Accepted Manuscript

The Semiotics of Medical Image Segmentation

John SH Baxter , Eli Gibson , Roy Eagleson , Terry M. Peters

PII: S1361-8415(17)30176-7
DOI: 10.1016/j.media.2017.11.007
Reference: MEDIMA 1314

To appear in: *Medical Image Analysis*

Received date: 17 April 2017
Revised date: 30 October 2017
Accepted date: 18 November 2017

Please cite this article as: John SH Baxter , Eli Gibson , Roy Eagleson , Terry M. Peters , The Semiotics of Medical Image Segmentation, *Medical Image Analysis* (2017), doi: 10.1016/j.media.2017.11.007

Highlights

- A semiotics-driven framework for understanding knowledge-driven interaction in medical image segmentation is proposed.
- This framework is grounded in Peircean semiotics in order to structure and characterize how particular interactions are interpreted by both the user and the computer.
- Using the notion of interface metaphors, this framework shows how metaphor quality metrics can be used to analyze interaction and improve ease-of-use in communicating complex anatomical knowledge.

The Semiotics of Medical Image Segmentation*

John SH Baxter[ab†], Eli Gibson[c], Roy Eagleson[b] and Terry M. Peters[ab]
[a] Robarts Research Institute
[b]Western University, London, Ontario Canada
[c] University College, London, UK

Abstract

As the interaction between clinicians and computational processes increases in complexity, more nuanced mechanisms are required to describe how their communication is mediated. Medical image segmentation in particular affords a large number of distinct loci for interaction which can act on a deep, knowledge-driven level which complicates the naive interpretation of the computer as a symbol processing machine. Using the perspective of the computer as dialogue partner, we can motivate the semiotic understanding of medical image segmentation. Taking advantage of Peircean semiotic traditions and new philosophical inquiry into the structure and quality of metaphors, we can construct a unified framework for the interpretation of medical image segmentation as a sign exchange in which each sign acts as an *interface metaphor*. This allows for a notion of *finite semiosis*, described through a schematic medium, that can rigorously describe how clinicians and computers interpret the signs mediating their interaction. Altogether, this framework provides a unified approach to the understanding and development of medical image segmentation interfaces.

Keywords: medical image segmentation, Peircean semiotics, human computer interaction, interface metaphors

1. Introduction

The notion of the computer as a blind symbol-processing engine is under-prepared to describe the complexity of modern computational systems and tasks. The idea that any communication with a computer must be symbolic (thus limiting the tool-set used to investigate it) by virtue of the computer's processor operating, on the lowest level, upon zeros and ones is philosophically naive and denies the flexibility of modern input and output methods (both physical and abstract) and their suitability for particular interfaces. One particularly interesting instance is medical image processing systems which lie at the confluence of two vastly disparate fields: computer vision and artificial intelligence in which ill-defined problems are structured, solved, and thus addressed; and personalized medicine in which the medical course of action is determined patient-by-patient preceded by diagnostic testing, often genetics or imaging based.

Although medical image processing is an umbrella term encompassing many distinct applications, this article focuses on the problem of medical image segmentation, defined as the partitioning of an image into a series of regions each corresponding to a organ or other anatomy of interest. As volumetric imaging such as magnetic

* Dedicated to the memory of Dr. Cesare Romagnoli whose love for medical imaging and philosophy inspired this work.
† John Baxter is the corresponding author at email address: jbaxter@robarts.ca

resonance imaging and computed tomography to digital histology has become invaluable to modern medicine, segmentation, fundamental for personalized medicine, has become more complex. The identification and delineation of anatomical structures or regions of interest is often seen as a fundamental step in processes as diverse as quantification of anatomical features (Bagci, et al., 2013) (Moonis, Liu, Udupa, & Hackney, 2002), physiological and anatomical modelling (Huang, et al., 2013) (Juneja, Harris, Kirby, & Evans, 2012), computer-assisted diagnosis (Wiemker, et al., 2014) (Armato & Sensakovic, 2004), the detection and localization of pathology (Prastawa, Bullitt, Ho, & Gerig, 2004) (Zheng, Chang, & Gur, 1995), determination of margins in cancer resection and other treatment planning (Peters & Cleary, 2008), and visualization of surgical scenes in image-guided interventions (Peters & Cleary, 2008). Additionally, medical image segmentation has served a vital role in medical imaging research from anatomical atlas generation and registration (Park, Bland, & Meyer, 2003), to finding correlations between imaging modalities (Turkbey, et al., 2012) (Goubran, et al., 2014). Thus, medical image processing (segmentation in particular) has become the meeting ground of clinicians of various stripes, basic science researchers in a variety of fields, and applied scientists in computer vision, biomedical engineering and medical technology. The techniques and algorithms in medical image segmentation vary widely from application to application. This diversity is dependent on the multiplicity of factors which govern the selection of segmentation frameworks, including the imaging modalities available, computational memory and efficiency constraints (Scholl, Aach, Deserno, & Kuhlen, 2011), financial and hardware considerations (Smistad, Falch, Bozorgi, Elster, & Lindseth, 2015), the structure of the anatomical regions of interest and any associated normal or pathological variability.

A common concept in computer programming is the separation of interface and implementation, which isolates the underlying computational processes from those which interact directly with the user. In that vein, a *segmentation algorithm* is the underlying computational processes that perform segmentation, whereas a *segmentation interface* is the front-end components through which the user communicates with the computer as well as the general or abstract, rather than specific, behaviour of the algorithm. Both of these, as well as the user's reasoning, are a part of the larger *segmentation process* which encompasses how the segmentation is performed as a whole. The diversity of segmentation algorithms and interfaces currently available is both large and growing. The purpose of this article is not to contend with literature reviews of medical image segmentation research, nor to enumerate the different computational processes and properties thereof. (Many such literature reviews are available, an all-too-brief list includes (Smistad, Falch, Bozorgi, Elster, & Lindseth, 2015) (Clarke, et al., 1995), (Pham, Xu, & Prince, 2000), & (Peng, Zhang, & Zhang, 2013).) One distinctly different literature review is that of Olabarriaga & Smeulders (2001) which proposes a three-pronged approach to understanding segmentation processes, rather than algorithms or interfaces, at an abstract level, focusing on the interaction between the clinical user and the computer. The first prong involves determining the type of input from the user during the segmentation process, the second the interpretation of user input as feedback to some computational mechanism, and the last the determination of the overall purpose of user intervention in this process. Although each individual category may display its age, having being published more than 15 years ago in a fast-paced research environment, the approach is fairly comprehensive and certainly motivates the concept of understanding human-computer interaction in medical image segmentation at an abstract level. Unfortunately, this paper is presented primarily as a survey of the state-of-the-art at the time and thus did not search for an underlying theory or rigorous conceptual framework to support its insights. In addition, since it was published in 2001, several advances have been made in medical image segmentation. The popularization of ITKSnap as a standard tool for manual segmentation and the development of frameworks such as MeVisLab (Ritter, et al., 2011) and 3D Slicer (Pieper, 2004), for designing medical image computing tasks, have provided a more common and repeatable basis for creating segmentation processes, giving an avenue for new interaction mechanisms to thrive. New algorithmic paradigms such as graph-cuts (Boykov & Jolly, 2001) and deep learning (Hinton, 2006) have also created new challenges in structuring complex segmentation problems. In this new environment, a unified approach to understanding segmentation can help elicit the fundamental commonalities between disparate approaches and the contrasts between similar ones, ultimately guiding conceptual development.

The purpose of this article is to develop that systematic program for conceptually analyzing segmentation processes by specifically updating and expanding upon the approach made by Olabarriaga & Smeulders (2001). The intent is to encourage readers regardless of field to critically examine all currently used and state-of-the-art segmentation algorithms and interfaces, both in research and in the clinic, in terms of their roles in the overall segmentation process and to do so from a rigorous philosophical point-of-view. Peircean semiotics, as well as more recent thought in cognitive science, can motivate that larger perspective. Section 2 motivates viewing the segmentation process as a dialogue involving a series of exchanged signs, which fundamentally necessitates a framework for semiotic analysis. Section 3 describes a conceptual framework for the design of segmentation processes from the view of information that must be communicated via a sign exchange. Section 4 describes a framework for understanding the input to, and output from, segmentation interfaces in terms of Peircean semiotics, which supports our conceptual framework. In Section 5, a series of quality metrics based on the cognitive science of analogical reasoning are proposed to assist in evaluating the efficacy and usability of individual signs. Section 6 uses the ITKSnap and MeVisLab interfaces as case studies in semiotic design and analysis. The final section discusses how this overall framework reflects the state-of-the-art in segmentation interfaces and provides a program for moving the state-of-the-art forward. The Appendix offers a concrete example of this semiotic analysis applied to two commonly used segmentation toolkits, highlighting strengths and weaknesses of each.

2. Segmentation Automaticity and Sign Exchange

Automaticity has always been of interest in medical image segmentation and has long been considered a spectrum along similar lines to that shown in Figure 1. Many techniques proposed in the literature are semi-automated or even fully automated, leading to highly consistent segmentations across multiple users. That being said, these techniques often lack generality for various reasons and can thus display a large amount of variability in performance across different patient images (Peng, Zhang, & Zhang, 2013) (Olabarriaga & Smeulders, 2001). On the opposite end of the spectrum, manual segmentation is often used as a gold standard, being just as general and as flexible as its user, but this process is known to display a large amount of user variability. A number of interactive or interpolated segmentation approaches effectively blur the distinction between manual segmentation and semi-automated segmentation.

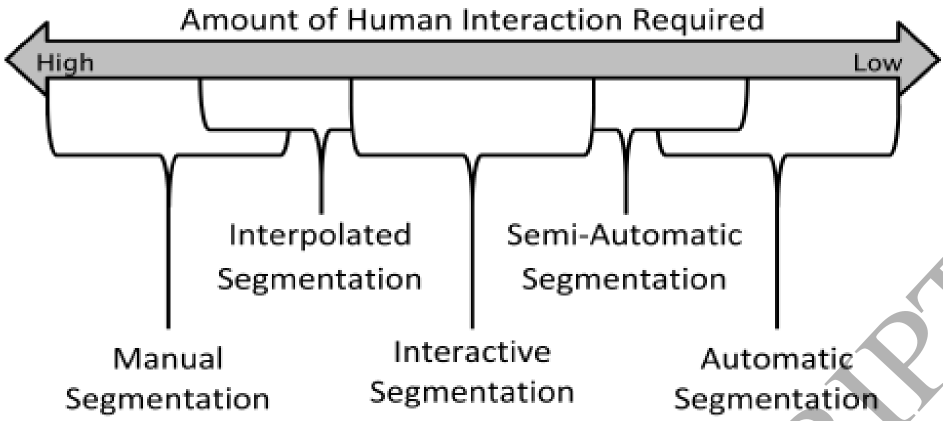


Figure 1: Automaticity dimension of medical image segmentation

One way to view the automaticity spectrum is the level of one-sidedness in who is providing information. In a purely manual approach, the process is completely one-sided, with the user providing all information with the computer serving a passive role as a receptacle. Fully automatic segmentation is similar, although the roles are reversed, with the computer playing the active role and the user passively receiving the resultant segmentation. Intermediate locations on the automaticity spectrum correspond to intermediate levels of one-sidedness, where the user and the computer both to some degree provide meaningful information and address the segmentation task as a team. This interpretation of automaticity through the lens of human-computer interaction has been called the *computer as dialogue partner* perspective defined as "humans and computers [being] regarded as partners in a dialogue. The interaction process is regarded as a communication process in which user and computer application act as both sender and receiver, and the computer application is seen as being able to show communicative behaviour similar to that of its human partner." (Kammersgaard, 1986, p. 350) When viewing the segmentation process as a dialogue, it is natural to ask in which language this dialogue takes place and to investigate its syntax and semantics. In the case of human-computer interaction, the medium for this communication is often considered the abstract realm of signs, whereby the process of interaction between humans and computers is, at its core, a process of sign exchange in which users provides a series of *input signs* to the computer which responds with a series of *output signs*.

But is the use of such a perspective beneficial? It has long been known that manual segmentation is tedious, time-consuming and prone to user error, leading to segmentation errors or incompleteness, hence the large body of research into more automatic methods which alleviate these issues. Thus any degree of automaticity (and thus interactivity) is an improvement on manual segmentation, making use of the computer's capabilities and fundamental consistency. For automatic and semi-automatic segmentation frameworks, the necessity of moving towards less automation is less clear and thus more controversial. No current automatic or semi-automatic method is perfect, but that does not imply that the pursuit of perfection is meritless; we must have positive evidence that moving towards the middle of the spectrum, towards more user input, is beneficial. Empirically, this has come in the form of additional tools for segmentation editing, which allow the user to view and correct an automatically or semiautomatically generated segmentation, which consistently improves efficiency and reduces variability, without sacrificing accuracy across a variety of expert-guided segmentation tasks (Silva, Santos, Madeira, & Silva, 2010) (Deely, et al., 2013) (Heckel, et al., 2014). In terms of medical ethics and law, responsibility for errors can fall upon the medical practitioner, and thus detecting and correcting semi-automatic and automatic segmentation errors is required to avoid negligence. Both of these encourage the physician to take a more active role in the segmentation process even though it is automated. Thus, incorporating knowledge and capabilities from both the user and the computer can readily be seen as beneficial. In that sense, interactive segmentation, despite not being common in either research or in clinical practice but falling between the two in terms of automaticity, should be seen as the expressive default for segmentation processes; the purely manual and purely automatic types currently favoured by the clinic and research communities respectively are the edge-cases in a typology of sign exchange.

With respect to understanding sign exchange in general, semiotics is the philosophical discourse regarding the classification, interpretation and understanding of signs, that is, *signifiers* that represent *objects* other than themselves by virtue of some *interpretant*. Semiotics has been used previously to investigate medical imaging from a philosophical point-of-view (Kanade, 1980) (Cohadon, 1994), but has received much more attention and has been, for the most part, more fruitful in the general field of human-computer interaction (Kammersgaard, 1986) (Bates, 1990) (Nake & Grabowski, 2001)(Yoon, 2003). This perspective is a natural fit with segmentation interfaces because of its focus on, and therefore encouragement of, active communication between the clinical user and the computer as central to the segmentation process.

By viewing segmentation as sign exchange, one can readily incorporate the common-sense notions of automaticity as used in the literature, but with a new methodological tool-kit, that of semiotics. Considering the computer as dialogue partner perspective, supported by recent results in segmentation editing encourages a middle route, that of interactive segmentation in which the user and the computer engage in a dialogue with the user providing initialization, correction, and guidance with various degrees of sparsity, and the computer using some intelligent segmentation interpolation, extrapolation, or pattern recognition capabilities (McGuinness & O'Connor, 2010). Semiotics provides us with a language for describing the types of input signs available to users and output signs available to computer, possibly providing a window into radically different methods of interaction.

3. Using Semiotics in Segmentation Design

Before continuing with a classification of signs in segmentation design, it is worthwhile to sketch how such knowledge can affect and guide this process. A conceptual drawing of a semiotics-driven design process for image segmentation interfaces is given in Figure 2. The first step in this design process is to situate the segmentation task in a wider clinical context that can inform each element. For this, one first identifies three classes of information regarding the segmentation process in general:

1. Constant information that is common across all segmentation problems encountered by the interface, (for example, if the interface is single purpose say for lung segmentation, constant information could be that there are two lungs which require segmentation, each being a single connected region)

2. Input information that can affect the results of the individual segmentation and thus must be provided by the user, (this information could include the location of different anatomical objects which vary from image to image due to patient variability and disease) and
3. Output information that the user expects to be available from their clinical tool both during the segmentation process and after completing it, (for example: a visualization of the object, anatomical measurements as in disease monitoring, or possible diagnoses as in computer-assisted diagnosis systems).

Having an in-depth understanding of these types of information is necessary for selecting the appropriate algorithms and interaction mechanisms for performing the segmentation task. Constant information is characterized by being unchanging across patients and thus reflects aspects of the segmentation process that can be built into or be a by-product of the underlying segmentation algorithm. (For example, if an object of interest is known to be a single connected component anatomically, this information could be used to constrain the space of available components encouraging algorithms like region growing, which are constrained to connected objects, or processing steps such as connected-component analysis, which can augment processes that do not naturally adhere to this constraint.) Input information, such as the precise size or spatial location of different organs of interest, varies from patient to patient. This information could be used to affect the segmentation process, because it reflects how the individual patient varies, and must be provided as user input. From an algorithmic point-of-view, this type of information must be incorporated into the underlying segmentation algorithm in an extendable manner, incorporating all possibilities, which can make it more difficult to effectively incorporate. The difference between constant information and input information may indicate which aspects of the segmentation process may be preserved between interactions and which may have to be recalculated, thus limiting performance. Output information can reflect how the segmentation process fits into a larger clinical goal, ensuring that the segmentation process adequately reflects the rationale for using segmentation at all. Extracting these types of information, regardless of how they are communicated, is the basis of traditional requirements engineering, and not necessarily semiotics. In addition, output information presented during the segmentation process should give the user insight into how the process is operating or provide intermediate results, which can be much more semiotically rich, fundamentally interacting with and inducing user interaction. These may overlap with output information expected of the completed segmentation process, but not necessarily.

For the last two types of information, some form of communication must take place between the user and the computer. Thus, after such information is identified, the process for its communication must also be designed and thus understanding the signs mediating this communication is of value. Choosing these communication mechanisms requires both an understanding of the information being provided by any individual sign, and coherence between these signs. Lastly, there is the more technical goal of merging the information provided by the user with the constant information, the aim of many papers in algorithm development. We will return to this diagram later in Section 5 to illustrate how semiotic considerations may help to investigate the application and integration of different signs into a segmentation process.

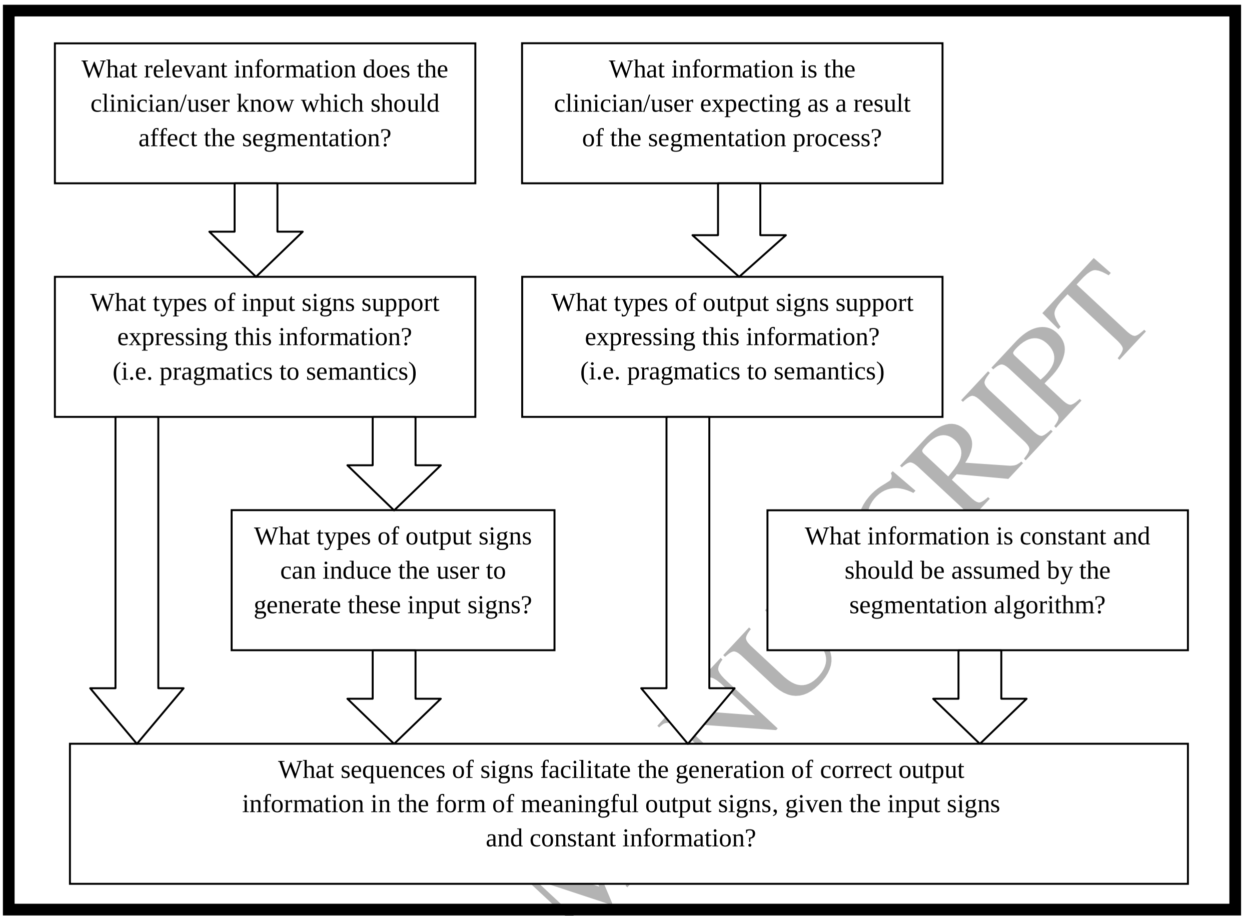


Figure 2: Conceptual process for interactive segmentation design based on sign exchange

4. Peircean Semiotics and a Taxonomy of Signs in Segmentation

The philosophic views of Charles Saunders Peirce are still the centerpiece of an entire school of semiotic discourse a century and a half after they were first proposed in 1867-8 (Peirce, 1868). To reiterate an earlier definition, a sign is a *signifier* which represents an *object* other than itself by virtue of some *interpretant*. This division of a sign into a signifier, an object, and an interpretant is one of the basic claims underlying Peircean semiotics, appearing through his discourse (Peirce, 1868) (Peirce, Hartshorne, & Weiss, 1933). Aside from this division, Peirce's most enduring semiotic classification is that expressing the relationship between the sign vehicle and the represented object: (Peirce, Hartshorne, & Weiss, 1933)

1. *Icons* whose representational power is derived from likeness,
2. *Indices* whose representational power is derived from a common existential or causal quality, and
3. *Symbols* whose representational power is a derived from socio-cultural conventions and, from the stand-point of the objects in question, could be considered arbitrary.

To give concrete examples, an icon could be how a photograph of a particular object represents that object. Fundamentally, icons require the sign to instantiate a number of properties that it shares with the object. The representational power doesn't arise from the fact that the picture and the object are causally related with the camera acting as an intermediate, but that the picture is a likeness of the object. An example of an index could be how a weathervane represents the direction of the wind. In this case, there is no visible quality that unites the fabric of the weathervane with the invisible air, but there is a causal relationship between the motion and position of the weathervane and that of the air. Indices fundamentally indentify or specify a property of an object, rather than instantiate said properties. Symbols are rife in that most natural language is symbolic. To use a non-linguistic example, symbols, like stop signs, have a particular meaning or indicate a particular behaviour by virtue of a convention that may vary between communities.

Although not part of this trichotomy, another useful sign category is that of the *part* whose representational power is derived solely from being a part of the object being represented (Sebeok, 2001). (This type of sign is commonly called a *symptom* in the semiotics literature, due to the early relationship between semiotics and medicine (Sebeok, 2001) but will be called a *part* or *partial sign* here in order to prevent confusion between it and the current medical definition of the word *symptom*.) For example, if one views a branch of a tree through a window, one could interpret that branch as signifying that particular entire tree. Similarly, a fragment of ancient pottery could signify the entire vessel. A more abstract example of a partial sign is how one or more samples from a probability distribution can signify the distribution as a whole, or how the elements of a mathematical set can signify the set as a whole. From a purely theoretical point of view, partial signs are a subset of the indexical in that there is a causal/existential relationship between them (e.g. something must have broken the entire vessel to create the fragment). However, they represent a very distinct relation, that of a part-whole relationship, which may be best separated from those when the signifier and object are related by causal chain or the specification of a property. A visual representation of this extended taxonomy is given in Figure 3. Sections 4.1 and 4.2 elucidate this classification in the context of medical image segmentation for both input and output signs respectively.

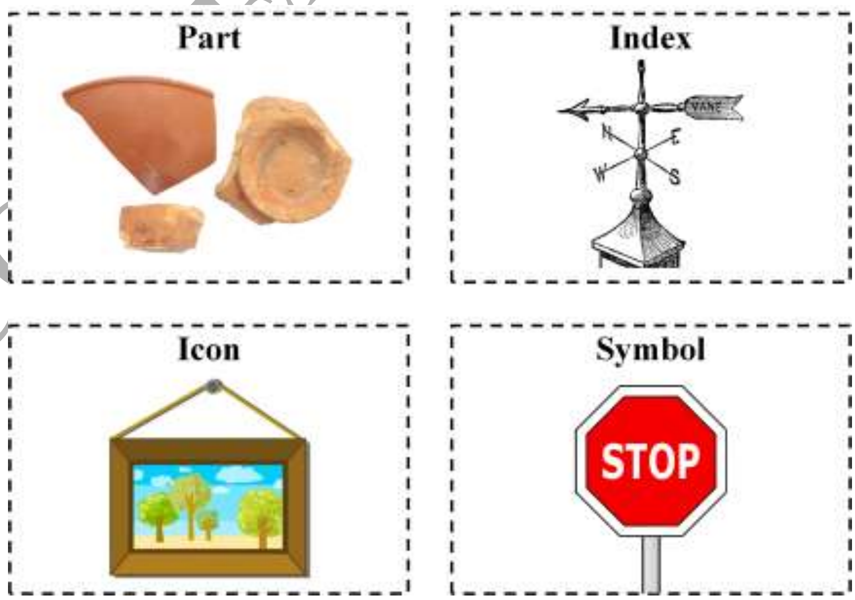


Figure 3: Extended Peircean taxonomy of signs

Peirce recognised early on that many signs are not homogenous, that they may contain aspects that can be classified differently (Peirce, Hartshorne, & Weiss, 1933). In this account, he further decomposes the object and interpretant based on the point in the overall semiotic process. With respect to the object, he differentiates between the *dynamic object,* which is the intended object represented by the sign as a whole, and the *immediate object,* which is some informationally-incomplete or intermediate object in the semiotic chain. Interpretants are similarly separated into *immediate, dynamic,* and *final* which reflect differences in how the sign is interpreted based on contextual information. In this account, it is unclear what the multiplicity, or number of objects and interpretants, of the sign is, and Pierce considered all signs to be potentially if not actually infinite. Not considering these distinctions is the source of the common view in human-computer interaction that the only signs being exchanged are symbolic (Nake & Grabowski, 2001) arising from the notion of the computer purely as a deterministic symbol-processing machine, which isolates the immediate, purely digital, object. This reductionist view may be suitable for some forms of interaction, such as pressing a button or typing a sentence, but it is difficult to apply to many modern input methods, such as taking a picture from a web-cam, that are not functionally symbolic. For pragmatic reasons, Sections 4.1 and 4.2 classify input and output signs based on their dynamic, rather than immediate, object, that is, according its primary method of denoting the object. As suggested by the dialogue partner perspective, this considers human-computer signs equivalently as if they concerned two humans. To clarify using an example, specifying a location in a segmented object will be considered a partial sign regardless of it is implemented as a mouse-click or inputting its numeric co-ordinates. Section 4.3 will extend this classification scheme to capture notions of sign heterogeneity, formulating it in a finite mechanism that can be more readily used for the design and analysis of human-computer interaction. Section 4.4 will discuss machine learning in segmentation, which has so far moved further towards automaticity, and how it may later incorporate more semiotically rich interactive components.

4.1 Input Signs

As stated in Section 2, input signs are indications given by the user to the computer in order to initialize a segmentation (in the context of semi-automated methods) or correct/edit a segmentation (in the context of interactive methods). In this context, we are referring to the end-user, not the algorithm designer or programmer who constructs the interface. These input signs should not be confused with training methods in general, as many fully-automated segmentation algorithms (by definition involving few to no input signs) rely extensively on training sets of a priori segmented images.

The types of input signs provided in interactive segmentation have been previously investigated (Olabarriaga & Smeulders, 2001) although not in a rigorous, Peircean perspective. This prior categorization into numeric, pictorial, and menu-based types, leaves a number of out-of-the-box input methods without a comfortable category in a way that a formal semiotic approach can avoid. Examples of different input signs in each of the proposed semiotic categorization are given Table 1.

Partial Input Signs	Iconic Input Signs
• Seed-point picking (Adams & Bischof, 1994) (Herman & Carvalho, 2001) and paint-brushing (Boykov & Jolly, 2001) (Yushkevich, et al., 2006) • Contouring and edge selection (Yushkevich, et al., 2006)(Barrett & Mortensen, 1998) (Poon, Hamarneh, & Abugharbieh, 2008) • Full segmentation of individual slices (in a volumetric dataset) (Schenk, Prause, & Peitgen, 2000)	• Atlas/template selection (Nowinski, Yang, & Yeo, 2000) • Pre-segmentation (Yin, et al., 2010)
Indexical Input Signs	**Symbolic Input Signs**
• Specifying object/label hierarchies (Delong & Boykov, 2009) (Delong, Gorelick, Veksler, & Boykov, 2012) (Rajchl, et al., 2015) (Baxter J. S., Rajchl, Peters, & Chen, 2015) (Baxter J. S., Rajchl, McLeod, Yuan, & Peters, 2017) • Specifying geometric properties (Gulshan, Rother, Criminisi, Blake, & Zisserman, 2010) • Specifying volumetric properties (Qiu, et al., 2014) • Co-segmentation and specifying deformation/resolution properties (Han, et al., 2011) (Bagci, Udupa, Yao, & Mollura, 2012) (Guo, et al., 2015)	• Stop/converge signals (Yushkevich, et al., 2006) • Algorithm selection as a whole (Koenig, et al., 2006) (Ritter, et al., 2011)

Table 1: Classification and Examples of Input Signs

In terms of input signs, one unintuitive action to consider is that the user selects the segmentation algorithm itself. Although this is largely implicit in many segmentation frameworks, some image analysis interfaces such as MeVisLab (Koenig, et al., 2006) (Ritter, et al., 2011) make this type of input sign explicit by allowing the user to create their own on-the-fly segmentation pipeline with basic processes such as thresholding and morphological operators, as connected boxes. These actions, the selection and modification of the segmentation pipeline as a whole, are largely symbolic in that the naming and graphical conventions dictate the segmentation pipeline. Similarly, user actions such as choosing an algorithm to run immediately, outside of the notion of a pipeline, could be viewed as symbolic. 3D Slicer (Pieper, 2004) allows this through the selection of modules and data to which a selected module applies.

Considering actions that don't modify the segmentation pipeline, arguably, the most common input signs used in modern segmentation frameworks are partial in that the user supplies a sign indicating a part of the segmentation. Common techniques include the picking of seed-points typical in traditional region growing (Adams & Bischof, 1994) (Hojjatoleslami & Kittler, 1998) and fuzzy connectedness (Udupa & Samarasekera, 1996)

(Herman & Carvalho, 2001) based frameworks. These single point methods can readily be extended to seed regions with higher density using paint-brush based approaches (Boykov & Jolly, 2001) (Boykov & Funka-Lea, 2006) (Grady, 2006). The previous signs discussed signified regions interior to the segmentation, but there are contour based approaches, such as LiveWire (Barrett & Mortensen, 1998), which rely on identifying portions of the region's boundary (Poon, Hamarneh, & Abugharbieh, 2008). ITKSnap (Yushkevich, et al., 2006) is particularly illustrative in that it includes both a paint-brush tool and a polygonal outlining tool more conceptually suitable for identifying edges. In the case of segmenting volumetric data, some segmentation methods rely on a full segmentation of individual 2D slices which are then propagated or interpolated throughout the remainder of the volume (Schenk, Prause, & Peitgen, 2000). To some extent, pure partial input signs are equivalent to Olabarriaga & Smeulders' pictorial input in that they specify locations in the image's spatial domain. (Olabarriaga & Smeulders, 2001) Although, as shown by the example in Section 4.3 in which samples from a distribution are considered partial signs for the distribution itself, partial signs are more general-purpose and do not have to be constrained to specifying image domain locations.

Iconic input signs arise when the user provides an entire complete segmentation (as opposed to a partial one) in terms of either the same image or an atlas. In terms of most atlas-based segmentation paradigms, the atlas is a fixed component and not subject to modification by the user. One illustrative counter-example is that developed by Nowinski *et al.* (2000) in which the user actively selects the atlas with which to segment the image. Separate from atlases is the concept of pre-segmentation in which the user provides a rough segmentation of the image which is then refined by the algorithm. This is often used as a pre-processing step for more complex methods designed to preserve the segmentation topology such as LOGISMOS (Yin, et al., 2010), although often these approaches are generated automatically rather than directly through user action.

The last category of input signs also happens to be the most conceptually diverse. Indexical input signs range from the selection of numeric parameters to the specification of descriptive label properties. The selection of some numeric parameters is indexical in that they specify or control properties of the segmentation in a direct way. For example, the selection of a threshold controls a property of the segmentation (minimum and/or maximum intensity). The number of segmentation interfaces that employ these signs are so numerous that it is difficult to find ones that are at their core parameter free. Olabarriaga & Smeulders (2001) dedicate an entire input category to numeric parameters.

However, indexical input signs do not all take the form of numeric parameters. The specification of segmentation properties is a relatively new and diverse collection of methods that is, in the authors' opinion, unappreciated. For example, recent work in min-cut and max-flow segmentation approaches have taken advantage of label orderings in the form either of hierarchies (Delong, Gorelick, Veksler, & Boykov, 2012) (Rajchl, et al., 2015) (Baxter J. S., Rajchl, Peters, & Chen, 2015) and more general groupings (Delong & Boykov, 2009) (Baxter J. S., Rajchl, McLeod, Yuan, & Peters, 2017) which can give the user control over how labels can be understood as the accumulation of simpler objects, although only one interface currently allows it as an run-time input sign separate from the segmentation algorithm as a whole (Baxter J. S., Rajchl, Peters, & Chen, 2015). Similarly, other convex/sub-modular optimization-based approaches have developed frameworks in which the user can control specific properties of the segmentation such as the object volume (Qiu, et al., 2014), geometric properties such as shape constraints (Gulshan, Rother, Criminisi, Blake, & Zisserman, 2010), or deformation/resolution differences across modalities (Han, et al., 2011) (Bagci, Udupa, Yao, & Mollura, 2012) (Guo, et al., 2015) in a largely indexical manner.

But what does this classification tell us about segmentation interfaces? One thing that it clearly indicates is a preference in terms of the interactive segmentation community for the symbolic, partial, and numerically indexical forms of interaction, with iconic and other indexical signs filling very niche roles. This is concerning as it is fundamentally opposed to the current paradigm in teaching anatomy which is almost solely dependent on the iconic (e.g. diagrammatic/pictorial representations of anatomy in textbooks) and non-numeric indexical (e.g. the description of the properties of the anatomy/physiology). Exploring these forms of input sign may offer a fruitful area of research, bridging the gap between how anatomy is taught and how it is practiced in the context of medical image segmentation (Yoon, 2003). With the increasing popularity of optimization based segmentation, there has been a research focus into incorporating novel geometrical or topological constraints into existing extendable segmentation paradigms which may dove-tail into the examination of more general indexical input signs in which these constraints may be communicated by the user.

4.2 Output Signs

Similar to the previous section, we begin with the classification of output signs, generated by the computer to represent the current segmentation, with a brief list of examples as presented in Table 2. Similar to partial input signs, partial output signs are common throughout medical volume segmentation interfaces. Displaying slices with labels identified using designated colours is likely the most common form with displaying object contours as a close second. These output signs are well-suited for pairing with their corresponding input signs in that both input and output could be simultaneously displayed on a single two-dimensional 'canvas' (Baxter J. S., Rajchl, Peters, & Chen, 2015) (Top, Hamarneh, & Abugharbieh, 2011).

The difference between iconic and partial output signs in these examples can be interpreted largely as a matter of dimensionality. That is, for volumetric image segmentation, labeled/contoured slices may give all of the information about the segmentation for that individual slice, which is only part of the entire volume. Thus, these output types may be interpreted as partial, rather than iconic where an imperfect but volumetric representation is given. In volumetric imaging, iconic output signs rely on some computer graphics processing to display a view of the segmentation in three dimensions. Isosurface rendering, where labelled volumes are transformed into three-dimensional polyhedra are common due to the prevalence of graphics libraries which inherently support this type of surface rendering (Yushkevich, et al., 2006) (Poon, Hamarneh, & Abugharbieh, 2008) (Baxter J. S., Rajchl, Peters, & Chen, 2015) (Top, Hamarneh, & Abugharbieh, 2011). More recently, advances in direct volume rendering has provided an alternative mode of three-dimensional visualization which is more suitable for segmentations in which there is a defined notion of local segmentation uncertainty (Landaeta, La Cruz, Baranya, & Vidal, 2015).

Partial Output Signs	Iconic Output Signs
• Coloured/labeled slices (Boykov & Jolly, 2001) (Boykov & Funka-Lea, 2006) (Grady, 2006) (Yushkevich, et al., 2006) (Baxter J. S., Rajchl, Peters, & Chen, 2015) (Top, Hamarneh, & Abugharbieh, 2011) • Contoured slices (Yushkevich, et al., 2006) (Barrett & Mortensen, 1998) (Poon, Hamarneh, & Abugharbieh, 2008) (Top, Hamarneh, & Abugharbieh, 2011)	• Isosurface rendering (Yushkevich, et al., 2006) (Poon, Hamarneh, & Abugharbieh, 2008) (Baxter J. S., Rajchl, Peters, & Chen, 2015) (Top, Hamarneh, & Abugharbieh, 2011) • Direct volume rendering (Landaeta, La Cruz, Baranya, & Vidal, 2015)
Indexical Output Signs	**Symbolic Output Signs**
• Energy/fitness functional values (Torsney-Weir, et al., 2011) • Property maps such as the Freesurfer (Fischl, 2012) cortical thickness and curvature maps	• Active learning indications or behaviours (Top, Hamarneh, & Abugharbieh, 2011) (Veeraraghavan, 2011)(Park S. H., 2014)(Pace, 2015) • Potential diagnoses (i.e. computer-assisted diagnosis)

Table 2: Classification and Examples of Output Signs

Examples of indexical and symbolic output signs are much less common in medical image segmentation with good reason. What benefit is there in providing information about a computer vision or image processing task aside from a direct representation of the result? What benefit may be gleaned from a possibly unintuitive level of indirection? For some metrics, such as segmentation uncertainty or a measurement of accuracy, reporting them directly may still give the user some insight into the more invisible facets of the segmentation result. A key example of an indexical output sign outside of simply reporting a metric can be found in the framework presented by Torsney-Weir *et al.* (2011) in which multiple segmentation results are shown simultaneously, organized by the values of the parameters used and the values of associated quality metrics (the Response view) and energy functional values (the Pareto Panel). The distinct benefit of this approach over a more traditional, partial and iconic approaches is that it allows for the selection of numerical parameters in an intuitive fashion, assessing the geometry of the parameter space and allowing for the exploration of multiple functionals in the case of optimization-based segmentation. Alternatively, the computer may provide abstract views of particular properties of the segmentation that may be of interest. For example, FreeSurfer (Fischl, 2012) provides inflated views of the segmented cortex which are coloured using the cortical thickness or curvature at the corresponding point.

Symbolic output signs could be based on actions that the computer performs in order to guide or inform the user. These actions should be seen as symbolic because their relationship to the segmentation itself is only interpretable through some convention, such as active learning (Top, Hamarneh, & Abugharbieh, 2011) (Veeraraghavan, 2011). Symbolic output signs are likely to be of increasing importance to machine learning based segmentation processes which rely heavily on inferring structure and content from user interaction. Specifically, symbolic output signs may provide the computer with some mechanism to explain or indicate some facet of its internal state in a way that is accessible to the human user and allows the user to create input that directly targets the computer's deficits. Top et al.'s (2011) as well as Veeraraghavan & Miller's(2011) active learning frameworks communicate the computer's uncertainty regarding a segmentation specifically by localizing it, showing the user where it would benefit most from interaction. Pace et al. (2015) take a similar approach, localizing segmentation error as a comparative gold standard. This form of guided interaction will hopefully become more prevalent with advances in machine learning, leading to a dialogue with more accessible semantic content.

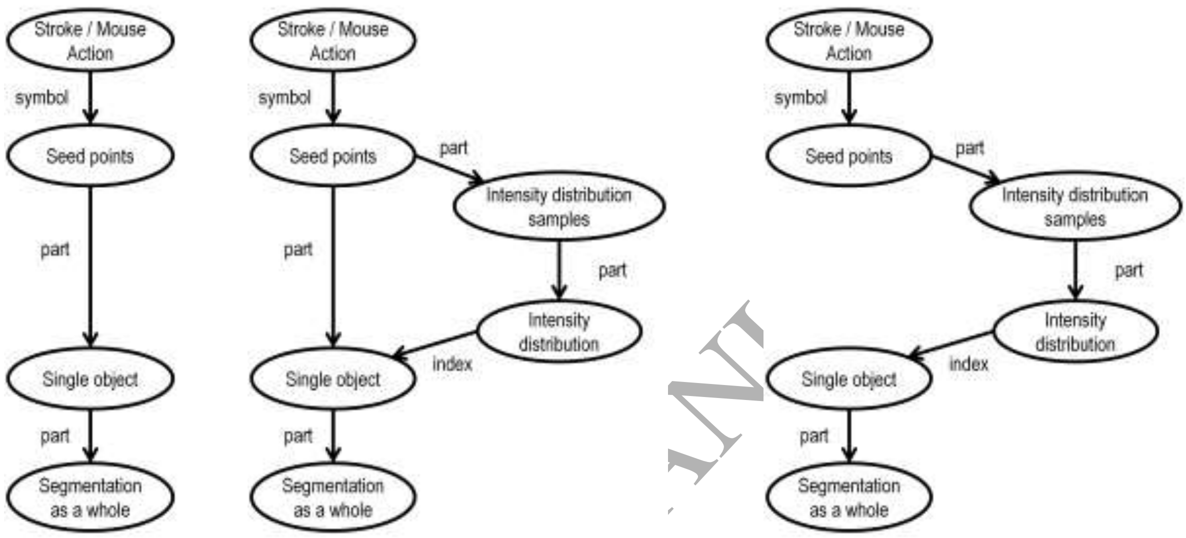
The distinction between indexical and symbolic output signs (as with all classifications of an entire sign) is necessarily blurry, but this should not weaken the argument that both of these classes of output sign are under-used in medical image segmentation interfaces, especially those designed for later algorithm design. Indexical output signs for example, offer a distinct possibility for pairing with indexical input signs (i.e. numeric parameter selection) in a more natural and intuitive way that the traditional method of varying parameters by repeated trial-and-error with little-to-no visualization support. One natural conclusion from this analysis is there should be some level of overlap between output and input signs with the computer producing output that can be used to more directly guide future user input.

4.3 Sign Graphs and Complex Interpretants

Although in Peirce's original classification (Peirce, 1868), the specified sign types are distinct, in practice the complexity of the underlying semiotic process and often the lack of in-depth computational knowledge of the users imply that elements in the taxonomy are not always mutually exclusive. In both computational and more general semiotic investigation, signs are often given multiple levels of meaning or mechanisms whereby this meaning is conveyed and this heterogeneity leads to a lack of distinctness in the typology of the overall sign. These considerations can be elucidated and clarified by explaining exactly how the signs are interpreted by both the computer and the user, but this can be very complex if left unstructured. Peirce recognized the problem of multiple levels of interpretation in his later semiotic accounts (Peirce, Hartshorne, & Weiss, 1933), but ultimately did not provide a singular framework suitable for modern field of human-computer interaction. Peirce's notion of *infinite semiosis* in which signs have an infinite constellation of meanings weakens its ability to be used in a human-computer stand-point where interpretations must happen in a finite time.

Thus, a finite description of interpretation, a *finite semiosis*, must be formulated which is sufficiently powerful to capture heterogeneity and multiple levels of meaning but compact enough to be quickly constructed and analyzed. Our chosen representation is a *sign graph* in which a sign can be decomposed into a finite series of simpler connected signs. In these graphs, each directed edge represents a simple interpretant connecting a sign vehicle to an object. The benefit of this particular representation is, for the majority of computational roles, that the sign graph used is explicitly known, encoded in the implementation of the segmentation algorithm. The creation of sign graphs to structure new segmentation algorithms and interfaces display echoes of traditional object-oriented design with computation being represented as an abundance of interacting, yet distinct, objects.

Sign graphs can be readily used to examine the heterogeneity of a particular sign equipped with more information as to how it is interpreted. For example, if a user segments an individual slice, which the computer uses as initialization for a 2D algorithm for the adjacent slices, this can be expressed by a sign graph with two partial edges (the segmented slice to the segmentation as a whole and the adjacent slice to the segmentation as a whole) and an iconic edge (the segmented slice to the adjacent slice) showing the heterogeneity inherent in a more complete description of the input sign.



(a) Novice interpretation of a paintbrush stroke

(b) More advanced interpretation of a paintbrush stroke with spatial and indexical components

(c) Interpretation of a paintbrush stroke solely as intensity sampler

Figure 4: Sign graphs representing three different interpretations of the same paintbrush stroke sign.

These sign graphs can be used also to represent the user's interpretant as well as that of the computer. Figure 4(a) demonstrates a novice user's possible interpretation of a single brush-stroke in which some action of the mouse (holding down a key and moving) is symbolically translated into the placement of a collection of seed points that are used to denote spatial parts of a single object in the segmentation. This may however be somewhat different to how a computer interprets the same action. For example, if the computer uses those seed points as spatial anchors (a partial sign interpretation) as well as samples of the single object's intensity distribution (a more heterogeneous sign with partial and indexical components), the interpretation used by the computer is more complex and can be represented as the sign graph in Figure 4(b). As the user gains experience or knowledge of the system, the user's interpretant may grow in complexity, ideally aligning itself with that of the computer. A third interpretation, Figure 4(c), was used by Rajchl *et al.* (2012) in which the seed points serve only to estimate the intensity distribution and not as a spatial anchor for the segmentation as a whole.

In terms of human-computer interaction, a fundamental aspect of design is the process of aligning *interface metaphors* which are mental models of how a computational system acts (Erikson, 1995). In this framework, all forms of communication between the human and the computer are seen as a metaphor in which the source and target domains are the elements of the user's and computer's sign graphs. These sign graphs thus capture a notion of the difference between novice and expert users by adding to or subtracting extraneous elements from the user's sign graph as said user intuits the behaviour and mechanisms of the computer. Similarly, if the computer employs mechanisms for learning the user's intention (as described by Olabarriaga & Smeulders (2001) as an *indirect role* of the computer, opposed to providing a level of abstraction over the underlying algorithms, which is described as a *direct role*), the computer’s sign graph may approach that of the user, although such a graph may not be explicitly represented in the computer.

The notion of a sign graph necessarily complicates the classification of input and output signs into parts, icons, indices, and symbols in that each sign graph may be heterogeneous and include multiple sign types. By varying these signs graphs with only slight steps, it is possible to blur the line between the classifications for the sign as a whole. This heterogeneity has its advantages; it subsumes the view of all interaction with a computer as being symbolic through the frequent occurrence of symbolic simple sign connecting the user’s physical action, such as moving the mouse, to an initial computational abstraction, the placement of seed points. However, unlike the purely symbolic approach, it encourages greater depth of explanation into how the computational processes can be understood by the user. Through the use of sign graphs in design, it also provides the designer a tool for probing the mental processes of the user, the side-effect being the better interpretation of the user by the computer. One day, this process might even be automated itself with computer programs learning the intention behind user input.

4.4 Semiotics and Machine Learning in Segmentation

With the rise of new machine learning techniques, the notion of interaction is of particular importance in designing segmentation processes. An overview of the state of the art would give the impression that many researchers in machine learning based segmentation have focused on increased automaticity through automatically learning more and more abstract structures, rather than relying on those structures being communicated by a user. Indeed, traditional supervised machine learning paradigms rely on being presented with correctly annotated segmentations. If training is performed interactively, the computer is ultimately treating these annotations as either icons, representing a collection of features via their instantiation, or as parts, representing samples from a complex distribution depending on the role of machine learning in the larger segmentation task. These more traditional modes of machine learning are currently the algorithms of choice including a number of segmentation processes that leverage deep learning (Ronneberger, 2015)(Zhang, 2015)(Chen, 2016). This mode of interaction is likely going to change if deep learning frameworks are going to be used in an interactive setting for the simple reason that training in purely deep learning architectures is by far the most computationally expensive task, and if the only form of interaction the user has on this network is to add more training samples, this computational expense may be infeasible. Note that this does not imply that deep learning or other frameworks which are computationally expensive to train will fall by the wayside, just that they will become parts of a larger interactive framework in which the focus of interaction is elsewhere or extensive technical work will need to be done to salvage interactivity.

One example of this technical work is the approach used by Wang et al. (2017) in which two neural networks are employed: one for providing an initial automatic segmentation and a second for refining segmentations based on partial user input. Both networks involve a prior training datasets however, which can limit their capability to learn interactively and thus become a locus for interaction. In terms of learning interactively, one technical breakthrough is One-Shot-Learning (Fei-Fei, 2006) in which reasonable inferences are made from limited sampled data. The benefit of this approach is that determining training samples could become interactive (predicated on the training being computationally efficient) as only a few need to be selected. Zero-shot learning from image classification may also fundamentally change interaction in segmentation as whole new classes could be added without training examples but instead via some form of indexical description. (Socher, 2013) Another element of deep learning architectures that may improve their chance of participating as an interactive element lies in the concept of *transfer learning*. Transfer learning is when a machine learning architecture is trained on one dataset but then retrained on another, often replacing the final few layers, in order to take advantage of learnt low-level structure common between the datasets (Pan, 2010)(Oquab, 2014). As only the last few layers need to be trained, user modification of the training samples and training process becomes much more feasible. An alternative possible mode of interaction is for the user to interactively change the error function to account for some more abstract knowledge about the intended segmentation. For example, Ben Taieb and Harmarneh (2016) produced a fully convolutional neural network in which the final layer is trained using a topologically informed error function. One can imagine this topology to be a locus for an indexical type of interaction similar to previous work in label orderings (Delong & Boykov, 2009)(Baxter J. S., Rajchl, Peters, & Chen, 2015)(Baxter J. S., Rajchl, McLeod, Yuan, & Peters, 2017). These avenues for interaction have yet to be explored but may allow for more semiotic richness in how humans interact with computers equipped with these types of machine learning.

5. Semiotics-Driven Segmentation Design

Returning to the discussion in Section 3, how does an understanding of signs and sign graphs facilitate the design of segmentation interfaces? Aside from the informational concerns specified in Section 3 which are the subject of traditional requirements engineering, semiotic concerns can also be used to guide development, specifically through the process of aligning interface metaphors (Erikson, 1995). For this, we can take advantage of Thagard and Holyoak's (1989) constraint satisfaction theory, which includes metrics to evaluate the efficacy of metaphors, to motivate three ways in which semiotics should guide development, viewing the act of interpretation of any sign as similar to an analogy between the interpretants made by both the user and the computer. For current interactive segmentation interfaces, the computer's interpretant for any given sign is explicit, and is encoded in the interface's implementation with the *a priori* knowledge of the interface designer.

The first major concern is that of structural consistency between the user and computer's interpretation of the sign. Note that this does not imply that the user or the computer has to have a full understanding of the myriad ways in which the sign can be interpreted; Olabarriaga & Smeulders' (2001) notion of the computer's direct role in computational abstraction is still essential. However, both the user and the computer require a shared path through the sign graph that allows the sign to have a partially predictable effect. In the example given in Figure 4, all of the nodes and edges in Figure 4(a) also appear in Figure 4(b) indicating a relatively high degree of structural consistency though not the absolute highest, given that some nodes in Figure 4(b) lack a corresponding node in Figure 4(a). Importantly, there is a shared path through the sign graph, e.g. that the seed points are considered inside the object, indicating some common ground for the user and computer to understand each other. This is in contrast to Figure 4(a) with Figure 4(c) in which no shared path exists and thus there is a low degree of structural similarity. One can readily see users becoming frustrated, especially when using the paintbrush as a mechanism for correcting erroneous *regions* in the segmentation. We have experimentally verified this under a more controlled scenario in Appendix A. Similarly, an output sign should engender an interpretation in the user that incorporates key elements and relations used by the computer in the signs generation.

The second concern we raise here is the notion of semantic similarity, that is, the similarity between the individual pairs of corresponding nodes or edges in the two sign graphs. For example, if both the user and computer interpret seed points as definitely part of the object of interest, that portion of the analogy has higher semantic similarity than if either the user or the computer viewed those points as only heuristic and not definitive. There is a wide body of thought that believes this is essential to the success of symbolic signs in human-computer interaction based of the *interface metaphor* and epitomized in common HCI constructs such as the desktop (Gatsou, 2011) (Macaranas, Antle, & Ricke, 2012). In the context of medical image segmentation, the same considerations apply for symbolic input signs, but the notion of semantic similarity is less well-defined for abstract constructs (such as seed points, distributions, algorithms etc…) that form the majority of a sign graph in modern interactive segmentation interfaces. Semantic and structural consistency are different in that structural similarities refers to how well a mapping between the user's and computer's interpretants can be created, regardless of the similarity of corresponding nodes, which is the purview of semantic similarity.

The last concern is that of pragmatic centrality, that is, how well the designed sign can support a particular high-level action. Olabarriaga & Smeulders (2001) provide some concept of these high-level actions, at least in terms of input signs, in their description of the role of the user. These roles center around a particular action, such as judging a segmentation as adequate or not, correcting a segmentation, building a segmentation process, etc… The question of pragmatic centrality, from the perspective of input signs, is does the mapping between the user and computer’s interpretation of a sign support these actions well? If a user's intention is to place a dense series of seed points, a paintbrush mechanism is simply more pragmatic, more suitable for that use, than say, individually picking each pixel which would be more time-consuming. From the output sign perspective, one has to evaluate whether or not an output sign effectively expresses the computer’s goal. For example, displaying cortical thickness information is easier on a spherical map, where the entire cortex can be visualized without worrying about crevices or tortuosity, than it would if it were painted directly onto an image slice.

As stated earlier, the computer's interpretant of any given sign is encoded in the implementation of the interface and thus can be seen as explicitly known by the interface designer. Thus, modifying and improving the computer's interpretant is a natural place to begin the process of aligning the interpretants according to structural consistency, semantic similarity, and pragmatic centrality. In order to do so, one must elicit sign graphs from users, that is, get a detailed understanding of how the user interprets and understands the problem domain. In knowledge communities, such as medicine, a large amount of domain information can be gathered from sources such as textbooks and clinical operating procedures, which can be used to obtain a general concept of elements in an idealized user's sign graph. This process may also furnish developers with the specialized terms and meanings associated with a particular field of medicine and what knowledge and tasks can be fundamentally support and be supported by medical image segmentation. It may even provide an understanding of the constant, input, and output information required in the segmentation problem. For understanding actual users, however, traditional human-computer interaction techniques which focus on an understanding of the user's mental state, such as the think aloud protocol (Lewis, 1982) should be used. Think aloud protocols have been previous used to elucidate fundamental differences between idealized and actual clinical users in a variety of fields including oncology (Jaspers, Steen, Van Den Bos, & Geenen, 2004), pulmonology (Linder, et al., 2006), and electronic health record management (Farri, et al., 2012). Note that this protocol on its own may not be sufficient for elucidating the entire sign graph under a variety of scenarios, but may have to be augmented with other techniques (Jaspers, 2009).

It is worth noting that modifying the interface and implementation is not the only mechanism for aligning sign graphs. Effective user training addresses the problem of aligning interpretants through a modification of the user's behaviour and mental state, rather than the computer's. Thus, in a semiotics driven approach, developing effective user training and explanatory manuals is an essential aspect of interface design that cannot be discarded or seen as separate. For many clinicians, continuing clinical education is a recognized portion of their practice, offering an avenue in which user training can be incorporated, provided it is recognized as efficiently and effectively aligning the user's and computer's interpretants.

By understanding the interaction as a co-ordinated series of sign graphs forming an analogy between the user and computer interpretants, conceptual tools from human-computer interaction and the cognitive science of analogies can be leveraged simultaneously with Peircean categorization. This combination subsumes several disparate views on segmentation design from the literature (specifically that of Olabarriaga & Smeulders (2001)) and from traditional human-computer interaction (the notion of the interface metaphor) and offers a series of fundamental questions which elucidate and critique the design of segmentation interfaces.

6. ITKSnap and MeVisLab as Case Studies in Semiotic Analysis

In order to make the use of semiotic analysis in medical image segmentation more concrete, two case studies on commonly used frameworks are presented. ITKSnap (Yushkevich, et al., 2006) is a well-known interface for general-purpose volumetric image segmentation supporting a variety of medical imaging file types. (The specific version used here is version 2.2.0, although version 3.6.0 involves largely the same interaction mechanisms.) ITKSnap is also equipped with a semi-automatic segmentation tool based on the active contour algorithm using a level-set based implementation. MeVisLab (Ritter, et al., 2011) takes a fundamentally different approach, emphasizing the creation of segmentation pipelines. (The specific version used here is version 2.8.2.) In our semiotic analysis, the first subsection will enumerate and briefly describe the input and output mechanisms provided to the user with their basic semiotic classification. The second subsection will contrast the approaches and their philosophy towards a general segmentation task from both a semiotic and information-centric perspective.

6.1 Enumeration of Input and Output Signs in ITKSnap

ITKSnap provides two types of input signs, one that edits the segmentation directly and another that specifies a semi-automated active contour algorithm. Both use a single label selection widget in order to choose a single label on which the input is applied. The segmentation editing input mechanisms for ITKSnap include a:

- **2D paintbrush tool** that allows users to paint or erase parts of a specified label on the individual slide. The 2D paintbrush is a fairly unambiguous, purely partial input sign.
- **2D contour tool** that allows users to add the interior of a polygonal or user-drawn contour to a pre-specified label. This mechanism involves three stages; the placement of vertices in the polygon prior to its closure, revising the placement of vertices/edges after closure, and the acceptance/rejection of the final polygon. Ultimately, the contour tool overall is in the collection of partial input signs, either modifying a part of the polygon (for intermediate stages) or the segmentation.
- **3D spray-paint tool** that allows user to assign the surface of a foreground label to another specified label where painted. This occurs in two stages, a painting stage and an acceptance/rejection stage. The 2D paintbrush is a fairly unambiguous, purely partial input sign. The acceptance/rejection stage is purely symbolic. This tool is particularly useful for defining walls or skins around other labels.
- **3D scalpel tool** that allows user to cut off a portion of the foreground object and assign that portion to another specified label. This occurs in two stages, a cut placement stage and an acceptance/rejection stage. This sign is indexical in that the user specifies a unambiguous spatial property for the segmentation to have which is more global than just specifying a part of the segmentation.

From a semiotic point-of-view, the segmentation editing tools for ITKSnap are much simpler than the snake tool, involving sign graphs with relatively few components and a single route for interpretation. Thus, one can assume they will be popular amongst users, especially novice users, as they require little training. The developers of ITKSnap have intuited this and thus provide one page (of ten) in their manual on all the segmentation editing tools compared to three pages on the final tool; the snake tool.

The semi-automatic active contour input mechanism (**snake tool**) includes:

- A **region-of-interest (ROI) sub-tool** where the user defines a rectangular region on the image on which to run the algorithm. The sides of the ROI can be re-positioned by the user via dragging.
- A **preprocessing sub-tool** where the user specifies how the image is transformed to be more amenable to active contour segmentation. This tool contains a series of numeric sliders which specify the parameters for one of two different simple pre-processing types.
- An **initialization sub-tool** which allows "bubbles" to be added which act as an initial pre-segmentation which is iteratively refined by the algorithm. The original labeling is automatically included in the pre-segmentation for version 2.2.0, but that has been removed for version 3.6.0.
- An **iterator sub-tool** which allows the user to advance the refinement process. The numeric parameters for the refinement operation as well as the exact algorithm used can also be set in a separate pop-up dialogue.

The snake tool overall is a fairly complex sign with many interacting components, this level of heterogeneity preventing the comfortable placement of the snake tool as a whole into a single sign category. The computer's interpretant of the snake tool is given as a sign graph in Figure 5. For the snake tool, the level of interactivity supported is relatively high, given that it is advertised as automatic. The user's ability to pause the iteration process and change its parameterization can be considered a limited degree of interactive feedback to the algorithm, although no direct segmentation editing is available within the tool itself. Overall, the sign graph representing this tool is complex which may leave it open to a number of misinterpretation issues. Additionally, it involves a number of symbolic components which are fundamentally more opaque to novice users, limiting their ease-of-use. One possible issue involves the initialization tool which combines user placed bubbles with the original segmentation to create a pre-segmentation with which to initialize the iterative algorithm. A novice user may interpret that input mechanism overall as a simple partial sign, rather than multiple partial signs combined with an iconic overall sign, which may lead it to be used sub-optimally or incorrectly. One way to correct this would be to, instead of placing bubbles, allow the user to directly edit the segmentation using the 2D segmentation editing tools, making it unambiguous that the original segmentation is a part of the pre-segmentation. Alternatively, the snake tool, by eliminating the region of interest and initialization sub-tools (and consolidating the pre-processing and iteration sub-tools into one) could be used as a much simpler iterative segmentation refinement tool. It would also encourage the tool to be used more in tandem with the other segmentation editing tools rather than as a stand-alone.

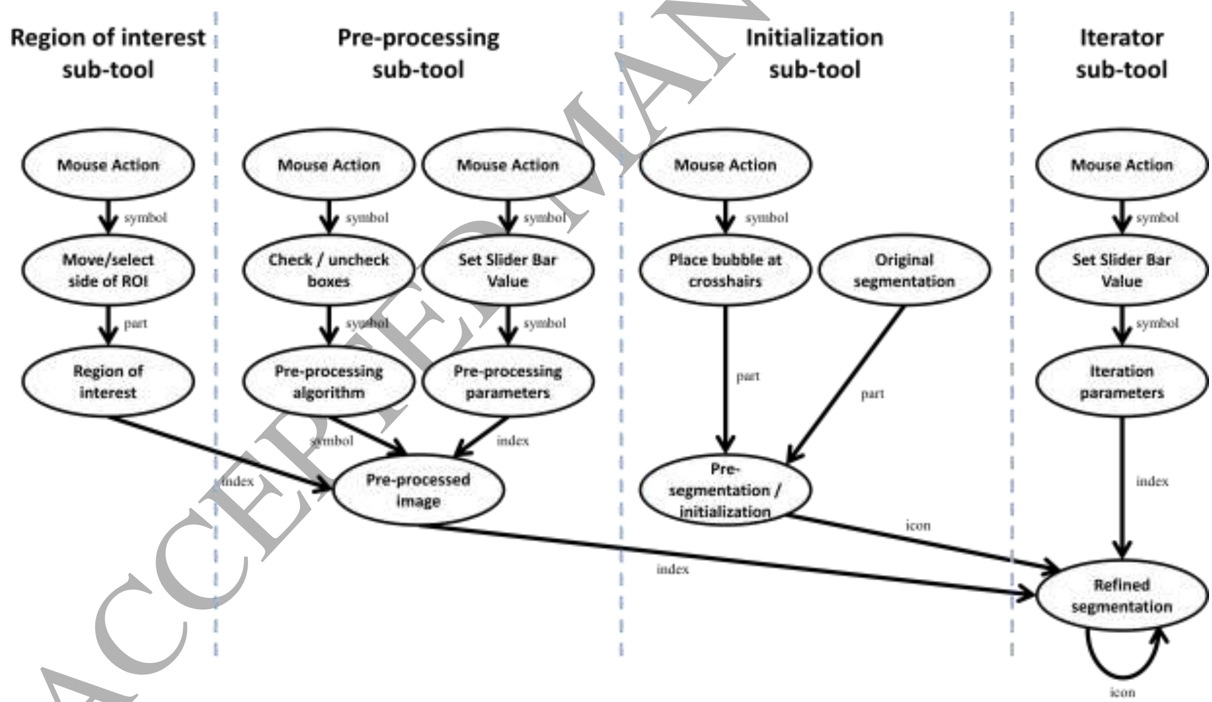


Figure 5: Sign graph used by the computer for interpreting the snake tool input mechanism.

One interesting aspect of each input mechanism is that its effect is largely isolated to a single label. For the snake tool, for example, only one label region is advanced, rather than multiple. For segmentation editing, this may not be an issue, but for the more automated segmentation algorithms, the level of expressivity with respect to multiple regions is severely hampered.

In terms of output mechanisms, ITKSnap offers three, each a fairly unambiguous case of partial, iconic, and indexical sign respectively:

- **Orthogonal slices** that, in the main interface, shows the currently labeling, and is used for showing intermediate representations for the segmentation editing and snake tools.
- **Isosurface rendering** that displays the segmented regions in 3D as meshes. For the 3D segmentation editing tools, intermediate results such as the location of the spray-paint and the cutting plane are also displayed in this rendering.
- **Volume and statistics panel** that shows a table of the volume, mean intensity and standard deviation of intensities for each label including the background.

ITKSnap actively pairs input mechanisms with some immediate feedback output mechanism. For example, all the segmentation editing tools are applied to the canvas on which the results are displayed (instantaneously for the 2D tools and after an update button is pressed for the 3D tools).

For the snake tool, intermediate representations are given some level of support. This includes the parameterization, in which some indexical output sign (e.g. a graph of the threshold function) or iconic output sign (e.g. demonstration of the effect on a toy segmentation) are provided. Although not directly reflecting the results of differences in intermediate objects on the final segmentation, these mechanisms still provide feedback to the user about the intermediate objects themselves. Outside of intermediate steps, however, there is no notion of persistence shown to the user. That is, a user cannot readily tell what input had what particular effect on the segmentation, with relatively little capability to undo. This input with immediate feedback mechanism present in most of ITKSnap's tools can be seen as a dialogue with much more limited scope within the greater dialogue. For the snake tool in particular, the tool as a whole expresses, using largely symbolic mechanisms, information about approximate segmentations in a way that is not well aligned with the user's knowledge and therefore a dialogue is necessary to make sure the computer and user agree on the interpretation. The scope of the dialogue within a dialogue is on the interpretation of the tool's behaviour, not directly about the segmentation problem itself.

6.2 Informational Assumptions Underlying ITKSnap

The last stage in our analysis is to consider the semiotic analysis to distil what information the segmentation interface assumes is constant (across segmentation tasks in general, being a general interface), a topic of input (varying across segmentation tasks and images), and a desired output (of the segmentation process). That is, As stated earlier, ITKSnap involves an active contour algorithm implemented using level sets that should also been seen as a result of the informational assumptions made by the interface. For the purposes of brevity, we will only detail a few assumptions which differ from a number of other general-purpose segmentation interfaces in the literature. In terms of constant information, ITKSnap appears to make the assumption that:

- *Segmented objects do not interact with each other*. Each input mechanism and algorithmic choice in ITKSnap, as indicated earlier, acts upon an individual label at a time. This is in contrast to approaches that make use of coupled regions.

- *Segmented objects, if automaticity is desired, are defined primarily by either homogenous intensity regions or strong boundary contrast but do not simultaneously require both.* The snake tool pre-processing steps only favour these two options with no ready mechanism for combining the two.

In terms of information varying across different segmentation tasks, ITKSnap assumes that:

- *Segmentation tasks can involve more than one label.* ITKSnap's label selector allows for a large number of labels to be specified and selected from, and the label editor allows for an arbitrary number of labels to be added.
- *General segmentation tasks likely involve a large amount of segmentation editing.* Even in ITKSnap's tutorials on automated segmentation, segmentation editing is a crucial component, often falling at the end of the segmentation process.
- *Segmentation problems may involve multiple greyscale or RGB images.* ITKSnap allows for multiple images to be put into a single visualization space using overlays. This gives the user the option to include multiple images to support a more complex segmentation pipeline.

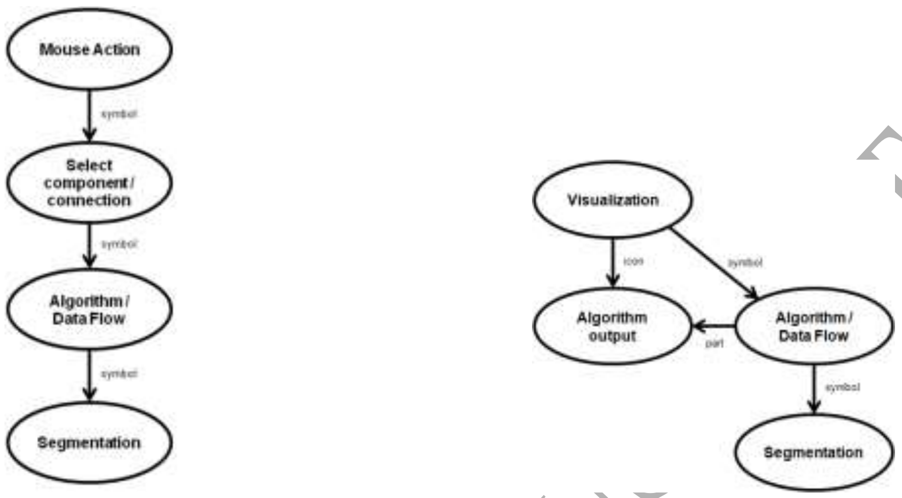
In terms of the goal of segmentation in a broader clinical context, ITKSnap makes relatively few assumptions other than that it will rarely be used in generating a result other than the segmentation itself. ITKSnap's export functionalities ensure that the segmentation can be extracted in multiple formats that inherently support different types of downstream processing, but supports very little in terms of advanced visualization, decision support, or other medical image processing operations. The operations it does support (volume, mean intensity, and standard deviation of intensity) are useful in some contexts, but are likely chosen more for their simplicity that completeness.

6.3 Enumeration of Input and Output Signs in MeVisLab

The primary mode of input allowed in MeVisLab involves the creation of pipelines. This comes in two modes: the placement/connection of pipeline components in the canvas which defines the pipeline, and the selection of parameters for each individual component. The variety of pipeline components include several basic segmentation operators (thresholding, seeded region growing, fuzzy clustering, etc...) with a number of other image processing operators either specifically for segmented images or for images in general. A potential issue with this approach is that the mechanism for placing and connecting pipeline components is entirely symbolic, thus relying heavily on the user's formal knowledge about image processing rather than underlying intuition about the interface. This may close off MeVisLab's use in clinical contexts as it draws from and emphasizes the user's knowledge of medical image processing rather than of the medical images being processed. The sign graph for this type of interaction is given in Figure 6 (a) noting that it is purely symbolic.

The user has the capability of creating different interaction mechanisms as a part of the constructed pipeline using the Open Inventor framework.

MeVisLab offers a large variety of output signs, the majority of which being filtered through some sort of iconic process. That is, for MeVisLab to display an indexical sign, e.g. the intensity distribution of an object, the precise formal meaning of that process must be symbolically constructed (i.e. pipeline components for segmentation and histogram calculation must be placed and connected appropriately) and the output of that process viewed as an icon. One benefit of this is that output is highly structured yet much more expressive with the possibility of viewing almost all of the computer's intermediate representations.



(a) Input mechanisms in MeVisLab (b) Output mechanisms in MeVisLab

Figure 6: Sign graph for majority of MeVisLab input mechanisms (a) and output mechanisms (b).

Unlike ITKSnap's most semiotically complex tool, the snake tool, MeVisLab keeps the input and output mechanisms have a high degree of semiotic simplicity, necessary for exposing the entirety of the underlying algorithmic process. The node "Algorithm / Data Flow" present in both sign graphs can largely been seen as a placeholder for a much more complex embedded sign graph that represents the algorithmic components in the particular pipeline being constructed and the intention of the user in choosing each of said components. Perhaps in future, MeVisLab will grow in complexity beyond this simplification and have the computer develop a deeper understanding of user intent from the pipeline components being chosen.

6.4 Informational Assumptions Underlying MeVisLab

MeVisLab's assumptions are entirely different from ITKSnap's. Having a much wider array of processing algorithms at the ready, the informational assumptions in terms of constant information are few. The effect that MeVisLab's design decisions have on the input and output mechanisms are much more interesting. In terms of input information, MeVisLab assumes that:

- *Segmentation tasks are primarily differentiated by the largely automated algorithms chosen to address them.* Unlike ITKSnap, MeVisLab never presents a single tool or collection of tools as being sufficient to address all segmentation tasks, but emphasizes the construction of appropriate tools above their use.
- *Automaticity is usually desirable and possible.* MeVisLab's design (and additional support for scripting) makes applying the same pipeline to a large series of images relatively straightforward and involves little manual effort.

- *Segmentation problems may involve multiple greyscale or RGB images.* MeVisLab is similar to ITKSnap in that it encourages the use of multiple images for a single segmentation task. MeVisLab, however, allows these additional images to be intermediate steps in the segmentation process rather than solely as input.

In terms of output information and the role of segmentation in a larger clinical context, MeVisLab assumes that:

- *The desired output of a segmentation task should be a facet of the segmentation process.* MeVisLab's library of analysis tools gives the user a much higher degree of expressivity in terms of potential output for indexical signs, and its visualization tools indicate the same for traditional iconic visualization.
- *The desired output of a segmentation task can have multiplicity.* Similar to the above point, there is no limit as to the number and type of output information provided, which encourages output information to be seen as multiplicitous, rather than singular.

6.5 Discussion

ITKSnap offers an interesting case for semiotic analysis because of how popular it is in the medical image segmentation research community as a tool for more efficient, if primarily manual, segmentation tasks. Analyzing it from the point of view of interactivity illustrates that ITKSnap aims for a balance between primarily automated segmentation with inherent segmentation editing capabilities, although the degree of automaticity is lower than the impressive suite of manual segmentation editing tools. Specifically analyzing the snake tool in terms of the level of interaction it provides illustrates its benefits and gives a potential guide as to how the tool could be improved.

MeVisLab takes the polar opposite approach from ITKSnap in terms of its interaction philosophy, using the algorithmic basis for medical image processing as the locus of interaction as opposed to aspects of the medical image itself. In that sense, the ideal user population for MeVisLab may not necessarily be radiologists and other clinicians, but medical software developers. On the other hand, it provides a number of tools that can be readily used to address individual patient variability and pathology, rather than a one-size-fits-all algorithmic basis which should resonate more with the medical community which can identify and take advantage of these differences.

Semiotic analysis focused on the interaction mechanisms in each interface provides interesting insights into how the two frameworks differ on a fundamental level in a way that is more expressive than enumerating the different algorithms supported by each. (That is, one could imagine an ITKSnap-looking interface with the same algorithmic capabilities of MeVisLab or vice versa, but the semiotic analysis of the two would still have validity.) By narrowing in on the fundamental differences in interactivity between these dramatically different segmentation interfaces, one can more readily analyze the philosophies and assumptions underlying their different conceptions of medical image processing.

7. Conclusions

The intention of this article was to extend the analysis performed by Olabarriaga & Smeulders (2001) on more philosophically rigorous grounds. We have done so by advocating for the use of conceptual tools in semiotics to examine both the seminal and the state-of-the-art in medical image segmentation. Thus, we initially motivate segmentation as a process of sign exchange in which to frame a semiotic approach, illustrate the use of sign classification in Peircean semiotics to classify simple signs and derive complex sign graphs which can aid the process of interactive segmentation design.

By considering segmentation as sign exchange between a human user and the segmentation interface, the automaticity spectrum can be readily reconstructed, emphasizing manual and fully-automatic segmentation as the edge-cases of a continuous framework. Viewing segmentation as a form of sign exchange, similar to current semiotic trends in human computer interaction, places an emphasis on understanding the structure and meaning of the communication between the user and the computer. This communication includes input signs, provided by the user in order to convey information or to edit the segmentation, and output signs, provided by the computer to convey the current segmentation or facets of the segmentation process.

The trichotomous nature of Peircean semiotics, dividing into the *iconic, indexical, and symbolic*, and the older semiotic notion of the *partial* furnish this perspective with a taxonomy applicable to both simple input signs and output signs. More nuanced meanings can be attributed to particular input and output mechanisms by considering them as a complex of simple signs that defines their interpretation from both a user and computer perspective. In addition, interpreting these signs as analogies leads to three lenses for evaluating their quality, through structural consistency, semantic similarity, and pragmatic centrality (Holyoak & Thagard, 1989). These lenses can be used as heuristic guides for segmentation process design at a more abstract level than traditional human-computer interaction.

With this semiotic tool set, clinicians and medical image processing researchers can expand their understanding of medical image segmentation. This semiotic approach uses well established analysis techniques in philosophy and human-computer interaction to expose paradigmatic undercurrents in how medical image segmentation is used and understood. Using this approach, segmentation interfaces can be understood in terms of the structured dialogue between a user and the segmentation interface that characterizes the segmentation process. Our hope is that similar investigations into the philosophy of medical image processing are conducted, yielding insights into how the ways we interact with medical images can move forward in fundamentally new and more effective directions.

Acknowledgements

The authors would like to acknowledge John Moore and Jonathan McLeod for their discussion and proof-reading. J.S.H. Baxter is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) Alexander Graham Bell Scholarship program.

Bibliography

Adams, R., & Bischof, L. (1994). Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 16*(6), 641-647.

Armato, S. G., & Sensakovic, W. F. (2004). Automated lung segmentation for thoracic CT: Impact on computer-aided diagnosis. *Academic Radiology, 11*(9), 1011-1021.

Bagci, U., Udupa, J. K., Mendhiratta, N., Foster, B., Xu, Z., Yao, J., . . . Mollura, D. J. (2013). Joint segmentation of anatomical and functional images: Applications in quantification of lesions from PET, PET-CT, MRI-PET, and MRI-PET-CT images. *Medical image analysis*, 929-945.

Bagci, U., Udupa, J. K., Yao, J., & Mollura, D. J. (2012). Co-segmentation of functional and anatomical images. *Medical Image Computing and Computer-Assisted Intervention.*

Barrett, W. A., & Mortensen, E. N. (1998). Interactive live-wire boundary extraction. *Medical image analysis, 1*(4), 331-341.

Bates, M. J. (1990). Where should the person stop and the information search interface start? *Information Processing & Management*, 575-591.

Baxter, J. S., Rajchl, M., McLeod, A. J., Yuan, J., & Peters, T. M. (2017). Directed Acyclic Graph Continuous Max-Flow Image Segmentation for Unconstrained Label Orderings. *International Journal of Computer Vision*.

Baxter, J. S., Rajchl, M., Peters, T. M., & Chen, E. C. (2015). Optimization-based interactive segmentation interface for multi-region problems. *SPIE Medical Imaging.*

BenTaieb, A., & Hamarneh, G. (2016). Topology aware fully convolutional networks for histology gland segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 460-468). Athens, Greece: Springer.

Boykov, Y. Y., & Jolly, M. P. (2001). Interactive graph cuts for optimal boundary \& region segmentation of objects in ND images. *IEEE International Conference on Computer Vision.*

Boykov, Y., & Funka-Lea, G. (2006). Graph cuts and efficient ND image segmentation. *International journal of computer vision, 70*(2), 109-131.

Chen, L.-C. G. (2016). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *arXiv preprint*, arXiv preprint arXiv:1606.00915.

Clarke, L. P., Velthuizen, R. P., Camacho, M. A., Heine, J. J., Vaidyanathan, M., Hall, L. O., . . . Silbiger, M. L. (1995). MRI segmentation: methods and applications. *Magnetic Resonance Imaging*, 343-368.

Cohadon, F. (1994). Images and icons. *Acta neurochirurgica*, 8-13.

Deely, M. A., Chen, A., Datter, R. D., Noble, J., Cmelak, A., Donnelly, E., . . . Dawant, B. M. (2013). Segmentation editing improves efficiency while reducing inter-expert variation and maintaining accuracy for normal brain tissues in the presence of space-occupying lesions. *Physics in Medicine and Biology*, 4071-4097.

Delong, A., & Boykov, Y. (2009). Globally optimal segmentation of multi-region objects. *IEEE International Conference on Computer Vision.*

Delong, A., Gorelick, L., Veksler, O., & Boykov, Y. (2012). Minimizing energies with hierarchical costs. *International journal of computer vision, 100*(1), 38-58.

Erikson, T. D. (1995). "Working with interface metaphors.". In *Readings in Human-Computer Interaction* (pp. 147-151). San Francisco, California, USA: Morgan Kaufmann Publishers.

Farri, O., Pieckiewicz, D. S., Rahman, A. S., Adam, T. J., Pakhomov, S. V., & Melton, G. B. (2012). A qualitative analysis of EHR clinical document synthesis by clinicians. *AMIA Annual Symposium Proceedings.*

Fei-Fei, L. R. (2006). One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 594-611.

Fischl, B. (2012). FreeSurfer. *Neuroimage , 62*(2), 774-781.

Gatsou, C. A. (2011). Text vs visual metaphor in mobile interfaces for novice user interaction. *Information Services & Use, 31*(3), 271-279.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science, 7*(2), 155-170.

Goubran, M., Hammond, R. R., de Ribaupierre, S., Burneo, J. G., Mirsattari, S., Steven, D. A., . . . Khan, A. R. (2014). Magnetic resonance imaging and histology correlation in the neocortex in temporal lobe epilepsy. *Annals of Neurology, 77*(2), 237-250.

Grady, L. (2006). Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*(11), 1768-1783.

Gulshan, V., Rother, C., Criminisi, A., Blake, A., & Zisserman, A. (2010). Geodesic star convexity for interactive image segmentation. *IEEE Conference on Computer Vision and Pattern Recognition.*

Guo, F., Yuan, J., Rajchl, M., Svenningsen, S., Capaldi, D. P., Sheikh, K., . . . Parraga, G. (2015). Globally Optimal Co-Segmentation of Three-dimensional Pulmonary 1 H and Hyperpolarized 3 He MRI with Spatial Consistence Prior. *Medical Image Analysis*.

Han, D., Bayouth, J., Song, Q., Taurani, A., Sonka, M., Buatti, J., & Wu, X. (2011). Globally optimal tumor segmentation in PET-CT images: A graph-based co-segmentation method. *Information Processing in Medical Imaging*, (pp. 245-256).

Heckel, F., Moltz, J. H., Meine, H., Geisler, B., Kiessling, A., D'Anastasi, M., . . . Hahn, H. K. (2014). On the evaluation of segmentation editing tools. *Journal of Medical Imaging*, 034005-034005.

Herman, G. T., & Carvalho, B. M. (2001). Multiseeded segmentation using fuzzy connectedness. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23*(5), 460-474.

Hinton, G. E.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 1527-1554.

Hojjatoleslami, S. A., & Kittler, J. (1998). Region growing: a new approach. *IEEE Transactions on Image Processing, 7*(7), 1079-1084.

Holyoak, K. J., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive science, 13*(3), 295-355.

Huang, Y., Dmochowski, J. P., Su, Y., Datta, A., Rorden, C., & Parra, L. C. (2013). Automated MRI segmentation for individualized modeling of current flow in the human head. *Journal of Neural Engineering, 10*(6).

Jaspers, M. W. (2009). A comparison of usability methods for testing interactive health technologies: methodological aspects and empirical evidence. *International journal of medical informatics, 78*(5), 340-353.

Jaspers, M. W., Steen, T., Van Den Bos, C., & Geenen, M. (2004). The think aloud method: a guide to user interface design. *International journal of medical informatics, 73*(11), 781-795.

Juneja, P., Harris, E. J., Kirby, A. M., & Evans, P. M. (2012). Adaptive breast radiation therapy using modeling of tissue mechanics: A breast tissue segmentation study. *International Journal of Radiation Oncology Biology Physics, 84*(3), 419-425.

Kammersgaard, J. (1986). Four different perspectives on human–computer interaction. *International Journal of Man-Machine Studies*, 343-362.

Kanade, T. (1980). Region segmentation: signal vs semantics. *Computer Graphics and Image Processing*, 279-297.

Koenig, M., Spindler, W., Rexilius, J., Jomier, J., Link, F., & Peitgen, H.-O. (2006). Embedding VTK and ITK into a visual programming and rapid prototyping platform. *SPIE Medical Imaging.*

Landaeta, L., La Cruz, A., Baranya, A., & Vidal, M.-E. (2015). SemVisM: semantic visualizer for medical images. *SPIE International Symposium on Medical Information Processing and Analysis.*

Lewis, C. (1982). *Using the" thinking-aloud" method in cognitive interface design.* IBM TJ Watson Research Center.

Linder, J. A., Rose, A. F., Palchuk, M. B., Chang, F., Schnipper, J. L., Chan, J. C., & Middleton, B. (2006). Decision support for acute problems: The role of the standardized patient in usability testing. *Journal of biomedical informatics, 39*(6), 648-655.

Macaranas, A., Antle, A. N., & Ricke, B. E. (2012). Bridging the gap: attribute and spatial metaphors for tangible interface design. *Sixth International Conference on Tangible, Embedded and Embodied Interaction.*

Manual, N. T. (1986). *Hart, SG and Staveland, L.* Hanover, Maryland, USA: NASA Center for Aerospace Information (CASI).

McGuinness, K., & O'Connor, N. E. (2010). A comparative evaluation of interactive segmentation algorithms. *Pattern Recognition*, 434-444.

Moonis, G., Liu, J., Udupa, J. K., & Hackney, D. B. (2002). Estimation of tumor volume with fuzzy-connectedness segmentation of MR images. *American Journal of Neuroradiology, 23*(3), 356-363.

Nake, F., & Grabowski, S. (2001). Human–computer interaction viewed as pseudo-communication. *Knowledge-Based Systems*, 441-447.

Nowinski, W. L., Yang, G. L., & Yeo, T. T. (2000). Computer-aided stereotactic functional neurosurgery enhanced by the use of the multiple brain atlas database. *IEEE Transactions on Medical Imaging, 19*(1), 62-69.

Olabarriaga, S. D., & Smeulders, A. W. (2001). Interaction in the segmentation of medical images: A survey. *Medical image analysis*, 127-142.

Oquab, M. L. (2014). Learning and transferring mid-level image representations using convolutional neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 1717-1724).

Pace, D. F. (2015). Interactive whole-heart segmentation in congenital heart disease. *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 80-88). Munich, Germany: Springer.

Pan, S. J. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 1345-1349.

Park, H., Bland, P. H., & Meyer, C. R. (2003). Construction of an abdominal probabilistic atlas and its application in segmentation. *IEEE Transactions on Medical Imaging, 22*(4), 483-492.

Park, S. H. (2014). "Interactive prostate segmentation using atlas-guided semi-supervised learning and adaptive feature selection. *Medical Physics*, 111715-1–111715-10.

Peirce, C. S. (1868). On a new list of categories. *Proceedings of the American Academy of Arts and Sciences*, 287-298.

Peirce, C. S., Hartshorne, C., & Weiss, P. (1933). *The Collected Papers of Charles Sanders Peirce.*

Peng, B., Zhang, L., & Zhang, D. (2013). A survey of graph theoretical approaches to image segmentation. *Pattern Recognition*, 1020-1038.

Peters, T., & Cleary, K. (2008). *Image-guided interventions: technology and applications.* Springer Science & Business Media.

Pham, D. L., Xu, C., & Prince, J. L. (2000). Current methods in medical image segmentation 1. *Annual review of biomedical engineering*, 315-337.

Pieper, S. M. (2004). 3D Slicer. *IEEE International Symposium on Biomedical Imaging: Nano to Macro*, (pp. 632-635).

Poon, M., Hamarneh, G., & Abugharbieh, R. (2008). Efficient interactive 3D Livewire segmentation of complex objects with arbitrary topology. *Computerized Medical Imaging and Graphics, 32*(8), 639-650.

Prastawa, M., Bullitt, E., Ho, S., & Gerig, G. (2004). A brain tumor segmentation framework based on outlier detection. *Medical image analysis, 8*(3), 275-283.

Qiu, W., Rajchl, M., Guo, F., Sun, Y., Ukwatta, E., Fenster, A., & Yuan, J. (2014). 3D Prostate TRUS Segmentation Using Globally Optimized Volume-Preserving Prior. *Medical Image Computing and Computer-Assisted Intervention.*

Rajchl, M., Baxter, J. S., McLeod, A. J., Yuan, J., Qiu, W., Peters, T. M., & Khan, A. R. (2015). Hierarchical Max-Flow Segmentation Framework For Multi-Atlas Segmentation with Kohonen Self-Organizing Map Based Gaussian Mixture Modeling. *Medical Image Analysis*.

Rajchl, M., Yuan, J., Ukwatta, E., & Peters, T. M. (2012). Fast interactive multi-region cardiac segmentation with linearly ordered labels. *9th IEEE International Symposium on Biomedical Imaging (ISBI)* (pp. 1409-1412). IEEE.

Ritter, F., Boskamp, T., Homeyer, A., Laue, H., Schwier, M., Link, F., & Peitgen, H.-O. (2011). Medical image analysis. *IEEE Pulse, 2*(6), 60-70.

Ronneberger, O. P. (2015). U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 234-241). Munich, Germany: Springer.

Schenk, A., Prause, G., & Peitgen, H.-O. (2000). Efficient semiautomatic segmentation of 3D objects in medical images. *Medical Image Computing and Computer-Assisted Intervention.*

Scholl, I., Aach, T., Deserno, T. M., & Kuhlen, T. (2011). Challenges of medical image processing. *Computer science-Research and development*, 5-13.

Sebeok, T. A. (2001). *Signs: An introduction to semiotics.* Toronto: University of Toronto Press.

Silva, S., Santos, B. S., Madeira, J., & Silva, A. (2010). A 3D tool for left ventricle segmentation editing. *Image Analysis and Recognition*, 79-98.

Smistad, E., Falch, T. L., Bozorgi, M., Elster, A. C., & Lindseth, F. (2015). Medical image segmentation on GPUs--A comprehensive review. *Medical Image Analysis*, 1-18.

Socher, R. G. (2013). Zero-shot learning through cross-modal transfer. *Advances in neural information processing systems* , 935-943.

Top, A., Hamarneh, G., & Abugharbieh, R. (2011). Active learning for interactive 3D image segmentation. *Medical Image Computing and Computer-Assisted Intervention*, (pp. 204-213).

Torsney-Weir, T., Saad, A., Moller, T., Hege, H.-C., Weber, B., Verbavatz, J., & Bergner, S. (2011). Tuner: Principled parameter finding for image segmentation algorithms using visual response surface exploration. *IEEE Transactions on Visualization and Computer Graphics, 17*(12), 1892-1901.

Turkbey, B., Mani, H., Aras, O., Rastinehad, A. R., Shah, V., Bernardo, M., . . . Pinto, P. (2012). Correlation of magnetic resonance imaging tumor volume with histopathology. *The Journal of Urology, 188*(4), 1157-1163.

Udupa, J. K., & Samarasekera, S. (1996). Fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation. *Graphical models and image processing, 58*(3), 246-261.

Veeraraghavan, H. a. (2011). Active learning guided interactions for consistent image segmentation with reduced user interactions. *IEEE International Symposium on Biomedical Imaging: From Nano to Macro* (pp. 1645-1648). IEEE.

Wang, G., Zuluaga, M., Li, W., Pratt, R., Patel, P., Aertsen, M., . . . Vercauteren, T. (2017). DeepIGeoS: A Deep Interactive Geodesic Framework for Medical Image Segmentation. *arXiv preprint*, arXiv:1707.00652.

Wiemker, R., Rogalla, P., Blaffert, T., Sifri, D., Hay, O., Shah, E., . . . Fleiter, T. (2014). Aspects of computer-aided detection (CAD) and volumetry of pulmonary nodules using multislice CT. *British Journal of Radiology, 78*(1).

Yin, Y., Zhang, X., Williams, R., Wu, X., Anderson, D. D., & Sonka, M. (2010). LOGISMOS—layered optimal graph image segmentation of multiple objects and surfaces: cartilage segmentation in the knee joint. *IEEE Transactions on Medical Imaging, 29*(12), 2023-2037.

Yoon, W. C. (2003). Task-Interface Matching: How we may design user interfaces. *15th International Ergonomics Association Triennial Congress.*

Yuan, J., Bae, E., & Tai, X.-C. (2010). A study on continuous max-flow and min-cut approaches. *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2217-2224). IEEE.

Yushkevich, P. A., Piven, J., Hazlett, H. C., Smith, R. G., Ho, S., Gee, J. C., & Gerig, G. (2006). User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. *NeuroImage, 31*(3), 1116-1128.

Zhang, W. L. (2015). Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage*, 214-224.

Zheng, B., Chang, Y.-H., & Gur, D. (1995). Computerized detection of masses in digitized mammograms using single-image segmentation and a multilayer topographic feature analysis. *Academic Radiology, 2*(11), 959-966.

Appendix A: Experimental Verification of Semiotic Principles

In order for a theoretical framework to be useful, it must have both explanatory and predictive potential. The explanatory potential of the semiotic framework has already been demonstrated via case studies. Section 5 describing the semiotic design process indicates some possible hypothesis that lead to predictive capabilities. To summarize Section 5, an interface should be easier to use provided that the sign graphs of the user and the computer for a particular tool align well, that is, they display a high degree of the three relational qualities described by Holyoak & Thagard (1989):

1. *Structural consistency:* the sign graphs contain a large number of corresponding nodes and edges, especially those that form a path through the graphs from the initial interaction to the output information,
2. *Semantic similarity:* the corresponding nodes in the sign graph are themselves similar, that is, the individual objects in the graph are understood similarly between the user and the computer, and
3. *Pragmatic centrality:* the degree to which an interpretation in the sign graph can support high level input information.

Structural consistency and the notion of *correspondence* has been the main focus in the literature regarding quantifying the efficacy of metaphors, either as a desirable property (Holyoak & Thagard, 1989) or as a necessary condition (Gentner, 1983). In addition, structural consistency is in a sense easier to heuristically evaluate than the other two properties which require either more in-depth knowledge of the user's mental model or the specifics of the anatomy being segmented. Thus, in order to show the predictive potential of our framework, we designed an experiment which attempts to isolate the structural consistency between the user's mental model and the computer's operation, measuring its impact on the interaction quality.

Methods

To evaluate the predictive capability of the semiotic framework, three visually identical interactive segmentation interfaces were created. These interfaces differed only in terms of how the input is interpreted by the computer, specifically through the sign graphs shown in Figure 4(a-c). The interface used a contouring mechanism for user input and the segmentation algorithm employed continuous max-flow (Yuan, Bae, & Tai, 2010) which provided spatial regularization. The spatial anchoring was achieved through Euclidean distance data terms that penalize areas being further away from the user provided input. The intensity information was incorporated by calculating the intensity distribution within the user-provided contours, using Bayes' theorem to create meaningful data terms. The mixed model (Figure 4(b)) used a weighted summation of both as its data term.

The interface displayed an icon of a desirable segmentation as well as a random image of the same anatomy taken from the same modality and contrast. The icon was created to give the participants a clearer understanding of the desired segmentation by way of an example. A screen shot of the interface is shown in Figure A1. Each trial consisted of segmenting a single two-dimensional cardiac MR image. The images are taken from the same patient as the icon, but from different slices at different parts of the cardiac cycle.

22 novice participants were recruited for this study. Each participant was asked to perform six trials using the segmentation interfaces. (Not all participants were able to perform all six tasks.) The order in which the

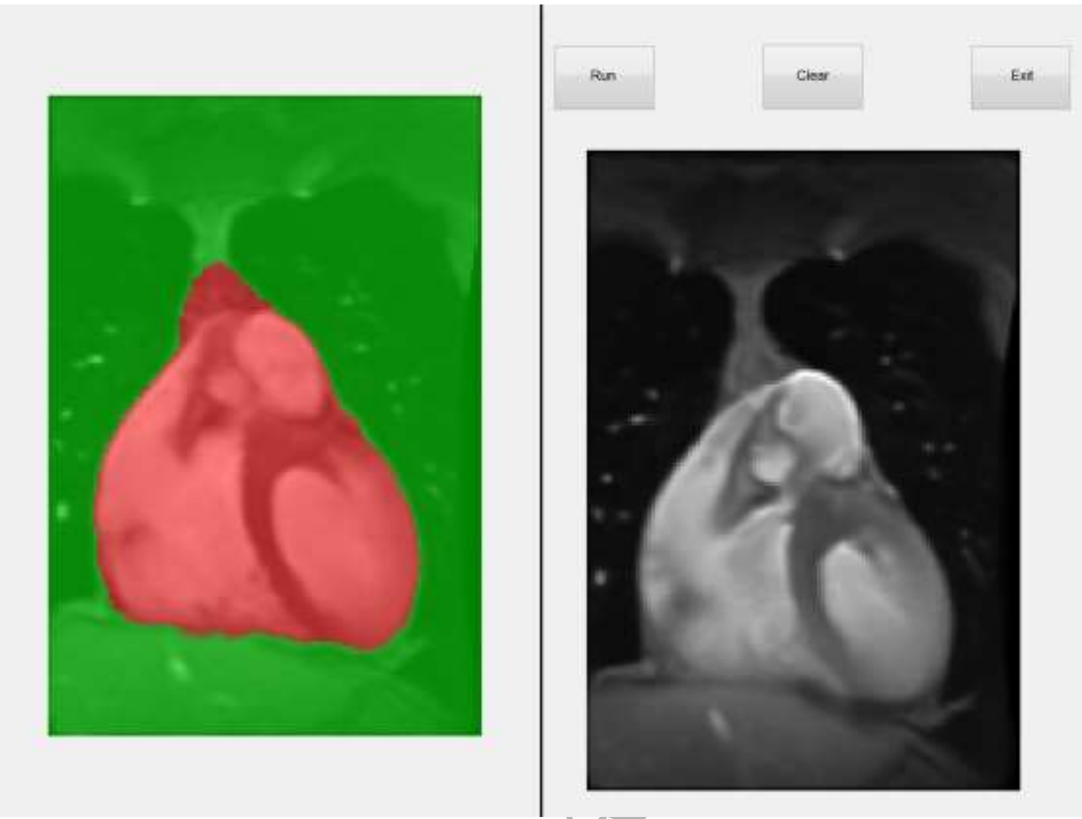


Figure A1: Experimental interface. The interface consisted of two windows, the left window showing an example segmentation to replicate, and the right window showing a similar image. Each version of the interface was visually identical, differing only in its behaviour. The three buttons, "Run", "Clear", and "Exit", run the underyling algorithm, reset the interface to its initial state, and exit the interface, respectively.

interfaces were shown was random with counterbalancing (i.e. a random permutation of (a),(b),(c) followed by the same permutation in reverse). To minimize the effect of training on the experiment, a minimum spacing of 24 hours between tasks was used. Before each trial, the participants were shown the interface and its contouring capabilities, then asked to fill in a questionnaire consisted of the question "What do you think the contour is telling the computer?" After each trial, the participants filled in a separate questionnaire consisting of the six NASA TLX (1986) scales and the question "How do you think the computer interprets the contours? What information does the computer use, extract, or detects from the contours?" Their answers are analyzed into four categories by two investigators; no structural consistency, structurally consistent with a shared path, fully structurally consistent, or not interpretable. (Not interpretable results were deemed so because of vagueness not permitting a clear indication of the user's mental model.) The dependant variables for measuring the interface's usability were the:

1. Mental demand from the NASA TLX scale,
2. Level of success from the NASA TLX scale,

3. Level of effort from the NASA TLX scale, and
4. Level of frustration from the NASA TLX scale.

The remaining NASA TLX scales (physical demand and temporal demand) where not selected as participants were allowed as much time as they desired to complete each task (mitigating temporal demand) and there was no physical difference in set-up between interfaces.

Hypothesis and Statistical Analysis

Our hypothesis tested regards the effect of *a priori* and *a posteriori* structural consistency on the performance of the interface. Specifically, we hypothesize that users whose mental model of the segmentation interface aligns well with how the interface is actually implemented will perform better, having a more positive evaluation of their interaction with the segmentation interface. Because of the learning effect, we expect users to improve in structural consistency after performing the trial. Thus, we assume that *a priori* structural consistency will have a more noticeable effect as less cognitive load is placed on the user to simultaneously learn how the system operates as well as perform the segmentation task. The statistical analysis performed was a set of four multi-factorial ANOVA tests containing the participant, trial number, *a priori* structural consistency, and *a posteriori* structural consistency as factors. Bonferroni correction was used to determine the significance of each factor.

Results and Discussion

The 22 novice participants altogether performed 113 segmentation tasks. No disagreement between the investigators was observed in terms of categorizing the written results of the questionnaire. As a test to ensure that the interface was learnable, we first checked whether using the interface caused participants to change their original explanations to those that were more structurally consistent. A matrix showing the association between the *a priori* structural consistency and the *a posteriori* structural consistency is shown in Table A1. Our results indicate that, in general, learning took place within the trial with more structurally consistent responses occurring after performing the experiment than before. Interestingly, individuals who had structurally inconsistent initial mental models were more likely to adapt fully structurally consistent ones than people who had sufficiently, but not fully, structural consistent mental models. This is possibly due to their mental models being insufficient to explain the computer's behaviour, thus necessitating revision, whereas those with shared path consistency did not have this motivation to adjust their mental model.

The results of the ANOVA table are shown in Table A2(a-d) and demonstrate that *a priori* structural consistency has a significant effect on performance in three of our four NASA TLX scales, and that *a posteriori* structural consistency does not. Due to the subjective nature of the NASA TLX, it is unsurprising that participants scaled their experiences of difficulty differently, thus leading to the significance of the participant factor in the statistical analysis. Table A3 lists the effect sizes for both *a priori* and *a posteriori* structural consistency. As expected, the effect sizes show that having an structurally inconsistent initial mental model leads to a higher perception of difficulty compared to those that were more structurally consistent. The effects are less coherent for *a posteriori* structural consistency, which may be due to there being less variability in the *a posteriori* structural consistency scores.

	***A Posteriori* Structural Consistency**					
	(n=113)	**-**	+	++	**N/A**	**Total**
***A Priori* Structural Consistency**	**-**	4	4	22	2	**32**
	+	1	20	19	2	**42**
	++	1	5	13	1	**20**
	N/A	1	5	8	5	**19**
	Total	**7**	**34**	**62**	**10**	

Table A1: Number of trials categorized by structurally consistency both *a priori* and *a posteriori*. Categories are: no structural consistency (-), structurally consistent with a shared path (+), fully structurally consistent (++), or not interpretable (N/A).

Source	DoF	F	p
Participant	21	2.99	0.0002
Trial Number	5	0.73	0.5995
***A priori* SC**	**3**	**4.30**	***0.0073**
A posteriori SC	3	0.02	0.9948
Error	80		

(a) NASA TLX Mental Demand Rating

Source	DoF	F	p
Participant	21	2.30	0.0042
Trial Number	5	0.89	0.4909
A priori SC	3	1.95	0.1283
A posteriori SC	3	0.42	0.7405
Error	80		

(b) NASA TLX Success Demand Rating

Source	DoF	F	p
Participant	21	2.32	0.0039
Trial Number	5	0.95	0.4564
***A priori* SC**	**3**	**5.43**	***0.0019**
A posteriori SC	3	0.68	0.5680
Error	80		

(c) NASA TLX Effort Demand Rating

Source	DoF	F	p
Participant	21	2.53	0.0016
Trial Number	5	1.73	0.1371
***A priori* SC**	**3**	**3.95**	***0.0111**
A posteriori SC	3	0.14	0.9343
Error	80		

(d) NASA TLX Frustration Demand Rating

Table A2: ANOVA tables for the four NASA TLX methods being combined. All p-values show are uncorrected. Significant factors after Bonferroni correction are shown in bold and marked with an asterisk.

		Mental Demand	Effort	Frustration
A Priori Structural Consistency	-	1.3253	0.8840	1.4622
	+	-0.4614	-0.2023	-1.0483
	++	-1.9323	-2.3698	-1.9112
	N/A	1.0685	1.6881	1.4974

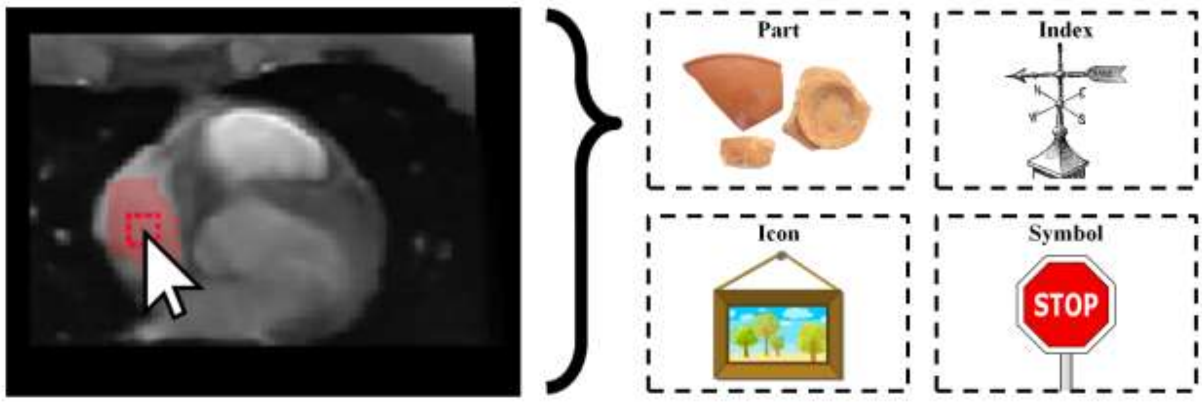
Table A3: Effect sizes for structural consistency on the three NASA TLX methods in which structural consistency has a significant effect. (Positive effects indicate higher perceived difficulty while negative effects indicate lower perceived difficulty.)

Conclusions

In this preliminary study, we have shown the efficacy of using sign graphs and metaphor analysis to evaluate and distinguish between three visually identical medical image segmentation interfaces. Specifically, our semiotic framework hypothesizes that improved structural consistency between the user's mental model and the computer's operation should have a positive effect on usability. Our results confirm this hypothesis for three of the four selected NASA TLX scales. The identical layout of the interfaces removes more traditional human factors analysis allowing us to focus on the knowledge and communication components emphasized by semiotic analysis. Our analysis also demonstrated a secondary result, that users with more structurally inconsistent mental states are more motivated to learn than those with sufficiently but not fully structurally consistent ones. This secondary finding, although not explicitly predicted, validates the explanatory value of our semiotic framework in terms of explaining how users learn from interaction with a segmentation interface.

These results should however be seen as preliminary. Our semiotic framework indicates that structural consistency is one of three metrics for evaluating the quality of sign graphs and thus of interaction mechanisms. Further research should be performed with the intent of isolating and evaluating semantic similarity and pragmatic centrality in order to quantify their effects.

Graphical abstract



Interaction in computation can be understood as a process of sign exchange in which each action can be described within the framework of Peircean semiotics providing a paradigm for analyzing interaction in medical image segmentation.