Check for updates

A view of computational models for image segmentation

Laura Antonelli[1] · **Valentina De Simone**[2] · **Daniela di Serafino**[3]

Abstract

Image segmentation is a central topic in image processing and computer vision and a key issue in many applications, e.g., in medical imaging, microscopy, document analysis and remote sensing. According to the human perception, image segmentation is the process of dividing an image into non-overlapping regions. These regions, which may correspond, e.g., to different objects, are fundamental for the correct interpretation and classification of the scene represented by the image. The division into regions is not unique, but it depends on the application, i.e., it must be driven by the final goal of the segmentation and hence by the most significant features with respect to that goal. Thus, image segmentation can be regarded as a strongly ill-posed problem. A classical approach to deal with ill posedness consists in incorporating in the model a-priori information about the solution, e.g., in the form of penalty terms. In this work we provide a brief overview of basic computational models for image segmentation, focusing on edge-based and region-based variational models, as well as on statistical and machine-learning approaches. We also sketch numerical methods that are applied in computing solutions to these models. In our opinion, our view can help the readers identify suitable classes of methods for solving their specific problems.

L. Antonelli, V. De Simone, and D. di Serafino contributed equally to this work.

Daniela di Serafino—deceased.

✉ Laura Antonelli
laura.antonelli@cnr.it

✉ Daniela di Serafino
daniela.diserafino@unina.it

Valentina De Simone
valentina.desimone@unicampania.it

1 Institute for High Performance Computing and Networking (ICAR), CNR, Via Pietro Castellino, 111, 80131 Napoli, Italy

2 Department of Mathematics and Physics, University of Campania "Luigi Vanvitelli", viale Abramo Lincoln, 5, 81100 Caserta, Italy

3 Department of Mathematics and Applications "R. Caccioppoli", University of Naples Federico II, Via Cintia, Monte S. Angelo, 80126 Napoli, Italy

⚛ Springer

1 Introduction

Image segmentation is a fundamental task of image processing, image analysis, image understanding, and pattern recognition. It has a very long history, whose origin may be dated back to about 50 years ago. A seminal paper is [1], where the authors pointed out that an important component of the Stanford Research Institute automation project was a set of programs providing the automaton with a means of interpreting visual data.

While it is possible to accurately represent the information in a real scene by an image, this representation alone does not enable us to highlight specific properties of the scene. Conversely, a description in terms of "natural" elements of the image, such as regions and boundaries of the visualized objects, represented in a uniform manner, provides easy access to useful global information, thus allowing recognition and extraction of specific image features. Thus, to generate a description of specific elements of the image, it is customary to *segment* the image into more parts (or *segments*). Figure 1 shows an example of two main types of segmentation, i.e., *instance segmentation*, which identifies the object instance of each pixel for every known object within an image, and *semantic segmentation*, which identifies the object category of each pixel for every known object within an image.

Image segmentation is used in many application fields, such as medical imaging [2], microscopy imaging [3], remote sensing [4], and document image analysis [5]. The choice between semantic and instance segmentation is generally dependent on the goal of the classification or object detection step that follows the segmentation phase. For example, in the segmentation of terrain in satellite imagery, we may use the semantic segmentation to distinguish different land areas, like vegetation, ground, water and building, or we may use the instance segmentation to distinguish different common weeds in agricultural fields (i.e., separate instances of objects belonging to the same class).

Since different applications may require different partitions to extract significant features, there is no single standard method for image segmentation. Thus, the segmentation problem has not a unique result, as shown in Fig. 2, where different segmentations of the same image are shown, resulting from different segmentation criteria. On the other hand, different methods are not equally effective in segmenting a specific type of image (real scenes, synthetic images, medical images, etc.), and the criteria to define a successful segmentation depend on the desired goal of the segmentation itself. Therefore, segmentation remains a challenging problem in image processing and computer vision, in spite of several decades of research.

We present image segmentation as a highly ill-posed problem, and discuss basic models that take into account a-priori information about the solution, attempting to put these models into a coherent mathematical framework. We look at the inclusion
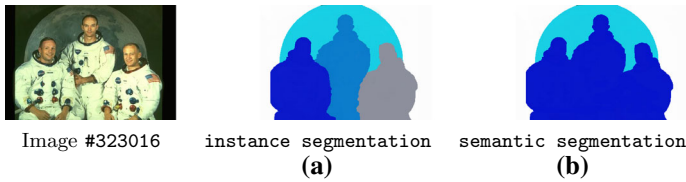
Fig. 1 Illustration of instance and semantic segmentation of the Berkeley database image #323016. The results displayed in (a) and (b) were produced by using Adobe Photoshop

Fig. 2 Segmentations of the Berkeley database image #323016 by different users (see https://www2.eecs. berkeley.edu/Research/Projects/CS/vision/grouping/resources.html.)

of a-priori information as a sort of regularization approach and show that it is ubiquitous in image segmentation models, from older "classical" ones to machine learning approaches, revealing links and similarities between them. Note that we focus on basic models in order to keep our discussion easy and get rid of technical details. We also sketch some numerical methods used in the application of the various models. Although this is only a simplified and partial view of image segmentation, we believe that it may give a contribution towards a better understanding of this huge field.

The rest of this paper is organized as follows. In Sect. 2 we present a mathematical formulation of image segmentation, and in Sect. 3 we discuss basic segmentation models, focusing on edge-based, region-based, statistical and machine learning ones. In Sect. 4 we give a quick overview of numerical techniques that may be used to solve the aforementioned models. Finally, we give some conclusions in Sect. 5.

2 Mathematical formulation of image segmentation

Let $\mathcal{I}$ be the set of the images defined in a domain $\Omega \subset \mathbb{R}^d$ ($d \geq 2$), $I_0 \in \mathcal{I}$ the observed image, and $\mathcal{P}_1, \ldots, \mathcal{P}_n$ logical predicates used to check n statements, expressed using features of the image, e.g., edges, smoothness, texture or color, so that $\mathcal{P}_k(A) = true$ if all the points of $A \subseteq \Omega$ satisfy the k-th statement. Just to give an example, in order to compute a two-region segmentation of a normalized gray-level image I_0, we can define $n = 2$ simple statements as follows, which involve the gray level of the intensity light to separate the background from the foreground:

$$\mathcal{P}_1(A) = \left\{ I^*(x) < \alpha, \forall x \in A \right\}, \quad \mathcal{P}_2(A) = \left\{ I^*(x) \geq \alpha, \forall x \in A \right\}$$

where I^* is a suitable approximation of I_0 and $\alpha \in (0, 1)$ is a suitable value.

Generalizing the definition in [6], the instance segmentation S of I_0 according to the predicates $\mathcal{P}_k$, $k = 1, \ldots, n$, consists of finding a decomposition of Ω into m components Ω_i, with $i = 1, \ldots, m$ and $m \geq n$, such that

1. $\Omega_i \neq \emptyset$, $\forall i \in \{1, \ldots, m\}$;
2. $\overset{\circ}{\Omega}_i \bigcap \overset{\circ}{\Omega}_j = \emptyset$, $\forall i, j \in \{1, \ldots, m\}$ with $i \neq j$, where $\overset{\circ}{\Omega}_k$ denotes the interior of Ω_k;
3. $\bigcup\limits_{i=1}^{m} \Omega_i = \Omega$;
4. $\forall i \in \{1, \ldots, m\}$ $\exists! \, k \in \{1, \ldots, n\}$ such that

   i. $\mathcal{P}_k(\Omega_i) = true$;
   ii. $\mathcal{P}_k(\Omega_i \bigcup \Omega_j) = false$, $\forall \, j \in \{1, \ldots, m\}$ with $j \neq i$.

   By adding to item 4
   iii. $\mathcal{P}_k(\Omega_j) = false$, $\forall \, j \in \{1, \ldots, m\}$ with $j \neq i$.

we also obtain the semantic segmentation.

We can define the segmentation S of I_0 as follows too. Let Σ be the set of possible segmentations of the images in $\mathcal{I}$ according to some criteria defined by the predicates $\mathcal{P}_k$. Then S can be expressed as

$$S = (u^*, I^*),$$

where u^* is a curve that matches the boundaries of the decomposition of Ω, i.e., $u^* = \cup_i \partial\Omega_i$[1], and I^* is a piecewise-smooth function defined on Ω that approximates I_0. In particular, we may assume that the restriction of I^* to any set $\overset{\circ}{\Omega}_i$ is piecewise differentiable. The segmentation S may be also identified directly by using a labeling operator Φ, i.e.,

$$S = \Phi(I^*), \tag{1}$$

where

$$\Phi(I(x)) = l_i \text{ if } x \in \Omega_i,$$

$I(x)$ is the value of I associated with x, and $l_i \in \mathcal{N} = \{l_1, l_2, \ldots, l_m\}$ is a label.

3 Basic segmentation models

We look at image segmentation as an ill-posed problem, whose solution is highly undetermined. Classical approaches for computing a solution of an ill-posed problem require additional information that enforces uniqueness and stability. To this end, suitably defined penalty terms can be applied. Then, the solution is obtained by minimizing an energy functional E containing a fidelity term $\mathcal{F}$ that measures the consistency of the candidate segmentation with the observed image, and a penalty term $\mathcal{P}$ that promotes

---

[1] The parametric representation of the curve u is defined by a continuous map $\gamma : X \longrightarrow \mathbb{R}^d$, where $X \subset \mathbb{R}$ is an interval and $u = \gamma(X)$. With a little abuse of notation we identify the curve u with the function γ.

solutions with suitable properties:

$$(I^*, u^*) := \arg \min_{(I,u)} E(I, u; I_0) = \arg \min_{(I,u)} (\mathcal{F}(I, u; I_0) + \lambda \mathcal{P}(I, u)). \qquad (2)$$

Here $\lambda > 0$ is a parameter that generally needs careful tuning to suitably balance $\mathcal{F}$ and $\mathcal{P}$ (see, e.g., [7] and the references therein).

The minimization problem (2) can be solved by writing the Euler-Lagrange equations, which can be derived by integrating by parts the energy functional and using the Gauss theorem along with the fundamental lemma of the calculus of variations. Then a numerical solution can be computed by applying a gradient descent approach, where the descent direction is parameterized through an artificial time, and by a finite-difference discretization. A widely used and effective alternative consists in discretizing problem (2) and then solving it by a numerical optimization method. We will come back to these two approaches in Sect. 4.

Recently, machine learning techniques have been successfully applied to segmentation problems. The key idea is to tune a generic model to a specific solution through learning against sample data (training data). The learning phase extracts prior information to be embedded into the penalty term from a large dataset containing pairs of type *(image, ground-truth label)* [8]. Machine learning approaches using unlabeled image data as training datasets are also available. Although these techniques successfully solve image segmentation problems, sometimes outperforming state-of-the-art variational models, they have been often designed on-demand for specific tasks used as "black-box" models and require a large amount of data to produce results.

In the next subsections we provide some examples of image segmentation models. Note that we focus on basic models, with the aim of providing a general idea of these approaches while avoiding technical details that are outside the scope of this work. It is also worth observing that these models are the basis of modern ones, developed either to improve the effectiveness of the original models in some applications [9] or to complement and refine Machine Learning techniques for segmentation [10].

3.1 Edge-based models

Edge-based models aim at finding $u^* = \cup_i \partial \Omega_i$ by solving the minimization problem (2) with respect to the curve u (note that I and I^* are not explicitly considered in this case). These models include the so-called *Active Contours* [11] or *Snakes*. Here the fidelity and regularization terms act as an internal force and an external force, respectively, which move the curve within the image to find the boundaries of the sets Ω_i. More precisely, the energy functional takes the form

$$E_{AC}(u) = \underbrace{\int_0^1 g(|\nabla I_0(u(s))|)^2 ds}_{\mathcal{F}} + \lambda \underbrace{\int_0^1 |u'(s)|^2 ds}_{\mathcal{P}}, \qquad (3)$$

where I_0 is the observed image, g is an edge-detector function and the curve u is parametrized by $s \in [0, 1]$. The first term attracts the curve toward the boundaries,

whereas the second one controls its smoothness, and as a result the curve u changes its shape like a snake.

The evolving curve is driven by surface properties, such as curvature and normal direction, and by image features, such as gray levels, hue or saturation in color images, and intensity gradient in 2D images or change in slope in 3D ones. For example, the mean curvature can be used and in this case the edge-detector function is also responsible for stopping the curve on the edges. The function g may be defined as

$$g(|\nabla I_0|) = \frac{1}{1 + |\nabla(G_\sigma * I_0)|^2},$$

where g is a positive and decreasing function, G_σ is the Gaussian kernel with standard deviation σ, and $*$ denotes the convolution operator.

In a Lagrangian approach, an initial curve is evolved by

$$\frac{\partial u}{\partial t} + \mathcal{L}(u) = 0, \tag{4}$$

where $\mathcal{L}$ is a differential operator. The simplest evolution is given by $\mathcal{L}(u) = FN$, where N is the normal to the curve and F is a constant that determines the speed of evolution. More generally, the evolution is driven by an external force. For example, in the mean-curvature evolution, $\mathcal{L}(u) = \kappa N$, where κ is the Euclidean curvature of u [12].

When u has an explicit representation, it is not easy to deal with topological changes like merge and split, and a re-parametrization of the curve may be required. Therefore, the evolution of the curve u is commonly described by level-set methods [13], thanks to their ability to follow topology changes, cusps and corners. In a level set approach, the curve u is implicitly represented by the zero-level set of a function $\phi(t, x)$, i.e., $u = \{x \in \Omega : \phi(t, x) = 0\}$. The level set formulations of the simplest evolution and the mean-curvature one read, respectively:

$$\frac{\partial \phi}{\partial t} = F|\nabla \phi|, \ F \in \mathbb{R} \quad \text{and} \quad \frac{\partial \phi}{\partial t} = \text{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)|\nabla \phi|.$$

3.2 Region-based models

Region-based models provide directly the segmentation by means of the image partition $\{\Omega_i, \ i = 1, \ldots, m\}$. Region-growing models are among the simplest models falling in this class, and in order to obtain accurate segmentations they have been merged with variational approaches where the evolution changes according to the minimization of an energy functional including region-based terms [14].

A very popular region-growing model was proposed by *Mumford and Shah* [15]. In this case, the functional E in (2) takes the form

$$E_{MS}(I, u) = \underbrace{\int_\Omega (I - I_0)^2 dx}_{\mathcal{F}} + \underbrace{\lambda \int_{\Omega-u} |\nabla I|^2 dx + \mu\, len(u)}_{\mathcal{P}}, \tag{5}$$

where $len(u)$ denotes the length of u, and λ and μ are positive parameters. The term $\mathcal{F}$ attempts to achieve the minimum distance between I_0 and its piecewise-smooth approximation I, and $\mathcal{P}$ attempts to reduce the variation of I within each set Ω_i while keeping the curve u as short as possible. Minimizing (5) in a suitable space provides an optimal pair (I^*, u^*) representing a simplified description of I_0 by means of a function with bounded variation and a set of edges [15]. Finally, in [16] the Mumford and Shah model is formulated as a deterministic refinement of a probabilistic model for image restoration.

A simplified version of the Mumford-Shah model is its restriction to piecewise-constant functions. The *Chan-Vese* model [17] is a particular case of the simplified version, aimed at obtaining a two-phase segmentation where the piecewise-constant function assumes only two values. Its functional E takes the following form:

$$E_{CV}(I, c_{in}, c_{out}) = \underbrace{\left(\int_\Omega H(I)\,(c_{in} - I_0)^2\,dx + \int_\Omega (1 - H(I))\,(c_{out} - I_0)^2\,dx \right)}_{\mathcal{F}}$$
$$+ \lambda \underbrace{\int_\Omega |\nabla H(I)|\,dx}_{\mathcal{P}}, \tag{6}$$

where H is the Heaviside function and c_{in} and c_{out} are the average values of the intensity in the foreground and background of the image, respectively. The solution I^* is the best approximation to I_0 among all the functions that take only two values.

Minimizing (6) is a nonconvex problem, thus solution methods may get stuck into local minima and result in unsatisfactory segmentations. Aiming to overcome this drawback, some strategies have been proposed, including the convexification of the functional by taking advantage of its geometric properties. An example is given by the two-phase partitioning model introduced by *Chan, Esedoḡlu and Nikolova* [18]:

$$E_{CEN}(I, c_{in}, c_{out}) = \underbrace{\int_\Omega \left((c_{in} - I_0)^2 I + (c_{out} - I_0)^2 (1 - I) \right) dx}_{\mathcal{F}}$$
$$+ \lambda \underbrace{\int_\Omega |\nabla I|\,dx}_{\mathcal{P}} \tag{7}$$

with $0 \le I \le 1$ and $c_{in}, c_{out} > 0$.

3.3 Statistical models

Statistical models usually provide a conditional probability, $P(S|I_0)$, of a segmentation $S \in \Sigma$ given the observed image I_0, and then select the segmentation with the highest probability. In the *Maximum A Posteriori (MAP)* approach the segmentation is given by

$$S^* = \arg \max_{S \in \Sigma} P(S|I_0).$$

According to the Bayes rule,

$$P(S|I_0) = \frac{P(I_0|S)P(S)}{P(I_0)},$$

where $P(S)$ is the prior probability measuring how well S satisfies certain properties of the given image, and $P(I_0|S)$ is the conditional probability measuring the likelihood of I_0 given S (see, e.g., [19]). Since the probability $P(I_0)$ is constant, the segmentation can be obtained by maximizing $P(I_0|S)P(S)$.

Markov Random Field (MRF) models offer a framework to define prior and likelihood by capturing properties of the image such as texture, color, etc. [20]. The segmentation is formulated within an image labeling framework, i.e., $S = \Phi(I(x))$, where the problem is reduced to find the labeling which maximizes the posterior probability. Label dependencies are modeled by an MRF.

Then, using the Hammersley-Clifford theorem, we get the Gibbs distribution

$$P(S) = \frac{1}{Z} \exp(-U(S)),$$

where the energy function U takes the form

$$U(S) = \sum_{c \in C} V_c(S_c),$$

C is the set of cliques of S, $V_c(S_c)$ is the potential of the clique $c \in C$ having the label configuration S_c, and Z is a normalizing constant.

When the nature of the observed image is unknown, the Gaussian distribution is often used to model the conditional probability $P(I_0|S)$. Setting

$$U_1(S) = -\ln P(I_0|S) \ \text{ and } \ U_2(S) = -\ln P(S),$$

we get $U(S) = U_1(S) + U_2(S)$, and then the original MAP estimation is equivalent to the following problem:

$$S^* = \arg \max_S P(S|I_0) = \arg \min_S U(S).$$

3.4 Machine learning models

Machine learning approaches, and in particular deep learning ones, are more and more used in solving image segmentation problems, also outperforming the previous approaches. Roughly speaking, machine learning approaches do not benefit from prior information on the solution as described above, but "learn" the segmentation from large training datasets. The aim of a machine learning approach is to define a segmentation model $f_\theta : \mathcal{I} \longrightarrow \Sigma$ such that the segmentation of I_0 can be obtained as $I^* = f_\theta(I_0)$. The function f_θ is usually nonlinear and θ is a large vector of parameters. The learning phase selects θ in order to minimize a loss functional $\mathcal{L}$ that measures the accuracy of the predicted segmentation $f_\theta(I_0)$.

In *supervised* machine learning, training data are available from databases of annotated segmentations, which provide a large number of pairs $(I_0, I^*) \in X \times Y \subset \mathcal{I} \times \Sigma$ ($X \times Y$ is named training set). The vector of parameters θ is obtained by minimizing a loss function plus a penalty term. For the sake of simplicity, we first consider a mean-square-error loss:

$$\theta^* = \arg \min_\theta \mathcal{L}(X, Y, \theta) = \arg \min_\theta \left(\sum_X \| f_\theta(I_0) - I^* \|^2 + \mathcal{P}_\theta(f_\theta(I_0)) \right). \quad (8)$$

Another widely used loss functions is the *Binary Cross Entropy* (BCE) loss, which measures the difference in information content between the actual and the predicted image segmentation:

$$\mathcal{L}_{BCE}(f_\theta(I_0), I^*) = -f_\theta(I_0) \log(I^*) - (1 - f_\theta(I_0)) log(1 - I^*).$$

It is based on the Bernoulli distribution and works well with equal data distributions among classes. Some variants of BCE, such as the *Weighted* BCE and the *Balanced* CE are also used for tuning false negatives and false positives, respectively. The *Shape-aware* (Sa) loss calculates the average point-to-the-curve Euclidean distance among points around the curve of the predicted segmentation, u^*, to the ground truth, $\bar{u}$, and use it as a coefficient to the cross-entropy (CE) loss function:

$$\mathcal{L}_{Sa}(f_\theta(I_0), I^*) = -\sum_{i \in \Sigma} CE(\theta(I_0), I^*) - \sum_{i \in \Sigma} i \, E_i \, CE(f_\theta(I_0), I^*),$$

where Σ contains the set of points where the prediction curve does not match the ground-truth curve, and $E_i = d(u_i^*, \bar{u}_i)$. The *Dice* loss, based on the well-known Dice coefficient metric, is also widely used to measure the similarity between two segmentations, and is defined as

$$\mathcal{L}_{Dice}(f_\theta(I_0), I^*) = 1 - \frac{2 f_\theta(I_0) I^* + 1}{f_\theta(I_0) + I^* + 1}.$$

In *unsupervised* machine learning, the training set is not equipped with annotated segmentations and the goal is to train f_θ to recognize specific patterns or image features
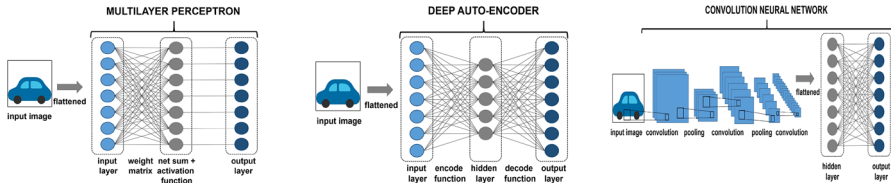
Fig. 3 Some neural network architectures used in image segmentation

in the data. This approach is sometimes referred to as self-supervised learning [21], because the information is extracted from the data themselves rather than from a set of "predictions" (i.e., given segmentations). Then the fidelity term in (8) takes the form

$$\sum_{\mathcal{I}} \parallel f_\theta(I_0) - \Phi(f_\theta(I_0)) \parallel^2,$$

where Φ is the labeling operator defined in (1).

In order to progressively extract higher-level features from the data, machine learning models use a multi-layer structure called *neural network*, consisting of successive function compositions. The number of layers is the depth of the model, hence the terminology *deep learning*. A neural network with L layers is a function

$$f_\theta : \mathcal{I} \times (H_1 \times \cdots \times H_L) \longrightarrow \Sigma, \quad f_\theta(I) = (f_L \circ f_{L-1} \circ \ldots \circ f_1)(I),$$

where $f_i : \mathbb{R}^{d_{i-1}} \times H_i \longrightarrow \mathbb{R}^{d_i}$ are the activation functions (each depending on a component θ_i of θ), $d_0 = d$ and $d_L = n$, with n equal to the number of features. The adjective "neural" comes from the fact that those networks are loosely inspired by neuroscience.

Neural network structures successfully used in image segmentation are the Multilayer Perceptron (MLP), the Deep Auto-Encoder (DAE) and the Convolutional Neural Network (CNN) [22–24]. Their basic schemes are shown in Fig. 3. The MLP is a neural network connecting multiple layers in a directed graph, which means that the signal path through the nodes only goes one way. Each node, apart from the input nodes, has a nonlinear activation function. An MLP uses backpropagation as a supervised learning technique. The DAE network structure typically consists of $2L$ layer functions, where the first L layers act as an encoding function with the input to each layer being of lower dimension than the input to the previous layer, and the remaining L layers increase the size of their inputs until the final layer has the same dimension as the image input. The first L layers are an MLP. CNNs divide the image into small areas and scan it one area at a time, to identify and extract features that are used to classify the image. A CNN mainly consists of three layers:

- *convolutional layer*: the image is analyzed a few pixels at a time to extract low-level features (edges, color, gradient orientation, etc.);
- *nonlinear layer*: an element-wise activation function creates a feature map with probabilities that each feature belongs to the required class;
- *pooling* or *downsampling layer*: the amount of features and computations in the network is reduced, hence controlling overfitting.

Among well-known deep neural network architectures successfully used in image segmentation, we mention SegNet [25], U-Net [26], and FCN [27].

4 Numerical techniques for segmentation models

The minimization in (2) is usually nontrivial and requires appropriate methods, taking into account the specific application. In this section we provide a brief summary of numerical methods that can be applied to segmentation models. We consider two approaches: *first discretize then optimize* and *first optimize then discretize*. In the former, all the quantities in (2) are discretized a priori and then optimization methods are applied to the resulting minimization problem in $\mathbb{R}^n$. In the latter, we first write optimality conditions for (2), which are generally partial differential equations (PDEs), and then solve those equations by suitable numerical methods, which discretize the equations. Finally, we also sketch some filtering techniques used in image segmentation, although they are not directly applied to the minimization problem (2). This is motivated by their use in some segmentation approaches, such as those based on deep learning.

For the sake of simplicity, here we consider $S = I$ (i.e., we neglect u in the segmentation $S = (I, u)$). For 2D images ($d = 2$) we denote by Ω_{n_x,n_y} the discretization of Ω consisting of a grid of $n_x \times n_y$ pixels,

$$\Omega_{n_x,n_y} = \{(i, j) : i = 1, \ldots, n_x, \ j = 1, \ldots, n_y\}.$$

We also identify each pixel with its center and denote by $S_{i,j}$ the value of S in (i, j). Finally, we consider the forward and backward difference operators defined as follows:

$$D^{+x}I_{i,j} = I_{i+1,j} - I_{i,j}, \quad D^{-x}I_{i,j} = I_{i,j} - I_{i-1,j},$$
$$D^{+y}I_{i,j} = I_{i,j+1} - I_{i,j}, \quad D^{-y}I_{i,j} = I_{i,j} - I_{i,j-1},$$

where we assume

$$I_{i-1,j} = I_{i,j} \ \text{for } i = 1, \quad I_{i,j-1} = I_{i,j} \ \text{for } j = 1,$$
$$I_{i+1,j} = I_{i,j} \ \text{for } i = n_x, \quad I_{i,j+1} = I_{i,j} \ \text{for } j = n_y,$$

i.e., we define by replication the values of I with indices outside Ω_{n_x,n_y}. Likewise, for 3D images the discretization of the image domain consists of a grid of $n_x \times n_y \times n_z$ voxels,

$$\Omega_{n_x,n_y,n_z} = \{(i, j, k) : i = 1, \ldots, n_x, \ j = 1, \ldots, n_y, \ k = 1, \ldots, n_z\},$$

and the forward and backward difference operators are defined as follows:

$$D^{+x}I_{i,j,k} = I_{i+1,j,k} - I_{i,j,k}, \quad D^{-x}I_{i,j,k} = I_{i,j,k} - I_{i-1,j,k},$$
$$D^{+y}I_{i,j,k} = I_{i,j+1,k} - I_{i,j,k}, \quad D^{-y}I_{i,j,k} = I_{i,j,k} - I_{i,j-1,k},$$

$$D^{+z}I_{i,j,k} = I_{i,j,k+1} - I_{i,j,k}, \quad D^{-z}I_{i,j,k} = I_{i,j,k} - I_{i,j,k-1}.$$

For simplicity, henceforth we consider $d = 2$.

4.1 First discretize then optimize

Numerical optimization offers a large variety of methods to compute the segmentation by solving the minimization problem coming from a discretization of (2), possibly subject to constraints that can drive the segmentation towards particular features. The choice of the optimization method depends on the properties of the objective function and/or the constraints.

Roughly speaking, at iteration k, optimization methods for nonlinear problems generate a function $\widetilde{E}(I; I_k)$ that approximates the discretized objective function E around I_k, and minimize it to obtain the next iterate (see, e.g., [28]). For example, given I_k, the $(k + 1)$-st iteration may be written as

$$\text{Define } \widetilde{E}(I; I_k) \text{ that approximates } E(I; I_0)$$
$$\widetilde{I}_{k+1} = \arg\min_{I} \widetilde{E}(I; I_k)$$
$$I_{k+1} = I_k + \alpha_k(\widetilde{I}_{k+1} - I_k)$$

where the step length α_k satisfies some criterion.

"Classical" optimization techniques, such as gradient or Newton-type methods, require regularity assumptions on the objective function (and the constraints, if any). However, many segmentation models are modeled as non-smooth optimization problems. There are two main approaches to deal with non-differentiability: smoothing and non-smoothing [29]. The former formulates the problem as a suitable smooth one and applies the aforementioned classical optimization methods. The latter does not modify the mathematical model, and thus uses methods not requiring smoothness. For the purpose of illustration, here we focus on (7), where non-smoothness comes from a discretization of the TV term.

A regularized discrete TV may be obtained as follows:

$$\int_{\Omega} |\nabla I| \, dx \approx \sum_{i,j} \sqrt{(D^{+x}I_{i,j})^2 + (D^{+y}I_{i,j})^2 + \epsilon},$$

where $\epsilon > 0$ is "suitably small", but other regularized versions may be considered, e.g., based on Huber-like functions [30]. In this case, gradient and higher-order methods [31–35] can be used efficiently. Another way of introducing smoothness consists in splitting the variables into their positive and negative parts (thus doubling the number of unknowns) and introducing new constraints, and then applying first- or higher-order methods for smooth problems, such as in [36–38].

Non-smoothing approaches avoid regularization of the non-smooth terms in the optimization problem. This is the case, for example, of methods based on forward-backward splitting techniques, such as proximal-gradient methods [39, 40], and the forward-backward Expectation Maximization (EM) method in [41]. ADMM and split

Bregman methods do not use smooth approximations too [7, 42–46]. The success of these approaches is based also on the availability in closed (and cheap) form of the proximal operator of the ℓ_1 norm by means of the well-known soft-thresholding, defined as

$$[\mathcal{S}(x, \gamma)]_{i,j} = \text{sign}(x_{i,j}) \cdot \max\left(|x_{i,j}| - \gamma, 0\right),$$

with $\gamma > 0$. The difficulties associated with the non-differentiability of the TV functional may be also overcome by reformulating the minimization problem as a saddle-point problem and solving it by a primal-dual algorithm such as the Chambolle-Pock one [47, 48].

EM algorithms [49] are also widely used to solve statistical models. They are based on the idea of splitting the (negative) log-likelihood into two terms and alternating between the computation of the expectation and its minimization.

Finally, stochastic versions of the previous methods are used in segmentation with deep learning, to limit the computational cost. The idea is to use only random samples of the data at each iteration, to estimate first-order and possibly second-order information according to the loss function, with the aim of significantly reducing the computation and hence the time [50, 51].

4.2 First optimize then discretize

Reducing imaging problems to PDEs is many years old, because of the availability of a large amount of methods and software for solving PDEs. PDE-based methods have been introduced in different ways, such as the Perona-Malik filtering [52], directly based on properties of the PDE [53], and the axiomatic scale space theory [54, 55].

In a variational approach, one derives the first-order optimality conditions via smoothing regularization, if it is needed. Let us consider, for example, the level-set formulation of the Chan-Vese model (6), where I is represented by a function ϕ such that $\phi(x) = 0$ provides the curve separating two regions of I (when $I = I^*$ the two regions identify the segmentation). Keeping c_{in} and c_{out} fixed and writing the Euler-Lagrange equations in a gradient-flow approach, we get

$$
\begin{aligned}
\frac{\partial \phi}{\partial t}(t, x) &= \delta_\varepsilon(\phi) \left(\lambda \, \text{div}\left(\frac{\nabla \phi}{|\nabla \phi|} \right) - (c_{in} - I_0)^2 + (c_{out} - I_0)^2 \right) && \text{in } (0, +\infty) \times \Omega, \\
\phi(0, x) &= \phi_0(x) && \text{in } \Omega, \qquad\qquad (9) \\
\frac{\delta_\varepsilon(\phi)}{|\nabla \phi|} \frac{\partial \phi}{\partial N} &= 0 && \text{on } \partial \Omega,
\end{aligned}
$$

where δ_ε is a regularized version of the Dirac measure, ϕ_0 is the initial-level function, and N is the exterior normal to the boundary $\partial \Omega$ [17].

Finite-difference schemes are popular methods for the numerical solution of (9). Of course, the discretization used in image segmentation must take into account the nature and the properties of the operators involved in the model. For example, edge preserving is similar to shock capturing in computational fluid dynamics, and hence finite-difference schemes based on hyperbolic conservation laws can be used [56].

Just to give an example, the level-set equation

$$\frac{\partial \phi}{\partial t} = F|\nabla \phi|$$

in Sect. 3.1 can be solved by using an upwind numerical scheme:

$$\phi^{n+1} = \Psi(\phi^n), \quad \Psi(\phi_{i,j}^n) = \phi_{i,j}^n - \Delta t(\max(F,0)\nabla^+\phi_{i,j}^n + \min(F,0)\nabla^-\phi_{i,j}^n),$$

where

$$\nabla^+\phi_{i,j}^n = \Big(\max\Big(\max(D^{-x}\phi_{i,j}^n, 0)^2, -\min(D^{+x}\phi_{i,j}^n, 0)^2\Big)$$
$$+ \max\Big(\max(D^{-y}\phi_{i,j}^n, 0)^2, -\min(D^{+y}\phi_{i,j}^n, 0)^2\Big)\Big)^{1/2},$$
$$\nabla^-\phi_{i,j}^n = \Big(\max\Big(\max(D^{+x}\phi_{i,j}^n, 0)^2, -\min(D^{-x}\phi_{i,j}^n, 0)^2\Big)$$
$$+ \max\Big(\max(D^{+y}\phi_{i,j}^n, 0)^2, -\min(D^{-y}\phi_{i,j}^n, 0)^2\Big)\Big)^{1/2}.$$

4.3 Filters

Discrete filters are often used in image segmentation, e.g., in machine learning approaches. A digital filter can be represented as an operator

$$L : I \in \mathcal{I} \longrightarrow \tilde{I} \in \mathcal{I}, \quad \tilde{I}_{ij} = L[I; W_{ij}],$$

where $W_{ij} \subset \Omega_{n_x, n_y}$. A popular discrete filter in image segmentation is the *convolution* filter, defined by

$$\tilde{I}_{i,j} = L_{a,b}[I; W_{i,j}] = \sum_{s=-a}^{a} \sum_{t=-b}^{b} h_{s,t} I_{i-s, j-t}, \tag{10}$$

with a and b positive integers such that $a \leq \frac{n_x-1}{2}$ and $b \leq \frac{n_y-1}{2}$, $W_{i,j} = \{(s,t) : s = -a, \ldots, a, \ t = -b, \ldots, b\}$, and $h_{s,t} \in \mathbb{R}$. The matrix $H = (H_{i,j}) = (h_{-a+i, -b+j}) \in \mathbb{R}^{(2a+1)\times(2b+1)}$ is called kernel matrix and depends on the features we want to extract from the image. Common choices of a and b are $a = b = 3$ and $a = b = 5$.

Edge-detection kernels are frequently used in image segmentation, especially in CNNs. For example, the first layer of a CNN is often responsible for capturing low-level features such as edges, color, and gradient orientation. In general, the choice of H determines the type of features to be extracted. The kernel matrix

$$H = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix}$$

is a vertical edge-detection kernel [57]. Another example is the Sobel operator, used to create an image emphasizing the edges [58]. It allows us to obtain either the gradient amplitude or the gradient direction of the image intensity at each point, by convolving the image with the kernel matrices

$$H_S^x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}, \quad H_S^y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}.$$

The gradient magnitude, G, and the angle of orientation of the edges, θ, are given by

$$|G_{i,j}| = \sqrt{(H_S^x * I)_{i,j}^2 + (H_S^y * I)_{i,j}^2}, \quad \theta_{i,j} = \arctan((H_S^y * I)_{i,j}/(H_S^x * I)_{i,j}).$$

A padding process is commonly used to preserve the dimension of the image after the convolution. It usually consists in the replication or reflection of the pixel values at the image border, or in adding an average gray or even zeros symmetrically around the border of the image. A pooling layer is usually inserted between two successive convolution layers, which is obtained by applying basic functions, such as max and mean, in a small window.

5 Conclusion

We presented a view of image segmentation, focusing on simple computational models and attempting to put them into a coherent framework where the inclusion of a-priori information about the solution is obtained by using penalty terms. We first introduced image segmentation and then outlined basic edge-based, region-based, statistical and machine learning models. We also sketched some numerical methods that can be employed to compute solutions to the models. We believe that our view of models and methods for image segmentation, although very far from being exhaustive, can help the readers understand much modern and sophisticated segmentation techniques, as well as select computational tools for their problems.

Declarations

References

1. Brice, C.R., Fennema, C.L.: Scene analysis using regions. Artif. Intell. **1**(3), 205–226 (1970). https://doi.org/10.1016/0004-3702(70)90008-1

2. Khalid, H., Hussain, M., Al Ghamdi, M.A., Khalid, T., Khalid, K., Khan, M.A., Fatima, K., Masood, K., Almotiri, S.H., Farooq, M.S., et al.: A comparative systematic literature review on knee bone reports from MRI, X-rays and CT scans using deep learning and machine learning methodologies. Diagnostics **10**(8), 518 (2020). https://doi.org/10.3390/diagnostics10080518

3. Bui, K., Fauman, J., Kes, D., Torres Mandiola, L., Ciomaga, A., Salazar, R., Bertozzi, A.L., Gilles, J., Goronzy, D.P., Guttentag, A.I., Weiss, P.S.: Segmentation of scanning tunneling microscopy images using variational methods and empirical wavelets. Pattern Anal. Appl. **23**(2), 625–651 (2020). https://doi.org/10.1007/s10044-019-00824-0

4. Hossain, M.D., Chen, D.: Segmentation for object-based image analysis (OBIA): A review of algorithms and challenges from remote sensing perspective. ISPRS J. Photogramm. Remote. Sens. **150**, 115–134 (2019). https://doi.org/10.1016/j.isprsjprs.2019.02.009

5. Eskenazi, S., Gomez-Kramer, P., Ogier, J.-M.: A comprehensive survey of mostly textual document segmentation algorithms since 2008. Pattern Recogn. **64**, 1–14 (2017). https://doi.org/10.1016/j.patcog.2016.10.023

6. Pal, N.R., Pal, S.K.: A review on image segmentation techniques. Pattern Recogn. **26**(9), 1277–1294 (1993). https://doi.org/10.1016/0031-3203(93)90135-J

7. Antonelli, L., De Simone, V., di Serafino, D.: Spatially adaptive regularization in image segmentation. Algorithms **13**, 226 (2020). https://doi.org/10.3390/a13090226

8. Lucas, A., Iliadis, M., Molina, R., Katsaggelos, A.K.: Using deep neural networks for inverse problems in imaging: Beyond analytical methods. IEEE Signal Process. Mag. **35**(1), 20–36 (2018). https://doi.org/10.1109/MSP.2017.2760358

9. Antonelli, L., De Simone, V., Viola, M.: Cartoon-texture evolution for two-region image segmentation. Comput Optim Appl (2022). https://doi.org/10.1007/s10589-022-00387-7

10. Yousefirizi, F., Rahmim, A.: Consolidating deep learning framework with active contour model for improved PET-CT segmentation. J. Nucl. Med. 62(supplement 1), 1415–1415 (2021). https://jnm.snmjournals.org/content

11. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. Int. J. Comput. Vision **1**, 321–331 (1991). https://doi.org/10.1007/BF00133570

12. Alvarez, L., Morel, J.M.: Formalization and computational aspects of image analysis. Acta Numer **3**, 1–59 (1994). https://doi.org/10.1017/S0962492900002415

13. Osher, S., Sethian, J.A.: Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulation. J. Comput. Phys. **79**, 12–49 (1988). https://doi.org/10.1016/0021-9991(88)90002-2

14. Revol-Muller, C., Grenier, T., Rose, J.-L., Pacureanu, A., Peyrin, F., Odet, C.: Region growing: when simplicity meets theory – region growing revisited in feature space and variational framework. In: Csurka, G., Kraus, M., Laramee, R.S., Richard, P., Braz, J. (eds.) Computer Vision, Imaging and Computer Graphics. Theory and Application, pp. 426–444. Springer, Berlin, Heidelberg (2013)

15. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Commun. Pure Appl. Math. **42**(5), 577–685 (1989). https://doi.org/10.1002/cpa.3160420503

16. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) **6**(6), 721–741 (1984). https://doi.org/10.1109/TPAMI.1984.4767596

17. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Process. **10**(2), 266–277 (2001). https://doi.org/10.1109/83.902291

18. Chan, T.F., Esedo?lu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. **66**(5), 1632–1648 (2006). https://doi.org/10.1137/040615286

19. Calvetti, D., Somersalo, E.: Inverse problems: from regularization to Bayesian inference. Wiley Interdisciplinary Reviews. Computational Statistics (WIREs) **10**(3), 1427–19 (2018). https://doi.org/10.1002/wics.1427

20. Kato, Z., Pong, T.C.: A Markov random field image segmentation model for color textured images. Image Vis. Comput. **24**(10), 1103–1114 (2006). https://doi.org/10.1016/j.imavis.2006.03.005

21. Doersch, C., Gupta, A., Efros, A.A.: Unsupervised visual representation learning by context prediction. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1422–1430 (2015). https://doi.org/10.1109/ICCV.2015.167

22. Furat, O., Wang, M., Neumann, M., Petrich, L., Weber, M., Krill, C.E., Schmidt, V.: Machine learning techniques for the segmentation of tomographic image data of functional materials. Front. Mater. **6**, 145 (2019). https://doi.org/10.3389/fmats.2019.00145

23. Minaee, S., Boykov, Y., Porikli, F., Plaza, A.J., Kehtarnavaz, N., Terzopoulos, D.: Image segmentation using deep learning: A survey. IEEE Trans. Pattern Anal. Mach. Intell. **44**(7), 3523–3542 (2022). https://doi.org/10.1109/TPAMI.2021.3059968

24. Haque, I.R.I., Neubert, J.: Deep learning approaches to biomedical image segmentation. Inform. Med. Unlocked **18**, 100297 (2020). https://doi.org/10.1016/j.imu.2020.100297

25. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. arXiv e-prints (2015) arXiv:1511.00561

26. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015, pp. 234–241. Springer, Cham (2015)

27. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015). https://doi.org/10.1109/CVPR.2015.7298965

28. Fountoulakis, K., Gondzio, J.: Performance of first- and second-order methods for ℓ_1-regularized least squares problems. Comput. Optim. Appl. **65**(3), 605–635 (2016). https://doi.org/10.1007/s10589-016-9853-x

29. Antonelli, L., De Simone, V.: Comparison of minimization methods for nonsmooth image segmentation. Commun. Appl. Ind. Math **9**, 68–96 (2018). https://doi.org/10.1515/caim-2018-0005

30. Weiss, P., Blanc-Féraud, L., Aubert, G.: Efficient schemes for total variation minimization under constraints in image processing. SIAM J. Sci. Comput. **31**(3), 2047–2080 (2009). https://doi.org/10.1137/070696143

31. Birgin, E.G., Martínez, J.M., Raydan, M.: Nonmonotone spectral projected gradient methods on convex sets. SIAM J. Optim. **10**(4), 1196–1211 (2000). https://doi.org/10.1137/S1052623497330963

32. Bonettini, S., Zanella, R., Zanni, L.: A scaled gradient projection method for constrained image deblurring. Inverse Prob. **25**(1), 015002 (2009). https://doi.org/10.1088/0266-5611/25/1/015002

33. Antonelli, L., De Simone, V., di Serafino, D.: On the application of the spectral projected gradient method in image segmentation. J. Math. Imaging Vis. **54**, 106–116 (2016). https://doi.org/10.1007/s10851-015-0591-y

34. di Serafino, D., Ruggiero, V., Toraldo, G., Zanni, L.: On the steplength selection in gradient methods for unconstrained optimization. Appl. Math. Comput. **318**, 176–195 (2018). https://doi.org/10.1016/j.amc.2017.07.037

35. di Serafino, D., Landi, G., Viola, M.: ACQUIRE: an inexact iteratively reweighted norm approach for TV-based Poisson image restoration. Appl. Math. Comput. **364**, 124678 (2020). https://doi.org/10.1016/j.amc.2019.124678

36. Figueiredo, M., Nowak, R., Wright, S.: Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. IEEE J. Sel. Top. Signal Processing **1**(4), 586–598 (2007). https://doi.org/10.1109/JSTSP.2007.910281

37. Fountoulakis, K., Gondzio, J., Zhlobich, P.: Matrix-free interior point method for compressed sensing problems. Math. Program. Comput. **6**(1), 1–31 (2014). https://doi.org/10.1007/s12532-013-0063-6

38. De Simone, V., di Serafino, D., Gondzio, J., Pougkakiotis, S., Viola, M.: Sparse approximations with interior point methods. To appear on SIAM Review (2022), accepted version available arXiv:2102.13608

39. Parikh, N., Boyd, S.: Proximal algorithms. Found. Tr. Optim. **1**(3), 123–231 (2014). https://doi.org/10.1561/2400000003

40. Bonettini, S., Loris, I., Porta, F., Prato, M.: Variable metric inexact line-search-based methods for nonsmooth optimization. SIAM J. Optim. **26**(2), 891–921 (2016). https://doi.org/10.1137/15M1019325

41. Sawatzky, A., Brune, C., Wübbeling, F., Kösters, T., Schäfers, K., Burger, M.: Accurate EM-TV algorithm in PET with low SNR. In: 2008 IEEE Nuclear Science Symposium Conference Record (2008). https://doi.org/10.1109/NSSMIC.2008.4774392

42. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Tr. Mac. Learn. **3**(1), 1–122 (2011). https://doi.org/10.1561/2200000016

43. Figueiredo, M.A.T., Bioucas-Dias, J.: Restoration of poissonian images using alternating direction optimization. IEEE Trans. Image Process. **19**(12), 3133–3145 (2010). https://doi.org/10.1109/TIP.2010.2053941

44. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split Bregman method: segmentation and surface reconstruction. J. Sci. Comput. **45**(1–3), 272–293 (2010). https://doi.org/10.1007/s10915-009-9331-z

45. Setzer, S.: Operator splittings, Bregman methods and frame shrinkage in image processing. Int. J. Comput. Vision **92**(3), 265–280 (2011). https://doi.org/10.1007/s11263-010-0357-3

46. De Simone, V., di Serafino, D., Viola, M.: A subspace-accelerated split Bregman method for sparse data recovery with joint ℓ_1-type regularizers. Electron. Trans. Numer. Anal. **53**, 406–425 (2020). https://doi.org/10.1553/etna_vol53s406

47. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vision **40**(1), 120–145 (2011). https://doi.org/10.1007/s10851-010-0251-1

48. Malitsky, Y., Pock, T.: A first-order primal-dual algorithm with linesearch. SIAM J. Optim. **28**(1), 411–432 (2018). https://doi.org/10.1137/16M1092015

49. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc. **39**(1), 1–38 (1977). https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

50. Jing, Y., Guanci, Y.: Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer. Algorithms **11**(3), 28 (2018). https://doi.org/10.3390/a11030028

51. Marin, D., Tang, M., Ayed, I.B., Boykov, Y.: Beyond gradient descent for regularized segmentation losses. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 10187–10196 (2019)

52. Perona, P., Malik, J.: Scale space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. **12**, 629–639 (1990). https://doi.org/10.1109/34.56205

53. Witkin, A.P.: Scale-space filtering. In: International Joint Conference on Artificial Intelligence, pp. 1019–1022 (1983)

54. Koenderink, J.: The structure of images. Biol. Cybern. **50**, 363–370 (1984). https://doi.org/10.1007/BF00336961

55. Alvarez, L., Guichard, F., Lions, P.L., Morel, J.M.: Axioms and fundamental equations of image processing. Arch. Ration. Mech. Anal. **123**, 199–257 (1993). https://doi.org/10.1007/BF00375127

56. Sethian, J.A.: Level Set Methods and Fast Marching Methods, 2nd edn. Cambridge University Press, UK (1999)

57. Baum, K.G.: Signal Filtering: Noise Reduction and Detail Enhancement, pp. 325–343. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-540-79567-4_27

58. Kanopoulos, N., Vasanthavada, N., Baker, R.L.: Design of an image edge detection filter using the Sobel operator. IEEE J. Solid-State Circuits **23**(2), 358–367 (1988). https://doi.org/10.1109/4.996