}, x \models \text{yellow}\} \cap C_R(\text{blue})$; **e** set $\{x|\mathcal{M}, x \models \text{yellow}\} \cap C_R(\text{blue})_{6d}$ (colour figure online)

represented in the figure in the obvious way—and that $\mathcal{V}(p) \cap \mathcal{V}(p') = \emptyset$ whenever $p \neq p'$. Suppose the input formula is *yellow S pink*. The result of the assignment $\text{Bad} := \text{TempBad}$ of the first iteration of the `repeat` is shown in Fig. 6b, where all nodes that satisfy $\neg(\text{yellow} \vee \text{pink})$ are marked blue. Note that this blue-colouring is *not* part of the model; we use it at a “meta-level” and only for describing the behaviour of the algorithm; the same will apply to points marked in cyan in the sequel. In Fig. 6c, the (only two) *yellow* points in the closure of the points indicated in blue are shown in cyan³; these are the points to be selected for being added to TempBad in the first iteration of the `repeat`. The new value of TempBad , resulting from the assignment, consists of all blue points of Fig. 6d. The body of the `repeat` is executed now with the new value of TempBad . In Fig. 6e, the (again only two) *yellow* points in the closure of the set of points in blue are shown in cyan. Note that such a closure refers to the model of Fig. 6d; this is abbreviated in Fig. 6e as $C_R(\text{blue})_{6d}$. The new value of TempBad , resulting from the assignment, consists of all points indicated in blue in Fig. 6f. The body of the `repeat` is executed now for the third time, and this results in no change in the value of TempBad : the fixed point is reached, the `repeat` block is exited and the points satisfying *yellow S pink* are the four *yellow* points in the bottom-left corner of Fig. 6f. In [23], it has been shown that, for any finite closure model $\mathcal{M} = ((X, C_R), \mathcal{A}, \mathcal{V})$ and SLCS formula Φ of size k , the model checking procedure terminates in $\mathcal{O}(k \cdot (|X| + |R|))$ steps.⁴

3 Incorporating distance

In this section, we detail the use of so-called *distance operators* in our research line, and we extend the logic fragment

³ In the caption, such a closure is abbreviated by $C_R(\text{blue})$ for space reasons.

⁴ The size of a formula is given by the number of operators in the formula: $\text{size}(p) = 1$; $\text{size}(\neg\Phi) = \text{size}(\mathcal{N}\Phi) = 1 + \text{size}(\Phi)$; $\text{size}(\Phi_1 \wedge \Phi_2) = \text{size}(\Phi_1 \mathcal{S} \Phi_2) = 1 + \text{size}(\Phi_1) + \text{size}(\Phi_2)$.

with a distance operator parametric on the specific notion of distance; we also give an account of the extension of the model checking algorithm necessary for the efficient implementation of the distance operator, based on the notion of *distance transform*.

Models of space as well as spatial logics can be extended with notions of *distance* (see e.g. [52,53,63,64]). Distances are very often expressed using the non-negative real numbers $\mathbb{R}_{\geq 0}$, like the Euclidean distance on continuous space.

For quasi-discrete closure spaces, especially when used as a representation of finite graphs, it is natural to consider *shortest-path distance*, where a path between two nodes is a sequence of consecutive edges connecting the first node to the second, and its length is given by the sum of the lengths of such edges. The length of an edge is often taken to be 1; however, other notions of distance can be more appropriate. For example, *sampling* a multi-dimensional Euclidean space is often done using a *regular grid*, that is, a graph in which the nodes are arranged on multiples of a chosen *unit interval* that may vary along each dimension of the space. Nodes are connected by edges using a chosen notion of adjacency (e.g. the orthogonal or orthodiagonal adjacency relations discussed before, but any choice may be reasonable, depending on the application context). Such graphs can then be *weighted* by associating with each edge the Euclidean distance between the nodes it connects. Graphs with nodes in a Euclidean space and weighted by Euclidean distance are known as *Euclidean graphs* and are naturally equipped with both Euclidean distance between nodes and (weighted) shortest-path distance—which is also called *Chamfer distance* in the particular case of Euclidean graphs with nodes arranged on a regular grid, which is the case of interest for MI. In two-dimensional imaging, pixels—with an application-dependent choice of adjacency—form a Euclidean graph, and Euclidean distance is the reference distance between (the centres of) two pixels.

Euclidean and Chamfer distances obviously divert, no matter how fine is the grid or how many neighbours are chosen in the adjacency relation, unless *all* pairs of nodes are linked by an edge (labelled with the Euclidean distance between the end-points of the edge). Therefore, in this context, Euclidean distance is considered *error-free*, and Chamfer distance is considered an approximation of the former. The chosen adjacency relation determines the precision-efficiency trade-off of the computed distance: the more pixels are considered adjacent, the more precise is the approximation, at the expenses of generating graphs with larger out-degrees. This is illustrated in Figs. 7 and 8. In the first figure, we show a two-dimensional, rectangular image where all and only points at a Euclidean distance larger than a given threshold k from the centre of the figure are coloured in red. In Fig. 8a, the points in red are those at a Chamfer distance larger than k from the centre; in particular, ortho-

Fig. 7 Threshold imposed on Euclidean distance from a point in the centre of image

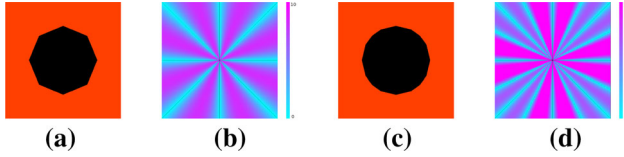
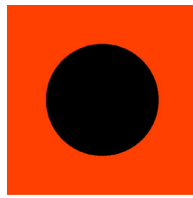


Fig. 8 Percentage error of Chamfer distance: **a** Chamfer distance with 8 adjacent pixels per node, i.e. 3×3 square centred on node; **b** percentage error, scale: 0–10; **c** Chamfer distance with 24 adjacent pixels per node, i.e. 5×5 square centred on node; **d** percentage error, scale: 0–2

diagonal adjacency has been used (each pixel has 8 other adjacent pixels). Figure 8b shows the percentage of error for each pixel with respect to the Euclidean distance, in a scale from 0 to 10%. Finally, in Fig. 8c we use Chamfer distance, the same threshold k and an adjacency relation where each pixel has 24 other adjacent pixels (i.e. the pixels that are adjacent to any pixel form a 5×5 square centred in the pixel). Figure 8d shows the percentage of error w.r.t. the Euclidean distance, in a scale from 0 to 2%. The percentage error $\delta(x)$ for Chamfer distance d_C is defined for each pixel x as $\delta(x) = \frac{|d_E(y,x) - d_C(y,x)|}{d_E(y,x)}$, where y is the central point of the image and d_E denotes the Euclidean distance.

3.1 SLCS with distance operators

In this section, we extend the fragment of SLCS presented in Sect. 2.2 with logic *distance* operators. We first introduce the notion of *distance closure spaces* and, to that purpose we recall the well-known notion of metric space:

Definition 6 A *metric space* is a pair (X, d) where X is a non-empty set (of *points*) and $d : X \times X \rightarrow \mathbb{R}_{\geq 0}$ is function that satisfies the following axioms, for all $x, y, z \in X$:

1. $d(x, y) = 0$ iff $x = y$ [IDENTITY OF INDISCERNIBLE];
2. $d(x, y) = d(y, x)$ [SYMMETRY];
3. $d(x, z) \leq d(x, y) + d(y, z)$ [TRIANGLE INEQUALITY].

Whenever X is a closure space (X, \mathcal{C}) , $((X, \mathcal{C}), d)$ is called a *metric closure space*. \square

Metric functions are easily extended to sets as follows:

Definition 7 Given metric space (X, d) , $x \in X$ and $Y, Z \subseteq X$ we let

1. $d(x, Y) = \inf\{d(x, y) | y \in Y\}$
2. $d(Y, Z) = \inf\{d(y, z) | y \in Y \text{ and } z \in Z\}$

Note that if $Y \neq \emptyset$ is finite, then $\inf\{d(x, y) | y \in Y\} = \min\{d(x, y) | y \in Y\}$; we let $d(x, \emptyset) = \infty$ by definition. \square

In the case of quasi-discrete closure spaces, symmetry may turn out to be too much restrictive. This is for instance the case when the relation R underlying the closure operator \mathcal{C}_R is not symmetric. Similarly, the triangle inequality is not fitting well when more qualitative distance “measures” are used, for instance when the codomain of d is composed of only three values, representing *short*, *medium*, and *large* distance, respectively. For all these reasons, for quasi-discrete closure spaces we often use the less restrictive notion of *distance space*, where only Axiom 1 of Definition 6 above is required.

Definition 8 A *distance closure space* is a tuple $((X, \mathcal{C}), d)$ where (X, \mathcal{C}) is a closure space and $d : X \times X \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$ is function satisfying $d(x, y) = 0$ iff $x = y$.

A *quasi-discrete distance closure space* is a distance closure space $((X, \mathcal{C}_R), d)$ where (X, \mathcal{C}_R) is a quasi-discrete closure space. \square

Distance operators can be added to spatial logics in various ways (see [52] for an introduction). For the purposes of the present paper, we extend SLCS with the operator \mathcal{D}^I , where I is an interval of $\mathbb{R}_{\geq 0}$. The intended meaning is that a point x of a distance closure model satisfies $\mathcal{D}^I \Phi$ if its distance from the set of points satisfying Φ falls in interval I . Below we provide the necessary formal definitions.

Definition 9 A *distance closure model* is a tuple $((X, \mathcal{C}), d, \mathcal{A}, \mathcal{V})$ consisting of a distance closure space $((X, \mathcal{C}), d)$, a valuation $\mathcal{A} : X \times A \rightarrow V$, assigning to each point and attribute the value of the attribute of the point and a valuation $\mathcal{V} : P \rightarrow 2^X$ assigning to each atomic predicate the set of points where it holds. A *quasi-discrete distance closure model* is a distance closure model $((X, \mathcal{C}_R), d, \mathcal{A}, \mathcal{V})$ where $((X, \mathcal{C}_R), d)$ is a quasi-discrete distance closure space. \square

As the definition of d might require the elements of R to be weighted—as in the case of Euclidean graphs—quasi-discrete distance closure models are often enriched with a R -weighting function $\mathcal{W} : R \rightarrow \mathbb{R}$ assigning the weight $\mathcal{W}(x, y)$ to each $(x, y) \in R$.

The satisfaction relation of our fragment of SLCS is extended as expected:

Definition 10 *Satisfaction* $\mathcal{M}, x \models \Phi$ at point $x \in X$ in model $\mathcal{M} = ((X, \mathcal{C}), d, \mathcal{A}, \mathcal{V})$ is defined by induction on the structure of formulas, by adding the equation below to those in Fig. 2:

$$\mathcal{M}, x \models \mathcal{D}^I \Phi \Leftrightarrow d(x, \{y | \mathcal{M}, y \models \Phi\}) \in I$$

$$\begin{aligned}
 \mathcal{D} <^c \Phi &\triangleq \mathcal{D}^{[0,c]} \Phi \\
 \mathcal{D} \leq^c \Phi &\triangleq \mathcal{D}^{[0,c]} \Phi \\
 \mathcal{D} =^c \Phi &\triangleq \mathcal{D}^{[c,c]} \Phi \\
 \mathcal{D} \geq^c \Phi &\triangleq \mathcal{D}^{[c,\infty)} \Phi \\
 \mathcal{D} >^c \Phi &\triangleq \mathcal{D}^{(c,\infty)} \Phi \\
 \mathcal{J} <^c \Phi &\triangleq \mathcal{D} <^c (\neg \mathcal{D} <^c \neg \Phi) \\
 \Phi_1 \mathcal{S}^I \Phi_2 &\triangleq ((\Phi_1 \wedge \neg \Phi_2) \mathcal{S}_I \Phi_2) \wedge \mathcal{D}^I \Phi_2
 \end{aligned}$$

Fig. 9 Additional derived operators

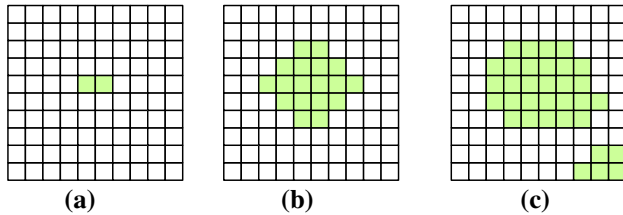


Fig. 10 The points of Fig. 3a satisfying $\mathcal{D} >^2 p$ (a), those satisfying $\mathcal{J} <^3 q$ (b), and those satisfying $\mathcal{J} <^2 q$ (c) are shown in green (colour figure online)

Note that the definition of the SLCS distance operator is *parametric* on the specific distance used. The particular meaning of the distance operator is fully characterised by the specific distance d of the underlying distance closure model. In this paper, we will use the Euclidean distance d_E and the Chamfer distance d_C .

We close this section with the definition of an additional set of derived operators shown in Fig. 9.

Again, with reference to Fig. 3a, Fig. 10a shows in green all the points satisfying $\mathcal{D} >^2 p$, according to the Chamfer distance (over the orthogonal adjacency relation).

Intuitively, the $\mathcal{J} <^c$ operator can be used as a form of filtering, eliminating small details caused by noise in the fine-scale structure of an image; this method is akin to the nested application of \mathcal{N} and \mathcal{I} described in Sect. 2.2, parameterised with respect to a chosen maximum size c of details to be suppressed. To see this, recall that $\mathcal{I}\Phi$ is defined as $\neg \mathcal{N} \neg \Phi$, therefore $\mathcal{N}\mathcal{I}\Phi$ is the same as $\mathcal{N}(\neg \mathcal{N} \neg \Phi)$, which is very similar to the definition of $\mathcal{J} <^c$, with \mathcal{N} replaced by $\mathcal{D} <^c$.

With reference to Fig. 3a, Fig. 10b shows in green the points satisfying $\mathcal{J} <^3 q$, whereas those satisfying $\mathcal{J} <^2 q$ are given in Fig. 10c.

The *bounded surrounded operator* $\Phi_1 \mathcal{S}^I \Phi_2$ is satisfied by a point x if and only if x satisfies Φ_1 and is strongly surrounded by points satisfying Φ_2 and its distance from such points falls in interval I . Note that, in the first argument of \mathcal{S}_I , it is required that $\neg \Phi_2$ holds as well; this ensures that all Φ_2 -points are at a distance of at least $\inf I$ from x .

In Fig. 11a (Fig. 11b, respectively), the points satisfying $q \mathcal{S}^{[2,2]} p$ ($q \mathcal{S}^{[2,3]} p$, respectively) are shown in green. Note that a similar operator has been defined in [64], which turns out to be stronger than $\mathcal{S}^{[a,b]}$, i.e. denoting the former

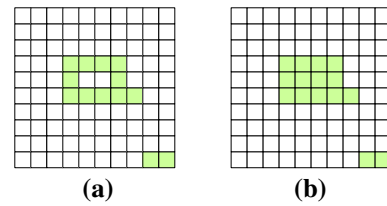


Fig. 11 The points of Fig. 3a satisfying $q \mathcal{S}^{[2,2]} p$ (a) and those satisfying $q \mathcal{S}^{[2,3]} p$ (b) are shown in green (colour figure online)

operator by $\hat{\mathcal{S}}^{[a,b]}$, we have that, for all formulas Φ_1, Φ_2 , $(\Phi_1 \wedge \neg \Phi_2) \hat{\mathcal{S}}^{[a,b]} \Phi_2$ implies $\Phi_1 \mathcal{S}^{[a,b]} \Phi_2$.

We close this section pointing out that, for finite models, the operator $\mathcal{D} \leq^c$ coincides with the operator $\exists \leq^c$ proposed in [71]:

$$\mathcal{M}, x \models \exists \leq^c \Phi \Leftrightarrow \exists y. d(x, y) \leq c \text{ and } \mathcal{M}, y \models \Phi$$

and similarly for $\mathcal{D} <^c$ and $\exists <^c$. Note that this coincidence does not hold in general, e.g. for Euclidean spaces. Our choice of the specific distance operator is motivated by its natural compatibility with distance transforms, as we illustrate in Sect. 3.2.

3.2 Model checking SLCS with distance operators

For distance-based operators, generally speaking, the time complexity of naive model checking algorithms is quadratic in the size of the space (see [64] for an example). However, given a Euclidean graph representing a multi-dimensional image, spatial model checking of formulas $\mathcal{D}^I \Phi$ for Euclidean or Chamfer distance can be done in linear time or quasi-linear time, respectively, with respect to the number of points of the space. This is achieved via so-called *distance transforms*, that are one of the subjects of topology and geometry in computer vision [51] and are extensively used in modern image processing and computer graphics [27]. In particular, effective linear-time algorithms have been recently introduced in the literature [37,58]. Basically, a *distance transform* takes a model \mathcal{M}_{in} as input and produces a model \mathcal{M}_{out} as output as follows. Let \mathcal{M}_{in} be a model $((X, \mathcal{C}_R), d), \mathcal{A}_{in}, \mathcal{V}, \mathcal{W}$ such that every point $x \in X$ has an attribute, say a_{in} , defined on a binary domain, say $\{0, 1\}$ —the value of such an attribute may represent the fact that the point satisfies a given formula Φ or not. The output model will be $\mathcal{M}_{out} = ((X, \mathcal{C}_R), d), \mathcal{A}_{out}, \mathcal{V}, \mathcal{W}$ such that every point $x \in X$ has an attribute, say a_{out} , and $\mathcal{A}_{out}(x, a_{out}) = d(x, \{y \in X \mid \mathcal{A}_{in}(y, a_{in}) = 1\})$.

The algorithm of Fig. 5 is extended with the case for $\mathcal{D}^I \Phi$, sketched in Fig. 13; note that, in the general case, the argument \mathcal{M} of Sat is a *weighted* finite distance closure model $((X, \mathcal{C}_R), d), \mathcal{A}, \mathcal{V}, \mathcal{W}$, where the distance function d may be defined using the weights in \mathcal{W} . It is assumed that two

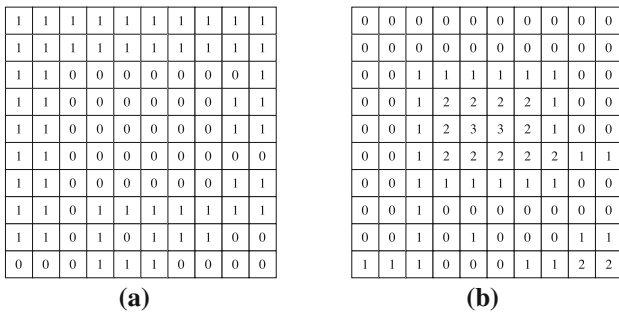


Fig. 12 Intermediate models for $\mathcal{D}^{>2} p$ and input model of Fig. 3a, according to the procedure of Fig. 13

```

 $\mathcal{D}^I \Phi$  : /** Step 1: Compute aux. model  $\mathcal{M}_s$  **/
for all  $x \in X$ :
if  $x \in \text{Sat}(\mathcal{M}, \Phi)$  then  $\mathcal{A}_s[x, a_s] := 1$ 
else  $\mathcal{A}_s[x, a_s] := 0$ 
end for all;
 $\mathcal{M}_s := ((X, \mathcal{C}_R), \mathcal{A}_s, \mathcal{V}, \mathcal{W})$ ;
/** Step 2 Compute aux. model  $\mathcal{M}_d$  **/
for all  $x \in X$ :
 $\mathcal{A}_d[x, a_d] := d(x, \{y \in X | \mathcal{A}_s[y, a_s] = 1\})$ ;
end for all;
 $\mathcal{M}_d := ((X, \mathcal{C}_R), \mathcal{A}_d, \mathcal{V}, \mathcal{W})$ ;
/** Step 3 **/
return  $\{x \in X | \mathcal{A}_d[x, a_d] \in I\}$ .
    
```

Fig. 13 Sketch of the model checking algorithm for $\mathcal{D}^I \Phi$

distinct attributes a_s (playing the role of a_{in} above) and a_d (playing the role of a_{out} above) are defined for all $x \in X$. Two intermediate, auxiliary models \mathcal{M}_s and \mathcal{M}_d are computed— \mathcal{M}_s in Step 1 and \mathcal{M}_d in Step 2—according to the distance transform method described above. In Fig. 13, for the sake of simplicity, functions \mathcal{A} , \mathcal{A}_s and \mathcal{A}_d are treated as (bi-dimensional) arrays. Again with reference to Fig. 3a, and formula $\mathcal{D}^{>2} p$ —shown in Fig. 10a—intermediate models \mathcal{M}_s and \mathcal{M}_d are shown in Fig. 12a, b, respectively. Here, the chosen notion of distance is Chamfer over orthogonal adjacency, considering the Euclidean distance between two adjacent voxels equal to the unit. The values in the points in Fig. 12a (Fig. 12b, respectively) are those of attribute a_s (a_d , respectively).

For what concerns the computation of the specific distance and the related distance transform, for Euclidean distance we use the algorithm that was proposed by Maurer in [58]. Such algorithm computes Euclidean distance transforms on anisotropic multi-dimensional grids (such as 2D and 3D medical images); it has linear complexity, its run-time is predictable, and it is among the most efficient algorithms for the purpose [36]. The general idea of the algorithm is to proceed by induction on the number of dimensions of the image. The distance transform problem in $n + 1$ dimensions is reduced to the problem in n dimensions by a technique that relies on multi-dimensional Voronoi maps. We refer the interested

reader to [27], where a theoretical study of the algorithm is provided. The specification described therein was closely followed in our implementation.

For shortest-path distances over arbitrary directed graphs, we use a variant of the well-known Dijkstra shortest-path algorithm, called “modified Dijkstra distance transform” in [44]. The standard Dijkstra algorithm uses a priority queue sorted by distance from a root node. The queue is initialised to the root node of the considered graph, whose priority is set to 0. In the modified version, when computing the distance transform from a set of nodes identified by formula Φ , the queue is initialised with all the nodes that satisfy Φ and have an outgoing edge reaching a node not satisfying Φ ; all such nodes have priority 0. The algorithm then proceeds as the standard algorithm. As a result, after termination, each node of the graph is labelled with the shortest-path distance from the set of nodes satisfying Φ , as required by the specification. The asymptotic run-time of this procedure is not linear but quasi-linear due to the usage of a priority queue. In this respect, research is still active to optimise the procedure in specific cases (see e.g. [75]). However, the effective run-time behaviour of the algorithm is highly dependent on the structure of the considered graph and the chosen implementation of data structures.

4 A logical framework for texture analysis

In this section, we define an additional logic operator that, when incorporated in the spatial logic presented in the previous sections, provides a logical framework for Texture Analysis (TA).

TA can be used for finding and analysing patterns in (medical) images, including some that are imperceptible to the human visual system. Patterns in images are entities characterised by distinct combinations of features, such as brightness, colour, shape, size. TA includes several techniques and has proved promising in a large number of applications in the field of medical imaging [16,30,50,56]; in particular, it has been used in *Computer Aided Diagnosis* [47,49,79] and for classification or segmentation of tissues or organs [17,69,70]. In TA, image textures are usually characterised by estimating some descriptors in terms of quantitative features. Typically, such features fall into three general categories: syntactic, statistical, and spectral [50]. Our work is mostly focused on *statistical* approaches to texture analysis. For two-dimensional images, statistical methods consist of extracting a set of statistical *descriptors* from the distributions of local features in a neighbourhood of each pixel.

In this paper, we explore the use of *first-order* statistical methods, that are statistics based on the probability distribution function of the intensity values of the pixels of parts,

or the whole, of an image. The classical first-order statistical approach to TA makes use of statistical indicators of the local distribution of image intensity around each pixel, such as *mean, variance, skewness, kurtosis, entropy* [73]. Although such indicators ignore the relative spatial placement of adjacent pixels, statistical operators are useful in MI because their application is invariant under transformations of the image, in particular *affine transformations* (rotation and scaling), which is necessary when analysing several images acquired in different conditions. It is worth mentioning that current research also focuses on constructing features using first-order operators, keeping some spatial coherence, but loosing at least partially the aforementioned invariance [76]. The method we propose is an attempt to improve over the classical setting described above, by analysing (the histograms of) statistical distributions directly.

In the following, we introduce a spatial logic operator that compares image regions in order to classify points that belong to sub-areas in the image where the statistical distribution of the intensity of pixels is *similar* to that of a chosen reference region. Several similarity measures exist (see [61]), that can be used to compare distributions in images. In particular, as a starting point, we use the *cross-correlation* function (also called *Pearson’s correlation coefficient*), that is often used in the context of *image retrieval*, but is also popular in other computer vision tasks. In MI, cross-correlation is also frequently used in the case of *image co-registration* ([13]).⁵

4.1 A logical operator for statistical similarity

The statistical distribution of the values of a numerical attribute in a set of points Y of a space—e.g. the grey levels of the pixels or voxels of an area of a black and white image—is approximated by the *histogram* h of the values of the attribute in the points belonging to Y , as described below. Given a minimum value m , a maximum value M , and a positive number of *bins* k , let $\Delta = (M - m)/k$ and define the histogram h as a function associating with each *bin* $i \in [1, k]$ the number of points that have the value of the attribute in the (half-open) interval $[(i - 1) \cdot \Delta + m, i \cdot \Delta + m)$. The minimum value m and the maximum value M are aimed at improving the resolution of histograms, by excluding rare peaks in the set of points—they might be due to artefacts in acquisition, in the case of images—and would result in a high number of empty bins. A formal definition is given below:

Definition 11 Given closure model $\mathcal{M} = ((X, \mathcal{C}), \mathcal{A}, \mathcal{V})$, we define function $\mathcal{H} : A \times 2^X \times \mathbb{R} \times \mathbb{R} \times \mathbb{N} \rightarrow (\mathbb{N} \rightarrow \mathbb{N})$ such

⁵ In image processing, the problem of *co-registration* is that of mapping two images coming from different sources to the same spatial domain, by finding transformations of the considered images that make given image features coincide.

that for all values $m, M \in \mathbb{R}$, with $m < M$, and $k, i \in \mathbb{N}$, with $k > 0$ and $i \in [1, k]$, letting $\Delta = \frac{M-m}{k}$:

$$\mathcal{H}(a, Y, m, M, k)(i) = |\{y \in Y | (i - 1)\Delta \leq \mathcal{A}(y, a) - m < i\Delta\}| \quad \square$$

So $\mathcal{H}(a, Y, m, M, k)$ is the histogram of the distribution of the values of attribute a of the points in Y , in the interval $[m, M]$ with step Δ .

The definition of *cross-correlation* between two histograms follows. In the sequel, for histogram $h : [1, k] \rightarrow \mathbb{N}$ we let $\bar{h} = \frac{1}{k} \sum_{i=1}^k h(i)$ denote the *mean* of h .

Definition 12 Let $h_1, h_2 : [1, k] \rightarrow \mathbb{N}$ be two histograms. The *cross-correlation* of h_1 and h_2 is defined as follows:

$$\mathbf{r}(h_1, h_2) = \frac{\sum_{i=1}^k (h_1(i) - \bar{h}_1)(h_2(i) - \bar{h}_2)}{\sqrt{\sum_{i=1}^k (h_1(i) - \bar{h}_1)^2} \sqrt{\sum_{i=1}^k (h_2(i) - \bar{h}_2)^2}} \quad \square$$

The value of \mathbf{r} is *normalised* so that $-1 \leq \mathbf{r}(h_1, h_2) \leq 1$; $\mathbf{r}(h_1, h_2) = 1$ indicates that h_1 and h_2 are *perfectly correlated* (that is, $h_1 = ah_2 + b$, with $a > 0$); $\mathbf{r}(h_1, h_2) = -1$ indicates *perfect anti-correlation* (that is, $h_1 = ah_2 + b$, with $a < 0$). On the other hand, $\mathbf{r}(h_1, h_2) = 0$ indicates no correlation. Note that normalisation makes the value of \mathbf{r} undefined for constant histograms, having therefore standard deviation of 0; in terms of statistics, a variable with such standard deviation is only (perfectly) correlated to itself. This special case is handled by letting $\mathbf{r}(h_1, h_2) = 1$ when both histograms are constant, and $\mathbf{r}(h_1, h_2) = 0$ when only one of the h_1 or h_2 is constant.

We are now ready for embedding the statistical similarity operator $\Delta_{\bowtie c} \left[\begin{smallmatrix} m & M & k \\ \rho & a & b \end{smallmatrix} \right]$ in ImgQL .

Definition 13 *Satisfaction* $\mathcal{M}, x \models \Phi$ at point $x \in X$ in model $\mathcal{M} = (((X, \mathcal{C}), d), \mathcal{A}, \mathcal{V})$ is defined by induction on the structure of formulas, by adding the following equation to those in Fig. 2, where $m, M \in \mathbb{R}$, with $m < M$, and $k \in \mathbb{N}$, with $k > 0$:

$$\mathcal{M}, x \models \Delta_{\bowtie c} \left[\begin{smallmatrix} m & M & k \\ \rho & a & b \end{smallmatrix} \right] \Phi \Leftrightarrow \mathbf{r}(h_a, h_b) \bowtie c$$

with

- $h_a(i) = \mathcal{H}(a, S(x, \rho), m, M, k)(i)$,
- $h_b(i) = \mathcal{H}(b, \{y | \mathcal{M}, y \models \Phi\}, m, M, k)(i)$,
- $\bowtie \in \{=, <, >, \leq, \geq\}$, and
- $S(x, \rho) = \{y \in X | d(x, y) \leq \rho\}$ is the *sphere* of radius ρ centred in x . □

So $\Delta_{\bowtie c} \left[\begin{smallmatrix} m & M & k \\ \rho & a & b \end{smallmatrix} \right] \Phi$ compares the region of the space constituted by the sphere of radius ρ centred in x against the

```

 $\Delta_{\geq c} \left[ \begin{smallmatrix} m & M & k \\ \rho & a & b \end{smallmatrix} \right] \Phi$  : for  $j := 1$  to  $k$ :
     $H_b[j] := 0$ 
  end for;
  for all  $y \in \text{Sat}(\mathcal{M}, \Phi)$ :
     $H_b[\text{Idx}(\mathcal{A}[y, b], m, M, k)] :=$ 
       $H_b[\text{Idx}(\mathcal{A}[x, b], m, M, k)] + 1$ 
  end for all;
  Result :=  $\emptyset$ ;
  for all  $x \in X$ :
    for  $j := 1$  to  $k$  :  $H_a[j] := 0$ 
    end for;
    for all  $y \in \{z \in X \mid d(x, z) \leq \rho\}$ :
       $H_a[\text{Idx}(\mathcal{A}[y, a], m, M, k)] :=$ 
         $H_a[\text{Idx}(\mathcal{A}[y, a], m, M, k)] + 1$ 
    end for all;
    if  $r(H_a, H_b) \geq c$ 
    then Result := Result  $\cup \{x\}$ ;
    end for all;
  return Result

```

Fig. 14 Sketch of the model checking algorithm for the statistical similarity operator

region characterised by Φ . The comparison is based on the cross-correlation of the histograms of the chosen attributes of (the points of) the two regions, namely a and b and both histograms share the same domain ($[m, M]$) and the same bins ($[1, k]$). In summary, the operator allows to check *to which extent the sphere around the point of interest is statistically similar to a given region (specified by) Φ* . For example, $\Delta_{\geq 0.7} \left[\begin{smallmatrix} 200 & 2000 & 100 \\ 10.0 & a & b \end{smallmatrix} \right] \top$ is true at voxels centred in a region—of radius 10.0—where the distribution of the values of attribute a has cross-correlation greater than 0.7 with the distribution of the values of attribute b in the whole image. In this case, cross-correlation is computed using 100 bins, and taking into account only values between 200 and 2000.

4.2 Model checking ImgQL with statistical similarity operators

The algorithm of Fig. 5 is extended with the case for $\Delta_{\geq c} \left[\begin{smallmatrix} m & M & k \\ \rho & a & b \end{smallmatrix} \right] \Phi$, sketched in Fig. 14. Auxiliary function Idx computes the bin index associated with the intensity level ℓ of the relevant attribute, relative to parameters m , M and k : $\text{Idx}(\ell, m, M, k) = (\ell \text{ div } \frac{M-m}{k}) + 1$. The extension of the algorithm for implementing the statistical similarity operator is straightforward. An array H_b , sized to the number of bins k , is allocated, initialised to 0 at each index, and the histogram h_b (see Definition 13) is stored in it, by iterating over all the points y satisfying Φ , finding the index i of the bin corresponding to the intensity level of y , and increasing the corresponding value of $H_b[i]$. An additional array H_a , sized to the number of bins k , is allocated. For each pixel x , H_a is (re-)initialised to 0 at each index, and all the points y laying in the sphere of radius ρ centred in x are examined;

for each y , the index i of its bin is identified, and the value of $H_a[i]$ is increased, so that when all the y have been examined, H_a represents the histogram h_a . The cross-correlation value $r(h_a, h_b)$ is then computed by simple calculations that are linear in the number of bins k .

This algorithm has time complexity proportional to $v \cdot \rho^n \cdot k$, where v is the number of pixels in the image and n the number of dimensions (indeed the number of pixels in an n -dimensional sphere is proportional to ρ^n). Since ρ and k are usually fixed for a given analysis, such algorithm can still be considered “linear” in the size of the image. This basic procedure is amenable to optimisation, for instance by observing that the spheres centred around two different points of the image may share some pixels; therefore, the histogram of each one could be computed starting from the histogram of the other, at the expenses of more memory needed to store the histogram of different points. We leave the study of similar optimisations for future work.

5 The tool topochecker

The tool `topochecker`⁶ is a global model checker, capable of analysing models specified as weighted graphs, RGB images, or grayscale medical images. The tool is implemented in the functional programming language OCaml,⁷ which provides a good balance between declarative features and computational efficiency. The output of the tool consists of a region of interest (ROI) for each formula, that is, an image where the specific region where such formula holds is coloured according to a user-specified colour. The spatial model checking algorithm is entirely run in central memory, and it uses memoisation (and on-disk caching) to store the intermediate results, so that when the same sub-formula is used more than once, results are reused.

In order to evaluate the efficiency of the tool, we have designed a set of benchmarks.⁸ Currently, we aim at evaluating the algorithms for Euclidean and Chamfer distance transforms, similarity search via statistical cross-correlation, and the reachability / surrounded primitives. Four benchmarks are currently considered: search for exit and blocked paths in a maze, similar to Example 7.4 in [23], which exercises the implementation of the *surrounded* operator (indirectly, via the *touch* operator); cross-correlation with various radiuses (plus thresholding), using the whole image as a target; Euclidean distance transform (plus thresholding); Chamfer distance transform (plus thresholding). The last three benchmark types use a checkerboard-like pattern

⁶ See <https://github.com/vincenzoml/topochecker>.

⁷ See <http://www.ocaml.org>.

⁸ Available for download at <https://github.com/vincenzoml/spatial-model-checking-benchmarks>.

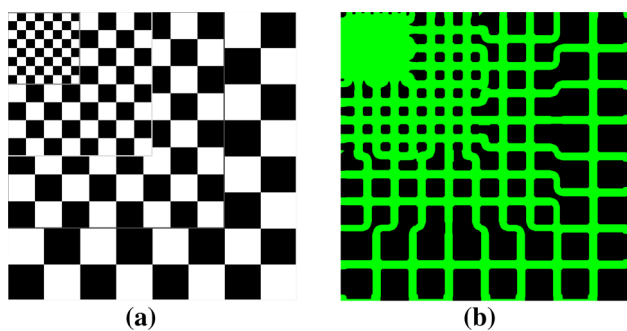


Fig. 15 A checkerboard-like pattern (a) and the result of the Δ operator applied to it (b)

with areas having differently sized squares (see Fig. 15a). Although the graphical results on such images are not surprising in general, we show in Fig. 15b the output from statistical cross-correlation, as it also illustrates how the operator works. Note that the “target” histogram (that of the whole image) mostly consists of an equal number of black and white points (plus a smaller number of points having an intermediate value, due to grey lines separating the different areas of the image). Therefore, the points that have high local cross-correlation with the whole image (depicted in green) are those that lay on the border of squares, whereas in the inner part of any square, the histogram only consists of either white or black points.

We have run our benchmarks on a laptop equipped with a 6th generation Core i7 CPU; Table 1 displays the image resolution, radius for cross-correlation, number of voxels, execution time in seconds, and memory used in kilobytes for each test. Execution times are as expected: the cross-correlation operator (benchmark “scmp”) is linear in the size of the model, as well as the Euclidean distance transform algorithm (“maurer”), and the surrounded operator (“maze”), whereas Chamfer distance, employing a variant of the Dijkstra shortest-path algorithm, has complexity $n \log n$, which is apparent when the size of the model grows by powers of two.

6 Medical image analysis with ImgQL

MR images are produced using different kinds of *sequences* of magnetic field gradients and radio-waves. Images so obtained are called *weighted* images; these can be further post-processed in various ways. For instance, typical weighted images are those produced using *Fluid-attenuated inversion recovery* pulse sequence (MR-FLAIR), *T2 weighted pulse sequence* (MR-T2w), or *diffusion-weighted images*, whereas the *Apparent Diffusion Coefficient* maps (ADC) are obtained via post-processing of diffusion-weighted images. A standard reference for such matters is [14]. In this section, we illustrate our approach on the seg-

Table 1 Benchmark results on some 2d synthetic images. Images have the same horizontal and vertical size

Benchmark	Resolution	Voxels	Time (s)	Memory (kb)
scmp	512 (r: 15)	262,144	12.01	20,156
scmp	512 (r: 20)	262,144	19.16	19,996
scmp	1024 (r: 5)	1,048,576	6.23	40,536
scmp	1024 (r: 10)	1,048,576	22.92	41,456
scmp	1024 (r: 15)	1,048,576	49.41	41,344
scmp	1024 (r: 20)	1,048,576	88.38	41,660
scmp	2048 (r: 5)	4,194,304	27.7	127,468
scmp	2048 (r: 10)	4,194,304	95.99	127,404
scmp	2048 (r: 15)	4,194,304	196.66	127,608
scmp	2048 (r: 20)	4,194,304	337.3	127,676
scmp	4096 (r: 5)	16,777,216	112.45	471,440
scmp	4096 (r: 10)	16,777,216	370.01	471,564
scmp	4096 (r: 15)	16,777,216	783.24	471,536
scmp	4096 (r: 20)	16,777,216	1352.89	438,656
maurer	512	262,144	1.66	22,808
maurer	1024	104,8576	6.74	49,824
maurer	2048	4,194,304	28.01	125,512
maurer	4096	16,777,216	113.3	477,044
chamfer	512	262,144	11.81	19,840
chamfer	1024	104,8576	83.81	41,464
chamfer	2048	4,194,304	627.68	125,612
chamfer	4096	16,777,216	5173.41	470,460

mentation of glioblastoma tumour and oedema in images obtained using MR-FLAIR. We first show the steps on 2D images and then present further results for 3D images.

For glioblastoma, our procedure was successfully tested on five images from different sources, that were acquired in very different conditions. However, validation of the methodology for actual clinical usage requires extensive clinical research. We refer to Sect. 6.2 for preliminary validation results and a more detailed discussion.

6.1 Example: segmentation of glioblastoma

In this example, we detail the specification of an analysis aimed at the segmentation of glioblastoma (GBM) and oedema in MR-FLAIR images. Being able to segment tumour and oedema in medical images can be of immediate use for *automatic contouring* applications in radiotherapy, and, in perspective, it can be helpful in detecting the invisible infiltrations in computer-aided diagnosis applications. The procedure is non-trivial, but every step is justified by morphological and spatial considerations on the arrangement of parts of the head and the brain.

Normal tissues of the head can be divided into several classes. The outer layer of the head consists of adipose tissue (and skin) surrounding the skull that in turn consists of bone and bone marrow. The skull encloses the brain tissues. The brain itself is suspended in cerebrospinal fluid (CSF) and isolated from the blood stream. Thresholds in the grey levels of images can be used to single out specific tissues in a medical image; however, in doing so, noise is generated in the form of (small, scattered) regions not belonging to the tissue. The relative positioning of tissues—the so-called *topological information* of the image—plays an important role in suppressing such noise. We will see in the following how such information is encoded by logic formulas in the methodology we propose.

GBMs are intracranial tumours composed of typically poorly marginated, diffusely infiltrating necrotic masses. Even if the tumour is totally resected, it usually recurs, either near the original site, or at more distant locations within the brain. GBMs are localised to the cerebral hemispheres and grow quickly to various sizes, from only a few centimetres, to lesions that cover a whole hemisphere. Infiltration beyond the visible tumour margin is always present. In MR-FLAIR images, GBMs appear *hyperintense* and surrounded by *vasogenic oedema*.⁹

Segmentation of GBM according to our method is performed in three steps:

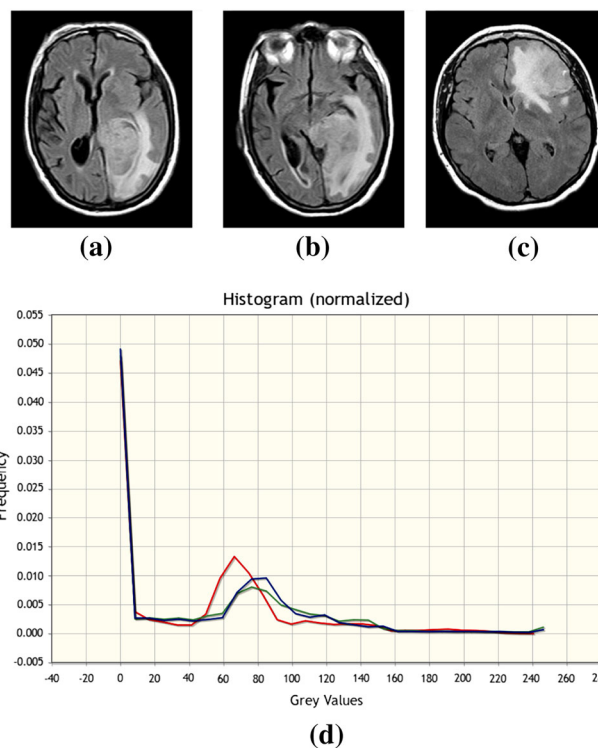


Fig. 16 **a** Slice of MR-FLAIR brain acquisition of a patient, Case courtesy of Dr. Ahmed Abd Rabou, Radiopaedia.org, rID: 22779; **b** a different slice of the acquisition in Fig. 16a; **c** slice of MR-FLAIR brain acquisition of a different patient—Case courtesy of A.Prof Frank Gaillard, Radiopaedia.org, rID: 5292; **d** histograms of Fig. 16a (in blue), of Fig. 16b (in green) and of Fig. 16c (in red) (colour figure online)

1. a preprocessing step (not using `topochecker`), aimed at normalisation of images, to make the choice of thresholds in our experiment applicable to different images;
2. brain segmentation, to limit the area of the image where the tumour is searched for;
3. tumour and oedema segmentation, which is the stated goal of this example.

6.1.1 Preprocessing

Histograms of grey levels of images¹⁰ of the same body part may differ from each other due to inter-patient or inter-scanner differences or depending on the actual acquisition volume (Fig. 16) or the file format used to store the image.¹¹

¹⁰ To ease visual comparison, in Figs. 16d, 17d, 18b, the histograms that we show are normalised so that the measure of the area below the curve is 1.

¹¹ For instance, *jpeg* images, as downloaded from *Radiopaedia.org*, typically use 8-bit precision (typical range 0–255) (see Fig. 16), whereas *dicom* images saved by scanners typically use 12 or 16-bit (for MR images, the typical range is 0–4096 or 0–65536, respectively) (see Fig. 18).

⁹ Vasogenic oedema is an abnormal accumulation of fluid from blood vessels, which is able to disrupt the blood–brain barrier and invade extracellular space.

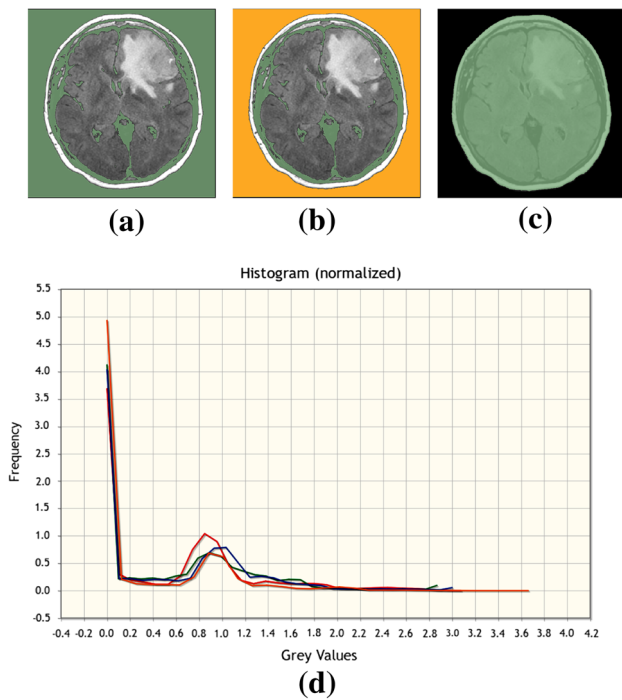


Fig. 17 Finding the mask for normalisation: **a** the pixels in Fig. 16c with grey levels below a given threshold are shown in green; **b** the sub-mask that touches the border of the image is shown in orange; **c** the mask of the image excluding the background is shown in green; **d** histograms of normalised version of images in Fig. 16a (in blue), in Fig. 16b (in green), in Fig. 16c (in red), and in Fig. 18a (in orange) (colour figure online)

More uniform results, on different images, can be obtained by dividing the intensity of each pixel by the average of the intensity levels of all the *significant* pixels in the image. A pixel is considered significant when it does not belong to the background. Significant pixels are selected using a Boolean mask (indicated by the green area in Fig. 17c). In order to compute such a mask, we start from the observation that the background (corresponding to the air surrounding the head of a patient) is darker than the rest of the image, so it mostly contributes to the initial part of its histogram. This situation is witnessed in the histogram by a peak close to 0. A threshold is thus selected for each image as the value immediately following such peak. Using this threshold, it is possible to isolate the background, by separating it from the head (Fig. 17a). Note that the obtained mask also includes cerebrospinal fluid (CSF) and bone. The part of the mask that touches the boundary of the whole image is then selected (Fig. 17b) and its complement, that is, the green area in Fig. 17c, is finally used to select the significant pixels to compute the mean value for normalisation. Figure 17d shows the histograms of images after normalisation.

We remark that *equalisation of histograms* is another form of normalisation, frequently used for texture analysis ([48]). We do not use this method as it changes the relationship

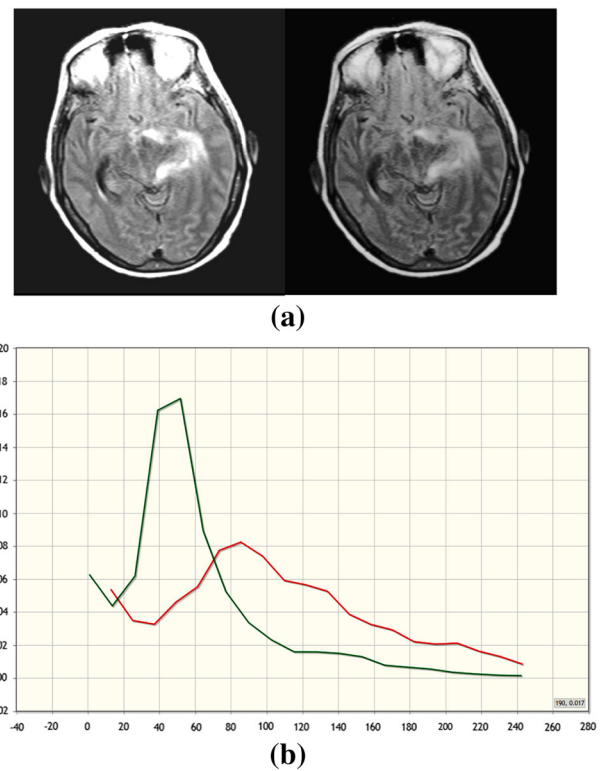


Fig. 18 Effect of histogram equalisation: **a** a slice of MR acquisition of brain, on the left, and its equalised version, on the right—case courtesy of Dr. Ahmed Abdrabou, Radiopaedia.org, rID: 39024; **b** histograms of grey levels of the original (green) and equalised (red) version of image in (a) (colour figure online)

between grey levels of different structures in the image (as shown in Fig. 18), that we use rather prominently for differentiating different tissues; normalisation of image intensity is sufficient for our purposes.

6.1.2 Brain segmentation

In this second phase of our method, we perform a segmentation of the brain in order to limit the search area of the tumour, by means of specific logic formulas. This improves the accuracy of the output (e.g. avoiding areas in bone marrow or skull) and reduces computing time.

In the process below, we fix some thresholds for identifying different tissues in the brain; note that, thanks to the preprocessing step described above, these can be kept uniform across different images.

Intuitively, the general model of a patient head that we use to segment the brain in MR-FLAIR images is defined as follows:

- Darker pixels in the head belong to CSF and bones;
- Brighter pixels belong both to adipose tissue surrounding the head, and to bone marrow;

- Also pixels belonging to the tumour and oedema are brighter than the surrounding tissues;
- The brain region is composed of white matter, grey matter, tumour and oedema;
- The brain (excluding the tumour) has intermediate intensities and is mainly surrounded by CSF.

For the segmentation of the brain, we use the normalised NIfTI image of the MR-FLAIR acquisition shown in Fig. 16c. In the rest of the analysis, the relevant attribute, i.e. the normalised grey level, of each pixel in this image is referred to as FLAIR.

It is convenient to define a few additional operators that serve as macros and that are specifically useful in the segmentation procedure that follows; for readability reasons, we use a prefix notation for these operators:

$$\begin{aligned} \text{grow}(\Phi_1, \Phi_2) &\triangleq (\Phi_1 \vee \Phi_2) \mathcal{S} \neg \Phi_2 \\ \text{denoise}(\Phi) &\triangleq \Phi \mathcal{T} \mathcal{D}^{\geq 2} \neg \Phi \\ \text{closeTo}(\Phi) &\triangleq \mathcal{D}^{< 3} \Phi \end{aligned}$$

Formula $\text{grow}(\Phi_1, \Phi_2)$ is inspired by the image segmentation method of *seeded region growing* [1]. This method starts from a number of *seed points* in the region of interest and examines neighbouring points to decide whether they should be added to the region. The definition of $\text{grow}(\Phi_1, \Phi_2)$ is a form of region growing, under the assumption that it is guaranteed that all points satisfying Φ_1 are also satisfying Φ_2 .

We start from points that satisfy property Φ_1 and to which all points satisfying property Φ_2 are added that, together with those satisfying Φ_1 , form a common region that is surrounded by points that do not satisfy Φ_2 .

Let A be the set of points satisfying formula Φ . Formula $\text{denoise}(\Phi)$ is used to remove small areas from A , as follows: first A is shrunk by 2 units; in doing so, some sub-areas of A may disappear; the areas that do not disappear are restored to their original shape by means of the touch operator. This operation is similar to $\mathcal{J}^{< 2}$, but it preserves the contours of the original area A . Formula $\text{closeTo}(\Phi)$ denotes the points that lay at a distance less than 3 units from the set of points satisfying Φ . For this analysis, the \mathcal{D} Chamfer distance operator uses 4 adjacent pixels per node and the distance units are in millimetres with respect to the actual dimension of the head, i.e. the *real-world* dimensions.

Next, we define a number of useful thresholds for the grey levels of the image that are used to obtain a first approximation of different kinds of tissue of interest:

$$\begin{aligned} \text{lowIntsty} &:= \text{FLAIR} < 0.5; \\ \text{medIntsty} &:= \text{FLAIR} > 0.5 \wedge \text{FLAIR} < 1.3; \\ \text{highIntsty} &:= \text{FLAIR} > 1.7; \\ \text{tumIntsty} &:= \text{FLAIR} > 1.17 \wedge \text{FLAIR} < 1.53; \\ \text{oedIntsty} &:= \text{FLAIR} \geq 1.47 \wedge \text{FLAIR} < 2.4; \end{aligned}$$

We distinguish three general levels of intensity (low, medium, and high), and two specific intensities that are typical for tumour and oedema, respectively.

We are now ready to start the segmentation procedure. First, we identify the points that are part of the background of the image. These all have a very low intensity, but there are other points in the image that have low intensity as well. What distinguishes the points of the background from the other low intensity points is that the background area touches the border of the image.

We assume that the special atomic proposition, named `border` is satisfied by the points that form the border of an image; this predicate is predefined in `topochecker`. This way points of the background are exactly those that satisfy the property `background`:

$$\text{background} \triangleq \text{lowIntsty} \mathcal{T} \text{border}$$

The points that satisfy `background` are shown in red in Fig. 19b. The original image is shown in Fig. 19a.

The next step is to look for the external border of the head, consisting of skin and adipose tissue. For our purposes, it is sufficient to identify the adipose tissue, since the brain is surrounded by the adipose tissue, which separates it from the skin.

Adipose tissue in the normalised MR-FLAIR images has intensity above 1.7, so of high intensity. As before, there may be other points with high intensity in the image, but we exploit the knowledge that adipose is at the external border of the head, and thus close to the background. These points can be found with the following formula:

$$\text{adipose} \triangleq \text{highIntsty} \mathcal{T} \text{closeTo}(\text{background})$$

The points that satisfy `adipose` are shown in green in Fig. 19b.

Using the properties `background` and `adipose`, it is not difficult to specify the points that are part of the head. These are all those points that are not part of the background or close to adipose tissue.

$$\text{head} \triangleq \neg(\text{closeTo}(\text{adipose}) \vee \text{background})$$

The points that satisfy `head` are the union of the green and red points in Fig. 19c (see below).

In the next steps, we show how we can distinguish the various tissues within the area of the head, namely the brain and the cerebrospinal fluid (CSF) that contains it. We start from the identification of points that are part of CSF. These are points that are within the head and that have low intensity:

$$\text{CSF} \triangleq \text{lowIntsty} \wedge \text{head}$$

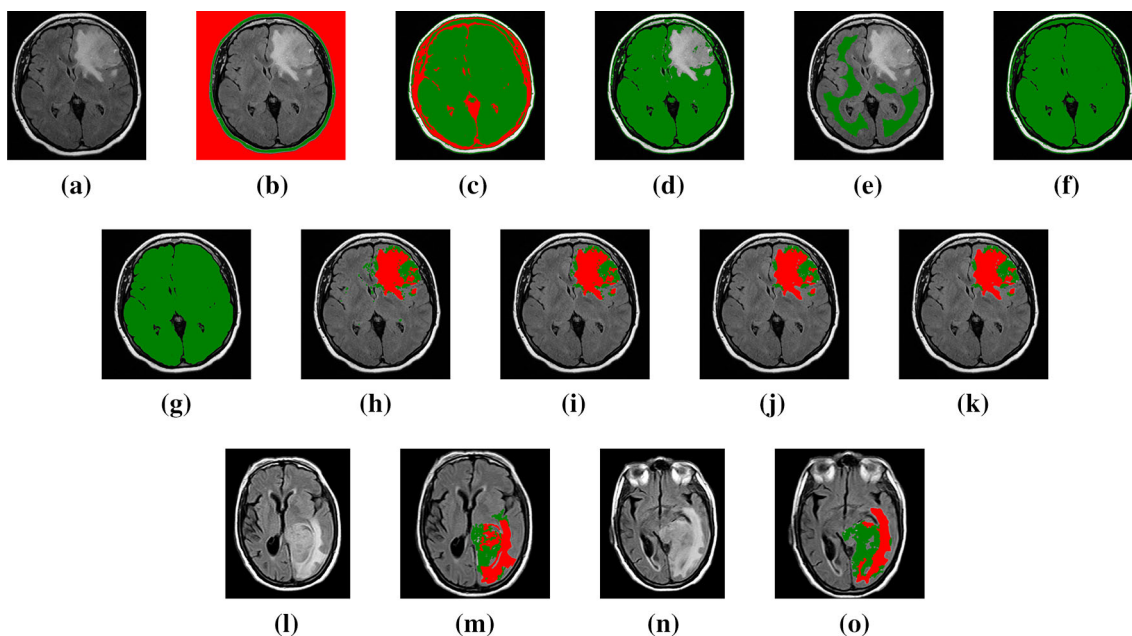


Fig. 19 Experimental results of using *topochecker* for segmentation of glioblastoma and oedema: **a** a slice of a brain acquisition of a patient—case courtesy of A.Prof Frank Gaillard, Radiopaedia.org, rID: 5292; **b** background (in red) and adipose tissue (in green); **c** output showing head (in red or green) and CSF (in red); **d** output of *brainApprox*; **e** output of *brainSeed*; **f** output of *noisyBrain*; **g** output of *brain*; **h** output of *tum0* (in green) and *oed0* (in red);

i output of *tum1* (in green) and *oed1* (in red); **j** output of *tum2* (in green) and *oed2* (in red); **k** output of *tumour* (in green) and *oedema* (in red) **l** a slice of a brain acquisition of another patient—case courtesy of Dr. Ahmed Abd Rabou, Radiopaedia.org, rID: 22779; **m** output of *tumour* (in green) and *oedema* (in red); **n** another slice from the second patient; **o** output of *tumour* (in green) and *oedema* (in red) (colour figure online)

The points that satisfy CSF are shown in red in Fig. 19c.

We proceed with segmentation of the brain in four subsequent steps. As a first approximation, we look for the points of the brain with medium intensity within the head (and that are not belonging to CSF). Within this approximation, we select some inner areas that are most certainly part of brain tissue and that can serve as a seed from which to ‘grow’ in a more precise way points belonging to the brain. Finally, we remove pieces that have been erroneously identified as part of the brain, but that are actually relatively small areas that are part of the skull or bone, having a similar intensity as that of the brain. This way we obtain all pixels that are actually part of the brain. The four steps of the specification are given below.

$$\begin{aligned}
 \text{brainApprox} &\triangleq \text{head} \wedge \neg \text{CSF} \wedge \text{medIntsty} \\
 \text{brainSeed} &\triangleq \mathcal{D}^{>10} \neg \text{brainApprox} \\
 \text{noisyBrain} &\triangleq \text{grow}(\text{brainSeed}, \text{head} \wedge \neg \text{CSF}) \\
 \text{brain} &\triangleq \text{noisyBrain} \mathcal{T} \text{brainSeed}
 \end{aligned}$$

The points that satisfy *brainApprox*, (*brainSeed*, and *noisyBrain*, respectively) are shown in green in Fig. 19d (Fig. 19e, f, respectively). The final result of the brain is shown in Fig. 19g.

6.1.3 GBM segmentation

In the final part of our analysis, we identify tumour and oedema regions. Since in MR-FLAIR, GBM and oedema are hyperintense areas, and the oedema is brighter than the tumour, we start by using the thresholds we introduced before that provide a rough segmentation of the image shown in Fig. 19a:

$$\begin{aligned}
 \text{tum0} &\triangleq \mathcal{J}^{<1}(\text{tumIntsty} \mathcal{S} (\text{brain} \vee \text{CSF})) \\
 \text{oed0} &\triangleq \mathcal{J}^{<1}(\text{oedIntsty} \mathcal{S} (\text{brain} \vee \text{CSF}))
 \end{aligned}$$

In Fig. 19h, we show in red the points that satisfy formula *oed0* referring to the oedema, and in green those satisfying formula *tum0* referring to the tumour. These are points that have the selected intensity (*oedIntsty* and *tumIntsty*, respectively) and are part of the brain tissue, i.e. they are surrounded by *brain* or *CSF*. Note that the regions *oed0* and *tum0* are partially overlapping. Moreover, we remove from these identified regions areas whose radius is smaller than 1mm using the $\mathcal{J}^{<1}$ operator.

An important constraint, that drastically reduces noise in the output of our analysis, is the *a priori* knowledge that oedema and tumour are very close to each other. We exploit this knowledge using the distance operator \mathcal{D} as follows:

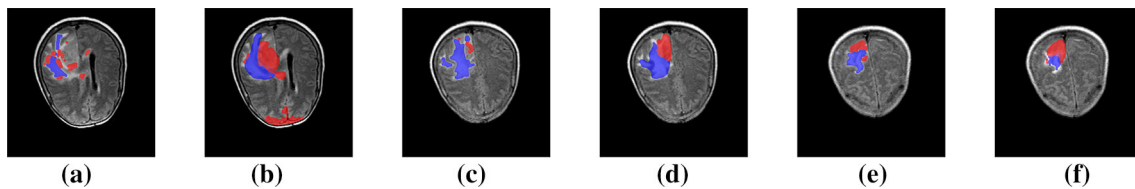


Fig. 20 Effects of using cross-correlation on a 3D volume. **a, c, e** target image for cross-correlation (glioblastoma in red, oedema in blue). **b, d, f** Output of the cross-correlation operator. Case courtesy of Dr. Ahmed Abdrabou, Radiopaedia.org, rID: 39024 (colour figure online)

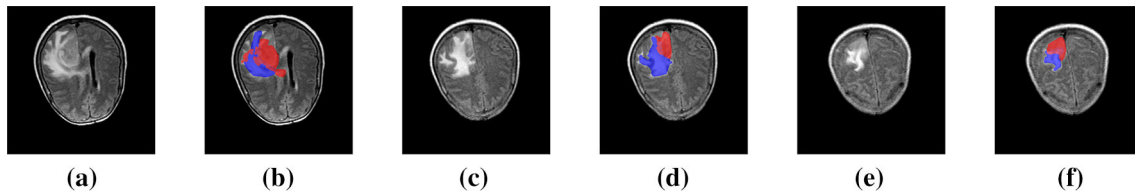


Fig. 21 Experimental results of using *topochecker* for segmentation of glioblastoma and oedema on a 3D volume. **a, c, e** Original slices. **b, d, f** Output of the segmentation of glioblastoma (in red) and oedema (in blue). Case courtesy of Dr. Ahmed Abdrabou, Radiopaedia.org, rID: 39024 (colour figure online)

$$\begin{aligned} \text{oeddst} &\triangleq \mathcal{D}^{\leq 2} \text{oed0} \\ \text{tum1} &\triangleq \text{tum0} \mathcal{T} \text{oeddst} \\ \text{oed1} &\triangleq \text{oed0} \wedge (\text{oeddst} \mathcal{R} \text{tum1}) \end{aligned}$$

We first define the region oeddst at distance less than 2mm from oed0 then select sub-regions of tum0 that *touch* oeddst (formula tum1) and sub-regions of oed0 that can reach points satisfying tum1 by passing only through points satisfying oeddst (formula oed1). The result is shown in Fig. 19i. Comparing the latter with Fig. 19h, we can observe that some green areas, located in the left half of the brain, have disappeared. These were points with a similar intensity as that of tumour tissue, but not actually part of it since they were not connected to the tumour. In this example, we used shortest-path distance as an approximation of Euclidean distance, for the sake of execution speed, as high accuracy for the distance is less important in this particular case.

$$\begin{aligned} \text{tum2} &\triangleq \text{denoise}(\text{tum1}) \\ \text{oed2} &\triangleq \text{denoise}(\text{oed1}) \end{aligned}$$

Figure 19j illustrates the areas characterised by tum2 (green) and oed2 (red). Compared to Fig. 19i, this removes a number of small detached areas of oedema that were located in the tumour area and should be considered as noise.

Finally, *tumour* and *oedema* are defined as being inter-reachable. This part could remove some separate areas that have tumour or oedema intensity but should not be considered as such since they are too far apart. In this particular case, no such areas were present apparently as can be observed comparing Fig. 19j with the final output of the segmentation in Fig. 19k.

$$\begin{aligned} \text{tumour} &\triangleq \text{tum2} \mathcal{T} \text{oed2} \\ \text{oedema} &\triangleq \text{oed2} \mathcal{T} \text{tum2} \end{aligned}$$

As a further result, in Fig. 19 we show the final segmentation of tumour and oedema on two other images from a different patient applying the same specification. Figure 19m shows the segmentation of the image in Figs. 16a, and 19o shows the one of the image in Fig. 16b. The original images are also shown aside of the result in Fig. 19 for more convenient comparison.

The whole analysis presented in this section has been carried out in 2D. The same approach also works in 3D, with minor modifications. Fig. 21 shows some slices of the segmentation of MR-FLAIR acquisition of the patient in Fig. 18a, using *topochecker* on the whole 3D volume image. Some minor modifications to the model checking session presented in this section were required; the most relevant one is that cross-correlation was used to enlarge the area of tum1 to $\text{tum1}'$ prior to using $\text{tum2} \triangleq \text{denoise}(\text{tum1}')$. See Fig. 20.

In general, there may be many different ways to obtain an accurate segmentation. Ideally, these should be robustly working for many different images, both in 2D and in 3D. In future work, we plan to investigate this in more detail and compare various variants from the point of view of robustness, accuracy and computational efficiency. Regarding the latter, the 2D analysis of GBM was performed on a standard laptop (equipped with an 6th generation Intel CORE i7 CPU, and 8 gigabytes of RAM) and performed in a little less than 1 minute, which as a first indication is in line with the current state of the art. However, the tool *topochecker* still does not scale well to 3D image analysis, as execution time for a 3D image (resolution: $512 \times 512 \times 1024$, about 6 million points) on the same machine takes about half an hour. We envisage

that new algorithms (mostly, for statistical cross-correlation) and the adoption of specialised imaging libraries—in place of the general-purpose graph-based approach which is currently implemented—may improve performance considerably.

6.2 Validation

The work presented in this section is aimed at providing an illustration of the analysis capabilities of our logic-based methodology, rather than providing complete clinical case studies. For instance, consider the glioblastoma specification, which is rather concise, consisting of a less than 30 lines long logical specification, and a simple preprocessing step. Although such procedure was successfully tested on five images from different sources and acquired in very different conditions, this is certainly not sufficient to validate our example as a glioblastoma segmentation methodology for future clinical usage. Future work aims at improving the method, eliminating corner cases in the formulas as much as possible, making it robust to different acquisition conditions and properly validating it. More generally speaking, clinical experimentation is the next step in our research programme.

However, some conclusions can already be drawn from the data we have, both with respect to efficiency and to accuracy of the obtained results.

Analysis time is proportional to the size of the image (the algorithm is linear). In the glioblastoma example, MR-FLAIR very often has a slice size of 256×256 pixels, multiplied by 20–30 slices. As a rough estimate, the execution time for the analysis of a single 1024×1024 -voxels slice—including preprocessing—on a standard laptop (equipped with an Intel CORE i7 CPU, and 8 gigabytes of RAM) currently stays below one minute. This information, although not being a fully fledged benchmark, provides a first indication that, efficiency-wise, our approach is in par with the state of the art in semi-automatic glioblastoma segmentation procedures (see for example [38]). We remark that our procedure makes use of a prototype general-purpose model checker, that could be amenable to further optimisation, e.g. employing specialised, well-known flood-filling algorithms for images for model checking the *surrounded* connective—in place of the current graph-theoretical method.

A preliminary assessment of the quality of the obtained results in the case of glioblastoma was performed for a patient of the A.O.U.S. university hospital. The patient underwent first surgery and then radiotherapy. We compared our results on the post-surgery MR-FLAIR with target volumes delineated on the pre-treatment Computed Tomography (CT) by one experienced radiotherapist. In particular, we considered the *gross tumour volume* (GTV), i.e. what can be seen or imaged, and the *clinical target volume* (CTV), which contains the GTV, plus a margin for sub-clinical disease spread which therefore cannot be fully imaged [15]. Usually for

Table 2 Score of comparison with manual segmentation (from [11])

	Mean	SD	Median	Range
Dice (%)	87	2	87	78–89
HD (mm)	3.50	1.20	3.18	2.47–5.45

glioblastomas, the CTV is defined as a 2–2.5-cm isotropic expansion of GTV within the brain. In order to quantify the effectiveness of our segmentation, we computed the *Dice coefficient* (DC), that we used to measure the morphological similarity between the manual segmentation MS and automatic segmentation AS . The coefficient is defined as $DC = \frac{2V(MS \cap AS)}{V(MS) + V(AS)}$, where $V(a)$ is the volume of a , that is, the number of voxels that belong a ; DC ranges from 0 to 1; 0 indicates no overlap, and 1 indicates complete overlap. The CT volume was co-registered to the FLAIR volume. Then, we considered the region R obtained as the union of the oedema and tumour regions, as found using our method. We compared R to the GTV contour, and furthermore we compared R , expanded by 2.5cm (as explained above) to the CTV contour. We obtained $DC = 0.76$ for GTV and $DC = 0.81$ for CTV. Although a single case does not have clinical significance, these results are very encouraging, and aligned with state-of-the-art methods for automatic and semi-automatic segmentation of glioblastoma [35]. In [11], a variant of the method we described was assessed on a dataset of 7 patients affected by GBM, that have undergone radiotherapy. Segmentation results were evaluated using the Dice coefficient and the *percent (0.95) Hausdorff distance* (HD) between obtained segmentation volumes and the CTV manually delineated by an expert radiotherapist for radiotherapy planning. The results are shown in Table 2. The evaluation was performed on a midrange portable with an intel i7 CPU and 8 GB of RAM, the same machine used for the other experiments in the current paper. The average execution time per patient (3D image size: $256 \times 256 \times 40$) was 10 minutes.

The state of the art in automated glioblastoma segmentation is represented by the Brain Tumor Image Segmentation Benchmark (BraTS) [60]. The top results in most recent editions, as already mentioned, are almost exclusively based on machine learning algorithms. The BraTS challenge is aimed at segmentation of the GTV, as opposed to the CTV that was studied in [11]. A direct comparison is therefore not possible. Future work will be directed to clinical studies of procedures formalised using spatial logics, including (possibly refined) versions of the one we presented, and to comparison with existing benchmarks and methods for each type of study.

Finally, we note that numeric thresholds and other parameters (e.g. the number of nested N constructs in some formulas, the number of bins for statistical analysis, etc.) have been chosen by the medical physicist in charge of the analysis,

on the basis of expert knowledge on the matter and in some cases by trial-and-error. The values that we used might prove stable in clinical validation (and this is the purpose of the preliminary normalisation of images that we use), but this is not yet to be taken for granted, or even to be expected in more general situations. Instead, *parameter calibration* on a per-image or per-study basis will be an important subject in our future research. Such calibration may be fully automatic (e.g. through *machine learning* techniques), but this is just one possibility. It would also make sense to adopt a semi-automatic approach (which is also frequent in state-of-the-art techniques, see e.g. [35,38,72,80]), involving human interaction with an expert to merely calibrate the parameters, rather than performing a full manual segmentation, in order to save a large part of the time (and costs) required for preparation to radiotherapy or surgery.

7 Conclusions and future work

This work provides a first, promising exploration of logical methods for declarative medical image analysis in the domain of radiotherapy. A declarative approach makes analysis transparent, reproducible, human-readable, and exchangeable, and permits domain experts who are not technicians to understand the specifications. Such advantages are akin to those obtained in other domains, such as the application of the *Structured Query Language* (SQL) in the field of databases, or the introduction of query languages (XPath, XSLT, ...) in semi-structured data management.

Logical properties are used as classifiers for points of an image; this can be used both for colouring regions that may be similar to diseased tissues, and therefore being diseased tissue in turn, and for colouring regions corresponding to organs of the human body. Envisaged applications range from *contouring* to *computer-aided diagnosis*. Our logic *ImgQL* is able to predicate on both shortest-path and Euclidean distance at the same time, and *topochecker* implements both operators. In MI, shortest-path distances proved useful so far mostly to speed up interactive development; this is implementation dependent, as the Modified Dijkstra transform that we use (see Sect. 3.2) currently performs faster than Maurer distance transform in our tests. We also considered the embedding of specific operators for MI in *ImgQL* such as an operator for texture analysis based on first-order statistical methods. Other options and operators could be considered following a similar approach, providing a way to include state-of-the-art analysis techniques that can be conveniently combined using the spatial operators of the logic.

It is noteworthy that the analysis we designed for glioblastoma segmentation can be used, with mild modifications, also to analyse the whole 3D volume of an image at once. 3D analysis is a relatively new application in medical imaging,

leveraging the precision/efficiency trade-off of more classical methods. Furthermore, 3D analysis may be combined with existing applications of *3D printing* in preparation for surgery (see [68]), by providing practitioners with models of a patient's body, with the relevant regions, identified by our method, printed in different colours. Such aspects constitute a further interesting line of research for future work.

Part of our ongoing work consists in identifying novel logical operators that are useful in medical imaging. So far, we only used operators that classify individual pixels or voxels. However, drawing inspiration from the family of *region calculi* (see [3]), one could also classify regions, taking advantage of “collective” observations on sets of voxels that belong to the same area. Some work in this direction is [23], including the definition of operators related to connectedness of regions; further work will be directed to the investigation of properties related to the size of regions, or to their morphological properties. Also, the “distance-bounded surrounded” operator defined in [64] could be useful in medical imaging. A limitation of the model checking algorithm in [64] is its quadratic complexity. We have shown that the application of distance transforms yields a *linear* algorithm for a weaker variant of the bounded surrounded operator for the case of images (that is, regular grids).

We recall that *topochecker* is a spatio-temporal model checker. Temporal reasoning could be exploited in future work to consider, for instance, the sequence of acquisitions of a patient in order to reason about the evolution of image features such as tumours, which is very important in radiotherapy applications.

Our experiments show that typical analyses carried out using spatial model checking in medical imaging require careful calibration of numeric parameters (for example, a threshold for the distance between a tumour and the associated oedema, or the size of areas identified by a formula, that are small enough to be considered *noise*, and ought to be filtered out). The calibration of such parameters might be performed using machine learning techniques. In this respect, future work could be focused on the application, in the context of our research line, of the methodology used in the development of the logic *SpaTeL*, aimed at signal analysis (see [6,7,42,45]), that pursues *machine learning* of the logical structure of image features. We emphasise that such a development, if implemented, would be a radical improvement in application of machine learning to medical imaging. It can be framed under the recent research trend on *explainable artificial intelligence*, as it would yield a procedure that can explain in terms of a human-readable language the methodology that a machine learning algorithm extrapolates from data. Our topological approach to spatial logics would be a key enabling technique for this purpose, as the formulas obtained in the *SpaTeL* approach are not meant to be intelligible by humans. It is worth noting that machine learning and

deep learning methods have also been applied to the detection of tumours in very recent literature [4,74]. On the other hand, our application of machine learning could as well be focused simply on the identification of numeric parameters, rather than logic formulas, that may depend on complex features of images.

The example of glioblastoma that we illustrated in Sect. 6.1 has immediate practical relevance. As we already mentioned, improvement and validation of the procedure are in progress; see for instance [10] for a revised version of the tool including validation of glioblastoma segmentation on a larger dataset. The normalisation step that we employ could be improved using state-of-the-art methods (see [54,57], and the references therein).

Planned future developments also include means for interactive refinement of analysis, based on visual fine-tuning of specific values (e.g. thresholds or distances) that may have a *non-linear* effect on the results of complex queries, with significant impact on methods that require human interaction—e.g. *interactive segmentation* in preparation for surgery, or *contouring* for radiotherapy planning.

Acknowledgements The authors wish to thank Dr. Marco Di Benedetto for suggesting the application of distance transforms to reduce the complexity of model checking of formulas with distances, and Dr. Valerio Nardone for contributing to the preliminary assessment of our experimental results.

References

- Adams, R., Bischof, L.: Seeded region growing. *IEEE Trans. Pattern Anal. Mach. Intell.* **16**, 641–647 (1994)
- Aiello, M.: *Spatial Reasoning: Theory and Practice*. Ph.D. thesis, Institute of Logic, Language and Computation, University of Amsterdam (2002)
- Aiello, M., Pratt-Hartmann, I., van Benthem, J.: *Handbook of Spatial Logics*. Springer, New York (2007)
- Akkus, Z., Galimzianova, A., Hoogi, A., Rubin, D.L., Erickson, B.J.: Deep learning for brain mri segmentation: state of the art and future directions. *J. Digit. Imaging* **30**, 449–459 (2017)
- Bartocci, E., Bortolussi, L., Loret, M., Nenzi, L.: Monitoring mobile and spatially distributed cyber-physical systems. In: Talpin, J.P., Derler, P., Schneider, K. (eds.) *Proceedings of the 15th ACM-IEEE International Conference on Formal Methods and Models for System Design, MEMOCODE 2017, Vienna, Austria, September 29–October 02, 2017*, pp. 146–155. ACM, New York (2017)
- Bartocci, E., Bortolussi, L., Milios, D., Nenzi, L., Sanguinetti, G.: Studying emergent behaviours in morphogenesis using signal spatio-temporal logic. In: *Hybrid Systems Biology: Fourth International Workshop, HSB 2015, Madrid, Spain, September 4–5, 2015. Revised Selected Papers*, pp. 156–172. Springer (2015)
- Bartocci, E., Gol, E.A., Haghghi, I., Belta, C.: A formal methods approach to pattern recognition and synthesis in reaction diffusion networks. *IEEE Trans. Control Netw. Syst.* **5**(1), 308–320 (2018)
- Bauer, S., Wiest, R., Nolte, L.-P., Reyes, M.: A survey of MRI-based medical image analysis for brain tumor studies. *Phys. Med. Biol.* **58**(13), R97 (2013)
- Belmonte, G., Ciancia, V., Latella, D., Massink, M.: From collective adaptive systems to human centric computation and back: spatial model checking for medical imaging. In: ter Beek, M., Loret, M. (eds.) *Proceedings of the Workshop on Formal Methods for the Quantitative Evaluation of Collective Adaptive Systems, FORECAST@STAF 2016, Vienna, Austria, 8 July 2016, Volume 217 of EPTCS*, pp. 81–92 (2016)
- Belmonte, G., Ciancia, V., Latella, D., Massink, M.: Voxlogica: a spatial model checker for declarative image analysis. In: *Tools and Algorithms for the Construction and Analysis of Systems—25th International Conference, TACAS 2019, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS, Logical Methods in Computer Science. Springer (to appear)* (2019)
- Belmonte, G., Ciancia, V., Latella, D., Massink, M., Biondi, M., De Otto, G., Nardone, V., Rubino, G., Vanzi, E., Banci Buonamici, F.: A topological method for automatic segmentation of glioblastoma in MR air for radio-therapy—ESMRMB 2017, 34th annual scientific meeting. *Magn. Reson. Mater. Phys. Biol. Med.* **30**(S1), 437 (2017)
- Brock, K.: *Image Processing in Radiation Therapy*. CRC Press, Boca Raton (2014)
- Brown, L.: A survey of image registration techniques. *ACM Comput. Surv.* **24**(4), 325–376 (1992)
- Brown, R., Cheng, N., Haacke, E., Thompson, M., Venkatesan, R. (eds.): *Magnetic Resonance Imaging*. Wiley, New York (2014)
- Burnet, N.: Defining the tumour and target volumes for radiotherapy. *Cancer Imaging* **4**(2), 153–161 (2004)
- Castellano, G., Bonilha, L., Li, L., Cendes, F.: Texture analysis of medical images. *Clin. Radiol.* **59**(12), 1061–1069 (2004)
- Chen, C., Da Ponte, J., Fox, M.: Fractal feature analysis and classification in medical imaging. *IEEE Trans. Med. Imaging* **8**(2), 133–142 (1989)
- Chetelat, G., Baron, J.: Early diagnosis of alzheimer’s disease: contribution of structural neuroimaging. *NeuroImage* **18**(2), 525–541 (2003)
- Ciancia, V., Gilmore, S., Grilletti, G., Latella, D., Loret, M., Massink, M.: Spatio-temporal model checking of vehicular movement in public transport systems. *Int. J. Softw. Tools Technol. Transf.* **20**(3), 289–311 (2018)
- Ciancia, V., Gilmore, S., Latella, D., Loret, M., Massink, M.: Data verification for collective adaptive systems: spatial model-checking of vehicle location data. In: *Eighth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASOW*, pp. 32–37. IEEE Computer Society (2014)
- Ciancia, V., Grilletti, G., Latella, D., Loret, M., and Massink, M.: An experimental spatio-temporal model checker. In: *Software Engineering and Formal Methods—SEFM 2015 Collocated Workshops, Volume 9509 of Lecture Notes in Computer Science*, pp. 297–311. Springer (2015)
- Ciancia, V., Latella, D., Loret, M., Massink, M.: Specifying and verifying properties of space. In: *Theoretical Computer Science—8th IFIP TC 1/WG 2.2 International Conference, TCS 2014, Rome, Italy, September 1–3, 2014. Proceedings, Volume 8705 of Lecture Notes in Computer Science*, pp. 222–235. Springer (2014)
- Ciancia, V., Latella, D., Loret, M., Massink, M.: Model checking spatial logics for closure spaces. *Log. Methods Comput. Sci.* **12**(4), 1–51 (2016)
- Ciancia, V., Latella, D., Loret, M., Massink, M.: Spatial logic and spatial model checking for closure spaces. In: Bernardo, M., De Nicola, R., Hillston, J. (eds.) *Formal Methods for the Quantitative Evaluation of Collective Adaptive Systems—16th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2016, Bertinoro, Italy, June 20–24, 2016, Advanced Lectures, Volume 9700 of Lecture Notes in Computer Science*, pp. 156–201. Springer, New York (2016)

25. Ciancia, V., Latella, D., Massink, M., Pakauskas, R.: Exploring spatio-temporal properties of bike-sharing systems. In: 2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASO Workshops, pp. 74–79. IEEE Computer Society (2015)
26. Ciancia, V., Latella, D., Massink, M., Paskauskas, R., Vandin, A.: A tool-chain for statistical spatio-temporal model checking of bike sharing systems. In: Margaria, T., Steffen, B. (eds.) Leveraging Applications of Formal Methods, Verification and Validation: Foundational Techniques—7th International Symposium, ISoLA 2016, Imperial, Corfu, Greece, October 10–14, 2016, Proceedings, Part I, Volume 9952 of Lecture Notes in Computer Science, pp. 657–673 (2016)
27. Ciesielski, K., Chen, X., Udupa, J., Grevera, G.: Linear time algorithms for exact distance transform. *J. Math. Imaging Vis.* **39**(3), 193–209 (2010)
28. Clarke, E., Emerson, A.: Design and synthesis of synchronization skeletons using branching-time temporal logic. In: Logic of Programs, Workshop, pp. 52–71. Springer, London (1982)
29. Clarke, E., Grumberg, O., Peled, D.: Model Checking. MIT Press, Boca Raton (1999)
30. Davnall, F., Yip, C., Ljungqvist, G., Selmi, M., Ng, F., Sanghera, B., Ganeshan, B., Miles, K.A., Cook, G.J., Goh, V.: Assessment of tumor heterogeneity: an emerging imaging tool for clinical practice? *Insights Imaging* **3**(6), 573–589 (2012)
31. De Nicola, R., Katoen, J., Latella, D., Loreti, M., Massink, M.: Model checking mobile stochastic logic. *Theor. Comput. Sci.* **382**(1), 42–70 (2007)
32. De Santis, S., Drakesmith, M., Bells, S., Assaf, Y., Jones, D.: Why diffusion tensor MRI does well only some of the time: variance and covariance of white matter tissue microstructure attributes in the living human brain. *NeuroImage* **89**, 35–44 (2014)
33. Despotović, I., Goossens, B., Philips, W.: MRI segmentation of the human brain: challenges, methods, and applications. *Comput. Math. Methods Med.* **1–23**, 2015 (2015)
34. Doi, K.: Computer-aided diagnosis in medical imaging: historical review, current status and future potential. *Comput. Med. Imaging Graph.* **31**(4–5), 198–211 (2007)
35. Dupont, C., Betrouni, N., Reyns, Vermandel, M.: On image segmentation methods applied to glioblastoma: state of art and new trends. *IRBM* **37**(3), 131–143 (2016)
36. Fabbri, R., Costa, L., Torelli, J., Odemir, B.: 2D Euclidean distance transform algorithms: a comparative survey. *ACM Comput. Surv.* **40**(1), 2:1–2:44 (2008)
37. Fabbri, R., Da Fontoura Da Costa, L., Torelli, J., Bruno, O.: 2D Euclidean distance transform algorithms: a comparative survey. *ACM Comput. Surv.* **40**(1), 2:1–2:44 (2008)
38. Fyllingen, E., Stensjøen, A., Berntsen, E., Solheim, O., Reinertsen, I.: Glioblastoma segmentation: comparison of three different software packages. *PLoS ONE* **11**(10), e0164891 (2016)
39. Galton, A.: The mereotopology of discrete space. In: Freksa, C., David, M. (eds.) Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science, Volume 1661 of Lecture Notes in Computer Science, pp. 251–266. Springer, Berlin (1999)
40. Galton, A.: A generalized topological view of motion in discrete space. *Theor. Comput. Sci.* **305**(1–3), 111–134 (2003)
41. Galton, A.: Discrete mereotopology. In: Calosi, C., Graziani, P. (eds.) Mereology and the Sciences: Parts and Wholes in the Contemporary Scientific Context, pp. 293–321. Springer, New York (2014)
42. Gol, E., Bartocci, E., Belta, C.: A formal methods approach to pattern synthesis in reaction diffusion systems. In: 53rd IEEE Conference on Decision and Control, pp. 108–113 (2014)
43. Gordillo, N., Montseny, E., Sobrevilla, E.: State of the art survey on MRI brain tumor segmentation. *Magn. Reson. Imaging* **31**(8), 1426–1438 (2013)
44. Grevera, G.: Distance transform algorithms and their implementation and evaluation. In: Farag, A.A., Suri, J.S. (eds.) Deformable Models, pp. 33–60. Springer (2007)
45. Grosu, R., Smolka, S., Corradini, F., Wasilewska, A., Entcheva, E., Bartocci, E.: Learning and detecting emergent behavior in networks of cardiac myocytes. *Commun. ACM* **52**(3), 97–105 (2009)
46. Haghghi, I., Jones, A., Kong, Z., Bartocci, E., Grosu, R., Belta, C.: Spatel: A novel spatial-temporal logic and its applications to networked systems. In: Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control, HSCC '15, pp. 189–198. ACM, New York (2015)
47. Han, F., Wang, H., Zhang, G., Han, H., Song, B., Li, L., Moore, W., Lu, H., Zhao, H., Liang, Z.: Texture feature analysis for computer-aided diagnosis on pulmonary nodules. *J. Digit. Imaging* **28**(1), 99–115 (2014)
48. Haralick, R., et al.: Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **3**(6), 610–621 (1973)
49. Heinonen, T., Arola, T., Kalliokoski, A., Dastidar, P., Rossi, M., Soimakallio, S., Hyttinen, J., Eskola, H.: Computer aided diagnosis tool for the segmentation and texture analysis of medical images. In: IFMBE Proceedings, pp. 274–276. Springer (2009)
50. Kassner, A., Thornhill, R.: Texture analysis: a review of neurologic MR imaging applications. *Am. J. Neuroradiol.* **31**(5), 809–816 (2010)
51. Kimmel, R., Kiryati, N., Bruckstein, A.: Sub-pixel distance maps and weighted distance transforms. *J. Math. Imaging Vis.* **6**(2), 223–233 (1996)
52. Kontchakov, R., Kurucz, A., Wolter, F., Zakharyashev, M.: Spatial logic + temporal logic = ? In: Aiello, M., Pratt-Hartmann, I., van Benthem, J. (eds.) Handbook of Spatial Logics, pp. 497–564. Springer (2007)
53. Kutz, O., Wolter, F., Sturm, H., Suzuki, N., Zakharyashev, M.: Logics of metric spaces. *ACM Trans. Comput. Log.* **4**(2), 260–294 (2003)
54. Lemieux, L., Hagemann, G., Krakow, K., Woermann, F.: Fast, accurate, and reproducible automatic segmentation of the brain in T1-weighted volume MRI data. *Magn. Reson. Med.* **42**(1), 127–135 (1999)
55. Li, C., Herndon, J., Novembre, F., Zhang, X.: A longitudinal magnetization transfer imaging evaluation of brain injury in a macaque model of NeuroAIDS. *AIDS Res. Hum. Retrovir.* **31**(3), 335–341 (2015)
56. Lopes, R., Ayache, A., Makni, N., Puech, P., Villers, A., Mordon, S., Betrouni, N.: Prostate cancer characterization on MR images using fractal features. *Med. Phys.* **38**(1), 83 (2011)
57. Madabhushi, A., Udupa, J.: New methods of MR image intensity standardization via generalized scale. *Med. Phys.* **33**(9), 3426–3434 (2006)
58. Maurer, C., Rensheng, Q., Raghavan, V.: A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions. *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(2), 265–270 (2003)
59. Mazzara, G., Velthuisen, R., Pearlman, J., Greenberg, H., Wagner, H.: Brain tumor target volume determination for radiation treatment planning through automated MRI segmentation. *Int. J. Radiat. Oncol. Biol. Phys.* **59**(1), 300–312 (2004)
60. Menze, B., et al.: The multimodal brain tumor image segmentation benchmark (brats). *IEEE Trans. Med. Imaging* **34**(10), 1993–2024 (2015)
61. Meshgi, K., Ishii, S.: Expanding histogram of colors with gridding to improve tracking accuracy. In: MVA (2015)

62. Mohan, G., Subashini, M.M.: Mri based medical image analysis: survey on brain tumor grade classification. *Biomed. Signal Process. Control* **39**, 139–161 (2018)
63. Nenzi, L., Bortolussi, L.: Specifying and monitoring properties of stochastic spatio-temporal systems in signal temporal logic. In: 8th International Conference on Performance Evaluation Methodologies and Tools, VALUETOOLS 2014, Bratislava, Slovakia, December 9–11, 2014. ICST (2014)
64. Nenzi, L., Bortolussi, L., Ciancia, V., Loreti, M., Massink, M.: Qualitative and quantitative monitoring of spatio-temporal properties. In: Runtime Verification—6th International Conference, RV 2015 Vienna, Austria, September 22–25, 2015. Proceedings, Volume 9333 of Lecture Notes in Computer Science, pp. 21–37. Springer (2015)
65. Nenzi, L., Bortolussi, L., Ciancia, V., Loreti, M., Massink, M.: Qualitative and quantitative monitoring of spatio-temporal properties with SSTL. *Log. Methods Comput. Sci.* **14**(4), 1–38 (2018). [https://doi.org/10.23638/LMCS-14\(4:2\)2018](https://doi.org/10.23638/LMCS-14(4:2)2018)
66. Prvu, O., Gilbert, D.: A novel method to verify multilevel computational models of biological systems using multiscale spatio-temporal meta model checking. *PLoS ONE* **11**(5), 1–43 (2016). 05
67. Reif, J., Sistla, A.: A multiprocess network logic with temporal and spatial modalities. *J. Comput. Syst. Sci.* **30**(1), 41–53 (1985)
68. Rengier, F., Mehndiratta, A., von Tengg-Kobligk, H., Zechmann, C.M., Unterhinninghofen, R., Kauczor, H.-U., Giesel, F.L.: 3D printing based on imaging data: review of medical applications. *Int. J. Comput. Assist. Radiol. Surg.* **5**(4), 335–341 (2010)
69. Rodriguez Gutierrez, D., Awwad, A., Meijer, L., Manita, M., Jaspán, T., Dineen, R., Grundy, R., Auer, D.: Metrics and textural features of MRI diffusion to improve classification of pediatric posterior fossa tumors. *Am. J. Neuroradiol.* **35**(5), 1009–1015 (2013)
70. Sharma, N., Ray, A., Sharma, S., Shukla, K., Pradhan, S., Aggarwal, L.: Segmentation and classification of medical images using texture-primitive features: application of BAM-type artificial neural network. *J. Med. Phys.* **33**(3), 119 (2008)
71. Sheremet, M., Wolter, F., Zakharyashev, M.: A modal logic framework for reasoning about comparative distances and topology. *Ann. Pure Appl. Log.* **161**(4), 534–559 (2010)
72. Simi, V., Joseph, J.: Segmentation of glioblastoma multiforme from MR images—a comprehensive review. *Egypt. J. Radiol. Nucl. Med.* **46**(4), 1105–1110 (2015)
73. Srinivasan, G., Shobha, G.: Statistical texture analysis. *Proc. World Acad. Sci. Eng. Technol.* **36**, 1264–1269 (2012)
74. Sundstrom, A., Grabocka, E., Bar-Sagi, D., Mishra, B.: Histological image processing features induce a quantitative characterization of chronic tumor hypoxia. *PLoS ONE* **11**(4), 1–30 (2016). 04
75. Thorup, M.: Undirected single-source shortest paths with positive integer weights in linear time. *J. ACM* **46**(3), 362–394 (1999)
76. Tijms, B., Series, P., Willshaw, D., Lawrie, S.: Similarity-based extraction of individual networks from gray matter MRI scans. *Cereb. Cortex* **22**(7), 1530–1541 (2011)
77. Toosy, A.: Diffusion tensor imaging detects corticospinal tract involvement at multiple levels in amyotrophic lateral sclerosis. *J. Neurol. Neurosurg. Psychiatry* **74**(9), 1250–1257 (2003)
78. van Benthem, J., Bezhanishvili, G.: Modal logics of space. In: Aiello, M., Pratt-Hartmann, I., van Benthem, J. (eds.) *Handbook of Spatial Logics*, pp. 217–298. Springer (2007)
79. Woods, B., Clymer, B., Kurc, T., Heverhagen, J., Stevens, R., Orsdemir, A., Bulan, O., Knopp, M.: Malignant-lesion segmentation using 4D co-occurrence texture analysis applied to dynamic contrast-enhanced magnetic resonance breast image data. *J. Magn. Reson. Imaging* **25**(3), 495–501 (2007)
80. Zhu, Y., Young, G., Xue, Z., Huang, R., You, H., Setayesh, K., Hatabu, H., Cao, F., Wong, S.: Semi-automatic segmentation software for quantitative clinical brain glioblastoma evaluation. *Acad. Radiol.* **19**(8), 977–985 (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.