Noise-Robust SLIC Superpixel for Natural Images

Li Dong* and Jiantao Zhou*†
*Department of Computer and Information Science, Faculty of Science and Technology,
University of Macau, Macau 999078, China
† UMacau Zhuhai Research Institute, Zhuhai 519080, China.
{yb47452, jtzhou}@umac.mo

***Abstract*—Superpixel algorithm aims to semantically group neighboring pixels into a coherent region. It could significantly boost the performance of the subsequent vision processing task such as image segmentation. Recently, the work *simple linear iterative clustering* (SLIC) [1] has drawn huge attention for its state-of-the-art segmentation performance and high computational efficiency. However, the performance of SLIC is dramatically degraded for noisy images. In this work, we propose three measures to improve the robustness of SLIC against noise: 1) a new pixel intensity distance measurement is designed by explicitly considering the within-cluster noise variance; 2) the spatial distance measurement is refined by exploiting the variation of pixel locations in a cluster; and 3) a noise-robust estimator is proposed to update the cluster centers by excluding the possible outliers caused by noise. Extensive experimental results on synthetic noisy images validate the effectiveness of those improvements. In addition, we apply the proposed noise-robust SLIC to superpixel-based noise level estimation task to demonstrate its practical usage.**

1. Introduction

Superpixel is originally from image over-segmentation. It is commonly defined as a local, semantically coherent region in the image. As a preprocessing step, superpixels capture the pixel redundancy and image structure. Due to this adaptiveness to the images structure, superpixels are often employed to replace the regular square patches, and consequently speedup the subsequent vision processing tasks. Therefore, superpixel scheme has already been a crucial module of many computer vision algorithms, such as image segmentation [2], [3], object recognition [4], depth estimation [5] and stereo image reconstruction [6].

In general, the existing superpixel algorithms can be classified into two categories: graph-based and gradient-ascent-based methods. Graph-based methods model the image as an undirected weighted graph, in which each pixel is regarded as a node, and edge weights are proportional to the similarity between neighboring pixels. Superpixels are generated by solving a global minimization optimization problem defined over the graph. In [7], Shi *et al.* proposed Normalized Cuts (N-Cut) algorithm. Utilizing the image structure and texture priors, this scheme recursively partitions the pixels by minimizing the cost function defined on partitioned boundaries. N-Cut generates regular superpixels with compact shape, while the main weakness is high computational complexity. Later on, Felzenszwalb *et al.* [8] proposed an efficient graph-based superpixel method by finding the minimum spanning tree on the graph model.

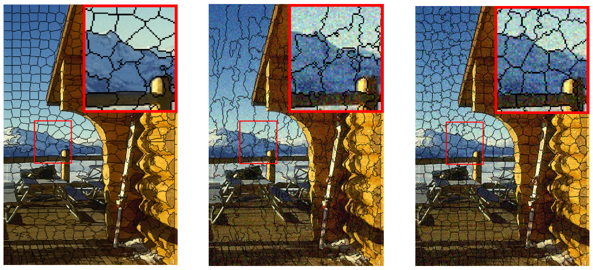

(a) SLIC+Clean (b) SLIC+Noisy (c) Proposed+Noisy

Figure 1: SLIC [1] is not robust to noise. (a) SLIC on clean image (b) SLIC on Gaussian noise polluted image ($\sigma = 20$) (c) Our proposed noise-robust SLIC on the noisy image.

Instead of using graph model, gradient-ascent-based algorithms attempt to formulate the superpixel segmentation as an unsupervised clustering problem. Technically, this clustering procedure is triggered by coarse cluster initialization, and then iteratively refine the pixel's assignment and cluster centers until some convergence criterion is reached. In early works, Mean-shift [9] grouped the pixels belonging the same mode into single superpixel. Other similar but faster gradient-ascent methods include Quick-shift [10] and Turbopixel [11]. More recently, Achanta *et al.* [1] proposed an efficient superpixel scheme *simple linear iterative clustering* (SLIC) [1]. It adapts k-means clustering algorithm on local neighboring pixels. SLIC has drawn huge attention for its state-of-the-art segmentation performance and extremely low computational cost. However, as shown in Figure 1-(a)(b), one implicit premise of SLIC is the input images should be clean. Otherwise, the performance will be signifi-

This work was supported in part by the Macau Science and Technology Development Fund under grants FDCT/009/2013/A1, FDCT/046/2014/A1, in part by the Research Committee at University of Macau under grants MRG007/ZJT/2015/FST, MRG021/ZJT/2013/FST, MYRG2014-00031-FST, MYRG2015-00056-FST, and MYRG2016-00137-FST, and in part by the National Science Foundation of China under grant 61402547.

978-1-5090-3555-7/16 $31.00 © 2016 IEEE
DOI 10.1109/CCBD.2016.64

cantly degraded. Unfortunately, in practice, the input images are typically polluted by noise to some extent. One intuitive solution to tackle this problem is to perform denoising before superpixel generation, but for some applications such as superpixel-based noise level estimation [12], they require to conduct superpixel algorithm directly on noisy images. This motivates us to develop noise-robust superpixel algorithm, which aims at preserving the boundary adherence of superpixels. Figure 1-(c) illustrates the superpixels obtained by our algorithm. In the enlarged region, one can observe the most boundaries near mountain peaks are preserved.

In this work, we propose three measures to improve the robustness of SLIC against noise: 1) a new pixel intensity distance measurement is designed by explicitly considering the within-cluster noise variance; 2) the spatial distance measurement is refined by exploiting the variation of pixel locations in a cluster; and 3) a noise-robust estimator is proposed to update the cluster centers by excluding the possible outliers caused by noise. Extensive experimental results on synthetic noisy images validate the effectiveness of those improvements. In addition, we apply the proposed noise-robust SLIC to superpixel-based noise level estimation task to demonstrate its usage in real application.

The rest of paper is organized as follows. In Section 2, we briefly review the conventional SLIC superpixel scheme. We propose noise-robust SLIC in Section 3. Experimental results on both clean and synthetic noisy images are given in Section 4, and we conclude this work in Section 5.

2. SLIC Superpixel

Technically, SLIC is an iterative unsupervised clustering procedure, which can be roughly regarded as a local version of the conventional k-means scheme. More specifically, it consists of four components: initialization, pixel assignment, cluster center update and post-processing. Before introducing those modules, we shall first define the mathematical notations of augmented pixel and cluster centers. A augmented pixel $\mathbf{p}$ is formed by concatenating pixel's intensities and location: $\mathbf{p} = [\mathbf{t}; \mathbf{s}]$, where $\mathbf{t} = [l, a, b]^\top$ denotes the pixel's color intensities in 3D CIELAB color space, and $\mathbf{s} = [x, y]^\top$ represents the pixel's spatial location in 2D image plane. Cluster center $\mathbf{c}$ has the same format with augmented pixel. We put a subscript i to indicate the index of $\mathbf{p}$ and $\mathbf{c}$. For instance, $\mathbf{c}_i$ and $\mathbf{p}_j$ are the i-th cluster center and the j-th augmented pixel, respectively. We use $l(\mathbf{p}_j) = i$ to denote that augmented pixel $\mathbf{p}_j$ is assigned to i-th cluster.

Initialization: SLIC samples K cluster centers with a regular grid sampling step S. Here the parameter S is provided by users, and $K = \sqrt{WH/S^2}$, where W, H are width and height of image, respectively. To avoid seeding on image edges, those cluster centers are further moved to the locations corresponding to the lowest gradient position in a 3×3 neighborhood.

Pixel assignment: the augmented pixel $\mathbf{p}_j$ is assigned to the nearest cluster, whose search region overlaps its location. Obviously, this task can only be done with a distance measurement, which is used to determine the cluster that augmented pixel should be assigned to. SLIC suggests to compute the distance by combining color similarity and spatial proximity. More concretely, the color distance and spatial distance between cluster center $\mathbf{c}_i$ and pixel $\mathbf{p}_j$ are defined as

$$\begin{aligned} d_t(\mathbf{c}_i, \mathbf{p}_j) &= \|\mathbf{t}_i - \mathbf{t}_j\|_2 \quad (1) \\ d_s(\mathbf{c}_i, \mathbf{p}_j) &= \|\mathbf{s}_i - \mathbf{s}_j\|_2 \quad (2) \end{aligned}$$

where $\|\cdot\|_2$ denotes the l_2 norm. Then, SLIC introduces a weighting factor w to fuse (1) and (2) into a one distance measurement

$$d_{SLIC}(\mathbf{c}_i, \mathbf{p}_j) = \sqrt{d_t^2(\mathbf{c}_i, \mathbf{p}_j) + w \cdot d_s^2(\mathbf{c}_i, \mathbf{p}_j)} \quad (3)$$

The weighting factor w also play two other roles. Firstly, it implicitly scales (1) and (2) into the same scale. Secondly, w balances the relative importance between (1) and (2). In practice, this parameter is empirically set by users.

Cluster center update: once each augmented pixel has been assigned to the nearest cluster, the cluster center update procedure is triggered to recompute the cluster centers by averaging all the augmented pixels associated with the cluster

$$\mathbf{c}_i = \frac{1}{|\mathcal{I}_i|} \sum_{j \in \mathcal{I}_i} \mathbf{p}_j \quad (4)$$

Here $\mathcal{I}_i$ is the set containing all the indices of augmented pixels that are assigned to i-th cluster, *i.e.*, $\mathcal{I}_i = \{j \mid l(\mathbf{p}_j) = i\}$. $|\cdot|$ returns the cardinal number of the corresponding set.

The aforementioned pixel assignment and cluster center update procedures are continuously repeated until some predefined termination condition is met, *e.g.*, fixing the number of iterations as 10.

Post-processing: the above clustering procedure does not enforce connectivity. Thus after completion of clustering, some “isolated” pixels that do not belong to the same connected component may exist. To resolve problem, SLIC reassigns those pixels to their nearest cluster using a connected components algorithm.

SLIC efficiently produces satisfactory superpixel segmentation for clean images. However, for noisy images, the generated superpixels are not consistent with the clean ones. By further examining each step of SLIC, we found two key steps are pixel assignment and cluster center update, in which the color and spatial distance measure (1) (2) and update procedure (4) play critical roles. Obviously, those distance measure and update procedure do not take the noise into account. We believe this is the main reason responsible for the bad superpixel segmentation performance. As will be seen in next section, we propose a new distance measure with a new cluster center update procedure to tackle this problem, by explicitly considering the noise influence.

3. Proposed Noise-Robust SLIC Superpixel

Our proposed scheme has similar algorithmic framework with SLIC. For clarity, we focus on designing the new pixel assignment and cluster center update procedures. The initial-

Algorithm 1 Noise-Robust SLIC Superpixel

```
Input: Noisy image, sampling step $S$, weighting factor $w$.
Initialize: Initialize the cluster centers for each cluster $i$
    $\mathbf{c}_i = [\mathbf{t}_i; \mathbf{s}_i] = [l_i, a_i, b_i, x_i, y_i]^\mathsf{T}$ by uniformly sampling
    in regular grid with step $S$. Set label $l(\mathbf{p}_j) = -\infty$ and
    distance $d(\mathbf{p}_j) = +\infty$ for each augmented pixel $\mathbf{p}_j$.
repeat
    for each cluster $i$ do
        for each pixel $\mathbf{p}_j$ in a $2S \times 2S$ region around the
        cluster center $\mathbf{c}_i$ do
            Compute the color distance using (5):
                $D_t(\mathbf{c}_i, \mathbf{p}_j) = \left\|\mathbf{\Phi}_i^{-1} \cdot (\mathbf{t}_i - \mathbf{t}_j)\right\|_2$
            Compute the spatial distance using (7):
                $D_s(\mathbf{c}_i, \mathbf{p}_j) = \left\|\mathbf{\Psi}_i^{-1} \cdot (\mathbf{s}_i - \mathbf{s}_j)\right\|_2$
            Compute the total distance:
                $D_{ours}(\mathbf{c}_i, \mathbf{p}_j) = \sqrt{D_t^2(\mathbf{c}_i, \mathbf{p}_j) + wD_s^2(\mathbf{c}_i, \mathbf{p}_j)}$
            if $D_{ours}(\mathbf{c}_i, \mathbf{p}_j) < d(j)$ then
                $d(\mathbf{p}_j) \leftarrow D_{ours}(\mathbf{c}_i, \mathbf{p}_j)$ and $l(\mathbf{p}_j) \leftarrow i$
            end if
        end for
    end for
    Update the cluster center using (15):
        $\mathbf{c}_i = \frac{1}{|\mathcal{M}_i|} \sum_{\mathbf{p}_j \in \mathcal{M}_i} \mathbf{p}_j$
until max iteration number is reached
```

ization and post-processing modules remains the same as the conventional SLIC. The entire algorithm is summarized in Algorithm 1.

3.1. Color Distance

Recall that, in (1), SLIC computes the color distance between augmented pixel with cluster center by directly measuring their Euclidean distance. This method suits for clean images because all pixels are genuine. However, for noisy images, the existence of noise would bias this distance measurement. To account for the impact of noise, we redesign the color distance measurement (1) as

$$D_t(\mathbf{c}_i, \mathbf{p}_j) = \left\|\mathbf{\Phi}_i^{-1} \cdot (\mathbf{t}_i - \mathbf{t}_j)\right\|_2 \tag{5}$$

where $\mathbf{\Phi}_i$ is a 3×3 diagonal matrix, which captures the color variance in each color channel for the i-th cluster. It is defined as

$$\mathbf{\Phi}_i = \begin{bmatrix} \sigma_l & 0 & 0 \\ 0 & \sigma_a & 0 \\ 0 & 0 & \sigma_b \end{bmatrix} \tag{6}$$

where σ_l, σ_a and σ_b are the standard deviation of each dimension in $[l, a, b]^\mathsf{T}$ color intensity vector which belongs to i-th cluster. Note that incorporating the inter-channel covariances is expected to slightly improve the final performance, but computational complexity will be a huge burden. Thus in this work, we assume the intensities in three color channels are mutually independent.

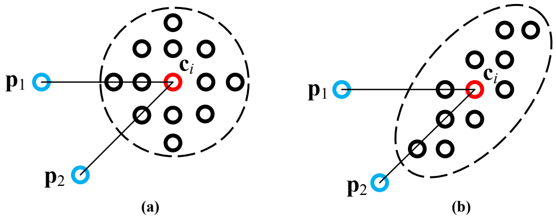


Figure 2: Spatial distance measurement (2) on regular and irregular shaped clusters, $d_s(\mathbf{c}_i, \mathbf{p}_1) = d_s(\mathbf{c}_i, \mathbf{p}_2)$ is assumed in both cases. (a) for regular-shaped cluster, (2) truthfully reflects that $\mathbf{p}_1$ and $\mathbf{p}_2$ has same possibility to be assigned to i-th cluster; (b) for irregular-shaped cluster, $\mathbf{p}_2$ has higher possibility to be assigned to i-th cluster than that of $\mathbf{p}_1$, while (2) still yields the same distance for $\mathbf{p}_1$ and $\mathbf{p}_2$.

3.2. Spatial Distance

Similarly, in (2), SLIC computes the spatial distance using spatial Euclidian distance. The smaller distance is, the higher possibility the pixel ought to be assigned to the cluster. Generally, (2) works well for regular-shaped clusters (*e.g.*, smooth regions). As an example, in Figure 2-(a), $d_s(\mathbf{c}_i, \mathbf{p}_1) = d_s(\mathbf{c}_i, \mathbf{p}_2)$ could truthfully reflect that $\mathbf{p}_1$ and $\mathbf{p}_2$ has same possibility to be assigned to i-th cluster. However, for irregular-shape clusters (*e.g.*, prominent edges) shown Figure 2-(b), (2) still treats $\mathbf{p}_1$ and $\mathbf{p}_2$ equally, which cannot reflects that $\mathbf{p}_2$ has higher possibility than $\mathbf{p}_1$'s to be assigned to i-th cluster. This is because it neglects the location distribution of pixels, *i.e.*, the spatial shape of the cluster. To utilize this information, we exploit the variances of spatial locations within a cluster. Specifically, we suggest to compute the spatial distance between cluster center $\mathbf{c}_i$ and augmented pixel $\mathbf{p}_j$ as

$$D_s(\mathbf{c}_i, \mathbf{p}_j) = \left\|\mathbf{\Psi}_i^{-1} \cdot (\mathbf{s}_i - \mathbf{s}_j)\right\|_2 \tag{7}$$

where $\mathbf{\Psi}_i$ is a 2×2 matrix, which is used to describle the pixel position variance of the irregular shape exhibited by i-th cluster in image plane. It is defined as

$$\mathbf{\Psi}_i = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{bmatrix} = \mathsf{sqrt}\{\mathbf{S}_i\} \tag{8}$$

Here function $\mathsf{sqrt}\{\cdot\}$ performs *element-wise* square root on the input matrix; $\mathbf{S}_i$ is the sample covariance matrix, which is calculated by

$$\mathbf{S}_i = \frac{1}{|\mathcal{I}_i| - 1} \sum_{j \in \mathcal{I}_i} (\mathbf{s}_j - \bar{\mathbf{s}})(\mathbf{s}_j - \bar{\mathbf{s}})^\mathsf{T} \tag{9}$$

where $\bar{\mathbf{s}}$ is the mean vector defined as

$$\bar{\mathbf{s}} = \frac{1}{|\mathcal{I}_i|} \sum_{j \in \mathcal{I}_i} \mathbf{s}_j \tag{10}$$

3.3. Cluster Center Update

The conventional SLIC updates i-th cluster center by averaging all the augmented pixels associated with this cluster. However, some prominent image edges/structures are easily mixed into a cluster as outliers. This phenomenon is more pronounced for noisy images. Those outliers make the original simple mean estimator for cluster center exhibit large bias and variance. To make a robust estimation, we propose to exclude the outliers by restricting the range of augmented pixels. Since the assumed noise model is spatial-independent, the added noise does not influence the update of $[x, y]^\mathsf{T}$ location component $\mathbf{s}$. In the sequel, we focus on updating the $[l, a, b]^\mathsf{T}$ color intensities component $\mathbf{t}$.

Specifically, for i-th cluster, we collect and pack all its associated $[l, a, b]^\mathsf{T}$ vectors into a single matrix $\mathbf{T}_i$

$$\mathbf{T}_i = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{|\mathcal{I}_i|}] \tag{11}$$

Then the median and standard deviation (std) of the $[l, a, b]^\mathsf{T}$ vector for $\mathbf{T}_i$ are computed as follows

$$\mathbf{m}_i = \mathsf{median}\{\mathbf{T}_i\} \tag{12}$$

$$\boldsymbol{\sigma}_i = \mathsf{std}\{\mathbf{T}_i\} \tag{13}$$

Note that the function $\mathsf{median}\{\cdot\}$ and $\mathsf{std}\{\cdot\}$ compute the median and std of $[l, a, b]^\mathsf{T}$ vector along each row independently. Instead of using all $[l, a, b]^\mathsf{T}$ vectors within i-th cluster, we deliberately select the augmented pixels whose $[l, a, b]^\mathsf{T}$ component resides in a restricted range. The indices of those selected augmented pixels form a new set $\mathcal{M}_i$, which can be expressed as

$$\mathcal{M}_i = \{j \mid \mathbf{t}_j \in [\mathbf{m}_i - \alpha\boldsymbol{\sigma}_i, \mathbf{m}_i + \alpha\boldsymbol{\sigma}_i]\} \tag{14}$$

where α is a predefined constant controlling the tolerance of deviation. It is set as 2 in this work. Finally, the cluster center is updated by averaging the augmented pixels which belong to set $\mathcal{M}_i$, *i.e.*,

$$\mathbf{c}_i = \frac{1}{|\mathcal{M}_i|} \sum_{j \in \mathcal{M}_i} \mathbf{p}_j \tag{15}$$

4. Experimental Results

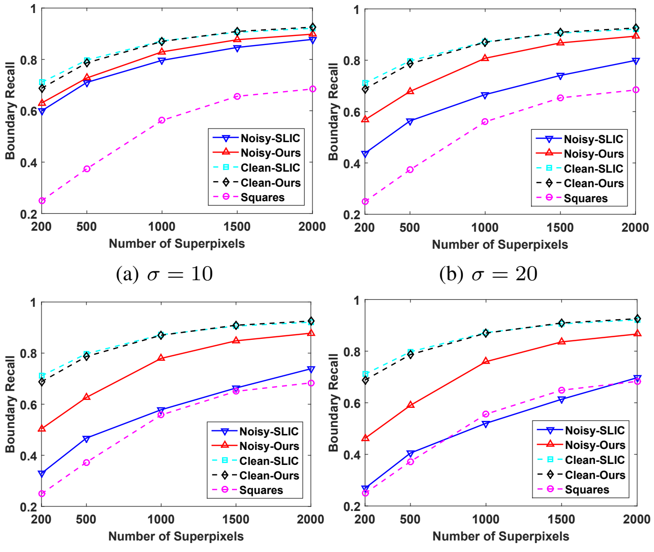
In this section, we conduct experiments to evaluate the performance of our method on Berkeley Segmentation Dataset (BSD500) [13]. BSD500 image set is composed of 500 natural images including landscapes, architectures, animal and human portrait in size 481×321. For each image, BSD500 also provides its corresponding human annotated edge maps. Two test images are shown in Figure 3. In our experiments, the synthetic noisy images are created by enforcing additive white Gaussian noise (AWGN) with a variety of noise levels σ. The default grid sampling step S is 16, and the weighting factor w is empirically fixed as 4 in all experiments.

We first compare the noise-robust ability between SLIC and our proposed method. For a superpixel algorithm, one of the most important property is its ability to adhere to image boundaries. This ability can be quantitatively measured by boundary recall [1], which indicates the percentage of the ground-truth edges that are resided within at least two pixels of a superpixel boundary. Higher boundary recall is preferred because it implies most of the image edges are successfully detected. Usually, the ground-truth edges are obtained by human annotation. In our experiments, we adopt the edge maps provided by BSD500 image set as ground-truth edges. The boundary recalls of SLIC and our method on both clean and noisy images are illustrated in Figure 4, in which the noise level σ ranges from 10 to 30. As a baseline, we present the boundary recall curves produced by uniform squares superpixels (denoted as "Squares"). From the experimental results shown in 4, we can draw the following conclusions: 1) for clean images, our method achieves comparable boundary recall score compared with SLIC; 2) for noisy images, the boundary recall is moderately declined with the increase of noise levels; but 3) our method still outperforms SLIC, especially for $\sigma > 10$; 4) with increment of the number of superpixels, the performance of our method on noisy images generally approaches to the clean ones; and finally 5) for large noise levels, *e.g.*, $\sigma = 40$, the performance of SLIC on noisy image is even inferior to uniform square benchmark superpixel. This again verifies that the SLIC is totally failed when handling heavy noise polluted images.

Furthermore, we visually compare the superpixel seg-

(a) 118035.jpg (b) 100007.jpg

Figure 3: Test images from BSD500 image set.



(a) $\sigma = 10$ (b) $\sigma = 20$

(c) $\sigma = 30$ (d) $\sigma = 40$

Figure 4: Comparison of boundary recall performance on BSD500 image set. For clean images, our method achieves comparable boundary recall with that of SLIC; for noisy case, our method outperforms SLIC.

mentation results yielded by our method with that of SLIC, both on clean and noisy images. As can be seen from Figure 5, the superpixel segmentation performance of proposed method is comparable with conventional SLIC for the clean cases. However, for noisy images, SLIC cannot correctly group perceptually meaningful regions into superpixels, *i.e.,* losing the ability to adhere to image edges. In the contrast, our proposed method could generate regular superpixels that are consistent with the clean ones.

Finally, we apply our method to superpixel-based noise level estimation application to demonstrate its piratical usage. Noise level estimation is a crucial preprocessing step in many image processing tasks such as blind image denoising [14]. Recently, Wu *et al.* [12] proposed a superpixel-based noise level estimation scheme, in which the noise level was estimated from the selected superpixels whose clean versions are assumed to be homogeneous. In this algorithm, the most crucial step is to generate superpixels. Wu suggested to use N-Cut [7] to complete this task. However, N-Cut does not explicitly taking the noise into account. The segmentation performance become worse with the increment of noise levels. Consequently, the estimation accuracy of noise level is dramatically dropped, especially for large noise levels.

We improve Wu's method by replacing the N-Cut superpixel module with our proposed scheme. The synthetic ground-truth noise level σ ranges from 10 to 40, and the estimated noise level $\hat{\sigma}$ that is close to the ground-truth σ is regarded as the best. The estimation accuracy comparison results are tabulated in Table 1, from which one can observe our method consistently achieves the highest estimation accuracy for all noise levels.

TABLE 1: Comparison of noise level estimation results. The best results are emphasized with boldface.

Image	True σ	[12]+N-Cut	$\lvert\sigma - \hat{\sigma}\rvert$	[12]+Ours	$\lvert\sigma - \hat{\sigma}\rvert$
118035	10	10.57	0.57	**10.33**	**0.33**
	20	20.15	0.15	**20.05**	**0.05**
	30	29.05	0.95	**29.87**	**0.13**
	40	39.09	0.91	**39.75**	**0.25**
100007	10	9.61	0.39	**9.94**	**0.06**
	20	19.09	0.91	**19.61**	**0.39**
	30	28.34	1.66	**29.13**	**0.87**
	40	37.90	2.10	**38.54**	**1.46**

5. Conclusion

In this work, we suggested three measures to improve the robustness of SLIC against noise. The pixel intensity and spatial distance measurements were redesigned by explicitly taking the noise into account. Furthermore, a robust cluster center estimator is proposed by excluding the possible outliers. Extensive experimental results on synthetic noisy images validate the effectiveness of those improvements. We also applied the proposed noise-robust SLIC to superpixel-based noise level estimation task to demonstrate its usage in real application. However, this work only considers the AWGN noise model. In the future, we would like to investigate more noise types such as speckle noise.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, 2012.

[2] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum, "Lazy snapping," in *ACM Trans. Graphics*, vol. 23, no. 3. ACM, 2004, pp. 303–308.

[3] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Proc. IEEE Intl Conf. Computer Vision*. IEEE, 2003, pp. 10–17.

[4] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configurations: Combining segmentation and recognition," in *Proc. IEEE Intl Conf. Computer Vision*, vol. 2. IEEE, 2004, pp. II–326.

[5] C. L. Zitnick and S. B. Kang, "Stereo for image-based rendering using image over-segmentation," *Intl J. Computer Vision*, vol. 75, no. 1, pp. 49–65, 2007.

[6] D. Hoiem, A. N. Stein, A. A. Efros, and M. Hebert, "Recovering occlusion boundaries from a single image," in *Proc. IEEE Intl Conf. Computer Vision*. IEEE, 2007, pp. 1–8.

[7] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[8] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Intl J. Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[9] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, 2002.

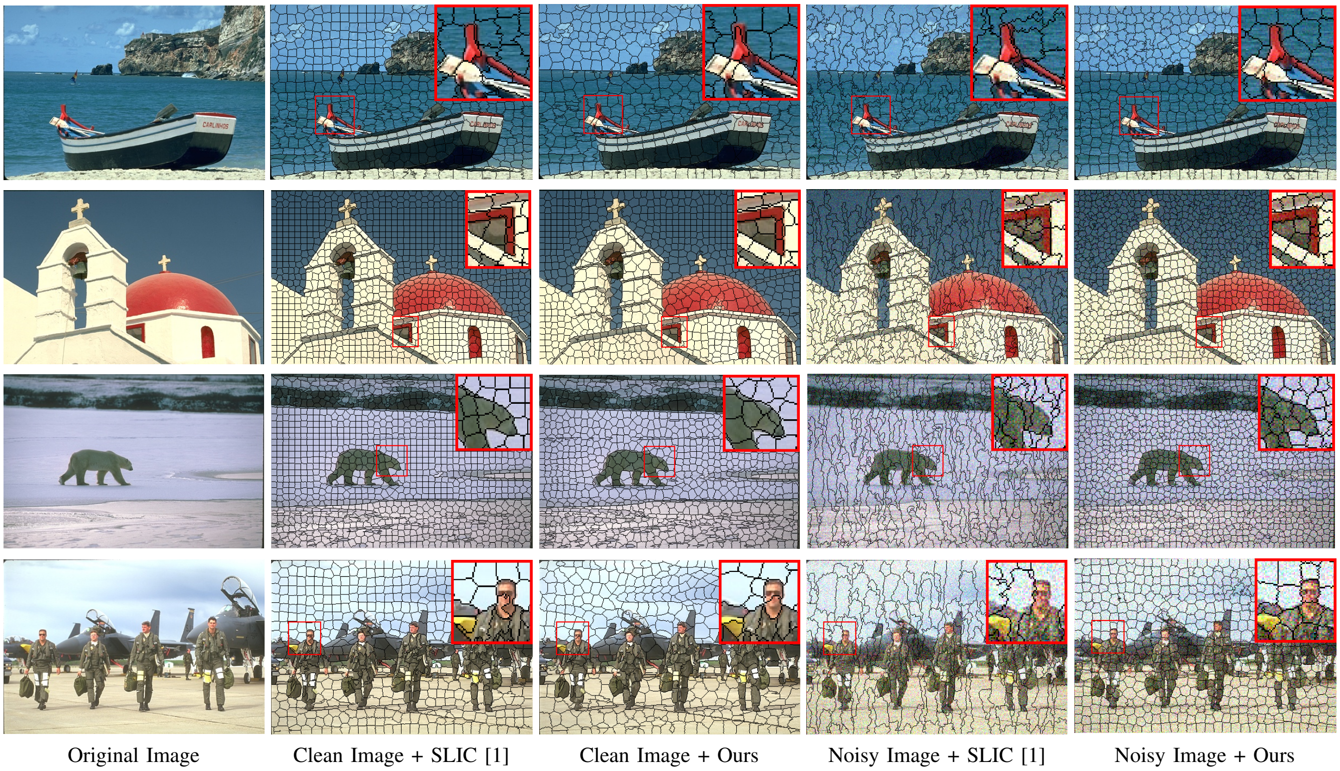
[10] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. European Conf. Computer Vision*. Springer, 2008, pp. 705–718.

[11] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi, "Turbopixels: Fast superpixels using geometric flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2290–2297, 2009.

[12] C.-H. Wu and H.-H. Chang, "Superpixel-based image noise variance estimation with local statistical assessment," *EURASIP Int. J. Image Video Process.*, vol. 2015, no. 1, pp. 1–12, Dec. 2015.

[13] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[14] X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 5226–5237, Dec. 2013.



Original Image Clean Image + SLIC [1] Clean Image + Ours Noisy Image + SLIC [1] Noisy Image + Ours

Figure 5: Visual comparison for superpixel segmentation between SLIC [1] and our proposed method on both clean and noisy images. From top to bottom, the added noise levels are $10, 20, 30$ and 40, respectively. From left to right: original images, SLIC applied on clean images, our method applied on clean images, SLIC applied on noisy images, our method applied on noisy images. For better visual comparison, part of image is enlarged and shown at the top-right corner.