



Innovating Medical Image Analysis via Spatial Logics

Gina Belmonte¹, Vincenzo Ciancia², Diego Latella², and Mieke Massink²(✉)

¹ Azienda Ospedaliera Universitaria Senese, Siena, Italy

² Consiglio Nazionale delle Ricerche - Istituto di Scienza e Tecnologie
dell'Informazione 'A. Faedo', CNR, Pisa, Italy

Mieke.Massink@isti.cnr.it

Abstract. Current computer-assisted medical imaging for the planning of radiotherapy requires high-level mathematical and computational skills. These are often paired with the case-by-case integration of highly specialised technologies. The lack of modularity at the right level of abstraction in this field hinders research, collaboration and transfer of expertise among medical physicists, engineers and technicians. The longer term aim of the introduction of spatial logics and spatial model checking in medical imaging is to provide an open platform introducing *declarative medical image analysis*. This will provide domain experts with a convenient and very concise way to specify contouring and segmentation operations, grounded on the solid mathematical foundations of Topological Spatial Logics. We show preliminary results, obtained using the spatial model checker **VoxLogicA**, for the automatic identification of specific brain tissues in a healthy brain and we discuss a selection of challenges for spatial model checking for medical imaging.

Keywords: Spatial logics · Closure Spaces ·
Spatial model checking · Medical imaging

1 Introduction

Spatial and Spatio-temporal logics and model checking are enjoying an increasing interest in Computer Science (see for instance [12, 13, 16, 25, 26, 35]). The main idea of spatial and spatio-temporal model checking is to use specifications written in logical languages to describe spatial properties and to automatically identify patterns and structures of interest. Spatial and spatio-temporal model checking have recently been applied in a variety of domains, ranging from Collective Adaptive Systems [10, 17, 18] to signals [35] and images [4, 13, 26], just to mention a few. The origins of spatial logics can be traced back to the forties of the previous century when McKinsey and Tarski recognised the possibility of reasoning on space using topology as a mathematical framework for the interpretation of modal logic (see [9] for a thorough introduction). In their work, modal logic formulas are interpreted as sets of points of a topological space. In particular, in



Fig. 1. Examples: open ball (left) and its topological closure (right)

that context, the modal operator \diamond is interpreted as the (logical representation of the) topological *closure* operator. Informally, this operator adds an (infinitely thin) border to an open set of points as illustrated in Fig. 1.

In recent work [12, 13], Ciancia et al. pushed such theoretical developments further to encompass arbitrary graphs as models of space. In that work *Closure spaces*, a generalisation of topological spaces, are used as underlying model for *discrete spatial logic* inspired by recent work by Galton [21–23]. This resulted in the definition of the *Spatial Logic for Closure Spaces* (SLCS), and a related model checking algorithm. Furthermore, in [11], a spatio-temporal logic, combining *Computation Tree Logic* with the spatial operators of SLCS was introduced. An (extended) model checking algorithm has been implemented in the prototype *spatio-temporal model checker topochecker*¹.

A completely different and, so far, little explored domain of application for spatial model checking is that of medical imaging. Medical imaging is concerned with the creation of visual representations of parts of the human body for the purpose of clinical analysis and in preparation of medical intervention. In our recent work [4, 6–8] we focused in particular on spatial model checking in the area of medical imaging for radiotherapy. One of the most important steps in the planning of radiotherapy is the accurate contouring of tissues and organs at risk in medical images, commonly produced by Computed Tomography (CT), Magnetic Resonance (MR), and Positron Emission Tomography (PET). Recent research efforts in the field of medical imaging are therefore focused on the introduction of automatic contouring procedures. These procedures are used to identify particular kinds of tissues. These can be for example parts of the brain (white matter², grey matter³) or tissues that could indicate diseases that need treatment. Such (semi-) automatic procedures would lead to an increase in accuracy and a considerable reduction in time and costs, compared to manual contouring – the current practice in most hospitals. The software for automatic contouring that is starting to appear on the market is, however, highly specialised for particular types of diseased tissue in particular parts of the body (e.g., “breast cancer”, or “glioblastoma” – a kind of malign tumour in the brain), lacks transparency to its users, provides little flexibility, and its accuracy is still not always satisfactory. In the last few years also deep learning algorithms have become very popular for medical image analysis. They are reaching good results and are computationally efficient, but they are also posing their own limiting factors such as lack of

¹ Topochecker: a topological model checker, see <http://topochecker.isti.cnr.it>, <https://github.com/vincenzoml/topochecker>.

² Part of the central nervous system in the brain.

³ Place where neurons are located in the outer part of the brain.

sufficiently large accurately labelled data sets, labelling uncertainty and problems to deal with rare cases (see for example a recent survey [29] and references therein) but also lack of explainability and transparency. Our recent work shows that, when comparing the accurate contouring of brain tumour tissue using a spatial model checking approach [8] with the best performing algorithms (among which many based on deep learning) on the public benchmark data set for brain tumours (BraTS 2017 [38]), our approach on 3D images is well in line with the state of the art, both in terms of accuracy and in terms of computational efficiency.

The work in the present paper is focusing on the identification of relevant tissues in the *healthy* brain such as white matter and grey matter rather than diseased tissue. As in our previous work, we do this using `VoxLogicA`, (*Voxel-based Logical Analyser*)⁴ the free and open source spatial model checker described in [8] which efficiently implements the spatial logic SLCS enriched with a number of specific operators for the domain of medical imaging that were introduced in [4, 8]. Furthermore, we provide a selection of challenges laying ahead for the use of spatial model checking in medical imaging as a valuable complementary method in this important area of research.

In Sect. 2, we briefly recall the spatial logic framework and some of the main aspects of spatial model checking based on Closure Spaces, and provide a number of illustrative examples that serve as a gentle introduction to the spatial logic. Section 3 illustrates further operators that are of particular interest in Medical Imaging. In Sect. 4 we show how these specific operators can be combined with the basic logic to identify tissues of interest in a healthy brain. In Sect. 5 we describe some of the main challenges for successful application of spatial model checking in the area of medical imaging for radiotherapy. Related work is described in Sect. 6. In Sect. 7 we provide some conclusions and an outlook for further research.

2 The Spatial Logic Framework

A 2D digital image can be modelled as an *adjacency space*, i.e. a set X of *cells* or *points*—each corresponding to a distinct *pixel*—together with an *adjacency relation* R among points. Usually, the so called *orthogonal* adjacency relation⁵ is used, where only pixels which share an edge count as adjacent; on the other hand, in the *ortho-diagonal* adjacency relation (see Fig. 2) pixels are adjacent as long as they share at least either an edge or a corner. Each pixel of an image is associated with one or more (colour) *intensities*; we model this by equipping the points with *attributes*. We assume sets A and V of attribute *names* and *values*, and an *attribute valuation* function \mathcal{A} such that $\mathcal{A}(x, a) \in V$ is the value of attribute a of point x . Attributes can be used in *assertions* α , i.e. boolean expressions, with standard syntax and semantics. Consequently, we abstract from related

⁴ `VoxLogicA`: <https://github.com/vincenzoml/VoxLogicA>.

⁵ Sometimes called von Neumann adjacency. The relation is reflexive and symmetric.

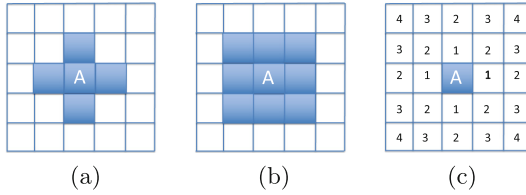


Fig. 2. Pixels that are orthogonally adjacent to pixel A (a) and orthodiagonally adjacent (b) are shown in blue. Distance transform (c) with distance to pixel A as attribute shown in each pixel for orthogonal adjacency and Manhattan distance function. (Color figure online)

details here and assume function \mathcal{A} extended in the obvious way; for instance, $\mathcal{A}(x, a \leq c) = \mathcal{A}(x, a) \leq c$, for appropriate constant c .

A similar reasoning applies to 3D—or, in general, multi-dimensional—images, where *voxels* are used instead of pixels and the (chosen) adjacency relation needs to be extended in the obvious way (an extended introduction to these matters is given in [4]).

Given a set of (attributed) points X with a binary relation $R \subseteq X \times X$ we define function $\mathcal{C}_R : 2^X \rightarrow 2^X$ with $\mathcal{C}_R(Y) \triangleq Y \cup \{x \mid \exists y \in Y. y R x\}$. It turns out that \mathcal{C}_R is a *closure function* and (X, \mathcal{C}_R) is a *closure space*⁶. Thus, adjacency spaces are a subclass of closure spaces.

A (quasi-discrete) *path* π in (X, \mathcal{C}_R) is a function $\pi : \mathbb{N} \rightarrow X$, such that for all $Y \subseteq \mathbb{N}$, $\pi(\mathcal{C}_{Succ}(Y)) \subseteq \mathcal{C}_R(\pi(Y))$, where π is implicitly lifted to sets in the usual way (i.e. $\pi(Y) = \{x \mid \exists y \in Y. \pi(y)\}$) and $(\mathbb{N}, \mathcal{C}_{Succ})$ is the closure space of natural numbers with the *successor* relation: $(n, m) \in Succ \Leftrightarrow m = n + 1$. Informally: the ordering in the path imposed by \mathbb{N} is compatible with relation R , i.e. if $\pi(i) \neq \pi(i + 1)$ then $\pi(i) R \pi(i + 1)$ ⁷.

A closure space (X, \mathcal{C}) can be enriched with a notion of *distance*, i.e. a function $d : X \times X \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$ such that $d(x, y) = 0$ iff $x = y$, leading to the *distance closure space* $((X, \mathcal{C}), d)$. The notion is easily lifted to sets $Y \neq \emptyset$: $d(x, Y) \triangleq \inf\{d(x, y) \mid y \in Y\}$, with $d(x, \emptyset) = \infty$.

In this paper, we use the version of the logic presented in [8], based on a reachability operator, as in [5], and recalled in the sequel. For given set P of *atomic predicates* p , and interval of \mathbb{R} I , the *syntax* of the logic is given below:

$$\Phi ::= p \mid \neg \Phi \mid \Phi \vee \Phi \mid \mathcal{N}\Phi \mid \rho \Phi[\Phi] \mid \mathcal{D}^I \Phi \tag{1}$$

⁶ The reader interested in the formal definition of closure spaces and on their properties is referred to the literature (see e.g. [12, 13, 21–23] and references therein). Here it suffices to say that $\mathcal{C}(Y)$ is essentially the set of points *close* to any point in Y ; note that, since closure spaces generalize topological spaces, in the latter, the closure operator \mathcal{C} coincides with topological closure, so that, for instance, in the monodimensional Euclidean space \mathbb{R} , $\mathcal{C}([0, 1)) = \mathcal{C}((0, 1)) = \mathcal{C}((0, 1]) = \mathcal{C}([0, 1]) = [1, 0]$.

⁷ We refer to [13] for a discussion on paths on the more general class of closure spaces, including e.g. Euclidean spaces, including e.g. Euclidean spaces.

Satisfaction $\mathcal{M}, x \models \Phi$ of a formula Φ at point $x \in X$ in *distance closure model* $\mathcal{M} = ((X, \mathcal{C}), d, \mathcal{A}, \mathcal{V})$ is defined in Fig. 3 by induction on the structure of formulas. It is assumed that space is modelled by the set of points of a distance closure model; each atomic predicate $p \in P$ models a specific *feature* of *points* and is thus associated with the points that have this feature. A point x satisfies $\mathcal{N}\Phi$ if it belongs to the closure of the set of points satisfying Φ , i.e. if x is *near* (or *close*) to a point satisfying Φ ; x satisfies $\rho\Phi_2[\Phi_1]$ if there is a path π rooted in x —i.e. with $x = \pi(0)$ —and an index ℓ such that $\pi(\ell)$ satisfies Φ_2 —i.e. $\mathcal{M}, \pi(\ell) \models \Phi_2$ —and all intermediate points in π , if any, satisfy Φ_1 —i.e. $\mathcal{M}, \pi(j) \models \Phi_1$, for all j with $0 < j < \ell$; x satisfies $\mathcal{D}^I\Phi$ if the distance of x from the set of points satisfying Φ falls in interval I ; in the sequel we will use standard abbreviations for denoting intervals I of interest as parameter of \mathcal{D} , such as: ‘ $< r$ ’ for $[0, r)$ and ‘ $\geq r$ ’ for $[r, \infty)$. Finally, the logic includes logical negation (\neg) and disjunction (\vee); as usual, the true (\top) and false (\perp) constants as well as conjunction (\wedge) are defined as derived operators.

$$\begin{aligned}
 \mathcal{M}, x \models p \in P & \Leftrightarrow x \in \mathcal{V}(p) \\
 \mathcal{M}, x \models \neg\Phi & \Leftrightarrow \mathcal{M}, x \models \Phi \text{ does not hold} \\
 \mathcal{M}, x \models \Phi_1 \vee \Phi_2 & \Leftrightarrow \mathcal{M}, x \models \Phi_1 \text{ or } \mathcal{M}, x \models \Phi_2 \\
 \mathcal{M}, x \models \mathcal{N}\Phi & \Leftrightarrow x \in \mathcal{C}(\{y \mid \mathcal{M}, y \models \Phi\}) \\
 \mathcal{M}, x \models \rho\Phi_2[\Phi_1] & \Leftrightarrow \text{there exists a path } \pi \text{ and an index } \ell \text{ such that the following holds:} \\
 & \quad \pi(0) = x \text{ and } \mathcal{M}, \pi(\ell) \models \Phi_2 \text{ and} \\
 & \quad \mathcal{M}, \pi(j) \models \Phi_1, \text{ for all } j \text{ with } 0 < j < \ell \\
 \mathcal{M}, x \models \mathcal{D}^I\Phi & \Leftrightarrow d(x, \{y \mid \mathcal{M}, y \models \Phi\}) \in I
 \end{aligned}$$

where, 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. Definition of the satisfaction relation

We provide a few simple examples to illustrate these basic spatial operators in Fig. 4. The examples are shown for a spatial model based on a 2D space of 100 points arranged as a 10×10 grid, with an orthogonal adjacency relation. We assume the set of atomic predicates P is the set $\{black, white, red\}$ and, in Fig. 4a, we show in black the points satisfying the atomic predicate *black* and similarly for *white* and *red*. In Fig. 4b the points satisfying formula $black \vee red$ are shown in green⁸; similarly, Fig. 4c shows the points satisfying $\neg(black \vee red)$, and Fig. 4d shows those satisfying $\mathcal{N}black$; all points of this model satisfy $\rho red[white]$, as shown in Fig. 4e while only the points satisfying *black* in the model satisfy also $black \wedge \rho red[white]$, as shown in Fig. 4f. Finally, Fig. 4g shows in green the points that satisfy $\mathcal{D}^{[2,3]}red$, i.e. those points that are at a distance of at least 2 and at most 3 from points satisfying *red* in Fig. 4a. In this case we assume that the underlying notion of distance is that of the Manhattan distance as shown in Fig. 2.

⁸ Note that this colour does *not* correspond to any atomic predicate and so it is not part of the model; we use it only for illustration purposes.

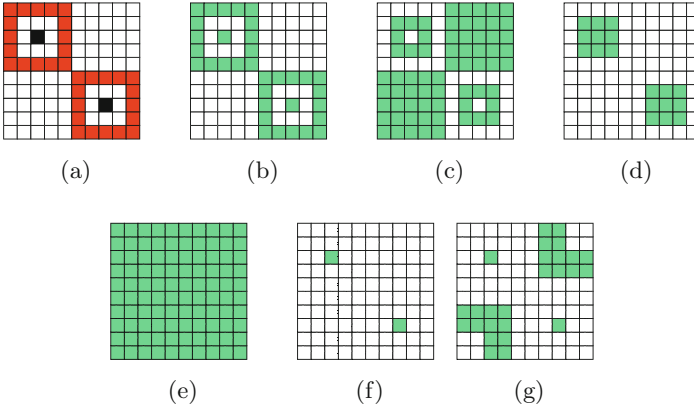


Fig. 4. An example model (a); the points shown in green are those satisfying $black \vee red$ (b), $\neg(black \vee red)$ (c), $Nblack$ (d), $\rho red[white]$ (e), $black \wedge \rho red[white]$ (f), and $\mathcal{D}^{[2,3]}red$ (g). (Color figure online)

In the version of the logic presented in [12,13] a *surrounded* operator \mathcal{S} was introduced for closure spaces, inspired by the *spatial until* operator discussed in [1] for topological spaces; x satisfies $\Phi_1 \mathcal{S} \Phi_2$ if it satisfies Φ_1 and in any path π with $x = \pi(0)$, if there is ℓ such that $\pi(\ell)$ does *not* satisfy Φ_1 , then there is j , $0 < j \leq \ell$, such that $\pi(j)$ 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 satisfying Φ_2 , i.e. x is *surrounded* by Φ_2 . In [8] it has been shown that the *surrounded* operator can be expressed using the *reaches* operator ρ as follows:

$$\Phi_1 \mathcal{S} \Phi_2 \equiv \Phi_1 \wedge \neg \rho (\neg(\Phi_1 \vee \Phi_2))[\neg \Phi_2]$$

In this paper \mathcal{S} will be considered as a derived operator. Again with reference to Fig. 4a we note that the two black points also satisfy $black \mathcal{S} (Nred)$.

3 Spatial Logic for Image Analysis

In this section we illustrate the use of the variant of SLCS briefly presented in Sect. 2 extended with a few additional operators introduced in [4,7,8], that are of particular interest for the domain of image analysis. In earlier work we focused on the contouring of diseased (brain) tissue [4,8]. In the next section we show how short but formal and unambiguous logic specifications can be used to identify typical parts of the human brain.

Before presenting the specifications, it is convenient to introduce a few additional derived operators, defined in Fig. 5.

Let us consider a point of a model which satisfies a formula of the form $\rho \Phi_2[\Phi_1]$; from the definition of the *reaches* operator, there is no guarantee that

$$\begin{aligned}
 \text{touch}(\Phi_1, \Phi_2) &\triangleq \Phi_1 \wedge \rho \Phi_2[\Phi_1] \\
 \text{grow}(\Phi_1, \Phi_2) &\triangleq \Phi_1 \vee \text{touch}(\Phi_2, \Phi_1) \\
 \text{flt}(r, \Phi_1) &\triangleq \mathcal{D}^{<r}(\mathcal{D}^{\geq r}\neg\Phi_1)
 \end{aligned}$$

Fig. 5. Definition of the *touch*, *grow* and *flt* derived operators.

such a point would also satisfy Φ_1 , i.e. the formula satisfied by the intermediate points of the path, if any. Such guarantee is ensured by the derived operator *touch*—i.e. a point satisfies $\text{touch}(\Phi_1, \Phi_2)$ if it satisfies Φ_1 and there is a path rooted in this point that reaches a point satisfying Φ_2 with all preceding points satisfying Φ_1 ; note that all such preceding points satisfy $\text{touch}(\Phi_1, \Phi_2)$ too. Figure 4c shows in green the points satisfying $\text{touch}(\text{white}, \text{red})$ in the model of Fig. 4a—that, by the way, in this specific model, happen to be the same as those satisfying $\neg(\text{black} \vee \text{red})$.

The formula $\text{grow}(\Phi_1, \Phi_2)$ is satisfied by points that satisfy Φ_1 and by points that satisfy Φ_2 and that are on a path that reaches a point satisfying Φ_1 . Figure 6a shows in green the points satisfying $\text{grow}(\text{red}, \text{white})$ in the model of Fig. 4a.

A point satisfies formula $\text{flt}(r, \Phi_1)$ if it is at a distance of less than r from the set of points that are at a distance at least r from the set of points that do *not* satisfy Φ_1 . This operator works as a filter; only contiguous areas satisfying Φ_1 that have a minimal diameter of at least $2r$ are preserved; these are also smoothed if they have an irregular shape (e.g. protrusions of less than the indicated distance). An example of the effect of the *flt* operator is shown in Figs. 6b to e. Let us consider the model of Fig. 6b—defined on only two atomic predicates, namely *black* and *white*—and let us consider the formula $\text{flt}(2, \text{black})$; Fig. 6c shows in green the points of the model of Fig. 6b satisfying $\neg\text{black}$, while those satisfying $\mathcal{D}^{\geq 2}(\neg\text{black})$ are shown in Fig. 6d and finally Fig. 6e shows those satisfying $\mathcal{D}^{<2}(\mathcal{D}^{\geq 2}(\neg\text{black}))$ —i.e. $\text{flt}(2, \text{black})$.

Furthermore a statistical similarity operator is introduced (see [4, 8]). It can search for tissue that has the same statistical texture characteristics as a provided texture sample by comparing the similarity of the histograms of the two textures. With reference to a point x , the statistical similarity operator $\Delta_{\triangleright\triangleleft c} \left[\begin{smallmatrix} m & M & k \\ r & a & b \end{smallmatrix} \right] \Phi$ compares the region of the image constituted by the sphere (hypercube) of radius r centred in x against the 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 range ($[m, M]$) and the same bins ($[1, k]$). In summary, the operator allows to check *to which extent the sphere (hypercube) around the point of interest is statistically similar to a given region (specified by) Φ* . This implements a form of *texture similarity*, which, in practice, works quite well for medical images, also since it is by definition invariant with respect to rotation. In Fig. 7 we report an example that was used in [4] as a benchmark. The benchmark uses a checkerboard-like pattern with areas having differently-sized squares (see Fig. 7a). Figure 7b shows the output of statistical cross-correlation—after thresholding—using as “target”

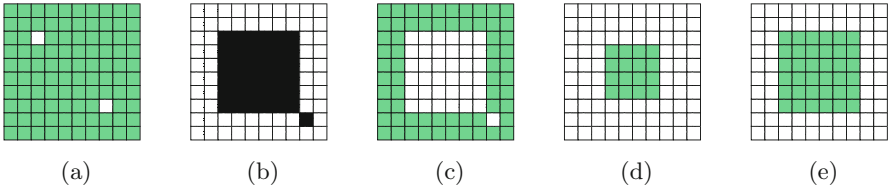


Fig. 6. With reference to the model in Fig. 4a, (a) shows in green the points satisfying $grow(red, white)$. In (c) (d, e, respectively) the points of the model shown in (b) that satisfy $\neg black (\mathcal{D}^{\geq 2}(\neg black), \mathcal{D}^{< 2}(\mathcal{D}^{\geq 2}(\neg black)))$ —i.e. $fit(2, black)$ —respectively are shown in green. (Color figure online)

region the whole image. The associated histogram 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.

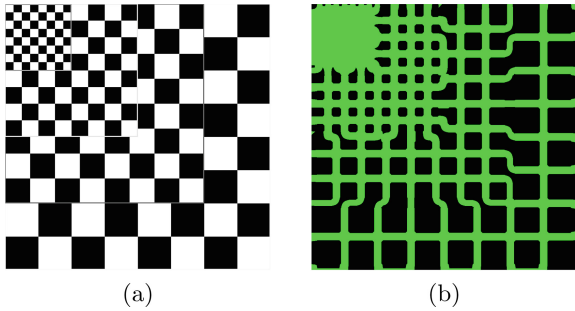


Fig. 7. A checkerboard-like pattern (a) and the result of the Δ operator applied to it (b). (Color figure online)

The *maxvol* operator is another operator introduced for the domain of image analysis. A point satisfies $maxvol \Phi$ if it belongs to the *largest* connected component of (the subspace induced by the) points that satisfy Φ . If there are more than one of such largest components, then the points of all such largest components satisfy the property $maxvol \Phi$.

Finally, a *percentiles* operator is introduced that assigns to each point of an image the percentile rank of its intensity among those that are part of the image. The interpretation of $percentiles(img, mask, c)$ considers the set of points S identified by the Boolean-valued mask *mask* in the image *img*—an image with an intensity attribute value associated to each point—and returns an image in

which the percentile attribute value of each voxel x is the fraction of points in S that have an intensity *below* that of x in *img*; c is a weight, between 0 and 1, used to take into account also the fraction of points that have intensity *equal* to that of x in the computation; more precisely, the percentile value v_x of each voxel x is defined by

$$v_x = \frac{l_x + (c \cdot e_x)}{N}$$

where l_x is the number of voxels in S having intensity *below* that of x , e_x is the number of voxels in S that have intensity *equal to* that of x , and N is the total number of voxels in S . This operator is used as a form of normalisation of the provided image so that the specification can be used on different cases without the need to recalibrate or explicitly normalise the intensities of the image which may differ in a similar way as normal photographs may show some over or under-exposition. To clarify this, let us just mention one example: the *hyperintense* areas of an image can be defined as those that have percentile rank higher than 0.95, no matter what is the range of the intensity of the source image, or even its numeric type and precision.

The spatial logic SLCS and the additional operators discussed in this section have been implemented in the free open source spatial model checker `VoxLogicA`. `VoxLogicA` is specifically designed for the analysis of (possibly multi-dimensional, e.g. 3D) digital images as a specialised image analysis tool, though it can also be used for 2D (general purpose) image analysis. It is tailored to usability and efficiency by employing state-of-the-art algorithms and open source libraries, borrowed from computational image processing, in combination with efficient spatial model checking algorithms. The source code and binaries of `VoxLogicA` as well as an exhaustive list of the available built-ins, a user manual and a mini-tutorial for the tool are available at the web site of the tool (see Footnote 4). Furthermore, a “standard library” is provided containing short-hands for commonly used functions, and for derived operators. For further details on the model checker and its implementation we refer to [8].

4 Illustration: Brain Segmentation

In previous work [4,7,8] we have focused on how the spatial model checker `VoxLogicA` can be used for contouring of brain tumours and associated oedema. In this section we present preliminary results on the identification of specific tissues in the healthy head and brain such as white and grey matter, the skull, the bone marrow and so on. For this purpose we use two simulated brain images⁹ from a set of twenty [2,28]. Simulated brain images have the advantage that data is generated and therefore the ‘ground truth’ is known, i.e. it is known for sure which points belong to which kind of tissue. This is very useful for the quantitative testing of image analysis methods.

⁹ See https://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_normal_20.html (Publicly available).

The specifications that we present serve the purpose to illustrate the flexibility and the potential of the approach, although we expect that our method will be further improved to obtain more accurate results in future work.

The syntax we use for the specifications is that of `VoxLogicA`, namely: `|`, `&`, `!` are boolean *or*, *and*, *not*; `distleq(c,phi)` is the formula $\mathcal{D}^{\leq c}\text{phi}$ (similarly, `distgeq`; distances are in millimeters); the *statistical similarity* operator (see Sect. 6) $\Delta_{\bowtie c} \begin{bmatrix} m & M & k \\ r & a & b \end{bmatrix} \Phi$ is written as `crossCorrelation(r,a,b,phi,m,M,k) \bowtie c`; where the function `crossCorrelation` computes the relevant cross-correlation value. The `>`. and `<`. operators (greater than, and less than, respectively) perform thresholding of the attribute values of the points of an image; `border` is true on voxels that lay at the border of the image. Other operators should be self-explaining or will be explained in passing.

Specification 1: Derived operators in `VoxLogicA`

```

1 import "stdlib.imgql"
2 let grow(a,b) = (a | touch(b,a))
3 let flt(r,a) = distleq(r,distgeq(r,!a))
4 load imgT1 = "INPUTDIR/NAME_t1.nii.gz"
5 let t1 = intensity(imgT1)
6 let similarT1To(a) = crossCorrelation(3,t1,t1,a,min(t1),max(t1),30)
7 let similarT1For1(a) =
  crossCorrelation(1,t1,t1,a,min(t1),max(t1),30)

```

In the first part of the specification, shown in Specification 1, standard derived operators are imported from the file `stdlib.imgql` (they will be explained when they are used) and the derived operators that were presented in Sect. 3 are defined, i.e. the operators `grow` and `flt`.

In line 4 the 3D magnetic resonance image (MRI) is loaded (which in this case is of type T1, short for T1-weighted-Fluid-Attenuated Inversion Recovery). In this case, the file is encoded using the NIfTI file format (`.nii` file name extension)¹⁰. In line 5 the name `t1` is bound to the attribute of each voxel of the image corresponding to the intensity of that voxel in the image. In line 6 and 7 two variants of the statistical similarity operator `crossCorrelation` are defined that share the same parameters, but use a different radius, 3 mm and 1 mm, respectively, around each point.

The second part of the specification is given in Specification 2, where the operations for the identification of the head and the background in the image are shown. Figure 8 shows a few intermediate results of Specification 2 that make it easier to follow the informal description below. Such figures across this section have been produced using the `save` command of the tool, which, when applied to a formula, generates and saves to disk a new binary image, where exactly those points satisfying the formula at hand are rendered using the boolean value *true*.

¹⁰ The NIfTI file format is a special data format by the Neuro-imaging Informatics Technology Initiative, <https://nifti.nimh.nih.gov/>.

Specification 2: Segmentation of head and background

```

1 let bg = percentiles(t1,t1 >. 0,0.5)
2 let bg1 = touch(bg <. 0.6,border)
3 let head1 = maxvol(flt(2,!bg1))
4 let head2 = distleq(3,head1)
5 let bg2 = maxvol(!head2)
6 let background = distleq(3,bg2)
7 let head=!background

```

Such a binary image (also called *region of interest*) can be loaded in viewers to produce coloured overlays that are superimposed to the original image. In line 1 `bg` (short for ‘background’) is defined where each point in the image is associated with the value of its percentile ranking attribute w.r.t. the intensity of the points. In line 2 this is used to identify all points from which a border point can be reached passing only through relatively dark (low intensity) points that satisfy `bg < .0.6`, shown in red in Fig. 8b. Note that the formula is satisfied also by points inside the skull; this is due to the fact that the image under analysis is a 3D image, so there are (3D) paths which do not lay in the 2D projection shown in the figure. In line 3 this is used to obtain `head1`, a first approximation of the area of the head as the maximum volume that is not identified as background—after some smoothening using the filter operator (Fig. 8c). In line 4 also all points at a distance of less than 3 mm from points satisfying `head1` are included (Fig. 8d). This temporary enlargement of the head area is useful to separate points that are part of the background from those that are part of the head, so that in line 5 the operator `maxvol` can be used to identify all points of the background (in red in Fig. 8e). After this, the temporary enlargement is removed using the `distleq` operator in line 6 (in blue in Fig. 8f) and the points that satisfied it are again part of the background.

The separation of the background from the part of the image that is really of interest, namely the head, is important for the next steps. In particular, the percentiles can now be obtained considering only the area of the head. Since the anatomy of any adult human head is very similar, the number of points in each percentile, i.e. the normalised grey-level, have a very similar distribution for any head regardless of the possible fluctuations in the luminosity of the images due to differences in the registration. Therefore, in line 1 of Specification 3 the percentile rank for each point satisfying `head` is obtained in `pt1`. In line 2 the similarity coefficient (cross correlation) is computed (for each voxel) and in line 3 we define the interior part of the head, i.e. those points of the head that are at a distance of at least 30 mm from the background (Fig. 9a). The following steps use these definitions to identify the white matter of the brain. In line 4 we seek to obtain an area of the brain that is certainly composed of white matter and is defined as the largest area in the inner part of the brain each point of which is in a 3 mm ball that has a cross correlation coefficient with the complete head of between 0.4 and 0.6 (Fig. 9b). These values have been obtained experimentally

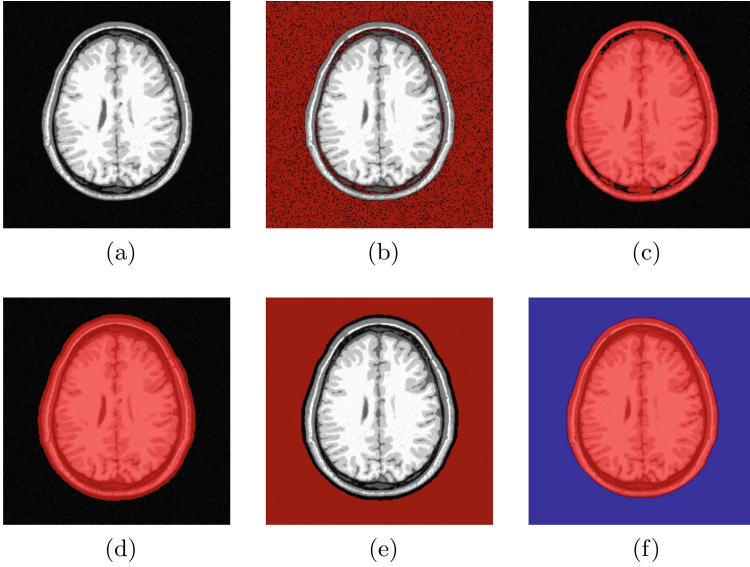


Fig. 8. Identification of head and background. Original axial view (a). In red points satisfying `bg1` (b); points satisfying `head1` (c); points satisfying `head2` (d); points satisfying `bg2` (e); points satisfying `head` (in red) and `background` (in blue) (f). (Color figure online)

in such a way that they correspond to points that are white matter and not other kind of tissue, such as the eyes, that also has a high intensity¹¹. Then, in line 5, we look for all the points that are similar to `white1` using a rather small radius (1mm) to obtain good accuracy (Fig. 9c). Subsequently, in line 6 the largest volume of points is selected that is sufficiently similar with a coefficient of at least 0.6 (Fig. 9d). This gives already a very good approximation. In line 7 this is somewhat refined by “closing tiny grey holes” in this area using a surrounded operator (Fig. 9e).

Specification 3: Segmentation of white matter

```

1 let pt1 = percentiles(t1,head,0.5)
2 let headSim = similarT1To(head)
3 let headInt = head & !(distleq(30,!head))
4 let white1 = maxvol((headSim <. 0.6) & (headSim >. 0.4) & headInt)
5 let whiteT1 = similarT1Tor1(white1)
6 let white2 = maxvol(whiteT1 >. 0.6)
7 let white = white2 | ((headSim >. 0.3) & surrounded((headSim >.
  0.3),white2))

```

¹¹ Again, note that we are processing a 3D image.

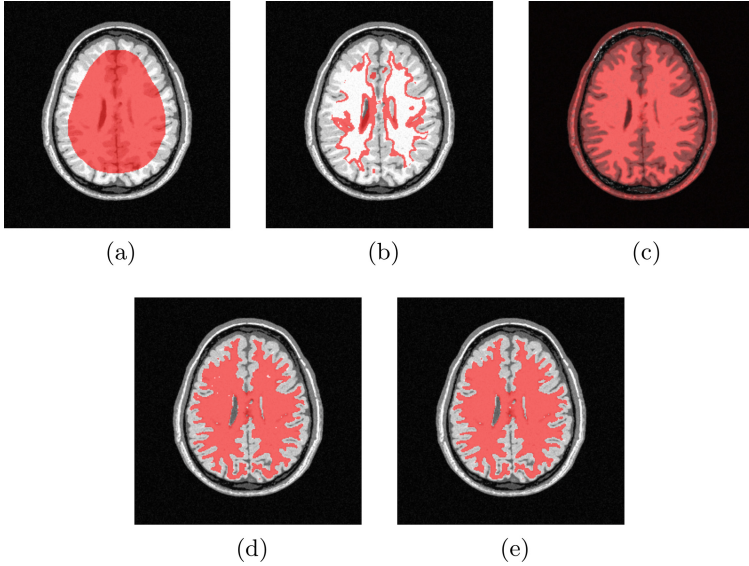


Fig. 9. Identification of white matter. In red points satisfying `headInt` (a); points satisfying `white1` (b); level of similarity `whiteT1` (c); points satisfying `white2` (d); points satisfying `white` (e). (Color figure online)

In Specification 4 we proceed by identifying the grey matter of the brain. In line 1 we take a larger internal portion of the head than that of `headInt` so that we are sure that the grey matter, which is mostly situated near the white matter towards the outside of the brain, is included. In line 2 an area is identified that is almost certainly part of the grey area, using expert knowledge on percentile ranking, similarity coefficient and position within the internal part of the head (Fig. 10a). In line 3 we use the knowledge that grey matter is attached to the white matter of the brain (Fig. 10b). We now have a good sample of grey matter, but there are some areas that are not covered (see the grey parts that are not red yet in the left and top part of the head in Fig. 10b). So we look for further texture that is similar to grey matter in line 4 and 5 (Fig. 10c and d). This indeed includes the previous grey areas that were missed out, but also includes other areas in the bottom outer part of the head. We use again the knowledge that grey and white matter are next to each other using `touch` and `distleq` and the knowledge that white and grey matter do not overlap, i.e. `!white` to exclude the areas on this outer part as they cannot be part of the grey matter for anatomical reasons. This gives us the final result for grey matter as shown in (Fig. 10e).

In a very similar fashion we can also identify the cerebrospinal fluid (CSF), the skull and the bone marrow for example. We omit the details here and only show the final results in Fig. 11. In the same figure we show some preliminary analysis of the quality of the described method by comparing our results with a ‘ground truth’ segmentation provided by the method in [2]. Figure 11 shows our

Specification 4: Segmentation of grey matter

```

1 let headInt2 = head & !(distleq(10,!head))
2 let grey1 = (headSim >. 0.6) & (pt1 <. 0.8) & headInt2
3 let grey2 = touch(grey1,white)
4 let greyT1 = similarT1To(grey2)
5 let grey4 = (greyT1 >. 0.3) & (pt1 <. 0.8) & (pt1 >. 0.4) & (whiteT1
  <. 0.8)
6 let grey = touch(grey4,white) & distleq(9,white) & !white

```

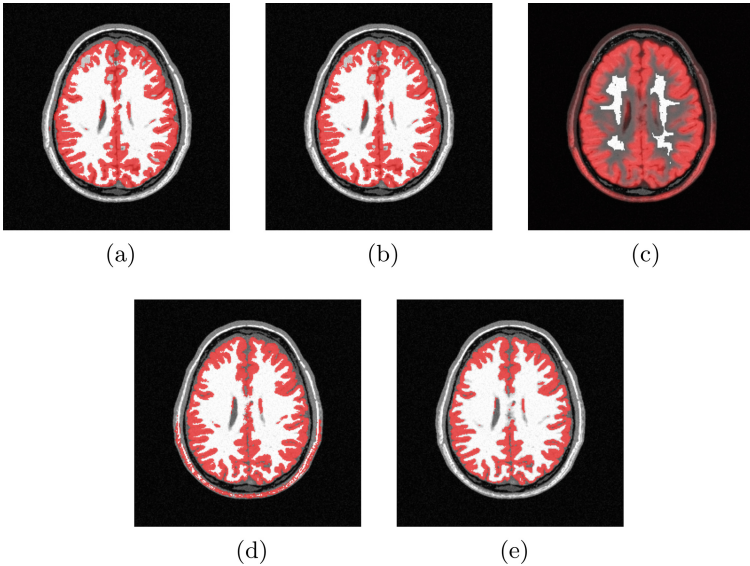


Fig. 10. Identification of grey matter. In red points satisfying `grey1` (a); points satisfying `grey2` (b); level of similarity `greyT1` (c); points satisfying `grey4` (d); points satisfying `grey` (e). (Color figure online)

results in red and the ‘ground truth’ segmentation of [2] for case study¹² Pat04 in blue. The points in pink are those where both analyses coincide.

Note that all the analyses are performed in 3D and can be viewed from all three directions. As an example, in Fig. 12 we show the three perspectives, axial, coronal and sagittal, for the grey matter for one particular cross-section.

We have applied the same specification also on case Pat05 of the same public database and have obtained very similar results. As a preliminary measure of similarity between the ‘ground truth’ segmentation and that obtained via spatial model checking we report here the Dice measure¹³, and the sensitivity and

¹² See http://brainweb.bic.mni.mcgill.ca/brainweb/anatomic_normal_20.html.

¹³ $Dice = 2 * TP / (2 * TP + FN + FP)$, where TP denotes True Positive and FN denotes False Negative.

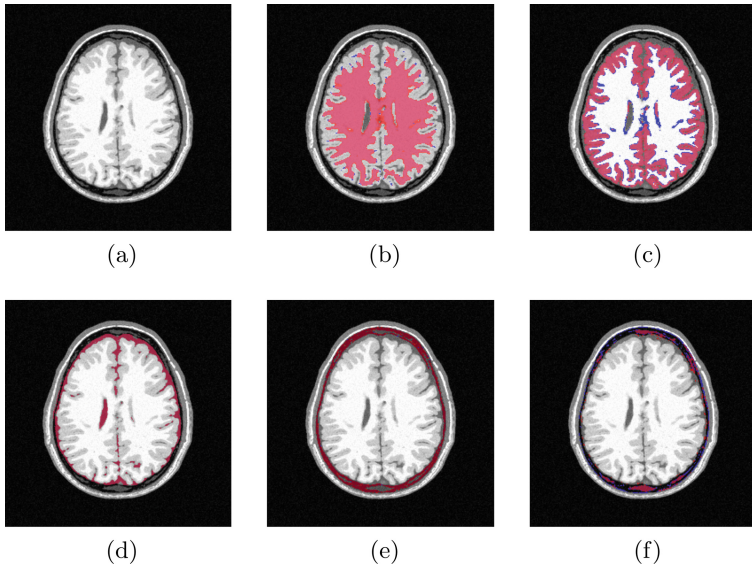


Fig. 11. Comparison with other results. Original axial view (a). In red the points satisfying our specification, in blue the ‘ground truth’ provided by the method in [2] for the case study of patient nr. 04 and in pink points that satisfy both analyses. White matter (b); grey matter (c); CSF (d); skull (e); bone marrow (f). (Color figure online)

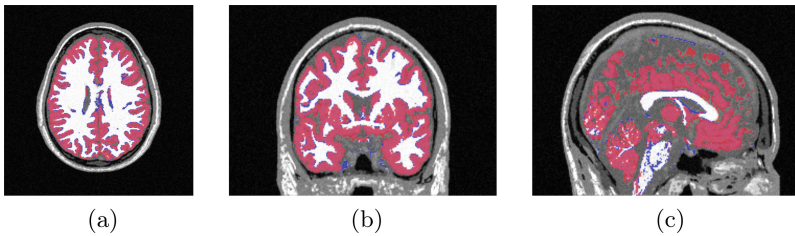


Fig. 12. Comparison in 3D perspective for grey matter. In red points satisfying our specification, in blue points identified by the ‘ground truth’ segmentation in [2] for the case study of patient nr. 04 and in pink points that satisfy both analyses. Axial (a), coronal (b) and sagittal (c) perspective. (Color figure online)

specificity measures for the grey and the white matter of the brain. Sensitivity measures the fraction of voxels that are correctly identified as part of a tumour (True Positives). Specificity measures the fraction of voxels that are correctly identified as *not* being part of a tumour (True Negatives). All these similarity coefficients give a result between 0 (no similarity) and 1 (perfect similarity) and are commonly used to provide a combined measure of the similarity of two segmentations. For example a Dice index of around 0.9 is considered as indicating very good similarity. This is so because there is no unique ‘gold standard’ for comparison as also manual expert markings have a certain level of variability (see also Sect. 5 and [32]).

Table 1. Similarity between the results of ‘ground truth’ segmentation and that obtained via spatial model checking for case studies Pat04 and Pat05

	Pat04			Pat05		
	Dice	Sensitivity	Specificity	Dice	Sensitivity	Specificity
Grey matter	0.90	0.91	0.98	0.89	0.88	0.99
White matter	0.89	0.85	1.0	0.90	0.85	1.0

Table 1 provides some first indication that the specification could be a good candidate to be applied to further cases for the segmentation of MRI images of healthy brains, much in the same way as we have done for tumour segmentation in [8]. Also from an execution time point of view our preliminary results obtained with `VoxLogicA` are encouraging as the complete segmentation (head, background, white matter, grey matter, CSF, skull and bone marrow) of the brain was obtained in less than 2 min on a MacBook Pro running MacOS Mojave, with 2.7 GHz Intel core i7 and 16 GB of memory. We leave the analysis of the other cases of the benchmark and a more complete comparison with the results of other techniques for segmenting healthy brain tissue for future work.

5 Challenges in Spatial Model Checking for Medical Imaging

Medical Imaging, and in particular brain tumour segmentation, is a very active and important area of research, see for example [33] and references therein. There are, however, also a great number of challenges. In the following we briefly describe some of them, and in particular those where we think that an approach based on spatial model checking may make a significant contribution to.

Modularity, Composition and Flexibility. An aspect that all logic-based model checking techniques have in common is their reliance on a relatively small set of basic logical operators. There are in general many choices for such minimal

sets of operators. Ideally, this provides at the same time a good expressivity and also the basis for the definition of a useful set of *derived* operators that match the level of domain specific reasoning of the user. In case of medical imaging for radiotherapy, for example, that could be the neuro-radiologist. Derived operators exploit the *compositionality* of the basic operators so that more complicated operators can be constructed out of the basic building blocks. Furthermore, the basic set should allow for very *efficient verification algorithms* and be minimal, such that the *correctness* of the algorithms can be proven with reasonable and acceptable effort. Finally, we would also expect a certain *flexibility and generality* of the approach requiring that the operators are not too much specifically tied to a particular application or even a single case study or type of analysis.

Although these notions have been studied extensively in the wider field of model checking, the solutions in the sense of particular sets of basic operators, that have been proposed need to be reinterpreted in the case of spatial model checking and founded on other mathematical theories such as closure spaces. Restricting space to more regular structures than general graphs may also lead to very significant increases in the efficiency of spatial model checking algorithms at the cost of some generality. Such increase in efficiency may, however, make the difference between a tool that can effectively improve the daily work of, for example, neuro-radiologists and a tool that is very general but too slow or requiring too much memory to be of practical use. These increases in efficiency are also due to the fact that for example very efficient existing image processing algorithms and related software packages can be exploited. For example, in *VoxLogica* [8] the state-of-the-art cross-platform and open source computational imaging library ITK¹⁴ was used for the efficient implementation of the basic spatial operators. What exactly constitutes the best choice of basic spatial operators, given the trade-offs, is still not a fully answered question, however some promising candidate sets have been proposed for this area in some of our earlier work [12, 13].

Interactive Feedback and Ease of Use. Whereas traditional model checkers and their extensions are in general used by experts in formal methods or by software engineers, in the case of spatial model checking for medical applications the expected users have a different background. Moreover, the specific analyses that clinicians or neuro-radiologists are envisioned to perform with these tools should be embedded into their daily activities and must be well-integrated with other environments that they use such as those for safe and secure archiving of images and their results. Furthermore, other aspects should be taken into account as well, such as cognitive fatigue and avoidance of information overload, allowing users to easily focus on their main critical(!) task at hand without being distracted by useless details. There are relatively few studies addressing these issues in some depth, even if the problem has been taken up by some research groups, see for example work by Gambino et al. [24] for a survey and some concrete proposals.

¹⁴ The Insight Segmentation and Registration Toolkit, see <https://itk.org> and <http://www.simpleitk.org>.

Explainability, Independent Reproducibility and Transferability. In recent years, following the success of the use of artificial intelligence and neural networks for image recognition tasks, much of the research in medical imaging and segmentation tasks in particular, has focused on these (probabilistic) learning algorithms (see for example [33, 38]). Such algorithms, however, usually depend on the availability of large and precisely annotated data sets for their learning phase. During such a learning phase the software is autonomously calibrating the value of thousands of parameters until the resulting program provides a satisfactory level of correct responses on the training data set. 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. Such learning algorithms may reach surprisingly good results, but they also pose some open challenges. First of all, developing large and precisely annotated data sets in delicate areas such as medical imaging and tumour segmentation is a time-consuming task that can only be performed, mostly manually, by specialists. This is difficult to achieve, not only because it is very laborious, but also because of the relatively high intra-expert and inter-expert variability; [32] quantifies the average of disagreement in identified contours by experts as $20 \pm 15\%$ and $28 \pm 12\%$, respectively, for manual segmentations of brain tumour images. Interactive approaches based on spatial model checking, in this context, may be of help to improve the generation of manual ground truth labels in a more efficient, transparent and reproducible way. Furthermore, the automatic algorithms that are obtained with a (deep) learning approach cannot provide human intelligible insight in why certain areas are identified as tumours. In other words, these procedures lack *explainability*. This is a more serious problem than it may seem, at first sight, since these algorithms (as any other in this context) do not *always* provide the correct results, and in the critical context of radiotherapy the preparations must comply to rigorous protocols where medical staff must be put in the condition to take responsibility for their decisions.

Extendability and Openness. Manual segmentation by experts 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. However, the expertise that has been gained by many practitioners in the field is very valuable and could in principle be exploited in improving segmentation procedures if such procedures could be easily expressed through and supported by computer assisted operations at the right level of abstraction. For example, if various segmentation procedures could be captured in an unambiguous way by the composition of a number of rather high-level operations in a formal specification, such specifications could be published, exchanged, discussed and improved directly by those experts working in the field. This challenge would call for an extendable framework, where new operations can be introduced, and that is sufficiently open to be used by a wider community.

Device Independence and Vendor Neutrality. Different research groups and institutions employ specific best-practices for image analysis, often locally built from home-made integration of different software technologies. This gives rise to a plethora of incompatible systems, very often of academic significance, but rarely used in clinical practice. Missing integration with existing hardware, hand-crafted procedures, lack of maintenance and accountability, difficult use, hard-coded dependence on the execution environment (e.g. specific operating system or hardware), are just some factors that hinder clinical application, medical procedure approval processes, and ethical scrutiny, creating a barrier between medical research and healthcare. Thus, successful technological transfer mostly happens by specialised, proprietary software solutions that are typically bundled with the hardware. The challenge here is to overcome the fragmentation of the medical imaging ecosystem, by providing a set of open standards and reference implementations, fostering a paradigm shift in the field in several ways such as by facilitating communication between research and healthcare providers and by providing technologies that are appropriate for intermediaries (such as manufacturers and vendors of Medical Imaging devices) to turn novel ideas into clinical practice. This may be pursued through the social computing capabilities of a spatial logic based language, that can attract experts of diverse fields to collaborate through a common communication infrastructure.

Privacy Issues. Regulation issues, especially related to privacy, may easily arise in an open platform. However, in the envisioned approach privacy is a key strength, rather than an issue. The definition of an open standard for image analysis, and its *free and open source software* implementation, will enable users to exchange analysis procedures, and establish common knowledge, without outsourcing the actual, privacy sensitive data, which can be handled on-site, obeying to the locally established practices.

The above is only a small selection of the many challenges in medical imaging for radiotherapy. We do by no means intend to present an exhaustive list. However, we think that the listed challenges are relevant and have shown where we expect that further research in a spatial logic-based method may lead to a useful contribution to advance this important field.

6 Related Work

Most of the present paper was dedicated to the potential of spatial model checking in the field of medical imaging, initiated in [7], and of image segmentation and contouring for the purpose of radiotherapy in particular (see [4, 6, 8]). However, spatial model checking has been explored in a number of other applications and it has been extended in several ways. In this section we provide a brief overview of recent related work.

A very valuable resource and reference on the topological origins of spatial logic is the *Handbook of Spatial Logics* [1]. This handbook describes several spatial logics, with applications far beyond topological spaces. Among them are not only logics that treat morphology, geometry, distance, or such as dynamic systems, but also a treatment of discrete models, that are particularly difficult to deal with from a topological perspective. See, for example [21], introducing the approach of Closure Spaces upon which the work in [12, 13, 16] is based.

Starting from a spatial formalism and from a temporal formalism, *spatio-temporal* logics may be defined, by introducing a mutually recursive nesting of spatial and temporal operators. Several combinations can be obtained, depending on the chosen spatial and temporal fragments, and the permitted forms of nesting of the two. A large number of possibilities are explored in [27], for spatial logics based on topological spaces. One such structure was investigated in the setting of closure spaces, namely the combination of the *Computation Tree Logic* (CTL) with SLCS, resulting in the *Spatio-Temporal Logic of Closure Spaces* (STLCS). In STLCS spatial and temporal fragments may be arbitrary and mutually nested.

STLCS is interpreted on a variant of Kripke models, where valuations are interpreted at points of a closure space. Fix a set P of proposition letters. STLCS formulae and *path* formulas are defined by the grammars shown below, where p ranges over P .

$$\Phi ::= p \mid \neg\Phi \mid \Phi \vee \Phi \mid \mathcal{N}\Phi \mid \rho \Phi[\Phi] \mid \mathbf{A}\varphi \mid \mathbf{E}\varphi \quad (2)$$

$$\varphi ::= \mathcal{X}\Phi \mid \Phi \mathcal{U}\Phi \quad (3)$$

The logic features the CTL path quantifiers \mathbf{A} (“for all paths”), and \mathbf{E} (“there exists a path”). As in CTL, such quantifiers must necessarily be followed by one of the path-specific temporal operators, such as¹⁵ $\mathcal{X}\Phi$ (“next”), $\mathbf{F}\Phi$ (“eventually”), $\mathbf{G}\Phi$ (“globally”), $\Phi_1 \mathcal{U}\Phi_2$ (“until”), but, unlike CTL, in this case Φ , Φ_1 and Φ_2 are STLCS formulas that may make use of spatial operators, e.g. \mathcal{N} , ρ and operators derived thereof (see Sect. 2.) The mutual nesting of such operators permits one to express spatial properties in which the involved points are constrained to certain temporal behaviours.

As a proof of concept, a model checking algorithm has been defined, which is a variant of the classical CTL labelling algorithm [3, 19], augmented with the algorithm in [10] for the spatial fragment. The algorithm, which operates on finite spaces, has been implemented as a prototype tool which is described in [11]. The same algorithm is also implemented in the tool `topochecker`.

The tool has been used to analyse a number of properties of vehicular movement in public transport systems in the context of *smart cities*. In [10], a bus transportation case study was developed, to detect problems in the automatic vehicle location (AVL) data that is provided as input to other systems that in

¹⁵ Some operators may be derived from others; for this reason in the definition of the language we use a minimal set of connectives. As usual in logics, there are several different choices for such a set.

turn provide information to passengers and system operators such as bus arrival prediction systems. Such data may contain errors originating in a problem with the hardware of the measurement device or also indicate operational problems experienced by bus drivers that encountered unexpected road works or accidents and have to deviate from their planned route.

In [15], spatio-temporal model checking has been used to study a phenomenon known as *clumping*, which may occur in so-called “frequent” services – those where a timetable is not published. Clumping occurs where one bus catches up with – or at least comes too close to – the bus which is in front of it. In [17] spatio-temporal model checking has been used to detect the emergent formation of ‘clusters’ of full (and empty) stations in the simulation traces of a Markov Renewal Process (MRP) model of large bike sharing systems [31]. Subsequently, spatio-temporal model checking has been used in combination with statistical model checking in [18] to analyse further properties of bike sharing systems.

The logics discussed so far characterise properties of *single points* in space. In [13] an extended version of SLCS has been defined that is able to express properties that *sets* of points may satisfy *collectively*. The resulting logic, the *Collective* SLCS, CSLCS, can be used for example, for expressing that the points satisfying a certain formula Φ_1 are *collectively* surrounded by points satisfying formula Φ_2 . The notion of *region* as set of points and related properties has been studied extensively in the literature, also in the context of discrete spaces (see [36,37] among others). For instance, RCC5D is a theory of region parthood for discrete spaces and RCC8D extends it with the topological notion of connection and the relations of disconnection, external connection, tangential and nontangential proper parthood and their inverse relations. In [14] an encoding of RCC8D into CSLCS is provided and it is shown how *topochecker* can be used for effectively checking the existence of a RCC8D relation between two given regions of a discrete space.

Two variants of the spatial modalities have also been added to the Signal Temporal Logic [20,30] leading to the Signal Spatio-Temporal Logic (SSTL). The first variant, the *bounded somewhere* operator $\diamond_{[w_1, w_2]}$ is borrowed from [34], while the second one, the *bounded surround* operator $\mathcal{S}_{[w_1, w_2]}$, is inspired by SLCS. The logic comes with a boolean and quantitative semantics which can be found in [34,35]. The boolean semantics defines when a formula is satisfied, the quantitative semantics provides an indication of the robustness with which a formula is satisfied, i.e. how susceptible it is to changing its truth value for example as a result of a perturbation in the signals. In [5] an extension of SSTL is presented which uses a reachability operator as a basic operator of the logic.

In [25] a variant of spatial logic is proposed where spatial properties are expressed using quad trees. The authors show that very complex spatial structures can be identified with the support of model checking algorithms as well as machine learning procedures. However, the formulation of spatial properties becomes rather complex. The combination of this spatial logic with linear time signal temporal logic, defined with respect to continuous-valued signals, has recently led to the spatio-temporal logic SpaTeL [26].

7 Conclusions

Medical imaging is a very broad and active field of research with particular requirements. In this work we have illustrated the basic framework of spatial verification and how spatial logic and spatial model checking can be used to identify various kinds of tissues in the healthy brain. The field of medical imaging is posing very particular challenges, not only of technical nature, but in particular also in terms of responsibility, explainability, transparency and reproducibility. Formal verification and spatial model checking may provide interesting complementary methods in this important field.

The presented specifications are a first proof of concept to show that it is indeed possible to identify various (healthy) brain tissues with the available operators in the presented logic. The results are promising from different perspectives, however, we expect that the specifications can be further improved to obtain better accuracy. Improvements are also foreseen from the methodological point of view. More work is needed to refine the analyses and check applicability to a larger set of images in particular with respect to stability and accuracy of the results, and to make the approach available in a clinical setting. The latter requires the design of appropriate case studies and establishing experimental protocols for clinical validation. This is planned as part of future work.

Acknowledgments. This paper was written for the Festschrift in honour of Director of Research Dr. Stefania Gnesi. We would like to thank Stefania for the many years she has been coordinating our Formal Methods and Tools Laboratory at ISTI-CNR, and we hope she will continue to contribute to our Lab for many years to come. She guided the group safely through the many periods of instability of very different nature, and she did so with confidence and optimism. If now we have so many young (and less young) motivated formal methods researchers in our group, that explore and develop new and creative directions of formal methods research, both in theory and for applications, this is made possible, in a large part, thanks to her tireless efforts in all these years.

Part of this work has been developed in the context of the Italian MIUR-PRIN 2017 project “IT MaTTerS: Methods and Tools for Trustworthy Smart Systems”.

References

1. Aiello, M., Pratt-Hartmann, I., van Benthem, J. (eds.): Handbook of Spatial Logics. Springer, Dordrecht (2007). <https://doi.org/10.1007/978-1-4020-5587-4>
2. Aubert-Broche, B., Griffin, M., Pike, G., Evans, A., Collins, D.: Twenty new digital brain phantoms for creation of validation image data bases. *IEEE Trans. Med. Imaging* **25**(11), 1410–1416 (2006). <https://doi.org/10.1109/TMI.2006.883453>
3. Baier, C., Katoen, J.: Principles of Model Checking. MIT Press, Cambridge (2008)
4. Banci Buonamici, F., Belmonte, G., Ciancia, V., Latella, D., Massink, M.: Spatial logics and model checking for medical imaging. *Int. J. Softw. Tools Technol. Transf.* (2019). <https://doi.org/10.1007/s10009-019-00511-9>. Online First

5. Bartocci, E., Bortolussi, L., Loretì, M., Nenzi, L.: Monitoring mobile and spatially distributed cyber-physical systems. In: Talpin, J., 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, 29 September–02 October 2017, pp. 146–155. ACM (2017). <https://doi.org/10.1145/3127041.3127050>
6. Belmonte, G., et al.: A topological method for automatic segmentation of glioblastoma in mr flair for radiotherapy - ESMRMB 2017, 34th annual scientific meeting. Magn. Reson. Mater. Phys. Biol. Med. **30**(S1), 437 (2017). <https://doi.org/10.1007/s10334-017-0634-z>
7. 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.H., Loretì, 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. EPTCS, vol. 217, pp. 81–92 (2016). <https://doi.org/10.4204/EPTCS.217.10>
8. Belmonte, G., Ciancia, V., Latella, D., Massink, M.: VoxLogicA: a spatial model checker for declarative image analysis. In: Vojnar, T., Zhang, L. (eds.) TACAS 2019, Part I. LNCS, vol. 11427, pp. 281–298. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17462-0_16. Preprint <http://arxiv.org/abs/1811.05677>
9. van Benthem, J., Bezhanishvili, G.: Modal logics of space. In: Handbook of Spatial Logics [1], pp. 217–298
10. Ciancia, V., Gilmore, S., Latella, D., Loretì, 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)
11. Ciancia, V., Grilletti, G., Latella, D., Loretì, M., Massink, M.: An experimental spatio-temporal model checker. In: Bianculli, D., Calinescu, R., Rumpe, B. (eds.) SEFM 2015. LNCS, vol. 9509, pp. 297–311. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-49224-6_24
12. Ciancia, V., Latella, D., Loretì, M., Massink, M.: Specifying and verifying properties of space. In: Diaz, J., Lanese, I., Sangiorgi, D. (eds.) TCS 2014. LNCS, vol. 8705, pp. 222–235. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44602-7_18
13. Ciancia, V., Latella, D., Loretì, M., Massink, M.: Model checking spatial logics for closure spaces. Log. Methods Comput. Sci. **12**(4) (2016). <http://lmcs.episciences.org/2067>
14. Ciancia, V., Latella, D., Massink, M.: Embedding RCC8D in the collective spatial logic CSLCS. In: Boreale, M., Corradini, F., Loretì, M., Pugliese, R. (eds.) Models, Languages, and Tools for Concurrent and Distributed Programming. LNCS, vol. 11665, pp. 260–277. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21485-2_15
15. Ciancia, V., Gilmore, S., Grilletti, G., Latella, D., Loretì, M., Massink, M.: Spatio-temporal model checking of vehicular movement in public transport systems. STTT **20**(3), 289–311 (2018). <https://doi.org/10.1007/s10009-018-0483-8>
16. Ciancia, V., Latella, D., Loretì, M., Massink, M.: Spatial logic and spatial model checking for closure spaces. In: Bernardo, M., De Nicola, R., Hillston, J. (eds.) SFM 2016. LNCS, vol. 9700, pp. 156–201. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-34096-8_6

17. Ciancia, V., Latella, D., Massink, M., Paškauskas, R.: Exploring spatio-temporal properties of bike-sharing systems. In: 2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASO Workshops 2015, Cambridge, MA, USA, 21–25 September 2015, pp. 74–79. IEEE Computer Society (2015). <https://doi.org/10.1109/SASOW.2015.17>
18. Ciancia, V., Latella, D., Massink, M., Paškauskas, R., Vandin, A.: A tool-chain for statistical spatio-temporal model checking of bike sharing systems. In: Margaria, T., Steffen, B. (eds.) ISO/LA 2016. LNCS, vol. 9952, pp. 657–673. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47166-2_46
19. Clarke, E.M., Grumberg, O., Peled, D.: Model Checking. MIT Press (2001). <http://books.google.de/books?id=Nmc4wEaLXFEC>
20. Donzé, A., Ferrère, T., Maler, O.: Efficient robust monitoring for STL. In: Sharygina, N., Veith, H. (eds.) CAV 2013. LNCS, vol. 8044, pp. 264–279. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39799-8_19
21. Galton, A.: The mereotopology of discrete space. In: Freksa, C., Mark, D.M. (eds.) COSIT 1999. LNCS, vol. 1661, pp. 251–266. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48384-5_17
22. Galton, A.: A generalized topological view of motion in discrete space. *Theor. Comput. Sci.* **305**(1–3), 111–134 (2003). [https://doi.org/10.1016/S0304-3975\(02\)00701-6](https://doi.org/10.1016/S0304-3975(02)00701-6)
23. Galton, A.: Discrete mereotopology. In: Calosi, C., Graziani, P. (eds.) Mereology and the Sciences. SL, vol. 371, pp. 293–321. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05356-1_11
24. Gambino, O., Rundo, L., Cannella, V., Vitabile, S., Pirrone, R.: A framework for data-driven adaptive GUI generation based on DICOM. *J. Biomed. Inform.* **88**, 37–52 (2018). <https://doi.org/10.1016/j.jbi.2018.10.009>
25. 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)
26. 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 2015, pp. 189–198. ACM, New York (2015)
27. Kontchakov, R., Kurucz, A., Wolter, F., Zakharyashev, M.: Spatial logic + temporal logic = ? In: Handbook of Spatial Logics [1], pp. 497–564
28. Kwan, R.S., Evans, A., Pike, G.: MRI simulation-based evaluation of image-processing and classification methods. *IEEE Trans. Med. Imaging* **18**(11), 1085–1097 (1999)
29. Litjens, G.J.S., et al.: A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017). <https://doi.org/10.1016/j.media.2017.07.005>
30. Maler, O., Nickovic, D.: Monitoring temporal properties of continuous signals. In: Lakhnech, Y., Yovine, S. (eds.) FORMATS/FTRTFT 2004. LNCS, vol. 3253, pp. 152–166. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30206-3_12
31. Massink, M., Paškauskas, R.: Model-based assessment of aspects of user-satisfaction in bicycle sharing systems. In: Sotelo Vazquez, M., Olaverri Monreal, C., Miller, J., Broggi, A. (eds.) 18th IEEE International Conference on Intelligent Transportation Systems, pp. 1363–1370. IEEE (2015). <https://doi.org/10.1109/ITSC.2015.224>
32. 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)

33. Menze, B., et al.: The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Trans. Med. Imaging* **34**(10), 1993–2024 (2015)
34. 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, 9–11 December 2014. *ICST* (2014)
35. Nenzi, L., Bortolussi, L., Ciancia, V., Loreti, M., Massink, M.: Qualitative and quantitative monitoring of spatio-temporal properties. In: Bartocci, E., Majumdar, R. (eds.) *RV 2015*. LNCS, vol. 9333, pp. 21–37. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23820-3_2
36. Randell, D.A., Cui, Z., Cohn, A.G.: A spatial logic based on regions and connection. In: Nebel, B., Rich, C., Swartout, W.R. (eds.) *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning (KR 1992)*, pp. 165–176. Morgan Kaufmann, Burlington (1992)
37. Randell, D.A., Landini, G., Galton, A.: Discrete mereotopology for spatial reasoning in automated histological image analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(3), 568–581 (2013). <https://doi.org/10.1109/TPAMI.2012.128>
38. Spyridon (Spyros) Bakas, et al. (eds.): 2017 International MICCAI BraTS Challenge: Pre-conference Proceedings, September 2017. https://www.cbica.upenn.edu/sbia/Spyridon.Bakas/MICCAI_BraTS/MICCAI_BraTS_2017_proceedings_shortPapers.pdf