An adaptive level set method based on joint estimation dealing with intensity inhomogeneity

Jiang Zhu1,2 | Yan Zeng1,2 | Jianqi Li2,3 | Shujuan Tian1,2 | Haolin Liu1,2

1 Key Laboratory of Hunan Province for Internet of Things and Information Security, Xiangtan University, Xiangtan, China
2 College of Automation and Electronic Information, Xiangtan University, Xiangtan, China
3 Hunan Province Cooperative Innovation Center for The Construction and Development of Dongting Lake Ecological Economic Zone, Hunan University of Arts and Science, Changde, China

Correspondence
Jianqi Li, College of Automation and Electronic Information, Xiangtan University, Xiangtan, China.
Email: jianqi_li@126.com

Abstract
Automatic object segmentation has been a challenging task due to intensity inhomogeneity. The traditional way is to eliminate the intensity inhomogeneity, which causes the object to lose useful intensity information. The authors propose an adaptive level set method for the segmentation of intensity inhomogeneous images. Firstly, global and local features are utilised to collaboratively estimate the image, which devotes to compensating for intensity inhomogeneity. The local estimation retains detailed spatial information, and the global estimation mainly contains the regional information of the partitioned object. Then, during the construction of the energy functional, joint estimation is introduced to create the external energy. To acquire the precise location of the boundary, a weighting factor indicated by the gradient is introduced into the internal energy. Finally, after the numerical calculation of the energy functional by additive operator splitting algorithm, this method achieves the desired performance in terms of accuracy and robustness. Experimental results verify this method outperforms the comparative methods and can be applied to many real-world scenarios.

1 | INTRODUCTION

Image segmentation occupies a fundamental position in the field of machine vision. As seen from the latest researches of surface defect detection, for example, leather defect [1], solder ball cracking [2], image acquisition enhancement [3], weld defect [4], stacked substrates counting [5], and so on, the object extraction and image segmentation remain the base tasks and critical issues. In practical or industrial scenarios, imperfection in camera and uneven illumination make intensity inhomogeneity inevitably occur in images. It means various contrasts may appear in a different position or inside a similar object. These unavoidable variations lead to poor segmentation and identification of image objects.

Recently, with the rapid development of deep learning, many image segmentation architectures have emerged. SegNet [6] uses an asymmetrical structure, the encoder is pooling, and the decoder is upsampling. The core lies in its decoder part. The decoding network maps low-resolution coded feature maps to full-resolution feature maps. U-Net [7] puts splicing features together in the channel dimension to form richer features. The major contribution of PSPNet [8] is Global Pyramid Pooling, which scales the feature map to several different sizes so that the features have better global and multi-scale information in improving segmentation accuracy. DeepLab [9] is called Dilated Convolution and uses the atrous algorithm to expand the receptive field and obtain more contextual information, which is pretty important in acquiring the features of various objects. The effect of the larger receptive field is that more global information of the image can be grasped when the feature map is reduced to a small size. Mask R-CNN [10] combines object detection with semantic segmentation which outputs the object category, bounding box, and mask, it can accurately locate the position of the object in the feature map. Actually, in the aforementioned segmentation models, many details and tiny objects cannot be effectively processed, while U-2-Net [11] performs talented in preserving details and edges. From its structure, we can see that U-2-Net obtains as much semantic information...
as possible by stacking more sequence-to-sequence model (i.e. encoder and decoder), and stacks the blocks of the work to get a larger gain. The architecture makes it the most successful and applicable salient segmentation model recently. However, deep learning is still faced with the following limitations. (1) Data-set requirement and labelling cost [12]. Deep learning bases on a great number of data, and the labelling cost of data to define the ground-truth is relatively large. (2) High hardware requirements [13]. Deep learning requires high computing power, and ordinary CPUs can no longer meet the requirements of deep learning. (3) Complicated model design [14, 15]. The design of deep learning models is very complex, requiring a lot of manpower, material resources, and time to develop new algorithms and models. Most researches and applications can only use ready-made models.

As an implicit active contour model, the level set method (LSM) is well investigated as it could track dynamic interfac-ing and shapes. LSM generally determines the driving energy to define a level set functional (LSF) and then solves a partial differential equation of LSF to conduct the contour evolution. Recently, various research works have focused on tackling intensity inhomogeneity, they can be classified into three categories.

The first category uses local information to construct energy functional [16–20] [21]. Generally, the spatially varying region information [16], local intensity means [17] and local intensity variations [19, 20] are efficient features to deal with intensity inhomogeneity. However, these methods consider that intensity inhomogeneity mainly affects the non-contour pixels, therefore, they are extremely sensitive to initialisation and sometimes invalid. The second category combines global and local intensity information to construct energy functional [22–26] [27]. Both average convolution operator and average mean are used to represent and describe the features of the original image [25]. In particular, the first two categories solely apply statistical constants (local or global information). They aim to obtain intensity differences, intensity averages, and variations to enhance the contrast between the foreground and background and ignore pre-processing of intensity inhomogeneity. The third category constructs energy functional based on image estimation and treats intensity inhomogeneity as a component [28] to approximate image regions linearly [29, 28, 30, 31, 32] or non-linearly [33–36]. The essential step of this category is to model intensity inhomogeneity as a bias field and define a region descriptor for images. Ali et al. [29] proposed a linear region descriptor to LSM for intensity inhomogeneous images. Unfortunately, the energy model they utilise is mostly for noise problems and inefficient to deal with intensity inhomogeneity, as intensity inhomogeneity remains the key matter. Zhang et al. [32] used the intensity average to linearly estimate images. Since intensity difference varies in different regions, it is unreliable to depend solely on average intensity. A LATE method based on Taylor Expansion [33] was proposed by using nonlinear estimation to deal with intensity inhomogeneity. However, this method still needs local region information to measure the intensity inhomogeneity and brings a huge loss of raw intensity information as the estimation process eliminates the intensity inhomogeneity severely. Wang et al. [36] introduced posterior probability to construct nonlinear LSF; the model only considers the probability of pixels belonging to the background or foreground, it is preferably suitable for images where probability distributions of high and low intensity tend to be equal (as images shown in their experiment).

Aiming at eliminating the annoying intensity inhomogeneity and improving the accuracy of object recognition in computer vision systems under great adaptability and performance, this paper combines the characteristics of the second and third categories aforementioned and proposes an efficient LSM that introduces joint estimation to acquire precise object segmentation. The main contributions of our method are summarised as follows:

- The joint estimation effectively compensates the disturbance of intensity inhomogeneity on objects; it directly utilises the intensity average and spatially varying average to linearly estimate the original image, so that both global and local information can make devotion to segmentation.

- Our estimation descriptor has no requirement of approximation to intensity inhomogeneity degree which is difficult to be evaluated precisely, avoids the information loss of object intensity, and guarantees the accurate segmentation.

- We introduce a weighting internal energy that serves as a regularising term rather than driving energy in [37] to control contour evolution. It reduces iteration time and makes the final contour locate the object boundary accurately.

- As the parameters in image estimation can be obtained automatically, no manual estimation and intervention is needed, the proposed method has strong adaptability to computer vision applications.

2 | RELATED WORK

2.1 | Image estimation

The estimation problem of the image region is treated as linear [28] and nonlinear [33]. For linear estimation, Li et al. [28] defined image approximation as

\[ I(x) = b(x) \cdot I_0 + n. \]  

where \( I_0 \) denotes the original image, \( n \) is the noise, and \( b(x) \) is a matrix of intensity inhomogeneity with the same dimension as the original image. Replace \( I_0 \) with local intensity level constant \( C_i \), the simplified format is defined by:

\[ I(x) = b(x) \cdot C_i, \]  

where \( C_i \) is the average mean of partitioned areas.

A non-linear estimation model LATE [33] is proposed, which utilises Taylor expansion to estimate images,

\[ I(x) \approx I(y) + I'(y)dx + \cdots + \frac{I^n(y)}{n!} dx^n. \]
The average mean of the inner and outer regions is selected as the constant term of Taylor’s formula, denoted by:

\[ I(x) \approx \text{mean}_i(x) + b'(y)C_i, \quad (4) \]

As aforementioned, the general way is to denote image descriptor by the product of intensity inhomogeneity \( b \) and piecewise constants \( \text{mean}_i(x) \) or \( C_i \) performing an image estimation. These constants \( \text{mean}_i(x) \) or \( C_i \) can be presented easily and accurately, while the intensity inhomogeneity \( b \) is difficult to be accurately evaluated and needs to be defined manually. It is the imprecise definition of intensity inhomogeneity that causes a great adverse impact on image estimation, makes the LSMs poorly adaptive, and leads to information loss of the object intensity.

### 2.2 Weighting internal energy

The internal energy which controls the direction of contour evolution [38, 37] generally includes area term and length term. The normal direction is derived from the gradient. Consequently, in [37], the method substitutes the weighting factor generated by gradient to traditional length term \( R(\phi) = \int_{\Omega} \delta(\phi)|\nabla \phi| dx \) and area term \( A(\phi) = \int_{\Omega} \phi \phi |dx| \) to obtain weighting length term \( R(\phi) = \int_{\Omega} \omega(\phi, k) \delta(\phi)|\nabla \phi| dx \) and weighting area term \( A(\phi) = \int_{\Omega} \omega(\phi, k) H(\phi) dx \).

In [37], both weighting length term and weighting area term are used to control level set evolution. However, the weighting area term is exactly unnecessary, as there exists a relationship: \( \text{Area}_{\text{inside}(C)} \leq c(\text{Length}(C))^{N/(N-1)} \). Hence, we apply only weighting length energy term to construct internal energy (details see Appendix I). Our weighting internal energy achieves desirable performance on improving contour smoothness and fitting degree.

### 2.3 Relevant LSMs

These relevant methods are briefly introduced and compared with our method in the experiment section. LBF [17] uses the convolution kernel to obtain the local features processing intensity inhomogeneity, given by Equation (5):

\[ E^{LBF}(\phi) = \lambda_1 \int_{\Omega} K_\sigma(x-y)[I_0(y) - f_2(x)]^2 H(\phi)dx + \lambda_2 \int_{\Omega} K_\sigma(x-y)[I_0(y) - f_2(x)]^2 (1 - H(\phi))dx, \quad (5) \]

where constants \( f_i \) are derived from local details of the image, \( I_0 \) denotes the original image.

Wang et al. proposed LGIF [24] model combining global and local information.

\[ E^{LGIF}(\phi) = (1 - \omega)(\lambda_1 \int_{\Omega} (I_0(x) - C_1)^2 H(\phi)dx + \lambda_2 \int_{\Omega} (I_0(x) - C_2)^2 (1 - H(\phi))dx) + \omega(\lambda_1 \int_{\Omega} K_\sigma(x-y)[I_0(y) - f_2(x)]^2 H(\phi)dx + \lambda_2 \int_{\Omega} K_\sigma(x-y)[I_0(y) - f_2(x)]^2 (1 - H(\phi))dx). \quad (6) \]

The model is a simple combination of CV model (the former two terms in Equation (6)) and LBF model (the last two terms of Equation (6)), so that the driving energy is originated from the original images with intensity inhomogeneity. LATE [33] uses local intensity mean and Taylor model to non-linearly estimate the original image, and its energy model is as shown in Equation (7).

\[ E^{LATE}(\phi) = K_\sigma(x-y)\lambda_1 \int_{\Omega} (I_0(x) - \text{mean}_1(x) - b'(y)C_1)^2 H(\phi)dx + \lambda_2 \int_{\Omega} K_\sigma(x-y)[I_0(x) - \text{mean}_2(x) - b'(y)C_2]^2 (1 - H(\phi))dx. \quad (7) \]

A linear likelihood model LSACM [32] is introduced and the driving energy function is as Equation (8). The related limitation of existing estimation is analysed in Section 2.1.

\[ E^{LSACM}(\phi) = \sum_{i} \lambda_i \int_{\Omega} K_\sigma(x-y)[\log(x)\sigma + \frac{(I_0(x) - b'(y)c)^2}{2\sigma^2}]dx. \quad (8) \]

### 3 METHODOLOGY

#### 3.1 Joint estimation of image region

Li et al. [28] ideally assumed the segmented region as a constant, which is impossible to be strictly a constant. However, according to the assumption, we can utilise constants to estimate image regions to decreasing the interference of the intensity inhomogeneity. Naturally, how to define the constants so that the raw information of the ideal image can be preserved becomes a significant problem deserving consideration.

Set \( \Omega \) be image domain, the contour \( \Psi \) divides \( \Omega \) into \( \Omega_1 \) and \( \Omega_2 \), where \( \Omega_1 \) and \( \Omega_2 \) are inside and outside the contour,
respectively. Let \( \phi : \Omega \rightarrow \mathbb{R} \) be Lipschitz function, these two disjoint regions \( \Omega_1 \) and \( \Omega_2 \) are defined as: \( \Omega_1 = \{ x : \phi(x) > 0 \} \) and \( \Omega_1 = \{ x : \phi(x) < 0 \} \).

The region descriptor is difficult to be denoted as intensity inhomogeneity may lead to an overlap of the intensity distribution. Now that each object in the ideal image tends to be a constant, the major task to estimate the image region is finding out a proper constant to depict the region. Our work combines local and global information of the image region to define the estimation, that is, finding two constants to represent local and global information of the image region. Two constants correspond to the spatially varying average and the average mean \( C \), respectively. \( f \) is used to conduct the local estimation which acquires spatially average of the image region, and the global estimation related to \( C \) mostly acts as a regularising term to guarantee the estimation accuracy of image regions.

We use Gaussian convolution to generate the constant \( f \), the Gaussian kernel \( K \) is used to complete the convolution process. Due to the larger details applied to local estimation, it can result in an estimation blur. Hence, the global constant \( C \) essentially represents the regularising information. The definitions are given, respectively.

\[
\begin{align*}
I_L(x) &= f_1 H(\phi) + f_2 (1 - H(\phi)), \\
I_G(x) &= C_1 H(\phi) + C_2 (1 - H(\phi)),
\end{align*}
\]

(9) (10)

where \( I_L(x) \) represents the local estimation, \( I_G(x) \) is the global estimation and \( H(\phi) \) denotes Heaviside function. The parameters \( f_1, C_1 \) and \( f_2, C_2 \) are constants that represent the features in \( \Omega_1 \) and \( \Omega_2 \). Then, we combine \( I_L(x) \) and \( I_G(x) \) to define the joint estimation as \( I(x) \):

\[
I(x) = m I_L(x) + (1 - m) I_G(x),
\]

(11)

where \( m \) is the weighting factor of both global and local estimation, and different segmentation effects can be obtained by adjusting the value of parameter \( m \).

The optimal values \( C_1, C_2, f_1 \) and \( f_2 \) are denoted by the following equations and Appendix II explains the acquisition process.

\[
\begin{align*}
C_1 &= \frac{\int_{\Omega} I_0(x) H(\phi) dx}{\int_{\Omega} H(\phi) dx}, \\
C_2 &= \frac{\int_{\Omega} I_0(x) (1 - H(\phi)) dx}{\int_{\Omega} 1 - H(\phi) dx}, \\
f_1 &= \frac{K_{G} * [H(\phi) I_0(x)]}{K_{G} * H(\phi)}, \\
f_2 &= \frac{K_{G} * [(1 - H(\phi)) I_0(x)]}{K_{G} * (1 - H(\phi))},
\end{align*}
\]

(12) (13) (14) (15)

where \( K_{G} \) is the Gaussian Kernel with standard deviation and the symbol * is the convolution operator.

The basic principle is using local estimation \( I_L(x) \) and global estimation \( I_G(x) \) for cooperated estimation. The global estimation mainly acts as a regularising term in the joint estimation. To control estimation weight, a factor \( m \) is used in the final estimation image \( I(x) \). The joint estimation effectively avoids the direct evaluation of intensity inhomogeneity. The information loss caused by eliminating intensity inhomogeneity is also indirectly compensated. Our joint estimation achieves a more prominent benefit than the traditional estimation of eliminating intensity inhomogeneity.

### 3.2 Construction of LSF

The driving energy \( F(\phi) \) is generally made up of three terms and denoted by \( F(\phi) = E(\phi) + R(\phi) + P(\phi) \). The external energy term \( E(\phi) \) plays the most important role in this model. The image estimation model is also applied to the external energy term. For internal energy \( R(\phi) \), the length term with weighting factors is used to control the accurate extraction of the target boundary. Finally, a penalty term \( P(\phi) \) is added to reduce the computational complexity of level set initialisation.

As for the external energy, we introduce the estimated image \( I(x) \) into the external energy, that is, obtain the difference between the estimated image and these constants link to the original image. This approach guarantees the information of the original image and estimated image synergises on overcoming intensity inhomogeneity. The external energy \( E(\phi) \) is given by:

\[
E(\phi) = \omega (\lambda_1 \int_{\Omega} (I(x) - C_1)^2 H(\phi) dx + \lambda_2 \int_{\Omega} (I(x) - C_2)^2 (1 - H(\phi)) dx)
\]

\[
+ (1 - \omega) \lambda_1 \int_{\Omega} (I(x) - I_0(x) - d_1)^2 H(\phi) dx
\]

\[
+ \lambda_2 \int_{\Omega} (I(x) - I_0(x) - d_2)^2 (1 - H(\phi)) dx,
\]

(16)

where \( \lambda_1 \) and \( \lambda_2 \) are constants controlling weights of information inside and outside level set contour. \( \omega \) is weight of different part in external energy. The constants \( C_1 \) and \( C_2 \) are denoted by Equations (12) and (13). Minimise the last two terms of the energy functional Equation (16), as various details briefly discussed in Appendix II, \( d_1 \) and \( d_2 \) are defined by Equations (17) and (18), respectively. The intensity feature of the image obtained by estimation guides the evolution to fit target boundary.

\[
d_1 = \frac{\int_{\Omega} (I(x) - I_0(x)) H(\phi) dx}{\int_{\Omega} H(\phi) dx},
\]

(17)

\[
d_2 = \frac{\int_{\Omega} (I(x) - I_0(x)) (1 - H(\phi)) dx}{\int_{\Omega} (1 - H(\phi)) dx},
\]

(18)
The weighting internal energy is used to promote the evolution speed and improves the target segmentation accuracy. It is written as follows:

\[ R(\phi) = \int_{\Omega} \eta(\phi, k) \delta(\phi) |\nabla \phi| \, dx, \]  

(19)

where \( k \) is a constant representing the number of contours adjacent to the current contour. Local edge information [39] used to construct weighting factor is derived from these adjacent contours. The specific form of the weighting factor [37] is defined as follows:

\[ \eta(\phi, k) = \beta(\phi, k)^{(1 - \gamma(\phi, k))}, \]  

(20)

where \( \beta(\phi, k) \) denotes the average strength of edge operators adjacent to the current contour.

\[ \beta(\phi, k) = \frac{1}{2k} \left( \sum_{i=1}^{k} (1 - g) \psi(\phi, n) \right) _{\Omega} \]

\[ + \sum_{i=1}^{k} \left( (1 - g) \psi(\phi, -n) \right) _{\Omega}, \]  

(21)

where \( \psi(\phi, n) \) is the adjacent contour to the current contour \( \Psi \) and contains the gradient information, denoted by:

\[ \psi(\phi, n) = \delta(\phi) |\nabla \phi| + n \tilde{N}, \]  

(22)

where \( \tilde{N} = \nabla \phi / |\nabla \phi| \), it helps to generate a new contour in the direction of zero level set \( \phi \). The value of constant \( n \) inside the current contour \( \Psi \) is negative and outside the contour is positive. The edge operator \( g \) is a formal formula as \( g = \frac{1}{1 + |\nabla \kappa_x + \kappa_y|} \). The gradient vector flow (GVF) field is a static external force field, \( \nabla \tilde{V} = [u(x, y), v(x, y)] \), \( u \) and \( v \) are the derivatives in two directions \( x \) and \( y \), respectively. According to the alignment between the ordinary field of the current contour and the GVF field of the adjacent contour, the average built-in parameter is given as follows:

\[ \gamma(\phi, k) = \frac{1}{2k} \left( \sum_{i=1}^{k} \left( \nabla \tilde{V}, \nabla \tilde{V} \right) _{\Omega} \right) _{\phi} \psi(\phi, n) \, dx \]

\[ + \sum_{i=1}^{k} \left( \nabla \tilde{V}, \nabla \tilde{V} \right) _{\Omega} \psi(\phi, -n) \, dx, \]  

(23)

where \( \tilde{V} \) is the GVF field of the original image. If the ordinary field \( \tilde{N} \) of the current contour coincides with the distribution \( \tilde{V} \), the value \( \gamma(\phi, k) \) approaches to 1.

Generally, the re-initialisation leads to heavy computational burden [40]; the energy penalty term \( P(\phi) \) assigns \( |\nabla \phi| \) to approaching 1 rather than specifically initialising \( \phi \) function, which eliminates contour islands and decreases the re-initialisation complexity.

\[ P(\phi) = \int_{\Omega} \frac{1}{2} (|\nabla \phi| - 1)^2 \, dx. \]  

(24)

Combining the external energy, internal energy and penalty energy, we model the energy functional utilising global and local estimation as follows:

\[ F(\phi) = \omega \lambda_1 \int_{\Omega} (I(x) - C_1)^2 H(\phi) \, dx \]

\[ + \lambda_2 \int_{\Omega} (I(x) - C_2)^2 (1 - H(\phi)) \, dx \]

\[ + (1 - \omega) \lambda_1 \int_{\Omega} (I(x) - I_0(x) - d_1)^2 H(\phi) \, dx \]

\[ + \lambda_2 \int_{\Omega} (I(x) - I_0(x) - d_2)^2 (1 - H(\phi)) \, dx \]

\[ + \mu \int_{\Omega} \eta(\phi, k) \delta(\phi) |\nabla \phi| \, dx + \int_{\Omega} \frac{1}{2} (|\nabla \phi| - 1)^2 \, dx, \]  

(25)

where \( \mu \) is the weight of internal energy.

### 3.3 Numerical calculation method

Appropriate numerical calculation is an essential step to solve complex LSFs. Upwind method, difference method [41] and Newtonian fast algorithm [42] are general ways. However, to the problems formulated as the optimisation of a cost function, the numerical optimisation typically relies on first-order and second-order information to guide the evolution [43]. We apply the special algorithm, which greatly reduces the expensive burden. \( C_i, d_i \) and \( f_i \) are denoted as aforementioned minimising relative energy term. The evolution of the contour is given by an initial contour at \( t \) of the function \( \phi(t, x, y) \). The optimisation can be defined by parameterising the descent direction with an artificial time \( t \) [44]. According to the chain rule [45], to minimise the energy functional Equation (25) with respect to function \( \phi, \lambda_1 = \lambda_2 = 1 \), the gradient descent function can be derived,

\[ \frac{\partial \phi}{\partial t} = - \frac{\delta F}{\delta \phi} = (|\nabla \phi| - 1) \text{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) \]

\[ + \mu \eta(\phi, k) \delta(\phi) \text{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) \]

\[ + \delta(\phi)[(1 - \omega)(I(x, y) - I_0(x, y) - d_1)^2 \]

\[ + (1 - \omega)(I(x, y) - I_0(x, y) - d_2)^2 \]

\[ - \omega(I(x, y) - C_1)^2 + \omega(I(x, y) - C_2)^2], \]  

(26)
where $\nabla$ is the gradient operator and $\text{div}(\cdot)$ is the divergence operator. The function $H(\phi)$ is the differential version of Heaviside function. The Dirac delta function $\delta(\phi)$ is the derivative of $H(\phi)$ defined by Equation (28),

$$H(x) = \frac{1}{2}(1 + \frac{2}{\pi} \arctan(x)),$$

$$\delta(x) = \frac{1}{\pi \varepsilon^2 + x^2}. \quad (28)$$

Let the latter term of Equation (26) be $F(x, y)$,

$$F(x, y) = \delta(\phi)[(1 - \omega)(I(x, y) - l_0(x, y) - d_1)^2 + (1 - \omega)(I(x, y) - l_0(x, y) - d_2)^2 - \omega(I(x, y) - C_1)^2 + \omega(I(x, y) - C_2)^2]. \quad (29)$$

In following process, the discretisation of $\phi$ is firstly required and iteration termination condition that $\phi$ comes to be stable must be satisfied. Therefore, we first discretise $\nabla \phi$ from $x$ and $y$ directions to avoid poorly implemented derivation operations and obtain a more suitable numerical calculation scheme, and the term $|\nabla \phi|$ is denoted by a two-dimensional format $|\nabla \phi| = \sqrt{\phi_x^2 + \phi_y^2}$, accordingly. We define $E = \frac{1}{|\nabla \phi|}$, and $\text{div}(\frac{\nabla \phi}{|\nabla \phi|}) = (\partial_x E \partial_y \phi + \partial_y E \partial_x \phi)$, the two-dimensional equation of the gradient descent function can be rewritten as,

$$\frac{\partial \phi}{\partial t} = (|\nabla \phi| - 1 + \mu \eta(\phi, k) \delta(\phi)) (\phi + \mu \eta(\phi, k) \delta(\phi))$$

$$+ \partial_y E \partial_x \phi + F(x, y). \quad (30)$$

We use $A_i$ to define the relationship between the current contour $\phi^n$ and the contour $\phi^{n+1}$ after evolution, and rewrite the contour evolution equation into a discrete format. According to the characteristics of the central difference method [46] and the additive operator splitting algorithm [29], we utilise the discrete relationship between $\phi^n$ and $\phi^{n+1}$ to replace $\frac{\partial \phi}{\partial t}$, and central finite difference to define discrete relationships in the horizontal and vertical directions. The following difference steps can be obtained,

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} = \sum_{i=1}^{2} A_i(\phi^n)(\phi^{n+1} + F(x, y)). \quad (31)$$

Equation (31) is the discrete central-finite-difference form of Equation (30), where $A_i$ are tri-diagonal matrices for discretisation and given by Equations (32) and (33), where $A_1(\phi^n)$ represents the horizontal correlation and $A_2(\phi^n)$ is the vertical correlation.
4.1 Parameter setting

We summarise the selection of parameters. The relevant parameters for the common situation are listed in Table 1, which are general choices in different LSMs. 

| Methods | \( \lambda \) | \( \omega \) | \( \mu \) | \( \sigma \) | \( m \) |
|---------|-------------|-------------|-------------|-------------|-------------|
| LBF     | 1           | 0          | 0.002 * 255^2 | 3           | --          |
| LCV     | 1           | 0.05       | 0.001 * 255^2 | 4           | --          |
| LGIF    | 1           | 0.01       | 0.001 * 255^2 | 3           | --          |
| LSACM   | 0.99        | --         | 0.000005 * 255^2 | 10          | --          |
| LATE    | 1           | --         | 0.001 * 255^2 | 25          | --          |
| OURS    | 1           | 0.05       | 0.002 * 255^2 | 3           | 0.9         |

The parameter \( \sigma \) is a scale parameter of the Gaussian kernel which greatly influences the performance of estimation. In our method, \( \sigma \) affects the global estimation, the estimated image blur reduces as \( \sigma \) decreases. When we segment images with not rule out that some methods (e.g. LSACM) need to adjust \( \lambda \) to obtain better results. 

\( \mu \) is the parameter that controls the weight of internal energy. Generally, the internal energy is derived from the contour length. \( \mu \) is usually 0.0001*255*255 or 0.0002*255*255 for most of the comparative methods, while LSACM needs to adjust the internal energy ratio to obtain better results.

\( \omega \) is the external energy factor, it takes the value \([0,1]\). In the two-phase segmentation, if one external energy factor is \( \omega \), the other is supposed to be \((1-\omega)\). Thus, when we are more inclined to local information during the segmentation process, the value of \( \omega \) that controls the local energy term is \([0.5,1]\). LBF, LSACM and LATE only have one type of external energy, so the \( \omega \) value is not set, and their default is 1.

\( \sigma \) is the scale of the Gaussian kernel. For images with dense details and high quality, we need more local information, the value of \( \sigma \) is small, usually less than 10. For images with intensity inhomogeneity and sparse distribution, the global information plays a vital role, the value of \( \sigma \) is usually greater than 10 (e.g. the \( \sigma \) value of LSACM and LATE, as they mainly deal with severe low-quality images).

\( m \) is an unique parameter of the proposed method, which controls the proportion of local estimation and global estimation. The adjustment of \( m \) is similar to \( \sigma \). The larger \( m \) represents a greater proportion of local features in the energy functional. Different from the value of \( \sigma \) that will affect the accuracy of target segmentation, \( m \) only controls the proportion of the energy functional and does not affect the image feature collection.

4.2 Performance evaluation of parameters

We firstly analyse the effect of one parameter varying and the other fixed. In this subsection, the chosen image is a severe intensity inhomogeneous image with two mushrooms (Figure 1) where the background has a similar intensity distribution with the object.

4.2.1 Experimental analysis of \( \sigma \)

The parameter \( \sigma \) is a scale parameter of the Gaussian kernel which greatly influences the performance of estimation. In our method, \( \sigma \) affects the global estimation, the estimated image blur reduces as \( \sigma \) decreases. When we segment images with
weaker boundary or larger intensity inhomogeneity, we set the smaller value of $\sigma$ to preserve more edge details. As shown in Figure 1a, the segmentation results with different $\sigma$ value and fixed $m = 0.9$ are given, and it is $\sigma = 5$ that results in the outstanding segmentation of the experimental image. The smaller $\sigma$ may result in the larger global redundant segmentation (Figure 1a Column 1–2). Oppositely, if $\sigma$ being too large, the estimation blur will appear. Consequently, as $\sigma$ grows (Figure 1a Column 4–5), the final segmentation can identify the object in general but unable to fit the real target. $\sigma$ ranges 3 to 5 achieves the satisfactory results, while $\sigma = 3$ is the most frequent choice. Since $\sigma$ is not only a parameter that controls the proportion, it also has certain functionality that controls the convolution operation and affects the process of feature collection. The larger the $\sigma$ value, the more unstable the target segmentation accuracy. As in low-quality images, the larger convolution kernel is utilised to reduce the impact of noise and intensity inhomogeneity, which makes the energy functional less enable to fit the object boundary. If the details inside the object are sparse, the $\sigma$ value should be smaller, which can reduce the interference of the non-target to the target. When the object has more internal details, a larger $\sigma$ value should be selected.

4.2.2 | Experimental analysis of $m$

The parameter $m$ is the other vital parameter for the proposed method, which adjusts the estimation weight of local information and global information, and the value of $m$ takes in $[0, 1]$. A small value of $m$ implies collecting image information in a small weight of local features and a large weight of average means. On the contrary, the larger values of $m$ imply collecting local features in large weight and regularising details in a small weight. Specifically, when the target image has more details or interference, the $m$ is set relatively larger to preserve local details in large weight and regularising details in a small weight. For this case, $m = 0.9$ achieves optimal segmentation performance. The smaller $m$ value leads to a wider global redundant area (Figure 1b Column 1–3). For general cases, the value of $m$ in the interval of $[0.7, 0.95]$ results in a better effect. In most cases, setting $m = 0.9$ achieves the greatest performance. Due to the existence of the $m$, the $\sigma$ in our method can be set to a smaller value to avoid a large kernel resulting in the instability of segmentation accuracy. It is a vital reason why our joint-

![Figure 1](image-url)

**FIGURE 1** The segmentation results of our method with fixed $\sigma$ and $m$: (a) different $\sigma$ and fixed $m = 0.9$; (b) different $m$ and fixed $\sigma = 3$

estimation-based method has high segmentation accuracy and strong robustness. Additionally, for images with complex targets, the value of $m$ is set relatively larger to preserve local details and extract global targets.

4.3 | Comparative experiments

4.3.1 | Comparison with state-of-the-art LSMs

In this part, we demonstrate three test sets of the methods LBF [17], LGIF [24], LCV [25], LSACM [32], LATE [33] and the proposed method on the normal, natural and medical images. Typically, the first column of numbers in Tables 2–7 represent images in corresponding figures. To quantify performance comparisons, the segmentation accuracy is measured by the index

$$JSC(O_m, O_t) = \frac{\mathcal{A}(O_m \cap O_t)}{\mathcal{A}(O_m \cup O_t)},$$

(37)

$O_m$ and $O_t$ are the segmented and real target area, respectively. $\mathcal{A}(O_m \cap O_t)$ is the intersection area of $O_m$ and $O_t$, and $\mathcal{A}(O_m \cup O_t)$ is the union area. JSC value equals to 1 represents the most ideal situation. Furthermore, the running time of each method are also compared. The best data in tables is emphasised in bold. We also utilise $\{\}$ to delete the running time link to segmentation whose JSC value is below 0.7, as they are meaningless to the analysis.

**Test Set 1: The performance on normal images.** Firstly, in experiment 1, the six methods are applied to normal images, the segmentation results are shown in Figure 2. Tables 2 and 3 record the JSC value and running time corresponding to each

| LBF | LGIF | LCV | LSACM | LATE | OURS |
|-----|------|-----|-------|------|------|
| 1   | 0.9968 | 0.9967 | 0.9965 | 0.9925 | 0.9903 | **0.9969** |
| 2   | 0.9910 | 0.9912 | 0.9912 | 0.9903 | 0.9903 | **0.9953** | 0.9948 |
| 3   | 0.9935 | 0.9930 | 0.9940 | 0.9932 | 0.9869 | **0.9942** |
| 4   | 0.9132 | 0.9876 | 0.9861 | 0.9856 | 0.9877 | **0.9903** |
| 5   | 0.9917 | 0.9952 | 0.9908 | 0.9910 | 0.7000 | **0.9973** |
FIGURE 2 The comparison of 6 methods on normal images: (a) LBF (b) LGIF (c) LCV (d) LSACM (e) LATE (f) OURS

processing in Figure 2 on normal images. It can be seen that LGIF, LCV, LSACM, and our method have a similar effect. The basic contour can perfectly identify the target, and the boundary of targets is accurately segmented. LBF and LATE can also achieve favourable contour segmentation, except LBF may have redundant contour parts (Figure 2a row 4). Because of the strong edge of objects and contrast of the images, the results of these different methods show nearly similar performance. Under the situation, as LATE eliminates intensity inhomogeneity which takes heavy information loss of the intensity, it becomes relatively ineffective and may acquire missing segmentation (Figure 2e row 5).

Quantitatively, in Table 2, it can be seen that the JSC values of the methods maintain above 0.98 on whole images. Table 3 represents the executive time of different methods on these images. LCV, LSACM and our method spend shorter running time relatively. Our method and LSACM are roughly the same on this index, while both segmentation accuracy and target contour smoothness are better than the LCV method. Worth emphasising, our method has the same segmentation performance as state-of-the-art methods on normal images while our executive time is apparently the shortest.

Test Set 2: The performance on natural images. For the challenging case, experiment 2 applies the six methods to natural images, as shown in Figure 3. Tables 4 and 5 are the JSC value and running time concerning the image processing in Figure 3. The drawback of LBF, LGIF, and LCV models is that they consider only the statistical information making them unable to fit real-world objects accurately and can only identify the main part of objects. Meanwhile, due to the inaccurate evaluation of intensity inhomogeneity, LSACM and LATE methods can complete the accurate segmentation but sometimes cause missing segmentation (Figure 3d row 5, Figure 3e row 5). Additionally, different from LSACM and LATE, our method has no necessity to evaluate intensity inhomogeneity degree, and consequently will not lead to missing segmentation.
advantages of the weighting factor are vividly reflected in the intensity-inhomogeneous images, our final contours are with high smoothness and no sharp blurs. Due to the joint estimation, our method performs much better than other methods in boundary segmentation, obvious boundary and less isolated island area occur in the results (Tables 4–5 data in bold).

Table 4 compares the segmentation accuracy of different methods. As shown in Table 4, for intensity inhomogeneity in natural images, LBF, LCV, and LGIF will cause local redundant segmentation which means the wrong segmentation inside targets. It should be noted that our model is superior to other methods in JSC, and achieves the shortest executive time for the majority of experiments. In Table 5, our method spends a shorter running time relatively than comparative methods. The experimental results show that the traditional method is not suitable for intensity inhomogeneous images, and will result in missing segmentation. Our method performs more outstanding than other methods from segmentation results to contour smoothness. The short running time and fast convergence speed are also revealed.

Test Set 3: The performance on medical images. For another challenging case, experiment 3 illustrates the six kinds of comparative results on medical images, as shown in Figure 4. Tables 6 and 7 represent the JSC value and running time of images in Figure 4, respectively. The LBF, LGIF, and LCV methods have poor robustness and trouble to stabilise in the contour evolution process, while LSACM and LATE relatively perform better. However, as the object regions are all treated with same intensity inhomogeneity, LATE fits the target better than LSACM, but it may cause missing segmentation (as Figure 4e rows 2 and 5). LSACM (as Figure 4d rows 5 and 8) is slightly better than LATE in missing segmentation, while the fitting degree of objects in LSACM is worse. The proposed method is capable to extract the edge of objects and owns the best fitting degree. We demonstrate the ability of our method to
handle the weak boundaries (see Figure 4); as the results show, it achieves satisfactory performance on detecting the accurate boundaries and fit the desirable objects.

The JSC values in Table 6 match the results in Figure 4, especially when the targets in the image are complicated. LBF, LGIF, and LCV perform poorly as they utilise only the average mean or pixel intensity. In Table 7, our running time is also the shortest. With the comprehensive consideration of JSC value, under high segmentation accuracy, the final segmentation contour of our method has a larger fitting degree to the target and better
extraction of the boundary. Our method is obviously superior to other methods.

### 4.3.2 Comparison with deep learning method

We compare the most classical deep-learning-based segmentation models, such as MaskRCNN, DeepLab, PSPNet, U-2-net with our method. As the method of transfer learning is a general way to solve the problem of missing data or small data sets and the test images in this experiment are mainly derived from the PASCAL VOC data set, the corresponding pre-trained networks on the PASCAL VOC data set are selected. The deep learning method is applied to the GPU and our method runs on the CPU, consequently, we ignore the meaningless comparison of running time.

**Figure 5** shows the results of the four methods and the proposed method. Table 8 is the JSC value corresponding to the segmentation result. It can be seen from these results that most deep learning methods are poorly adaptable. Since the features obtained by the deep learning method on the training set are stable, the closer the feature of the test image is to the training set, the better the segmentation results obtained. Deep learning methods have unsatisfactory segmentation accuracy for small targets and images with large differences between target features and training set. Secondly, the measurement standard of deep learning methods in large data sets is mainly for the total accuracy rate of all test images, but this does not mean that an excellent segmentation effect can be obtained for each image, as the segmentation results of images (7), (8), (9) and (12) shown in Figure 5a, and corresponding images (1), (4), (7), (12), (14) and (15) in Figure 5b,c. The unsatisfactory segmentation effect of the former three methods mainly because of the mismatch between the test images and training images. Among these deep-learning-based segmentation models, U-2-Net, as a method for salient objects, achieves the best performance. However, due to the total concern of the object saliency, it may cause a small bug. For example, the image (13) in Figure 5d, it ignores the goose itself to extract the most prominent eye of the goose. As Table 8

![Figure 5](image)

**Table 8** JSC values of deep learning method

| Mask-RCNN | DeeplabV2 | PSPNet | U-2-Net | OURS |
|-----------|-----------|--------|---------|------|
| 1         | 0.9801    | -      | -       | 0.9824 | 0.9935 |
| 2         | 0.9947    | 0.8656 | 0.9532  | 0.9946 | 0.9941 |
| 3         | 0.9804    | 0.9420 | 0.9886  | 0.9950 | 0.9932 |
| 4         | 0.9755    | 0.1982 | -       | 0.9774 | 0.9912 |
| 5         | 0.9701    | 0.9754 | 0.9772  | 0.9936 | 0.9938 |
| 6         | 0.9915    | 0.9705 | 0.9733  | 0.9920 | 0.9900 |
| 7         | 0.5116    | 0.1521 | 0.3022  | 0.9918 | 0.9926 |
| 8         | 0.5032    | 0.7946 | 0.8656  | 0.9805 | 0.9869 |
| 9         | 0.2523    | -      | 0.9611  | 0.9932 | 0.9935 |
| 10        | 0.9903    | 0.9912 | 0.9856  | 0.9932 | 0.9902 |
| 11        | 0.9704    | 0.9821 | 0.9810  | 0.9923 | 0.9766 |
| 12        | -         | -      | -       | 0.9922 | 0.9922 |
| 13        | 0.9592    | 0.9768 | 0.9924  | 0.2074 | 0.9927 |
| 14        | 0.9935    | 0.3641 | -       | 0.9932 | 0.9935 |
| 15        | 0.9815    | -      | -       | 0.9826 | 0.9950 |
| 16        | 0.9900    | 0.9904 | 0.9905  | 0.9905 | 0.9911 |
shows, in most cases, compared to these deep learning models, our method has the best segmentation accuracy. In other cases, our method can also approach the considerable performance. It is the proposed method that can provide high segmentation accuracy and adorable adaptability.

4.4 Robustness of contour initialisation

To prove the robustness over the initial contour, we test the varying positions and shapes, and the satisfactory experimental results are obtained.

**Situation 1: The positions of initial contour.** In Figure A.1, we set four typical situations of initial contour: contour inside object (Figure A.1(a), column 1), contour encloses one objects (Figure A.1(a) column 2), contour outside objects (Figure A.1(a) column 3) and contour encloses all objects (Figure A.1(a) column 4). Each situation iterates 30 times, and the final results of 10 times (Figure A.1(b)), 20 times (Figure A.1(c)) and 30 times (Figure A.1(d)) are recorded. From Figure A.1, we observe that although the evolution direction of each initial contours is different, the final segmentation result converges to the real object. To the experimental image in Figure A.1, after 30 times iterations, JSC values of four situations are all equal to 0.996.

**Situation 2: The shapes of initial contour.** Furthermore, we evaluate the sensitivity to the shapes of the initial contour. Figure A.2 records part of different initial contours in different shapes (Figure A.2(a)) and their JSC values (Figure A.2(b)) of final contours. The JSC values slightly change and approximately approach to 0.990. Our method shows the expected robustness of various types of contour shapes. The experimental results prove that our method is extremely robust to the initial contour, and the difference of the initial contour will not affect the segmentation performance.

5 CONCLUSION

We have proposed a LSM that utilises joint estimation to improve the segmentation accuracy of intensity inhomogeneous images. The local statistical intensity information and the global intensity average are jointly incorporated to estimate the original image. We introduce a weighting internal term containing gradient information as a regularising term. A specific additive operator splitting algorithm is designed for calculation. Although our method is highly adaptable and achieves substantial performance, it remains limitations in the following aspects. On one hand, the proposed method considers the balance between global features and local features, which may lead to a reduction in the accuracy of tiny object segmentation. On the other hand, due to the existence of edge-based weighting factor in the energy functional and the discretisation of the derivative in the numerical calculation process, it will inevitably increase the computing cost. In future research, we will focus on a more efficient way to extract the features of tiny objects.

ACKNOWLEDGMENTS

The authors want to thank all of the people and organisations who help them a lot! This work was supported by the National Natural Science Foundation of China under Grants (No. 61703157, No. 61902336); Natural Science Foundation of Hunan Province, China (No. 2020JJ6063, No. 2019JJ50592); Hunan province science and technology project funds (2018TP1036); CERNET Innovation Project (NGII20160310).

REFERENCES

1. Aslam, M., et al.: On the application of automated machine vision for leather defect inspection and grading: A survey. IEEE Access 7, 176065–176086 (2019)
2. Huang, C., Hong, J., Huang, E.: Developing a machine vision inspection system for electronics failure analysis. IEEE Trans. Components, Packaging Manufacturing Technol. 9(9), 1912–1925 (2019)
3. Liu, Z., et al.: A novel stereo vision measurement system using both line scan camera and frame camera. IEEE Trans. Instrumentation Measurement 68(10), 3563–3575 (2019)
4. Sun, J., et al.: An effective method of weld defect detection and classification based on machine vision. IEEE Trans. Ind. Informatics 15(12), 6322–6333 (2019)
5. Zhao, H., Dai, R., Xiao, C.: A machine vision system for stacked substrates counting with a robust stripe detection algorithm. IEEE Trans. Sys., Man, and Cybernetics: Systems 49(11), 2352–2361 (2019)
6. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Machine Intell. 39(12), 2481–2495 (2017)
7. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention, pp. 234–241. Springer, Berlin (2015)
8. Zhao, H., et al.: Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, 2017, pp. 2881–2890
9. Chen, L.C., et al.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Machine Intell. 40(4), 834–848 (2017)
10. He, K., et al.: Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969
11. Qin, X., et al.: U2-net: Going deeper with nested u-structure for salient object detection. Pattern Recognition 106, 107404 (2020)
12. Minaee, S., et al.: Image segmentation using deep learning: A survey. arXiv preprint arXiv:200105566 (2020)
13. LeCun, Y.: 1.1 deep learning hardware: Past, present, and future. In: 2019 IEEE International Solid-State Circuits Conference-(ISSCC), San Francisco, CA, 2019, pp. 12–19 (2019)
14. Chen, K., et al.: Hybrid task cascade for instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4974–4983 (2019)
15. Hatamizadeh, A., Sengupta, D., Terzopoulos, D.: End-to-end deep convolutional active contours for image segmentation. arXiv preprint arXiv:190913359 (2019)
16. Ji, Z., et al.: Active contours driven by local likelihood image fitting energy for image segmentation. Info. Sci. 301, 285–304 (2015)
17. Li, C., et al.: Implicit active contours driven by local binary fitting energy. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, 2007, pp. 1–7
18. Liu, L., et al.: Active contour driven by multi-scale local binary fitting and kullback-leibler divergence for image segmentation. Multimedia Tools Appl. 76(7), 10149–10168 (2017)
19. Liu, Y., et al.: A binary level set variational model with l1 data term for image segmentation. Signal Process. 155, 193–201 (2019)
APPENDIX A

Denote area term $\text{Area}(\text{inside}(\psi))$ as $\gamma$, length term $\text{Length}(\psi)$ as $\nu$. In LSM, the task is to minimise energy term made up by area term $\gamma$ and length term $\nu$, as given by: $\min \gamma = \nu + \gamma$. According to the relationship between length term and area term in [44][47]:

$$\text{Area}(\text{inside}(\psi)) \leq c(\text{Length}(\psi))^N/(N-1), \quad (A.1)$$

where $N$ denotes the arbitrary dimension of the level set contour. That is, $\gamma \leq \epsilon \cdot \nu^N/N$, substituting the inequality into formula we can get a new expression:

$$\gamma \leq \epsilon \cdot \nu^N/N + \gamma. \quad (A.2)$$

As formula $\epsilon \cdot \nu^N/N + \gamma$ is monotonically increasing, to find the minimisation of $\gamma$ is to solve corresponding $\nu$. Thus, it is enough
to solely use regularising length term which can achieve the same benefit as both area and length term.

APPENDIX B

Denote $N_i$ as constant (i.e. $C_i$, $f_i$, $d_i$) to be calculated, the optimal constant can be chosen to minimise the energy functional. Get the partial derivative of energy functional with respect to $N_i$ and set it to 0, as: \[
\frac{\partial F}{\partial N_i} = 0,
\] solve the equation and the optimal $N_i$ can be obtained. As the partial derivatives terms independent of $N_i$ are equal to 0, we consider only related energy functional terms linked to $N_i$; the optimal $C_i$ in Equation (16) is to minimise the equivalent of CV model [44]; the optimal $f_i$ in Equation (16) is to minimise the equivalent of LBF model as Equation (5); the optimal $d_i$ is to minimise the last two terms of Equations 16 as Equation (B.1) shows:

\[
E(\phi) = \omega \int_{\Omega} (I(x) - I_0(x) - d_1)^2 H(\phi) dx
+ (1 - \omega) \int_{\Omega} (I(x) - I_0(x) - d_2)^2 (1 - H(\phi)) dx.
\]

(B.1)