

A Reformulated Convex and Selective Variational Image Segmentation Model and its Fast Multilevel Algorithm*

Abdul K. Jumaat[†] and Ke Chen[‡]

Abstract

Selective image segmentation is the task of extracting one object of interest among many others in an image based on minimal user input. Two-phase segmentation models cannot guarantee to locate this object, while multiphase models are more likely to classify this object with another features in the image. Several selective models were proposed recently and they would find local minimizers (sensitive to initialization) because non-convex minimization functionals are involved. Recently, Spencer-Chen (CMS 2015) has successfully proposed a convex selective variational image segmentation model (named CDSS), allowing a global minimizer to be found independently of initialisation. However, their algorithm is sensitive to the regularization parameter μ and the area parameter θ due to nonlinearity in the functional and additionally it is only effective for images of moderate size. In order to process images of large size associated with high resolution, urgent need exists in developing fast iterative solvers. In this paper, a stabilized variant of CDSS model through primal-dual formulation is proposed and an optimization based multilevel algorithm for the new model is introduced. Numerical results show that the new model is less sensitive to parameter μ and θ compared to the original CDSS model and the multilevel algorithm produces quality segmentation in optimal computational time.

AMS subject classifications: 62H35, 65N22, 65N55, 74G65, 74G75

Key words: Active contours, image segmentation, level sets, multilevel, optimization methods, energy minimization

1 Introduction

22 Image segmentation is a fundamental task in image processing aiming to obtain meaningful
23 partitions of an input image into a finite number of disjoint homogeneous regions. Segmentation
24 models can be classified into two categories, namely, edge based and region based models; other
25 models may mix these categories. Edge based models refer to the models that are able to
26 drive the contours towards image edges by influence of an edge detector function. The snake
27 algorithm proposed by Kass et al. [33] was the first edge based variational model for image
28 segmentation. Further improvement on the algorithm with geodesic active contours and the
29 level-set formulation led to effective models [14, 49]. Region-based segmentation techniques try
30 to separate all pixels of an object from its background pixels based on the intensity and hence
31 find image edges between regions satisfying different homogeneity criteria. Examples of region-
32 based techniques are region growing [30, 9], watershed algorithm [30, 10], thresholding [30, 53],
33 and fuzzy clustering [50]. The most celebrated (region-based) variational model for the images
34 (with and without noise) is the Mumford-Shah [43] model, reconstructing the segmented image
35 as a piecewise smooth intensity function. Since the model cannot be implemented directly and
36 easily, the Mumford-Shah general model [43] was often approximated. The Chan-Vese (CV)

*Center for Mathematical Imaging Techniques and Department of Mathematical Sciences, University of Liverpool, United Kingdom. Web: <http://www.liv.ac.uk/cmit>

[†]Email: abdulkj@liverpool.ac.uk

[‡]Corresponding author's email: k.chen@liverpool.ac.uk

37 [21] model is simplified and reduced from [43], without approximation. The simplification is to
 38 replace the piecewise smooth function by a piecewise constant function (of two constants c_1 , c_2
 39 or more) and, in the case of two phases, the piecewise constant function divides an image into
 40 the foreground and the background. A new variant of the CV model [21] has been proposed
 41 by [8] by taking the Euler’s elastica as the regularization of segmentation contour that can
 42 yield to convex contours. Another interesting model named second order Mumford-Shah total
 43 generalized variation was developed by [26] for simultaneously performs image denoising and
 44 segmentation.

45 The segmentation models described above are for global segmentation due to the fact that
 46 all features or objects in an image are to be segmented (though identifying all objects is not
 47 guaranteed due to non-convexity). Selective image segmentation aims to extract one object
 48 of interest in an image based on some additional information of geometric constraints [28,
 49 47, 52]. This task cannot be achieved by global segmentation. Some effective models are
 50 Badshah-Chen [7] and Rada-Chen [47] which used a mixed edge based and region based ideas,
 51 and area constraints. Both models are non-convex. A non-convex selective variational image
 52 segmentation model, though effective in capturing a local minimiser, is sensitive to initialisation
 53 where the segmentation result relies heavily on user input.

54 While the above selective segmentation models are formulated based on geometric con-
 55 straints in [28, 29], there are another way of defining the geometric constraints that can be
 56 found in [41] where geometric points outside and inside a targeted object are given. Their mo-
 57 del make use the Split Bregman method to speed up convergence. Although our paper based on
 58 geometric constraint defining in [28, 29], later, we shall compare our work with [41]. We called
 59 their model as NCZZ model.

60 In 2015, Spencer-Chen [52, 51] has successfully designed a Convex Distance Selective Seg-
 61 mentation model (named as CDSS). This variational model allows a global minimiser to be
 62 found independently of initialisation, given knowledge of c_1 , c_2 . The CDSS model [52] is chal-
 63 lenging to solve due to its penalty function $\nu(u)$ being highly nonlinear. Consequently, the
 64 standard addition operator splitting method (AOS) is not adequate. An enhanced version of
 65 the AOS scheme was proposed in [52] by taking the approximation of $\nu'(u)$ which based on its
 66 linear part [52, 51]. Another factor that affects the [52] model is how to choose the combination
 67 values of the regularization parameters μ and θ (other parameters can be fixed as suggested by
 68 [52, 51]). For a simple (synthetic) image, it is easy to get a suitable combination of parameter μ
 69 and θ which gives a good segmentation result. However, for other real life images, it is not trivial
 70 to determine a suitable combination of μ and θ simultaneously; our experiments show that high
 71 segmentation accuracy is given by the model in a small range of μ and θ and consequently the
 72 model is not ready for general use. Of course, it is known that an AOS method is not designed
 73 for processing large images.

74 We remark that the most recent, convex, selective, variational image segmentation model
 75 was by Liu *et al.* [35] in 2018. This work is based on [7, 12, 47]. We named their model as
 76 the CMT model. Although this paper is based on [52, 51], we shall compare our work with the
 77 CMT model [35] later.

78 Both the fast solvers multilevel and multigrid methods are developed using the idea of
 79 hierarchy of discretization. However, multilevel method is based on discretize-optimize scheme
 80 (algebraic) where the minimization of a variational problem is solved directly without using par-
 81 tial differential equation (PDE). In contrast, a multigrid method is based on optimize-discretize
 82 scheme (geometric) where it solves a PDE numerically. The two methods are inter-connected
 83 since both can have geometric interpretations and use similar inter-level information transfers
 84 [32].

85 Multigrid methods have been used to solve a few variational image segmentation models in
 86 the level set formulation. For geodesic active contours models, linear multigrid methods are
 87 developed [34, 45, 46]. In 2008, Badshah and Chen [5] has successfully implemented a nonlinear

88 multigrid method to solve an elliptical partial differential equation. In 2009, Badshah and Chen
 89 [6] have also developed two nonlinear multigrid algorithms for variational multiphase image
 90 segmentation. All these multigrid methods mentioned above are based on an optimize-discretize
 91 scheme where a multigrid method is used to solve the resulting Euler Lagrange partial differential
 92 equation (PDE) derived from the variational problem. While the practical performance of the
 93 latter methods (closer to this work) is good, however, the multigrid convergence is not achieved
 94 due to smoothers having a bad smoothing rate (and non-smooth coefficients with jumps near
 95 edges that separate segmented domains). Therefore the above nonlinear multigrid methods
 96 behave like the cascadic multigrids [42] where only one multigrid cycle is applied.

97 An optimization based multilevel method is based on a discretize-optimize scheme where
 98 minimization is solved directly (without using PDEs). The idea has been applied to image
 99 denoising and deblurring problems [16, 17, 18]. However, the method is found to get stuck to
 100 local minima due to non-differentiability of the energy functional. To overcome that situation,
 101 Chan and Chen [16] have proposed the “patch detection” idea in the formulation of the multilevel
 102 method which is efficient for image denoising problems. However, as image size increases, the
 103 method can be slow because of the patch detection idea searches the entire image for the possible
 104 patch size on the finest level after each multilevel cycle [32].

105 This paper investigates both the robust modeling and fast solution issues by making two con-
 106 tributions. Firstly, we propose a better model than CDSS. In looking for possible improvement
 107 on the selective model CDSS, we are inspired by several works [11, 3, 4, 15, 20, 13] on non-
 108 selective segmentation. The key idea that we will employ in our new model is the primal-dual
 109 formulation which allows us to “ignore” the penalty function $\nu(u)$, otherwise creating problems
 110 of parameter sensitivity. We remark that similar use of the primal-dual idea can be found in D.
 111 Chen et al. [22] to solve a variant of Mumford-Shah model which handles the segmentation of
 112 medical images with intensity inhomogeneities and also in Moreno et al. [40] for solving a four
 113 phase model for segmentation of brain MRI images by active contours. Secondly, we propose a
 114 fast optimization based multilevel method for solving the new model, which is applicable to the
 115 original CDSS [52], in order to achieve fast convergence especially for images with large size. We
 116 will consider the differentiable form of variational image segmentation models and develop the
 117 multilevel algorithm for the resulting models without using a “patch detection” idea. We are
 118 not aware of similar work done for segmentation models in the variational convex formulation.

119 The rest of the paper is organized in the following way. In Section 2, we first briefly review
 120 the non-convex variant of the Spencer-Chen CDSS model [52]. This model gives foundation for
 121 the CDSS. In Section 3, we give our new primal-dual formulation of the CDSS model and in
 122 Section 4 present the optimization based multilevel algorithm. We proposed a new variant of
 123 the multilevel algorithm in Section 5 and discuss their convergence in Section 6. In Section 7
 124 we give some experimental results before concluding in Section 8.

125 2 Review of existing variational selective segmentation models

126 As discussed, there exist many variational segmentation models in the literature on global
 127 segmentation and few on selective image segmentation models. For the latter, we will review two
 128 segmentation models below that are directly related to this work. We first review a nonconvex
 129 selective segmentation model called the Distance Selective Segmentation [52]. Then, we discuss
 130 the convex version of DSS called the Convex Distance Selective Segmentation model [52] before
 131 we introduce a new CDSS model based on primal-dual formulation and address the fast solution
 132 issue in these models.

133 Assume that an image $z = z(x, y)$ comprises of two regions of approximately piecewise
 134 constant intensities of distinct values (unknown) c_1 and c_2 , separated by some (unknown) curve
 135 or contour Γ . Let the object to be detected be represented by the region Ω_1 with the value c_1
 136 inside the curve Γ whereas outside Γ , in $\Omega_2 = \Omega \setminus \Omega_1$, the intensity of z is approximated with

137 value c_2 . In a level set formulation, the unknown curve Γ is represented by the zero level set of
138 the Lipschitz function such that

$$\begin{aligned}\Gamma &= \{(x, y) \in \Omega : \phi(x, y) = 0\}, \\ \Omega_1 &= \text{inside}(\Gamma) = \{(x, y) \in \Omega : \phi(x, y) > 0\}, \\ \Omega_2 &= \text{outside}(\Gamma) = \{(x, y) \in \Omega : \phi(x, y) < 0\}.\end{aligned}$$

Let n_1 geometric constraints be given by a marker set

$$A = \{w_i = (x_i^*, y_i^*) \in \Omega, 1 \leq i \leq n_1\} \subset \Omega$$

139 where each point is near the object boundary Γ , not necessarily on it [47, 54]. The selective
140 segmentation idea tries to detect the boundary of a single object among all homogeneity intensity
141 objects in Ω close to A ; here $n_1 (\geq 3)$. The geometrical points in A define an initial polygonal
142 contour and guide its evolution towards Γ [54].

143 It should be remarked that applying a global segmentation model first and selecting an
144 object next amount provide an alternative to selective segmentation. However this approach
145 would require a secondary binary segmentation and is not reliable because the first round of
146 segmentation cannot guarantee to isolate the interested object often due to non-convexity of
147 models.

148 2.1 Distance Selective Segmentation model

149 The Distance Selective Segmentation (DSS) model [52] was proposed by Spencer and Chen
150 [52] in 2015. The formulation is based on the special case of the piecewise constant Mumford-
151 Shah functional [43] where it is restricted to only two phase (i.e. constants), representing the
152 foreground and the background of the given image $z(x, y)$.

153 Using the set A , construct a polygon Q that connects up the markers. Denote the function
154 $P_d(x, y)$ as the Euclidean distance of each point $(x, y) \in \Omega$ from its nearest point $(x_p, y_p) \in Q$:

$$P_d(x, y) = \sqrt{(x - x_p)^2 + (y - y_p)^2} = \min_{q \in Q} \|(x, y) - (x_q, y_q)\|$$

155 and denote the regularized versions of a Heaviside function by

$$H_\varepsilon(\phi(x, y)) = \frac{1}{2} \left(1 + \frac{2}{\pi} \arctan \left(\frac{\phi}{\varepsilon} \right) \right).$$

156 Then the DSS in a level set formulation is to minimize a cost function defined as follows

$$\begin{aligned}\min_{\phi, c_1, c_2} D(\phi, c_1, c_2) &= \mu \int_{\Omega} g(|\nabla z|) |\nabla H_\varepsilon(\phi)| d\Omega + \int_{\Omega} H_\varepsilon(\phi) (z - c_1)^2 d\Omega \\ &+ \int_{\Omega} (1 - H_\varepsilon(\phi)) (z - c_2)^2 d\Omega + \theta \int_{\Omega} H_\varepsilon(\phi) P_d d\Omega\end{aligned}\quad (1)$$

157 where μ and θ are nonnegative parameters. In this model $g(s) = \frac{1}{1 + \gamma s^2}$ is an edge detector
158 function which helps to stop the evolving curve on the edge of the objects in an image. The
159 strength of detection is adjusted by parameter γ . The addition of new distance fitting term is
160 weighted by the area parameter θ . Here, if the parameter θ is too strong the final result will
161 just be the polygon P which is undesirable.

162 2.2 Convex Distance Selective Segmentation model

163 The above model from (1) was relaxed to obtain a constrained Convex Distance Selective Seg-
164 mentation (CDSS) model [52]. This was to make sure that the initialisation can be flexible.
165 The CDSS was obtained by relaxing $H_\varepsilon \rightarrow u \in [0, 1]$ to give:

$$\min_{0 \leq u \leq 1} CDSS(u, c_1, c_2) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \int_{\Omega} ru d\Omega + \theta \int_{\Omega} P_d u d\Omega \quad (2)$$

166 and further an unconstrained minimization problem:

$$\min_u CDSS(u, c_1, c_2) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \int_{\Omega} ru d\Omega + \theta \int_{\Omega} P_d u d\Omega + \alpha \int_{\Omega} \nu(u) d\Omega \quad (3)$$

167 where $r = (c_1 - z)^2 - (c_2 - z)^2$ and $|\nabla u|_g = g(|\nabla z|)|\nabla u|$, $\nu(u) = \max\{0, 2|u - \frac{1}{2}| - 1\}$ is
168 an exact (non-smooth) penalty term, provided that $\alpha > \frac{1}{2}\|r + \theta P_d\|_{L^\infty}$ (see also [19]). For fixed
169 c_1, c_2, μ, θ , and $\kappa \in [0, 1]$, the minimizer u of (2) is guaranteed to be a global minimizer defining
170 the object by $\Sigma = \{(x, y) : u(x, y) \geq \kappa\}$ [52, 19, 11]. The parameter κ is a threshold value and
171 usually $\kappa = 0.5$.

172 In order to compute the associated Euler Lagrange equation for u they introduce the regu-
173 larized version of $\nu(u)$:

$$\nu(u) = \left[\sqrt{(2u-1)^2 + \varepsilon} - 1 \right] H(\sqrt{(2u-1)^2 + \varepsilon} - 1), \quad H(x) = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{x}{\varepsilon}\right).$$

174 Consequently, the Euler Lagrange equation for u in equation (3) is the following

$$\mu \nabla \left(g \frac{\nabla u}{|\nabla u|} \right) + f = 0, \quad \text{in } \Omega, \quad \frac{\partial u}{\partial \vec{n}} = 0, \quad \text{on } \partial\Omega \quad (4)$$

175 where $f = -r - \theta P_d - \alpha \nu'(u)$. When u is fixed, the intensity values c_1, c_2 are updated by

$$c_1(u) = \frac{\int_{\Omega} uz d\Omega}{\int_{\Omega} u d\Omega}, \quad c_2(u) = \frac{\int_{\Omega} (1-u)z d\Omega}{\int_{\Omega} (1-u) d\Omega}.$$

176 Notice that the nonlinear coefficient of equation (4) may have a zero denominator where the
177 equation is not defined. A commonly adopted idea to deal with this is to introduce a positive
178 parameter β to (4), so the new Euler Lagrange equation becomes

$$\mu \nabla \left(g \frac{\nabla u}{\sqrt{|\nabla u|^2 + \beta}} \right) + f = 0, \quad \text{in } \Omega; \quad \frac{\partial u}{\partial \vec{n}} = 0, \quad \text{on } \partial\Omega$$

179 which corresponds to minimize the following differentiable form of (3)

$$\min_u CDSS(u, c_1, c_2) = \mu \int_{\Omega} g \sqrt{|\nabla u|^2 + \beta} d\Omega + \int_{\Omega} ru d\Omega + \theta \int_{\Omega} P_d u d\Omega + \alpha \int_{\Omega} \nu(u) d\Omega. \quad (5)$$

180 According to [52, 51], the standard AOS which generally assumes f is not dependent on u
181 is not adequate to solve the model. This mainly because the term $\nu'(u)$ in f does depend on u ,
182 which can lead to stability restriction on time step size t . Moreover, the shape of $\nu'(u)$ means
183 that changes in f between iterations are problematic near $u = 0$ and $u = 1$, as small changes in
184 u produce large changes in f . In order to tackle the problem, they proposed a modified version
185 of AOS algorithm to solve the model by taking the approximation of $\nu'(u)$ which based on its
186 linear part.

187 A successful segmentation result can be obtained depending on suitable combination of
188 parameter μ, θ and the set of marker points defined by a user. For a simple image such as
189 synthetic images, this task of parameters selection is easy and one can get a good segmentation
190 result. However, for real life images, it is non-trivial to determine a suitable combination of
191 parameters μ and θ . It becomes more challenging if a model is sensitive to μ and θ where only
192 a small range of the values work to give high segmentation quality. Hence, a more robust model
193 that is less dependent on the parameters needs to be developed. In addition, to process images
194 of large size, fast iterative solvers need to be developed as well. This paper is motivated by
195 these two problems.

196 We refer to the CDSS model solved by the modified AOS as **SC0**.

197 3 A reformulated CDSS model

198 We now present our work on a reformulation of the CDSS model in the primal-dual framework
199 which allows us to “ignore” the penalty function $\nu(u)$, otherwise creating problems of parameter
200 sensitivity. We remark that similar use of the primal-dual idea can be found in [22] and [40]. To
201 see more background of this framework, refer to the convex regularization approach by Bresson
202 et al. [11], Chambolle [15], and others [3, 4, 20, 13].

203 Our starting point is to rewrite (3) as follows:

$$\min_{u,w} J(u, w) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \int_{\Omega} r w d\Omega + \theta \int_{\Omega} P_d w d\Omega + \alpha \int_{\Omega} \nu(w) d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega \quad (6)$$

204 where w is the new and dual variable, the right-most term enforces $w \approx u$ for sufficiently small
205 $\rho > 0$ and $|\nabla u|_g = g(|\nabla z|)|\nabla u|$. One can observe that if $w = u$, the dual formulation is reduced
206 to the original CDSS model [52].

207 After introducing the term $(u - w)^2$, it is important to note that convexity still holds with
208 respect to u and w (otherwise finding the global minimum cannot be guaranteed). This can be
209 shown below. Write the functional (6) as the sum of two terms:

$$J(u, w) = S(u, w) + Q(u, w), \quad S(u, w) = \int_{\Omega} \frac{1}{2\rho} (u - w)^2 d\Omega, \quad TV_g(u) = \int_{\Omega} |\nabla u|_g d\Omega$$

$$Q(u, w) = TV_g(u) + \int_{\Omega} (r + \theta P_d) w d\Omega + \alpha \int_{\Omega} \nu(w) d\Omega.$$

210 For the functional $Q(u, w)$, we can show that the weighted total variation term $TV_g(u)$ is convex
211 below. The remaining two terms (depending on w only) are known to be convex from [52, 51].
212 By definition of convex functions, showing that the weighted total variation is a convex can be
213 done directly. Let $u_1 \neq u_2$ be two functions and $\varphi \in [0, 1]$. Then

$$\begin{aligned} TV_g(\varphi u_1 + (1 - \varphi) u_2) &= \int_{\Omega} |\nabla(\varphi u_1 + (1 - \varphi) u_2)|_g d\Omega \\ &= \int_{\Omega} |\varphi \nabla u_1 + (1 - \varphi) \nabla u_2|_g d\Omega \\ &\leq \varphi \int_{\Omega} |\nabla u_1|_g d\Omega + (1 - \varphi) \int_{\Omega} |\nabla u_2|_g d\Omega \\ &= \varphi TV_g(u_1) + (1 - \varphi) TV_g(u_2). \end{aligned}$$

214 Similarly, for the functional $S(u, w)$, let $u, w : \Omega \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}$ and $u_1 \neq u_2 \neq u_3 \neq u_4$. Then

$$\begin{aligned} S[\varphi(u_1, u_2) + (1 - \varphi)(u_3, u_4)] &= S[\varphi u_1 + (1 - \varphi) u_3, \varphi u_2 + (1 - \varphi) u_4] \\ &= \int_{\Omega} [\varphi u_1 + (1 - \varphi) u_3 - \varphi u_2 - (1 - \varphi) u_4]^2 d\Omega \\ &= \int_{\Omega} [\varphi(u_1 - u_2) + (1 - \varphi)(u_3 - u_4)]^2 d\Omega \\ &\leq \varphi \int_{\Omega} (u_1 - u_2)^2 d\Omega + (1 - \varphi) \int_{\Omega} (u_3 - u_4)^2 d\Omega \\ &= \varphi S(u_1, u_2) + (1 - \varphi) S(u_3, u_4). \end{aligned}$$

215 Alternatively, the *Hessian* $[(u - w)^2] = \begin{pmatrix} 2 & -2 \\ -2 & 2 \end{pmatrix}$. Clearly the principal minors are $\Delta_1 =$
216 2 , $\Delta_2 = 0$ which indicates that the *Hessian* $[(u - w)^2]$ is positive semidefinite and so $S(u, w)$
217 is convex.

218 As the sum of two convex functions Q, S is also convex, thus $J(u, w)$ is convex.

219 Using the property that J is differentiable, consequently, the unique minimizer can be com-
220 puted by minimizing J with respect to u and w separately, iterating the process until convergence
221 [11, 15]. Thus, the following minimization problems are considered:

222 i). when w is given: $\min_u J_1(u, w) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega;$

223 ii). when u is given: $\min_w J_2(u, w) = \int_{\Omega} rwd\Omega + \theta \int_{\Omega} P_d w d\Omega + \alpha \int_{\Omega} \nu(w) d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega.$

224 Next consider how to simplify J_2 further and drop its α term. To this end, we make use of
225 the following proposition:

226 **Proposition 1** *The solution of $\min_w J_2$ is given by:*

$$w = \min \{ \max \{ u(x) - \rho r - \rho \theta P_d, 0 \}, 1 \}. \quad (7)$$

227

228 **Proof:** Assume that α has been chosen large enough compared to $\|f\|_{L^\infty}$ so that the exact
229 penalty formulation holds. We now consider the w -minimization of the form

230 $\min_w \int_{\Omega} \left(\alpha \nu(w) + \frac{1}{2\rho} (u - w)^2 + wF(x) \right) d\Omega$, where the function F is independent of w . We use
231 the claim made by [11].

232 **Claim** [11]: If $u(x) \in [0, 1]$ for all x , then so is $w(x)$ after the w -minimization. Conversely, if
233 $w(x) \in [0, 1]$ for all x , then so is $u(x)$ after the u -minimization.

234 This claim allows us to “ignore” the $\nu(w)$ terms: on one hand, its presence in the energy is
235 equivalent to cutting off $w(x)$ at 0 and 1. On the other hand, if $w(x) \in [0, 1]$, then the above
236 w -minimization can be written in this equivalence form: $\min_{w \in (0,1)} \int_{\Omega} \left(\frac{1}{2\rho} (u - w)^2 + wF(x) \right) d\Omega.$

237 Consequently, the point-wise optimal $w(x)$ is found as $\frac{1}{\rho} (u - w) = F(x) \Rightarrow w = u - \rho F(x).$

238 Thus the w -minimization can be achieved through the following update:

239 $w = \min \{ \max \{ u(x) - \rho F(x), 0 \}, 1 \}.$ For $\min_w J_2$, let $F(x) = r + \theta P_d$. Hence, we deduce the
240 result for w . \square

241 Therefore, our new model is defined as

$$\min_{u, w \in (0,1)} J(u, w) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \int_{\Omega} r w d\Omega + \theta \int_{\Omega} P_d w d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega.$$

242 In alternating minimization form, the new formulation is equivalent to solve the following

$$\min_u J_1(u, w) = \mu \int_{\Omega} |\nabla u|_g d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega, \quad (8)$$

$$\min_{w \in (0,1)} J_2(u, w) = \int_{\Omega} r w d\Omega + \theta \int_{\Omega} P_d w d\Omega + \frac{1}{2\rho} \int_{\Omega} (u - w)^2 d\Omega. \quad (9)$$

243 Notice that the term $\nu(w)$ is dropped in (9) and the explicit solution is given in (7) that is
244 hopefully the new resulting model becomes less sensitive to parameter’s choice. Now it only
245 remains to discuss how to solve (8).

246 4 An optimization based multilevel algorithm

247 This section presents our multilevel formulation for two convex models: first the CDSS model
248 (5) (for later use in comparisons) and then our newly proposed primal-dual model in (8)-(9).

249 For simplicity, we shall assume $n = 2^L$ for a given image z of size $n \times n$. The standard
250 coarsening defines $L + 1$ levels: $k = 1$ (finest), $2, \dots, L, L + 1$ (coarsest) such that level k has
251 $\tau_k \times \tau_k$ “superpixels” with each “superpixels” having pixels $b_k \times b_k$ where $\tau_k = n/2^{k-1}$ and
252 $b_k = 2^{k-1}$. Figure 2 (a-e) show the case $L = 4, n = 2^4$ for an 16×16 image with 5 levels: level
253 1 has each pixel of the default size of 1×1 while the coarsest level 5 has a single superpixel
254 of size 16×16 . If $n \neq 2^L$, the multilevel method can still be developed with some coarse level
255 superpixels of square shapes and the rest of rectangular shapes.

256 **4.1 A multilevel algorithm for CDSS**

257 Our goal is to solve (5) using a multilevel method in discretize-optimize scheme without ap-
258 proximation of $\nu'(u)$. The finite difference method is used to discretize (5) as done in related
259 works [13, 16]. The discretized version of (5) is given by

$$\begin{aligned} \min_u CDSS(u, c_1, c_2) &\equiv \min_u CDSS^a(u_{1,1}, u_{2,1}, \dots, u_{i-1,j}, u_{i,j}, u_{i+1,j}, \dots, u_{n,n}, c_1, c_2) \\ &= \bar{\mu} \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} g_{i,j} \sqrt{(u_{i,j} - u_{i,j+1})^2 + (u_{i,j} - u_{i+1,j})^2} + \beta \end{aligned} \quad (10)$$

260

$$+ \sum_{i=1}^n \sum_{j=1}^n \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) u_{i,j} + \theta \sum_{i=1}^n \sum_{j=1}^n P_{d_{i,j}} u_{i,j} + \alpha \sum_{i=1}^n \sum_{j=1}^n \nu_{i,j}$$

261 where $\bar{\mu} = \frac{\mu}{h}$, $c_1 = \frac{\sum_{i=1}^n \sum_{j=1}^n z_{i,j} u_{i,j}}{\sum_{i=1}^n \sum_{j=1}^n u_{i,j}}$, $c_2 = \frac{\sum_{i=1}^n \sum_{j=1}^n z_{i,j} (1 - u_{i,j})}{\sum_{i=1}^n \sum_{j=1}^n (1 - u_{i,j})}$,

$$h = \frac{1}{(n-1)}, \quad \nu_{i,j} = \left[\sqrt{(2u_{i,j} - 1)^2 + \varepsilon} - 1 \right] \left(\frac{1}{2} + \frac{1}{\pi} \arctan \frac{\sqrt{(2u_{i,j} - 1)^2 + \varepsilon} - 1}{\varepsilon} \right),$$

262

$$g_{i,j} = (x_i, y_j) \quad \text{and} \quad P_{d_{i,j}} = (x_i, y_j).$$

263 Here u denotes a row vector.

264 As a prelude to multilevel methods, minimize (10) by a coordinate descent method (also
265 known as relaxation algorithm) on the finest level 1:

266 Given $u^{(m)} = \left(u_{i,j}^{(m)} \right)$ with $m = 0$;

$$\text{Solve } u_{i,j}^{(m)} = \arg \min_{u_{i,j} \in \mathbb{R}} CDSS^{loc}(u_{i,j}, c_1, c_2) \text{ for } i, j = 1, 2, \dots, n; \quad (11)$$

267 Set $u_{i,j}^{(m+1)} = \left(u_{i,j}^{(m)} \right)$ and repeat the above steps with $m = m + 1$ until stopped.

268 Here equation (11) is simply obtained by expanding and simplifying the main model in (10)

269 i.e.

$$\begin{aligned} &CDSS^{loc}(u_{i,j}, c_1, c_2) \\ &\equiv CDSS^a \left(u_{1,1}^{(m-1)}, u_{2,1}^{(m-1)}, \dots, u_{i-1,j}^{(m-1)}, u_{i,j}, u_{i+1,j}^{(m-1)}, \dots, u_{m,n}^{(m-1)}, c_1, c_2 \right) - CDSS^{(m-1)} \\ &= \bar{\mu} \left[g_{i,j} \sqrt{(u_{i,j} - u_{i+1,j}^{(m)})^2 + (u_{i,j} - u_{i,j+1}^{(m)})^2} + \beta \right. \\ &\quad + g_{i-1,j} \sqrt{(u_{i,j} - u_{i-1,j}^{(m)})^2 + (u_{i-1,j}^{(m)} - u_{i-1,j+1}^{(m)})^2} + \beta \\ &\quad \left. + g_{i,j-1} \sqrt{(u_{i,j} - u_{i,j-1}^{(m)})^2 + (u_{i,j-1}^{(m)} - u_{i+1,j-1}^{(m)})^2} + \beta \right] \\ &\quad + u_{i,j} \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) + \theta P_{d_{i,j}} u_{i,j} + \alpha (\nu_{i,j}) \end{aligned}$$

270 with Neumann's boundary condition applied where $CDSS^{(m-1)}$ denotes the sum of all terms in
271 $CDSS^a$ that do not involve $u_{i,j}$. Clearly one seems that this is a coordinate descent method.
272 It should be remarked that the formulation in (11) is based on the work in [13] and [16].

273 Using (11), we illustrate the interaction of $u_{i,j}$ with its neighboring pixels on the finest level
274 1 in Figure 1. We will use this basic structure to develop a multilevel method.

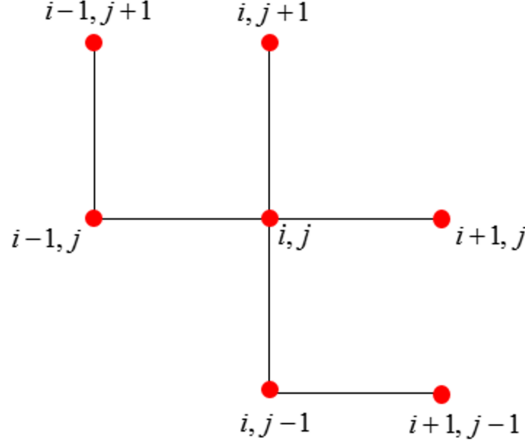


Figure 1: The interaction of $u_{i,j}$ at a central pixel (i, j) with neighboring pixels on the finest level 1. Clearly only 3 terms (pixels) are involved with $u_{i,j}$ (through regularization)

275 The Newton method is used to solve the one-dimensional problem from (11) by iterating
276 $u^{(m)} \rightarrow u \rightarrow u^{(m+1)}$:

$$\begin{aligned} & \bar{\mu}g_{i,j} \frac{2u_{i,j} - u_{i+1,j}^{(m)} - u_{i,j+1}^{(m)}}{\sqrt{(u_{i,j} - u_{i+1,j}^{(m)})^2 + (u_{i,j} - u_{i,j+1}^{(m)})^2 + \beta}} + \bar{\mu}g_{i-1,j} \frac{u_{i,j} - u_{i-1,j}^{(m)}}{\sqrt{(u_{i,j} - u_{i-1,j}^{(m)})^2 + (u_{i-1,j}^{(m)} - u_{i-1,j+1}^{(m)})^2 + \beta}} \\ & + \bar{\mu}g_{i,j-1} \frac{u_{i,j} - u_{i,j-1}^{(m)}}{\sqrt{(u_{i,j} - u_{i,j-1}^{(m)})^2 + (u_{i,j-1}^{(m)} - u_{i+1,j-1}^{(m)})^2 + \beta}} + \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) + \theta P_{d_{i,j}} + \alpha \nu_{i,j}' = 0 \end{aligned}$$

277 giving rise to the form

$$u_{i,j}^{new} = u_{i,j}^{old} - T^{old} / B^{old} \quad (12)$$

278 where

$$\begin{aligned} T^{old} &= \bar{\mu}g_{i,j} \frac{2u_{i,j}^{old} - u_{i+1,j}^{(m)} - u_{i,j+1}^{(m)}}{\sqrt{(u_{i,j}^{old} - u_{i+1,j}^{(m)})^2 + (u_{i,j}^{old} - u_{i,j+1}^{(m)})^2 + \beta}} + \bar{\mu}g_{i-1,j} \frac{u_{i,j}^{old} - u_{i-1,j}^{(m)}}{\sqrt{(u_{i,j}^{old} - u_{i-1,j}^{(m)})^2 + (u_{i-1,j}^{(m)} - u_{i-1,j+1}^{(m)})^2 + \beta}} \\ & + \bar{\mu}g_{i,j-1} \frac{u_{i,j}^{old} - u_{i,j-1}^{(m)}}{\sqrt{(u_{i,j}^{old} - u_{i,j-1}^{(m)})^2 + (u_{i,j-1}^{(m)} - u_{i+1,j-1}^{(m)})^2 + \beta}} + \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) \\ & + \theta P_{d_{i,j}} + \alpha \nu_{i,j}'^{(old)} \end{aligned}$$

279

$$\begin{aligned} B^{old} &= \bar{\mu}g_{i,j} \frac{2}{\sqrt{(u_{i,j}^{old} - u_{i+1,j}^{(m)})^2 + (u_{i,j}^{old} - u_{i,j+1}^{(m)})^2 + \beta}} - \bar{\mu}g_{i,j} \frac{(2u_{i,j}^{old} - u_{i+1,j}^{(m)} - u_{i,j+1}^{(m)})^2}{\sqrt{\left((u_{i,j}^{old} - u_{i+1,j}^{(m)})^2 + (u_{i,j}^{old} - u_{i,j+1}^{(m)})^2 + \beta \right)^{\frac{3}{2}}}} \\ & + \bar{\mu}g_{i-1,j} \frac{1}{\sqrt{(u_{i,j}^{old} - u_{i-1,j}^{(m)})^2 + (u_{i-1,j}^{(m)} - u_{i-1,j+1}^{(m)})^2 + \beta}} - \bar{\mu}g_{i-1,j} \frac{(u_{i,j}^{old} - u_{i-1,j}^{(m)})^2}{\sqrt{\left((u_{i,j}^{old} - u_{i-1,j}^{(m)})^2 + (u_{i-1,j}^{(m)} - u_{i-1,j+1}^{(m)})^2 + \beta \right)^{\frac{3}{2}}}} \\ & + \bar{\mu}g_{i,j-1} \frac{1}{\sqrt{(u_{i,j}^{old} - u_{i,j-1}^{(m)})^2 + (u_{i,j-1}^{(m)} - u_{i+1,j-1}^{(m)})^2 + \beta}} - \bar{\mu}g_{i,j-1} \frac{(u_{i,j}^{old} - u_{i,j-1}^{(m)})^2}{\sqrt{\left((u_{i,j}^{old} - u_{i,j-1}^{(m)})^2 + (u_{i,j-1}^{(m)} - u_{i+1,j-1}^{(m)})^2 + \beta \right)^{\frac{3}{2}}}} \\ & + \alpha \nu_{i,j}''^{(old)}. \end{aligned}$$

To develop a multilevel method for this coordinate descent method, we interpret solving (11) as looking for the best correction constant \hat{c} at the current approximation $u_{i,j}^{(m)}$ on level 1 (the finest level) that minimizes for c i.e.

$$\min_{u_{i,j} \in \mathbb{R}} CDSS^{loc}(u_{i,j}, c_1, c_2) = \min_{c \in \mathbb{R}} CDSS^{loc}\left(u_{i,j}^{(m)} + c, c_1, c_2\right).$$

280 Hence, we may rewrite (11) in an equivalent form:

281 *Given* $\left(u_{i,j}^{(m)}\right)$ with $m = 0$,

$$282 \text{ Solve } \hat{c} = \arg \min_{c \in \mathbb{R}} CDSS^{loc} \left(u_{i,j}^{(m)} + c, c_1, c_2\right), u_{i,j}^{(m)} = u_{i,j}^{(m)} + \hat{c} \text{ for } i, j = 1, 2, \dots, n; \quad (13)$$

283 *Set* $u_{i,j}^{(m+1)} = \left(u_{i,j}^{(m)}\right)$ and repeat the above steps with $m = m + 1$ until a prescribed
284 *stopping on* m .

285 It remains to derive the simplified formulation for each of the subproblems associated with
286 these blocks on level k e.g. the multilevel method for $k=2$ is to look for the best correction
287 constant to update each 2×2 block so that the underlying merit functional, relating to all four
288 pixels (see Fig.2(b)), achieves a local minimum. For levels $k = 1, \dots, 5$, Figure 2 illustrates the
289 multilevel partition of an image of size 16×16 pixels from (a) the finest level (level 1) until (e) the
290 coarsest level (level 5). Observe that $b_k \tau_k = n$ on level k , where τ_k is the number of boxes and b_k
291 is the block size. So from Figure 2(a), $b_1 = 1$ and $\tau_1 = n = 16$. On other levels $k = 2, 3, 4$ and 5,
292 we see that block size $b_k = 2^{k-1}$ and $\tau_k = 2^{L+1-k}$ since $n = 2^L$. Based on Figure 1, we illustrate
293 a box \odot interacting with neighboring pixels \bullet in level 3. In addition, Figure 2 (f) illustrates that
294 fact that variation by $c_{i,j}$ inside an active block only involves its boundary of precisely $4b_k - 4$
295 pixels, not all b_k^2 pixels, in that box, denoted by symbols $\triangleleft, \triangleright, \Delta, \nabla$. This is important in
296 efficient implementation.

297 With the above information, we are now ready to formulate the multilevel approach for
298 general level k . Let's set the following: $b = 2^{k-1}$, $k_1 = (i - 1)b + 1$, $k_2 = ib$, $\ell_1 = (j - 1)b + 1$,
299 $\ell_2 = jb$, and $c = (c_{i,j})$. Denoted the current \tilde{u} then, the computational stencil involving c on
300 level k can be shown as follows

$$\begin{array}{c|ccc|c}
\vdots & \vdots & \dots & \vdots & \vdots \\
\hline
\tilde{u}_{k_1-1, \ell_2+1} + c_{i-1, j+1} & \tilde{u}_{k_1, \ell_2+1} + c_{i, j+1} & \dots & \tilde{u}_{k_2, \ell_2+1} + c_{i, j+1} & \tilde{u}_{k_2+1, \ell_2+1} + c_{i+1, j+1} \\
\hline
\tilde{u}_{k_1-1, \ell_2} + c_{i-1, j} & \tilde{u}_{k_1, \ell_2} + c_{i, j} & \dots & \tilde{u}_{k_2, \ell_2} + c_{i, j} & \tilde{u}_{k_2+1, \ell_2} + c_{i+1, j} \\
\vdots & \vdots & \dots & \vdots & \vdots \\
\tilde{u}_{k_1-1, \ell_1} + c_{i-1, j} & \tilde{u}_{k_1, \ell_1} + c_{i, j} & \dots & \tilde{u}_{k_2, \ell_1} + c_{i, j} & \tilde{u}_{k_2+1, \ell_1} + c_{i+1, j} \\
\hline
\tilde{u}_{k_1-1, \ell_1-1} + c_{i-1, j-1} & \tilde{u}_{k_1, \ell_1-1} + c_{i, j-1} & \dots & \tilde{u}_{k_2, \ell_1-1} + c_{i, j-1} & \tilde{u}_{k_2+1, \ell_1-1} + c_{i+1, j-1} \\
\vdots & \vdots & \dots & \vdots & \vdots
\end{array} \quad (14)$$

301 The illustration shown above is consistent with Figure 2 (f) and the key point is that interior
302 pixels do not involve $c_{i,j}$ in the formulation's first nonlinear term. This is because the finite
303 differences are not changed at interior pixels by the same update as in

$$\begin{aligned}
& \sqrt{(\tilde{u}_{k,l} + c_{i,j} - \tilde{u}_{k+1,l} - c_{i,j})^2 + (\tilde{u}_{k,l} + c_{i,j} - \tilde{u}_{k,l+1} - c_{i,j})^2} + \beta \\
& = \sqrt{(\tilde{u}_{k,l} - \tilde{u}_{k+1,l})^2 + (\tilde{u}_{k,l} - \tilde{u}_{k,l+1})^2} + \beta.
\end{aligned}$$

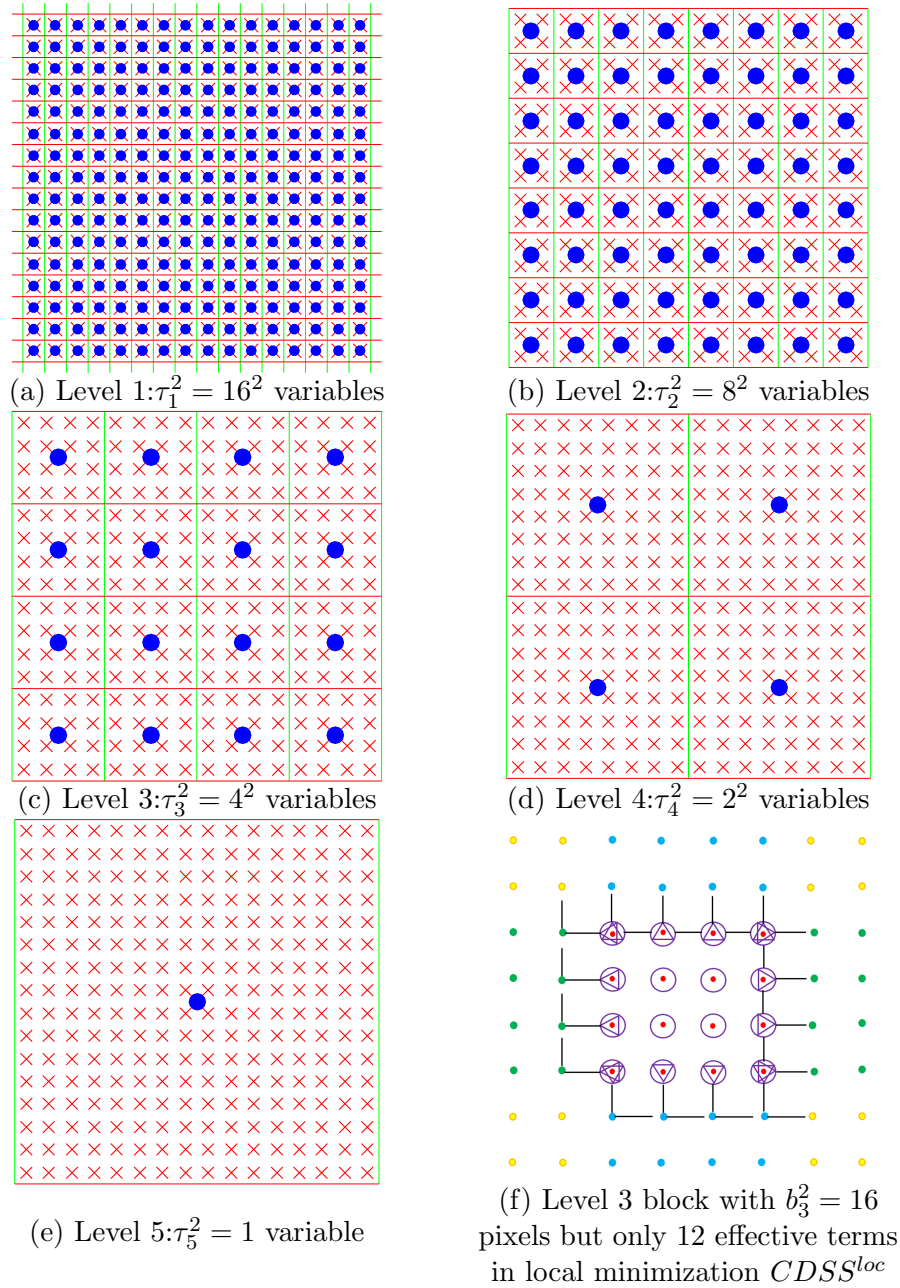


Figure 2: Illustration of partition (a)-(e). The red “ \times ” shows image pixels, while blue \bullet illustrates the variable c . (f) shows the difference of inner and boundary pixels interacting with neighboring pixels \bullet . The four middle boxes \odot indicate the inner pixels which do not involve c , others boundary pixels denoted by symbols \triangleleft , \triangleright , \triangle , ∇ involve c as in (13) via CDS^{loc} .

304

Then, minimizing for c , the problem (13) is equivalent to minimize the following

$$\begin{aligned}
F_{SC1}(c_{i,j}) = & \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2} g_{k_1-1,\ell} \sqrt{[c_{i,j} - (\tilde{u}_{k_1-1,\ell} - \tilde{u}_{k_1,\ell})]^2 + (\tilde{u}_{k_1-1,\ell} - \tilde{u}_{k_1-1,\ell+1})^2 + \beta} \\
& + \bar{\mu} \sum_{k=k_1}^{k_2-1} g_{k,\ell_2} \sqrt{[c_{i,j} - (\tilde{u}_{k,\ell_2+1} - \tilde{u}_{k,\ell_2})]^2 + (\tilde{u}_{k,\ell_2} - \tilde{u}_{k+1,\ell_2})^2 + \beta} \\
& + \bar{\mu} g_{k_2,\ell_2} \sqrt{[c_{i,j} - (\tilde{u}_{k_2,\ell_2+1} - \tilde{u}_{k_2,\ell_2})]^2 + [c_{i,j} - (\tilde{u}_{k_2+1,\ell_2} - \tilde{u}_{k_2,\ell_2})]^2 + \beta} \\
& + \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2-1} g_{k_2,\ell} \sqrt{[c_{i,j} - (\tilde{u}_{k_2+1,\ell} - \tilde{u}_{k_2,\ell})]^2 + (\tilde{u}_{k_2,\ell} - \tilde{u}_{k_2,\ell+1})^2 + \beta} \\
& + \bar{\mu} \sum_{k=k_1}^{k_2} g_{k,\ell_1-1} \sqrt{[c_{i,j} - (\tilde{u}_{k,\ell_1-1} - \tilde{u}_{k,\ell_1})]^2 + (\tilde{u}_{k,\ell_1-1} - \tilde{u}_{k+1,\ell_1-1})^2 + \beta} \\
& + \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (\tilde{u}_{k,\ell} + c_{i,j}) \left((c_1 - z_{k,\ell})^2 - (c_2 - z_{k,\ell})^2 \right) \\
& + \theta \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (\tilde{u}_{k,\ell} + c_{i,j}) P_{d_{k,\ell}} + \alpha \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} \nu (\tilde{u}_{k,\ell} + c_{i,j})
\end{aligned} \tag{15}$$

305

where the third term may be simplified using $(c - a)^2 + (c - b)^2 + \beta = 2(c - \frac{a+b}{2})^2 + 2(\frac{a-b}{2})^2 + \beta$.

306

Further the local minimization problem for block (i, j) on level k with respect to $c_{i,j}$ amounts

307

to minimising the following equivalent functional

$$\begin{aligned}
F_{SC1}(c_{i,j}) = & \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2} g_{k_1-1,\ell} \sqrt{(c_{i,j} - h_{k_1-1,\ell})^2 + v_{k_1-1,\ell}^2 + \beta} + \bar{\mu} \sum_{k=k_1}^{k_2-1} g_{k,\ell_2} \sqrt{(c_{i,j} - v_{k,\ell_2})^2 + h_{k,\ell_2}^2 + \beta} \\
& + \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2-1} g_{k_2,\ell} \sqrt{(c_{i,j} - h_{k_2,\ell})^2 + v_{k_2,\ell}^2 + \beta} + \bar{\mu} \sum_{k=k_1}^{k_2} g_{k,\ell_1-1} \sqrt{(c_{i,j} - v_{k,\ell_1-1})^2 + h_{k,\ell_1-1}^2 + \beta} \\
& + \bar{\mu} \sqrt{2} g_{k_2,\ell_2} \sqrt{(c_{i,j} - \bar{v}_{k_2,\ell_2})^2 + \bar{h}_{k_2,\ell_2}^2 + \frac{\beta}{2}} + \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (c_{i,j}) \left((c_1 - z_{k,\ell})^2 - (c_2 - z_{k,\ell})^2 \right) \\
& + \theta \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (\tilde{u}_{k,\ell} + c_{i,j}) P_{d_{k,\ell}} + \alpha \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} \nu (\tilde{u}_{k,\ell} + c_{i,j})
\end{aligned} \tag{16}$$

308

where we have used the following notation (which will be used later also):

$$\begin{aligned}
h_{k,\ell} &= \tilde{u}_{k+1,\ell} - \tilde{u}_{k,\ell}, & v_{k,\ell} &= \tilde{u}_{k,\ell+1} - \tilde{u}_{k,\ell}, & v_{k_2,\ell_2} &= \tilde{u}_{k_2,\ell_2+1} - \tilde{u}_{k_2,\ell_2}, \\
h_{k_2,\ell_2} &= \tilde{u}_{k_2+1,\ell_2} - \tilde{u}_{k_2,\ell_2}, & \bar{v}_{k_2,\ell_2} &= \frac{v_{k_2,\ell_2} + h_{k_2,\ell_2}}{2}, & \bar{h}_{k_2,\ell_2} &= \frac{v_{k_2,\ell_2} - h_{k_2,\ell_2}}{2}, \\
h_{k_1-1,\ell} &= \tilde{u}_{k_1,\ell} - \tilde{u}_{k_1-1,\ell}, & v_{k_1-1,\ell} &= \tilde{u}_{k_1-1,\ell+1} - \tilde{u}_{k_1-1,\ell}, & v_{k,\ell_2} &= \tilde{u}_{k,\ell_2+1} - \tilde{u}_{k,\ell_2}, \\
h_{k,\ell_2} &= \tilde{u}_{k+1,\ell_2} - \tilde{u}_{k,\ell_2}, & h_{k_2,\ell} &= \tilde{u}_{k_2+1,\ell} - \tilde{u}_{k_2,\ell}, & v_{k_2,\ell} &= \tilde{u}_{k_2,\ell+1} - \tilde{u}_{k_2,\ell}, \\
v_{k,\ell_1-1} &= \tilde{u}_{k,\ell_1} - \tilde{u}_{k,\ell_1-1}, & h_{k,\ell_1-1} &= \tilde{u}_{k+1,\ell_1-1} - \tilde{u}_{k,\ell_1-1}.
\end{aligned}$$

309

For solution on the coarsest level, we look for a **single** constant update for the current

310

approximation \tilde{u} that is

$$\begin{aligned}
\min_c \{ F_{SC1}(\tilde{u} + c) = & \sum_{i=1}^n \sum_{j=1}^n (\tilde{u}_{i,j} + c) \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) \\
& + \bar{\mu} \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} g_{i,j} \sqrt{(\tilde{u}_{i,j} + c - \tilde{u}_{i,j+1} - c)^2 + (\tilde{u}_{i,j} + c - \tilde{u}_{i+1,j} - c)^2 + \beta} \\
& + \theta \sum_{i=1}^n \sum_{j=1}^n P_{d_{i,j}} (\tilde{u}_{i,j} + c) + \alpha \sum_{i=1}^n \sum_{j=1}^n \nu (\tilde{u}_{i,j} + c) \}
\end{aligned}$$

311

which is equivalent to

$$\begin{aligned}
\min_c \{ F_{SC1}(\tilde{u} + c) = & \sum_{i=1}^n \sum_{j=1}^n (\tilde{u}_{i,j} + c) \left((c_1 - z_{i,j})^2 - (c_2 - z_{i,j})^2 \right) \\
& + \theta \sum_{i=1}^n \sum_{j=1}^n P_{d_{i,j}} (\tilde{u}_{i,j} + c) + \alpha \sum_{i=1}^n \sum_{j=1}^n \nu (\tilde{u}_{i,j} + c) \}.
\end{aligned} \tag{17}$$

312 The solutions of the above local minimization problems, solved by a Newton method as
313 in (12) or a fixed point method for t iterations (inner iteration), defines the update solution
314 $u = u + Q_k c$ where Q_k is the interpolation operator distributing $c_{i,j}$ to the corresponding $b_k \times b_k$
315 block on level k as illustrated in (14). Then we obtain a multilevel method if we cycle through
316 all levels and all blocks on each level until the relative error in two consecutive cycles (outer
317 iteration) is smaller than tol or the maximum number of cycle, $maxit$ is reached.

318 Finally our proposed multilevel method for CDSS is summarized in Algorithm 1. We will
319 use the term **SC1** to refer this multilevel Algorithm 1.

Algorithm 1 SC1 – Multilevel algorithm for the CDSS model

Given z , an initial guess u , the stop tolerance (tol), and maximum multilevel cycle ($maxit$) with $L + 1$ levels,

- 1) Set $\tilde{u} = u$.
 - 2) Smooth for t iteration the approximation on the finest level 1 that is solve (11) for $i, j = 1, 2, \dots, n$
 - 3) Iterate for t times on each coarse level $k = 2, 3, \dots, L, L + 1$:
 - > If $k \leq L$, compute the minimizer c of (16)
 - > Solve (17) on the coarsest level $k = L + 1$
 - > Add the correction $u = u + Q_k c$ where Q_k is the interpolation operator distributing $c_{i,j}$ to the corresponding $b_k \times b_k$ block on level k as illustrated in (14).
 - 4) Check for convergence using the above criteria. If not satisfied, return to Step 1. Otherwise exit with solution $u = \tilde{u}$.
-

320 In order to get fast convergence, it is recommended to start updating our multilevel algorithm
321 from the fine level to the coarse level. In a separate experiment we found that if we adjust the
322 coarse structure before the fine level, the convergence is slower. In addition, we recommend the
323 value of inner iteration $t = 1$ is used to update the algorithm in a fast manner.

324 4.2 A multilevel algorithm for the proposed model

325 We now consider our main model as expressed by (8)–(9). Minimizations of J is with respect
326 to u in (8) and w in (9) respectively. The solution of (9) can be obtained analytically following
327 Proposition 1. It remains to develop a multilevel algorithm to solve (8).

328 Similar to the last subsection, the discretized form of the functional $J_1(u, w)$ of problem (8)
329 is as follows:

$$\min_u \{J_1(u, w) = \bar{\mu} \sum_{i=1}^{n-1} \sum_{j=1}^{n-1} g_{i,j} \sqrt{(u_{i,j} - u_{i,j+1})^2 + (u_{i,j} - u_{i+1,j})^2} + \beta + \frac{1}{2\rho} \sum_{i=1}^n \sum_{j=1}^n (u_{i,j} - w_{i,j})^2\} \quad (18)$$

330 Clearly this is a much simpler functional than the CDSS model (10) so the method can be
331 similarly developed.

332 Consider the minimization of (18) by the coordinate descent method on the finest level 1:

333 Given $u^{(m)} = (u_{i,j}^{(m)})$ with $m = 0$;

$$\text{Solve } u_{i,j}^{(m)} = \arg \min_{u_{i,j} \in \mathbb{R}} J_1^{loc}(u_{i,j}, c_1, c_2) \text{ for } i, j = 1, 2, \dots, n; \quad (19)$$

334 Set $u_{i,j}^{(m+1)} = (u_{i,j}^{(m+1)})$ and repeat the above steps with $m = m + 1$ until a prescribed
335 stopping on m .

336

Here

$$\begin{aligned}
J_1^{loc}(u_{i,j}, c_1, c_2) = J_1 - J_0 &= \bar{\mu} g_{i,j} \sqrt{\left(u_{i,j} - u_{i+1,j}^{(m)}\right)^2 + \left(u_{i,j} - u_{i,j+1}^{(m)}\right)^2 + \beta} \\
&\quad + \bar{\mu} g_{i-1,j} \sqrt{\left(u_{i,j} - u_{i-1,j}^{(m)}\right)^2 + \left(u_{i-1,j} - u_{i-1,j+1}^{(m)}\right)^2 + \beta} \\
&\quad + \bar{\mu} g_{i,j-1} \sqrt{\left(u_{i,j} - u_{i,j-1}^{(m)}\right)^2 + \left(u_{i,j-1} - u_{i+1,j-1}^{(m)}\right)^2 + \beta} \\
&\quad + \frac{1}{2\rho} (u_{i,j} - w_{i,j})^2.
\end{aligned}$$

337 The term J_0 refers to a collection of all terms that are not dependent on $u_{i,j}$. For $u_{i,j}$ at
338 the boundary, Neumann's condition is used. Note that each subproblem in (19) is only one
339 dimensional, which is the key to the efficiency of our new method.

340 To introduce the multilevel algorithm, it is of interest to rewrite (19) in an equivalent form:

341

$$\hat{c} = \arg \min_{c \in \mathbb{R}} J_1^{loc}\left(u_{i,j}^{(m)} + c, c_1, c_2\right), \quad u_{i,j}^{(m)} = u_{i,j}^{(j)} + \hat{c} \text{ for } i, j = 1, 2, \dots, n. \quad (20)$$

342 Using the stencil in (14), the problem (20) is equivalent to minimize the following

$$\begin{aligned}
F_2(c_{i,j}) &= \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2} g_{k_1,\ell} \sqrt{\left[c_{i,j} - (\tilde{u}_{k_1-1,\ell} - \tilde{u}_{k_1,\ell})\right]^2 + (\tilde{u}_{k_1-1,\ell} - \tilde{u}_{k_1-1,\ell+1})^2 + \beta} \\
&\quad + \bar{\mu} \sum_{k=k_1}^{k_2-1} g_{k,\ell_2} \sqrt{\left[c_{i,j} - (\tilde{u}_{k,\ell_2+1} - \tilde{u}_{k,\ell_2})\right]^2 + (\tilde{u}_{k,\ell_2} - \tilde{u}_{k+1,\ell_2})^2 + \beta} \\
&\quad + \bar{\mu} g_{k_2,\ell_2} \sqrt{\left[c_{i,j} - (\tilde{u}_{k_2,\ell_2+1} - \tilde{u}_{k_2,\ell_2})\right]^2 + \left[c_{i,j} - (\tilde{u}_{k_2+1,\ell_2} - \tilde{u}_{k_2,\ell_2})\right]^2 + \beta} \\
&\quad + \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2-1} g_{k_2,\ell} \sqrt{\left[c_{i,j} - (\tilde{u}_{k_2+1,\ell} - \tilde{u}_{k_2,\ell})\right]^2 + (\tilde{u}_{k_2,\ell} - \tilde{u}_{k_2,\ell+1})^2 + \beta} \\
&\quad + \bar{\mu} \sum_{k=k_1}^{k_2} g_{k,\ell_1-1} \sqrt{\left[c_{i,j} - (\tilde{u}_{k,\ell_1-1} - \tilde{u}_{k,\ell_1})\right]^2 + (\tilde{u}_{k,\ell_1-1} - \tilde{u}_{k+1,\ell_1-1})^2 + \beta} \\
&\quad + \frac{1}{2\rho} \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (u_{k,\ell} + c_{i,j} - w_{k,\ell})^2.
\end{aligned} \quad (21)$$

343 After some algebraic manipulation to simplify (21), we arrive at the following

$$\begin{aligned}
F_2(c_{i,j}) &= \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2} g_{k_1-1,\ell} \sqrt{(c_{i,j} - h_{k_1-1,\ell})^2 + v_{k_1-1,\ell}^2 + \beta} + \bar{\mu} \sum_{k=k_1}^{k_2-1} g_{k,\ell_2} \sqrt{(c_{i,j} - v_{k,\ell_2})^2 + h_{k,\ell_2}^2 + \beta} \\
&\quad + \bar{\mu} \sum_{\ell=\ell_1}^{\ell_2-1} g_{k_2,\ell} \sqrt{(c_{i,j} - h_{k_2,\ell})^2 + v_{k_2,\ell}^2 + \beta} + \bar{\mu} \sum_{k=k_1}^{k_2} g_{k,\ell_1-1} \sqrt{(c_{i,j} - v_{k,\ell_1-1})^2 + h_{k,\ell_1-1}^2 + \beta} \\
&\quad + \bar{\mu} \sqrt{2} g_{k_2,\ell_2} \sqrt{(c_{i,j} - \bar{v}_{k_2,\ell_2})^2 + \bar{h}_{k_2,\ell_2}^2 + \frac{\beta}{2}} + \frac{1}{2\rho} \sum_{k=k_1}^{k_2} \sum_{\ell=\ell_1}^{\ell_2} (u_{k,\ell} + c_{i,j} - w_{k,\ell})^2.
\end{aligned} \quad (22)$$

344 On the coarsest level ($L + 1$), a **single** constant update for the current \tilde{u} is given as

$$\min_c \{ F_2(\tilde{u} + c) = \frac{1}{2\rho} \sum_{i=1}^n \sum_{j=1}^n (u_{i,j} + c - w_{i,j})^2 \} \quad (23)$$

345 which has a simple and explicit solution.

346 Then, we obtain a multilevel method if we cycle through all levels and all blocks on each
347 level. The process is stopped if the relative error in two consecutive cycles (outer iteration) is
348 smaller than tol or the maximum number of cycle, $maxit$ is reached.

349 The overall procedure to solve the new primal-dual model is given in Algorithm 2. We will
350 use the term **SC2** to refer this algorithm to solve the proposed model expressed in (8) and (9).
351 Again, in order to update the algorithm in a fast manner, we recommend to adjust the fine
352 level before the coarse level and to use the inner iteration $t = 1$.

Algorithm 2 SC2 – Algorithm to solve the new primal-dual model

Given image z , an initial guess u , the stop tolerance (tol), and maximum multilevel cycle ($maxit$) with $L + 1$ levels. Set $w = u$,

1) Solve (8) to update u using the following steps:

- i). Set $\tilde{u} = u$.
- ii). Smooth for t iteration the approximation on the finest level 1 that is solve (19) for $i, j = 1, 2, \dots, n$
- iii). Iterate for t times on each coarse level $k = 2, 3, \dots, L, L + 1$:
 - > If $k \leq L$, compute the minimizer c of (22)
 - > Solve (23) on the coarsest level $k = L + 1$
 - > Add the correction $u = u + Q_k c$ where Q_k is the interpolation operator distributing $c_{i,j}$ to the corresponding $b \times b$ block on level k as illustrated in (14).

2) Solve (9) to update w :

- i). Set $\tilde{w} = w$.
- ii). Compute w using the formula (7).

3) Check for convergence using the above criteria. If not satisfied, return to Step 1. Otherwise exit with solution $u = \tilde{u}$ and $w = \tilde{w}$

353 5 A new variant of the multilevel algorithm SC2

354 Our above proposed method defines a sequence of search directions based in a multilevel setting
355 for an optimization problem. We now modify it so that the new algorithm has a formal decaying
356 property.

Denote the functional in (18) by $g(u) : \mathbb{R}^{n^2} \rightarrow \mathbb{R}$ and represent each subproblem by

$$c^* = \operatorname{argmin}_{c \in \mathbb{R}} g(u^\ell + cp^\ell), \quad u^{\ell+1} = u^\ell + c^*p^\ell, \quad p^\ell = \tilde{\mathbf{e}}^{\ell(\bmod K)+1}, \quad \ell = 0, 1, 2, \dots$$

357 where $K = \sum_{k=0}^L \frac{n^2}{4^k} = (4n^2 - 1)/3$ is the total number of search directions across all levels
358 $1, 2, \dots, L + 1$ for this unconstrained optimization problem. We first investigate these search

- 381 i) $g(u)$ is continuously differentiable in $D_0 = [0, 1]^{n^2} \subset \mathbb{R}^{n^2}$;
382 ii) the sequence $\{q^\ell\}$ is uniformly linearly independent;
383 iii) the sequence $\{u^\ell\}$ is strongly downward (decaying) with respect to $g(u)$;
384 iv) $\lim_{\ell \rightarrow \infty} g'(u^\ell)q^\ell / \|q^\ell\| = 0$,
385 v) the set $S = \{u \in D_0 \mid g'(u) = 0\}$ is non-empty.

Here $q'(u) = (\nabla g(u))^T$. Then we have the convergence of $\{u^\ell\}$ to a critical point u^* [44, Thm 14.1.4]

$$\lim_{\ell \rightarrow \infty} \inf_{u \in S} \|u^\ell - u^*\| = 0.$$

386 We now verify these conditions. Firstly condition i) is evident if $\beta \neq 0$ and condition
387 ii) also holds since ‘essentially periodic’ implies ‘uniformly linearly independent’ [44, §14.6.3].
388 Condition v) requires an assumption of existence of stationary points for $g(u)$. Below we focus
389 on verifying iii)-iv). From [44, Thm 14.2.7], the construction of $\{u^\ell\}$ via (25) ensures that the
390 sequence $\{u^\ell\}$ is strongly downward and further $\lim_{\ell \rightarrow \infty} g'(u^\ell)q^\ell / \|q^\ell\| = 0$. Hence conditions
391 iii)-iv) are satisfied.

392 Note condition iii) and the assumption of $g(u)$ being hemivariate imply that $\lim_{\ell \rightarrow \infty} \|u^{\ell+1} -$
393 $u^\ell\| = 0$ from [44, Thm 14.1.3]. Further condition iv) and the fact $\lim_{\ell \rightarrow \infty} \|u^{\ell+1} - u^\ell\| = 0$ lead
394 to the result $\lim_{\ell \rightarrow \infty} g'(u^\ell) = \mathbf{0}$. Finally by [44, Thm 14.1.4], the condition $\lim_{\ell \rightarrow \infty} g'(u^\ell) = \mathbf{0}$
395 implies $\lim_{\ell \rightarrow \infty} \inf_{u \in S} \|u^\ell - u^*\| = 0$. Hence the convergence is proved.

396 Next, we will give the complexity analysis of our **SC1**, **SC2** and **SC2M**. Let $N = n^2$ be the
397 total number of pixels (unknowns). First, we compute the number of floating point operations
398 (flops) for **SC1** for level k as follows:

Quantities	Flop counts for SC1
h, v	$4b_k\tau_k^2$
θ terms	$2N$
data terms	$2N$
α terms	$2N$
s smoothing steps	$38b_k\tau_k^2s$

400 Then, the flop counts for all level is $W_{SC1} = \sum_{k=1}^{L+1} (6N + 4b_k\tau_k^2 + 38b_k\tau_k^2s)$ where $k = 1$
401 (finest) and $k = L + 1$ (coarsest). Noting $b_k = 2^{k-1}$, $\tau_k = n/b_k$, $N = n^2$, we compute the upper
402 bound for **SC1** as follows:

$$\begin{aligned} W_{SC1} &= 6(L+1)N + \sum_{k=1}^{L+1} \left(\frac{4N}{b_k} + \frac{38Ns}{b_k} \right) = 6(L+1)N + (4 + 38s)N \sum_{k=0}^L \left(\frac{1}{2^k} \right) \\ &< 6N \log n + 14N + 76Ns \approx O(N \log N) \end{aligned}$$

403 Similarly, the flops for **SC2** is given as

Quantities	Flop counts for SC2
h, v	$4b_k\tau_k^2$
ρ term	$2N$
w term	$6N$
s smoothing steps	$31b_k\tau_k^2s$

405 Hence, the total flop counts for SC2 is $W_{SC2} = 6N + \sum_{k=1}^{L+1} (2N + 4b_k\tau_k^2 + 31b_k\tau_k^2s)$. This
406 gives the upper bound for **SC2** as

$$W_{SC2} = 6N + 2(L+1)N + \sum_{k=1}^{L+1} \left(\frac{4N}{b_k} + \frac{31Ns}{b_k} \right) = 6N + 2(L+1)N + (4+31s)N \sum_{k=0}^L \left(\frac{1}{2^k} \right) \\ < 2N \log n + 16N + 62Ns \approx O(N \log N)$$

407 Finally, the approximate cost of an extra operation $\nabla J^T q^\ell$ in **SC2M** is $2N$ that results to
408 the total flop counts for SC2M as $W_{SC2M} = 6N + \sum_{k=1}^{L+1} (4N + 4b_k\tau_k^2 + 31b_k\tau_k^2s)$. This gives the
409 upper bound for **SC2M** as

$$W_{SC2M} = 6N + 4(L+1)N + \sum_{k=1}^{L+1} \left(\frac{4N}{b_k} + \frac{31Ns}{b_k} \right) = 6N + 4(L+1)N + (4+31s)N \sum_{k=0}^L \left(\frac{1}{2^k} \right) \\ < 4N \log n + 18N + 62Ns \approx O(N \log N)$$

410 One can observe that both **SC1**, **SC2** and **SC2M** are of the optimal complexity $O(N \log N)$
411 expected of a multilevel method and $W_{SC1} > W_{SC2M} > W_{SC2}$.

412 7 Numerical experiments

413 This section will demonstrate the performance of the developed multilevel methods through
414 several experiments. The algorithms to be compared are:

Name		Algorithm	Description
CMT	Old	:	The selective segmentation model proposed by Liu <i>et al.</i> [35] solved by a multilevel algorithm.
NCZZ	Old	:	The interactive image segmentation model proposed by Nguyen <i>et al.</i> [41] solved by a Split Bregman method.
BC	Old	:	The selective segmentation model proposed by Badshah and Chen [7] solved by an AOS algorithm.
RC	Old	:	The selective segmentation model proposed by Rada and Chen [47] solved by an AOS algorithm.
SC0	Old	:	The modified AOS algorithm [52] for the CDSS model [52].
SC1	New	:	The multilevel Algorithm 1 for the CDSS model [52].
SC2	New	:	The multilevel Algorithm 2 for the new primal-dual model (8)–(9).
SC2M	New	:	The modified multilevel algorithm for SC2.

416 There are five sets of tests carried out. In the **first set**, we will choose the best multilevel
417 algorithm among SC1, SC2 and SC2M by comparing their segmentation performances in terms
418 of CPU time (in seconds) and quality. The segmentation quality is measured based on the
419 Jaccard similarity coefficient (JSC):

$$JSC = \frac{|S_n \cap S_*|}{|S_n \cup S_*|}$$

420 where S_n is the set of the segmented domain u and S_* is the true set of u (which is only easy to
421 obtain for simple images). The similarity functions return values in the range $[0, 1]$. The value
422 1 indicates perfect segmentation quality while the value 0 indicates poor quality.

423 In the **second set**, we will perform the speed, quality, and parameter sensitivity test for the
424 chosen multilevel algorithm (from set 1) and compare its performance with SC0. In the **third**,
425 **fourth**, and **fifth set**, we will perform the segmentation quality comparison of the chosen

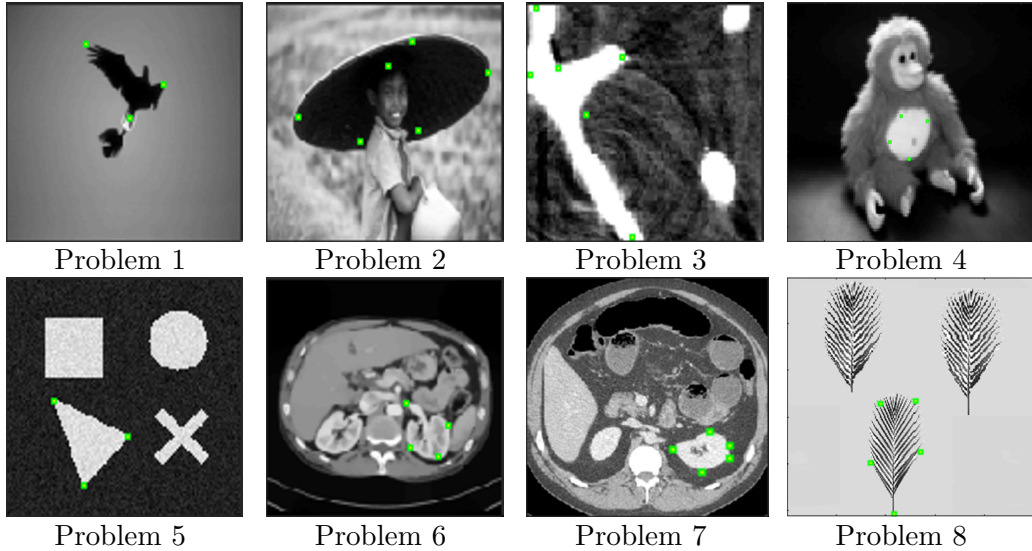


Figure 3: Segmentation test images and markers.

426 multilevel algorithm (from set 1) with CMT model [35], NCZZ model [41], and BC model [7]
427 and RC model [47] respectively.

428 The test images used in this paper are listed in Figure 3. We remark that Problems 1-2 are
429 obtained from the Berkeley segmentation dataset and benchmark [38], while Problems 3-4 are
430 obtained from database provided by [25]. All algorithms are implemented in MATLAB R2017a
431 on a computer with Intel Core i7 processor, CPU 3.60GHz, 16 GB RAM CPU.

432 As a general guide to choose suitable parameters for different images, our experimental
433 results recommend the following. The parameters $\bar{\mu} = \mu$ can be between 10^{-5} and 5×10^5 ,
434 $\beta = 10^{-4}$, ρ in between 10^{-5} and 10^{-1} , and γ in between $1/255^2$ and 10. Tuning the parameter
435 θ depends on the targeted object. If the object is too close to a nearby boundary then θ should
436 be large. Segmenting a clearly separated object in an image needs just a small θ .

437 7.1 Test Set 1: Comparison of SC1, SC2, and SC2M

438 In the first experiment, we compare the segmentation speed and quality for SC1, SC2 and
439 SC2M using test Problem 1-4 with size of 128×128 . Here, we take $\bar{\mu} = 1$, $\beta = 10^{-4}$, $\rho = 10^{-3}$,
440 $\theta = 1000$ (Problem 1-3), $\theta = 2000$ (Problem 4), $\varepsilon = 0.12$, $\gamma = 10$, $tol = 10^{-2}$ and $maxit = 10^4$.

441 Figure 4 shows successful selective segmentation results by SC1, SC2 and SC2M for Problem
442 4. The segmentation quality for all algorithms is the same (JSC=0.96). However, SC2 performs
443 faster (4.9 seconds) than SC1 (10.5 seconds) and SC2M (6.3 seconds).

444 The remaining results are tabulated in Table 1. We can see for all four test problems, SC2
445 gives the highest accuracy and performs the fastest compared to SC1 and SC2M.

446 Next, we test the performance of all the multilevel algorithms to segment Problem 5 in
447 different resolutions. We take $\bar{\mu} = 1$, $\beta = 10^{-4}$, $\rho = 10^{-5}$, $\theta = 5000$, $\varepsilon = 0.12$, $\gamma = 10$,
448 $tol = 10^{-3}$ and $maxit = 10^4$. The segmentation results for image size 1024×1024 are shown in
449 Figure 5. The CPU times needed by SC2 to complete the segmentation of image size 1024×1024
450 is 413.2s while SC1 and SC2M need 690.6s and 636.1s respectively which implies that SC2 can
451 be 277s faster than SC1 and 222s faster than SC2M. All the algorithms reach equal quality of
452 segmentation.

453 The remaining result in terms of quality and CPU time are tabulated in Table 2. Column
454 6 (ratios of the CPU times) shows that SC1, SC2 and SC2M are of complexity $O(N \log N)$.

Table 1: Test Set 1 – Comparison of computation time (in seconds) and segmentation quality of SC1, SC2, and SC2M for Problem 1- 4. Clearly, for all four test problems, SC2 gives the highest accuracy and performs fast segmentation process compared to SC1 and SC2M.

Algorithm	Problem	Iteration	CPU time (s)	JSC
SC1	1	6	7.0	0.82
	2	12	20.0	0.82
	3	15	24.4	0.91
	4	6	10.5	0.96
SC2	1	5	5.9	0.82
	2	8	8.7	0.82
	3	4	4.9	0.91
	4	4	4.9	0.96
SC2M	1	5	7.9	0.79
	2	8	11.7	0.82
	3	5	7.9	0.85
	4	4	6.3	0.96

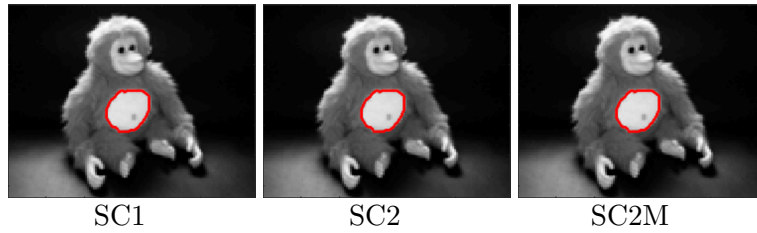


Figure 4: Test Set 1 – Segmentation of Problem 4 using our multilevel algorithms SC1, SC2, and SC2M with same quality (JSC=0.96) achieved. However, SC2 performs faster (4.9 seconds) compared to SC1 (10.5 seconds) and SC2M (6.3 seconds).

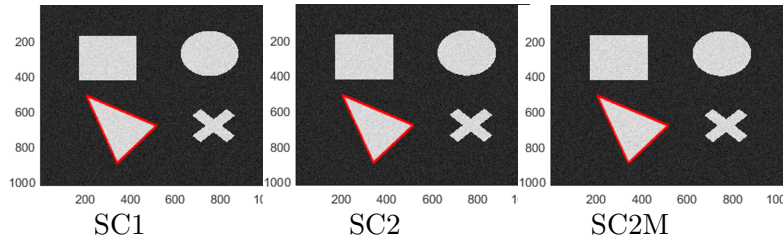


Figure 5: Test Set 1 – Segmentation of Problem 5 of size 1024x1024 for SC1, SC2, and SC2M. SC2 can be 277 seconds faster than SC1 and 222 seconds faster than SC2M : see Table 2. All algorithms give similar segmentation quality.

Table 2: Test Set 1 – Comparison of computation time (in seconds) and segmentation quality of SC1, SC2 and SC2M for Problem 5. The time ratio, t_n/t_{n-1} close to 4.4 indicates $O(N \log N)$ speed. Clearly, all algorithms have similar quality but SC2 is faster than SC1 and SC2M for all image sizes.

Algorithm	Size $N = n \times n$	Unknowns N	Iteration	Time, t_n	$\frac{t_n}{t_{n-1}}$	JSC
SC1	128×128	16384	6	10.6		1.0
	256×256	65536	7	43.5	4.1	1.0
	512×512	262144	7	173.7	4.0	1.0
	1024×1024	1048576	7	690.6	4.0	1.0
SC2	128×128	16384	8	8.7		1.0
	256×256	65536	7	23.7	2.7	1.0
	512×512	262144	8	103.9	4.4	1.0
	1024×1024	1048576	8	413.2	4.0	1.0
SC2M	128×128	16384	8	11.6		1.0
	256×256	65536	7	36.5	3.1	1.0
	512×512	262144	8	156.7	4.3	1.0
	1024×1024	1048576	8	636.1	4.1	1.0

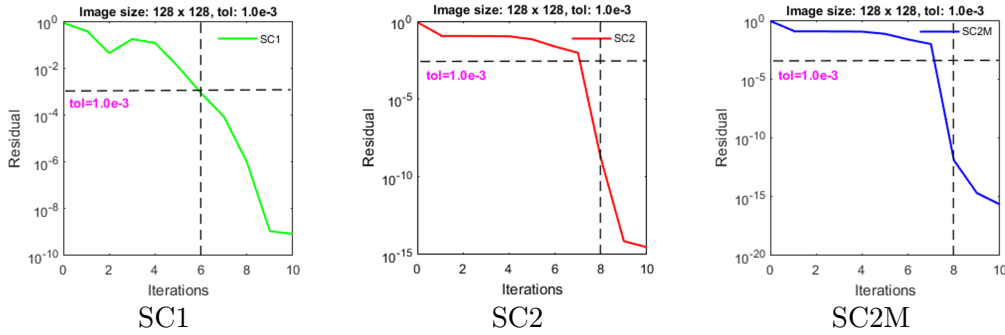


Figure 6: Test Set 1 – The residual plots for SC1, SC2, and SC2M to illustrate the convergence of the algorithms. The extension up to 10 iterations shows that the residual of the algorithms keep reducing. The residual for SC2 and SC2M decrease rapidly compared to SC1.

455 Again, we can see that for all image sizes, all algorithms have equal quality but SC2 is faster
456 than other algorithms.

457 To illustrate the convergence of our multilevel algorithms, we plot in Figure 6 the residuals
458 of SC1, SC2 and SC2M in segmenting Problem 5 for size 128×128 based on Table 2. There we
459 extend the iterations up to 10. As we can see, the residuals of the algorithms keep reducing.
460 The residuals for SC2 and SC2M decrease more rapidly than SC1.

461 Based on the experiments above, we observe that SC2 performs faster than the other two
462 multilevel algorithms. In addition, for all problems tested, SC2 gives the higher segmentation
463 quality than SC1 and SC2M. Therefore in practice, we recommend SC2 as the better multilevel
464 algorithm for our convex selective segmentation method.

465 7.2 Test Set 2: Comparison of SC2 with SC0

466 The second set starts with the speed and quality comparison of SC2 with SC0 in segmenting
467 Problem 5 with multiple resolutions. We take $\bar{\mu} = \mu = 1$, $\beta = 10^{-4}$, $\rho = 10^{-5}$, $\theta = 5000$,
468 $\varepsilon = 0.01$, $\gamma = 10$, $tol = 10^{-6}$ and $maxit = 5000$.

Table 3: Test Set 2 – Comparison of computation time (in seconds) and segmentation quality of SC0 and SC2 for Problem 5 with different resolutions. Again, the time ratio, $t_n/t_{n-1} \approx 4.4$ indicates $O(N \log N)$ speed since $N_L = n_L^2 = (2^L)^2 = 4^L$ and $kN_L \log N_L / (kN_{L-1} \log N_{L-1}) = 4L / (L - 1) \approx 4.4$. Clearly, all algorithms have similar quality but SC2 is faster than SC0 for all image sizes. Here, (**) means taking too long to run. For image size 512×512 , SC2 performs 33 times faster than SC0.

Algorithm	Size	Time,	$\frac{t_n}{t_{n-1}}$	JSC
	$N = n \times n$	t_n	t_{n-1}	
SC0	128×128	243.5		1.0
	256×256	872.7	3.6	1.0
	512×512	3803.1	4.4	1.0
	1024×1024	**	**	**
SC2	128×128	8.6		1.0
	256×256	27.2	3.2	1.0
	512×512	112.0	4.1	1.0
	1024×1024	453.6	4.1	1.0

469 The segmentation results are tabulated in Table 3. The ratios of the CPU times in column
 470 4 show that SC0 and SC1 are of complexity $O(N \log N)$. The symbols (**) indicates that too
 471 much time is taken to complete the segmentation task. For all image sizes, SC0 and SC2 give
 472 the same high quality.

473 Next, we shall test parameter sensitivity for our recommended SC2. We focus on three
 474 important parameters: the regularization parameter μ , the regularising parameter β and the
 475 area parameter θ . The SC2 results are compared with SC0.

476 **Test on parameter μ .** The regularization parameter μ in a segmentation model not only
 477 controls a balance of the terms but also implicitly defines the minimal diameter of detected
 478 objects among a possibly noisy background [54]. Here, we test sensitivity of SC2 for different
 479 regularization parameters μ in segmenting an object in Problem 6 and compare with SC0 in
 480 terms of segmentation quality. We set $\beta = 10^{-4}$, $\rho = 10^{-5}$, $\varepsilon = 0.01$, $\gamma = 1/255^2$, $\theta = 5000$,
 481 $tol = 10^{-5}$ and $maxit = 10^4$.

482 Figure 7a shows the value of JSC for SC0 and SC2 respectively for different values of μ .
 483 Clearly, SC2 is successful for larger range of μ than SC0. This finding implies that SC2 is less
 484 dependent to parameter μ than SC0.

485 **Test on area parameter θ .** As a final comparison of SC0 and SC2, we will test how the
 486 area parameter θ effects the segmentation quality of SC0 and SC2. For this comparison, we
 487 use Problem 6 and set $\bar{\mu} = \mu = 100$, $\beta = 10^{-4}$, $\rho = 10^{-3}$, $\varepsilon = 0.01$, $\gamma = 1/255^2$, $tol = 10^{-5}$
 488 and $maxit = 10^4$. Figure 7b shows the value of JSC for SC0 and SC2 respectively for different
 489 values of θ . We observe that SC2 is successful for a larger range of θ than SC0. This finding
 490 implies that SC2 is less sensitive to parameter θ than SC0.

491 **Test on parameter β .** Finally, we examine the sensitivity of our proposed SC2 on
 492 parameter β . The parameter β is used to avoid singularity or to ensure the original cost
 493 function is differentiable and it should be as small as possible (close to 0) so that the mo-
 494 dified cost function (having β) in (18) is close to the original cost function in (8). We have
 495 chosen to segment an object (organ) in Problem 6. Six different values of β are tested:
 496 $\beta = 1, 10^{-1}, 10^{-5}, 10^{-10},$ and 10^{-15} . Here, $\bar{\mu} = 100$, $\rho = 10^{-3}$, $\theta = 5500$, $\gamma = 1/255^2$,
 497 $tol = 10^{-3}$ and $maxit = 10^4$. For quantitative analysis, we compute the energy value in equa-
 498 tion (6) (that has no β) and the JSC value. Both values are tabulated in Table 4. One can
 499 see that as β decreases, the energy value gets closer to each other. The segmentation quality
 500 measured by JSC values remain the same as β decreases. This result indicates that SC2 is not
 501 sensitive to β ; large energy values for large β are expected.

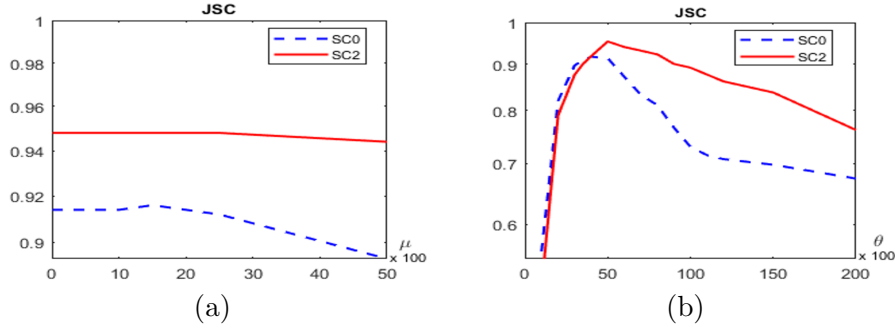


Figure 7: Test Set 2 – The segmentation accuracy for SC0 and SC2 in segmenting Problem 6 using different values of parameter μ in (a) and parameter θ in (b). The results demonstrate that SC2 is successful for a much larger range for both parameters.

Table 4: Test Set 2 – Dependence of our SC2 on β for segmenting Problem 6 in Figure 3.

β	JSC	Energy
1	0.95	-5.326416e+04
10^{-1}	0.95	-5.325908e+04
10^{-5}	0.95	-5.326213e+04
10^{-10}	0.95	-5.326153e+04
10^{-15}	0.95	-5.326122e+04

502 7.3 Test Set 3: Comparison of SC2 with CMT model [35]

503 In this test set 3, we investigate how the number of markers and threshold values will effect
504 the segmentation quality for CMT model [35] and our SC2. For this purpose, we use the test
505 Problem 4. We set $\bar{\mu} = 10^{-5}$, $\beta = 10^{-4}$, $\rho = 20$, $\theta = 3.5$, $\gamma = 20$, $tol = 10^{-3}$ and $maxit = 10^4$.
506 The first row in Figure 8 shows the Problem 4 with different number of markers. There are 4
507 markers in (a1), 6 markers in (b1) and 9 markers used in (c1). The results given by CMT and
508 SC2 using the markers with different threshold value are plotted respectively in the second row.

509 We observe that CMT performs well only when the number of markers used is large while
510 our SC2 is less sensitive to the number of markers used. In addition, it is clearly shown that
511 the range of threshold values that work for SC2 is wider than CMT. Consequently, our SC2 is
512 more reliable than CMT.

513 7.4 Test Set 4: Comparison of SC2 with NCZZ model [41]

514 For almost all of the test images in Figure 3, we see that the NCZZ model [41] gives same
515 satisfactory results as our SC2. For brevity, we will not show too many cases where both
516 models give satisfactory results; Figure 9 shows the successful segmentation of an organ in
517 Problem 7 of size 256×256 by NCZZ model. There two types of markers are used to label
518 foreground region (red) and background region (blue) for the NCZZ model [41] as shown in
519 Figure 9(a). Successful segmentation results (zoom in) by NCZZ model [41] and our SC2 for
520 Problem 7 are shown in (b) and (c) respectively using the following parameters; $\bar{\mu} = 0.01$,
521 $\beta = 10^{-4}$, $\rho = 10^{-3}$, $\theta = 3000$, $\gamma = 10$, $tol = 10^{-2}$ and $maxit = 10^4$.

522 However, according to the authors [41], the model unable to segment semi-transparent boun-
523 daries and sophisticated shapes (such as bush branches or hair in a clean way. In Figure 10,
524 we demonstrate the limitation of NCZZ model using Problems 1 and 8. The set of parameters

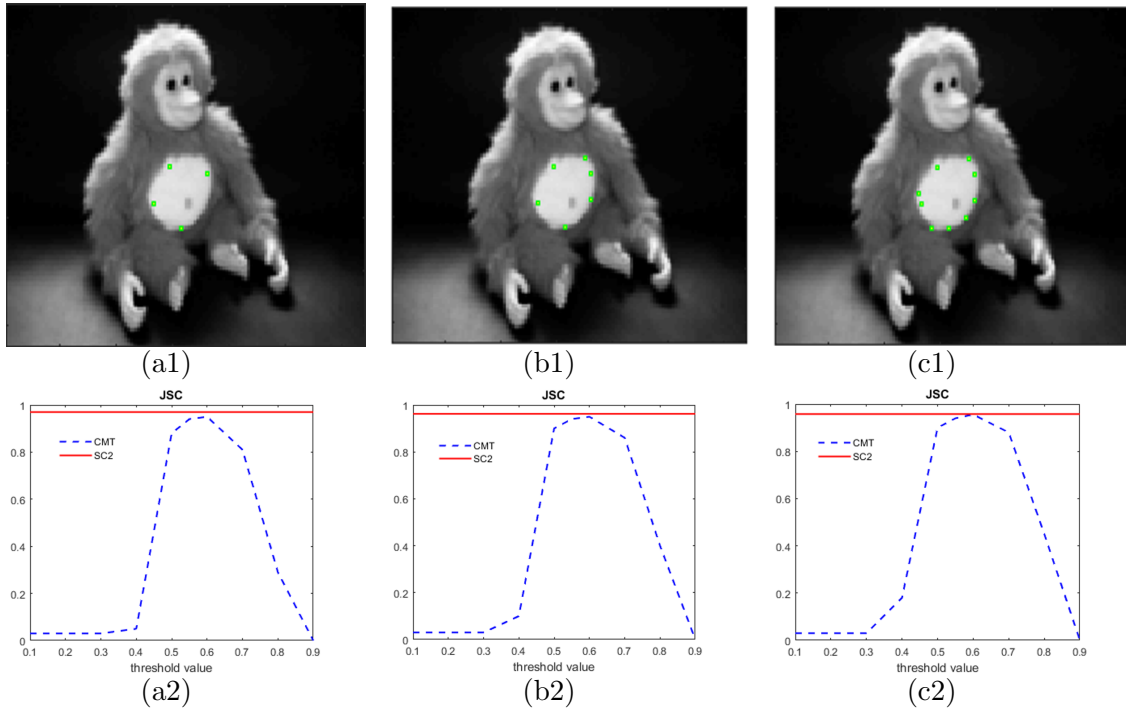


Figure 8: Test Set 3 – Comparison of SC2 with CMT model [35]. First row shows different numbers of markers used for Problem 4. Second row demonstrates the respective results (a2), (b2) and (c2) for (a1), (b1) and (c1) with different threshold values. Clearly, CMT performs well only when the number of markers used is large while our SC2 seems less sensitive to the number of markers used. Furthermore, the range of threshold value that works for SC2 is wider than CMT.

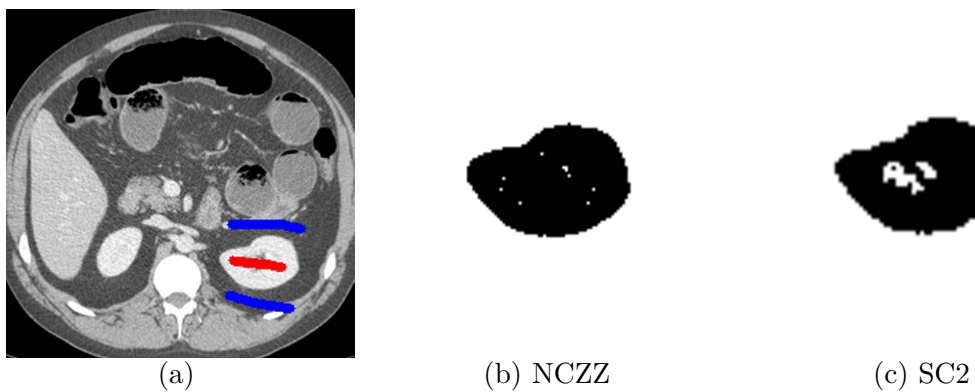


Figure 9: Problem 7 in Test Set 4 – Two types of markers used to label foreground region (red) and background region (blue) for NCZZ model [41] in (a). Successful segmentation result (zoom in): (b) by NCZZ model [41] and (c) by our SC2 (only using foreground markers).

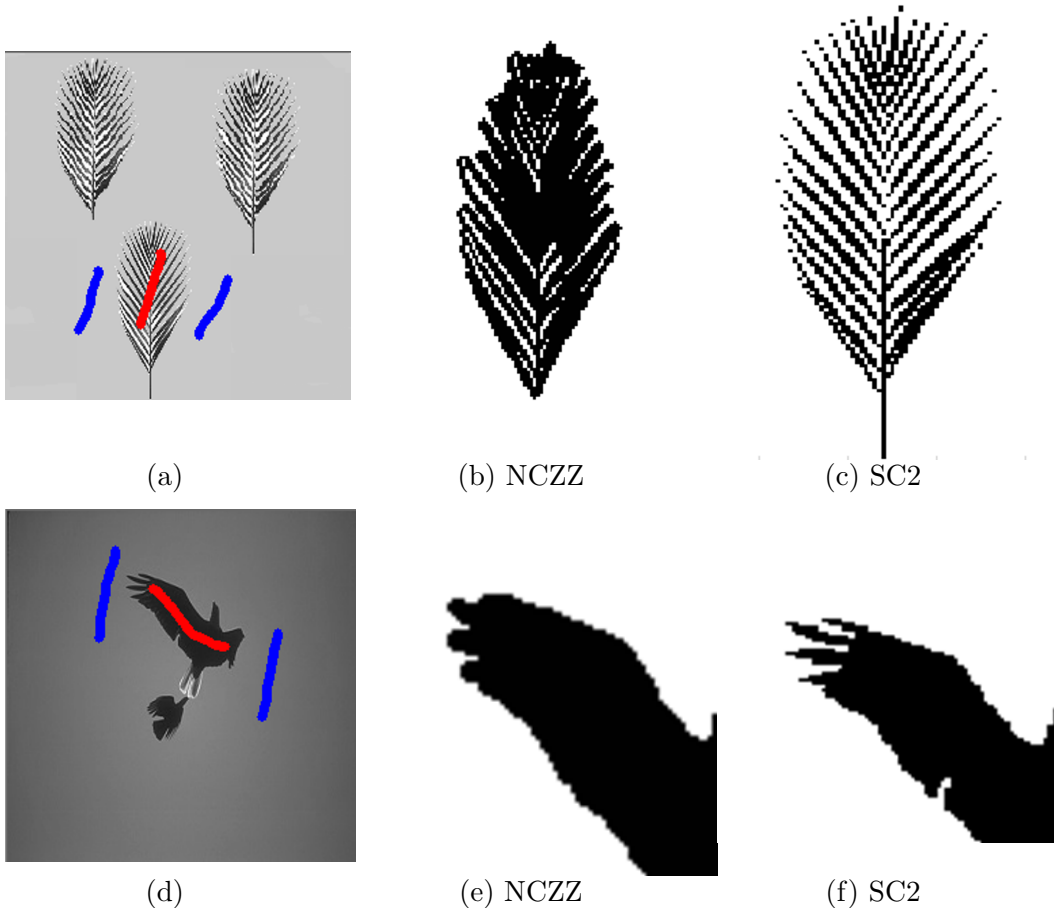


Figure 10: Problems 1,8 in Test Set 4 – (a) and (d) show the foreground markers (red) and background markers (blue) for NCZZ model [41]. Zoomed segmentation results in (b) and (e) demonstrate the limitation of NCZZ model [41] that is unable to segment semi-transparent boundaries and sophisticated shapes (such as bush branches or hair as explained in [41]) in a clean way. Our SC2 gives cleaner segmentation for the same problems as illustrated in (c) and (f).

525 are $\bar{\mu} = 0.01$, $\beta = 10^{-4}$, $\rho = 10^{-3}$, $\theta = 2000$ (Figure 10(a)), $\theta = 400$ (Figure 10(d)), $\gamma = 10$,
526 $tol = 10^{-2}$ and $maxit = 10^4$.

527 Zoomed segmentation results in Figure 10(b) and (e) demonstrate the limitation of NCZZ
528 model [41]. As comparison, our SC2 gives cleaner segmentation as illustrated in Figure 10(c)
529 and (f) for the same problems.

530 7.5 Test Set 5: Comparison of SC2 with BC [7] and RC [47]

531 Finally, we compare the performance of SC2 with two non-convex models namely BC model
532 [7] and RC model [47] for different initializations in segmenting Problem 3. We set $\bar{\mu} = 128 \times$
533 128×0.05 , $\beta = 10^{-4}$, $\rho = 10^{-4}$, $\theta = 1000$, $\gamma = 5$, $tol = 10^{-4}$ and $maxit = 10^4$. Figures 11(a)
534 and 11(b) show two different initializations with fixed markers.

535 The second row shows the results for all three models using the first initialization in (a) and
536 the third row using the second initialization in (b). It can be seen that under different initiali-
537 zations, our SC2 will result in the same, consistent segmentation curves (hence independent of
538 initializations) showing the advantage of a convex model. However, the segmentation results for
539 BC and RC models are heavily dependent on the initialization; a well known drawback of non-

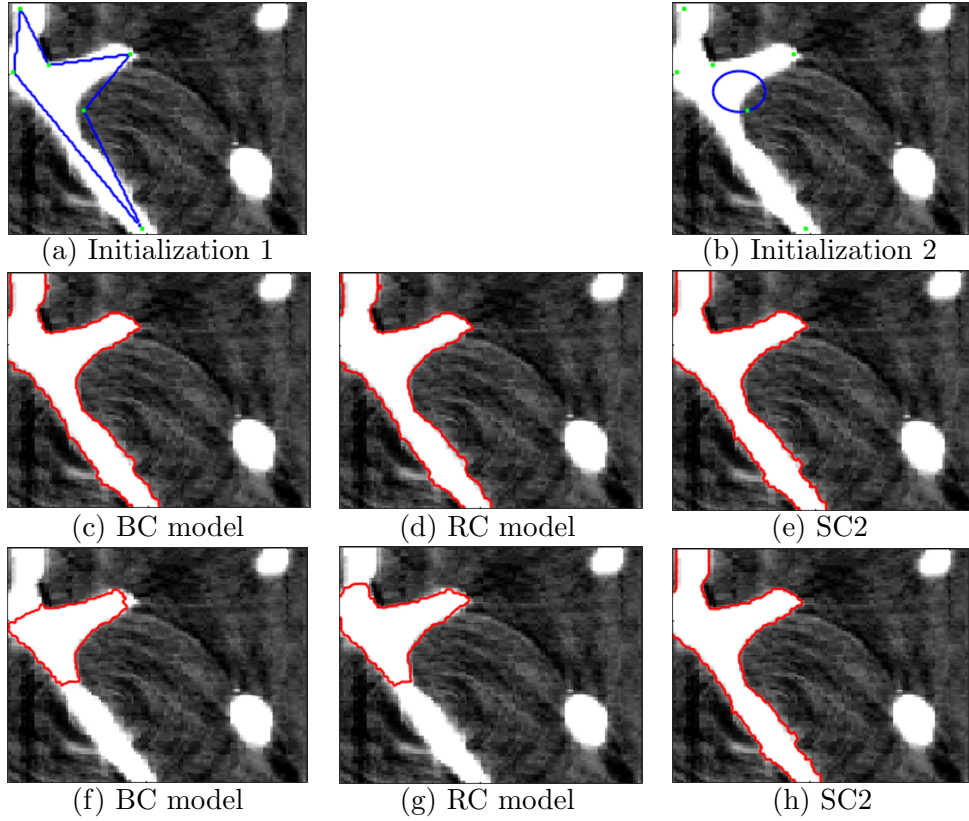


Figure 11: Test Set 5 – Performance comparison of BC, RC and SC2 models using 2 different initializations. With Initialization 1 in (a), the segmentation results for BC, RC, and SC2 models are illustrated on second row (c-e) respectively. With Initialization 2 in (b), the results are shown on third row (f-h). Clearly, SC2 gives a consistent segmentation result indicating that our SC2 is independent of initializations while BC and RC are sensitive to initializations due to different results obtained.

540 convex models. In addition, the segmentation result of non-convex models is not guaranteed to
 541 be a global solution.

542 8 Conclusions

543 In this work, we present a new primal-dual formulation for CDSS model [52] and propose an
 544 optimization based multilevel algorithm SC2 to solve the new formulation. In order to get a
 545 stronger decaying property than SC2, a new variant of SC2 named as SC2M is proposed. We
 546 also have developed a multilevel algorithm for the original CDSS model [52] called as SC1.

547 Five sets of tests are presented to compare eight models. In **Test Set 1** of the experiment,
 548 we find that all the multilevel algorithms have the expected optimal complexity $O(N \log N)$.
 549 However, SC2 converges faster than SC1 and SC2M. In addition, for all tested images, SC2
 550 gives high accuracy compared to SC1 and SC2M. Practically, we recommend SC2 as the better
 551 multilevel algorithm for convex and selective segmentation method. In **Test Set 2**, we have
 552 performed the speed and quality comparisons of SC2 with SC0. Results show that SC2 performs
 553 much faster than SC0. Both algorithms deliver same high quality for the tested problem. We
 554 also have run the sensitivity test for our recommended algorithm SC2 towards parameters μ
 555 and θ . Comparison of SC2 with SC0 shows that SC2 is less sensitive to the regularization
 556 parameters μ and θ . Moreover, SC2 is also less sensitive for parameter β . In **Test Set 3**, we

557 compare the segmentation quality of SC2 with the recent model CMT. The result demonstrates
 558 that SC2 performs better than CMT even for few markers. Moreover, the range of threshold
 559 values that work for SC2 is wider than CMT. In **Test Set 4**, the segmentation quality of SC2
 560 is compared with NCZZ model. For the tested problem, it is clear that SC2 has successfully
 561 reduced the difficulty of NCZZ model that is unable to segment semi-transparent boundaries
 562 and sophisticated shapes. The final **Test Set 5** demonstrates the advantage of SC2 being a
 563 convex model (independent of initializations) compared to two non-convex models (BC and
 564 RC).

565 In future work, we will extend SC2 to 3D formulation and develop an optimization based
 566 multilevel approach for higher order selective segmentation models.

567 Acknowledgements

568 The first author would like to thank to Faculty of Computer and Mathematical Sciences, Uni-
 569 versiti Teknologi MARA Shah Alam and Ministry of Higher Education of Malaysia for funding
 570 a scholarship to support this research. The second author is grateful to the support from the
 571 UK EPSRC for the grant EP/N014499/1.

572 References

- 573 [1] R. Adams and L. Bischof. Seeded region growing. *IEEE Trans. Pattern Anal. Mach. Intell.*,
 574 16(6), pp:641–647,1994.
- 575 [2] G. Aubert and P. Kornprobst. *Mathematical problems in image processing: Partial diffe-*
 576 *rential Equations and the Calculus of Variations*. Springer, 2001.
- 577 [3] J. F. Aujol, and A. Chambolle. Dual norms and image decomposition models. *International*
 578 *Journal of Computer Vision*, 63(1), pp:85–104, 2005.
- 579 [4] J. F. Aujol, G. Gilboa, T. F. Chan, and S. Osher. Structure-texture image decomposition-
 580 modelling, algorithms, and parameter selection. *International Journal of Computer Vision*,
 581 67(1), pp:111–136, 2006.
- 582 [5] N. Badshah and K. Chen. Multigrid method for the Chan-Vese model in variational seg-
 583 mentation. *Commun. Comput. Phys.*, 4(2), pp:294–316, 2008.
- 584 [6] N. Badshah and K. Chen. On Two multigrid algorithms for modelling variational multiphase
 585 image segmentation. *IEEE Transactions on Image Processing*, 18(5), pp:1097–1106,2009.
- 586 [7] N. Badshah and K. Chen. Image selective segmentation under geometrical constraints using
 587 an active contour approach. *Commun. Comput. Phys.*, 7(4), pp:759–778, 2010.
- 588 [8] E. Bae, X.C. Tai and W. Zhu. Augmented lagrangian method for an euler’s elastica based
 589 segmentation model that promotes convex contours. *Inverse Problems and Imaging*, 11(1),
 590 pp:1–23, 2017.
- 591 [9] I. N. Bankman, T.Nizialek, I.Simon, O. B.Gatewood, I. N.Weinberg, and W. R. Brody.
 592 Segmentation algorithms for detecting microcalcifications in mammograms. *IEEE Trans.*
 593 *Inform. Techn. Iomed.* 1(2), pp:161–173, 1993.
- 594 [10] S. Beucher. Segmentation Tools in Mathematical Morphology. *SPIE 1350, Image Algebra*
 595 *and Morphological Image Processing*, pp:70–84, 1990.

- 596 [11] X. Bresson, S. Esedoglu, P. Vandergheynst, J. P. thiran, and S. Osher. Fast global mini-
597 mization of the active contour/snake model. *Journal of Mathematical Imaging Vision*, 28,
598 pp:151–167, 2007.
- 599 [12] X. Cai, R. Chan, and T. Zeng. A two-stage image segmentation method using a convex
600 variant of the Mumford-Shah model and thresholding. *SIAM Journal of Imaging Sciences*,
601 6, pp:368–390, 2013.
- 602 [13] J. Carter. Dual methods for total variation-based image restoration. PhD thesis,
603 UCLA,2001.
- 604 [14] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of*
605 *Computer Vision*, 22(1), pp:61–79, 1997.
- 606 [15] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of*
607 *Mathematical Imaging and Vision*, 20(1-2), pp:89–97, 2004.
- 608 [16] T. F. Chan and K. Chen. An optimization-based multilevel algorithm for total variation
609 image denoising, *Multiscale Model. Simul.*, 5, pp: 615–645, 2006.
- 610 [17] R. H. Chan and K. Chen. Multilevel algorithm for a Poisson noise removal model with
611 total variation regularisation, *Int. J. Comput. Math.*, 84, pp: 1183–1198, 2007.
- 612 [18] R. H. Chan and K. Chen. A Multilevel algorithm for simultaneously denoising and deblur-
613 ring images, *SIAM J.Sci.Comput.*, 32, pp:1043–1063, 2010.
- 614 [19] T.F. Chan, S. Esedoglu, and M. Nikilova. Algorithm for finding global minimizers of image
615 segmentation and denoising models. *SIAM Journal on Applied Mathematics*, 66, pp:1932–
616 1648, 2006.
- 617 [20] T.F. Chan, G.Golub, and P. Mulet. A nonlinear primal-dual method for total variation-
618 based image restoration. *SIAM Journal on Scientific Computing*, 20(6), pp:1964–1977, 1999.
- 619 [21] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on Image*
620 *Processing*, 10(2), pp:266–277, 2001.
- 621 [22] D. Chen, M. Yang, and L. D. Cohen. Global minimum for a variant mumford-shah model
622 with application to medical image segmentation. *Computer Methods in Biomechanics and*
623 *Biomedical Engineering: Imaging and Visualization*,1(1), pp:48–60, 2013.
- 624 [23] D. Comaniciu, and P. Meer. Mean shift: A robust approach toward feature space analysis.
625 *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, pp:603–619, 2002.
- 626 [24] M. Comer, C. Bouman, and J. Simmons. Statistical Methods for Image Segmentation and
627 Tomography Reconstruction. *Microscopy and Microanalysis*, 16(S2), pp:1852-1853, 2010.
- 628 [25] T. Dietenbeck, M. Alessandrini, D. Friboulet, O. Bernard. CREASEG: a free software for
629 the evaluation of image segmentation algorithms based on level-set. In *IEEE International*
630 *Conference On Image Processing*, Hong Kong, China, pp:665–668, 2010.
- 631 [26] J. Duan, W. Lu, Z. Pan and L. Bai. New second order Mumford-Shah model based on
632 Γ -convergence approximation for image processing. *Infrared Physics and Technology*, 76,
633 pp:641–647, 2016.
- 634 [27] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions and the bayesian re-
635 storation of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6),
636 pp:721–741, 1984.

- 637 [28] C. Gout, C. Le Guyader, and L.A. Vese. Segmentation under geometrical conditions with
638 geodesic active contour and interpolation using level set methods. *Numerical Algorithms*,
639 39, pp:155-173, 2005.
- 640 [29] C.L. Guyader and C. Gout. Geodesic active contour under geometrical conditions theory
641 and 3D applications. *Numerical Algorithms*, 48, pp:105-133, 2008.
- 642 [30] M. Hadhoud, M. Amin, and W. Dabbour. Detection of Breast Cancer Tumor Algorithm
643 Using Mathematical Morphology and Wavelet Analysis. In *Proceedings of GVIP 05 Confe-*
644 *rence, CICC. Cairo, Egypt, 2005.*
- 645 [31] S. Hanov. Wavelets and edge detection (preprint). stevehanov.ca/cs698_wavelet, April 2006.
- 646 [32] A. K. Jumaat and K. Chen. An optimization-based multilevel algorithm for variational
647 image segmentation models. *Electronic Transactions on Numerical Analysis*, 46, pp: 474-
648 504, 2017.
- 649 [33] M. Kass, M. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. *International*
650 *Journal of Vision*, 1, pp:321-331, 1987.
- 651 [34] A. Kenigsberg, R. Kimmel, and I. Yavneh. A Multigrid Approach for Fast Geodesic Active
652 Contours, *Comput. Sci. Depart., The Technion—Israel Int. Technol., Haifa, Tech. Rep. CIS-*
653 *2004-06, 2004.*
- 654 [35] C. Liu, M. K. P. Ng, and T. Zeng. Weighted variational model for selective image segmen-
655 tation with application to medical images. *Pattern Recognition*, 16, pp:367-379, 2018.
- 656 [36] J. Malik, T. Leung, and J. Shi. Contour and texture analysis for image segmentation.
657 *International Journal of Computer Vision*, 43, pp:7-27, 2001.
- 658 [37] S. Mallat. *A Wavelet Tour Of Signal Processing*. Academic Press, USA, 1998.
- 659 [38] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural
660 images and its application to evaluating segmentation algorithms and measuring ecological
661 statistics. *Proc. 8th Int. Conf. Computer Vision*, pp: 416-423, 2001.
- 662 [39] J. Mille, R. Bone, P. Makris, and H. Cardot. Narrow band region-based active contours
663 and surfaces for 2D and 3D segmentation. *Computer Vision and Image Understanding*, 113,
664 pp:946-965, 2009.
- 665 [40] J. C. Moreno, V. B. S. Prasath, H. Proenca, and K. Palaniappan. Fast and globally convex
666 multiphase active contours for brain MRI. *Computer Vision and Image Undrstanding*, 125,
667 pp:237-250, 2014.
- 668 [41] T. N. A. Nguyen, J. Cai, J. Zhang, and J. Zheng. Robust interactive image segmentation
669 using convex active contours. *IEEE Trans. on Image Processing*, 21(8), pp:3734-3743, 2012.
- 670 [42] S. Morigi, L. Reichel and F. Sgallari, Noise-reducing cascadic multilevel methods for linear
671 discrete ill-posed problems. *Numerical Algorithms*, 53, pp:1-22, 2010.
- 672 [43] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and asso-
673 ciated variational problems. *Communications on Pure Applied Mathematics*, 42, pp:577-685,
674 1989.
- 675 [44] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several*
676 *Variables*, Academic Press, New York, London, 1970.

- 677 [45] G. Papandreou and P. Maragos. A fast multigrid implicit algorithm for the evolution of
678 geodesic active contours. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Wa-
679 shington, DC, 2, pp: 689–694, 2004.
- 680 [46] G. Papandreou and P. Maragos. Multigrid geometric active contour models. IEEE Tran-
681 sactions on Image Processing, 16(1), pp: 229–240, Jan. 2007.
- 682 [47] L. Rada and K. Chen. Improved Selective Segmentation Model using One Level set. Journal
683 of Algorithm & Computational Technology, 7(4), pp: 509-541, 2013.
- 684 [48] D. Sen and S. K. Pal. Histogram thresholding using fuzzy and rough measures of association
685 error. IEEE Transactions on Image Processing, 18(4), pp:879–888, 2009.
- 686 [49] J. A. Sethian. Level Set Methods and Fast Marching Methods: Evolving Interfaces in Com-
687 putational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge
688 University Press, 1999.
- 689 [50] A. I. Shihab. Fuzzy Clustering Algorithms and Their Application to Medical Image Ana-
690 lysis. PhD dissertation, Dept. of Computing, Univ. of London, 2000.
- 691 [51] J. Spencer. Variational methods for image segmentation. PhD thesis, University of Liver-
692 pool, 2016.
- 693 [52] J. Spencer and K. Chen. A convex and selective variational model for image segmentation.
694 Communication in Mathematical Sciences, 13(6), pp: 1453–1472, 2015.
- 695 [53] J. S. Weszka. A Survey of Threshold Selection Techniques. Computer Graphics and Image
696 Proc. 7, pp: 259–265, 1978.
- 697 [54] J. Zhang, K.Chen, B.Yu, and D.Gould. A local information based variational model for
698 selective image segmentation. J. Inverse Problems and Imaging, 8, pp: 293–320, 2014.