Communications on Applied Mathematics and Computation
https://doi.org/10.1007/s42967-023-00339-w

ORIGINAL PAPER

An Efficient Smoothing and Thresholding Image Segmentation Framework with Weighted Anisotropic-Isotropic Total Variation

Kevin Bui[1] · Yifei Lou[2] · Fredrick Park[3] · Jack Xin[1]

Received: 13 October 2022 / Revised: 28 September 2023 / Accepted: 9 October 2023

Abstract

In this paper, we design an efficient, multi-stage image segmentation framework that incorporates a weighted difference of anisotropic and isotropic total variation (AITV). The segmentation framework generally consists of two stages: smoothing and thresholding, thus referred to as smoothing-and-thresholding (SaT). In the first stage, a smoothed image is obtained by an AITV-regularized Mumford-Shah (MS) model, which can be solved efficiently by the alternating direction method of multipliers (ADMMs) with a closed-form solution of a proximal operator of the $\ell_1 - \alpha\ell_2$ regularizer. The convergence of the ADMM algorithm is analyzed. In the second stage, we threshold the smoothed image by K-means clustering to obtain the final segmentation result. Numerical experiments demonstrate that the proposed segmentation framework is versatile for both grayscale and color images, efficient in producing high-quality segmentation results within a few seconds, and robust to input images that are corrupted with noise, blur, or both. We compare the AITV method with its original convex TV and nonconvex $\mathrm{TV}^p(0 < p < 1)$ counterparts, showcasing the qualitative and quantitative advantages of our proposed method.

Keywords Image segmentation · Non-convex optimization · Mumford-Shah (MS) model · Alternating direction method of multipliers (ADMMs) · Proximal operator

✉ Kevin Bui
kevinb3@uci.edu

Yifei Lou
yflou@unc.edu

Fredrick Park
fpark@whittier.edu

Jack Xin
jxin@math.uci.edu

[1] Department of Mathematics, University of California, Irvine, Irvine, CA 92697-3875, USA

[2] Department of Mathematics, University of North Carolina, Chapel Hill, Chapel Hill, NC 27599, USA

[3] Department of Mathematics and Computer Science, Whittier College, Whittier, CA 90602, USA

Published online: 24 January 2024

Mathematics Subject Classification 65K10 · 68U10 · 90C26 · 90C90

1 Introduction

Image segmentation is a prevalent, challenging problem in computer vision, aiming to partition an image into several regions that represent specific objects of interest. Each partitioned region has similar features such as edges, colors, and intensities. One segmentation method is the Mumford-Shah (MS) model [63] well known for its robustness to noise. It finds the optimal piecewise-smooth approximation of an input image that incorporates region and boundary information to facilitate segmentation. Given a bounded, open set $\Omega \subset \mathbb{R}^2$ with Lipschitz boundary and an observed image $f\colon \Omega \to [0,1]$, the MS model can be expressed as an energy minimization problem,

$$\min_{u,\Gamma} E_{\mathrm{MS}}(u,\Gamma) := \frac{\lambda}{2}\int_{\Omega}(f-u)^2\,\mathrm{d}x + \frac{\mu}{2}\int_{\Omega\setminus\Gamma}|\nabla u|^2\,\mathrm{d}x + \mathrm{Length}(\Gamma), \tag{1}$$

where $\lambda, \mu > 0$ are weighing parameters, $\Gamma \subset \Omega$ is a compact curve representing the boundaries separating disparate objects, and $u\colon \Omega \to \mathbb{R}$ is an approximation of f that is smooth in $\Omega \setminus \Gamma$ but possibly discontinuous across Γ. The middle term $\int_{\Omega\setminus\Gamma}|\nabla u|^2\,\mathrm{d}x$ ensures that u is piecewise-smooth, or more specifically differentiable on $\Omega \setminus \Gamma$. The last term "Length(Γ)" measures the perimeter of Γ that can be mathematically expressed as $\mathcal{H}^1(\Gamma)$, which is the one-dimensional Hausdorff measure in $\mathbb{R}^2$ [4]. It is challenging to solve for the minimization problem (1) due to its nonconvex nature and difficulties in discretizing the unknown set of boundaries. Pock et al. [70] proposed a convex relaxation of (1) together with an efficient primal-dual algorithm. For the boundary issue, one early attempt involved a sequence of (local) elliptic variational problems [2] to approximate the energy functional (1). Later, nonlocal approximations were adopted in [14, 36] and a finite element approximation was developed in [15].

By relaxing u from piecewise-smooth to piecewise-constant, Chan and Vese (CV) [20] proposed a two-phase model to segment the image domain Ω into two regions that are inside and outside of the curve Γ. The curve can be represented by a level-set function ϕ that is Lipschitz continuous and satisfies

$$\begin{cases} \phi(x) > 0 & \text{if } x \text{ is inside } \Gamma,\\ \phi(x) = 0 & \text{if } x \text{ is at } \Gamma,\\ \phi(x) < 0 & \text{if } x \text{ is outside } \Gamma. \end{cases}$$

The Heaviside function $H(\phi)$ is defined by $H(\phi) = 1$ if $\phi \geqslant 0$ and $H(\phi) = 0$ otherwise. The CV model is given by

$$\begin{aligned} \min_{c_1,c_2,\phi} E_{\mathrm{CV}}(c_1,c_2,\phi) :=& \lambda\int_{\Omega}|f-c_1|^2 H(\phi)\,\mathrm{d}x + \lambda\int_{\Omega}|f-c_2|^2(1-H(\phi))\,\mathrm{d}x\\ &+ \nu\int_{\Omega}|\nabla H(\phi)|\,\mathrm{d}x, \end{aligned} \tag{2}$$

where λ, ν are two positive parameters and $c_1, c_2 \in \mathbb{R}$ are mean intensity values of the two regions. Originally, the CV model (2) was solved by finite difference methods [21, 35]. Later Chan et al. [19] formulated a convex relaxation of CV so that it can be solved by convex optimization techniques such as the split Bregman [37, 38], alternating direction method of multipliers (ADMMs) [8], and the primal-dual hybrid gradient (PDHG) [16, 33]. As an alternative to the level-set formulation (2), a diffuse-interface approximation to the CV model was considered in [32], which can be solved efficiently by the Merrimen-Bence-Osher scheme [62]. The (two-phase) CV model can be naively extended to the multiphase segmentation [77] but with a limitation that it can only deal with the power-two number of segmentation regions. The multiphase CV model was later combined with fuzzy membership functions [50] to segment an arbitrary number of regions.

Another approach of finding a piecewise-constant solution to the MS model is the smoothing-and-thresholding (SaT) framework [12]. In SaT, one first finds a smoothed image u by solving a convex variant of the MS model:

$$\min_u \frac{\lambda}{2}\int_\Omega (f - Au)^2 \,\mathrm{d}x + \frac{\mu}{2}\int_\Omega |\nabla u|^2 \,\mathrm{d}x + \int_\Omega |\nabla u| \,\mathrm{d}x, \tag{3}$$

where $\lambda > 0, \mu > 0$, and A is a linear operator. Specifically, A is the identity operator if one wants to segment a noisy image f, while it can be a blurring operator for the desire of segmenting a blurry and noisy image f. The middle term $\int_\Omega |\nabla u|^2 \,\mathrm{d}x$ extends the piecewise-smooth regularization $\int_{\Omega\setminus\Gamma} |\nabla u|^2 \,\mathrm{d}x$ in (1) to the entire image domain Ω. The last term $\int_\Omega |\nabla u| \,\mathrm{d}x$ is the total variation (TV) that approximates the length term in (1) based on the coarea formula [19]. After obtaining a piecewise-smooth approximation, one segments the image domain into K regions by thresholding u with $K-1$ appropriately selected values. The SaT has several advantages over the MS model (1) and the CV model (2). First, the smoothing stage involves a strictly convex problem (3) to guarantee a unique solution that can be found by numerous convex optimization algorithms. Second, the thresholding stage allows for segmenting any number of regions via a clustering algorithm such as K-means clustering [3, 41]. Lastly, thresholding is independent of smoothing; in other words, thresholding can be adjusted to obtain a visually appealing segmentation without going back to smoothing again. The SaT was adapted to segment images corrupted by Poisson or multiplicative Gamma noise [17]. For color images, the SaT was extended to quaternion space [84] or evolved into the "smoothing, lifting, and thresholding" (SLaT) framework [11]. The additional lifting stage in SLaT adds the Lab (perceived lightness, red-green, and yellow-blue) color space to provide more discriminatory information than the conventional RGB color space with correlated color channels. The idea of lifting can also improve image segmentation of grayscale images whose pixel intensities vary dramatically, referred to as *intensity inhomogeneity*. Traditional methods that deal with inhomogeneity include preprocessing [43] and intensity correction [49, 82]. By generating an additional image channel [54], SaT/SLaT yields better segmentation results for grayscale images that suffer from intensity inhomogeneity.

Note that the convex approximation of the length term in (1) by $\int_\Omega |\nabla u| \,\mathrm{d}x$ in (3) is not optimal, since the Hausdorff measure is nonconvex. For a better approximation, Wu et al. [85] adopted a nonconvex term $\int_\Omega |\nabla u|^p \,\mathrm{d}x$ for $(0 < p < 1)$, referred to as TV^p, leading to a nonconvex problem:

$$\min_u \frac{\lambda}{2}\int_\Omega (f-Au)^2\,\mathrm{d}x + \frac{\mu}{2}\int_\Omega |\nabla u|^2\,\mathrm{d}x + \int_\Omega |\nabla u|^p\,\mathrm{d}x. \tag{4}$$

If $p=1$, TV^p becomes the TV model. Generally, TV^p outperforms TV in image restoration and segmentation [26, 42, 47, 55, 90]. The TV^p regularization originated from the ℓ_p quasinorm, which is more effective than the convex ℓ_1 norm in recovering sparse signals from an underdetermined linear system [24, 87]. Recently, a series of works [56, 58, 88] has demonstrated through experiments that the nonconvex regularizer $\ell_1-\ell_2$ outperforms ℓ_1 and ℓ_p when the linear system is highly coherent. The $\ell_1-\ell_2$ model can be generalized to $\ell_1-\alpha\ell_2$ for $\alpha\in[0,1]$ to allow for sparsity control via the parameter α. Theoretical analyses of the $\ell_1-\alpha\ell_2$ family have been investigated in [31, 34, 53, 88] that justify its superior performances. When applying $\ell_1-\alpha\ell_2$ on the image gradient, Lou et al. [59] proposed a weighted difference of anisotropic and isotropic TV (AITV) that yields better results over TV and TV^p for image denoising and deconvolution. AITV is robust against impulsive noise for image reconstruction [53], and it yields satisfactory segmentation results in the CV model and the fuzzy region competition model [9]. Recently, an AITV-based segmentation model was discussed in [86]. However, these models are solved by a difference-of-convex algorithm (DCA) [48, 67, 68] that requires solving a TV-type subproblem iteratively, thus being computationally expensive.

In this paper, we propose an efficient ADMM framework to solve the AITV variant of (3) and demonstrate its efficiency and effectiveness in the SaT/SLaT framework through various numerical experiments. The efficiency lies in the closed-form solution [57] of the proximal operator for $\ell_1-\alpha\ell_2$ to avoid nested loops in DCA as considered in [9, 86]. The main contributions of this paper are summarized as follows.

i. We provide model analysis such as the coerciveness and the existence of global minimizers for the AITV-regularized variant of (3).
ii. We develop an efficient ADMM algorithm for minimizing the AITV-based MS model based on the proximal operator of $\ell_1-\alpha\ell_2$ with a convergence guarantee.
iii. We conduct extensive numerical experiments to showcase that the SaT/SLaT framework with AITV regularization is a competitive segmentation method, especially using our proposed ADMM algorithm. The segmentation framework is robust to noise, blur, and intensity inhomogeneity.
iv. We demonstrate experimentally that the proposed ADMM framework is significantly more efficient than DCA used in [9, 86] in producing segmentation results of comparable or even better quality.

The paper is organized as follows. Section 2 summarizes mathematical notations and reviews the SaT/SLaT framework. Section 3 provides the analysis of the AITV-regularized MS model that can be solved by ADMM. The convergence analysis of the algorithm subsequently follows. Section 4 presents extensive experiments on various grayscale and color images, comparing the AITV SaT/SLaT framework to other state-of-the-art segmentation methods to demonstrate the effectiveness of the proposed approaches. Lastly, we conclude the paper in Sect. 5.

2 Preliminaries

2.1 Notations

For simplicity, we adopt the discrete notations for images and mathematical models. Without loss of generality, an image is represented as an $M \times N$ matrix, so the image domain is $\Omega = \{1, 2, \cdots, M\} \times \{1, 2, \cdots, N\}$. Then, we denote $X := \mathbb{R}^{M \times N}$. We adopt the linear index for the 2D image, where for $u \in X$, we have $u_{i,j} \in \mathbb{R}$ to be the $((i-1)M+j)$th component of u. The gradient operator $\nabla \colon X \to X \times X$ is denoted by $\nabla u = (\nabla_x u, \nabla_y u)$ with ∇_x and ∇_y being the horizontal and vertical forward difference operators, respectively, with the periodic boundary condition. Specifically, the (i, j)th entry of ∇u is defined by

$$(\nabla u)_{i,j} = \begin{bmatrix} (\nabla_x u)_{i,j} \\ (\nabla_y u)_{i,j} \end{bmatrix},$$

where

$$(\nabla_x u)_{i,j} = \begin{cases} u_{i,j} - u_{i,j-1} & \text{if } 2 \leqslant j \leqslant N, \\ u_{i,1} - u_{i,N} & \text{if } j = 1, \end{cases}$$

and

$$(\nabla_y u)_{i,j} = \begin{cases} u_{i,j} - u_{i-1,j} & \text{if } 2 \leqslant i \leqslant M, \\ u_{1,j} - u_{M,j} & \text{if } i = 1. \end{cases}$$

For $p = (p_x, p_y) \in X \times X$, its $((i-1)M+j)$th component is $p_{i,j} = \begin{bmatrix} (p_x)_{i,j} \\ (p_y)_{i,j} \end{bmatrix} \in \mathbb{R}^2$. We define the following norms on $X \times X$:

$$\|p\|_1 = \sum_{i=1}^{M} \sum_{j=1}^{N} \left(|(p_x)_{i,j}| + |(p_y)_{i,j}| \right),$$

$$\|p\|_2 = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} |(p_x)_{i,j}|^2 + |(p_y)_{i,j}|^2},$$

$$\|p\|_{2,1} = \sum_{i=1}^{M} \sum_{j=1}^{N} \sqrt{(p_x)_{i,j}^2 + (p_y)_{i,j}^2}.$$

Lastly, the proximal operator for a function $f \colon \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ at $y \in \mathbb{R}^n$ is given by

$$\operatorname{prox}_f(y) = \operatorname*{arg\,min}_{x \in \mathbb{R}^n} f(x) + \frac{1}{2}\|x - y\|_2^2.$$

2.2 Review of SaT/SLaT

Both SaT and SLaT frameworks consist of two general steps: (i) smoothing to extract a piecewise-smooth approximation of a given image and (ii) thresholding to segment the regions via K-means clustering. SLaT has an intermediate stage called lifting, which generates additional color channels as opposed to the RGB color space for the smoothed image. More details for each stage are described below.

2.2.1 First Stage: Smoothing

Let $f = (f_1, \cdots, f_d) \in X^d$, where d represents the number of channels in the image f. For example, when the image f is grayscale, we have $d = 1$, and when it is color, we have $d = 3$. In general, f can be a multichannel image. Some of its channels could be generated from the original image to provide more information for segmentation. For example, the intensity inhomogeneity image [54] is generated as an additional channel that quantifies the amount of intensity inhomogeneity in the original image.

The discretized model of (3) for each channel $\ell = 1, \cdots, d$ can be expressed as

$$\min_{u_\ell} \frac{\lambda}{2}\|f_\ell - Au_\ell\|_2^2 + \frac{\mu}{2}\|\nabla u_\ell\|_2^2 + \|\nabla u_\ell\|_{2,1}, \tag{5}$$

where $\lambda > 0$, $\mu > 0$, and $\|\nabla u_\ell\|_2^2$ is a smoothing term to reduce the staircase effects caused by the isotropic TV $\|\nabla u_\ell\|_{2,1}$. We assume the same pair of parameters (λ, μ) across channels. In summary, we obtain a smooth approximation u_ℓ for each channel f_ℓ by solving (5).

2.2.2 Intermediate Stage: Lifting

For a color image $f = (f_1, f_2, f_3) \in X^3$, where f_1, f_2, and f_3 are the red, green, and blue channels, respectively, we can obtain (u_1, u_2, u_3) by applying the smoothing stage to each channel of f. Instead of using (u_1, u_2, u_3), SLaT transforms (u_1, u_2, u_3) into $(\bar{u}_1, \bar{u}_2, \bar{u}_3)$ in the Lab space (perceived lightness, red-green, and yellow-blue) [60] and operates on a new vector-valued image $(u_1, u_2, u_3, \bar{u}_1, \bar{u}_2, \bar{u}_3)$. The rationale is that RGB channels are highly correlated, while the Lab space relies on numerical color differences to approximate the color differences perceived by the human eye. As a result, $(u_1, u_2, u_3, \bar{u}_1, \bar{u}_2, \bar{u}_3)$ leads to better segmentation results compared to (u_1, u_2, u_3).

2.2.3 Final Stage: Thresholding

After rescaling the image obtained after smoothing and/or lifting, we denote the resultant image by $u^* \in [0, 1]^D$. For example, we have $D = 1$ when applying SaT to a grayscale image, and we have $D = 6$ when applying SLaT to a color image. Suppose the number of segmented regions is given and denoted by K. The thresholding stage applies K-means clustering to the vector-valued image u^*, providing K centroids $c_1, c_2, \cdots, c_K$ as constant vectors. These centroids are used to form the regions

$$\Omega_k = \left\{ (i,j) \in \Omega\colon \|u^*_{i,j} - c_k\|_2 = \min_{1 \leqslant \kappa \leqslant K} \|u^*_{i,j} - c_\kappa\|_2 \right\}$$

for $k = 1, \cdots, K$ such that Ω_k's are disjoint and $\bigcup_{k=1}^K \Omega_k = \Omega$. Using the centroids and regions, we can obtain a piecewise-constant approximation of f, denoted by

$$\tilde{f} = (\tilde{f}_1, \cdots, \tilde{f}_d) \text{ such that } \tilde{f}_\ell = \sum_{k=1}^{K} c_{k,\ell} \mathbb{1}_{\Omega_k}, \ \forall \ell = 1, \cdots, d, \tag{6}$$

where $c_{k,\ell}$ is the ℓth entry of c_k and

$$\mathbb{1}_{\Omega_k} = \begin{cases} 1 & \text{if } (i,j) \in \Omega_k, \\ 0 & \text{if } (i,j) \notin \Omega_k. \end{cases}$$

Recall that $d = 1$ when the image f is grayscale and $d = 3$ when it is color.

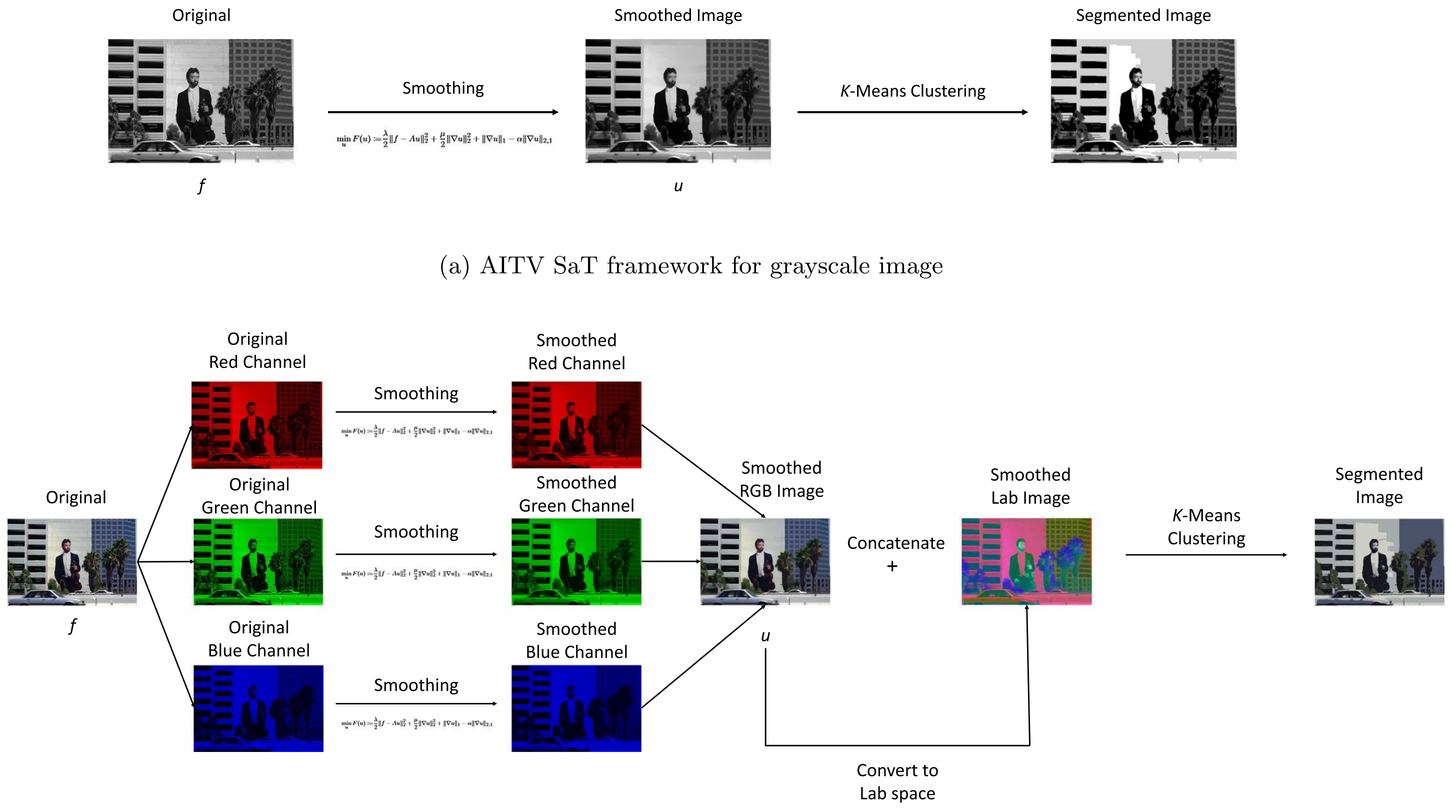
3 Smoothing with AITV Regularization

We replace the isotropic TV in (5) by a weighted difference of anisotropic and isotropic TV, i.e.,

$$\min_u F(u) := \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|\nabla u\|_1 - \alpha\|\nabla u\|_{2,1} \tag{7}$$

with $\lambda > 0, \mu > 0, \alpha \in [0, 1]$. AITV is a more suitable alternative to TV (no matter whether it is anisotropic or isotropic) since TV typically fails to recover oblique edges [7, 28], which can be preserved by AITV [9, 59]. To simplify notations, we omit the subscript ℓ in (5) because the smoothing model is applied channel by channel independently. We show that our model (7) admits a global solution in Sect. 3.1. To find a solution to (7), we describe in Sect. 3.2 the ADMM scheme with its convergence analysis conducted in Sect. 3.3. The overall AITV SaT/SLaT framework for segmentation is visualized in Fig. 1 and summarized in Algorithm 1.

```
Algorithm 1: AITV SaT/SLaT
Input:
  • image $f = (f_1, \cdots, f_d)$
  • blurring operator $A$
  • fidelity parameter $\lambda > 0$
  • smoothing parameter $\mu > 0$
  • AITV parameter $\alpha \in [0, 1]$
  • the number of regions in the image $K$
Output: segmentation $\tilde{f}$
Stage one: compute $u_\ell$ by solving (7) for $\ell = 1, \cdots, d$.
Stage two:
if $f$ is a color image, i.e., $d = 3$ then
    transfer $u = (u_1, u_2, u_3)$ into Lab space to obtain $(\bar{u}_1, \bar{u}_2, \bar{u}_3)$ and
    concatenate to form $(u_1, u_2, u_3, \bar{u}_1, \bar{u}_2, \bar{u}_3)$.
else
    go to stage three.
Stage three: apply $K$-means to obtain $\{(c_l, \Omega_k)\}_{k=1}^{K}$ and compute $\tilde{f}$
  by (6).
```


Original
f
Smoothing
$\min_u F(u) := \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|\nabla u\|_1 - \alpha\|\nabla u\|_{2,1}$
Smoothed Image
u
K-Means Clustering
Segmented Image
Original
f
Original Red Channel
Original Green Channel
Original Blue Channel
Smoothing
$\min_u F(u) := \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|\nabla u\|_1 - \alpha\|\nabla u\|_{2,1}$
Smoothing
$\min_u F(u) := \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|\nabla u\|_1 - \alpha\|\nabla u\|_{2,1}$
Smoothing
$\min_u F(u) := \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|\nabla u\|_1 - \alpha\|\nabla u\|_{2,1}$
Smoothed Red Channel
Smoothed Green Channel
Smoothed Blue Channel
Smoothed RGB Image
u
Concatenate
+
Convert to Lab space
Smoothed Lab Image
K-Means Clustering
Segmented Image

(a) AITV SaT framework for grayscale image

(b) AITV SLaT framework for color image

Fig. 1 AITV SaT/SLaT framework visualized

3.1 Model Analysis

In Theorem 1, we establish the existence of a global solution to (7) by showing that its objective function F is coercive in Lemma 1.

Lemma 1 *If* $\lambda > 0, \mu > 0, \alpha \in [0, 1]$, *and* $\ker(A) \cap \ker(\nabla) = \{0\}$, *then* F *defined in* (7) *is coercive.*

Proof We prove by contradiction. Suppose there exists a sequence $\{u_n\}_{n=1}^{\infty}$ and a constant $C > 0$ such that $\|u_n\|_2 \to \infty$ and $F(u_n) < C$ for all $n \in \mathbb{N}$. We define a sequence $\{v_n\}_{n=1}^{\infty}$, where $v_n = \frac{u_n}{\|u_n\|_2}$ and thereby satisfies $\|v_n\|_2 = 1$ for all $n \in \mathbb{N}$. Since $\{v_n\}_{n=1}^{\infty}$ is bounded, there exists a convergent subsequence $\{v_{n_k}\}_{k=1}^{\infty}$ such that $v_{n_k} \to v^*$ and $\|v^*\|_2 = 1$.

It follows from $\|\nabla u\|_{2,1} \leqslant \|\nabla u\|_1$ that

$$F(u) \geqslant \frac{\lambda}{2}\|Au - f\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 \geqslant \frac{\lambda}{2}(\|Au\|_2 - \|f\|_2)^2 + \frac{\mu}{2}\|\nabla u\|_2^2.$$

Since $F(u_n) < C$, we have $\|\nabla u_n\|_2 < \sqrt{\frac{2C}{\mu}}$ and $\|Au_n\|_2 < \sqrt{\frac{2C}{\lambda}} + \|f\|_2$. As a result, we have

$$\|Av_{n_k}\|_2 = \frac{\|Au_{n_k}\|_2}{\|u_{n_k}\|_2} < \frac{\sqrt{\frac{2C}{\lambda}} + \|f\|_2}{\|u_{n_k}\|_2},$$
$$\|\nabla v_{n_k}\|_2 = \frac{\|\nabla u_{n_k}\|_2}{\|u_{n_k}\|_2} < \frac{\sqrt{2C}}{\sqrt{\mu}\|u_{n_k}\|_2}.$$

After taking the limit $n_k \to \infty$, we get $\|Av^*\|_2 = 0$ and $\|\nabla v^*\|_2 = 0$, which implies that $v^* = 0$ due to the assumption that $\ker(A) \cap \ker(\nabla) = \{0\}$. However, it contradicts with $\|v^*\|_2 = 1$, and hence F is coercive.

Theorem 1 *If* $\lambda > 0, \mu > 0, \alpha \in [0, 1]$, *and* $\ker(A) \cap \ker(\nabla) = \{0\}$, *then* F *has a global minimizer.*

Proof As F is lower bounded by 0, it has a minimizing sequence $\{u_n\}_{n=1}^{\infty}$. Without loss of generality, we assume $u_1 = 0$. Since F is coercive by Lemma 1, we have $F(u_n) \leqslant F(0) < \infty$, showing that $\{\|\nabla u_n\|_1\}_{n=1}^{\infty}$ and $\{\|Au_n\|_2\}_{n=1}^{\infty}$ are bounded. As $\ker(A) \cap \ker(\nabla) = \{0\}$, we have $\{u_n\}_{n=1}^{\infty}$ shall be bounded. Then, there exists a convergent subsequence $\{u_{n_k}\}_{k=1}^{\infty}$ such that $u_{n_k} \to u^*$. Since A and ∇ are both bounded, linear operators, we have $Au_{n_k} \to Au^*$ and $\nabla u_{n_k} \to \nabla u^*$. Since norms are continuous and thereby lower semi-continuous, we have

$$\begin{aligned}
&\|\nabla u^*\|_1 - \alpha\|\nabla u^*\|_{2,1} \leqslant \liminf_{k\to\infty}\left(\|\nabla u_{n_k}\|_1 - \alpha\|\nabla u_{n_k}\|_{2,1}\right),\\
&\|\nabla u^*\|_2^2 \leqslant \liminf_{k\to\infty}\|\nabla u_{n_k}\|_2^2,\\
&\|Au^* - f\|_2^2 \leqslant \liminf_{k\to\infty}\|Au_{n_k} - f\|_2^2.
\end{aligned}$$

Altogether, we obtain $F(u^*) \leqslant \liminf_{k\to\infty} F(u_{n_k})$, which implies that u^* minimizes $F(u)$.

3.2 Numerical Scheme

We describe an efficient algorithm to minimize (7) via ADMM. In particular, we introduce an auxiliary variable $w = (w_x, w_y) \in X \times X$ and rewrite (7) into an equivalent constrained optimization problem

$$\begin{aligned} \min_{u,w} \quad & \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|w\|_1 - \alpha\|w\|_{2,1} \\ \text{s.t.} \quad & \nabla u = w, \end{aligned} \tag{8}$$

where $w_x = \nabla_x u$ and $w_y = \nabla_y u$. Then, the corresponding augmented Lagrangian is expressed by

$$\begin{aligned} \mathcal{L}_\delta(u, w, z) :=& \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|w\|_1 - \alpha\|w\|_{2,1} \\ &+ \langle z, \nabla u - w\rangle + \frac{\delta}{2}\|\nabla u - w\|_2^2 \\ =& \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \|w\|_1 - \alpha\|w\|_{2,1} \\ &+ \frac{\delta}{2}\left\|\nabla u - w + \frac{z}{\delta}\right\|_2^2 - \frac{1}{2\delta}\|z\|_2^2, \end{aligned} \tag{9}$$

where $\delta > 0$ is a penalty parameter and $z = (z_x, z_y) \in X \times X$ is a dual variable. The ADMM iterations proceed as follows:

$$u_{t+1} \in \arg\min_u \mathcal{L}_{\delta_t}(u, w_t, z_t), \tag{10a}$$

$$w_{t+1} \in \arg\min_w \mathcal{L}_{\delta_t}(u_{t+1}, w, z_t), \tag{10b}$$

$$z_{t+1} = z_t + \delta_t(\nabla u_{t+1} - w_{t+1}), \tag{10c}$$

$$\delta_{t+1} = \sigma\delta_t, \ \sigma \geqslant 1. \tag{10d}$$

Note that $\sigma = 1$ reduces to the original ADMM framework [8]. We consider an adaptive penalty parameter δ_t by choosing $\sigma > 1$. In fact, the parameter $\sigma > 1$ controls the numerical convergence speed of the algorithm in the sense that a larger σ leads to a fewer number of iterations the algorithm needs to run before satisfying a stopping criterion. However, if δ_t increases too quickly, the ADMM algorithm will numerically converge within a few iterations, which may yield a low-quality solution. Thus, a small σ is recommended and we discuss its choice in experiments (Sect. 4).

Next we elaborate on how to solve the two subproblems (10a) and (10b). The subproblem (10a) is written as

$$u_{t+1} \in \arg\min_u \frac{\lambda}{2}\|f - Au\|_2^2 + \frac{\mu}{2}\|\nabla u\|_2^2 + \langle z_t, \nabla u - w_t\rangle + \frac{\delta_t}{2}\|\nabla u - w_t\|_2^2.$$

The first-order optimality condition of (10a) is given by

$$\left[\lambda A^\top A - (\mu + \delta_t)\Delta\right] u_{t+1} = \lambda A^\top f + \delta_t \nabla^\top \left(w_t - \frac{z_t}{\delta_t} \right),$$

where $\Delta = -\nabla^\top \nabla$ is the Laplacian operator. If $\ker(A) \cap \ker(\nabla) = \{0\}$, then $\lambda A^\top A - (\mu + \delta_t)\Delta$ is positive definite. By assuming the periodic boundary condition, $A^\top A$ and Δ are block circulant, so we can solve for u_{t+1} via the fast Fourier transform $\mathcal{F}$ [18, 64, 83]. By the Convolution Theorem, the closed-form solution for u_{t+1} is

$$u_{t+1} = \mathcal{F}^{-1}\left(\frac{\lambda \mathcal{F}(A)^* \circ \mathcal{F}(f) + \delta_t \mathcal{F}(\nabla)^* \circ \mathcal{F}\left(w_t - \frac{z_t}{\delta_t}\right)}{\lambda \mathcal{F}(A)^* \circ \mathcal{F}(A) - (\mu + \delta_t)\mathcal{F}(\Delta)} \right),$$

where $\mathcal{F}^{-1}$ is the inverse Fourier transform, $*$ denotes complex conjugate, $\circ$ denotes componentwise multiplication, and division is also componentwise.

Denote $w_{i,j} = \begin{bmatrix} (w_x)_{i,j} \\ (w_y)_{i,j} \end{bmatrix} \in \mathbb{R}^2$ as the (i, j)th entry of w. The subproblem (10b) can be expressed as

$$w_{t+1} \in \arg\min_w \|w\|_1 - \alpha \|w\|_{2,1} + \frac{\delta_t}{2} \left\| \nabla u_{t+1} + \frac{z_t}{\delta_t} - w \right\|_2^2 .$$

Expanding (10b), we get

$$\arg\min_w \sum_{(i,j)\in\Omega} \left(\|w_{i,j}\|_1 - \alpha \|w_{i,j}\|_2 + \frac{\delta_t}{2} \left\| (\nabla u_{t+1})_{i,j} + \frac{(z_t)_{i,j}}{\delta_t} - w_{i,j} \right\|_2^2 \right), \tag{11}$$

which shows that $w_{i,j}$ can be solved elementwise. Specifically, the optimal solution of $w_{i,j} \in \mathbb{R}^2$ is related to the proximal operator for $\ell_1 - \alpha\ell_2$ defined by

$$\operatorname{prox}(y; \alpha, \beta) := \operatorname{prox}_{\beta(\|\cdot\|_1 - \alpha\|\cdot\|_2)}(y) = \arg\min_x \|x\|_1 - \alpha\|x\|_2 + \frac{1}{2\beta}\|x - y\|_2^2. \tag{12}$$

The closed-form solution for (12) is given in Lemma 2 [57]. By comparing (11) and (12), the w-update is given by, for any $(i,j) \in \Omega$,

$$(w_{t+1})_{i,j} = \operatorname{prox}\left((\nabla u_{t+1})_{i,j} + \frac{(z_t)_{i,j}}{\delta_t}; \alpha, \frac{1}{\delta_t} \right).$$

Lemma 2 ([57]) *Given $y \in \mathbb{R}^n$, $\beta > 0$, and $\alpha \geqslant 0$, the optimal solution to (12) can be discussed separately into the following cases.*

(i) *When $\|y\|_\infty > \beta$, we have*

$$x^* = (\|\xi\|_2 + \alpha\beta)\frac{\xi}{\|\xi\|_2},$$

where $\xi = \operatorname{sign}(y) \circ \max(|y| - \beta, 0)$.

(ii) *When $(1-\alpha)\beta < \|y\|_\infty \leqslant \beta$, then x^* is a one-sparse vector such that one chooses $i \in \arg\max_j(|y_j|)$ and defines $x_i^* = \left(|y_i| + (\alpha - 1)\beta\right)\operatorname{sign}(y_i)$ and the rest of the elements equal to* 0.

(iii) *When* $\|y\|_\infty \leqslant (1-\alpha)\beta$, *then* $x^* = 0$.

In summary, the ADMM scheme that minimizes (7) is presented in Algorithm 2.

```
Algorithm 2: ADMM for minimizing the AITV-Regularized smoothing model
Input:
  • image $f$
  • blurring operator $A$
  • fidelity parameter $\lambda > 0$
  • smoothing parameter $\mu > 0$
  • AITV parameter $\alpha \in [0,1]$
  • penalty parameter $\delta_0 > 0$
  • penalty multiplier $\sigma \geqslant 1$
  • relative error $\epsilon > 0$
Output: $u_t$
Initialize $u_0, w_0, z_0$.
Set $t = 0$.
while $\frac{\|u_t - u_{t-1}\|_2}{\|u_t\|_2} > \epsilon$ do
    $u_{t+1} = \mathcal{F}^{-1}\left(\frac{\lambda\mathcal{F}(A)^* \circ \mathcal{F}(f) + \delta_t \mathcal{F}(\nabla)^* \circ \mathcal{F}\left(w_t - \frac{z_t}{\delta_t}\right)}{\lambda\mathcal{F}(A)^* \circ \mathcal{F}(A) - (\mu + \delta_t)\mathcal{F}(\Delta)}\right)$
    $(w_{t+1})_{i,j} = \text{prox}\left((\nabla u_{t+1})_{i,j} + \frac{(z_t)_{i,j}}{\delta_t};\ \alpha, \frac{1}{\delta_t}\right), \quad \forall (i,j) \in \Omega$
    $z_{t+1} = z_t + \delta_t(\nabla u_{t+1} - w_{t+1})$
    $\delta_{t+1} = \sigma\delta_t$
    $t := t+1$
```

3.3 Convergence Analysis

We aim to analyze the convergence for Algorithm 2. It is true that global convergence of ADMM has been established in [29] for certain classes of nonconvex optimization problems, but unfortunately it cannot be applied to our problem (8) since the gradient operator ∇ is not surjective. Instead of global convergence, we manage to achieve weaker subsequential convergence for two cases: $\sigma = 1$ and $\sigma > 1$. The proof of $\sigma > 1$ is adapted from [39, 89].

Before providing convergence results for ADMM, we provide a definition of subdifferential for general functions. For a function $h\colon \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, we denote the (limiting) subdifferential by $\partial h(x)$ [73, Definition 11.10], which is defined as a set

$$\partial h(x) = \{v \in \mathbb{R}^n\colon \exists \{(x_t, v_t)\}_{t=1}^{\infty} \text{ s.t. } x_t \to x,\ h(x_t) \to h(x),\ v_t \in \hat{\partial} h(x_t), \text{and } v_t \to v\}$$

with

$$\hat{\partial} h(x) = \left\{ v \in \mathbb{R}^n\colon \liminf_{z \to x, z \neq x} \frac{h(z) - h(x) - \langle v, z - x\rangle}{\|z - x\|_2} \geqslant 0 \right\}.$$

Since $\hat{\partial} h(x) \subset \partial h(x)$ where h is finite on x, the graph $x \mapsto \partial h(x)$ is closed [27, 73] by definition

$$v_t \in \partial h(x_t),\ x_t \to x,\ h(x_t) \to h(x),\ v_t \to v \implies v \in \partial h(x).$$

First, we present a lemma and a proposition whose proofs are delayed to the appendix.

Lemma 3 *Suppose that* $\ker(A) \cap \ker(\nabla) = \{0\}$. *Let* $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ *be generated by* (10a)–(10d) *with* $\sigma \geqslant 1$. *The following inequality holds*:

$$\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) - \mathcal{L}_{\delta_t}(u_t, w_t, z_t) \leqslant \frac{\sigma+1}{2\sigma^t \delta_0}\|z_{t+1} - z_t\|_2^2 - \frac{\zeta}{2}\|u_{t+1} - u_t\|_2^2, \qquad (13)$$

where $\zeta > 0$ *is the smallest eigenvalue of* $\lambda A^\top A + (\mu + \delta_0)\nabla^\top \nabla$.

Proposition 1 *Suppose that* $\ker(A) \cap \ker(\nabla) = \{0\}$. *Let* $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ *be generated by* (10a)–(10d). *Assume one of the conditions holds*:

- $\sigma = 1$ *and* $\sum_{i=0}^{\infty} \|z_{i+1} - z_i\|_2^2 < \infty$,
- $\sigma > 1$.

Then, we have the following statements:

(a) *the sequence* $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ *is bounded*,

(b) $u_{t+1} - u_t \to 0$ *as* $t \to \infty$.

Proposition 1 reveals an advantage of using the adaptive penality parameter with $\sigma > 1$. For $\sigma = 1$, we require $\sum_{i=0}^{\infty} \|z_{i+1} - z_i\|_2^2 < \infty$ in order for the iterates $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ of Algorithm 2 to be bounded and to satisfy the relative stopping criterion $\frac{\|u_t - u_{t-1}\|_2}{\|u_t\|_2} < \epsilon$. The requirement $\sum_{i=0}^{\infty} \|z_{i+1} - z_i\|_2^2 < \infty$ is no longer necessary if $\sigma > 1$.

Finally, we establish the subsequential convergence in Theorem 2 under stronger conditions compared to the ones in Proposition 1. These conditions are motivated by a series of works [22, 23, 45, 46, 51, 55] that proved the theoretical convergence of ADMM in solving TV-based inverse problems.

Theorem 2 *Let* $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ *be generated by* (10a)–(10d). *Assume one set of the following conditions holds*:

- $\sigma = 1$ *and* $\sum_{i=0}^{\infty} \|z_{i+1} - z_i\|_2^2 < \infty$,
- $\sigma > 1$, $\delta_t(w_{t+1} - w_t) \to 0$, *and* $z_{t+1} - z_t \to 0$.

Then there exists a subsequence of $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ *whose limit point* (u^*, w^*, z^*) *is a KKT point of* (8) *that satisfies*

$$0 = \lambda A^\top (Au^* - f) - \mu \Delta u^* + \nabla^\top z^*, \qquad (14a)$$

$$z^* \in \partial\big(\|w^*\|_1 - \alpha\|w^*\|_{2,1}\big), \tag{14b}$$

$$\nabla u^* = w^*. \tag{14c}$$

Proof By Proposition 1, $\{(u_t, w_t, z_t)\}_{t=1}^{\infty}$ is bounded, and hence there exists a subsequence that converges to a point (u^*, w^*, z^*), denoted by $(u_{t_k}, w_{t_k}, z_{t_k}) \to (u^*, w^*, z^*)$. Proposition 1 also establishes $\lim_{t\to\infty} u_{t+1} - u_t = 0$, which implies that $\lim_{k\to\infty} u_{t_k+1} = \lim_{k\to\infty} u_{t_k} = u^*$. Either set of assumptions establishes $\lim_{k\to\infty} z_{t_k+1} = \lim_{k\to\infty} z_{t_k} = z^*$. The optimality conditions at iteration t_k are

$$0 = \lambda A^\top(Au_{t_k+1} - f) - \mu\Delta u_{t_k+1} + \delta_{t_k}\nabla^\top(\nabla u_{t_k+1} - w_{t_k}) + \nabla^\top z_{t_k}, \tag{15a}$$

$$0 \in \partial\big(\|w_{t_k+1}\|_1 - \alpha\|w_{t_k+1}\|_{2,1}\big) - \delta_{t_k}\big(\nabla u_{t_k+1} - w_{t_k+1}\big) - z_{t_k}, \tag{15b}$$

$$z_{t_k+1} = z_{t_k} + \delta_{t_k}(\nabla u_{t_k+1} - w_{t_k+1}). \tag{15c}$$

Next we discuss two sets of assumptions individually.

If $\sigma = 1$, then $\delta_{t_k} = \delta_0$ for each iteration t_k. Together with $\lim_{t\to\infty} z_{t+1} - z_t = 0$, we have $\lim_{t\to\infty} \nabla u_t - w_t = 0$ by (10c) and

$$\nabla u^* = \lim_{k\to\infty} \nabla u_{t_k} = \lim_{k\to\infty}(\nabla u_{t_k} - w_{t_k}) + \lim_{k\to\infty} w_{t_k} = w^*,$$

leading to (14c). According to (15a), the point u_{t_k+1} satisfies

$$\begin{aligned} 0 &= \lambda A^\top(Au_{t_k+1} - f) - \mu\Delta u_{t_k+1} + \delta_0\nabla^\top(\nabla u_{t_k+1} - w_{t_k}) + \nabla^\top z_{t_k} \\ &= \lambda A^\top(Au_{t_k+1} - f) - \mu\Delta u_{t_k+1} + \delta_0\nabla^\top(\nabla u_{t_k+1} - \nabla u_{t_k}) + \delta_0\nabla^\top(\nabla u_{t_k} - w_{t_k}) \\ &\quad + \nabla^\top z_{t_k}. \end{aligned}$$

Then (14a) holds after taking $k \to \infty$. Finally, we have

$$\lim_{k\to\infty} w_{t_k+1} = \lim_{k\to\infty}(w_{t_k+1} - \nabla u_{t_k+1}) + \lim_{k\to\infty} \nabla u_{t_k+1} = \lim_{k\to\infty} \nabla u_{t_k} = w^*.$$

If $\sigma > 1$ and $\delta_t(w_{t+1} - w_t) \to 0$, we substitute (15c) into (15a) and simplify it to obtain

$$\begin{aligned} 0 &= \lim_{k\to\infty} \lambda A^\top(Au_{t_k+1} - f) - \mu\Delta u_{t_k} + \delta_{t_k}\nabla^\top(w_{t_k+1} - w_{t_k}) + \nabla^\top z_{t_k+1} \\ &= \lambda A^\top(Au^* - f) - \mu\Delta u^* + \nabla^\top z^*. \end{aligned}$$

We need to prove $\lim_{k\to\infty} w_{t_k+1} = w^*$. Since $\{z_t\}_{t=1}^{\infty}$ is bounded in this case, there exists $C > 0$ such that $\|z_t\|_2 \leqslant C$. By (10c), we have

$$\begin{aligned} \|w_{t+1} - w_t\|_2 &\leqslant \|w_{t+1} - \nabla u_{t+1}\|_2 + \|\nabla u_{t+1} - \nabla u_t\|_2 + \|\nabla u_t - w_t\|_2 \\ &= \left\|\frac{z_{t+1} - z_t}{\delta_t}\right\|_2 + \|\nabla u_{t+1} - \nabla u_t\|_2 + \left\|\frac{z_t - z_{t-1}}{\delta_{t-1}}\right\|_2 \\ &\leqslant \frac{4C}{\delta_{t-1}} + \|\nabla u_{t+1} - \nabla u_t\|_2. \end{aligned}$$

Taking the limit $t \to \infty$, we obtain $\|w_{t+1} - w_t\|_2 \to 0$ and $w_{t+1} - w_t \to 0$. It follows that

$$\lim_{k\to\infty} w_{t_k+1} - w_{t_k} = 0 \implies \lim_{k\to\infty} w_{t_k+1} = \lim_{k\to\infty} w_{t_k} = w^*.$$

Then (15c) implies

$$\|\nabla u^* - w^*\|_2 = \lim_{k\to\infty} \|\nabla u_{t_k+1} - w_{t_k+1}\|_2 = \lim_{k\to\infty} \frac{1}{\delta_{t_k}} \left\| z_{t_k+1} - z_{t_k} \right\|_2 \leqslant \lim_{k\to\infty} \frac{2C}{\delta_{t_k}} = 0.$$

As a result, we have $\nabla u^* = w^*$.

By substituting (15c) into (15b), we have

$$z_{t_k+1} \in \partial\big(\|w_{t_k+1}\|_1 - \alpha\|w_{t_k+1}\|_{2,1}\big), \quad \forall k \in \mathbb{N}.$$

By continuity, we have $\|w_{t_k+1}\|_1 - \alpha\|w_{t_k+1}\|_{2,1} \to \|w^*\|_1 - \alpha\|w^*\|_{2,1}$. Together with the fact that $(w_{t_k+1}, z_{t_k+1}) \to (w^*, z^*)$, we obtain $z^* \in \partial\big(\|w^*\|_1 - \alpha\|w^*\|_{2,1}\big)$.

Therefore, if either set of assumptions holds, then (u^*, w^*, z^*) is a KKT point of (8).

4 Experimental Results

We examine the SaT/SLaT framework by comparing the isotropic TV[1] [11, 12], the TV$^p(0 < p < 1)$ [85], and the AITV. The experiment comparison also includes the AITV-regularized CV and fuzzy region (FR) models [9], the iterative convolution-thresholding method (ICTM) [81], and the TV^p-regularized Mumford-Shah (TV^p MS) model without the bias term [55] together with the Potts model [71] solved by either a primal-dual algorithm[2] [69] or ADMM[3] [74]. In particular, the primal-dual algorithm solves a convex relaxation of the Potts model [69]:

$$U^* = \arg\min_{U\in S} \sum_{k=1}^{K} \left[\lambda \sum_{(i,j)\in\Omega} (u_k)_{i,j} |(u_k)_{i,j} - c_k|^2 + \|\nabla u_k\|_{2,1} \right], \tag{16}$$

where K is the number of regions specified in an image, $\{c_k\}_{k=1}^K \subset \mathbb{R}$ are constant values, and

$$S = \Bigg\{ U = (u_1, u_2, \cdots, u_K) \in X^K :\ \forall (i,j) \in \Omega, \sum_{k=1}^{K} (u_k)_{i,j} = 1; \\ (u_k)_{i,j} \in [0,1], k = 1, \cdots, K \Bigg\}.$$

Once getting U^* from (16), the regions of an image can be approximated by

$$\Omega_\kappa = \left\{ (i,j) \in \Omega :\ \kappa = \arg\max_{1\leqslant k\leqslant K} (u_k^*)_{i,j} \right\}$$

[1] MATLAB code is available at https://xiaohaocai.netlify.app/download/.

[2] Python code is available at https://github.com/VLOGroup/pgmo-lecture/blob/master/notebooks/tv-potts.ipynb and a translated MATLAB code is available at https://github.com/kbui1993/MATLAB_Potts.

[3] Code is available at https://github.com/mstorath/Pottslab.

with $\kappa = 1, \cdots, K$. For short, we refer (16) as the convex Potts model. To apply ADMM, Storath and Weinmann [74] considered the following version of the Potts model:

$$\min_{u} \lambda \|u - f\|_2^2 + \|\nabla u\|_0. \tag{17}$$

Since it does not admit a segmentation result with a chosen number of regions, we develop its SaT version called SaT-Potts that solves (17), followed by the K-means clustering for segmentation. Both (16) and (17) can deal with multichannel input; please refer to [69, 74] for more details.

To ease the parameter tuning, we scale the pixel intensity of all the testing images in our experiments to [0, 1]. Stage 1 of the isotropic TV SaT/SLaT is solved using the authors' official code that is implemented by a similar ADMM algorithm to Algorithm 2 with $\sigma = 1$. Stage 1 of TV^p and AITV SaT/SLaT is solved by Algorithm 2 with $\sigma = 1.25$ using the appropriate proximal operators. We set the penalty parameter in Algorithm 2 to be $\delta_0 = 1.0, 2.0$ for grayscale and multichannel images, respectively. The stopping criterion for the ADMM algorithms is until $\frac{\|u_{t+1} - u_t\|_2}{\|u_{t+1}\|_2} < 10^{-4}$ with a maximum number of 300 iterations. We compare the proposed ADMM algorithm with our own DCA implementation for AITV SaT/SLaT as described in [86]. Note that its inner minimization subproblem is solved by semi-proximal ADMM [40], which has more parameters than ADMM. We use the default parameter setting as suggested in [86].

To quantitatively evaluate the segmentation performance, we use two metrics: DICE index [30] when the ground truth is available and PSNR when the ground truth is unavailable. The DICE index is given by

$$\text{DICE} = 2\frac{\#\{R(i) \cap R'(i)\}}{\#\{R(i)\} + \#\{R'(i)\}},$$

where $R(i)$ is the set of pixels with label i in the ground-truth image f, $R'(i)$ is the set of pixels with label i in the segmented image $\tilde{f}$, and $\#\{R\}$ refers to the number of pixels in the set R. Following the works of [45, 52, 65, 74], we use PSNR to determine how well the segmented image $\tilde{f}$ approximates the original image f. It is computed by $10\log_{10}(1/\text{MSE})$, where MSE is the mean square error between f and $\tilde{f}$.

We tune various parameters in the investigated algorithms to achieve the best DICE indices or PSNRs for synthetic or real images, respectively. The fidelity parameter λ and the smoothing parameter μ are tuned for each image, which will be specified later. For TV^p SaT/SLaT, we only consider $p = 1/2, 2/3$ because they are the only values that have closed-form solutions [13, 87] for their proximal operators. For the AITV related algorithms, we tune $\alpha \in \{0.2, 0.4, 0.6, 0.8\}$. For the SaT-Potts model [74], we use a default setting for the other parameters. For the convex Potts model [69], we run the algorithm for up to 150 iterations with the same stopping criterion as AITV does.

All experiments are performed in MATLAB R2022b on a Dell laptop with a 1.80 GHz Intel Core i7-8565U processor and 16.0 GB of RAM. In the general SaT/SLaT framework, we use some MATLAB built-in functions. In Stage 2, `makecform('srgb2lab')` is used to convert RGB to Lab. In Stage 3, `kmeans` performs K-means++ clustering [3] for up to 100 iterations five times with different initialization and selects the best arrangement among the five solutions. We also parallelize Stage 1 for color, or generally multichannel, images to speed up the computation. To compute DICE and PSNR, we use the MATLAB functions `dice` and `psnr`.

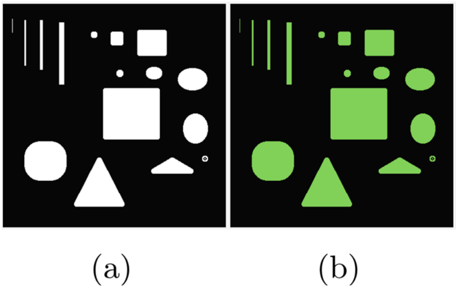


Fig. 2 Synthetic images for two-phase segmentation. **a** Grayscale image. **b** Color image whose regions have pixel value (128, 230, 64). The size of both images is 385 × 385

Table 1 Comparison of the DICE indices and computational times (seconds) between the segmentation methods applied to Fig. 2a corrupted in four cases

	65% RV		65% SP		Blur and 50% RV		Blur and 50% SP	
	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)
(Original) SaT	0.974 8	4.71	0.964 1	5.18	0.955 7	6.15	0.949 8	8.19
TV^p SaT	0.975 1	2.14	0.964 7	2.33	0.953 9	2.74	0.947 5	3.76
AITV SaT (ADMM)	**0.979 3**	2.32	**0.965 8**	2.04	**0.958 1**	2.43	**0.952 2**	2.54
AITV SaT (DCA)	0.978 3	23.22	0.964 4	24.65	0.948 8	40.93	0.943 4	35.23
AITV CV	0.978 6	91.26	0.965 5	121.57	0.932 8	121.11	0.919 0	151.68
ICTM	0.432 2	**0.50**	0.432 1	**0.18**	0.531 9	**0.82**	0.506 5	**0.22**
TV^p MS	0.968 1	4.96	0.953 3	10.14	0.936 9	3.20	0.927 1	6.08
Convex Potts	0.975 5	8.01	0.963 7	6.81	0.910 1	7.99	0.913 2	6.85
SaT-Potts	0.971 4	4.67	0.955 9	4.24	0.930 5	4.39	0.918 0	4.11

Number in bold indicates either the highest DICE index or the fastest time among the segmentation methods for a given corrupted image

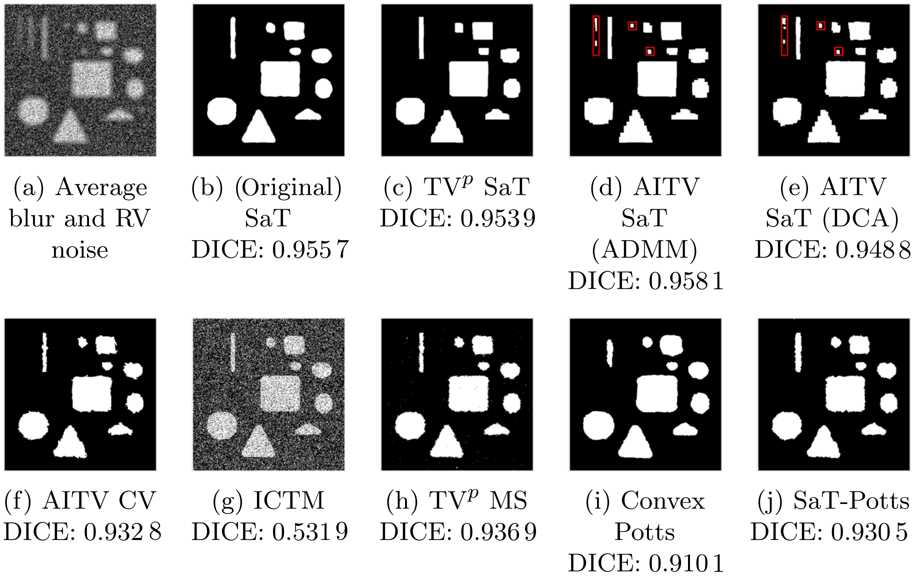
The AITV SaT/SLaT codes are available at https://github.com/kbui1993/Official_AITV_SaT_SLaT.

4.1 Two-Phase Segmentation on Synthetic Images

We compare the proposed ADMM algorithm of AITV SaT/SLaT with the other SaT/SLaT methods, the Potts models, ICTM, TV^p MS, and the AITV CV model on the synthetic images presented in Fig. 2. We corrupt the images with either random-valued (RV) or salt-and-pepper (SP) impulsive noises. Additionally, we consider blurring the image before adding impulsive noises. Specifically, we use an average blur `fspecial` (‘`average`’, `15`) for Fig. 2a and a motion blur `fspecial` (‘`motion`’, `5`, `45`) for Fig. 2b. For the SaT/SLaT methods applied to both images in Fig. 2, we tune the parameters $\lambda \in [1, 10]$ and $\mu \in [0.2, 6]$.

4.1.1 Synthetic Grayscale Images

We apply the competing segmentation methods on four types of input data based on Fig. 2a, i.e., 65% RV noise, 65% SP noise, average blur followed by 50% RV, and average blur



(a) Average blur and RV noise

(b) (Original) SaT DICE: 0.955 7

(c) TV^p SaT DICE: 0.953 9

(d) AITV SaT (ADMM) DICE: 0.958 1

(e) AITV SaT (DCA) DICE: 0.948 8

(f) AITV CV DICE: 0.932 8

(g) ICTM DICE: 0.531 9

(h) TV^p MS DICE: 0.936 9

(i) Convex Potts DICE: 0.910 1

(j) SaT-Potts DICE: 0.930 5

Fig. 3 Segmentation results of Fig. 2a corrupted with average blur followed by 50% RV noise. Regions boxed in red are only identified by AITV SaT

followed by 50% SP. The resulting DICE indices together with the computational times are recorded in Table 1. For all four cases, our proposed AITV SaT (ADMM) achieves the highest DICE indices with generally the second fastest times. The fastest time is attained by ICTM, but it yields the worst results, indicating that it performs poorly on images corrupted by impulsive noise. The AITV CV model, ICTM, TV^p MS model, and the Potts models perform worse than the SaT methods on blurry images because, unlike the SaT methods, they do not account for blurring. Lastly, we point out that solving (7) in the AITV SaT model by ADMM yields higher DICE in significantly less time than by DCA.

Visual segmentation results are presented in Fig. 3 under the RV noise with average blur. Both AITV SaT methods identify the middle rectangle of the three rectangles at the top left corner and the two smallest circles above the middle square in Fig. 3. These regions are enclosed in red boxes. As a result, identifying more regions than the other methods and having a smoother segmentation than its DCA counterpart, AITV SaT (ADMM) has the highest DICE index for this case.

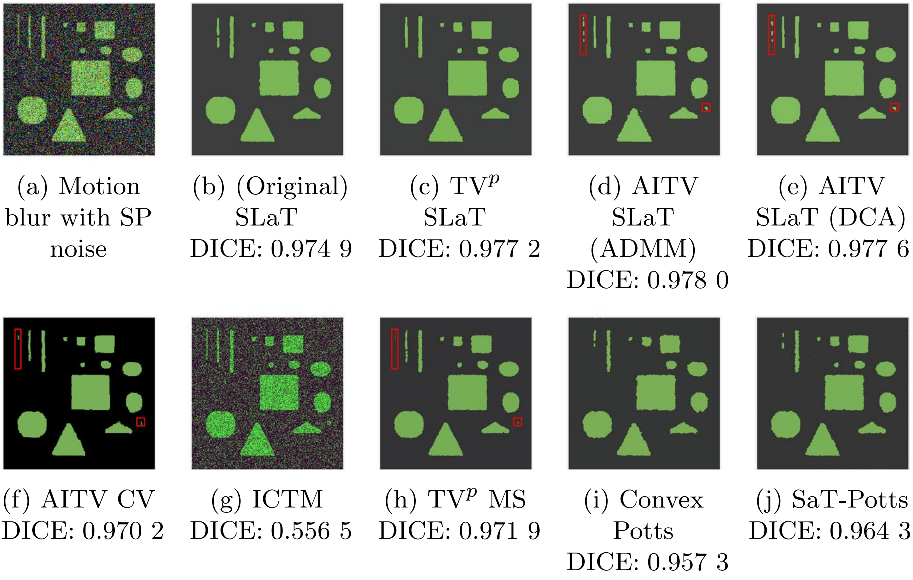
4.1.2 Synthetic Color Images

The (original) color image, Fig. 2b, is corrupted by either 60% impulsive noise or motion blur followed by 45% noise. Table 2 records the DICE indices and the computational times of various segmentation methods applied on all the four cases. For the noisy images without blur, AITV SLaT (ADMM) attains comparable DICE indices as the best AITV CV method and its DCA counterpart but with significantly less computational time. For the blurry, noisy inputs, AITV SLaT (ADMM) attains the highest DICE indices. In general, as an alternative to its DCA counterpart, AITV SLaT (ADMM) gives satisfactory segmentation results under a reasonable amount of time.

Table 2 Comparison of the DICE indices and computational times (seconds) between the segmentation methods applied to Fig. 2b corrupted in four cases

	60% RV		60% SP		Blur and 45% RV		Blur and 45% SP	
	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)
(Original) SLaT	0.981 4	11.74	0.963 7	12.61	0.984 5	11.07	0.974 9	12.17
TV^p SLaT	0.982 2	4.54	0.973 1	4.77	0.986 3	8.05	0.977 2	6.27
AITV SLaT (ADMM)	0.983 9	3.31	0.974 8	4.67	**0.987 2**	6.01	**0.978 0**	6.45
AITV SLaT (DCA)	0.984 9	41.27	0.975 3	47.09	0.986 6	44.54	0.977 6	61.64
AITV CV	**0.989 3**	84.56	**0.980 6**	113.08	0.977 1	92.41	0.970 2	103.99
ICTM	0.478 8	**1.02**	0.458 9	**0.25**	0.578 2	**1.35**	0.556 5	**0.35**
TV^p MS	0.979 9	3.71	0.968 8	54.10	0.979 1	3.71	0.971 9	3.52
Convex Potts	0.962 9	9.20	0.961 4	7.50	0.972 8	7.31	0.957 3	8.10
SaT-Potts	0.980 6	7.66	0.967 2	7.70	0.976 0	6.24	0.964 3	6.53

Number in bold indicates either the highest DICE index or the fastest time among the segmentation methods for a given corrupted image


(a) Motion blur with SP noise

(b) (Original) SLaT DICE: 0.974 9

(c) TV^p SLaT DICE: 0.977 2

(d) AITV SLaT (ADMM) DICE: 0.978 0

(e) AITV SLaT (DCA) DICE: 0.977 6

(f) AITV CV DICE: 0.970 2

(g) ICTM DICE: 0.556 5

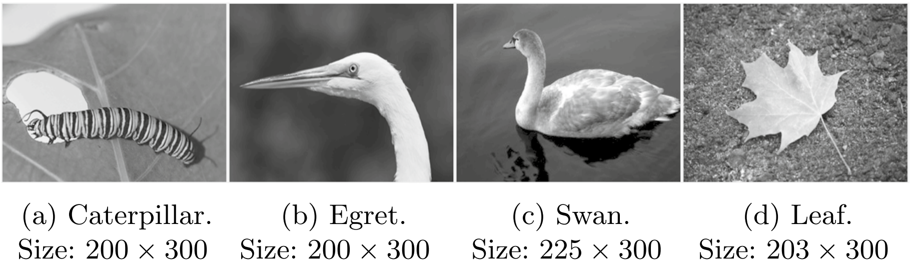
(h) TV^p MS DICE: 0.971 9

(i) Convex Potts DICE: 0.957 3

(j) SaT-Potts DICE: 0.964 3

Fig. 4 Segmentation results of Fig. 2b corrupted with motion blur followed by 45% SP noise. Regions boxed in red are only identified by the AITV SLaT, AITV CV, and TV^p MS

Figure 4 illustrates the visual results under the SP noise with motion blur case. AITV SLaT, AITV CV, and TV^p MS are able to partially segment the leftmost rectangle in the upper left corner and the small circular region right of the triangular region. These regions are boxed in red to showcase the main differences in the results outputted by the segmentation methods. By taking account for blur, AITV SLaT (ADMM) has the highest DICE index for this case while having a significantly faster time than its DCA counterpart.

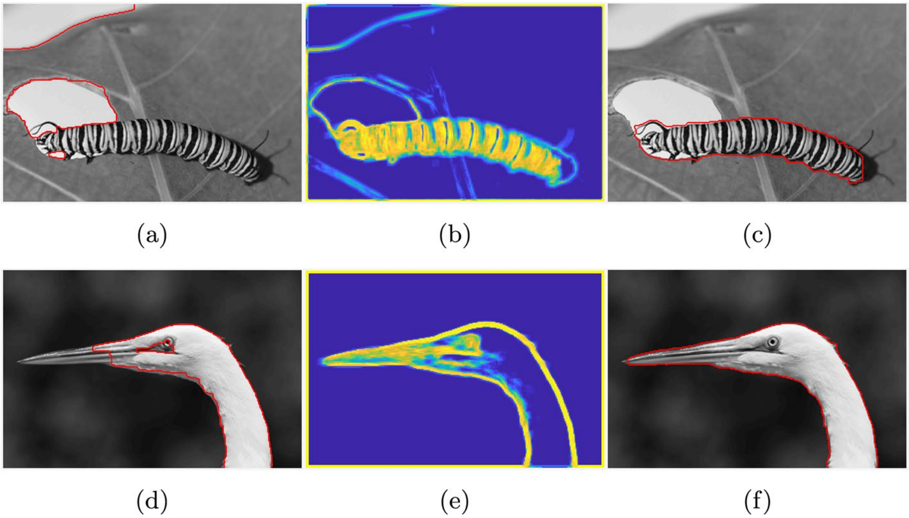

(a) Caterpillar. Size: 200 × 300

(b) Egret. Size: 200 × 300

(c) Swan. Size: 225 × 300

(d) Leaf. Size: 203 × 300

Fig. 5 Real, grayscale images for image segmentation


(a)

(b)

(c)

(d)

(e)

(f)

Fig. 6 AITV SaT results on real grayscale images with and without IIH images. Left column: AITV SaT results without IIH images. Middle column: IIH images. Right column: AITV SaT results with IIH images

4.2 Real Grayscale Images with Intensity Inhomogeneities

We examine real images with intensity inhomogeneities [1], as shown in Fig. 5. Intensity inhomogeneities can be problematic for image segmentation because of the dramatically varying pixel intensities in the local regions of an image. For example, we apply AITV SaT (ADMM) to Figs. 5a, b to exemplify the challenges of segmenting the object of interest. In Fig. 6a, no part of the caterpillar is segmented while in Fig. 6d, most of the egret's beak is not segmented. However, by incorporating the intensity inhomogeneity (IIH) images [54] shown in Figs. 6b, e as additional channels, AITV SaT accounts for intensity inhomogeneity and is able to segment the caterpillar in Fig. 6c and the egret and its beak in Fig. 6f.

Following the work of [54], we incorporate an IIH image by appending it as an additional channel to the original image to facilitate segmentation. To generate the IIH image, one calculates an IIH-indicator D

Table 3 Comparison of the DICE indices and computational times (seconds) between the segmentation methods applied to Fig. 5

	Fig. 5a		Fig. 5b		Fig. 5c		Fig. 5d	
	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)	DICE	Time (s)
(Original) SaT	0.881 8	3.10	0.967 7	3.42	0.919 1	5.09	0.928 8	4.01
TV^p SaT	0.889 9	1.97	0.847 5	1.89	**0.922 7**	2.35	**0.936 8**	2.06
AITV SaT (ADMM)	0.888 8	1.76	**0.968 6**	1.45	0.917 3	**2.20**	0.926 9	1.53
AITV SaT (DCA)	0.879 5	19.52	0.843 5	12.28	0.905 3	25.05	0.926 9	18.21
AITV CV	0.756 8	50.86	0.942 3	46.57	0.891 3	107.67	0.914 1	15.89
ICTM	0.623 0	**0.25**	0.951 6	**1.39**	0.868 8	2.90	0.912 9	**0.16**
TV^p MS	0.678 2	8.33	0.934 6	8.50	0.784 6	10.34	0.917 9	6.24
Convex Potts	**0.890 2**	2.23	0.525 7	2.27	0.813 1	4.69	0.917 3	1.96
SaT-Potts	0.876 9	2.10	0.961 3	2.08	0.916 5	2.53	0.912 0	2.37

Number in bold indicates either the highest DICE index or the fastest time among the segmentation methods for a given image

$$D = \frac{1}{|\Omega|} \sum_{(i,j)\in\Omega} \left(\frac{1}{|\Omega_{(i,j)}|} \sum_{(i',j')\in\Omega_{(i,j)}} |u_{i',j'} - \bar{u}_{i,j}|^2 \right),$$

where $\Omega_{(i,j)}$ is a neighborhood centered at pixel (i, j) and $\bar{u}_{i,j}$ is the average pixel intensity in the neighborhood $\Omega_{(i,j)}$. Using the IIH-indicator D, the IIH-image is calculated by

$$u^{\mathrm{IIH}}_{i,j} = \frac{1}{|\Omega_{(i,j)}|} \sum_{(i',j')\in\Omega_{(i,j)}} \mathbb{1}_{\Omega_{(i,j)}}(i',j'),$$

where

$$\mathbb{1}_{\Omega_{(i,j)}}(i',j') = \begin{cases} 1 & \text{if } |\bar{u}_{i,j} - u_{i',j'}|^2 \geqslant D, \\ 0 & \text{if } |\bar{u}_{i,j} - u_{i',j'}|^2 < D. \end{cases}$$

For our experiments, $\Omega_{(i,j)}$ is a 7×7 patch centered at pixel (i, j).

When the IIH image is added as a channel to the grayscale image, we smooth each channel and then apply K-means clustering for the SaT methods. For the other segmentation methods, we consider their multichannel extensions to process the two channels that are composed of grayscale and IIH.

For the images in Fig. 5 (after rescaling the pixel intensities to [0, 1]), we corrupt them with motion blur `fspecial` (‘`motion`’, `5`, `0`) followed by Gaussian noise with mean 0 and variance 0.001. We tune the parameters $\lambda \in [0.25, 5]$ and $\mu \in [5, 60]$ for the SaT methods. For each image, the ground truth is determined from the segmentation results by three human subjects. A pixel is declared an object of interest in the ground truth if at least two subjects agree [1]. The DICE indices and computational times of the segmentation algorithms are recorded in Table 3 while the segmentation results and their ground truths are presented in Figs. 7, 8, 9, and 10. Note that some segmentation results have the image border segmented because of the boundary artifacts created by the IIH image (see Figs. 6b, e). For all four images, AITV SaT (ADMM) is among the top three methods with the highest DICE indices. It provides satisfactory results in

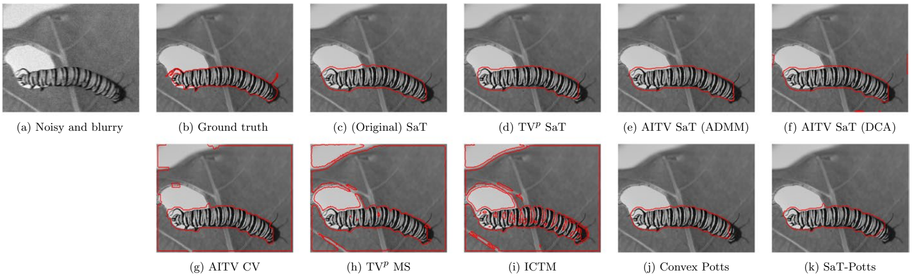


Fig. 7 Segmentation results of Fig. 5a corrupted by motion blur followed by Gaussian noise

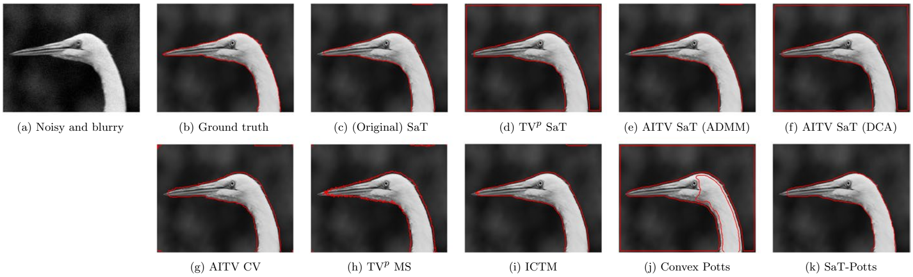


Fig. 8 Segmentation results of Fig. 5b corrupted by motion blur followed by Gaussian noise

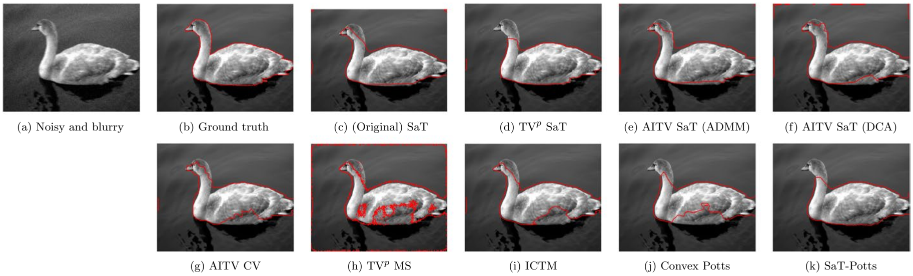


Fig. 9 Segmentation results of Fig. 5c corrupted by motion blur followed by Gaussian noise

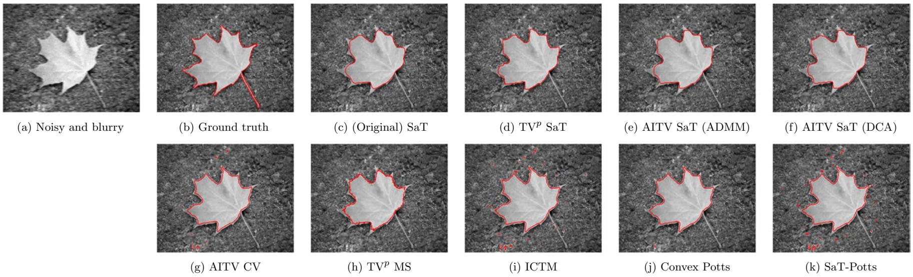
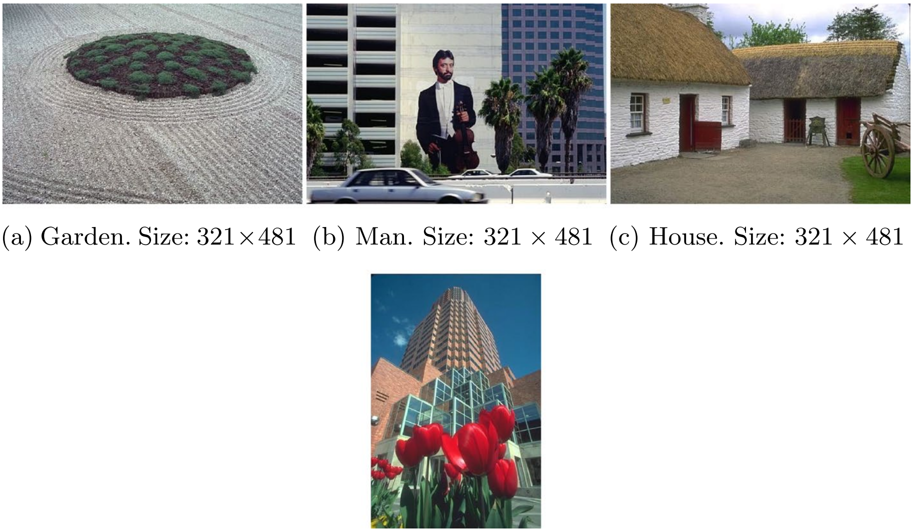


Fig. 10 Segmentation results of Fig. 5d corrupted by motion blur followed by Gaussian noise



(a) Garden. Size: 321×481 (b) Man. Size: 321×481 (c) House. Size: 321×481

(d) Building. Size: 481×321

Fig. 11 Real color images for image segmentation

about 2 s. Moreover, it outperforms its DCA counterpart in terms of DICE indices and computational times, especially for Fig. 5b. For Figs. 7, 8, and 9, although both algorithms solve the same model (7), they output different results. As (7) is nonconvex, it is possible that ADMM and DCA attain different solutions.

4.3 Real Color Images

Four real color images taken from [61] are presented in Fig. 11 for segmentation. The images are corrupted with either Gaussian noise of mean zero and variance 0.025 or 10% SP noise. We segment Fig. 11a with $k = 3$ regions, Fig. 11b with $k = 5$ regions, Fig. 11c with $k = 6$ regions, and Fig. 11d with $k = 8$ regions. Because ground truth is unavailable, we use PSNR to evaluate the segmentation result as a piecewise-constant approximation of the original image. For the SLaT methods, we tune the parameters $\lambda \in [2, 30]$ and $\mu \in [0.05, 1.0]$ for all the images.

Table 4 records the PSNR values and computational times in seconds of the segmentation algorithms while Figs. 12, 13, 14, and 15 present the visual results. Overall, AITV SLaT (ADMM) is generally among the top three methods with the best PSNR values for both noise cases. In fact, for SP noise, AITV SLaT (ADMM) has the second best PSNRs while being significantly faster than TV^p MS that has the best PSNRs.

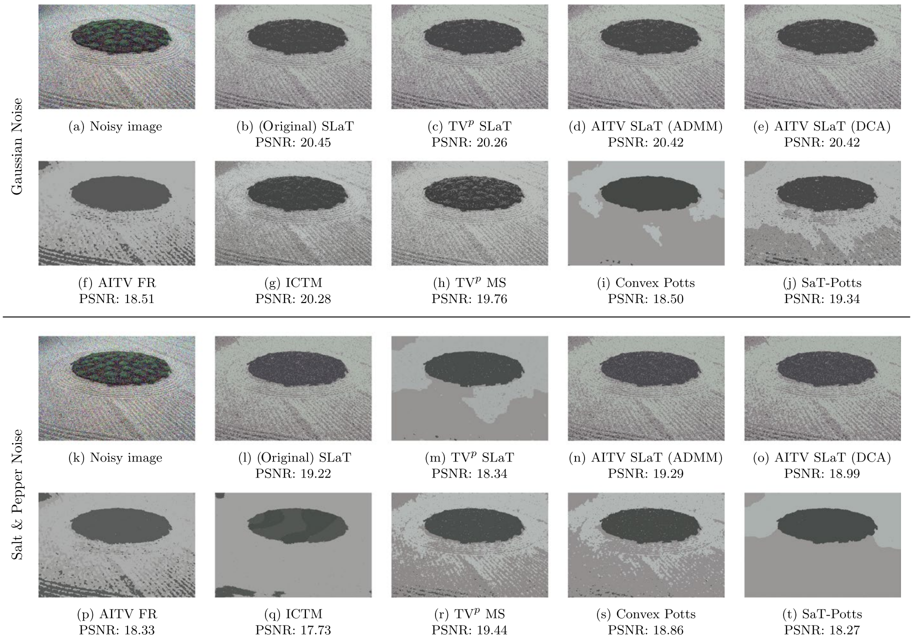
In Fig. 12, the sand lines are segmented in fine details by the SLaT methods, ICTM, and TV^p MS in the Gaussian noise case and by the (original) SLaT, AITV SLaT, and TV^p MS in the SP noise case. In Fig. 13, TV^p SLaT, AITV SLAT, AITV FR, TV^p MS, and SaT-Potts can clearly segment the multiple rows of windows on the top part of the building on the right. Under the SP noise, no algorithms succeed in the windows, but AITV SLaT (ADMM) and TV^p MS are able to preserve some parts of the man's eyes and the palm

Table 4 Comparison of the PSNRs and computational times (seconds) between the segmentation methods applied to the images in Fig. 11 corrupted with either Gaussian noise with mean zero and variance 0.025 or 10% SP noise

	Garden (Fig. 11a) $k = 3$		Man (Fig. 11b) $k = 5$		House (Fig. 11c) $k = 6$		Building (Fig. 11d) $k = 8$	
	PSNR	Time (s)	PSNR	Time (s)	PSNR	Time (s)	PSNR	Time (s)
(a) Gaussian noise								
(Original) SLaT	**20.45**	7.09	21.11	13.68	21.88	13.56	21.76	17.60
TV^p SLaT	20.26	11.58	22.11	13.59	21.93	9.83	21.63	12.16
AITV SLaT (ADMM)	20.42	7.52	22.19	15.51	21.85	10.99	**21.78**	11.90
AITV SLaT (DCA)	20.42	38.27	**22.21**	53.51	21.81	47.54	**21.78**	52.73
AITV FR	18.51	148.62	21.47	354.32	20.35	447.50	19.91	583.13
ICTM	20.28	**6.36**	20.29	11.05	19.89	**5.22**	18.63	12.41
TV^p MS	19.76	88.64	22.10	52.55	**22.10**	169.34	21.35	53.62
Convex Potts	18.50	14.93	21.30	52.22	20.64	76.07	21.17	152.46
SaT-Potts	19.34	9.78	21.71	**7.60**	21.62	7.31	21.56	**7.85**
(b) SP noise								
(Original) SLaT	19.22	**6.53**	19.83	16.26	20.95	13.71	20.08	18.79
TV^p SLaT	18.34	9.06	19.56	9.57	20.43	9.19	19.90	11.77
AITV SLaT (ADMM)	19.29	9.30	20.13	12.57	21.08	9.59	20.53	16.10
AITV SLaT (DCA)	18.99	47.62	20.09	76.38	20.91	84.83	19.97	96.29
AITV FR	18.33	144.43	19.96	307.03	19.25	368.40	19.15	759.58
ICTM	17.73	27.95	18.44	24.06	17.95	9.34	16.57	20.40
TV^p MS	**19.44**	97.90	**20.25**	44.43	**21.27**	113.61	**20.60**	101.82
Convex Potts	18.86	14.95	19.26	45.86	19.62	76.83	19.18	154.43
SaT-Potts	18.27	8.58	18.94	**6.48**	20.20	**9.77**	19.45	**7.97**

Number in bold indicates either the highest PSNR or the fastest time among the segmentation methods for a given image

trees' green color and foliage. Despite AITV SLaT (ADMM) having a lower PSNR, the palm trees are greener in the segmentation result of AITV SLaT (ADMM) than TV^p MS. In Fig. 14, under Gaussian noise, despite having lower PSNRs, both ADMM and DCA of AITV SLaT are able to more clearly segment the bottom half of the wheel at the lower right corner than TV^p SLaT and TV^p MS. Moreover, the roofs in the segmentation results of AITV SLaT are mostly brown while they have a considerable amount of green in the results of the TV^p models. Under the SP noise, AITV SLaT (ADMM) provides the most visually appealing segmentation result even though its PSNR is not the best. TV^p MS identifies the green color of the grass and most of the wheels on the bottom right corner compared to any other methods. Lastly, for Fig. 15, under Gaussian noise, the SLaT methods, the Potts methods, and TV^p MS produce visually similar segmentation results. Under SP noise, AITV SLaT (ADMM) and TV^p MS segment more windows at the top of the building than any other methods. For all four figures, AITV SLaT (ADMM) and AITV SLaT (DCA) produce segmentation results with similar PSNR values, but the former is up to five times faster than the latter. Although ICTM and SaT-Potts are the fastest methods, their segmentation results are less satisfactory.



(a) Noisy image

(b) (Original) SLaT PSNR: 20.45

(c) TV^p SLaT PSNR: 20.26

(d) AITV SLaT (ADMM) PSNR: 20.42

(e) AITV SLaT (DCA) PSNR: 20.42

(f) AITV FR PSNR: 18.51

(g) ICTM PSNR: 20.28

(h) TV^p MS PSNR: 19.76

(i) Convex Potts PSNR: 18.50

(j) SaT-Potts PSNR: 19.34

(k) Noisy image

(l) (Original) SLaT PSNR: 19.22

(m) TV^p SLaT PSNR: 18.34

(n) AITV SLaT (ADMM) PSNR: 19.29

(o) AITV SLaT (DCA) PSNR: 18.99

(p) AITV FR PSNR: 18.33

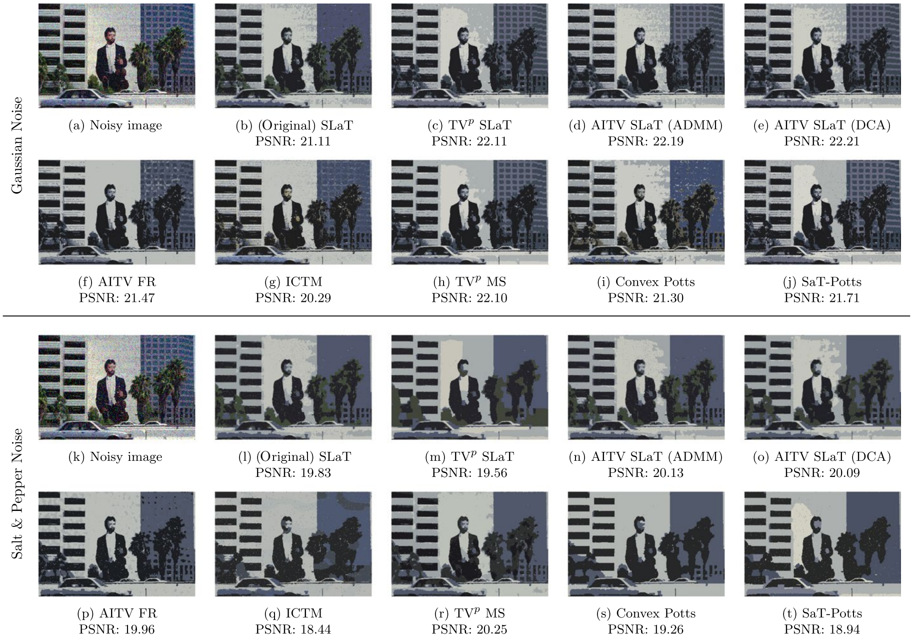
(q) ICTM PSNR: 17.73

(r) TV^p MS PSNR: 19.44

(s) Convex Potts PSNR: 18.86

(t) SaT-Potts PSNR: 18.27

Fig. 12 Segmentation results into $k = 3$ regions of Fig. 11a corrupted by either Gaussian noise of mean zero and variance 0.025 or 10% SP noise



(a) Noisy image

(b) (Original) SLaT PSNR: 21.11

(c) TV^p SLaT PSNR: 22.11

(d) AITV SLaT (ADMM) PSNR: 22.19

(e) AITV SLaT (DCA) PSNR: 22.21

(f) AITV FR PSNR: 21.47

(g) ICTM PSNR: 20.29

(h) TV^p MS PSNR: 22.10

(i) Convex Potts PSNR: 21.30

(j) SaT-Potts PSNR: 21.71

(k) Noisy image

(l) (Original) SLaT PSNR: 19.83

(m) TV^p SLaT PSNR: 19.56

(n) AITV SLaT (ADMM) PSNR: 20.13

(o) AITV SLaT (DCA) PSNR: 20.09

(p) AITV FR PSNR: 19.96

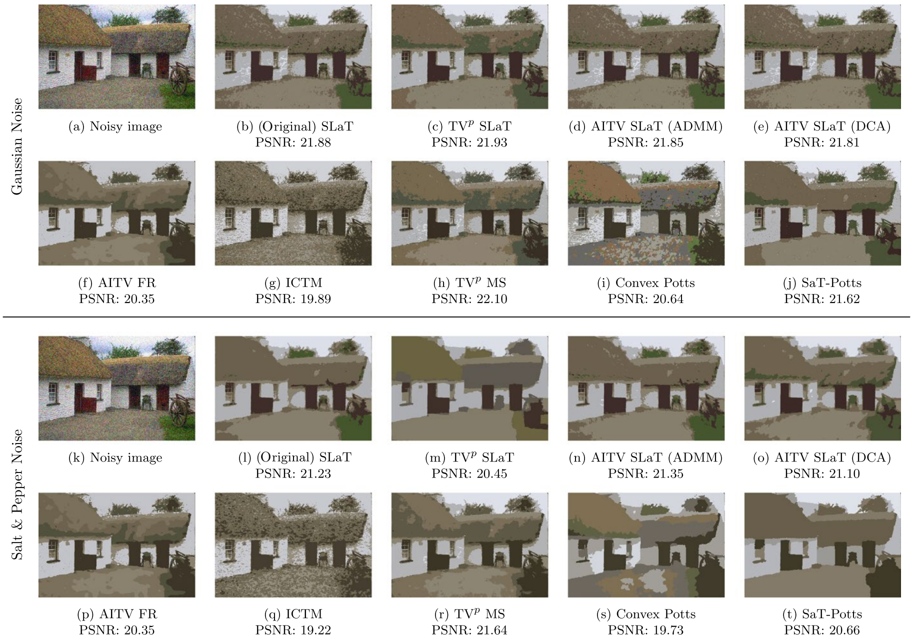
(q) ICTM PSNR: 18.44

(r) TV^p MS PSNR: 20.25

(s) Convex Potts PSNR: 19.26

(t) SaT-Potts PSNR: 18.94

Fig. 13 Segmentation results into $k = 5$ regions of Fig. 11b corrupted by either Gaussian noise of mean zero and variance 0.025 or 10% SP noise



(a) Noisy image

(b) (Original) SLaT PSNR: 21.88

(c) TV^p SLaT PSNR: 21.93

(d) AITV SLaT (ADMM) PSNR: 21.85

(e) AITV SLaT (DCA) PSNR: 21.81

(f) AITV FR PSNR: 20.35

(g) ICTM PSNR: 19.89

(h) TV^p MS PSNR: 22.10

(i) Convex Potts PSNR: 20.64

(j) SaT-Potts PSNR: 21.62

(k) Noisy image

(l) (Original) SLaT PSNR: 21.23

(m) TV^p SLaT PSNR: 20.45

(n) AITV SLaT (ADMM) PSNR: 21.35

(o) AITV SLaT (DCA) PSNR: 21.10

(p) AITV FR PSNR: 20.35

(q) ICTM PSNR: 19.22

(r) TV^p MS PSNR: 21.64

(s) Convex Potts PSNR: 19.73

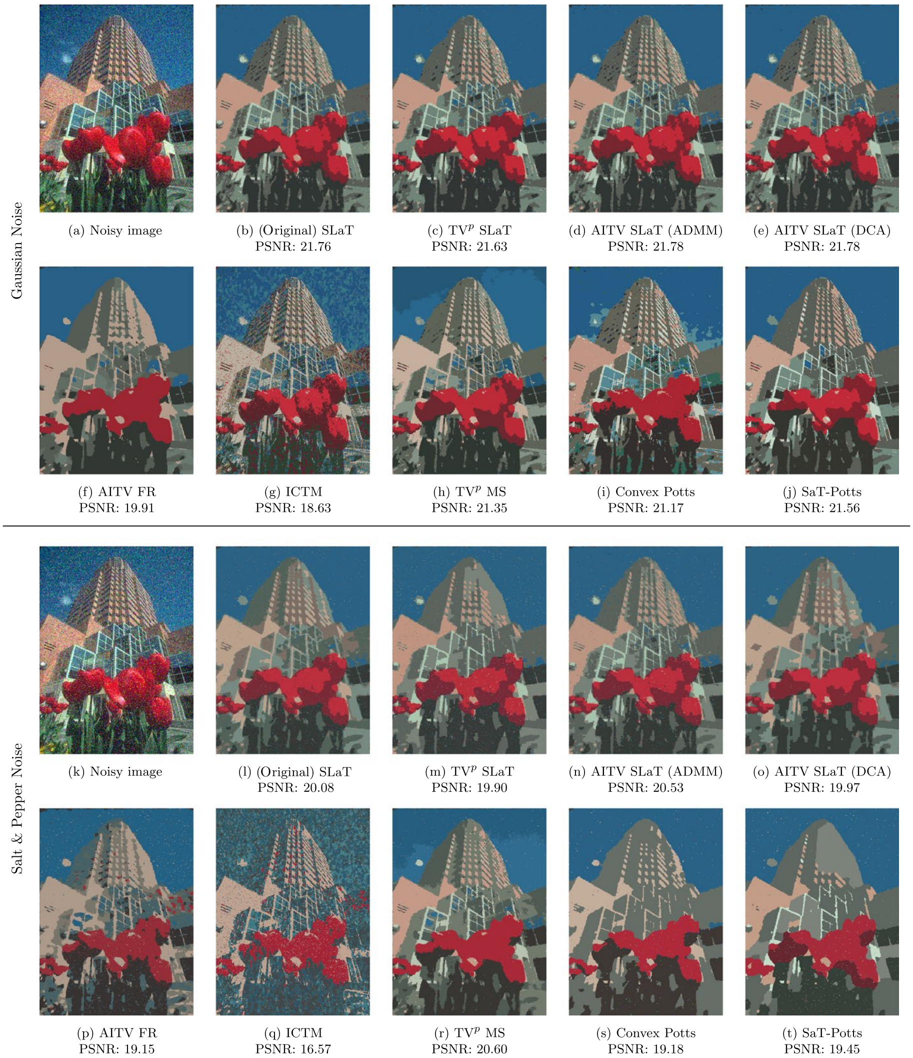
(t) SaT-Potts PSNR: 20.66

Fig. 14 Segmentation results into $k = 6$ regions of Fig. 11c corrupted by either Gaussian noise of mean zero and variance 0.025 or 10% SP noise

4.3.1 HSV vs. Lab

The HSV (hue, saturation, and value) space is another popular, approximately uniform color space that could be used instead of Lab space for the SLaT methods. It was used to derived features for improving image segmentation algorithms [6, 10, 25, 44, 66, 75]. Some works [66, 76] claim that HSV space is better than Lab space for image segmentation. However, we provide numerical evidence to show that HSV space may not be as effective as Lab space for the SLaT methods.

To compare the segmentation results between HSV and Lab for the AITV SLaT method, we replace Lab with HSV in Algorithm 1 and apply the HSV-based algorithm to the images in Fig. 11 corrupted with Gaussian noise with mean zero and variance 0.025. Figure 16 compares the segmentation results and the PSNR values between using HSV and Lab spaces. Overall, we observe that using Lab space for AITV SLaT leads to higher PSNR values and more detailed segmentation. For Fig. 16a, using Lab space identifies more of the fine sand lines than HSV space. Unlike using HSV space, AITV SLaT with Lab is able to identify the windows on the right side of Fig. 16b and the wheel on the bottom right corner of Fig. 16c. Lastly, for Fig. 16d, the tulips are clearly redder and more segmented in the result of AITV SLaT with Lab than with HSV.



(a) Noisy image

(b) (Original) SLaT PSNR: 21.76

(c) TV^p SLaT PSNR: 21.63

(d) AITV SLaT (ADMM) PSNR: 21.78

(e) AITV SLaT (DCA) PSNR: 21.78

(f) AITV FR PSNR: 19.91

(g) ICTM PSNR: 18.63

(h) TV^p MS PSNR: 21.35

(i) Convex Potts PSNR: 21.17

(j) SaT-Potts PSNR: 21.56

(k) Noisy image

(l) (Original) SLaT PSNR: 20.08

(m) TV^p SLaT PSNR: 19.90

(n) AITV SLaT (ADMM) PSNR: 20.53

(o) AITV SLaT (DCA) PSNR: 19.97

(p) AITV FR PSNR: 19.15

(q) ICTM PSNR: 16.57

(r) TV^p MS PSNR: 20.60

(s) Convex Potts PSNR: 19.18

(t) SaT-Potts PSNR: 19.45

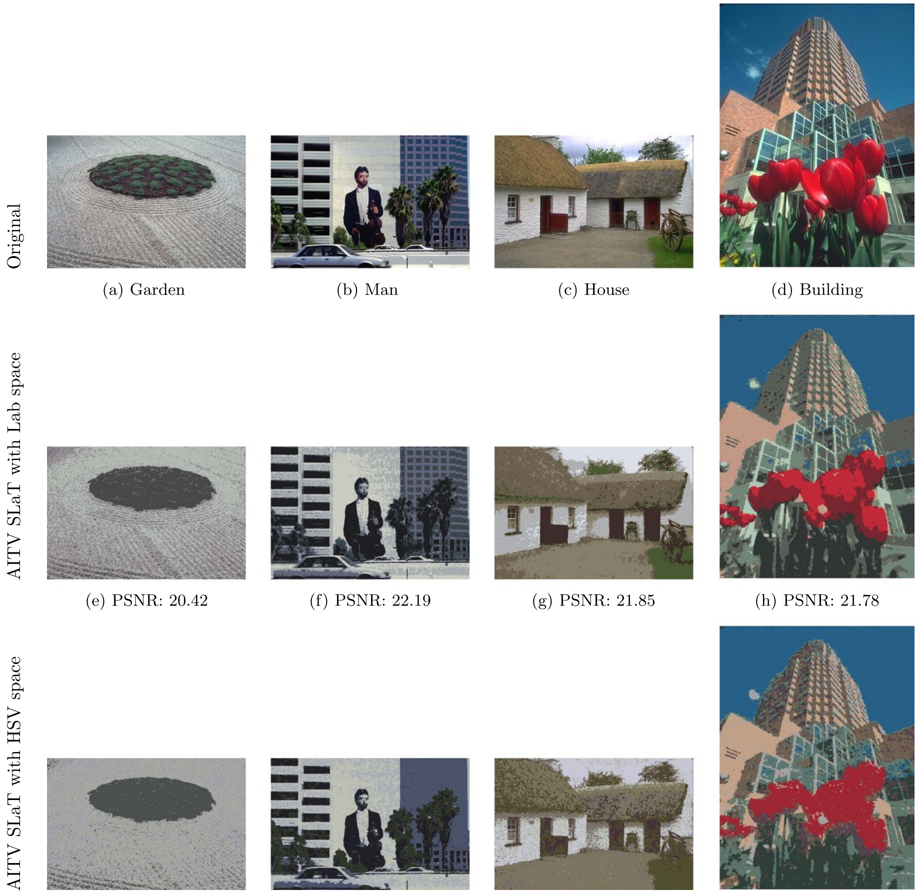
Fig. 15 Segmentation results into $k = 8$ regions of Fig. 11d corrupted by either Gaussian noise of mean zero and variance 0.025 or 10% SP noise

4.4 Parameter Analysis

4.4.1 Model Parameters of (7)

We analyze the following parameters in (7).

- λ: this fidelity parameter weighs how close the approximation Au^* is to the original image f, where u^* is a solution to (7). When the image f has a large amount of noise, choosing a small value for λ is recommended.



(a) Garden (b) Man (c) House (d) Building

(e) PSNR: 20.42 (f) PSNR: 22.19 (g) PSNR: 21.85 (h) PSNR: 21.78

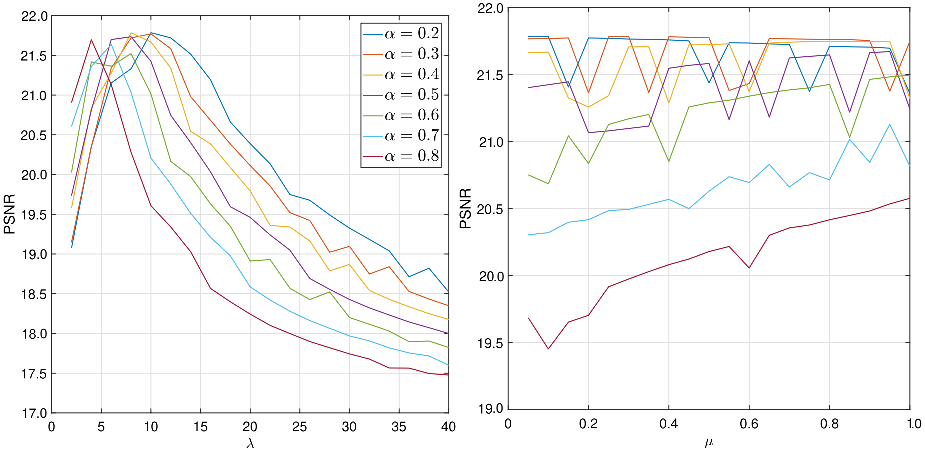
(i) PSNR: 18.25 (j) PSNR: 19.93 (k) PSNR: 19.62 (l) PSNR: 20.08

Fig. 16 Comparison between using Lab space vs. HSV space for the AITV SLaT method

- μ: this smoothing parameter determines the smoothness of the solution u^* of (7), which may help with denoising. However, choosing a large value for μ will deteriorate important edge information in u^*.
- $\alpha \in [0, 1]$: this sparsity parameter determines the gradient vector sparsity at each pixel, which is important in preserving edge information. However, choosing a large α may result in preserving some noise in the solution u^*.

To perform sensitivity analysis on the model parameters, we apply AITV SLaT (ADMM) with parameters $\delta_0 = 2$ and $\sigma = 1.25$ to Fig. 11d corrupted with Gaussian noise with mean 0 and variance 0.025. We examine the sparsity parameter $\alpha \in \{0.2, 0.3, \cdots, 0.8\}$ while we vary either the fidelity parameter λ with $\mu = 0.10$ fixed or the smoothing parameter μ with $\lambda = 10$ fixed. The sensitivity analysis is visualized in Fig. 17.

Figure 17a shows that the PSNR has a concave relationship with respect to the fidelity parameter λ for each value of α. We observe that larger value of α leads to higher



(a) Changes in PSNR with respect to λ with $\mu = 0.10$ fixed

(b) Changes in PSNR with respect to μ with $\lambda = 10$ fixed

Fig. 17 Sensitivity analysis on the model parameters λ and μ to Fig. 11d corrupted by Gaussian noise with mean 0 and variance 0.025

PSNR for smaller value of λ. More specifically, when $\lambda \leqslant 5.0$, the order of the PSNR curves follows the increasing value of α. However, the order is reversed when λ becomes large enough, such as when $\lambda \geqslant 10$. Figure 17b shows that with respect to the smoothing parameter, PSNR is generally increasing when $0.4 \leqslant \alpha \leqslant 0.8$ while it appears to be robust for $\alpha = 0.2, 0.3$.

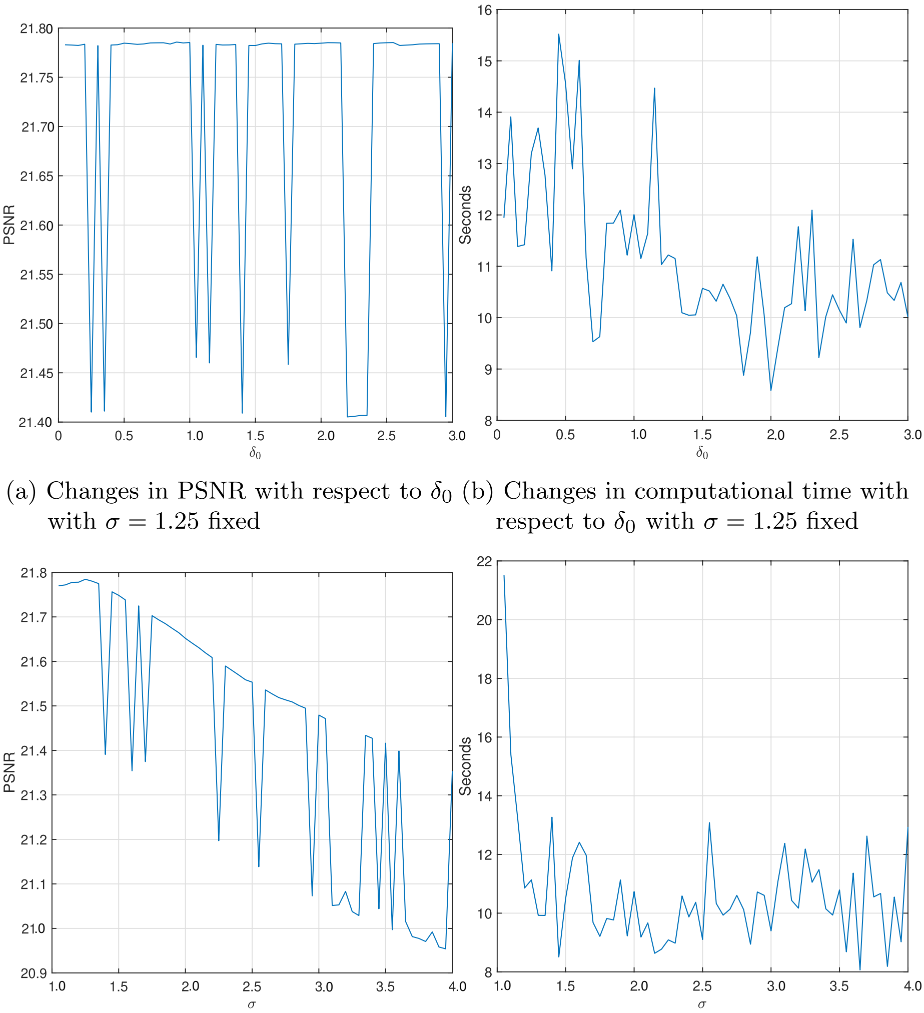
4.4.2 Algorithm Parameters of Algorithm 2

We analyze the following parameters introduced in the ADMM algorithm that solves (7).

- δ_0: this penalty parameter weighs the quadratic difference between the original variable ∇u and the auxiliary variable w.
- σ: this penalty multiplier determines the numerical convergence speed of the ADMM algorithm.

We perform sensitivity analysis on AITV SLaT (ADMM) with model parameters $\lambda = 10, \mu = 0.1$, and $\alpha = 0.2$ to Fig. 11a corrupted with Gaussian noise with mean 0 and variance 0.025. When varying δ_0, we fix $\sigma = 1.25$ while when varying σ, we fix $\delta_0 = 2$. Figure 18 visualizes the sensitivity analysis of the algorithm parameters.

According to Figs. 18a, b, the penalty parameter δ_0 does not have much influence on the PSNR, but it does affect the speed of the ADMM algorithm. When $\delta_0 < 1.5$, the computational time is between 9.5 s and 15.5 s, but when $\delta_0 \geqslant 1.5$, it decreases to between about 9 s and 12 s. As shown in Figs. 18c, d, the penalty multiplier σ does have an impact on both the PSNR and the algorithm's numerical convergence. As σ increases, the PSNR generally decreases. When $\sigma < 1.25$, the algorithm can be as slow as up to 22 s, but when $\sigma \geqslant 1.25$, it does speed up to between 8 s and 13 s.



(a) Changes in PSNR with respect to δ_0 with $\sigma = 1.25$ fixed

(b) Changes in computational time with respect to δ_0 with $\sigma = 1.25$ fixed

(c) Changes in PSNR with respect to σ with $\delta_0 = 2$ fixed

(d) Changes in computational time with respect to σ with $\delta_0 = 2$ fixed

Fig. 18 Sensitivity analysis on the ADMM algorithm parameters δ_0 and σ

5 Conclusion

In this paper, we proposed an efficient ADMM algorithm for the SaT/SLaT framework that utilizes AITV regularization. When designing the ADMM algorithm, we incorporated the proximal operator for the $\ell_1 - \alpha\ell_2$ regularization [57]. We provided convergence analysis of ADMM to demonstrate that the algorithm subsequentially converges to a KKT point under certain conditions. In our numerical experiments, the AITV SaT/SLaT using our ADMM algorithm produces high-quality segmentation results within a few seconds. In addition, this work shows the effectiveness of using nonconvex regularizations in image processing. As for future work, we will explore other nonconvex regularizations, such as transformed ℓ_1 [91, 92], as alternative options to AITV and TV^p $(0 < p < 1)$ under the SaT/SLaT framework. To simplify the SLaT framework for color images, we plan to apply these nonconvex regularizations in quaternion space to complement the quaternion-based SaT model [84] with ℓ_1/ℓ_2 regularization [72, 78–80].

Appendix A Proofs of Sect. 3.3

Appendix A.1 Proof of Lemma 3

Proof It is straightforward that $u^\top A^\top Au = \|Au\|_2^2 \geqslant 0$ and $u^\top \nabla^\top \nabla u = \|\nabla u\|_2^2 \geqslant 0$ for any $u \in X$, so $\zeta \geqslant 0$. If $\zeta = 0$, then there exists a nonzero vector $x \in X$ such that $\lambda\|Ax\|_2^2 + (\mu + \delta_0)\|\nabla x\|_2^2 = \lambda x^\top A^\top Ax + (\mu + \delta_0)x^\top \nabla^\top \nabla x = 0$. Then, we shall have $x \in \ker(A) \cap \ker(\nabla)$, contradicting that $\ker(A) \cap \ker(\nabla) = \{0\}$. Therefore, $\zeta > 0$ and hence we get

$$\lambda\|Au\|_2^2 + (\mu + \delta_0)\|\nabla u\|_2^2 \geqslant \zeta\|u\|_2^2, \quad \forall u \in X.$$

As $\delta_{t+1} \geqslant \delta_t$ $(\sigma \geqslant 1)$, $\mathcal{L}_{\delta_t}(u, w_t, z_t)$ is a strongly convex function of u with parameter $\zeta > 0$. Fixing w_t, z_t, the minimizer u_{t+1} of $\mathcal{L}_{\delta_t}(u, w_t, z_t)$ in (10a) satisfies the following inequality [5, Theorem 5.25]:

$$\mathcal{L}_{\delta_t}(u_{t+1}, w_t, z_t) - \mathcal{L}_{\delta_t}(u_t, w_t, z_t) \leqslant -\frac{\zeta}{2}\|u_{t+1} - u_t\|_2^2. \tag{A1}$$

As w_{t+1} is the optimal solution to (10b), we have

$$\mathcal{L}_{\delta_t}(u_{t+1}, w_{t+1}, z_t) - \mathcal{L}_{\delta_t}(u_{t+1}, w_t, z_t) \leqslant 0. \tag{A2}$$

It follows from the update (10c) that

$$\begin{aligned}\mathcal{L}_{\delta_t}(u_{t+1}, w_{t+1}, z_{t+1}) - \mathcal{L}_{\delta_t}(u_{t+1}, w_{t+1}, z_t) &= \langle z_{t+1} - z_t, \nabla u_{t+1} - w_{t+1}\rangle \\ &= \frac{1}{\delta_t}\|z_{t+1} - z_t\|_2^2.\end{aligned} \tag{A3}$$

Similarly, we get

$$\begin{aligned}\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) - \mathcal{L}_{\delta_t}(u_{t+1}, w_{t+1}, z_{t+1}) &= \frac{\delta_{t+1} - \delta_t}{2}\|\nabla u_{t+1} - w_{t+1}\|_2^2 \\ &= \frac{\delta_{t+1} - \delta_t}{2\delta_t^2}\|z_{t+1} - z_t\|_2^2.\end{aligned} \tag{A4}$$

Combining (A1)–(A4) leads to the desired inequality

$$\begin{aligned}\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) - \mathcal{L}_{\delta_t}(u_t, w_t, z_t) \leqslant{}& \frac{\delta_{t+1}-\delta_t}{2\delta_t^2}\|z_{t+1}-z_t\|_2^2 + \frac{1}{\delta_t}\|z_{t+1}-z_t\|_2^2\\ &- \frac{\zeta}{2}\|u_{t+1}-u_t\|_2^2\\ ={}& \frac{\sigma+1}{2\sigma^t\delta_0}\|z_{t+1}-z_t\|_2^2 - \frac{\zeta}{2}\|u_{t+1}-u_t\|_2^2.\end{aligned}$$

Appendix A.2 Proof of Proposition 1

Proof (a) We start by proving the boundedness of $\{z_t\}_{t=1}^{\infty}$. The optimality condition of (10b) at iteration t is expressed by

$$0 \in \partial\big(\|w_{t+1}\|_1 - \alpha\|w_{t+1}\|_{2,1}\big) - \delta_t\big(\nabla u_{t+1} - w_{t+1}\big) - z_t. \tag{A5}$$

Together with (10c), we have

$$z_{t+1} \in \partial\big(\|w_{t+1}\|_1 - \alpha\|w_{t+1}\|_{2,1}\big) \subset \partial\|w_{t+1}\|_1 - \alpha\partial\|w_{t+1}\|_{2,1}, \tag{A6}$$

which implies that there exist two vectors $v_1 \in \partial\|w_{t+1}\|_1$ and $v_2 \in \partial\|w_{t+1}\|_{2,1}$ such that $z_{t+1} = v_1 - \alpha v_2$. For any $v \in \partial\|w\|_1$, we have

$$(v_x)_{i,j} = \operatorname{sign}((w_x)_{i,j}) \text{ and } (v_y)_{i,j} = \operatorname{sign}((w_y)_{i,j}), \tag{A7}$$

which guarantees that $\|v\|_\infty \leqslant 1$. If $z \in \partial\|w\|_{2,1}$, then

$$z_{i,j} = \begin{cases} \frac{w_{i,j}}{\|w_{i,j}\|_2} & \text{if } \|w_{i,j}\|_2 \neq 0,\\ \in \{z_{i,j} \in \mathbb{R}^2 : \|z_{i,j}\|_2 \leqslant 1\} & \text{if } \|w_{i,j}\|_2 = 0.\end{cases} \tag{A8}$$

By (A8), we have $\|(v_2)_{i,j}\|_2 \leqslant 1$, which means that $\|v_2\|_\infty \leqslant 1$. As a result, $\|z_{t+1}\|_\infty \leqslant \|v_1\|_\infty + \alpha\|v_2\|_\infty \leqslant 2$. Altogether, we arrive at an upper bound, i.e.,

$$\begin{aligned}\|z_{t+1}\|_2 &= \sqrt{\sum_{i,j}\big(|(z_{t+1,x})_{i,j}|^2 + |(z_{t+1,y})_{i,j}|^2\big)}\\ &\leqslant \sqrt{2^2(2MN)} = 2\sqrt{2MN}.\end{aligned} \tag{A9}$$

By telescoping summation of (13), we have for all t that

$$\begin{aligned}\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) &\leqslant \mathcal{L}_{\delta_0}(u_0, w_0, z_0) + \frac{\sigma+1}{2\delta_0}\sum_{i=0}^{t}\frac{1}{\sigma^i}\|z_{i+1}-z_i\|_2^2\\ &\leqslant \mathcal{L}_{\delta_0}(u_0, w_0, z_0) + \frac{\sigma+1}{2\delta_0}\sum_{i=0}^{\infty}\frac{1}{\sigma^i}\|z_{i+1}-z_i\|_2^2.\end{aligned}$$

Now that $\{z_t\}_{t=1}^{\infty}$ is bounded, then $\{\|z_{t+1}-z_t\|_2^2\}_{t=1}^{\infty}$ is bounded. Denote $C := \sup_{t\in\mathbb{N}} \|z_{t+1}-z_t\|_2^2$. If $\sigma = 1$ and $\sum_{i=0}^{\infty}\|z_{i+1}-z_i\|_2^2 < \infty$, then $\{\mathcal{L}_{\delta_t}(u_t, w_t, z_t)\}_{t=1}^{\infty}$ is uniformly bounded above. On the other hand, if $\sigma > 1$, then we get

$$\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) \leqslant \mathcal{L}_{\delta_0}(u_0, w_0, z_0) + \frac{C(\sigma+1)}{2\delta_0} \sum_{i=0}^{\infty} \frac{1}{\sigma^i} < \infty,$$

where the infinite sum converges for $\sigma > 1$. In either case, we have that $\{\mathcal{L}_{\delta_t}(u_t, w_t, z_t)\}_{t=1}^{\infty}$ is uniformly bounded above, and hence there exists a constant $\tilde{C} > 0$ such that $\mathcal{L}_{\delta_t}(u_t, w_t, z_t) < \tilde{C}$.

Since $\|w\|_{2,1} \leqslant \|w\|_1$, we have

$$\frac{\mu}{2}\|\nabla u_t\|_2^2 - \frac{1}{2\delta_t}\|z_t\|_2^2 \leqslant \mathcal{L}_{\delta_t}(u_t, w_t, z_t) \leqslant \tilde{C}.$$

This suggests an upper bound of $\|\nabla u_t\|_2$, i.e.,

$$\|\nabla u_t\|_2 \leqslant \sqrt{\frac{2}{\mu}\left(\tilde{C} + \frac{1}{2\delta_t}\|z_t\|_2^2\right)} \leqslant \sqrt{\frac{2}{\mu}\left(\tilde{C} + \frac{4MN}{\delta_0}\right)}.$$

Moreover, we observe that

$$\frac{\lambda}{2}\|f - Au_t\|_2^2 - \frac{1}{2\delta_t}\|z_t\|_2^2 \leqslant \mathcal{L}_{\delta_t}(u_t, w_t, z_t) \leqslant \tilde{C}.$$

As $\{z_t\}_{t=1}^{\infty}$ is bounded, then $\{\|f - Au_t\|_2^2\}_{t=1}^{\infty}$ is bounded as well. Altogether $\{F(u_t)\}_{t=1}^{\infty}$ is a bounded sequence, and hence we conclude that $\{u_t\}_{t=1}^{\infty}$ is bounded by coercivity in Lemma 1. Lastly, from (10c), we have

$$\|w_t\|_2 \leqslant \left\|\nabla u_t - \frac{z_t - z_{t-1}}{\delta_{t-1}}\right\|_2^2 \leqslant \frac{4\sqrt{2MN}}{\delta_0} + \sqrt{\frac{2}{\mu}\left(\tilde{C} + \frac{4MN}{\delta_0}\right)},$$

proving that $\{w_t\}_{t=1}^{\infty}$ is bounded.

(b) By Lemma 3, we can derive

$$\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) \leqslant \mathcal{L}_0(u_0, w_0, z_0) + \frac{(\sigma+1)}{2\delta_0} \sum_{i=0}^{t} \frac{1}{\sigma^i}\|z_{i+1} - z_i\|_2^2 - \frac{\zeta}{2}\sum_{i=0}^{t}\|u_{i+1} - u_i\|_2^2.$$

By (A9), we have

$$\mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) \geqslant -\frac{1}{2\delta_{t+1}}\|z_{t+1}\|_2^2 \geqslant -\frac{4MN}{\delta_0}, \ \forall t \in \mathbb{N}. \tag{A10}$$

Combining the two inequalities gives us

$$\begin{aligned}-\frac{4MN}{\delta_0} + \frac{\zeta}{2}\sum_{i=0}^{t}\|u_{i+1} - u_i\|_2^2 &\leqslant \mathcal{L}_{\delta_{t+1}}(u_{t+1}, w_{t+1}, z_{t+1}) + \frac{\zeta}{2}\sum_{i=0}^{t}\|u_{i+1} - u_i\|_2^2 \\ &\leqslant \mathcal{L}_0(u_0, w_0, z_0) + \frac{(\sigma+1)}{2\delta_0}\sum_{i=0}^{t}\frac{1}{\sigma^i}\|z_{i+1} - z_i\|_2^2.\end{aligned}$$

As $t \to \infty$, we obtain

$$0 \leqslant \frac{\zeta}{2} \sum_{i=0}^{\infty} \|u_{i+1} - u_i\|_2^2 \leqslant \mathcal{L}_{\delta_0}(u_0, w_0, z_0) + \frac{(\sigma+1)}{2\delta_0} \sum_{i=0}^{\infty} \frac{1}{\sigma^i} \|z_{i+1} - z_i\|_2^2 + \frac{4MN}{\delta_0}.$$

Earlier in proving the boundedness of $\{\mathcal{L}_{\delta_t}(u_t, w_t, z_t)\}_{t=1}^{\infty}$, we show that the summation $\sum_{i=0}^{\infty} \frac{1}{\sigma^i} \|z_{i+1} - z_i\|_2^2$ converges. As a result, the summation $\sum_{i=0}^{\infty} \|u_{i+1} - u_i\|_2^2$ converges, which implies that $u_{t+1} - u_t \to 0$.

Acknowledgements The authors thank Xu Li for providing code and answering questions about the IIH image in [54]. The authors also thank Elisha Dayag for writing the initial code for the TV^p-regularized Mumford-Shah model described in [55]. The work was partially supported by the NSF grants DMS-1854434, DMS-1952644, DMS-2151235, DMS-2219904, and CAREER 1846690. We thank the two reviewers for their valuable feedback in improving the quality of the manuscript.

Availability of Data and Materials The images in Sect. 4.1 are available from the corresponding author on reasonable request. The images in Sect. 4.2 are available at https://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/. The images in Sect. 4.3 are available at https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/.

Code Availability Code generated is available at https://github.com/kbui1993/Official_AITV_SaT_SLaT.

Declarations

Conflict of Interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Alpert, S., Galun, M., Basri, R., Brandt, A.: Image segmentation by probabilistic bottom-up aggregation and cue integration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 315–327 (2007)
2. Ambrosio, L., Tortorelli, V.M.: Approximation of functional depending on jumps by elliptic functional via Γ-convergence. Commun. Pure Appl. Math. **43**(8), 999–1036 (1990)
3. Arthur, D., Vassilvitskii, S.: k-means++: the advantages of careful seeding. In: 2007 ACM-SIAM Symposium on Discrete Algorithms (SODA'07), pp. 1027–1035 (2007)
4. Bar, L., Chan, T.F., Chung, G., Jung, M., Kiryati, N., Mohieddine, R., Sochen, N., Vese, L.A.: Mumford and Shah model and its applications to image segmentation and image restoration. In: Scherzer, O. (ed.) Handbook of Mathematical Methods in Imaging, pp. 1095–1157. Springer, New York, NY (2011). https://doi.org/10.1007/978-0-387-92920-0_25
5. Beck, A.: First-Order Methods in Optimization. SIAM, Philadelphia (2017)
6. Benninghoff, H., Garcke, H.: Efficient image segmentation and restoration using parametric curve evolution with junctions and topology changes. SIAM J. Imaging Sci. **7**(3), 1451–1483 (2014)
7. Birkholz, H.: A unifying approach to isotropic and anisotropic total variation denoising models. J. Comput. Appl. Math. **235**(8), 2502–2514 (2011)
8. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. **3**(1), 1–122 (2011)
9. Bui, K., Park, F., Lou, Y., Xin, J.: A weighted difference of anisotropic and isotropic total variation for relaxed Mumford-Shah color and multiphase image segmentation. SIAM J. Imaging Sci. **14**(3), 1078–1113 (2021)
10. Burdescu, D.D., Brezovan, M., Ganea, E., Stanescu, L.: A new method for segmentation of images represented in a HSV color space. In: Advanced Concepts for Intelligent Vision Systems: 11th International Conference, ACIVS 2009, Bordeaux, France, September 28–October 2, 2009. Proceedings 11, pp. 606–617. Springer (2009)
11. Cai, X., Chan, R., Nikolova, M., Zeng, T.: A three-stage approach for segmenting degraded color images: smoothing, lifting and thresholding (SLaT). J. Sci. Comput. **72**(3), 1313–1332 (2017)

12. Cai, X., Chan, R., Zeng, T.: A two-stage image segmentation method using a convex variant of the Mumford-Shah model and thresholding. SIAM J. Imaging Sci. **6**(1), 368–390 (2013)
13. Cao, W., Sun, J., Zongben, X.: Fast image deconvolution using closed-form thresholding formulas of $L_q(q = \frac{1}{2}, \frac{2}{3})$ regularization. J. Vis. Commun. Image Represent. **24**(1), 31–41 (2013)
14. Chambolle, A.: Finite-differences discretizations of the Mumford-Shah functional. ESAIM: Math. Model. Numer. Anal. **33**(2), 261–288 (1999)
15. Chambolle, A., Dal Maso, G.: Discrete approximation of the Mumford-Shah functional in dimension two. ESAIM: Math. Model. Numer. Anal. **33**(4), 651–672 (1999)
16. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
17. Chan, R., Yang, H., Zeng, T.: A two-stage image segmentation method for blurry images with Poisson or multiplicative gamma noise. SIAM J. Imaging Sci. **7**(1), 98–127 (2014)
18. Chan, R.H., Ng, M.K.: Conjugate gradient methods for Toeplitz systems. SIAM Rev. **38**(3), 427–482 (1996)
19. Chan, T.F., Esedoglu, S., Nikolova, M.: Algorithms for finding global minimizers of image segmentation and denoising models. SIAM J. Appl. Math. **66**(5), 1632–1648 (2006)
20. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Process. **10**(2), 266–277 (2001)
21. Chan, T.F., Yezrielev Sandberg, B., Vese, L.A.: Active contours without edges for vector-valued images. J. Vis. Commun. Image Represent. **11**(2), 130–141 (2000)
22. Chang, H., Lou, Y., Duan, Y., Marchesini, S.: Total variation-based phase retrieval for Poisson noise removal. SIAM J. Imaging Sci. **11**(1), 24–55 (2018)
23. Chang, H., Lou, Y., Ng, M.K., Zeng, T.: Phase retrieval from incomplete magnitude information via total variation regularization. SIAM J. Sci. Comput. **38**(6), A3672–A3695 (2016)
24. Chartrand, R., Yin, W.: Iteratively reweighted algorithms for compressive sensing. In: 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3869–3872. IEEE (2008)
25. Chen, T.-W., Chen, Y.-L., Chien, S.-Y.: Fast image segmentation based on K-means clustering with histograms in HSV color space. In: 2008 IEEE 10th Workshop on Multimedia Signal Processing, pp. 322–325. IEEE (2008)
26. Chen, X., Ng, M.K., Zhang, C.: Non-Lipshitz ℓ_p-regularization and box constrained model for image reconstruction. IEEE Trans. Image Process. **21**(12), 4709–4721 (2012)
27. Clarke, F.: Functional Analysis, Calculus of Variations and Optimal Control, vol. 264. Springer, Heidelberg (2013)
28. Condat, L.: Discrete total variation: new definition and minimization. SIAM J. Imaging Sci. **10**(3), 1258–1290 (2017)
29. Deng, W., Yin, W.: On the global and linear convergence of the generalized alternating direction method of multipliers. J. Sci. Comput. **66**(3), 889–916 (2016)
30. Dice, L.R.: Measures of the amount of ecologic association between species. Ecology **26**(3), 297–302 (1945)
31. Ding, L., Han, W.: $\alpha\ell_1 - \beta\ell_2$ regularization for sparse recovery. Inverse Probl. **35**(12), 125009 (2019)
32. Esedoglu, S., Tsai, Y.-H.R.: Threshold dynamics for the piecewise constant Mumford-Shah functional. J. Comput. Phys. **211**(1), 367–384 (2006)
33. Esser, E., Zhang, X., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science. SIAM J. Imaging Sci. **3**(4), 1015–1046 (2010)
34. Ge, H., Chen, W., Ng, M.K.: New restricted isometry property analysis for $\ell_1 - \ell_2$ minimization methods. SIAM J. Imaging Sci. **14**(2), 530–557 (2021)
35. Getreuer, P.: Chan-Vese segmentation. Image Process. Line **2**, 214–224 (2012)
36. Gobbino, M.: Finite difference approximation of the Mumford-Shah functional. Commun. Pure Appl. Math. **51**(2), 197–228 (1998)
37. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split Bregman method: segmentation and surface reconstruction. J. Sci. Comput. **45**(1/2/3), 272–293 (2010)
38. Goldstein, T., Osher, S.: The split Bregman method for L1-regularized problems. SIAM J. Imaging Sci. **2**(2), 323–343 (2009)
39. Gu, S., Xie, Q., Meng, D., Zuo, W., Feng, X., Zhang, L.: Weighted nuclear norm minimization and its applications to low level vision. Int. J. Comput. Vis. **121**(2), 183–208 (2017)
40. Han, D., Sun, D., Zhang, L.: Linear rate convergence of the alternating direction method of multipliers for convex composite programming. Math. Oper. Res. **43**(2), 622–637 (2018)
41. Hartigan, J.A., Wong, M.A.: A k-means clustering algorithm. J. R. Stat. Soc.: Ser. C (Appl. Stat.) **28**(1), 100–108 (1979)

42. Hintermüller, M., Tao, W.: Nonconvex TV^q-models in image restoration: analysis and a trust-region regularization-based superlinearly convergent solver. SIAM J. Imaging Sci. **6**(3), 1385–1415 (2013)
43. Hou, Z.: A review on MR image intensity inhomogeneity correction. Int. J. Biomed. Imaging **2006**, 049515 (2006)
44. Huang, Z.-K., Liu, D.-H.: Segmentation of color image using EM algorithm in HSV color space. In: 2007 International Conference on Information Acquisition, pp. 316–319. IEEE (2007)
45. Jung, M.: Piecewise-smooth image segmentation models with L^1 data-fidelity terms. J. Sci. Comput. **70**(3), 1229–1261 (2017)
46. Jung, M., Kang, M., Kang, M.: Variational image segmentation models involving non-smooth data-fidelity terms. J. Sci. Comput. **59**(2), 277–308 (2014)
47. Lanza, A., Morigi, S., Sgallari, F.: Constrained TV$_p - \ell_2$ model for image restoration. J. Sci. Comput. **68**(1), 64–91 (2016)
48. Le Thi, H.A., Pham Dinh, T.: DC programming and DCA: thirty years of developments. Math. Program. **169**(1), 5–68 (2018)
49. Li, C., Kao, C.-Y., Gore, J.C., Ding, Z.: Minimization of region-scalable fitting energy for image segmentation. IEEE Trans. Image Process. **17**(10), 1940–1949 (2008)
50. Li, F., Ng, M.K., Zeng, T.Y., Shen, C.: A multiphase image segmentation method based on fuzzy region competition. SIAM J. Imaging Sci. **3**(3), 277–299 (2010)
51. Li, F., Osher, S., Qin, J., Yan, M.: A multiphase image segmentation based on fuzzy membership functions and L1-norm fidelity. J. Sci. Comput. **69**(1), 82–106 (2016)
52. Li, F., Zhu, Y.: Smoothing and clustering guided image decolorization. Image Anal. Stereol. **40**(1), 17–27 (2021)
53. Li, P., Chen, W., Ge, H., Ng, M.K.: $\ell_1 - \alpha\ell_2$ minimization methods for signal and image reconstruction with impulsive noise removal. Inverse Probl. **36**(5), 055009 (2020)
54. Li, X., Yang, X., Zeng, T.: A three-stage variational image segmentation framework incorporating intensity inhomogeneity information. SIAM J. Imaging Sci. **13**(3), 1692–1715 (2020)
55. Li, Y., Wu, C.L., Duan, Y.: The TV$_p$ regularized Mumford-Shah model for image labeling and segmentation. IEEE Trans. Image Process. **29**, 7061–7075 (2020)
56. Lou, Y., Osher, S., Xin, J.: Computational aspects of constrained $L_1 - L_2$ minimization for compressive sensing. In: Le, T., Hoai, A., Pham, D., Tao, N., Ngoc, T. (eds.) Modelling, Computation and Optimization in Information Systems and Management Sciences, pp. 169–180. Springer (2015)
57. Lou, Y., Yan, M.: Fast $L_1 - L_2$ minimization via a proximal operator. J. Sci. Comput. **74**(2), 767–785 (2018)
58. Lou, Y., Yin, P., He, Q., Xin, J.: Computing sparse representation in a highly coherent dictionary based on difference of L_1 and L_2. J. Sci. Comput. **64**(1), 178–196 (2015)
59. Lou, Y., Zeng, T., Osher, S., Xin, J.: A weighted difference of anisotropic and isotropic total variation model for image processing. SIAM J. Imaging Sci. **8**(3), 1798–1823 (2015)
60. Luong, Q.-T.: Color in computer vision. In: Chen, C.H., Pau, L.F., Wang, P.S.P. (eds.) Handbook of Pattern Recognition and Computer Vision, pp. 311–368. World Scientific, Singapore (1993)
61. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings, vol. 2, pp. 416–423. IEEE (2001)
62. Merriman, B., Bence, J.K., Osher, S.J.: Motion of multiple junctions: a level set approach. J. Comput. Phys. **112**(2), 334–363 (1994)
63. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. Commun. Pure Appl. Math. **42**(5), 577–685 (1989)
64. Ng, M.K., Chan, R.H., Tang, W.-C.: A fast algorithm for deblurring models with Neumann boundary conditions. SIAM J. Sci. Comput. **21**(3), 851–866 (1999)
65. Ono, S.: L_0 gradient projection. IEEE Trans. Image Process. **26**(4), 1554–1564 (2017)
66. Paschos, G.: Perceptually uniform color spaces for color texture analysis: an empirical evaluation. IEEE Trans. Image Process. **10**(6), 932–937 (2001)
67. Pham Dinh, T., Le Thi, H.A.: Convex analysis approach to DC programming: theory, algorithms and applications. Acta Math. Vietnam **22**(1), 289–355 (1997)
68. Pham Dinh, T., Le Thi, H.A.: A DC optimization algorithm for solving the trust-region subproblem. SIAM J. Optim. **8**(2), 476–505 (1998)
69. Pock, T., Chambolle, A., Cremers, D., Bischof, H.: A convex relaxation approach for computing minimal partitions. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009, pp. 810–817. IEEE (2009)

70. Pock, T., Cremers, D., Bischof, H., Chambolle, A.: An algorithm for minimizing the Mumford-Shah functional. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1133–1140. IEEE (2009)
71. Potts, R.B.: Some generalized order-disorder transformations. In: Mathematical Proceedings of the Cambridge Philosophical Society, vol. 48, pp. 106–109. Cambridge University Press, Cambridge (1952)
72. Rahimi, Y., Wang, C., Dong, H., Lou, Y.: A scale-invariant approach for sparse signal recovery. SIAM J. Sci. Comput. **41**(6), A3649–A3672 (2019)
73. Rockafellar, R.T., Wets, R.J.-B.: Variational Analysis, vol. 317. Springer, Heidelberg (2009)
74. Storath, M., Weinmann, A.: Fast partitioning of vector-valued images. SIAM J. Imaging Sci. **7**(3), 1826–1852 (2014)
75. Sural, S., Qian, G., Pramanik, S.: Segmentation and histogram generation using the HSV color space for image retrieval. In: Proceedings. International Conference on Image Processing, vol. 2, pp. II–II. IEEE (2002)
76. Toure, S., Diop, O., Kpalma, K., Maiga, A.S.: Best-performing color space for land-sea segmentation. In: 2018 41st International Conference on Telecommunications and Signal Processing (TSP), pp. 1–5. IEEE (2018)
77. Vese, L.A., Chan, T.F.: A multiphase level set framework for image segmentation using the Mumford and Shah model. Int. J. Comput. Vision **50**(3), 271–293 (2002)
78. Wang, C., Tao, M., Chuah, C.-N., Nagy, J., Lou, Y.: Minimizing L_1 over L_2 norms on the gradient. Inverse Probl. **38**(6), 065011 (2022)
79. Wang, C., Tao, M., Nagy, J.G., Lou, Y.: Limited-angle CT reconstruction via the L_1/L_2 minimization. SIAM J. Imaging Sci. **14**(2), 749–777 (2021)
80. Wang, C., Yan, M., Rahimi, Y., Lou, Y.: Accelerated schemes for the L_1/L_2 minimization. IEEE Trans. Signal Process. **68**, 2660–2669 (2020)
81. Wang, D., Wang, X.-P.: The iterative convolution-thresholding method (ICTM) for image segmentation. Pattern Recogn. **130**, 108794 (2022)
82. Wang, X.-F., Huang, D.-S., Huan, X.: An efficient local Chan-Vese model for image segmentation. Pattern Recogn. **43**(3), 603–618 (2010)
83. Wang, Y., Yang, J., Yin, W., Zhang, Y.: A new alternating minimization algorithm for total variation image reconstruction. SIAM J. Imaging Sci. **1**(3), 248–272 (2008)
84. Wu, T., Mao, Z., Li, Z., Zeng, Y., Zeng, T.: Efficient color image segmentation via quaternion-based L_1/L_2 regularization. J. Sci. Comput. **93**(1), 9 (2022)
85. Wu, T., Shao, J., Gu, X., Ng, M.K., Zeng, T.: Two-stage image segmentation based on nonconvex $\ell_2 - \ell_p$ approximation and thresholding. Appl. Math. Comput. **403**, 126168 (2021)
86. Wu, T., Zhao, Y., Mao, Z., Shi, L., Li, Z., Zeng, Y.: Image segmentation via Fischer-Burmeister total variation and thresholding. Adv. Appl. Math. Mech. **14**(4), 960–988 (2022)
87. Xu, Z., Chang, X., Xu, F., Zhang, H.: $L_{1/2}$ regularization: a thresholding representation theory and a fast solver. IEEE Trans. Neural Netw. Learn. Syst. **23**(7), 1013–1027 (2012)
88. Yin, P., Lou, Y., He, Q., Xin, J.: Minimization of ℓ_{1-2} for compressed sensing. SIAM J. Sci. Comput. **37**(1), A536–A563 (2015)
89. You, J., Jiao, Y., Lyu, X.L., Zeng, T.: A nonconvex model with minimax concave penalty for image restoration. J. Sci. Comput. **78**(2), 1063–1086 (2019)
90. Zeng, C., Wu, C.L.: On the edge recovery property of noncovex nonsmooth regularization in image restoration. SIAM J. Numer. Anal. **56**(2), 1168–1182 (2018)
91. Zhang, S., Xin, J.: Minimization of transformed L_1 penalty: closed form representation and iterative thresholding algorithms. Commun. Math. Sci. **15**(2), 511–537 (2017)
92. Zhang, S., Xin, J.: Minimization of transformed L_1 penalty: theory, difference of convex function algorithm, and robust application in compressed sensing. Math. Program. **169**(1), 307–336 (2018)