Chapter 8
Scene Analysis Using Morphological Mathematics and Fuzzy Logic

Victoria Lynn Fox, Mariofanna Milanova and Salim Al-Ali

Abstract Owing to compound textural features, intensity inhomogeneity, image layers, and variations of statistics inherent, the segmenting of complicated images into areas of similarity for scene analysis is a challenging task. In this work, a morphological active contour is developed to increase efficiency of current active contour schemes and a fuzzy clustering energy is incorporated into the active contour algorithm to increase accuracy and flexibility. Finally, to aid in the segmentation of figures for scene analysis, a visual attention is incorporated into the fuzzy clustering. The savings in computational efficiency garnered from using a morphological curve evolution rather than a partial differential equation and corresponding Euler-Lagrange equations combined with the expert knowledge garnered from a visual attention fuzzy logic scheme translates into a highly accurate and efficient segmentation method for scene analysis.

Keywords Segmentation · Morphological active contour · Fuzzy energy · Natural images · Visual attention

V.L. Fox (✉)
Department of Applied Science, University of Arkansas at Little Rock,
2801 S. University Avenue Little Rock, Little Rock, AR 72204, USA
e-mail: vlfox@ualr.edu

M. Milanova · S. Al-Ali
Department of Computer Science, University of Arkansas at Little Rock,
2801 S. University Avenue Little Rock, Little Rock, AR 72204, USA
e-mail: mgmilanova@ualr.edu

S. Al-Ali
e-mail: sgsaeed@ualr.edu

M.N. Favorskaya and L.C. Jain (eds.), *Computer Vision in Control Systems-1*,
Intelligent Systems Reference Library 73, DOI 10.1007/978-3-319-10653-3_8

8.1 Introduction

While the statistics of large data sets of images follow certain regularities, statistics of singular images are found to be capable of a large variance in statistical analysis [1]. Therefore, the applications, which work with individual images, must seek to exploit the variability of the image while acknowledging the established statistical properties of natural images as a group. As an example, consider the spatial structure of a natural image, which is often irregular with contours produced by different boundaries, markings, and shadows. These boundaries can sometimes be determined by exploiting luminance and contrast within the given natural scene. In a survey of natural image statistics, Geisler [2] notes, in general, natural images, i.e. images with significant local covariance, tend to have large variations in local luminance and contrast with a low correlation in their average joint distribution. However, global statistics have shown that strong features tend to cluster in natural images and, thus, ignoring global information in light of only local information often results in poor segmentation results, when considering natural images [3]. Therefore, the segmentation protocols seeking to exploit luminance and contrast in a natural image must consider both local and global statistics, which leads to an increase in computational cost in segmentation algorithms. It bears mentioning that the statistics involving luminance and contrast are of only one group of statistical models for natural images. To effectively segment natural images, one must also consider other statistical information, such as textural information given by entropy and homogeneity, depth given by the scene scale of the image or levels of color saturation in a multispectral image. With each additional feature space, the complexity of a given algorithm grows. Therefore, it is very important for the mechanism of an image segmentation method to be as efficient as possible with low computational cost while minimizing error.

While the last two decades have seen a large variety of image segmentation methods with many able to produce reasonable segmentations on images with moderate complexity, see [1, 2, 4] and their references as examples of effective segmentation techniques, computational efficiency is still a concern, when segmenting a complex image. Many state-of-the-art methods can become impractically time-consuming or are limited in the types or size of images that can be processed. Often, feature vectors are sacrificed in order to increase computational speed resulting in a lower level of accuracy in general for the algorithm [3]. The proposed method presented in this work is a computational low-cost segmentation method that effectively segments a variety of complex images. This method makes use of mathematical morphology, fuzzy logic clustering, and visual attention, all of which are incorporated into a hybrid, level set active contour method.

The chapter is organized as the follows. Background material is presented in Sect. 8.2. The proposed method is developed in Sect. 8.3. Conclusion is situated in Sect. 8.4.

8.2 Background Material

Let us consider briefly the background materials including segmentation methods (Sect. 8.2.1), the basics of morphological mathematics (Sect. 8.2.2), application of fuzzy logic in imaging (Sect. 8.2.3), and visual attention issues (Sect. 8.2.4).

8.2.1 Segmentation Methods

The objective of segmentation is to partition an image into regions. With the assumption that every section in an image is sufficiently homogenous, edge-based segmentation determines the transition between two sections on the basis of discontinuities alone. When this assumption is not valid, a region-based segmentation usually provides a more realistic segmentation product.

In order to detect meaningful discontinuities between sections, most edge-detection techniques employ the use of first- and second-derivatives. The first-order derivative of choice is the gradient vector of an image $I(x, y)$ given as Eq. 8.1, which is obtained by the partial derivatives at every pixel location.

$$\nabla I = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \partial I/\partial x \\ \partial I/\partial y \end{bmatrix} : \Omega \to \Re^2 \tag{8.1}$$

To determine the presence of edges, the magnitude of the gradient vector is computed with Eq. 8.2.

$$|\nabla I| = \sqrt{g_x^2 + g_y^2} = \sqrt{(\partial I/\partial x)^2 + (\partial I/\partial y)^2} : \Omega \to \Re \tag{8.2}$$

Since it is zero in areas of constant intensity and its values are related to the degree of intensity change in areas of variable intensity. The Laplacian of an image function $I(x, y)$ is the sum of the second-order derivatives, defined by Eq. 8.3.

$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{dy^2} : \Omega \to \Re \tag{8.3}$$

While the Laplacian is seldom used by itself for edge detection due to its sensitivity to noise, it is powerful, when used in combination with other edge-detection techniques. The edge detection by gradient operations generally performs well only in images with defined intensity transitions and relatively low noise. However, computationally, the gradient operator methods have a relatively lower cost than other segmentation methods because the computation can be performed with a local filtering operation [5].

While the edge-based segmentation focuses on discontinuities in intensity levels, the region-based techniques find the regions directly. The basic formulation of region-based segmentation with R representing a region in the image can be given with a series of conditions. The first condition states that every pixel must be assigned a region while the second condition requires that points in a region be connected (e.g., 4- or 8-connected) and the third condition indicates the regions must be disjoint. The fourth condition states that pixels in a segmented region must share some predefined common features while the fifth condition indicates that adjacent regions are different in the sense of the predefined feature spaces [6].

The region growing is a technique that merges regions of interest into a larger region of interest. The pixel aggregation is an example of a region growing technique. In pixel aggregation, an initial set of seed points grows regions from the seeds by joining neighboring pixels, if they satisfy given criteria. In its most basic form, segmentation starts with two initial seeds, and then the region grows, if neighboring pixels satisfy the following criteria: $|I(x, y) - I(seed)| < \tau$, in which τ is some predefined threshold. The selection of initial seeds is often based on the nature of applications or images. If a priori information is not obtainable, then the clustering techniques must be used to determine the pixels that can be used as seeds. Despite the simple nature of the algorithm, there are several problematic areas in the implementation of region growing: descriptors of region properties alone can yield misleading results, if connectivity is not taken into consideration.

The statistical estimation is another common approach in a region-based segmentation [7]. When considering the statistical segmentation of images, authors generally suppose the existence of two random fields: the field of "classes" and the field of "measurements." With this method, two sections are considered to be homogenous and accordingly merged, if they have common parameter values within a given threshold. In application, the parameters of a section cannot be directly observed; rather they can only be inferred from the observed data. This inference, if often made using Bayes's rule and the conditional probability density function $p(\boldsymbol{I}(x,y)|\boldsymbol{\theta}_m)$, which presets the conditional probability statistic derived from the data $(\boldsymbol{I}(x,y))$, will be observed, given that section m has the parameter values of $\boldsymbol{\theta}_m$. In typical statistical region merging algorithms [8], stochastic estimates in the parameter space are obtained for different sections, and merging decisions are based on the similarity of these parameters.

Unfortunately, there is a limitation of most estimation-based segmentation methods in that they do not explicitly represent the uncertainty in the estimated parameter values and, therefore, are prone to error, when the parameter estimates are poor. To counteract this limitation, a Bayesian probability of homogeneity uses all of the information contained in the statistical image model rather than just estimating parameter values. The probability of homogeneity is based on the ability to formulate a prior probability density function on the parameter space, and measures the uniformity by taking the expectation of the data likelihood over a posterior parameter space.

In general, the region-based methods yield more reasonable segmentations than edge-based algorithms, when an image has relatively large noise and/or requires the use of local properties combined with global properties. However, the complexity and computational cost of region-based methods can be large, particularly when considering methods based upon partial differential equations, e.g. active contour methods. Active contours are the energy-based segmentation methods that seek to guide partitioning of an image via the minimization of a cost functional.

The first efforts in formulating the boundary detection problem as an energy minimization problem resulted in the energy-minimizing splines guided by external constraint forces that pull the splines towards objects of interest [9]. These splines were dubbed snakes by their creators, Kass, Witkin, and Terzopoulos, because the contours appeared to slither across an image as it moved toward local minima. In the classical formulation, the boundary detection consisted of matching a deformable model to an image by means of energy minimization. Representing the position of a snake parametrically by $C(s) = (x(s), y(s): 0 \le s \le L): \Re \rightarrow \Omega$, where L denotes the length of the contour C, s is the arc length, and Ω represents the entire domain of an image $I(x, y)$, the energy functional can be written as Eq. 8.4, where E_{int} and E_{ext} represent the internal energy and external energy functions.

$$E(C) = E_{int} + E_{ext} \tag{8.4}$$

The internal energy function determines the smooth shape (regularity) of the contour. A common choice for the internal energy is a functional given by Eq. 8.5, where $C(s) \approx C(n) = \{(x(n), y(n)): 0 \le n \le N, s = 0 + n\Delta s\}$ and $L = N\Delta s$.

$$E_{int} = \int_0^L \alpha|C'(s)|^2 + \beta|C''(s)|^2 ds \approx \sum_0^N \alpha|C'(n)|^2 + \beta|C''(n)|^2 \Delta s \tag{8.5}$$

Here α controls the tension of the contour and β controls the rigidity of the contour. The external energy term determines the criteria of contour evolution depending on the image $I(x, y)$ and can be defined as Eq. 8.6, where $E_{img}(x, y)$ denotes a scalar function defined on the image plane, so the local minimum of E_{img} attracts the snakes to edges.

$$E_{ext} = \int_0^L E_{img}(C(s))ds \approx \sum_{n=0}^N E_{img}(C(n))\Delta s \tag{8.6}$$

The edge attraction function, E_{img}, was originally presented as a combination of three separate functionals: $E_{img} = w_{line}E_{line} + w_{edge}E_{edge} + w_{term}E_{term}$. Since the presentation of the original algorithm, there have been many other functionals suggested for E_{img} and one common example is a function of image gradient, given

by Eq. 8.7, which incorporates a Gaussian smoothing filter, G_σ, with standard deviation σ and a suitably chosen constant λ.

$$E_{img}(x,y) = \frac{1}{\lambda|\nabla G_\sigma * I(x,y)|} : \Omega \rightarrow \Re \tag{8.7}$$

In order to solve the geometric contour problem, one must find the contour C that minimizes the total energy term E within the given set of weights, α, β, and λ.

In order to achieve an accurate location of edges, the classical snake algorithm must be initialized sufficiently near the edge or object of interest. Estimating a correct position of an initial contour without prior knowledge is a challenging problem. Also, classical geometric contours are only able to separate a region into two sub-regions and cannot, subsequently split into multiple boundaries or merge from multiple initial contours. In order to correct this deficiency, Hamilton-Jacobi formulations [5] were applied to active contours and resulted in the creation of geodesic active contours.

The level set function $\phi(x,y)$ was proposed by Oshar and Sethian [5] as a formulation to implement active contours. Oshar and Sethian represented a contour implicitly via a two-dimensional Lipschitz-continuous function $\phi(x,y): \Omega \rightarrow \Re$ defined on the image plane. On a particular level, usually the zero level, the level set function is defined as a contour, such as Eq. 8.8, where Ω denotes the entire image plane.

$$\mathcal{C} = \{(x,y): \phi(x,y) = 0\}, \forall (x,y) \in \Omega \tag{8.8}$$

As the level set function increases from the initial stage, the corresponding set of contours, $\mathcal{C}$, moves toward the exterior.

By using the zero level, the contour can be defined as the border between a positive area and negative area. Thus, the contour can be identified by checking the sign of $\phi(x,y)$. Using the zero level, the level set is usually represented by Eq. 8.9.

$$\phi(x,y) = \begin{cases} < 0(x,y) \text{ inside } C \\ = 0(x,y) \text{ on } C \\ > 0(x,y) \text{ outside } C \end{cases} \tag{8.9}$$

The initial level set function $\phi_0(x,y): \Omega \rightarrow \Re$ is usually given as a signed distance from the initial contour such as in Eq. 8.10 in a way that $\pm D(a, b)$ denotes a signed distance between a and b and $N_{x,y}(\mathcal{C}_0)$ denotes the nearest neighbor pixel on the initial contours $\mathcal{C} = \mathcal{C}(t = 0)$ from (x, y).

$$\phi_0(x,y) = \{\phi(x,y): t = 0\} = \pm D\big((x,y), N_{x,y}(C_0)\big), \forall (x,y) \in \Omega \tag{8.10}$$

The deformation of the contour is generally represented as a Partial Differential Equation (PDE). The initial proposal for a formulation of contour evolution using the magnitude of the gradient, given by Osher and Sethian [5], states by Eq. 8.11,

where v denotes a constant speed term to push or pull the contour and $\kappa(\cdot) : \Omega \rightarrow \Re$ denotes the mean curvature of the level set function $\phi(x, y)$ given by Eq. 8.12.

$$\frac{\partial \phi(x,y)}{\partial t} = |\nabla \phi(x,y)|(v + \varepsilon\kappa(\phi(x,y))) \tag{8.11}$$

$$\kappa(\phi(x,y)) = \operatorname{div}\left(\frac{\nabla\phi}{\|\nabla\phi\|}\right) = \frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}} \tag{8.12}$$

The curvature term is used to control the regularity of the contour as the internal energy term does in the classic snake model while ε controls the balance between the smoothness and the robustness of the evolution.

Chan and Vese [10] proposed a new form of contour evolution that is very popular in current research methods, the active contour without the edges method. The length of the contour $|\mathcal{C}|$ can be approximated by a function of $\phi(x, y)$ such as in Eq. 8.13, where $H_\varepsilon(\cdot)$ denotes the regularized form of the unit step function, $H(\cdot): \Omega \rightarrow \Re$ given by Eq. 8.14 and $\delta_\varepsilon(\cdot)$ denotes the derivative of $H_\varepsilon(\cdot)$.

$$|\mathcal{C}| \approx L_\varepsilon(\phi(x,y)) = \int |\nabla H_\varepsilon(\phi(x,y))| dxdy = \int \delta_\varepsilon(\phi(x,y))|\nabla\phi(x,y)| dxdy \tag{8.13}$$

$$H(x,y) = \begin{cases} 1 \text{ if } \phi(x,y) \geq 0 \\ 0 \text{ if } \phi(x,y) < 0 \end{cases} \quad \forall (x,y) \in \Omega \tag{8.14}$$

Since $H_\varepsilon(\cdot)$ produces either a 0 or 1 depending on the sign of the input, $\delta_\varepsilon(\cdot)$ produces nonzero results only on the contour of Eq. 8.13, where $\phi(x, y) = 0$. The associated Euler-Lagrange equation [11] obtained by minimizing $L_\varepsilon(\cdot)$ with respect to ϕ and parameterizing the descent directions by an artificial time t is given by Eq. 8.15.

$$\frac{\partial \phi(x,y)}{\partial t} = \delta_\varepsilon(\phi(x,y))\kappa(\phi(x,y)) \tag{8.15}$$

The contour evolution motivated by this equation can be interpreted as the motion by mean curvature minimizing the length of the contour. Therefore, Eq. 8.12 is considered as the motion motivation by partial differential equation, while Eq. 8.15 is considered as the motion motivated by energy minimization.

A convenient characteristic of level-set contours is that the contour can split or merge as the topology of the level set function changes. As a result, level set methods can detect more than one boundary simultaneously and several initial contours can be placed. The computational cost of level set methods, however, is high because computation should be done on the same dimension as the image plane. Yet, due to the convenience and flexibility of level set methods, they are a practical method to use in the research of segmenting multi-region images.

8.2.2 Morphological Mathematics

First introduced by Matheron [12] and Serra [13], the mathematical morphology views an image as a set of geometric structures then transforms it with the use of a smaller geometrically defined set, commonly referred to as a structuring element. The structuring element is translated over the image set and with the use of basic set operations (i.e. union and intersection), the fundamental operations of dilation and erosion are obtained.

For a binary image, entries consisting of 0 represent background information and entries consisting of 1 represent foreground entries. In a binary erosion, $A \ominus B$, in which A is eroded by the structuring element B, consists of all points, for which the translation of B over A fits inside of A. In other words, it is the set operation $A \ominus B = \{x | B_x \subseteq A\}$. A binary dilation is the dual operation to erosion and is defined by the set complementation of erosion. The dilation of a set A by structuring element B is given by $A \oplus B = [A^c \ominus (-B)]^c$. To dilate A by B, B is rotated around the origin to create $(-B)$, the complement of A is eroded by $(-B)$, and then the complement of the erosion is taken. To illustrate with a sample binary image matrix, let the binary image be represented by A and the structuring element is represented by B as given by the matrix in Eq. 8.16. The erosion of A by B is given by the matrix in Eq. 8.17 while the dilation of A by B is given by the matrix in Eq. 8.18.

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \tag{8.16}$$

$$A \ominus B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{8.17}$$

$$A \oplus B = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \end{bmatrix} \tag{8.18}$$

From erosion and dilation, the morphological operations of opening and closing can be defined. A morphological opening on a binary image is defined as $A \circledcirc B = (A \ominus B) \oplus B$, where the image A is eroded by B, and then the result of the erosion is dilated by B. Using the sets defined in Eq. 8.16, $A \circledcirc B$ has the results represented in Fig. 8.1.

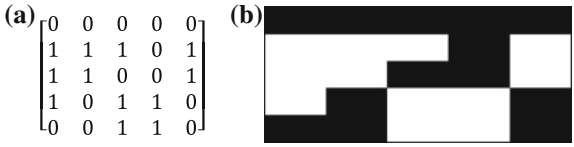

Fig. 8.1 Example of morphological opening on a binary set. **a** A matrix form of $A \circledcirc B$. **b** An image representation of $A \circledcirc B$

The morphological closing on a binary image is defined as $A \odot B = (A \oplus B) \ominus B$. Using the sets defined in Eq. 8.16, $A \odot B$ results in the matrix and image representation in Fig. 8.2.

For a binary structuring element B, the locations, where B is equal to zero, are referred to as neutral elements since they do not affect the image during the morphological operation. For gray-scale morphology, the morphological operations transform the gray-scale image into a binary data set with an extra dimension representing the gray-level. Since the gray-scale level of an image is bound to a finite domain of $[0, m]$, the neutral elements of a gray-scale structuring element are the elements with values of m [14]. This is due to the nature of the definition of gray-scale morphology operators, in which they are used as the invariants to the maximum and minimum operators.

In continuous gray-scale morphology, images are viewed as functions mapping a grid to $\mathbb{R} \cup (-\infty, \infty)$. Since gray-scale images are restricted to integer values between some range of values (i.e. 0–255 for 8 bit), it is necessary to restrict the discrete gray-scale morphology mapping to the integer range imbedded in the image format. With these preliminaries stated, the erosion and dilation of an gray-scale image, $A(x)$, by a structuring element $B(x)$, can be defined as $(A \ominus B)(x) = \underset{y \in B}{inf}[A(y) - B(y - x)]$ and $(A \oplus B)(x) = \underset{y \in B}{sup}[A(y) - B(y - x)]$, respectively.

8.2.3 Fuzzy Logic

Fuzzy logic is composed of multi-logic systems that have been developed in opposition to the classical logic, which uses an "on/off" switch in its assessment of membership. In fuzzy logic sets, the membership is determined by values assigned

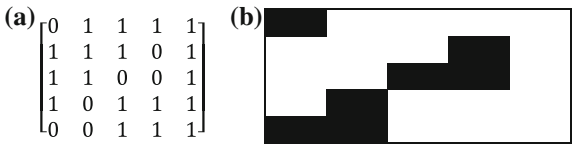

Fig. 8.2 Example of morphological closing on a binary set. **a** A matrix form of $A \odot B$. **b** An image representation of $A \odot B$

to linguistic expressions and human decisions. The use of fuzzy logic in imaging is primarily in the practice of image clarity and identification of objects [15]. It is a modification of this second usage, i.e. identification of objects, where segmentation has begun to use fuzzy logic.

A linguistic variable is a term used in our natural language to describe some concept that usually has vague or ill-defined values. For example, if researcher tried to describe the frequency heights of an image in the frequency domain, the linguistic variable would be "height" and the typical values would be "low", "medium", and "high" and would define our clustering. The fuzzy expert system process is composed of four steps:

1. Fuzzification—convert the data to fuzzy sets via membership functions.
2. Inference—perform all fuzzy logical operations and apply an implication method.
3. Composition—apply an aggregation method for fuzzy sets acquired in the inference step.
4. Defuzzification—convert the final fuzzy conclusion back to raw data to obtain final weights.

It is important to note that fuzzy logic is not logic that is fuzzy; rather it is the logic of fuzziness. While the linguistic variable may be filled with ambiguity, the output of the defuzzification is a value that will guide the image segmentation.

There are several fuzzy membership functions to help cluster the values of the linguistic variable. In choosing the linguistic variables and terms for the fuzzy logic model, it is important to be directed by the following guidelines [16]:

1. The features should carry enough information about the image and should not require any domain-specific knowledge for their extraction.
2. They should be easy to compute in order for the approach to be feasible for a large image collection.
3. They should relate well with the human perceptual characteristics since users will finally determine the suitability of the retrieved images.

8.2.4 Visual Attention

In 1964, Neisser [17] presented a popular model, in which human vision consists of pre-attentive and attentive stages. The pre-attentive stage focuses on local spatial discontinuity, while in the attentive stage, relationships between these discontinuities are created and clustering takes place. In the pre-attentive stage, the principles of proximity, simplest form, and continuity factor into the decision, of where the spatial discontinuities take place. The attention stage additionally takes note of similarity features (e.g. color, luminosity, texture) and shape to aid in the process.

In computational vision, the numerous approaches to the attention model of scene analysis can be categorized as two methodologies: bottom-up and top-down.

In bottom-up attention, dissimilarities attract the attention of the vision model. In the top-down methodology, the viewer searches for a specific feature and the expected objects receive the attention [18]. Through both of these methodologies, a saliency map can be created to aid in the segmentation of the visual scene. In general, bottom-up attention models are based upon feature detection (e.g. orientation, shape, texture) that is easily estimated by a computer while top-down models are more subjective and increase computational complexity in that they depend on contextual clues, objectives, and expert knowledge of the viewer. In an effort to reduce computational cost, the method presented here will make use of the bottom-up saliency model known as Context-Aware Saliency [19, 20].

Context-Aware Saliency Detection (CASD) makes use of the principles of Gestalt vision psychology in its formulation. Particularly, the model makes use of following information:

1. Local, pre-attentive features such as color, texture, and contrast.
2. Global attentiveness, which identify features that deviate from the norm.
3. Perceptual organization rules such as a visual scene containing at least one center of gravity.
4. High-level factors such as distance priors or shape priors.

Using the local-global feature fusion and perceptual organization rules, the CASD detects the salient objects along with the regions of the image around the salient object in order to lend context to the salient region. The local, pre-attentive features give distinctive areas a high saliency and homogenous regions a low saliency score. Frequently occurring features are classified as part of the ground and rare features are classified as part of the figure per global attentiveness. Perceptual organization groups salient pixels that are in close proximity to each other and discounts salient pixels that are not connected. Finally, a center prior contributes to the determination, of which of the salient pixels have the highest fixation levels. Thus, the CASD makes use of the similarity and proximity principles of Gestalt psychology with exceptionally low computational cost. In particular, the method evaluates pixels in patches in order to evaluate the context of each pixel. Considering a single patch p_i of scale s at each pixel, a single pixel i is salient, when the patch containing the pixel is unique with respect to all other patches in the image. This capability of incorporating context patches with the salient object gives the CASD model the flexibility needed for scene analysis.

8.3 Proposed Method

The proposed method is based on morphological operations (Sect. 8.3.1), hybrid morphological contour (Sect. 8.3.2), and representation of fuzzy morphological contour with visual attention (Sect. 8.3.3).

8.3.1 Morphological Mean Curvature

In the image processing application of level set active contours, the curve, $\mathcal{C}: \mathbb{R}^+ \times [0, 1] \rightarrow \mathbb{R}^2$, is represented implicitly as a level set of an embedding function. If we set $u: \mathbb{R}^+ \times \mathbb{R}^2 \rightarrow \mathbb{R}$ as an implicit representation of our contour, it will become $\mathcal{C}(t) = \{(x, y) | u(t, (x, y)) = 0\}$. During the contour evolution, infinitesimal change of the contour is controlled by differential operators. In other words, a differential operator $\mathcal{D}$ guides the contour evolution with the partial differential equation $\mathcal{C}(t) = \mathcal{D}(\mathcal{C})$. In partial differential equation formulations of active contours, the smoothing force regularly takes the form of mean curvature motion and acts as a regularization term.

The underlying principal of mean curvature motion is the evolution of a simple closed curve, whose points move in the direction of the normal with specified velocity. Rewriting $\mathcal{D}(\mathcal{C}) = \mathcal{F} \cdot \mathcal{N}$, where $\mathcal{N}$ is the normal to the contour and $\mathcal{F}$ is a scalar field, one can determine the velocity of evolution at each point on the contour. In level set implementations, the evolution of u(x, y) is $\frac{\partial u}{\partial t} = |\nabla u| \cdot \mathcal{F}$ and will equal Eq. 8.19, when $\mathcal{F}$ is the divergence of the normalized gradient (i.e. Euclidean curvature of $\mathcal{C}$) and gives the curvature of the implicit curve at each point.

$$\frac{\partial \mathrm{u}}{\partial \mathrm{t}} = |\nabla u| div\left(\frac{\nabla u}{|\nabla u|}\right) \tag{8.19}$$

The parameter $u(x, y)$ must be discretized in image processing in order to be applied to the grid of image information, which is usually expressed as pixels in two dimensional applications. The discretization of the differential operator is not always a trivial task and results in one of the losses of efficiency in many image processing contour applications. As a result, the search for a low-cost estimator of mean curvature motion is an area of active research.

One of the more significant contributions to the topic is provided by [21], in which it is proven the two-dimensional mean curvature term can be replaced by the mean of two morphological operators for a single iteration of the method. To morphologically approximate mean curvature, we let $\mathcal{B}$ represent line segments of set length then define the morphological continuous line operators as mentioned in Eqs. 8.20–8.21.

$$(\mathcal{A}_h u)(\boldsymbol{x}) = \underset{B \in \mathcal{B}}{sup} \underset{\boldsymbol{y} \in \boldsymbol{x} + hB}{inf} u(\boldsymbol{y}) \tag{8.20}$$

$$(\mathcal{J}_h u)(\boldsymbol{x}) = \underset{B \in \mathcal{B}}{inf} \underset{\boldsymbol{y} \in \boldsymbol{x} + hB}{sup} u(\boldsymbol{y}) \tag{8.21}$$

Using these operators, let us then define the mean operator as Eq. 8.22, in which the scheme in [21] relates the mean operator with the mean curvature motion by Eq. 8.23.

$$(\mathcal{F}_h u)(\boldsymbol{x}) = \frac{(\mathcal{A}_h u)(\boldsymbol{x}) + (\mathcal{J}_h u)(\boldsymbol{x})}{2} \tag{8.22}$$

$$(\mathcal{F}_h u)(x) = u(x) + \frac{1}{4}h^2|\nabla u| div\left(\frac{\nabla u}{|\nabla u|}\right)(x) + O\left(h^3\right) \tag{8.23}$$

Using a small h and subtracting $u(x)$ from each side of (Eq. 8.23), results in the infinitesimal generator of the $\mathcal{F}_h$ operator will be the following:

$$\lim_{h\to 0^+} h^{-1}\left[\left(\mathcal{F}_{\sqrt{4h}}u\right)(\boldsymbol{x}) - u(\boldsymbol{x})\right] = |\nabla u| div\left(\frac{\nabla u}{|\nabla u|}\right)(\boldsymbol{x}). \tag{8.24}$$

From Eq. 8.24 one can solve the mean curvature motion by means of the $\mathcal{F}_h$ operator. However, since the $\mathcal{F}_h$ operator generates new level set values after a single iteration, it ceases to be morphological. In [22] and [23], Alarez et al. modify the Catte, Dibos, and Koepfler scheme with the use of operator composition, which states that given any two operators $\mathcal{P}_h^1$ and $\mathcal{P}_h^2$, we have, for a small h, Eq. 8.25.

$$\mathcal{P}_{h/2}^2 \circ \mathcal{P}_{h/2}^1 u \approx \frac{\mathcal{P}_h^2 u + \mathcal{P}_h^1 u}{2} \tag{8.25}$$

From this, Alarez et al. show that the non-morphological operator $\mathcal{F}_{\sqrt{4h}}$ can be approximated by the morphological operator represented in Eq. 8.26 with a base of $\mathcal{B}^2$ and is equivalent to Eq. 8.11.

$$\mathcal{A}_{\sqrt{h}} \circ \mathcal{J}_{\sqrt{h}} \approx \frac{\mathcal{A}_{\sqrt{h}} u + \mathcal{J}_{\sqrt{h}} u}{2} \tag{8.26}$$

8.3.2 A Hybrid Morphological Contour

Some authors lay out the format for a hybrid morphological contour [24, 25]. This section is a brief review of the algorithm for the contour extraction. While the combination of an edge based and region based active contour, commonly referred to as a hybrid contour, results in increased computational complexity in order to mitigate the shortcomings of either method alone, the hybrid morphological contour has a low complexity and circumventing a method's shortcoming does not significantly add to the computational cost.

In the hybrid method, the coupling of the strong edge term and region statistics creates a symbiotic relationship. When the edge term is low, the curve is attracted toward the region of interest. However, when the curve is far away from an edge, the region statistics take control of the curve evolution and the contour resists becoming a stationary model. Using the Active Contour without Edges presented as a basis for the active contour (Eq. 8.27) with the region statistics are used by the third and fourth terms.

$$F(c_1, c_2, \mathcal{C}) = \mu(Length\ of\ \mathcal{C}) + p(Area\ in\ \mathcal{C}) + \lambda_1 \int_{in\mathcal{C}} |u - c_1|^2 d\boldsymbol{x} + \lambda_1 \int_{out\,\mathcal{C}} |u - c_2|^2 dx \tag{8.27}$$

Parameters λ_1 and λ_2 weight the importance of the regions inside and outside the curve, respectively, while c_1 and c_2 are the average intensity levels inside and outside the contour. In the hybrid morphological active contour, the third and fourth terms of Eq. 8.27 are incorporated directly into the algorithm.

The first term of the Active Contour without Edges is replaced with the morphological mean curvature evolution described in Eq. 8.25 while the second term becomes the edge-based portion of the hybrid method. In edge based methods, the contour flow is often represented with the formulation given in Eq. 8.28, where $g(I)|\nabla u|v$ is the balloon force, $\nabla g(I)\nabla u$ is the edge attraction force, and $g(I)|\nabla u| div\left(\frac{\nabla u}{|\nabla u|}\right)$ is mean curvature motion.

$$\frac{\partial u}{\partial t} = g(I)|\nabla u|v + \nabla g(I)\nabla u + g(I)|\nabla u| div\left(\frac{\nabla u}{|\nabla u|}\right) \tag{8.28}$$

The parameter $g(I)$ represents an edge image attractor usually obtained from an edge detector, u denotes the contour, and v is an inflation (or deflation) constant. Focusing on the balloon force, $g(I)$ could be obtained from any edge detector appropriate for the image. Traditionally, one would use an edge detector, which is low in the edges of the image such as Eq. 8.29.

$$g(I) = \frac{1}{\sqrt{1 + \alpha|\nabla G_\sigma * I|}} \tag{8.29}$$

In the hybrid morphological active contour method, morphological operations of dilation and erosion are used to approximate the balloon force. The dilation of a function is defined as $(\mathcal{D}_h u)(\boldsymbol{x}) = sup_{\boldsymbol{y}\in hB} u(\boldsymbol{x} - \boldsymbol{y})$ while erosion is defined by $(\mathcal{E}_h u)(\boldsymbol{x}) = inf_{\boldsymbol{y}\in hB} u(\boldsymbol{x} - \boldsymbol{y})$. The radius of the operator is denoted by h and B is a disk structuring element of radius one. The function $u_d: \mathbb{R}^+ \times \mathbb{R}^2 \to \mathbb{R}$, where $u_d(t, \boldsymbol{x}) = \mathcal{D}_t u_o(\boldsymbol{x})$ is the solution to Eq. 8.30 for the initial condition $u_d(0, \boldsymbol{x}) = u_o(\boldsymbol{x})$ [26].

$$\frac{\partial u_d}{\partial t} = |\nabla u_d| \tag{8.30}$$

As a result, $\mathcal{D}_h$ is the infinitesimal generator of Eq. 8.9. Using a comparable rational, we have the function $u_d: \mathbb{R}^+ \times \mathbb{R}^2 \to \mathbb{R}$, where $u_e(t, \boldsymbol{x}) = \mathcal{E}_t u_o(\boldsymbol{x})$ is the solution to Eq. 8.31.

$$\frac{\partial u_e}{\partial t} = -|\nabla u_e| \tag{8.31}$$

Using the morphological operators $\mathcal{D}_h$ and $\mathcal{E}_h$, one can now solve level set evolution PDEs. In the balloon force term, $g(I)$ manages the balloon force in individual sections of the curve. The smaller $g(I)$ becomes, the closer the curve is to the edge. With the use of a threshold, factor $g(I)$ can be discretized into the morphological formulation. The product $|\nabla u|v$ leads to the PDES in Eqs. 8.30 and 8.31. If v is positive, the PDE becomes the dilation PDE. Likewise, if v is negative, then the erosion PDE is used.

In the Active Contour without Edges, the internal and external forces are combined through addition of the terms. Our hybrid morphological active contour combines them by iteratively interchanging their discretized formulations. In every iteration, first the balloon force with the edge attraction energy will be applied, then the region force is applied, and at last the mean curvature motion over the embedded level set function u is computed. Given the contour evolution at iteration, $u^n : \mathbb{R}^2 \rightarrow \{0, 1\}$, u^{n+1} is defined using the steps of the algorithm mentioned below (Eq. 8.32).

Algorithm (1)

Step 1

$$u^{n+1}_{balloon}(\boldsymbol{x_i}) = \begin{cases} (\mathcal{D}_d u)(\boldsymbol{x}_i) & \text{if } g(I)(\boldsymbol{x}_i) > 0 \text{ and } v > 0 \\ (\mathcal{E}_d u)(\boldsymbol{x}_i) & \text{if } g(I)(\boldsymbol{x}_i) > 0 \text{ and } v < 0 \\ u^n_{balloon} & \text{otherwise.} \end{cases}$$

Step 2

$$u^{n+1}_{region} = \begin{cases} 1 & \text{if } \left|\nabla u^{n+1}_{balloon}\right|[\left(\lambda_1(I-c_1)^2 - \lambda_2(I-c_2)^2\right](\boldsymbol{x_i}) < 0 \\ 0 & \text{if } \left|\nabla u^{n+1}_{balloon}\right|[\left(\lambda_1(I-c_1)^2 - \lambda_2(I-c_2)^2\right](\boldsymbol{x_i}) > 0 \\ u^{n+1}_{ballon}(\boldsymbol{x}_i) & \text{otherwise} \end{cases} \tag{8.32}$$

$$c_1 = \frac{\int_\Omega I * H(u) d\boldsymbol{x}}{\int_{\text{inside } \mathcal{C}} d\boldsymbol{x}} \text{ and } c_2 = \frac{\int_\Omega I * (1 - H(u)) d\boldsymbol{x}}{\int_{\text{inside } \mathcal{C}} d\boldsymbol{x}}$$

Step 3

$$u^{n+1} = \begin{cases} (\mathcal{A}_d \circ \mathcal{J}_d u^{n+1}_{region})(\boldsymbol{x}_i) \text{ if } g(I)(\boldsymbol{x}) > 0 \\ u^{n+1}_{region}(\boldsymbol{x}_i) \quad \text{otherwise} \end{cases}$$

The experimental results presented in [24] and [25] clearly show the efficiency and robustness of the hybrid morphological active contour.

8.3.3 A Fuzzy Morphological Contour with Visual Attention

A fuzzy morphological approach for contour with visual attention supposes the analysis of fuzzy energy, building of visual attention model, and creation of combined algorithm.

In [27], Krindis and Chatzis introduced a new type of energy to drive active contours during the segmentation process. This energy, referred to as fuzzy energy, was derived from using a fuzzy logic clustering method and then employs the membership values and weights into the active contour formulation. Specifically, the following functionals are incorporated into a regional active contour model provided by Eq. 8.33, where Ω is the image domain and C is an evolving curve such that $C \subset \Omega$.

$$F_1(C) + F_2(C) = \int_{\Omega} [u(x,y)]^m |I(x,y) - c_1|^2 dxdy + \int_{\Omega} [1 - u(x,y)]^m |I(x,y) - c_2|^2 dxdy \tag{8.33}$$

An image $I(x, y)$ is clustered into two regions by a fuzzy clustering algorithm, where $u(x, y)$ represents the membership values of a pixel for each region and m is a weighting exponent on each fuzzy membership. The model is formulated in a pseudo-level set due to the fact that membership values of $u(x, y) = [0, 1]$. The pseudo-level set is a set of Lipschitz similar function $u: I \rightarrow \mathbb{R}$ presented in Eq. 8.34 and maintains the ideology of using membership values to define $u(x, y)$.

$$\begin{cases} C = \{(x,y) \in I : u(x,y) = 0.5\} \\ C_{inside} = \{(x,y) \in I : u(x,y) > 0.5\} \\ C_{outside} = \{(x,y) \in I : u(x,y) < 0.5\} \end{cases} \tag{8.34}$$

Unfortunately, while the Fuzzy Energy based Active Contour proves to be computationally efficient, it suffers in its inability to robustly segment textural or multispectral images. However, the flexibility provided by fuzzy energy can be translated into the hybrid morphological active contour by changing the region-based step of the algorithm and morphing the level set into the pseudo-level set of Eq 8.34. The region step will take the form of Eq. 8.35, where c_1 represents the average inside the contour and c_2 is the average outside the contour.

$$u_{region}^{n+1} = \frac{1}{1 + \left(\frac{\lambda_1 (I - c_1)^2(\boldsymbol{x})}{\lambda_2 (I - c_2)^2(\boldsymbol{x})} \right)^{\frac{1}{m-1}}} \tag{8.35}$$

$$c_1 = \frac{\int_{\Omega} I * (u)^m d\boldsymbol{x}}{\int_{\Omega} (u)^m d\boldsymbol{x}} \quad c_2 = \frac{\int_{\Omega} I * (1-u)^m d\boldsymbol{x}}{\int_{\Omega} (1-u)^m d\boldsymbol{x}}$$

Fig. 8.3 Example of boundary extraction. **a** An original image courtesy of Caltech [28]. **b** A saliency map generated by CASD. **c** Edges from saliency map

The parameter m represents the fuzzy weighting exponent defined in the fuzzy rules for the clustering.

The visual attention model is incorporated into the hybrid morphological active contour in two ways. First, the saliency of an image is calculated and then transformed into an edge image to help define the boundary of the salient object (see Fig. 8.3) giving the algorithm the image created by $g(I)$ from Eq. 8.28. Second, the visual attention result is compared to the fuzzy clustering results, and the fuzzy clustering result, which most closely matches the saliency image, is used as the basis for the fuzzy energy (see Fig. 8.4).

The algorithm incorporating fuzzy energy and visual attention into the hybrid morphological active contour is given below, where the membership values from the fuzzy clustering most similar to the saliency image give u_0.

Algorithm (2)

$$pixel\ is \begin{cases} \text{on } C & \text{if} \quad I : u_0(x,y) = 0.5 \\ \text{inside } C & \text{if} \quad I : u_0(x,y) > 0.5 \\ \text{outside } C & \text{if} \quad I : u_0(x,y) < 0.5 \end{cases}$$

The Algorithm 2 includes the following steps.

Step 1

$$u^{n+1}_{balloon}(x) = \begin{cases} (\mathcal{D}_d u)(x) & \text{if } g(I)(x) > 0 \text{ and } v > 0 \\ (\mathcal{E}_d u)(x) & \text{if } g(I)(x) > 0 \text{ and } v < 0 \\ u^n_{balloon} & \text{otherwise.} \end{cases}$$

Fig. 8.4 Examples of fuzzy clustering result with three classes. **a** Class 1 membership map. **b** Class 2 membership map. **c** Class 3 membership map

The parameter $g(I)$ is an edge attractor image calculated using Eq. 8.28 on the saliency image.

Step 2

$$u_{region}^{n+1} = \frac{1}{1 + \left(\frac{\lambda_1 (I - c_1)^2(\boldsymbol{x})}{\lambda_2 (I - c_2)^2(\boldsymbol{x})}\right)^{\frac{1}{m-1}}}$$

Fig. 8.5 Segmentation results of fuzzy morphological active contour with visual attention. **a** An original image. **b** Segmentation result. **c** An original image. **d** Segmentation result

where $c_1 = \frac{\int_\Omega I*(u_{balloon})^m dx}{\int_\Omega (u_{ballon})^m dx}$, $c_2 = \frac{\int_\Omega I*(1-u_{balloon})^m dx}{\int_\Omega (1-u_{balloon})^m dx}$, and m is the fuzzy energy membership weighting exponent.

Step 3

$$u^{n+1} = \begin{cases} (\mathcal{A}_d \circ \mathcal{J}_d u^{n+1}_{region})(x_i) \text{ if } g(I)(x) > 0 \\ u^{n+1}_{region}(x_i) \quad \text{otherwise} \end{cases}$$

The images in the experiment are taken from the computational vision dataset at Caltech [28], GrabCut [29], and the Berkley Segmentation Dataset and Benchmark [30]. The images are chosen at random and are grayscale or RGB. The images have illumination artifacts, shadows, texture, the multiple objects to segment, intensity inhomogeneity, and noise, i.e. typical image artifacts that make segmentation of visual scenes a nontrivial task. Figure 8.5 demonstrates a random sampling of the results.

The comparison of the fuzzy c-means clustering to the saliency image is conducted automatically and uses similarity cues to make the selection, of which result to use. The membership function values of each pixel in the chosen clustering result are then used to create the level set function. This results in the use of fuzzy energy in the active contour and prompts a change in the calculations of the level set during the region competition portion of the method. As demonstrated with the sample image results, the algorithm successfully segments salient figures in interior and exterior environments. It also successfully segmented foreground figures in areas of high texture. These results lend themselves to the effective segmentation of static images for scene analysis.

8.4 Conclusion

The images segmented in this work represent just the tip of the possible image types this method has the potential to segment. It would be trivial to extend the work to a true multispectral algorithm as well as incorporating more texture cues into the fuzzy membership rules. The extension of the algorithm to video sequences would also be a simple matter of changing the image parameters to video and incorporating a comparison module for each frame of videos. The use of a saliency model to compare clustering results leading into the membership values for the contour evolution helps ensure the method will segment a salient object as defined by human vision research. As a final point, the savings in computational efficiency garnered from using a morphological curve evolution rather than a partial differential equation and corresponding Euler-Lagrange equations translates into a highly accurate and efficient segmentation method.

References

1. Wei W, Xin Y (2007) A modified multiphase level set evolution scheme for ariel image segmentation. Int J Pattern Recogn Artif Intel 21(7):1195–1212
2. Ilea DE, Whelan PF (2011) Image segmentation based on the integration of colour-texture descriptors—a review. Pattern Recogn 44(10):2479–2501
3. Mittal A, Sofat S, Hancock E (2012) Detection of edges in color images: a review and evaluative comparison of state-of-the-art techniques. In: Kamel M, Karray F, Hagras H (eds) Autonomous and intelligent systems, LNCS. Springer, Berlin
4 Chan T, Vese L, Sandberg Y (2000) Active contours without edges for vector-valued images. J Vis Commun Image Represent 11(2):130–141
5. LaValle M, Hutchinson S (1995) A Bayesian segmentation methodology for parametric image models. IEEE Trans Pattern Anal Mach Intel 17(2):211–217
6. Gonzalez R, Woods R, Eddins S (2009) Digital image processing using matlab, 2nd edn. Gatesmark Publishing, Knoxville
7. Pieczynski W (1992) Statistical image segmentation. Mach Graph Vis 1(1–2):261–268
8. Silverman J, Cooper D (1988) Bayesian clustering for unsupervised estimation of surface and texture models. IEEE Trans Patt Anal Mach Intel 10(4):482–496
9. Kass M, Witkin A, Terzopoulos D (1988) Snakes: active contour models. Int J Comput Vis 1 (4):321–331
10. Chan T, Vese L (2001) Active contours with edges. IEEE Trans Image Process 10(2):266–277
11. Sethian J (1999) Level set methods and fast marching methods, 2nd edn. Cambride monographs on applied and computational mathematics. Cambridge University Press, Cambridge
12. Matheron G (1975) Random sets and integral geometry. Wiley, New York
13. Serra J (1982) Image analysis and mathematical morphology, vol 1. Academic Press, London
14. Gurillo P, Frago N, Fuentes R (2003) Fuzzy morphological operators. Image Process Mathware Soft Comput 10(3):8–100
15. Mario I, Chacon M (2006) Fuzzy logic for image processing: definition and applications of a fuzzy image processing scheme. In: Bai Y, Zhuang H, Wang D (eds) Advanced fuzzy logic technology in industrial applications
16. Mondal K, Dutta P, Bhattacharyya S (2012) Fuzzy logic based gray image extraction and segmentation. Int J Sci EngResearch 3(4):1–14
17. Neisser U (1964) Visual search. Sci Am 210(6):94–102
18. Zhang Q, Gu G, Xiao H (2009) Image segmentation based on visual attention mechanism. J Multimedia 4(6):363–370
19. Goferman S, Zelnik-Manor L, Tal A (2010) Context-aware saliency detection. IEEE Int Conf Comput Vis Patt Recogn (CVPR'2010):2376–2383
20. Goferman S, Zelnik-Manor L, Tal A (2012) Context-aware saliency detection. IEEE Trans Patt Anal Mach Intel 34(10):1915–1926
21. Catte F, Dibos F, Koepfler G (1995) A morphological scheme for mean curvature motion and applications to anisotropic diffusion and motion of level sets. SIAM J Numer Anal 32 (6):1895–1909
22. Alvarez L, Baumela L, Henriquez P, Marquez-Neila P (2010) Morphological snakes. IEEE Int Conf Comput Vis Pat Recogn (CVPR'2010):2197–2202
23. Marquez-Neila P, Baumela L, Alvarez L (2014) A morphological approach to curvature-based evolution of curves and surfaces. IEEE Trans Patt Anal Mach Intel 36(1):2–17
24. Fox V, Milanova M, Al-Ali S (2013) A morphological multiphase active contour for vascular segmentation. Int J Bioinf Biosci 3(3):1–12
25. Fox V, Milanova M, Al-Ali S (2013) A hybrid morphological active contour for natural images. Int J Comput Sci Eng Appl 3(4):1–13
26. Kimmel R (2004) Numerical geometry of images: theory, algorithms, and applications. Springer Science + Business Media, New York

27. Krinidis S, Chatzis V (2009) Fuzzy energy-based active contours. IEEE Trans Image Process 18(2):2747–2755
28. Fei-Fei L, Fergus R, Perona P (2007) Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. J Comput Vis Image Underst 106(1):59–70
29. Rother C, Kolmogorov V, Blake A (2004) GrabCut: interactive foreground extraction using iterated graph cuts. In: SIGGRAPH'04, vol 23(3), pp 309–314
30. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: 8th International Conference on Computer Vision (ICCV'2001), vol 2, pp 416–423