Available online at www.sciencedirect.com

SCIENCE DIRECT®

J. Vis. Commun. Image R. 16 (2005) 333–358

JOURNAL OF VISUAL Communication & IMAGE Representation

www.elsevier.com/locate/jvci

A logic framework for active contours on multi-channel images☆

Berta Sandberg, Tony F. Chan*

*UCLA, Mathematics Departement, 405 Hilgard Avenue, Los Angeles,
CA 90095-1555, United States*

Received 19 February 2004; accepted 18 August 2004
Available online 11 November 2004

Abstract

We propose a mathematical framework for object detection using logic operations as a structure for defining multi-channel segmentation. The model combines object information from the different channels into *any* logic combination. We consider active contour methods which use one initial contour that would evolve from the information given in each channel simultaneously. Specific models are derived based on the single-channel region based "active contours without edges" [IEEE Trans. Image Process. 10 (2) (2001) 266] model. Numerical experiments show that the method is able to find general intersections, unions, and complements of the regions of objects of both synthetic and realistic images.
© 2004 Elsevier Inc. All rights reserved.

Keywords: Multi-channel; Segmentation; Logic operations; Active contours

1. Introduction

Much has been written on active contour segmentation of multi-channel images. There are papers that discuss methods for color images (Sapiro, 1997; Zhu and

☆ This work was supported in part by ONR Contract N00014-96-1-0277, NSF Contract DMS-9973341 and NIH P20 MH65166.

* Corresponding author. Fax: +1 310 206 6673.
E-mail addresses: bsand@math.ucla.edu (B. Sandberg), chan@math.ucla.edu (T.F. Chan).

1047-3203/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2004.08.005

Yuille, 1996; Dibos and Koepfler, 1997), texture images convolved with filters (Chan et al., 2002; Sapiro and Ringach, 1996; Paragios and Deriche, 1999), multispectral images with occlusion in some channels and noise in others (Chan et al., 1999), and image sequences (Guichard, 1998; Yezzi and Soatto, 2003). Many of these models attempt to extract parts of an object from each of the channels and to recombine this information in a logical fashion. In most of these cases, the segmentation is some combination of occluded objects, or a combination of noisy images.

An example of occluded channels is given in Fig. 1. Most models for multi-channel segmentation would find a triangle that is the union of both channels as the desired segmentation.

While taking the union is reasonable, our view is that this is too limiting. We want to define a general framework, which allows the user to choose any logical combination of object information from each channel depending on the specific application. These segmentations can be described using combinations of intersection, union, and negation of the objects in the images. The user decides which logic operator is appropriate for the segmentation. In Fig. 2, examples of logic segmentation are given using union, intersection, and negation.

In this paper, we would like to stress two areas of interest that we cover. We define a logic framework for multi-channel image segmentation, as a general idea, regardless of the segmentation model used. This opens up solutions that have not been given previously, and explains the segmentation solutions that have been solved by other models. We then apply the logic framework to a region based model, Active Contours without edges (Chan and Vese, 2001). For a region based model, we found that models which treat inside the region and outside the region in the same manner, will have these two sides competing. For example, if we want the union of the objects

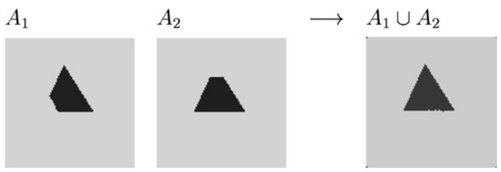


Fig. 1. A synthetic example of an object (a triangle) in two different channels. In A_1, the lower left corner is missing. In A_2 the upper corner is missing. Most multi-channel models converge to the union of the objects that are in the channels.

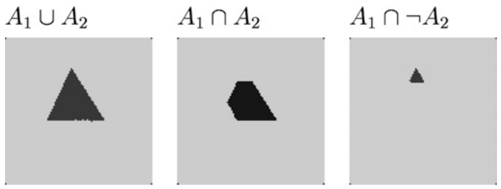


Fig. 2. Different logical combinations for the sample image, $A_1 \cup A_2$ is the union of the objects in each channel, $A_1 \cap A_2$ is the intersection, $A_1 \cap \neg A_2$ the object in A_1 that is not in A_2.

in two channels, it is equivalent to say that we want the intersection of the outside of the objects. The union and intersection will not be achieved using the same formulation, and these differences need to be taken into consideration.

We will use synthetic and real images to illustrate our examples. The synthetic images are used to illustrate how the models work. No details of the numerical implementation will be given. These are standard and can be found in (Aubert and Kornprobst, 2001; Chan and Vese, 2001; Osher and Fedkiw, 2002).

The outline of the paper is as follows: in Section 2, we describe a logic framework for multi-channel image segmentation, in Section 3 we review the background for the active contours without edges model. In Section 4, we will develop the logical extension of the active contours without edges model. In Section 5 we give examples using synthetic and real images of logic operators that are developed in Section 4. In Section 6, we compare the logic model of active contours without edges to the vector valued model in Chan et al. (1999). Section 7 is the conclusion which summarizes the work and discusses possible further work on the topic.

2. A logic framework for multi-channel image segmentation

There are a number of multi-channel papers written. Most of them extend a scalar segmentation model to multi-channel by giving a single formulation that combines the seperate channels together. A common example that we have seen is of summing a scalar model over the channels. Examples of such models include: region competition algorithm by Zhu and Yuille (1996), stereoscopic segmentation (Yezzi and Soatto, 2003), and vector valued active contours (Chan et al., 1999). By limiting themselves to a single model, these algorithms limit the possible solutions that may be obtained. Using a logic formulation, we extend the possibilities of multiple solutions. Putting them into a logic framework allows the user the power of choice, without searching for a new model or heuristic. The logic framework presents the different possibilities as part of the model. We compare the logic model with the vector valued model described in Chan et al. (1999) in Section 6. Other models which also sum a scalar model over channels would yield similar explanations and results.

An example where several different logic operations would be of interest is a hypothetical case of tumor detection in medical images. A sequence of two brain images of the same subject taken over time is given in Fig. 3. Several different scenarios of segmentation may be relevant. For example, to observe a new tumor growth we might want to detect objects in the second image that are not present in the first, i.e., $\neg A_1 \cap A_2$ where A_1 is the original image and A_2 is the second image in a time sequence. Another possibility is to observe the shrinking of a tumor, looking at objects in the first image that are not in the second, i.e., $A_1 \cap \neg A_2$. The user decides what is of interest and uses the model that corresponds to the appropriate logic operation. This particular example would not be solved by the models mentioned above (Chan et al., 1999; Yezzi and Soatto, 2003; Zhu and Yuille, 1996), because these models want to combine the objects together.

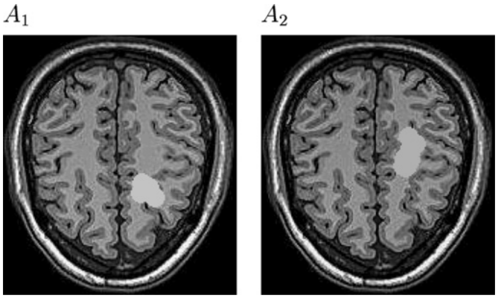


Fig. 3. An example of an MRI of a brain over time. The first image is a brain with a synthetic tumor in one place, while in the second image taken at a later time, the synthetic tumor is in a different place. Information of interest could be finding the tumor in the first image that is not in the second, $A_1 \cap \neg A_2$ or the tumor in the second image that is not in the first, $\neg A_1 \cap A_2$.

It has been suggested that parsing the image into objects of different textures and colors, as in Tu et al. (2003), would achieve a similar conclusion. However, parsing the image into different textures leaves out channel location of the object. Logic operations take intensity contrasts as well as channel location. The purpose of this paper is to develop a general framework that accommodates different possibilities of segmentation, not to explore one particular segmentation model.

One direct way of accomplishing logic segmentations is to segment each channel independently, by the model of your choice, followed by bitwise logic operations. However, there are some drawbacks to this approach. First, if there are many images, it is cumbersome and costly to perform all the segmentations separately. On a deeper level, such an approach often gives undesirable segmentations because the information in the channels is not taken together. Assuming that each image is independent of the other, each channel is segmented separately, causing valuable information to be lost or spurious information to be retained. In Fig. 4, such a situation is illustrated. In the two channels, the images (a triangle) are occluded and noisy. When each one is segmented separately, due to the noise, the segmentation has jagged boundaries. We then do bitwise logic operations $A_1 \cap A_2$, $A_1 \cup A_2$, and $A_1 \cap \neg A_2$. While the union and intersection of objects are successful, the one involving negation gives false points because of the jagged boundaries.

Another possibility, is to perform logic operations on the channels forming a combined image, followed by a segmentation of the image. This is attempted on the occluded triangle images that are in Fig. 1. The contrast values in the images are 0 or 1. If the intensity of the object is known ahead of time to be 1 inside the object and 0 outside the object, direct bitwise logic operations can be performed on the pixels to get the desired solution (see Fig. 5A) Using the same pixel logic rules for channels that have 0 inside the object and 1 outside the object the outcome reverses the segmentation solutions, the union performs an intersection and vice versa (see Fig. 5B). If the intensity value 0 is inside the object in one channel and 1 inside the object in the other channel, the outcome is inconsistent with either of the previous solutions (see Fig. 5C). In order to do pixel logic, it is necessary to know the

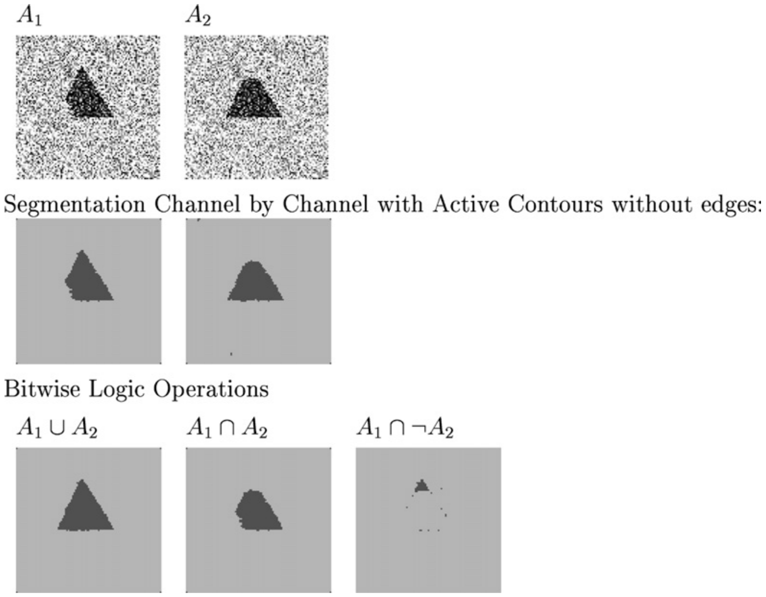


Fig. 4. In this example the segmentation is done independently on each channel. Using the active contours without edges model, the two images are combined using bitwise OR and AND. In $A_1 \cap \neg A_2$ spurious points arise as a result of noise in the channels.

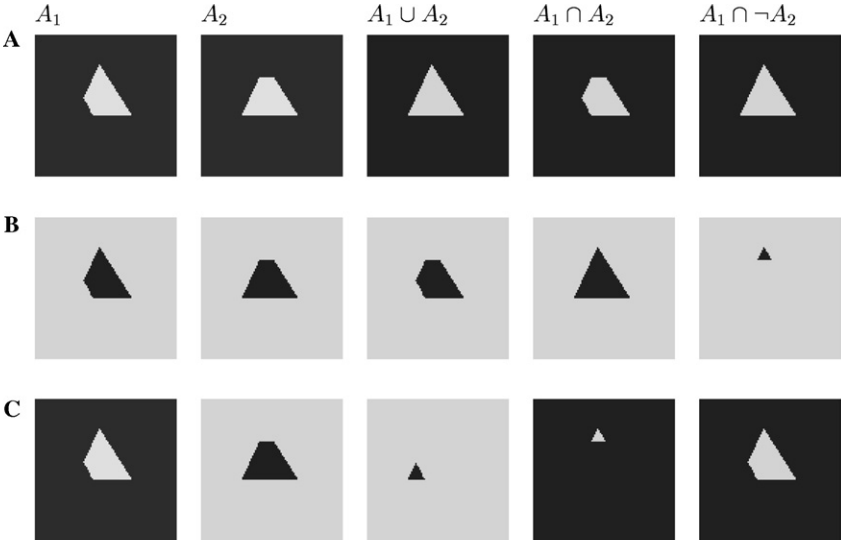


Fig. 5. Pixel logic operations will yield different results depending on the intensity values inside and outside the object (A–C). To get consistent results using pixel logic the intensity values must be known for inside the object in each channel.

intensity of the object in each channel, otherwise the results will be unpredictable. To perform pixel logic consistently, the contrast value inside and outside the object must be known in each channel. This is a lot of information to have ahead of time. Thus, other methods need to be investigated for logic operations on multi-channel images.

In this paper, we investigate object detection using a single contour for all the channels in the image set using active contour models. Many active contour models for scalar images have the following variational form

$$\inf F(u_0, C) = \lambda \int_C J + R,$$

where u_0 is the image, C is the evolving curve, $\int_C J$ is the term related to the image, and R is the regularization term. We will use the active contours without edges model developed by Chan and Vese (2001). The derivation of the logic model is based on extending the functional J to a form suitable for multi-channel images. Implementation of the logic operations is achieved by various combinations of the channels. The Chan–Vese model will have one initial contour to find one "object" in the image. The model has a natural extention for several contours and thus several "objects" in the image (Vese and Chan, 2002), which can also be extended for use in multi-channel logic operations.

3. Background for the scalar active contours without edges model

Let Ω be a bounded open subset of $\Re^2$, with $\partial\Omega$ the boundary. Let u_0 be a given image such that $u_0 : \overline{\Omega} \to \Re$. Let $C(s) : [0,1] \to \Re^2$ be a piecewise parameterized C^1 curve.

We first recall the Chan–Vese model (Chan and Vese, 2001), which has the following form

$$\inf_{c^+, c^-, C} F(c^+, c^-, C),$$

where

$$F(c^+, c^-, C) = \mu|C| + \lambda^+ \int_{\mathrm{in}(C)} |u_0 - c^+|^2 \,\mathrm{d}x + \lambda^- \int_{\mathrm{out}(C)} |u_0 - c^-|^2 \,\mathrm{d}x, \qquad (1)$$

where $|C|$ denotes the length of C, c^+, and c^- are constant unknowns representing the "average" value of u_0 inside and outside the curve, respectively. The parameters $\mu > 0$, and λ^+,$\lambda^- > 0$, are weights for the regularizing term and the fitting term, respectively.

Minimizing the fitting error in (1), the model approximates the image u_0 with a piecewise constant function, taking only two values, namely c^+ and c^-, and with one edge C, the boundary between these two constant regions. The object to be detected will be given by one of the regions, and the curve C will be the boundary of the object. The additional length term is a regularizing term, and has a scaling role. If μ is large, only larger objects are detected, while for small μ, objects of smaller size are also detected. Because the model does not make use of a stopping edge-function based on the gradiant, such as the snakes model developed by Kass et al. (1988) it can detect edges both with and without gradient. It is well known that (1) can be viewed as a special case of the Mumford-Shah segmentation (Mumford and Shah, 1989).

For curve evolution, the level set method has been used extensively, in particular where the motion is governed by mean curvature, as in Osher and Sethian (1988). This formulation behaves well even with cusps, corners, and automatic topological changes.

We rewrite the original model (1) in the level set formulation. Let the evolving curve C be embedded as the zero level set of a Lipschitz continuous function ϕ, i.e., $C(\phi)=\{(x,y)\in\Omega:\phi(x,y)=0\}$, with ϕ having opposite signs on each side of C. Following Zhao et al. (1996) and Chan and Vese (2001), the energy can be written as

$$F(c^+,c^-,\phi)=\mu|C(\phi)|+\lambda^+\int_{\phi\geqslant 0}|u_0(x,y)-c^+|^2\mathrm{d}x\mathrm{d}y+\lambda^-\int_{\phi<0}|u_0(x,y)-c^-|^2\mathrm{d}x\mathrm{d}y.$$

Using the Heaviside function H, defined by

$$H(z)=\begin{cases}1 & \text{if } z\geqslant 0,\\ 0 & \text{if } z<0,\end{cases}$$

and the Dirac Delta function $\delta(z)=(\mathrm{d}/\mathrm{d}z)H(z)$ (in the sense of distributions), we can rewrite the energy functional as follows:

$$F(c^+,c^-,\phi)=\mu\int_\Omega\delta(\phi(x,y))|\nabla\phi(x,y)|+\lambda^+\int_\Omega|u_0(x,y)-c^+|^2H(\phi(x,y))\,\mathrm{d}x\,\mathrm{d}y$$
$$+\lambda^-\int_\Omega|u_0(x,y)-c^-|^2(1-H(\phi(x,y)))\,\mathrm{d}x\,\mathrm{d}y.$$

Minimizing $F(c^+,c^-,\phi)$ with respect to the constants c^+ and c^-, for a fixed ϕ, yield the following expressions for c^+ and c^-, function of ϕ

$$\begin{cases}c^+=\text{average}(u_0) & \text{on } \phi\geqslant 0,\\ c^-=\text{average}(u_0) & \text{on } \phi<0.\end{cases}$$

Minimizing the energy $F(c^+,c^-,\phi)$ with respect to ϕ, for fixed c^+ and c^-, using a gradient descent method, yields the associated Euler-Lagrange equation for ϕ, governed by the mean curvature and the error terms (see Chan and Vese, 2001 for more details).

$$\frac{\partial\phi}{\partial t}=\delta_\epsilon\left[\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)-\lambda^+(u_0-c^+)^2+\lambda^-(u_0-c^-)^2\right] \tag{2}$$

in Ω, and with the boundary condition

$$\frac{\delta_\epsilon(\phi)}{|\nabla\phi|}\frac{\partial\phi}{\partial\vec{n}}=0$$

on $\partial\Omega$, where $\vec{n}$ denotes the unit normal at the boundary of Ω.

Using a level set formulation with this model allows the initial contour to find any number of objects from an initial contour anywhere in the image. For general information, one may consult (Osher and Fedkiw, 2002) and (Sapiro, 2001). A multiphase extention has been documented in Vese and Chan (2002), and a texture segmentation in Chan et al. (2002).

4. Logic operations on region based active contours

The Chan–Vese active contours without edges method is a region based method. This is a significant benefit, as it is especially important when finding logical combinations of objects. That is why we chose the model presented in Section 2.

Fig. 5 is an example to show why active contours without edges does a good job of multi-channel segmentation. Rather then comparing contrast of the object, it compares the fitting errors of each channel. The model does not care that each channel has different intensity values, instead it wants a contour that will minimize the fitting errors based on the average value for each channel.

We will look at an example for two channels. To find the union of an object, one can take the union of the objects designated as the black values in the first row of Fig. 6. But another way to look at it is to look at the intersection of the outside of the objects, designate in black in the second row in Fig. 6. Adding the information together will give us the union of the object. From the perspective of the object both as a region in C, as well as, the complement of the region outside C.

To set up the logical framework we define two separate logic variables, z_i^{in} and z_i^{out}, to denote whether a point (x,y) is in C or not:

$$z_i^{\text{in}}(u_0^i, x, y, C) = \begin{cases} 0 & \text{if}\,(x,y) \in C \text{ and } (x,y) \text{ inside the object in channel } i, \\ 1 & \text{otherwise,} \end{cases}$$

$$z_i^{\text{out}}(u_0^i, x, y, C) = \begin{cases} 1 & \text{if}\,(x,y) \notin C \text{ and } (x,y) \text{ is inside the object in channel } i, \\ 0 & \text{otherwise.} \end{cases}$$

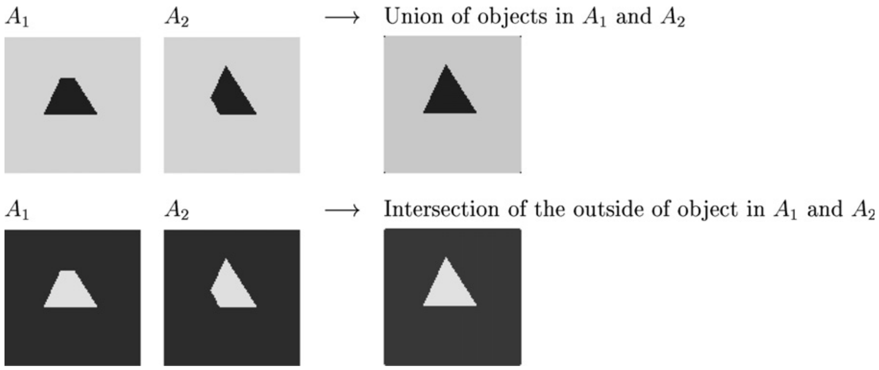

A_1
A_2
$\longrightarrow$ Union of objects in A_1 and A_2
A_1
A_2
$\longrightarrow$ Intersection of the outside of object in A_1 and A_2

Fig. 6. Logic operations inside and outside the object. The upper triple of images show that the union of the inside (black) region gives the union of the two objects in A_1 and A_2. The bottom triple shows that the intersection of the outside (black) region gives the complement to the union of two objects.

A natural way to define z_i^{in} and z_i^{out} for the Chan–Vese model is as follows:

$$\begin{aligned} z_i^{\text{in}}(u_0^i, x, y, C) &= \frac{|u_0^i(x,y) - c_+^i|^2}{\max_{(x,y)\in u_0^i} u_0^i}, \\ z_i^{\text{out}}(u_0^i, x, y, C) &= \frac{|u_0^i(x,y) - c_-^i|^2}{\max_{(x,y)\in u_0^i} u_0^i}. \end{aligned} \tag{3}$$

Note that we use "0" as the "true" value, and "1" as the "false" value, which is the reverse of the usual convention. This is more convenient because our framework is based on a minimizing of an objective function and thus we want the "0" value correspond to "true."

For the complement of the object in channel i we define

$$\begin{aligned} z_i^{\text{in}'} &= 1 - z_i^{\text{in}}, \\ z_i^{\text{out}'} &= 1 - z_i^{\text{out}}. \end{aligned} \tag{4}$$

Following the structure of logic operators, we now want to define a truth table for the logic model the the variables described above. Table 1 shows the binary inputs and outputs of the truth table for three logic operations $A_1 \cup A_2$, $A_1 \cap A_2$, and $A_1 \cap \neg A_2$. We treat the points inside C separately from those outside C.

Continuing with the two channel example $A_1 \cup A_2$, we define it in truth table form. The truth table needs to reflect the union of z_i^{in} and the intersection of z_i^{out}. For the point $(x,y) \in C$ the variable z_i^{in} is defined. If the point $(x,y) \in C$ is in the object in either channel the logic model returns 0, otherwise it returns 1—this reflects the union of the inside of the object. If $(x,y) \in \Omega \backslash C$, the variable z_i^{out} is defined. The logic model returns 0 if (x,y) is *not* in the object in either channel, otherwise it will return 1—this represents the intersection of the outside of the object. The column marked $A_1 \cup A_2$ relates this information. The logic operations $A_1 \cap A_2$, and $A_1 \cap \neg A_2$ are calculated in a similar fashion. For intersection of objects, we take the intersection of the inside of objects and the union of the outside of objects. For negation we substitute z_i' for z_i as shown in (4).

Table 1
The truth table for the active contours without edges model

	Truth table for two channels						
	z_1^{in}	z_2^{in}	z_1^{out}	z_2^{out}	$A_1 \cup A_2$	$A_1 \cap A_2$	$A_1 \cap \neg A_2$
$(x,y) \in C$	1	1	1	1	1	1	1
	1	0	1	1	0	1	1
	0	1	1	1	0	1	0
	0	0	1	1	0	0	1
$(x,y) \in \Omega \backslash C$	1	1	1	1	1	1	0
	1	1	1	0	1	0	1
	1	1	0	1	1	0	0
	1	1	0	0	0	0	0

Table 1 summarizes the different logic models with input parameters z_i^{in} and z_i^{out}, but this table only has binary values. In practical implementation, we have to allow these variables to take on real values as this can happen in (3).

Therefore it is necessary to find interpolation functions for the union and intersection of variables z_i for $i = 1, \ldots, n$. In Table 2 the functional dependence on z_1 and z_2 is stated for the union and intersection. We will find an interpolation function that will fit these points and maintain monotonic values between 0 and 1.

There are many functions that could interpolate Table 2. For the union function of logic variables we choose

$$f_{\cup} = (z_1 \cdot z_2)^{1/2}.$$

For intersection interpolation function we choose

$$f_{\cap} = 1 - ((1 - z_1)(1 - z_2))^{1/2}.$$

The square roots of the products are taken to keep them of the same order as the original scalar model.

Combining the interpolation functions for union of inside the objects, and intersection outside the objects we get the union of objects

$$f_{A_1 \cup A_2}(x,y) = \sqrt{z_1^{\text{in}}(x,y) z_2^{\text{in}}(x,y))} + 1 - \sqrt{(1 - z_1^{\text{out}}(x,y))(1 - z_2^{\text{out}}(x,y))}.$$

Likewise, to get the intersection of objects, we combine the intersection of the inside with the union of the outside, resulting in the following objective function for the intersection of objects

$$f_{A_1 \cap A_2}(x,y) = 1 - \sqrt{(1 - z_1^{\text{in}}(x,y))(1 - z_2^{\text{in}}(x,y))} + \sqrt{z_1^{\text{out}}(x,y) z_2^{\text{out}}(x,y))}.$$

An example of a logical operation involving three channels that combines union, intersection and negation of objects is $(A_1 \cap \neg A_2) \cup A_3$. To derive the corresponding objective function, one can treat the term in parenthesis first, and then incorporate the objective function for that with the union operation with A3. We first calculate the logic operation $A_1 \cap \neg A_2$

$$f_{(A_1 \cap \neg A_2)} = 1 - ((1 - z_1^{\text{in}}) z_2^{\text{in}})^{1/2} + (z_1^{\text{out}}(1 - z_2^{\text{out}}))^{1/2}.$$

Then we take the union of $f_{(A_1 \cap \neg A_2)}$ with A_3 and get the following:

Table 2
The table for the interpolation functions of z_i^{in}s and z_i^{out}s

Table of points for interpolation function			
z_1	z_2	$z_1 \cup z_2$	$z_1 \cap z_2$
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	0

$$f_{(A_1\cap\neg A_2)\cup A_3}=((1-((1-z_1^{\text{in}})z_2^{\text{in}})^{1/2})z_3^{\text{in}})^{1/2}+1-(1-((z_1^{\text{out}}(1-z_2^{\text{out}}))^{1/2})(1-z_3^{\text{out}}))^{1/2}. \tag{5}$$

In the above, we have used the interpolation functions to directly derive the objective functions corresponding to a given logical expression. Even though we have bypassed the corresponding truth table, it can be easily verified that the resulting objection functions do interpolate the function values given in the truth table.

It is straightforward to extend the two channel case to n channels. First consider the union case. Let the logic operation be expressed as:

$$L_1(A_1)\cup L_2(A_2)\cup\cdots\cup L_n(A_n),$$

where $L_i(A_i)$ is either A_i or $\neg A_i$. The objective function for this is as follows:

$$f_{L_1(A_1)\cup\cdots\cup L_n(A_n)}=\left(\prod_{i=1}^{n}l_i\left(z_i^{\text{in}}\right)\right)^{1/n}+1-\left(\prod_{i=1}^{n}\left(1-l_i\left(z_i^{\text{out}}\right)\right)\right)^{1/n},$$

where

$$l_i(z_i^{\text{in}})=\begin{cases}z_i^{\text{in}} & \text{if } L_i(A_i)=A_i,\\ z_i^{\text{in}'} & \text{if } L_i(A_i)=\neg A_i.\end{cases}$$

Similarly, for the intersection case:

$$L_1(A_1)\cap L_2(A_2)\cap\cdots\cap L_n(A_n),$$

the objective function is:

$$f_{L_1(A_1)\cap\cdots\cap L_n(A_n)}=1-\left(\prod_{i=1}^{n}\left(1-l_i\left(z_i^{\text{in}}\right)\right)\right)^{1/n}+\left(\prod_{i=1}^{n}l_i\left(z_i^{\text{out}}\right)\right)^{1/n}.$$

The functionals to be minimized in the model can now be written as:

$$\begin{aligned}F_{L_1(A_1)\cup\cdots\cup L(A_n)}=\mu|C|+\lambda\Bigg[&\int_{\text{inside}(C)}\left(\prod_{i=1}^{n}l_i\left(z_i^{\text{in}}\right)\right)^{1/n}\mathrm{d}x\\&+\int_{\text{outside}(C)}\left(1-\left(\prod_{i=1}^{n}\left(1-l_i\left(z_i^{\text{out}}\right)\right)\right)^{1/n}\right)\mathrm{d}x\Bigg],\end{aligned}$$

and

$$\begin{aligned}F_{L_1(A_1)\cap\cdots\cap L_n(A_n)}=\mu|C|+\lambda\Bigg[&\int_{\text{inside}(C)}\left(1-\left(\prod_{i=1}^{n}\left(1-l_i\left(z_i^{\text{in}}\right)\right)\right)^{1/n}\right)\mathrm{d}x\\&+\int_{\text{outside}(C)}\left(\prod_{i=1}^{n}l_i\left(z_i^{\text{out}}\right)\right)^{1/n}\mathrm{d}x\Bigg].\end{aligned}$$

The functional may be written using the level set formulation as described in Section 2.

Now we can rewrite the functional F for a general $f(z_1^{\text{in}}, z_1^{\text{out}}, \ldots)$ using the level set function ϕ.

The objective function for the variational model is

$$F(\phi, c^+, c^-) = \mu|C(\phi)| + \lambda\left[\int_\Omega f_{\text{in}}(z_1^{\text{in}}, \ldots, z_n^{\text{in}})H(\phi) \right. \\ \left. + f_{\text{out}}\left(z_1^{\text{out}}, \ldots, z_n^{\text{out}}\right)(1 - H(\phi))\,\mathrm{d}x\right]. \tag{6}$$

Derivation of the Euler–Lagrange equation are similar to that of the scalar model and yield the following differential equation (which at steady state gives the solution):

$$\frac{\partial\phi}{\partial t} = \delta(\phi)\left[\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \lambda(f_{\text{in}}(z_1^{\text{in}}, \ldots, z_n^{\text{in}}) - f_{\text{out}}(z_1^{\text{out}}, \ldots, z_n^{\text{out}}))\right]$$

with the boundary condition

$$\frac{\delta(\phi)}{|\nabla\phi|}\frac{\partial\phi}{\partial\vec{n}} = 0$$

on $\partial\Omega$, where $\vec{n}$ denotes the unit normal at the boundary of Ω. For example, for the two logic models presented earlier, the corresponding Euler–Lagrange equations are:

$$\frac{\partial\phi_{L_1(A_1)\cup\cdots\cup L_n(A_n)}}{\partial t} = \delta_\epsilon(\phi)\left[\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \lambda\left(\left(\prod_{i=1}^{n} l_i\left(z_i^{\text{in}}\right)\right)^{1/n} \right.\right. \\ \left.\left. + 1 - \left(\prod_{i=1}^{n}\left(1 - l_i\left(z_i^{\text{out}}\right)\right)\right)^{1/n}\right)\right],$$

$$\frac{\partial\phi_{L_1(A_1)\cap\cdots\cap L_n(A_n)}}{\partial t} = \delta_\epsilon(\phi)\left[\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - \lambda\left(1 - \left(\prod_{i=1}^{n}\left(1 - l_i\left(z_i^{\text{in}}\right)\right)\right)^{1/n} \right.\right. \\ \left.\left. + \left(\prod_{i=1}^{n} l_i\left(z_i^{\text{out}}\right)\right)^{1/n}\right)\right].$$

Even though the form is complicated, the implementation is very similar to that of the scalar model that is in (2). The details for this scheme can be found in Chan and Vese (2001).

5. Experimental results

In this section, we show some examples of the performance of the logical active contours models described in Section 3. First, simple images are shown to illustrate the basic principles of the framework. Then more complicated and realistic images are introduced to demonstrate the robustness of the model.

Fig. 7 shows two different occlusions of a triangle. Using our new model, we are able to recover the union, intersection, and negation of the objects in the channels using the functionals described above. In this case, where there is no noise, the constant λ is set to be the order of the $(\max u_0^i(x,y))^2$. When there is noise in the image, λ is set to be smaller so that the noise is not included in the segmentation.

Fig. 8 shows a time evolution on two channels which had the contrast reversed. Active contours without edges model has a contrast invariant nature. Unlike using logic on two images followed by segmentation (shown in Fig. 5), this model is able to find the object regardless of contrast differences.

In Fig. 9, a 3 channel example is used to illustrate more possibilities of different logic operations, including combinations of union and intersection. We ran the model for four cases. The first two are $A_1 \cap A_2 \cap A_3$ and $A_1 \cup A_2 \cup A_3$:

$$f_{A_1 \cap A_2 \cap A_3} = 1 - \left(\prod_{i=1}^{3} (1 - z_i^{\text{in}}) \right)^{1/3} + \left(\prod_{i=1}^{3} z_i^{\text{out}} \right)^{1/3},$$

$$f_{A_1 \cup A_2 \cup A_3} = \left(\prod_{i=1}^{3} z_i^{\text{in}} \right)^{1/3} + 1 - \left(\prod_{i=1}^{3} (1 - z_i^{\text{out}}) \right)^{1/3}.$$

The other two cases have examples with negation and mixing of unions and intersections. The first case is $(A_1 \cap \neg A_2) \cup A_3$. This is a particularly complicated case as the union, intersection, and negation are all mixed in. We used the interpolation functions defined previously:

$$f_{(A_1 \cap \neg A_2) \cup A_3} = ((1 - ((1 - z_1^{\text{in}}) z_2^{\text{in}})^{1/2}) z_3^{\text{in}})^{1/2} + 1 - (1 - ((z_1^{\text{out}}(1 - z_2^{\text{out}}))^{1/2}) \times (1 - z_3^{\text{out}}))^{1/2}.$$

In this case the example has negation and intersection for three channels $(A_1 \cap \neg A_2) \cap A_3$, defined below

$$f_{(A_1 \cap \neg A_2) \cap A_3} = 1 - ((1 - z_1^{\text{in}}) z_2^{\text{in}} (1 - z_3^{\text{in}}))^{1/3} + (z_1^{\text{out}} (1 - z_2^{\text{out}}) z_3^{\text{out}})^{1/3}.$$

The three channel examples give an idea of the many possible logic combinations possible for more than two channels.

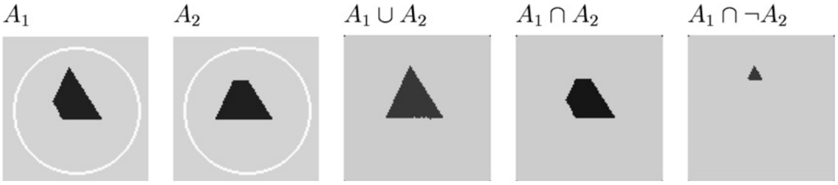


Fig. 7. The basic example for a two channel logic model. The first two images are the initial images with the initial contour. The three rightmost images show the resulting segmentation.

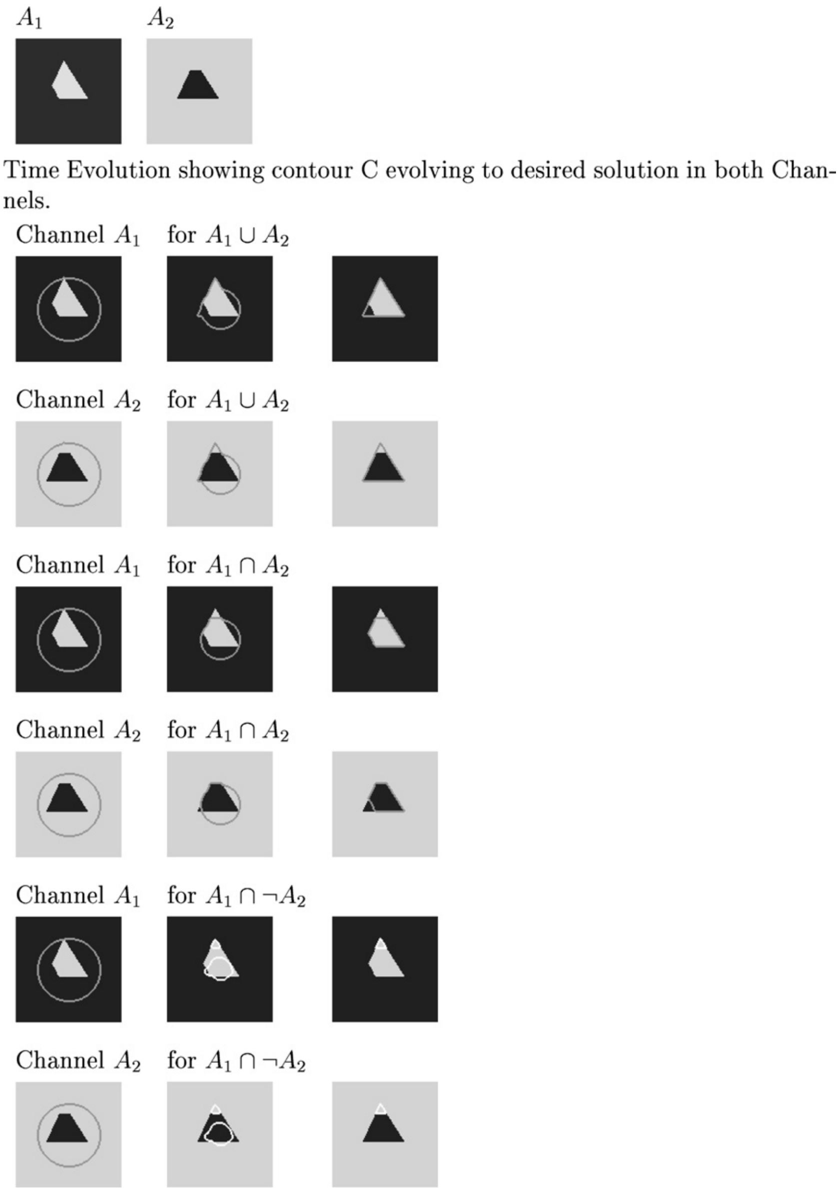

A_1
A_2
Time Evolution showing contour C evolving to desired solution in both Channels.
Channel A_1 for $A_1 \cup A_2$
Channel A_2 for $A_1 \cup A_2$
Channel A_1 for $A_1 \cap A_2$
Channel A_2 for $A_1 \cap A_2$
Channel A_1 for $A_1 \cap \neg A_2$
Channel A_2 for $A_1 \cap \neg A_2$

Fig. 8. A time evolution for a 2 channel image with channels having reverse contrast. The active contours without edges model is contrast invariant, thus the logic model is able to segment to desired object.

The logic model's robustness can be seen in the next two examples in Figs. 10 and 11. The initial contour does not have to surround the object in order to find the object desired, as can be seen in Fig. 10. Likewise the model finds an unregistered image segmentation in Fig. 11.

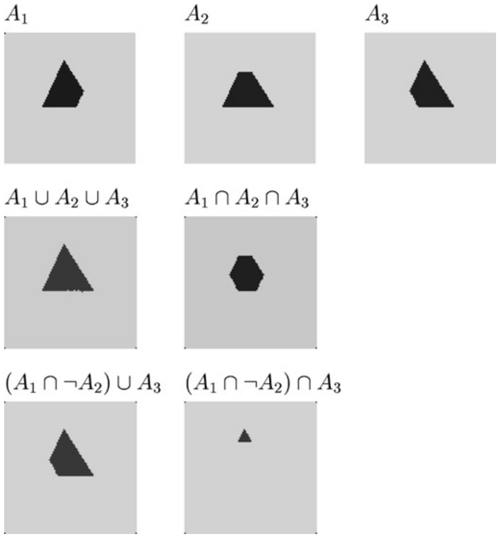


Fig. 9. An example for the 3 channel case of the logic model. The model found the desired segmentations using the functions defined in Section 3.

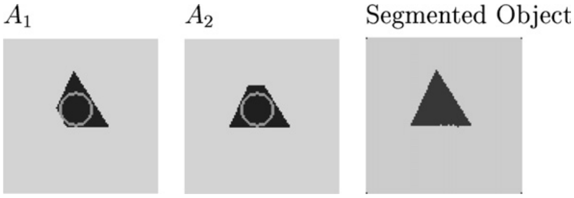


Fig. 10. Region based logic model with initial contour inside the object. Even though the initial contour is inside the object, the desired segmentation $A_1 \cup A_2$ is found.

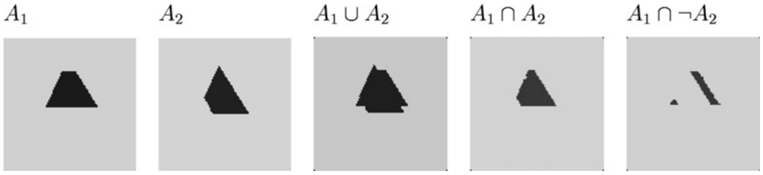


Fig. 11. Region based logic model for unregistered images. The logic model still find $A_1 \cup A_2$ and $A_1 \cap A_2$ successfully, since the model is looking at the regions of union and intersection. Notice that when $A_1 \cap \neg A_2$ is found, it contains a part of the object that is due to the images being unregistered rather than the intrinsic object difference.

All the examples so far have images which are piecewise constant. The parameters z_i^{in} and z_i^{out} will have two possible values each, a larger and smaller value. Thus it mimics the truth table values.

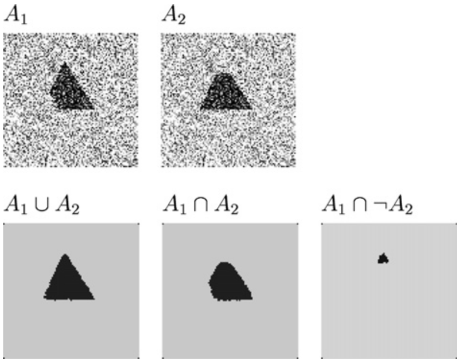


Fig. 12. Region based logic model for noisy images. This is the same example as that for the channel-by-channel case (Fig. 4). While the union of the two channels is of comparable quality, the intersection is better when done using the logic operations method.

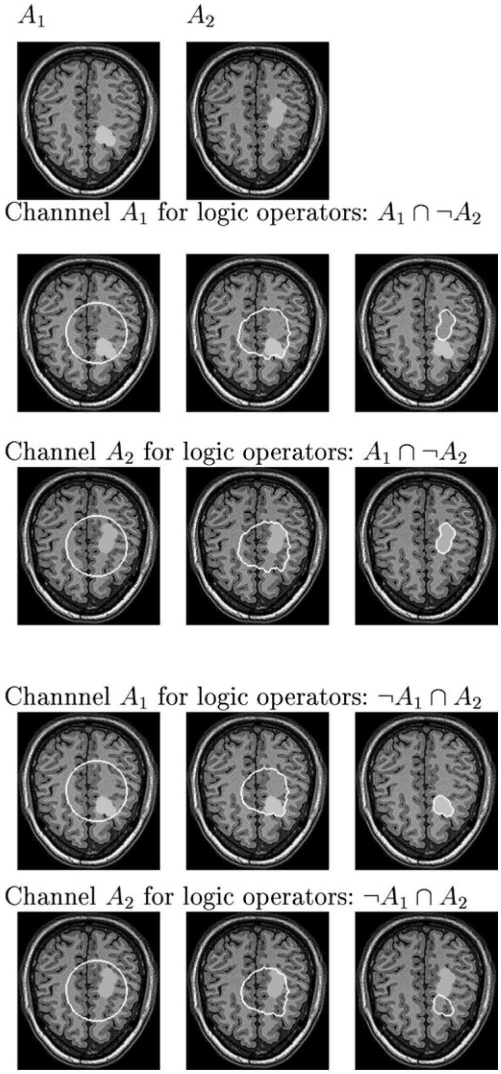
In the next example, the synthetic images are not piecewise constant. In Fig. 12, noise was added. The logic model found the logic operations $A_1 \cup A_2$, $A_1 \cap A_2$, and $A_1 \cap \neg A_2$ with the same consistency as the synthetic examples without noise. No spurious data is detected, which is an improvement over Fig. 4 which was done using channel by channel segmentation.

Now we come to the examples with real images. The only parameter that was used was μ to control the contour in the cases when the image became very noisy. These solutions had the same weight for inside the contour C, as well as outside the contour C. Using logic operators allows a more global solution. Thus for logic operators the solution will give the union, intersection or complement, as requested, in most situations regardless of initial contour, and without weighing of inside/outside contour C separately.

Fig. 13 shows two brain images seen previously in Fig. 3. They are two MRIs of the brain taken in a time sequence, each with a synthetic tumor placed in a different spot. Using logic operation $A_1 \cap \neg A_2$ the tumor in the first image may be extracted, i.e., the logic operations find the object in the first image that is different from the second. The reverse is also true. Using the logic model that describes $\neg A_1 \cap A_2$, the model finds the object in the second image that is not in the first. This happens to be a very complicated example as there are a lot of features and textures. Not only does the model finds the tumor, using logic operations gives the user the capability to define more precisely how the information from the different channels are to be combined in order to obtain a desired segmentation, as well as the freedom to use all possible logical combinations using a systematic framework.

In Fig. 14, we take two airplane images with a noisy environment and add synthetic occlusions in each channel. In Figs. 15–17, logic operations $A_1 \cup A_2$, $A_1 \cap A_2$, and $A_1 \cap \neg A_2$ are performed on the images. In the logic operations of $A_1 \cap \neg A_2$ the model finds the object that is in A_1 and not in A_2 which includes the occlusion in A_1.

For our final example, we present a segmentation of a textured image which has been convolved with a Gabor transform. In Fig. 18, an image of two zebras



A_1 A_2

Channnel A_1 for logic operators: $A_1 \cap \neg A_2$

Channel A_2 for logic operators: $A_1 \cap \neg A_2$

Channnel A_1 for logic operators: $\neg A_1 \cap A_2$

Channel A_2 for logic operators: $\neg A_1 \cap A_2$

Fig. 13. Time evolution of region based logic model on a MRI scan of the brain. The first channel A_1, has a synthetic brain tumor in one place, the second image the synthetic brain tumor is in a different place. The images are registered. By design we want to find the tumor that is in A_1 and not A_2, $A_1 \cap \neg A_2$. Likewise we want to find the tumor in A_2 that is not in A_1 and $\neg A_1 \cap A_2$.

with a grassy background. The image is convolved with Gabor transforms. The best transforms are chosen, the heuristics of this are discussed in Chan et al. (2002). Finally the union of the images would be take to produce the segmentation of the textures. This is done with a large μ in order to ignore noise that is produced, but the functions inside C and outside C are weighted equally, thus it was simple to implement.

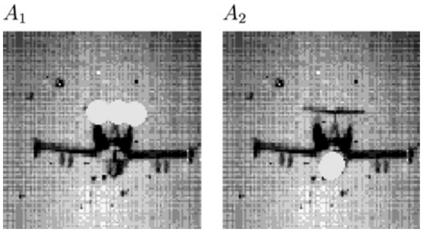


Fig. 14. The original airplane images, each channel with a different occlusion.

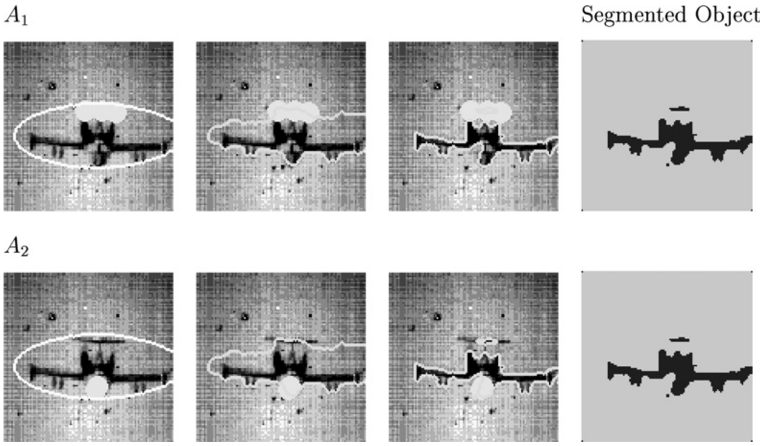


Fig. 15. Time evolution of active contours for airplane images using the logic model to obtain $A_1 \cup A_2$. The last image is the final segmentation of the object.

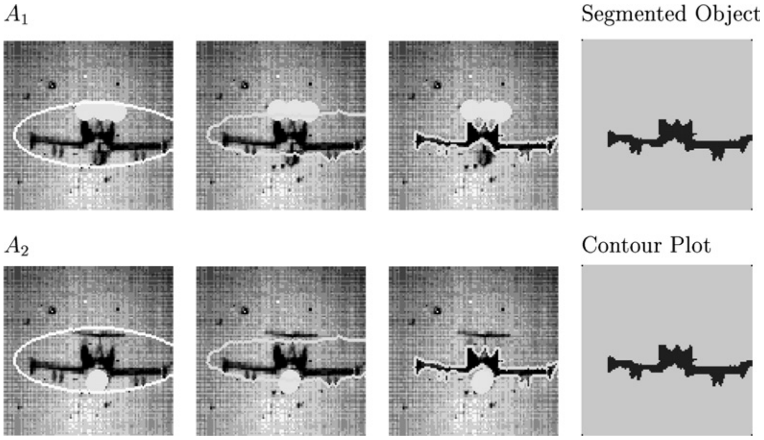


Fig. 16. Time evolution of active contours for airplane images using the logic model to obtain $A_1 \cap A_2$. The last image is the final segmentation of the object.

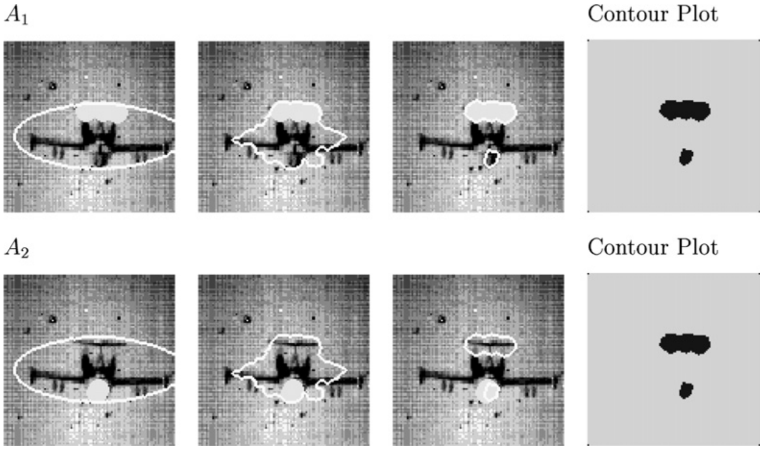


Fig. 17. Time evolution of the active contours for the airplane images that obtains $A_1 \cap \neg A_2$. The segmentation finds the occluded object in A_2, which includes the occlusion in A_1. The last image is the final segmentation of the object.

6. Comparison of logic active contour model with the vector valued active contour model

In the next section, we compare the logic operations to the vector valued active contours (Chan et al., 1999) which is pde based. Vector valued active contours is very similar to other multi-channel segmentation models which sum across the channels such as region competition algorithm by Zhu and Yuille (1996) and stereoscopic segmentation (Yezzi and Soatto, 2003). These algorithms offer a valid segmentation solution to a single solution. We applied logic operations to the active contours model, as we described above. Below we compare how these algorithms interact with the sample images we have shown in the section on experimental results.

The objective of this paper is related to that of the vector valued active contours model in Chan et al. (1999). Both papers try to combine information from different channels of an image in order to derive an active contour segmentation. The vector based model in Chan et al. (1999) is as follows:

$$\inf_{C,c^+,c^-} F(C,c^+,c^-) = \mu|C| + \lambda^+ \int_{\text{in}(C)} \sum_{i=1}^{n} |u_0^i - c_i^+|^2 \mathrm{d}x + \lambda^- \int_{\text{out}(C)} \sum_{i=1}^{n} |u_0^i - c_i^-|^2 \mathrm{d}x.$$

Empirically, this model appears to give the union of the objects in the different channels. In this section, we will try to compare the two different approaches. While the model in (1) may seem very different from the model presented in Section 3, with a little calculation we will see that it does follow a logic based format. Taking a Taylor expansion of the interpolation function that finds the intersection of a region, gives the vector valued functions, i.e.

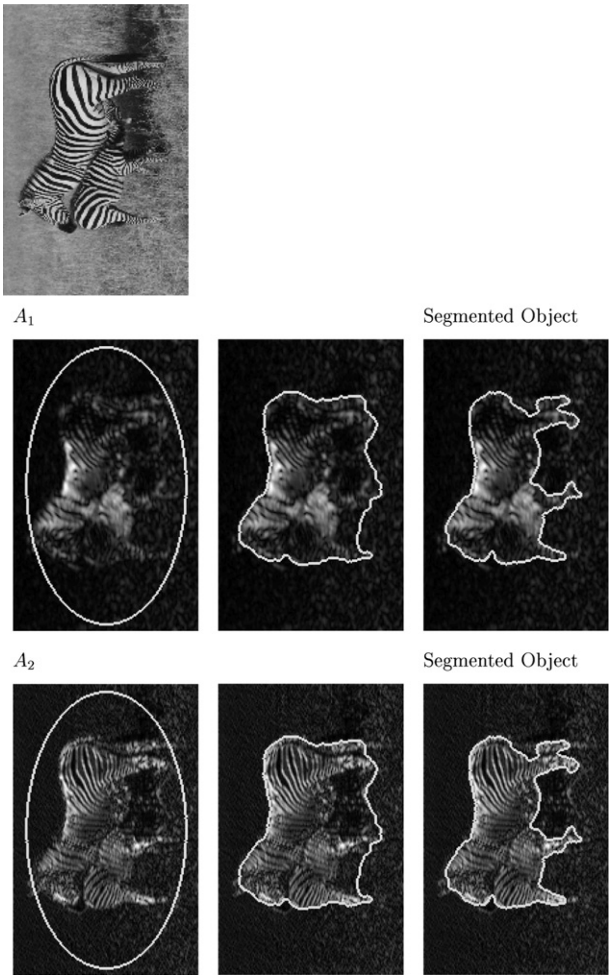


Fig. 18. The original image of a zebra, followed by time evolution of active contours on two Gabor transforms using logic operator $A_1 \cup A_2$.

$$1 - \sqrt{(1 - z_1)(1 - z_2)} = \frac{z_1 + z_2}{2} + O(z_1 z_2).$$

The leading order term $z_1 + z_2$ corresponds to the vector based model. Thus within the framework of our logic model, the vector model is similar to taking the intersection of the inside of the contour and the intersection of the outside of the contour. From this perspective, there is a conflicting objective of the function inside the contour, and the function outside the contour, in the vector model.

We want to illustrate the difference between the vector model and the logic models. We do this is in two ways: first as a 1-D analysis in which we can work the functional out analytically, second, a carefully chosen example to illustrate the differences.

Let us consider the 1-D example in Fig. 19. In this example, a is the length of the object in Channel A_1, b is the length of the object in A_2, T is the total length of the channels. We will calculate the functionals for the vector valued and the logic models. In this example, t is the position at which the contour is located. The intensity values are c inside the object, and d outside.

For this example we can find the exact solution for the functionals in terms of a, b, c, d, t, and T. The exact form of the variational formulations can be derived for both the vector model and the logic models. We will compare the fitting terms so we set $\mu = 0$ for the functions in (6) and (1). Three segments need to be considered to calculate the functional.

For $t < a$, we have:

$$F_{\text{vec}}(t) = \frac{(c-d)^2}{T-t}((T-a)(a-t)+(T-b)(b-t)),$$

$$F_{A_1 \cup A_2}(t) = \frac{(c-d)^2}{T-t}T - \frac{(c-d)^2}{T-t}\sqrt{(T-(a-t)(T-a))(T-(T-b)(b-t))},$$

$$F_{A_1 \cap A_2}(t) = \frac{(c-d)^2}{T-t}\sqrt{(T-a)(a-t)(T-b)(b-t)}.$$

For $a \leqslant t \leqslant b$, we have:

$$F_{\text{vec}}(t) = (c-d)^2\left(\frac{a(t-a)}{t} + \frac{(T-b)(b-t)}{(T-t)}\right),$$

$$F_{A_1 \cup A_2}(t) = \frac{(c-d)^2}{T-t}T - \frac{(c-d)^2}{T-t}\sqrt{T(T-(T-b)(b-t))},$$

$$F_{A_1 \cap A_2}(t) = \frac{(c-d)^2}{t}T - \frac{(c-d)^2}{t}\sqrt{T(T-a(t-a))}.$$

For $b < t < T$, we have:

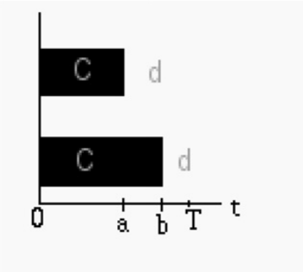


Fig. 19. This a 1-D example where channel A_1 and A_2 both start from the same point. a is the length of A_1, b is the length of A_2, T is total length of the channels. The intensity inside the image is c and outside is d. $A_1 \cap A_2 = A_1$, while $A_1 \cup A_2 = A_2$.

$$\begin{aligned}
F_{\mathrm{vec}}(t) &= \frac{(c-d)^2}{t}(a(t-a)+b(t-b)),\\
F_{A_1\cup A_2}(t) &= \frac{(c-d)^2}{t}T - \sqrt{(ab(t-a)(t-b))},\\
F_{A_1\cap A_2}(t) &= \frac{(c-d)^2}{t}T\\
&\quad - \frac{(c-d)^2}{t}\sqrt{(1-a(t-a))(1-b(t-b))}.
\end{aligned}$$

When we graph the functionals (see Fig. 20), we find that F_{vec} has two minima: one at $t=a$ the other at $t=b$. One of the minima is global and the other is local depending on a, b, and T. If the contour C starts inside the smaller object, the functional will go to the minimum that corresponds to is the intersection of the objects while if the contour starts outside the smaller object, it converges to $t=b$ which corresponds to the union of the two objects. If the contour is in between, it will go to the closest edge.

Graphing $F_{A_1\cap A_2}$ in Fig. 20B, we see that the only minimum occurs at the intersection of the two objects, in our case it is at $t=a$. Likewise, the minimum for $F_{A_1\cup A_2}$ occurs at the union of the two channels, which is at $t=b$. The above observations are verified in Fig. 21, which shows the actual results of segmentation using the vector model and the two logic models.

This 1-D example gives some insight into the differences between the vector and logic models. We see that the vector model segments the image depending on the initial contour, rather than according to a global logic criteria. The logic models do not depend on the initial contour to find the global results, in a piecewise constant image.

On the other hand, to use the logic models properly, one needs to know a priori which logic operator to use. If the initial contour is chosen carefully, the vector model can do a good job in choosing a desirable solution. An example of such a situation

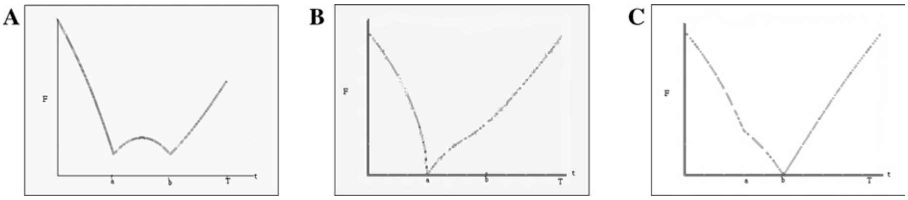


Fig. 20. (A) This is the graph of the objective function for the vector valued model. The t-axis denotes the position of the contour in the model. It has 2 local minima, one at the intersection of the two objects located at $(a, F(a))$, and the other is located at $(b, F(b))$ which corresponds the union of the objects. The minima depends on where the initial contour is located. If it is inside the smaller object, the model converges to the intersection, otherwise it converges to the union. (B) This graph is the region based logic model that finds the $A_1 \cap A_2$ of the two objects. Its global and only minimum occurs at $(a, F(a))$. (C) This graph is the region based logic model that finds $A_1 \cup A_2$ of the two objects. Its global and only minimum occurs at $(b, F(b))$.

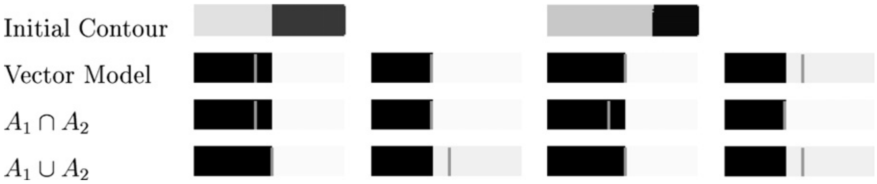


Fig. 21. This is a simple example that shows the object found for the vector model depends on the initial contour for the segmentation, if the contour begins inside the object, it will find $A_1 \cap A_2$ of the images, otherwise it will find $A_1 \cup A_2$ of the images. Calculating the logic model using $A_1 \cap A_2$, the same segmentation is found for the initial contour inside or outside the object. Likewise $A_1 \cup A_2$ is found for the initial contour inside and outside the object.

is given in Fig. 22. This is an example of the Kanisza face/vase image which has been artificially occluded. Looking at the dark inside object, one sees a vase, but looking at the outside object one sees two faces.

In Fig. 23 we have information we want to preserve that is both inside and outside the object. The occlusion can be of the ''inside'' object, which we will define as the vase, when we would want the intersection of the outside object (i.e., faces). However, the occlusion can also be of the ''outside'' (faces) object, in which case we want the intersection of the inside. We can make the vector model look for the intersection of the vase (i.e., $A_1 \cap A_2$) if the initial contour is small and inside both channels. If the initial contour surrounds the objects in both channels it will act as the logic model for the union of the vase ($A_1 \cup A_2$), or intersection of the faces. Since the vector model depends on the initial contour, we can choose an initial contour close to the

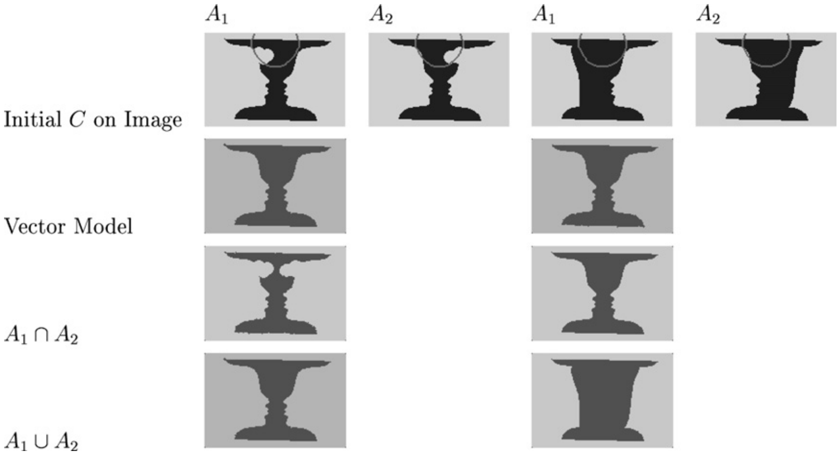


Fig. 22. In this example, we show when the vector model might be more appropriate than the logic model. In the first set of images, we need to take $A_1 \cup A_2$ to recover the Kanizsa face vase image, while in the second we need to take $A_1 \cap A_2$ to get the same image back. This requires a priori knowledge at the time the image is being segmented, while the vector model finds the desired object using the same initial contours in both cases.

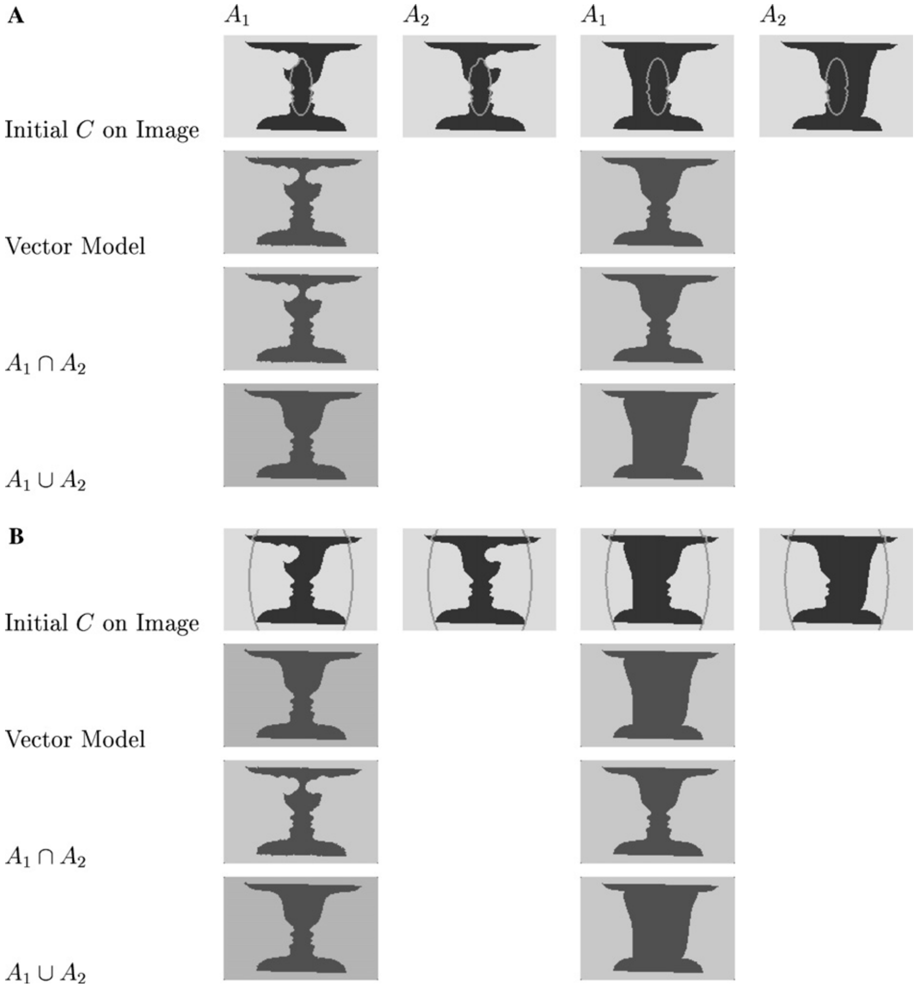


Fig. 23. In this example, we show how using the vector model one can get different logic models through the manipulation of the initial contours. For (A) the initial contour is inside the object in both cases, and the vector model computes the intersection of the inside objects which is the logic equivalent of $A_1 \cap A_2$. For (B) the initial contour is outside the object in both cases and the vector model computes the the intersection of the outside objects which is the logic model equivalent to $A_1 \cup A_2$.

boundary of the inside object (vase) or the outside object face to get the desired affect.

Rather than choosing which logic operation to use, a careful choice of the initial contour for the vector model can give the desired result, which can correspond to either the union or the intersection of the inside object.

From these examples, we can draw the following general conclusion, the logic models will give a robust segmentation for union and intersection of the images, as well as any logical combination of channels. However in situations when one does not know ahead of time which model is preferable, a careful guess of the initial contour using the vector model may give the best compromise.

7. Conclusion

A generalized model for multi-channel images has been presented. It allows the user to choose the information that is extracted from the set of images using general logic combinations of union, intersection, and negation. This was presented for region-based models. We demonstrated the viability of the model using region-based models using the particular example of the Chan and Vese (2001) active contours without edges model, is able to find the unions and intersections of regions (objects).

Experiments on two channel and three channel systems were presented. They demonstrated the ability of the models to detect the objects as described by the logic operations accurately and robustly. This model assumes the same scalar segmentation model on all channels (in our case active contours without edges). For a given set of images, we expect this model to work, if the scalar model applied to the channels seperately will identify the objects successfully. The parameters of the model are chosen based on the noise level of the images.

We compared the logic models to the vector model derived previously. This showed some interesting elements to the concept of a global minimum versus multiple minima. An example was made that shows that perhaps sometimes a local minimum, i.e., a vector model is preferable.

It would be interesting and straightforward to apply this model to object tracking in movie sequences, and in registration of multi-channel images.

References

Aubert, G., Kornprobst, P., 2001. Mathematical Problems in Image Processing. Springer, Berlin.

Chan, T., Sandberg, B., Vese, L., 1999. Active contours without edges for vector-valued images. J. Visual Commun. Image Representation 11, 130–141.

Chan, T., Sandberg, B., Vese, L., 2002. Active contours without edges for textured images, CAM report 02-28.

Chan, T., Vese, L., 2001. Active contours without edges. IEEE Trans. Image Process. 10 (2), 266–277.

Dibos, F., Koepfler, G., 1997. Color segmentation using a variational formulation, Actes du 16me Colloque GRETSI, pp. 367–370.

Guichard, F., 1998. A morpholocial affine and galilean invarient scale space for movies. IEEE Image Process. 7 (3), 444–456.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: active contour models. Int. J. Comput. Vision 1, 321–331.

Mumford, D., Shah, J., 1989. Optimal approximation by piecewise-smooth functions and associated variational problems. Commun. Pure Appl. Math. 42, 577–685.

Osher, S., Fedkiw, R., 2002. Level Set Methods and Dynamic Implicit Surfaces. Springer, Berlin.

Osher, S., Sethian, J.A., 1988. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulation. J. Comput. Phys. 79, 12–49.

Paragios, N., Deriche, R., 1999. Geodesic active regions for supervised texture segmentation. In: Proceedings of the 7th International Conference on Computer Vision, pp. 100–115.

Sapiro, G., 1997. Color snakes. Comput. Vision Image Und., 247–253.

Sapiro, G., 2001. Geometric Partial Differential Equations and Image Analysis. Cambridge University Press, Cambridge.

Sapiro, G., Ringach, D.L., 1996. Anisotropic diffusion of multivalued images with applications to color filtering. IEEE Trans. Image Process. 5, 1582–1586.

Tu, Z., Chen, X., Yuille, A., Zhu, S., 2003. Image parsing: unifying segmentation, detection and recognition. In: Proceedings ICCV.

Vese, L., Chan, T., 2002. A multiphase level set framework for image segmentation using the mumford and shah model. Int. J. Comput. Vision 50 (3), 271–293.

Yezzi, A., Soatto, S., 2003. Stereoscopic segmentation. Int. J. Comput. Vision 53 (1), 31–43.

Zhao, H., Chan, T., Merriman, B., Osher, S., 1996. A variational level set approach to multiphase motion. J. Comput. Phys. 127, 179–195.

Zhu, S.C., Yuille, A., 1996. Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation. IEEE Trans. Pattern Anal. Machine Intell. 18 (9), 884–900.