Bayesian Adaptive Superpixel Segmentation

Supplemental Material

Roy Uziel Meitar Ronen Oren Freifeld
Computer Science, Ben-Gurion University
uzielr@post.bgu.ac.il meitarr@post.bgu.ac.il orenfr@cs.bgu.ac.il

Abstract

This document contains additional visual examples/demonstrations and few additional explanations that were omitted from the main text due to page limits. In the first section, we show various kinds of additional results: 1) for the face-detection experiment; 2) visual comparisons with other methods on additional examples from the BSD and SBD datasets (mentioned in the paper); 3) examples showing the adaptation of K (i.e.*, the number of superpixels), to the image content; 4) visualization of the splits and merges; 5) an illustration of the effect of α on splits and merges, and thus on the eventual K. In the second section we focus on the connectivity constraints and the parallelized implementation we used for the label updates. In the last section, we provide the equations for the sufficient stastics and a closed form of the posterior updates mentioned in section 4 in the paper.*

Contents

1. Additional Visual Results and Visual Demonstrations of Splits and Merges **1**
1.1. Face Detection Results 1
1.2. Additional Visual Comparisons on Images from the BSD and SBD Datasets 1
1.3. Different Values of K_{final} 1
1.4. Visualization of Splits and Merges 1
1.5. Visualizing the Effect of α 1

2. Connectivity, Simple Points, and Parallelization **2**
2.1. Connectivity and Simple Points 2
2.2. Parallelization 2

3. Computing the Conditional Modes and the Hastings Ratios **2**
3.1. The Closed-form Solutions for the Conditional Models 2
3.2. Hastings Ratios 3

1. Additional Visual Results and Visual Demonstrations of Splits and Merges

1.1. Face Detection Results

We show here additional results of the face-detection downstream application experiment which was described in our paper. For this experiment we used images of groups of people from [12]. We counted the number of faces detected by a state-of-the-art face detector [14] in each image. Next, for each superpixel method, we created mean-color images, by coloring each superpixel with its mean color. Then, we ran the face detector again, this time on the mean-color images. For each method and its mean-color images, we counted how many faces the detector found. Figs. 1, 2, 3, and 4 demonstrate how BASS preserves more details compared with the other methods, resulting in more interpretable output images which in turn lead to higher face-detection rates.

1.2. Additional Visual Comparisons on Images from the BSD and SBD Datasets

Figures 5–14 contain additional comparisons between BASS and the other methods.

1.3. Different Values of K_{final}

Figure 15 demonstrates how BASS adapts the number of superpixels (K) to the image.

1.4. Visualization of Splits and Merges

Figures 16, 17 and 18 illustrate the splits and merges.

1.5. Visualizing the Effect of α

Figure 19 demonstrates the effect of different values of α.

2. Connectivity, Simple Points, and Parallelization

2.1. Connectivity and Simple Points

In this section we elaborate on the connectivity constraints that were briefly mentioned in Sections 3 and 4 of the paper. Following [6, 7], our approach is based on digital-topology concepts. More specifically, we utilize the concept of *simple points* [3], to ensure that our label updates do not break the (simple-) connectivity of each superpixel.

Changing a label in a (topologically-) valid segmentation might break connectivity. A point whose label can be changed without breaking connectivity is called a simple point. It can be shown that the answer to the question whether a pixel is a simple point or not is a function of only the labels of its neighbors in a 3×3 neighborhood when considering simply-connected regions, or a 5×5 neighborhood when considering connected regions. For more details, see [6, 3].

In our approach, a pixel's label can be changed only in the case it is a simple point. For a binary segmentation (*i.e.*, two classes: "background" and "foreground"), a pixel is a simple point if and only if changing its label does not change the number of *connected components* (CC) in its neighborhood for both classes. This ensures that each superpixel remains a simply-connected region. An example of a simple-point configuration and an example of a non-simple point configuration for the binary case ($K = 2$) are shown in Fig. 21. Since checking whether a pixel is a simple point or not is a function of only the surrounding 3×3 lattice, it can be computed in a short constant time using a precomputed lookup table containing all the possible 2^8 configurations.

In order to generalize to the non-binary case, we used a one-versus-all approach, where the label in question is considered as 1, and all other labels in the surrounding 3×3 lattice get the same label, 0. In other words, when we test whether setting a certain label for the pixel in question will retain connectivity, we consider all the other labels as the same "background" label. Then, we use the binary-case solution to determine if this point is simple or not. Notice that the maximum number of candidate labels that a pixel can get is the number of its adjacent neighbors, 4. Thus, even in the non-binary case the test can be done in a constant and short time.

We allow a pixel to change its label only if in the one-versus-all it is determined to be a simple point. Meaning, if the candidate labels are a, b, c, d, and only the a, b labels passed the one-versus-all simple-point test, then the only possible assignments the pixel can get would be a or b.

2.2. Parallelization

Despite the topology constraints, and following Freifeld *et al.* [7], we utilized the fact that the simple-point test is a function of only the 3×3 surrounding lattice to parallelize our inference. However, while the overly-conservative approach in [7] parallelized over $N/9$ of the pixels, we parallelize over $N/4$ of the pixels (see Fig. 20), and thus achieve better parallelization of the label updates.

3. Computing the Conditional Modes and the Hastings Ratios

In this section we further explain some inference details that were omitted from the paper due to space limits.

3.1. The Closed-form Solutions for the Conditional Models

Recall that the priors we used are the Normal-Inverse Wishart (NIW) prior and multivariate Normal-Inverse-Gamma (NIG) priors. The key reason that the NIW and NIG priors are used is that both are conjugate to the Gaussian likelihood. Likewise, the Dirichlet-distribution prior is conjugate to the Categorical likelihood. See [8] for more details. Thus, the posteriors are of the same functional form as the priors, and the updates from the priors to the posteriors are given in closed form via sufficient statistics [8]. Particularly, the priors are:

$$\begin{aligned} p(\boldsymbol{\mu}_{j,\boldsymbol{l}}, \boldsymbol{\Sigma}_{j,\boldsymbol{l}}) =& \mathrm{NIW}(\boldsymbol{\mu}_{j,\boldsymbol{l}}, \boldsymbol{\Sigma}_{j,\boldsymbol{l}}; \boldsymbol{m}_{j,\boldsymbol{l}}, \kappa_{j,\boldsymbol{l}}, \boldsymbol{\Lambda}_{j,\boldsymbol{l}}, \nu_{j,\boldsymbol{l}}) \\ p(\boldsymbol{\mu}_{j,\boldsymbol{c}}, \sigma^2_{j,\boldsymbol{c}}) =& \mathrm{NIG}(\boldsymbol{\mu}_{j\ \boldsymbol{c}}, \sigma^2_{j,\boldsymbol{c}}; \boldsymbol{m}_{j,\boldsymbol{c}}, \kappa_{j,\boldsymbol{c}}, a_{j,\boldsymbol{c}}, b_{j,\boldsymbol{c}}) \\ p((\boldsymbol{\pi}_j)_{j=1}^K) =& \mathrm{Dir}((\boldsymbol{\pi}_j)_{j=1}^K; \alpha)\,. \end{aligned} \tag{1}$$

The sufficient statistics [10, 8], for $j \in \{1, \ldots, K\}$, are:

$$\begin{aligned} \boldsymbol{t}_{j,\boldsymbol{l}} &= \sum\nolimits_{i:z_i=j} \boldsymbol{l}_i \in \mathbb{R}^2 & \boldsymbol{t}_{j,\boldsymbol{c}} &= \sum\nolimits_{i:z_i=j} \boldsymbol{c}_i \in \mathbb{R}^3 \\ \boldsymbol{T}_{j,\boldsymbol{l}} &= \sum\nolimits_{i:z_i=j} \boldsymbol{l}_i \boldsymbol{l}_i^T \in \mathbb{R}^{2\times 2} & \boldsymbol{T}_{j,\boldsymbol{c}} &= \sum\nolimits_{i:z_i=j} \boldsymbol{c}_i \odot \boldsymbol{c}_i \in \mathbb{R}^3 \\ n_j &= |\{i : z_i = j\}| \end{aligned} \tag{2}$$

(where $\odot$ is elementwise multiplication).

Hence, by conjugacy, the posteriors are:

$$p(\boldsymbol{\Sigma}_{j,\boldsymbol{l}} | \boldsymbol{z}, (\boldsymbol{x}_i)_{i=1}^N) = \mathrm{IW}(\boldsymbol{\Sigma}_{j,\boldsymbol{l}}; \boldsymbol{\Lambda}^*_{j,\boldsymbol{l}}, \nu^*_{j,\boldsymbol{l}}) \tag{3}$$

$$p(\boldsymbol{\mu}_{j,\boldsymbol{l}} | \boldsymbol{\Sigma}_{j,\boldsymbol{l}}, \boldsymbol{z}, (\boldsymbol{x}_i)_{i=1}^N) = \mathcal{N}\left((\boldsymbol{\mu}_{j,\boldsymbol{l}}; \boldsymbol{m}^*_{j,\boldsymbol{l}}, \frac{\boldsymbol{\Sigma}_{j,\boldsymbol{l}}}{\kappa^*_{j,\boldsymbol{l}}}\right) \tag{4}$$

$$p(\sigma^2_{j,\boldsymbol{c}} | \boldsymbol{z}, (\boldsymbol{x}_i)_{i=1}^N) = \mathrm{IG}(\sigma^2_{j,\boldsymbol{c}}; a^*, b^*) \tag{5}$$

$$p(\boldsymbol{\mu}_{j,\boldsymbol{c}} | \sigma^2_{j,\boldsymbol{c}}, \boldsymbol{z}, (\boldsymbol{x}_i)_{i=1}^N) = \mathcal{N}\left(\boldsymbol{\mu}_{j,\boldsymbol{c}}; \boldsymbol{m}^*_{j,\boldsymbol{c}}, \frac{\sigma^2_{j,\boldsymbol{c}}}{\kappa^*_{j,\boldsymbol{c}}}\right) \tag{6}$$

where the posterior updates are:

$$\alpha_j^* = \alpha + n_j \tag{7}$$

$$\kappa_{j,\boldsymbol{l}}^* = \kappa_{j,\boldsymbol{l}} + n_j \qquad \kappa_{j,\boldsymbol{c}}^* = \kappa_{j,\boldsymbol{c}} + n_j \qquad \nu_{j,\boldsymbol{l}}^* = \nu_{j,\boldsymbol{l}} + n_j$$

$$\boldsymbol{m}_{j,\boldsymbol{l}}^* = \tfrac{\kappa_{j,\boldsymbol{l}}\boldsymbol{m}_{j,\boldsymbol{l}}+\boldsymbol{t}_{j,\boldsymbol{l}}}{\kappa_{j,\boldsymbol{l}}^*} \qquad \boldsymbol{S}_j = \boldsymbol{T}_{j,\boldsymbol{l}} - \tfrac{\boldsymbol{t}_{j,\boldsymbol{l}}}{n_j}\tfrac{\boldsymbol{t}_{j,\boldsymbol{l}}^T}{n_j}$$

$$\boldsymbol{\Lambda}_{j,\boldsymbol{l}}^* = \boldsymbol{\Lambda}_{j,\boldsymbol{l}} + \boldsymbol{S}_j + \tfrac{\kappa_{j,\boldsymbol{l}} n_j}{\kappa_{j,\boldsymbol{l}}^*}\left(\tfrac{\boldsymbol{t}_{j,\boldsymbol{l}}}{n_j}\tfrac{\boldsymbol{t}_{j,\boldsymbol{l}}^T}{n_j} - \boldsymbol{m}_{j,\boldsymbol{l}}\boldsymbol{m}_{j,\boldsymbol{l}}^T\right)$$

$$\boldsymbol{m}_{j,\boldsymbol{c}}^* = \tfrac{\kappa_{j,\boldsymbol{c}}\boldsymbol{m}_{j,\boldsymbol{c}}+\boldsymbol{t}_{j,\boldsymbol{c}}}{\kappa_{j,\boldsymbol{c}}^*}$$

$$a_{\mathrm{j},\boldsymbol{c}}^* = a_{j,\boldsymbol{c}} + \tfrac{1}{2}n_j$$

$$b_{j,\boldsymbol{c}}^* = b_{j,\boldsymbol{c}} + \tfrac{1}{2}\left(\tfrac{m_{j,\boldsymbol{c}}^2}{\kappa_{j,\boldsymbol{c}}} + \boldsymbol{T}_{j,\boldsymbol{c}} - \tfrac{\boldsymbol{t}_{j,\boldsymbol{c}}^2}{2}\right). \tag{8}$$

Finally, all these distributions above have well-known closed-form expressions for their modes (which are the ones used in the paper) where these modes depend on the posterior updates above.

3.2. Hastings Ratios

The derivation of the Hastings ratios below is based on [5]. Note that what is referred below as a sub-superpixel corresponds to a sub-cluster in their paper (which was unrelated to superpixels). First, we compute the marginal likelihood of both the color space with an NIG prior (Eq. (10)), and the location space with NIW prior (Eq. (11)). Let j_l , j_r denote the proposed sub-superpixels created after splitting superpixel j. Let j_1 , j_2 denote the adjacent superpixels, while $j_{1,2}$ denotes the superpixel obtained by merging superpixels j_1 and j_2. For $a \in j, j_l, j_r, j_1, j_2, j_{1,2}$, let $\boldsymbol{x}^a = (\boldsymbol{x}_{\boldsymbol{l}}^a, \boldsymbol{x}_{\boldsymbol{c}}^a)$ be the set of all measurements associated with the cluster of interest. Let $\Gamma(\cdot)$ denote the univariate gamma distribution, and let $\Gamma_2(\cdot)$ denote the 2D multivariate gamma distribution. The marginal likelihood [8] of $\boldsymbol{x}^a$ is

$$f(\boldsymbol{x}^a;\lambda) = f(\boldsymbol{x}_{\boldsymbol{c}}^a;\lambda) f(\boldsymbol{x}_{\boldsymbol{l}}^a;\lambda) \tag{9}$$

where

$$f(\boldsymbol{x}_{\boldsymbol{c}}^a;\lambda) = \frac{|k^*|^{\frac{1}{2}} b^a \Gamma(a^*)}{|k|^{\frac{1}{2}} {b^*}^{a^*} \Gamma(a) \pi^{\frac{n_j}{2}} 2^{n_j}}\,, \tag{10}$$

and

$$f(\boldsymbol{x}_{\boldsymbol{l}}^a;\lambda) = \frac{\Gamma_2(\frac{\nu^*}{2})|\Lambda_0|^{\nu/2}}{\pi^{n_j}\Gamma_2(\frac{\nu}{2})|\Lambda^*|^{\nu^*/2}}\left(\frac{k}{k^*}\right) \tag{11}$$

(where all the hyperparamters in these three equations are those associated with the cluster of interest).

We are now in position to compute the Hastings ratios for splits (Eq. (12) below) and merges (Eq. (13) below). During the inference iterations, these ratios are used to determine if applying a split or a merge explains the data better than the current state. To compute these ratios, we first compute, using Eq. (9), the marginal likelihoods for the superpixel which is a candidate for a split, for its subclusters, for the the two superpixels which are candidates to be merged together, and for the proposed merged superpixel. Adapting the results from [5] to the choice of our priors (NIW and NIG), these ratios are:

$$H_{\text{split}} = \frac{\alpha\Gamma(n_{j_l})f(\boldsymbol{x}^{j_l};\lambda)\Gamma(n_{j_r})f(\boldsymbol{x}^{j_r};\lambda)}{\Gamma(n_j)f(\boldsymbol{x}^j;\lambda)} \tag{12}$$

and

$$H_{\text{merge}} = \frac{\Gamma(n_{j_1}+n_{j_2})f(\boldsymbol{x}^{j_{1,2}};\lambda)\Gamma(\alpha)\Gamma(\frac{\alpha}{2}+n_{j_1})\Gamma(\frac{\alpha}{2}+n_{j_2})}{\alpha\Gamma(n_{j_1})\Gamma(n_{j_2})f(\boldsymbol{x}^{j_1};\lambda)f(\boldsymbol{x}^{j_2};\lambda)\Gamma(\alpha+n_{j_1}+n_{j_2})\Gamma(\frac{\alpha}{2})\Gamma(\frac{\alpha}{2})}\,. \tag{13}$$

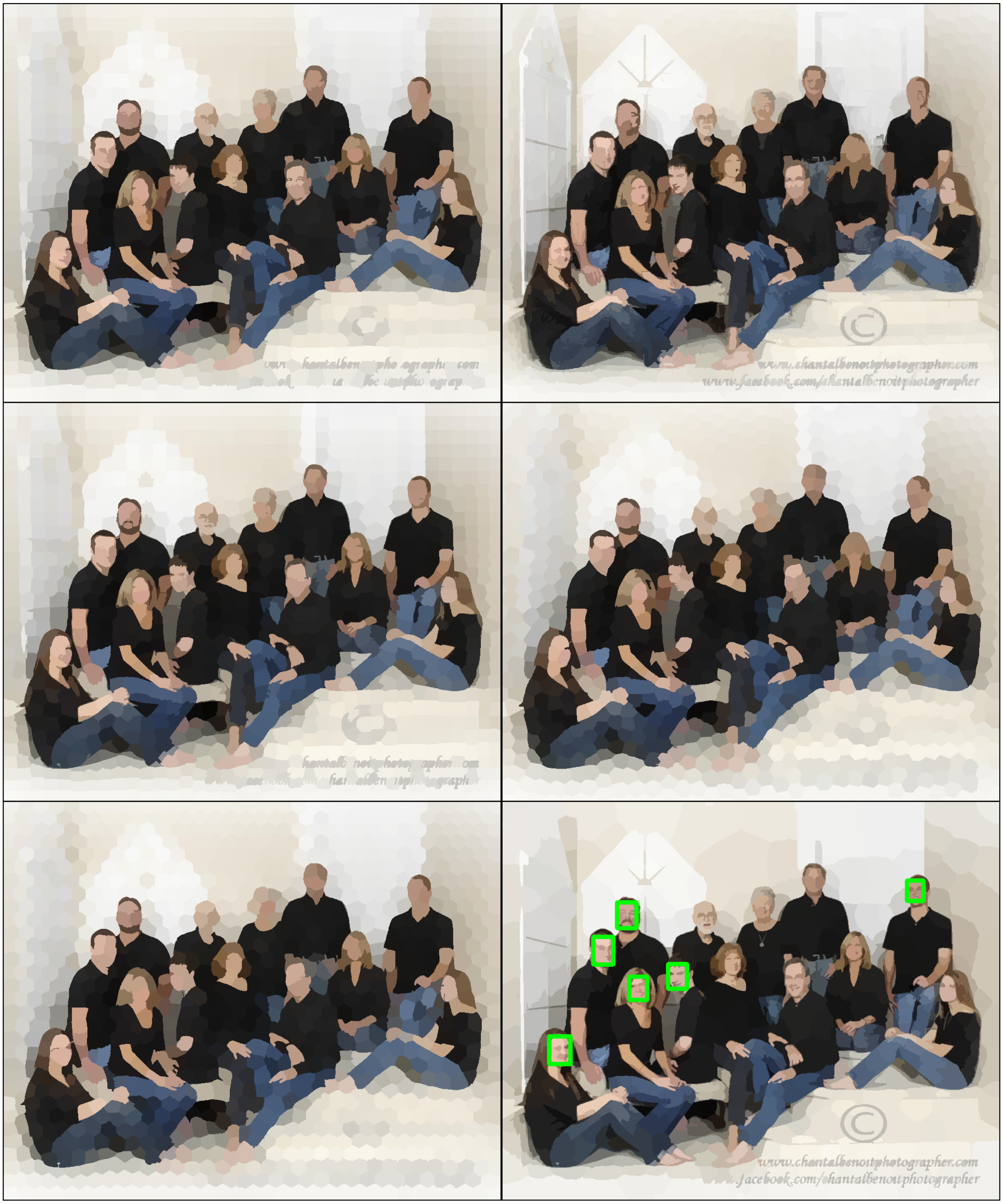

www.chantalbenoitphotographer.com
www.facebook.com/chantalbenoitphotographer
www.chantalbenoitphotographer.com
facebook.com/chantalbenoitphotographer

Figure 1: Example face-detection results using the SP mean colors. All methods were initialized, and ended with, $K \approx 1100$. Image taken from WIDER-face dataset [12]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

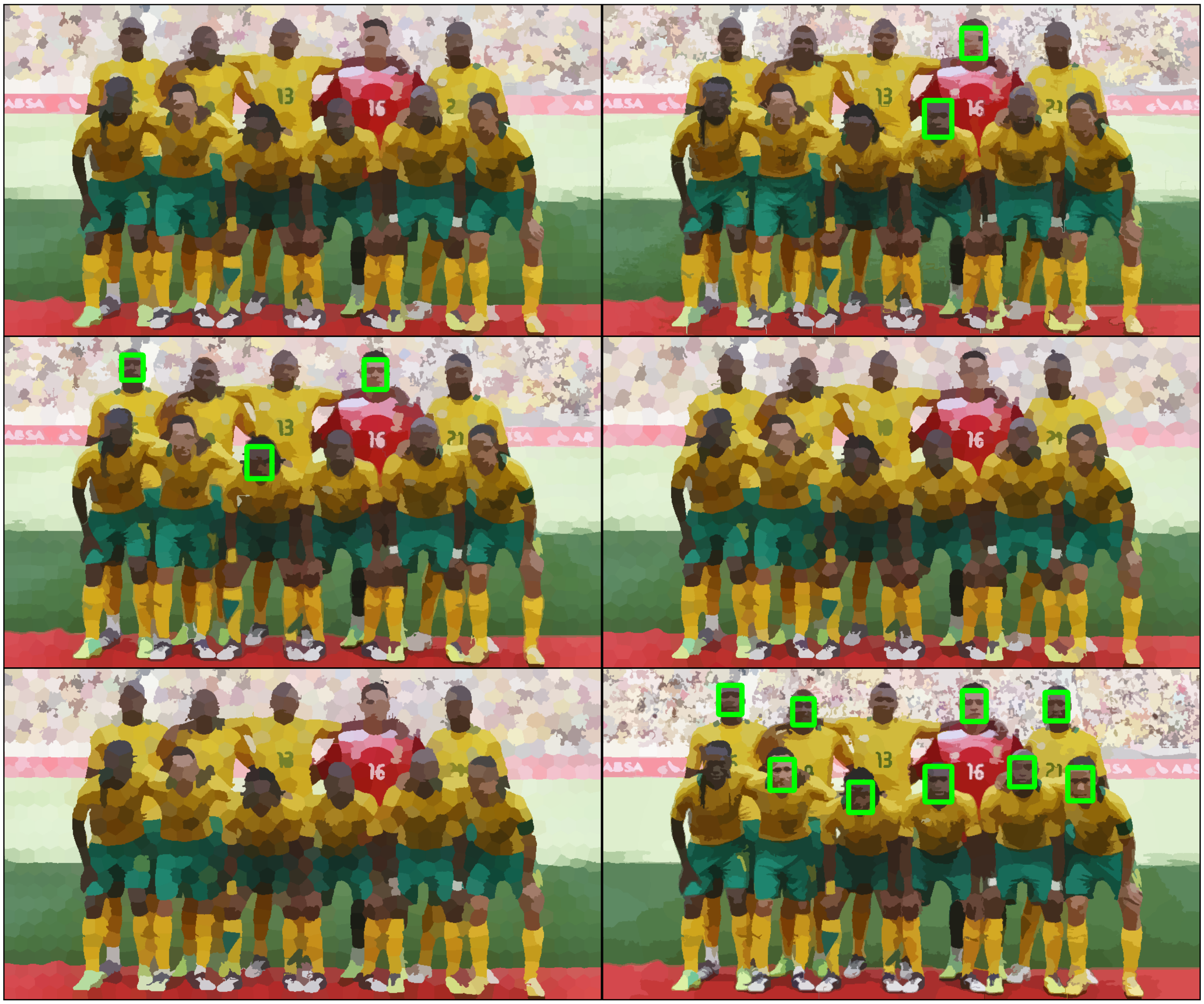


Figure 2: Example face-detection results using the SP mean colors. All methods were initialized, and ended with, $K \approx 1100$. Image taken from WIDER-face dataset [12]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

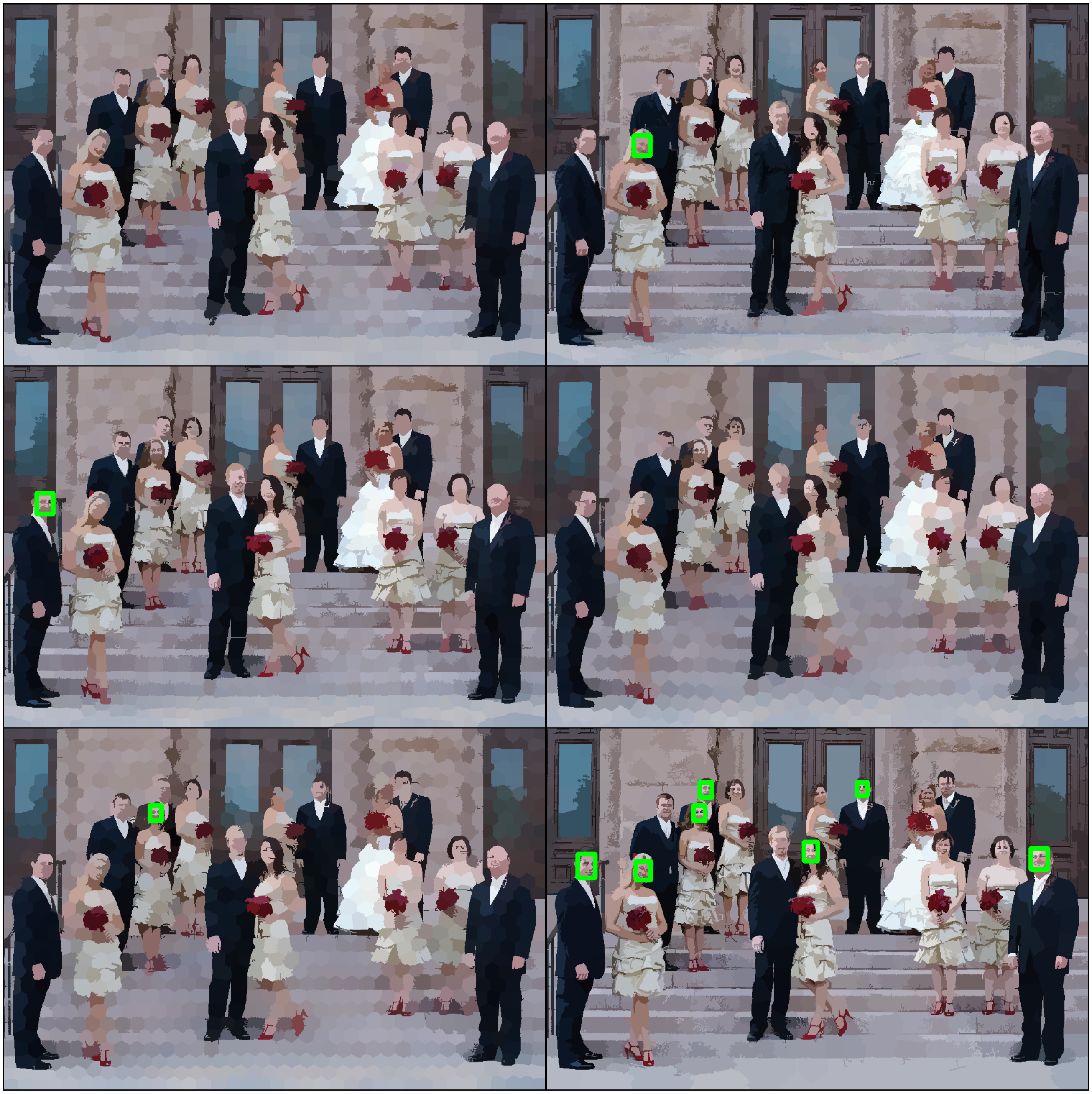


Figure 3: Example face-detection results using the SP mean colors. All methods were initialized, and ended with, $K \approx 1100$. Image taken from WIDER-face dataset [12]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

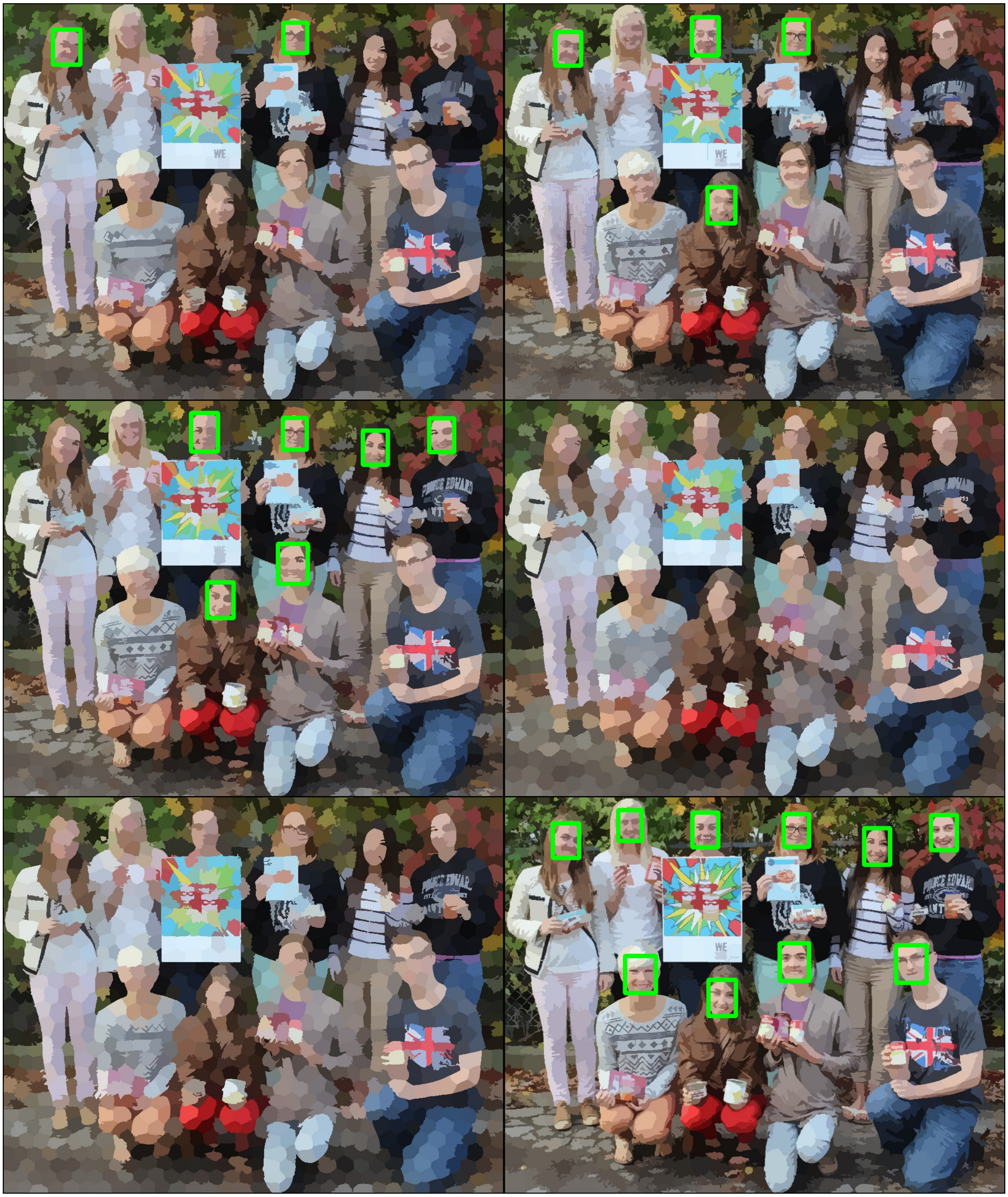


Figure 4: Example face-detection results using the SP mean colors. All methods were initialized, and ended with, $K \approx 1100$. Image taken from WIDER-face dataset [12]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

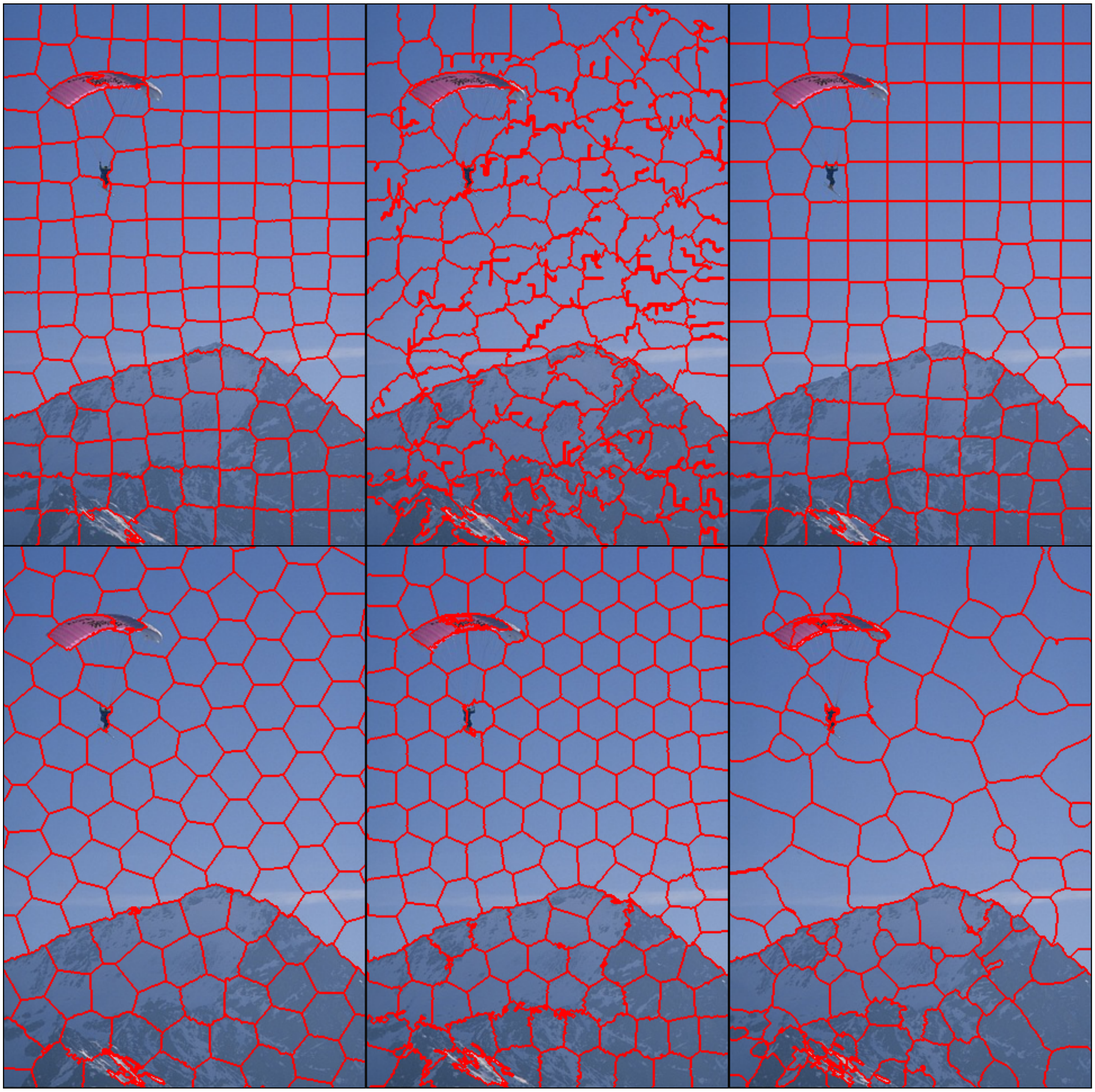


Figure 5: A visual comparison of SP boundaries overlaid over original images. All methods were initialized, and ended with, $K = 150$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

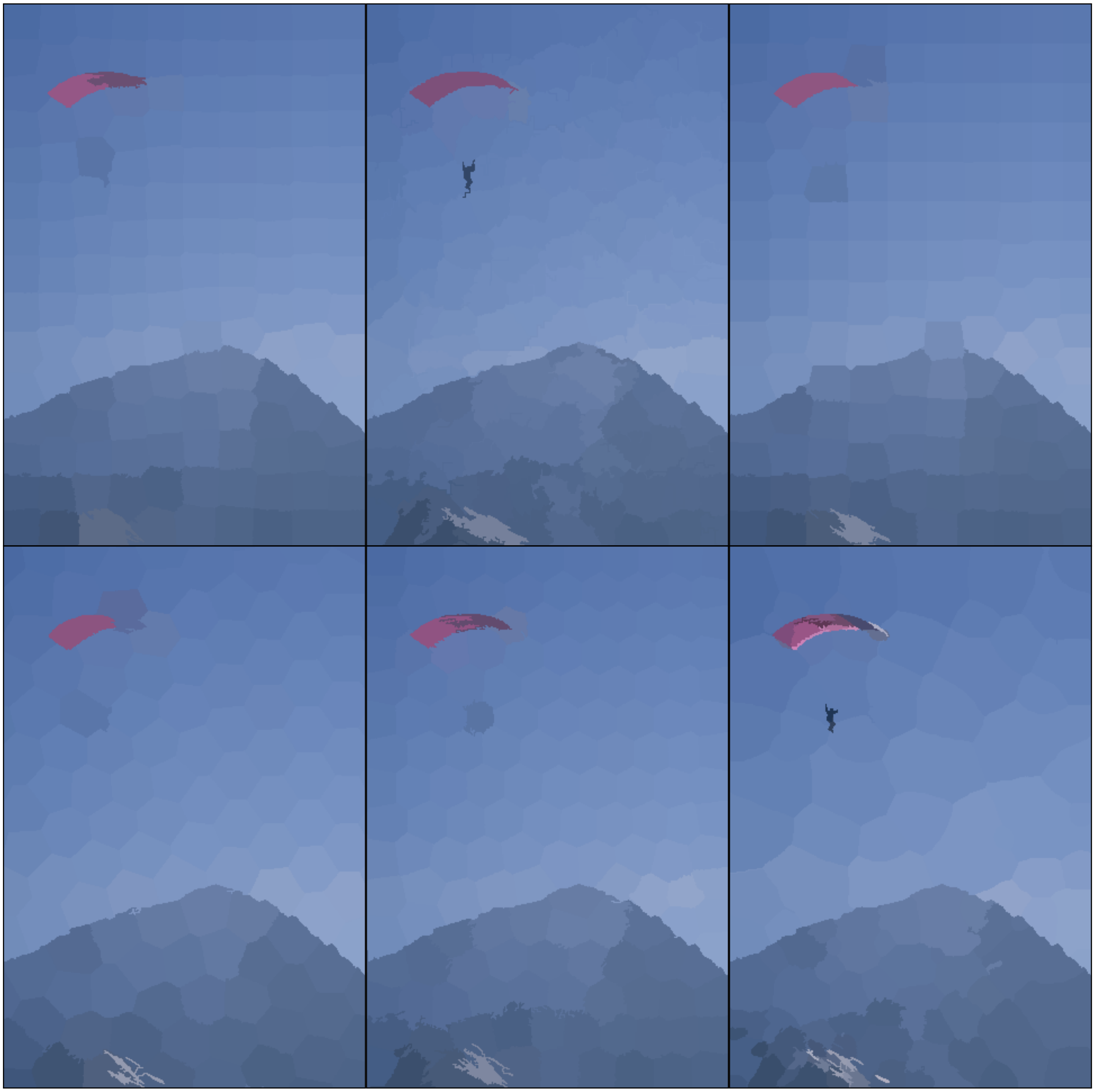


Figure 6: A visual comparison of SP means. All methods were initialized, and ended with, $K = 150$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

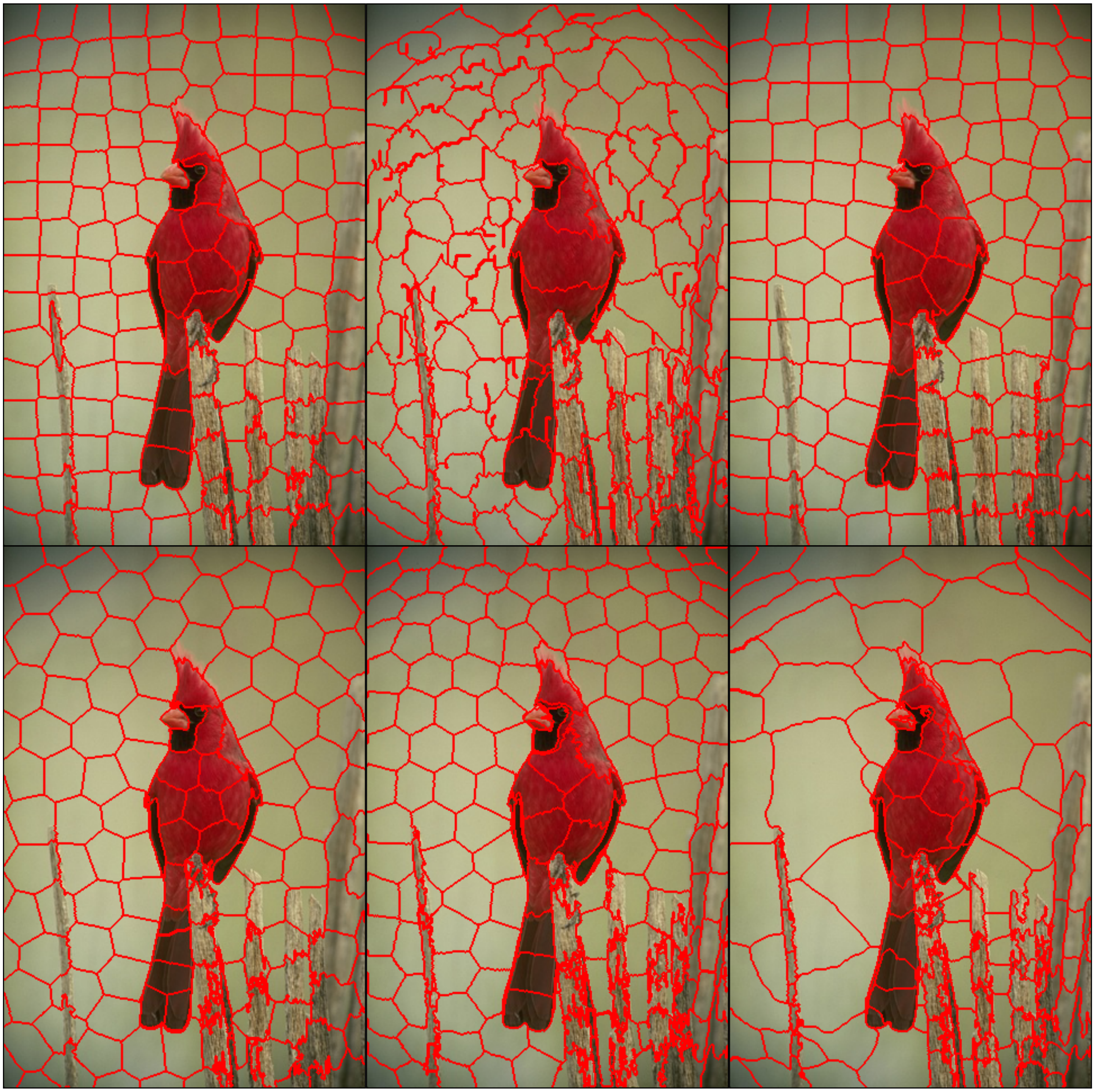


Figure 7: A visual comparison of SP boundaries overlaid over original images. All methods were initialized, and ended with, $K = 150$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

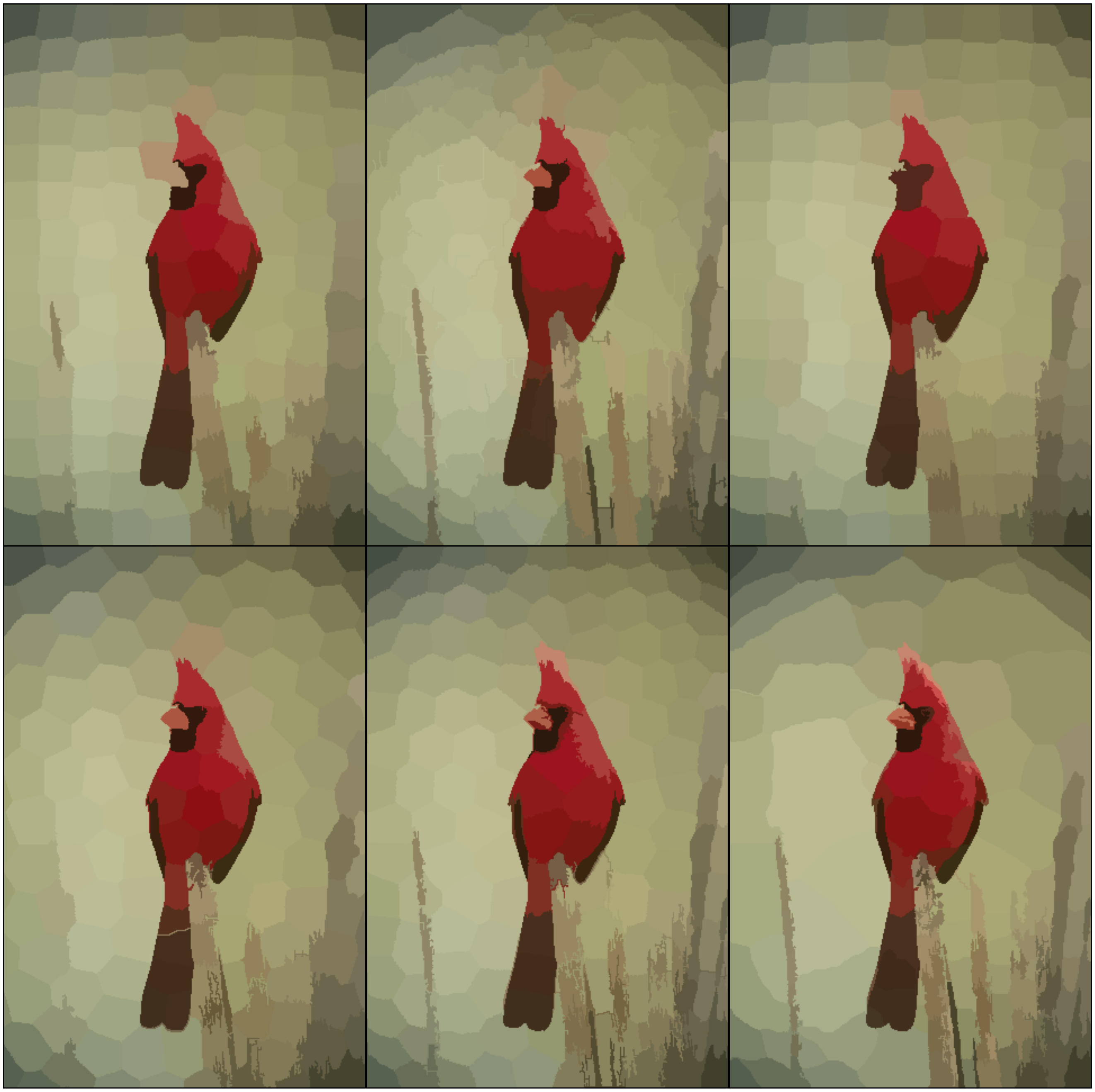


Figure 8: A visual comparison of SP means. All methods were initialized, and ended with, $K = 150$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

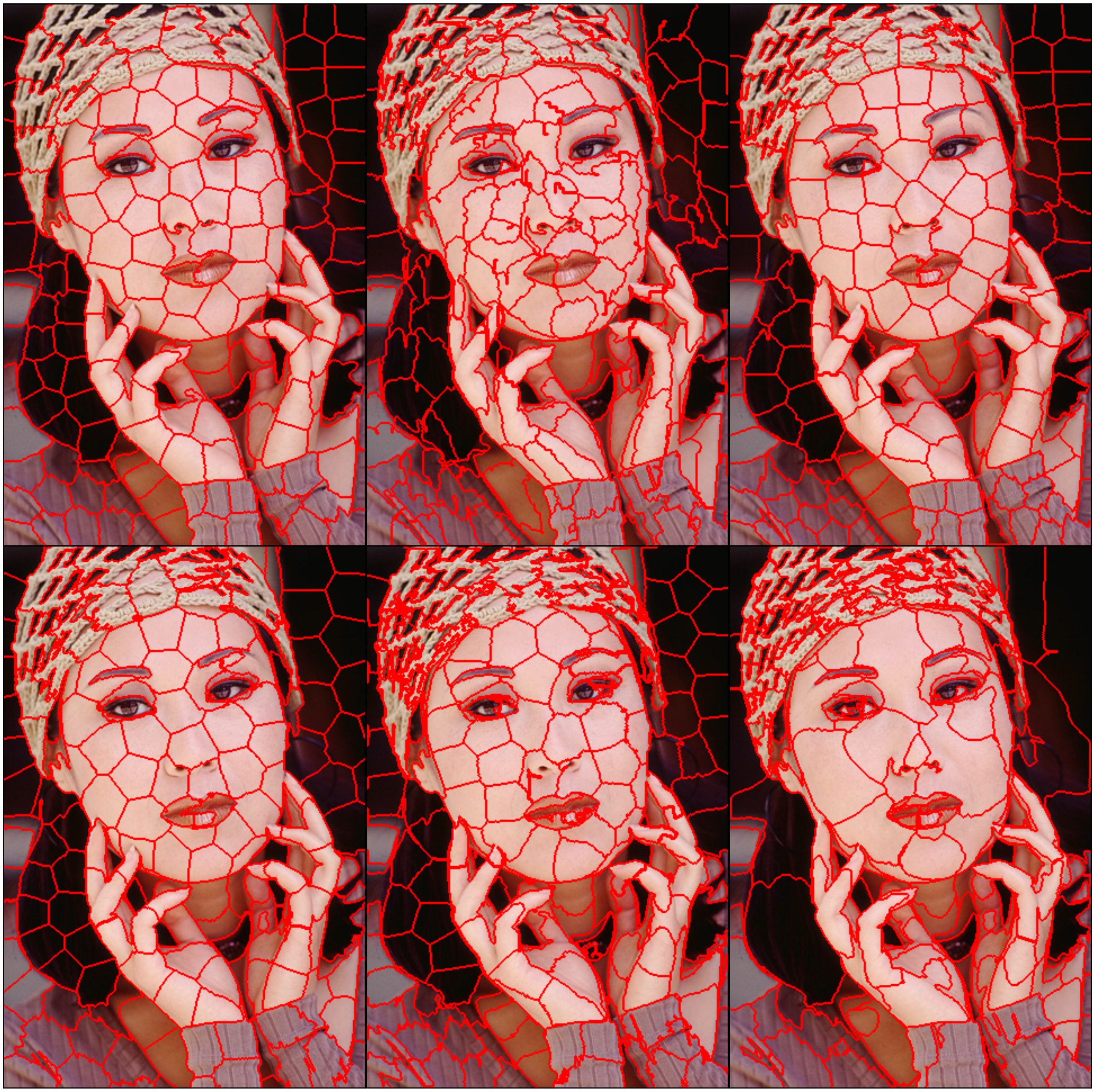


Figure 9: A visual comparison of SP boundaries overlaid over original images. All methods were initialized, and ended with, $K = 180$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

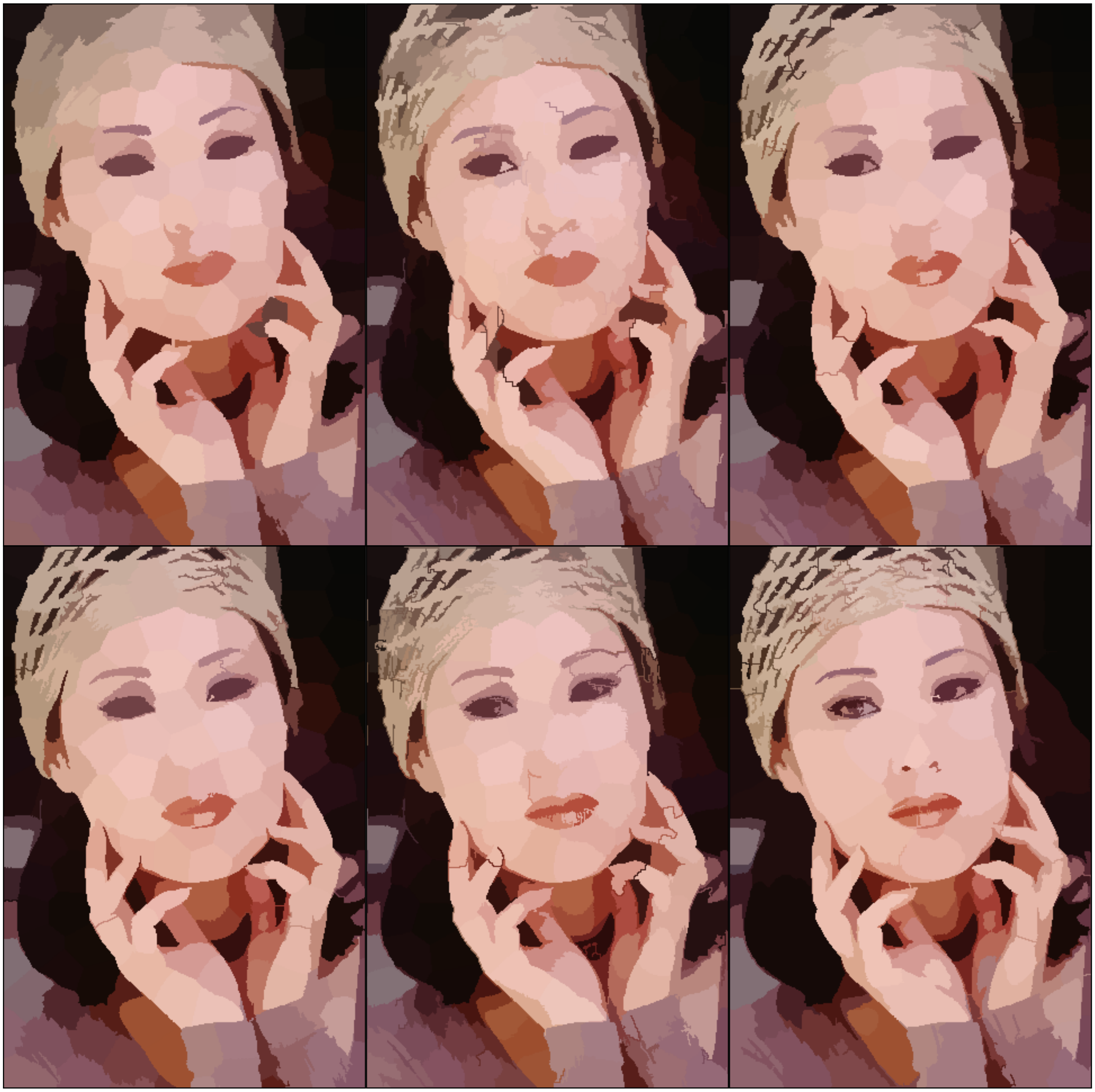


Figure 10: A visual comparison of SP means. All methods were initialized, and ended with, $K = 180$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11], ETPS [13]; second row: TSP [6], FSCSP [7], BASS.

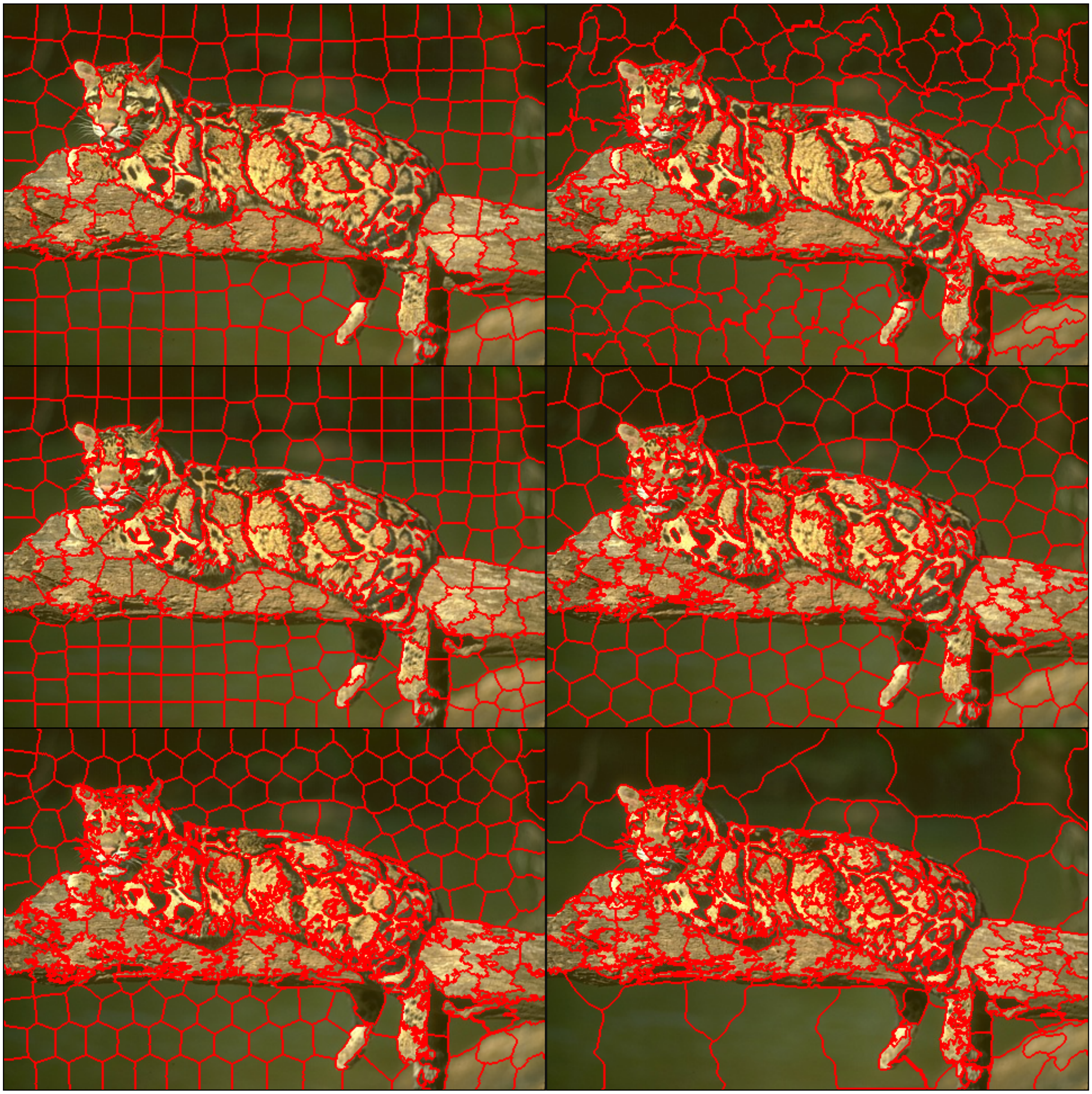


Figure 11: A visual comparison of SP boundaries overlaid over original images. All methods were initialized, and ended with, $K = 250$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

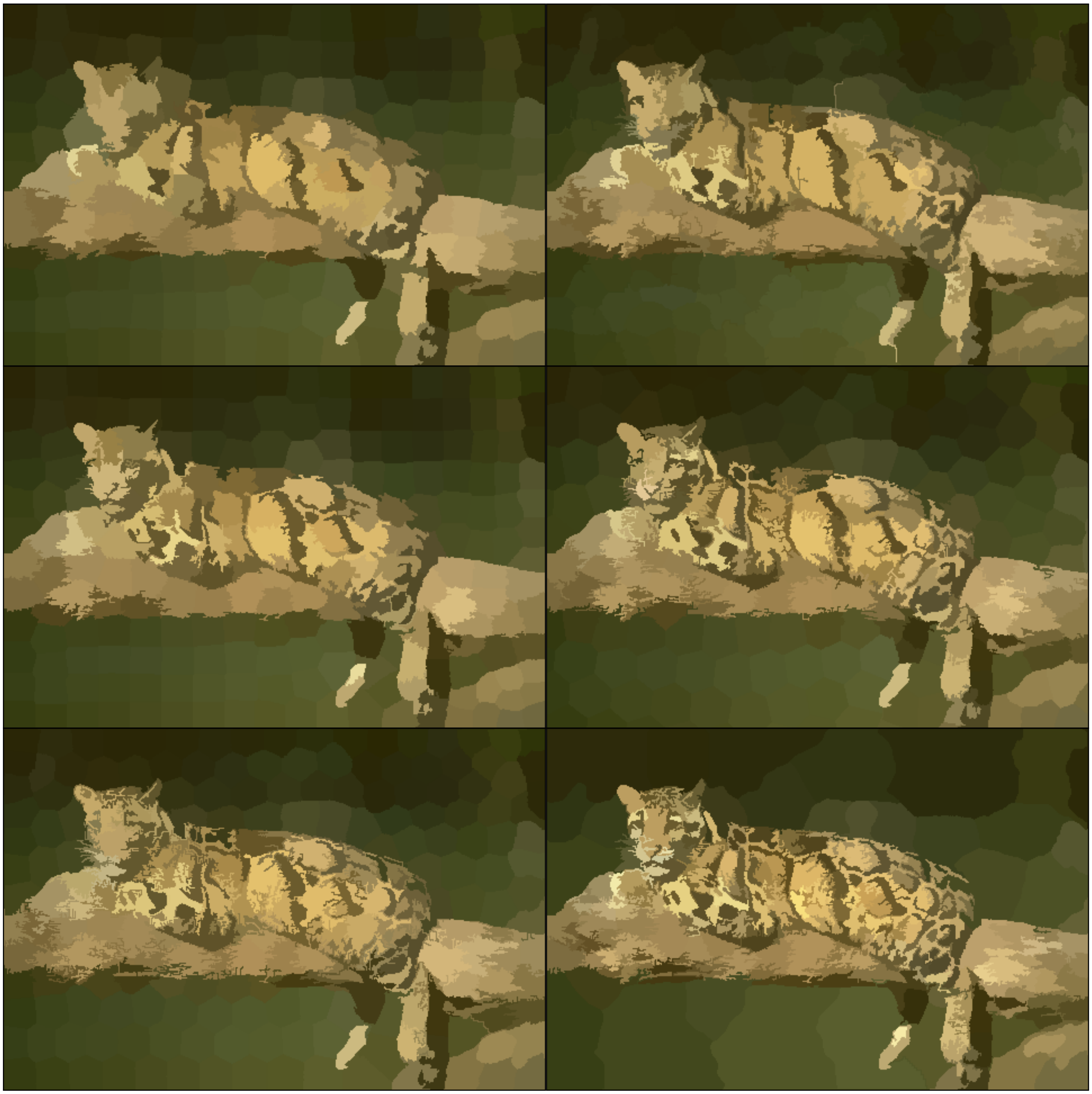


Figure 12: A visual comparison of SP means. All methods were initialized, and ended with, $K = 250$. Image taken from BSDS500 dataset [2]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

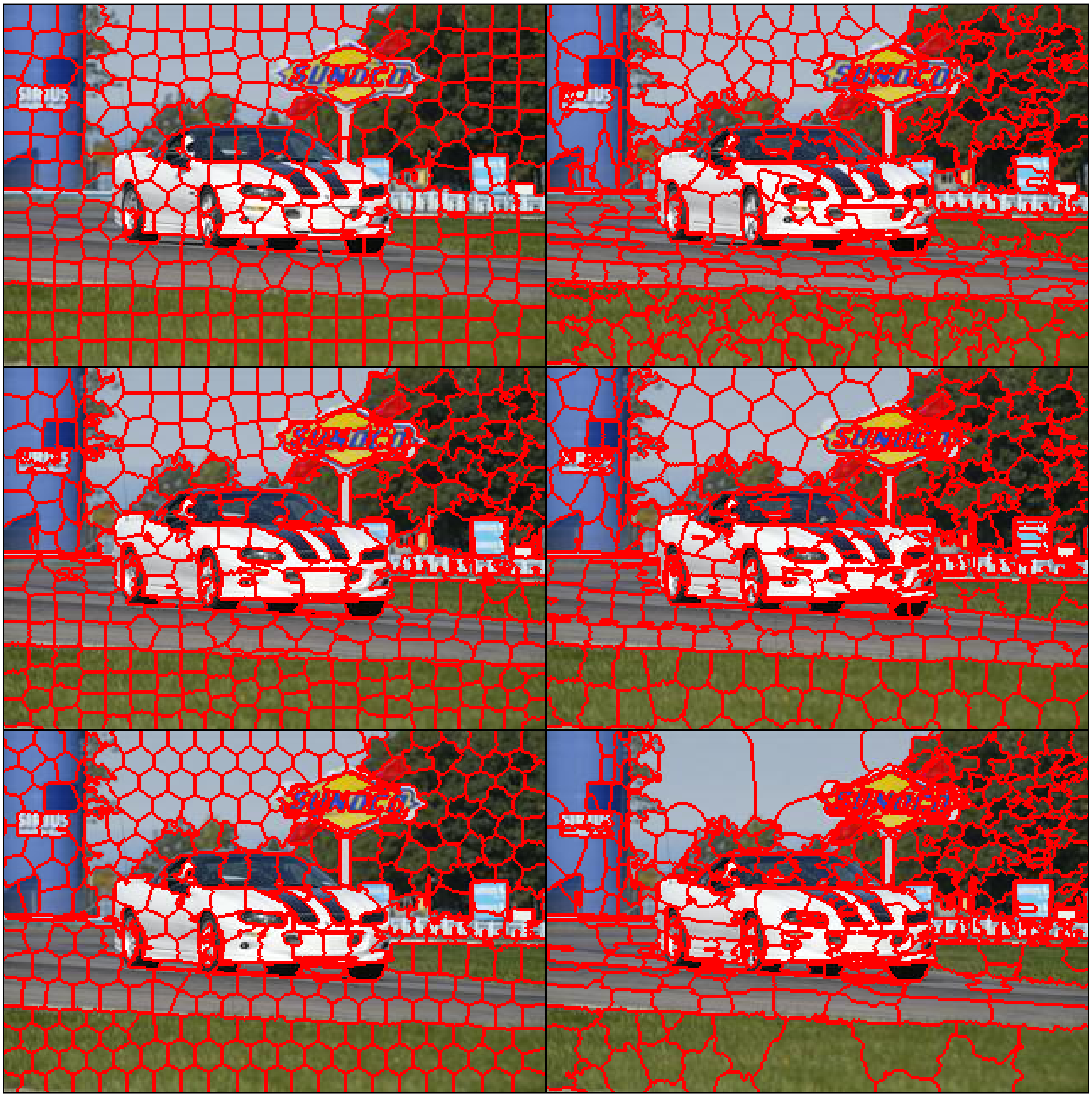


Figure 13: A visual comparison of SP boundaries overlaid over original images. All methods were initialized, and ended with, $K = 250$. Image taken from SBD dataset [9]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.

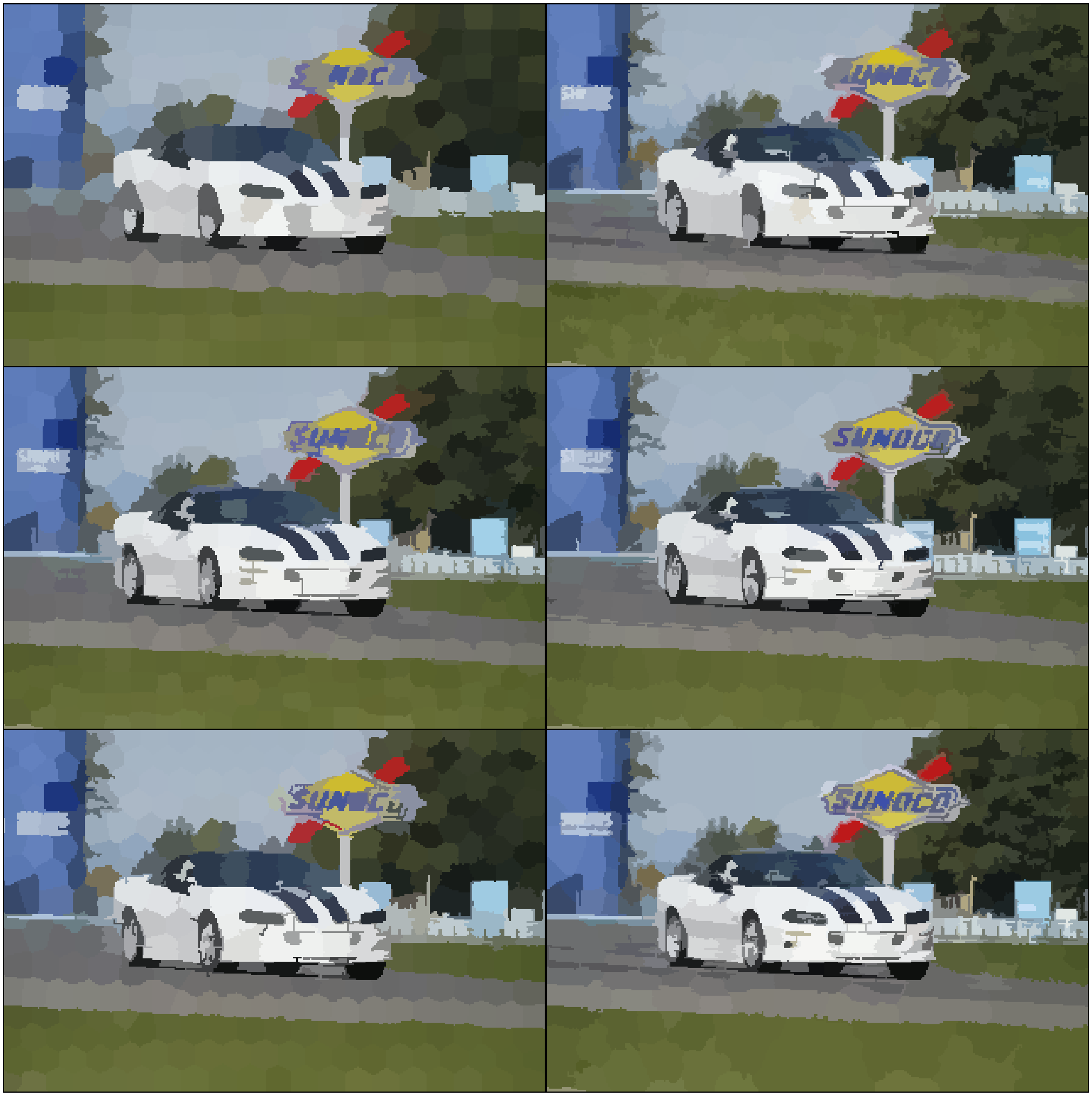
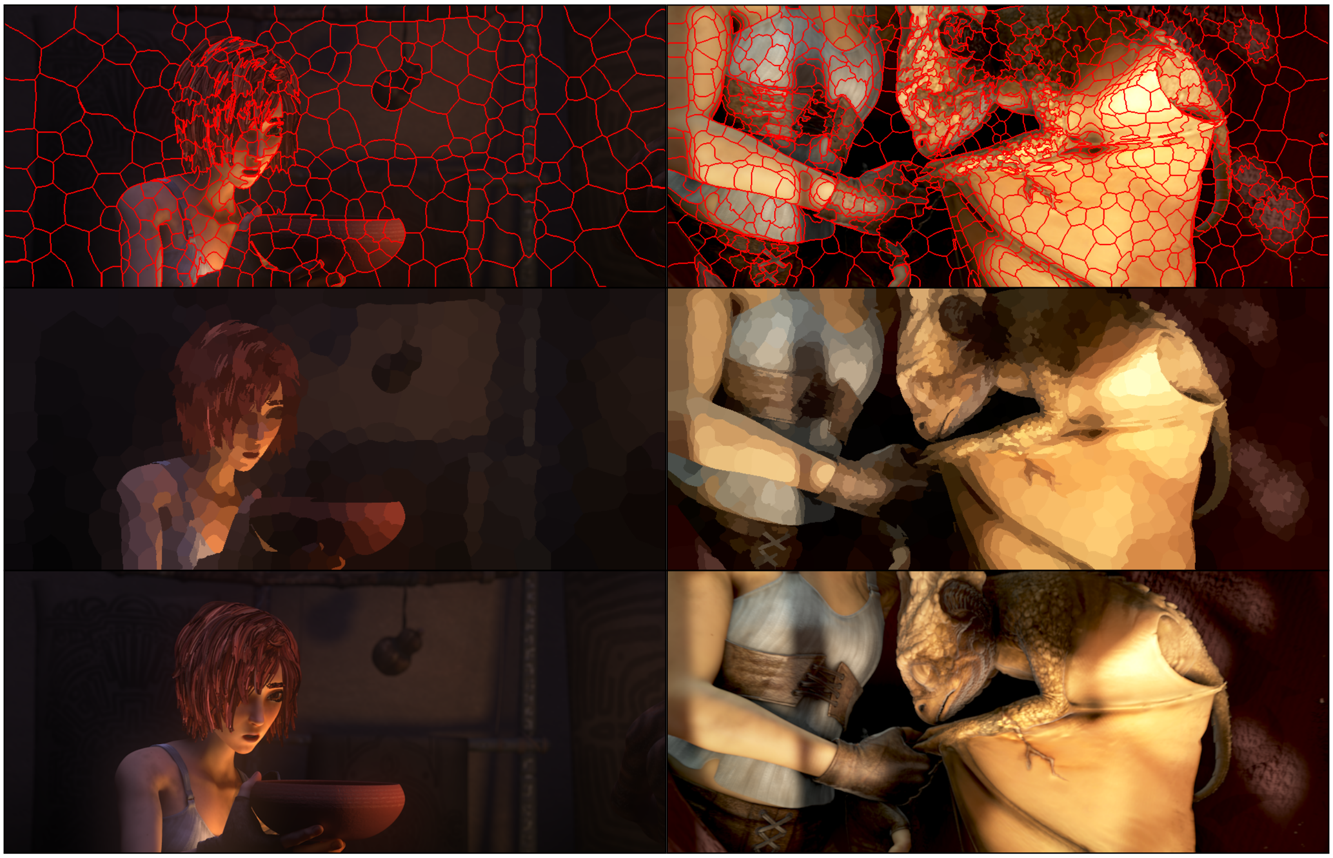


Figure 14: A visual comparison of SP means. All methods were initialized, and ended with, $K = 250$. Image taken from SBD dataset [9]. From left to right, first row: SLIC [1], re-SEEDS [11]; second row: ETPS [13], TSP [6]; third row: FSCSP [7], BASS.



$K_{\text{final}} = 345SP$

$K_{\text{final}} = 908SP$

Figure 15: Adapting K to the image content: BASS, when applied to both images with $K_0 = 550$ and the same hyperparamaters, converged to a different K_{final} in each image. Top to bottom: superpixel boundaries; superpixels colored by their mean colors; original images (taken from [4]).

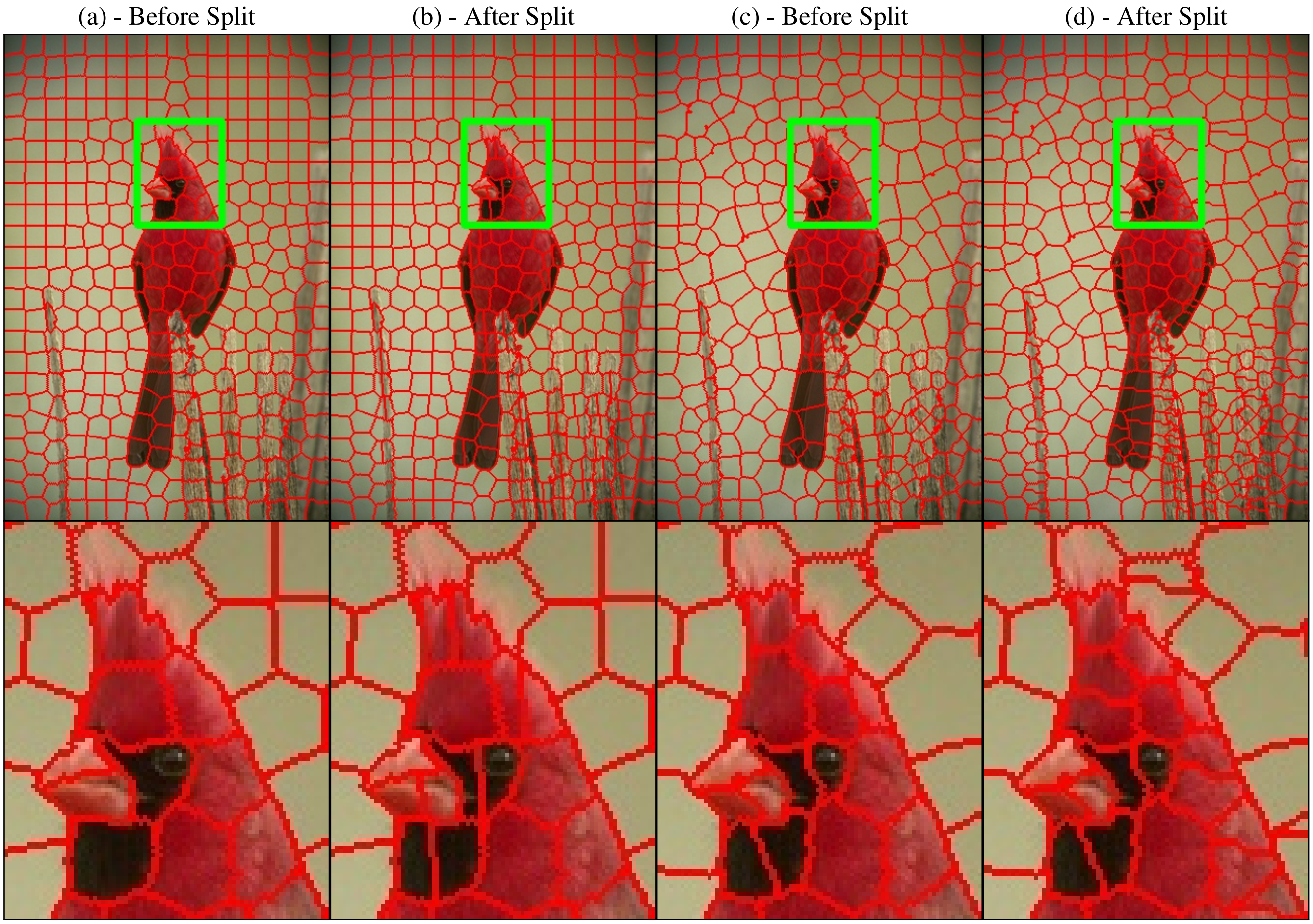


Figure 16: Visualization of the split step. (b), (d) illustrate the superpixels one iteration after (a), (c), respectively. Between (b) and (c) there are few iterations and a few merges. In the figure, there are both vertical and horizontal splits in the foreground of the image. The process described in the figure repeats itself till convergence (not shown here).

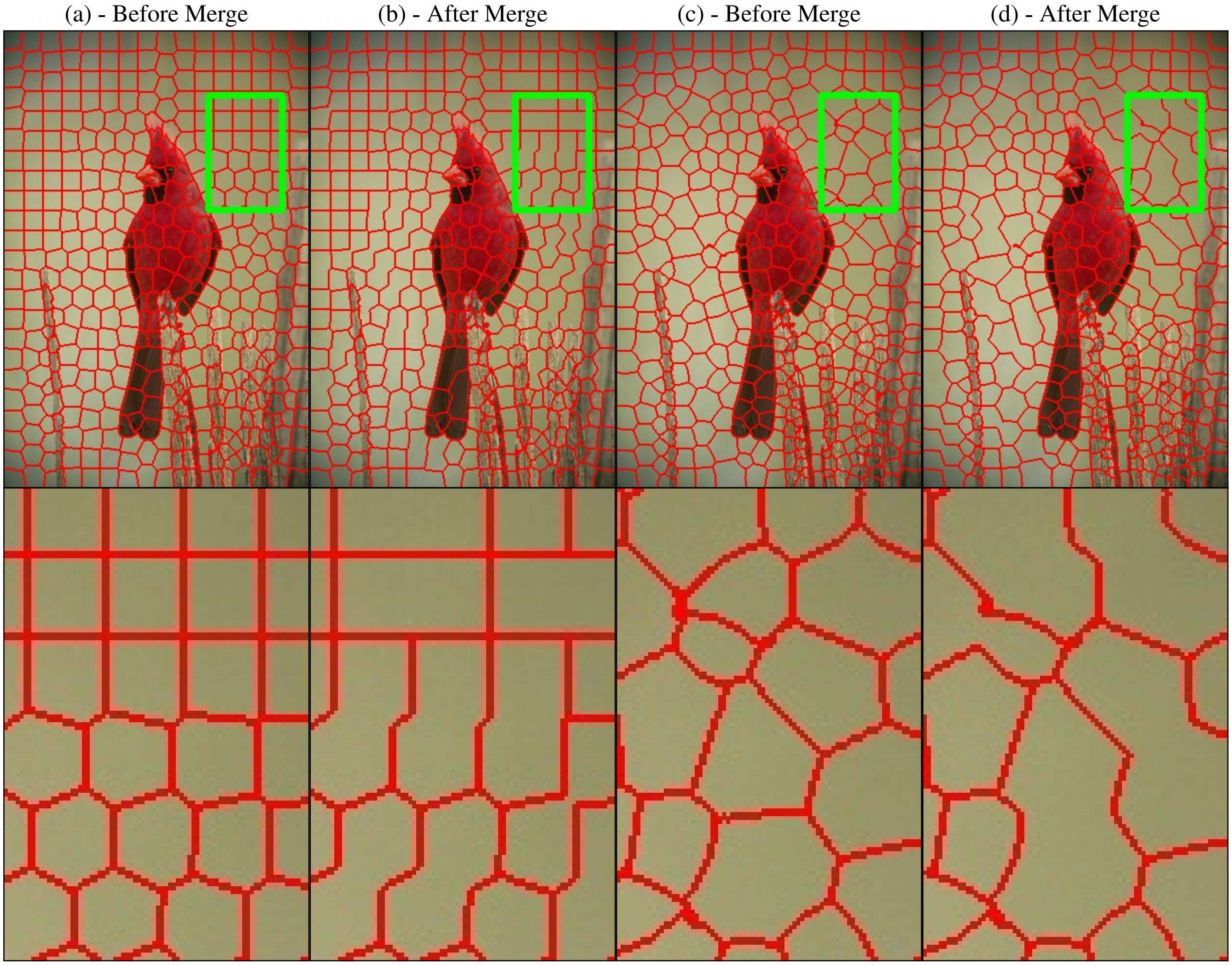


Figure 17: Visualization of the merge step. (b), (d) illustrate the superpixels one iteration after (a), (c), respectively. Between (b) and (c) there are few iterations and a few splits. The process described in the figure repeats itself till convergence (not shown here).

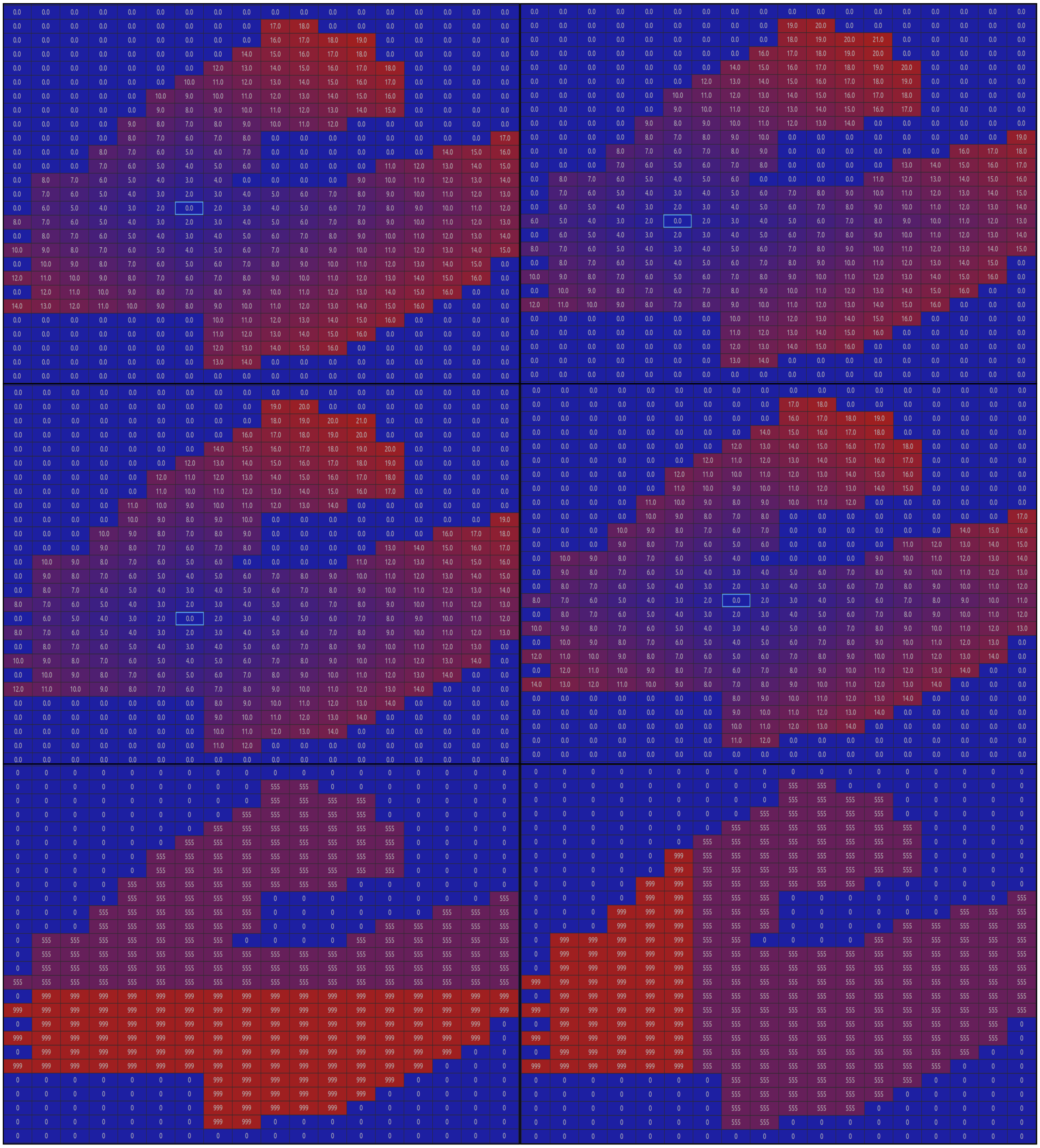
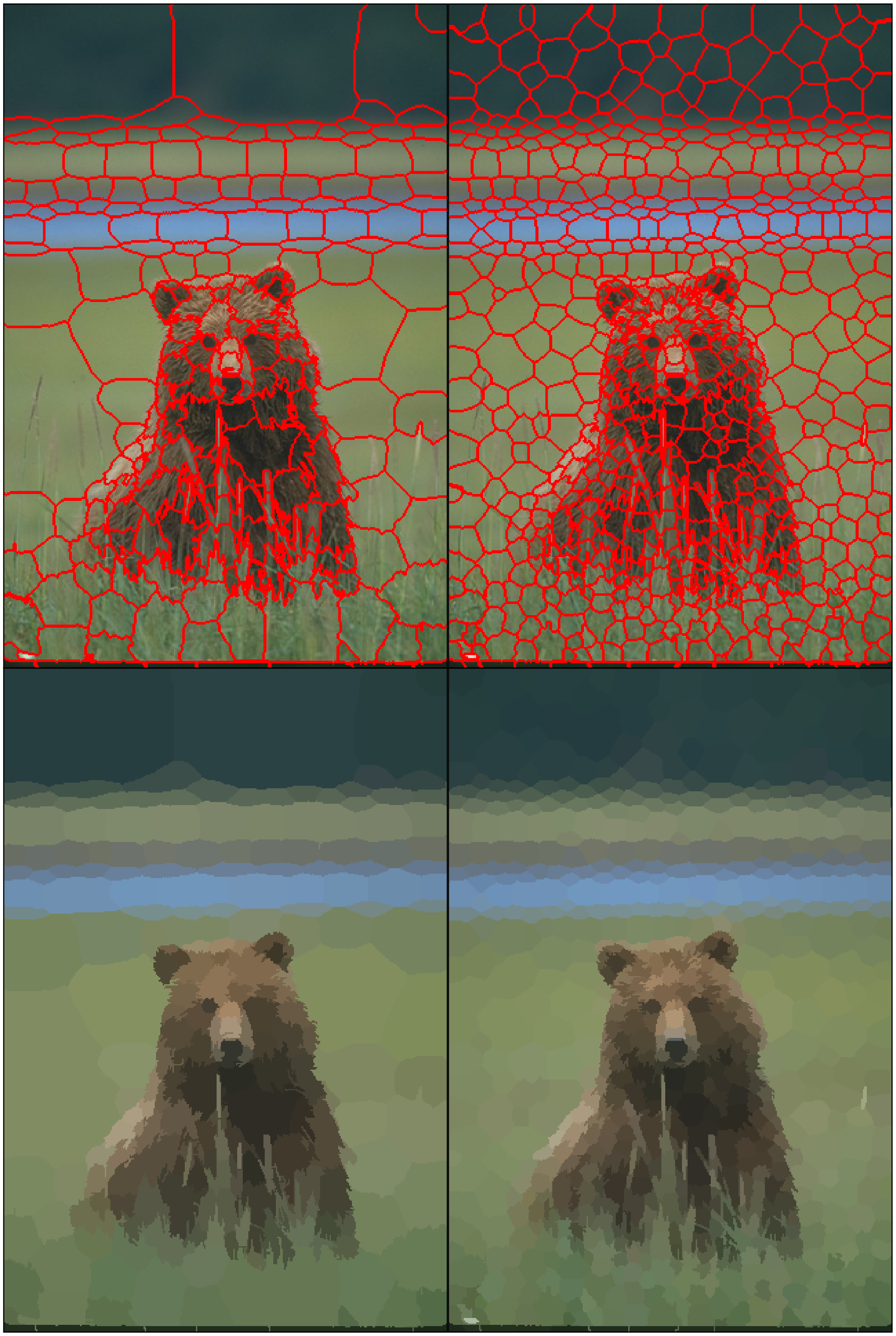


Figure 18: An example of the splitting process. We demonstrate the split of the same superpixel presented in red, which has an irregular shape, making it hard to split while maintaining connectivity, once in the horizontal case (left column) and once in the vertical one (right column). The first and second rows represent the distance of each pixel from the centers c_1^1, c_1^2 respectively, as a heat map; the colder the color the closer the pixel to the center. Each pixel is associated with its new sub-superpixel by taking the minimal distance from c_1^1 and c_1^2 . The last row illustrates the superpixel after the split. By alternating between horizontally and vertically BASS gained the flexibility in superpixels' shape that enables it to adhere to the boundaries of small, complex objects.



$K_{\text{final}} = 250SP$ $K_{\text{final}} = 600SP$

Figure 19: An example demonstrating the effect of the hyper-parameter α used in (12), (13) on the numbers of splits and merges. Increasing α encourages more splits and fewer merges, thus directly changes the final number of superpixels. Started with $K_0 = 550$, converged to a different K_{final} in each image. Top row: superpixel boundaries; Bottom row: superpixels colored by their mean colors. Original images (taken from [2]).

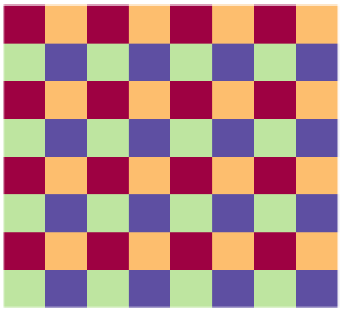
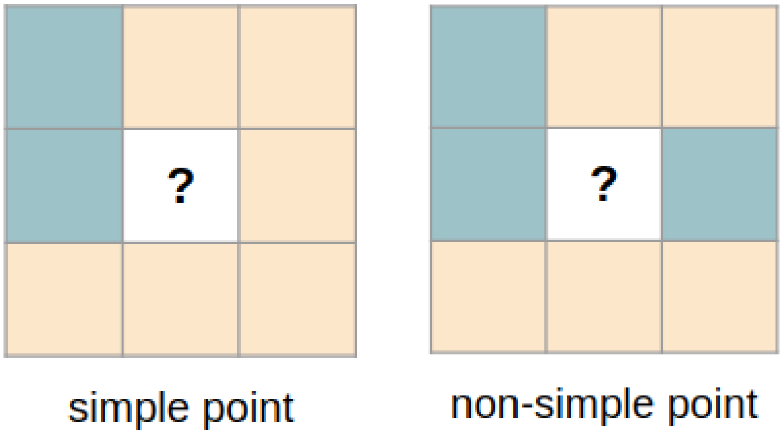


Figure 20: A visualization of the parallelization. The image is partitioned into 4 different sets such that each 2×2 block includes one representative from each set (as is indicated by the colors). In each set, the labels (not showed here) are conditionally independent of each other.



(a)

(b)

Figure 21: A simple-point test for the binary case. (a) The central pixel is a simple point [6, 3]: regardless what its label is, the number of connected components in either color is unchanged (1 green, 1 orange). (b) The central pixel is a non-simple point: its label affects, *e.g.*, the number of the yellow connected components.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2011. 8, 9, 10, 11, 12, 13, 14, 15, 22

[3] Giles Bertrand. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Pattern recognition letters*, 15(10):1003–1011, 1994. 2, 23

[4] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 18

[5] Jason Chang and John W Fisher. Parallel sampling of DP mixture models using sub-cluster splits. In *NIPS*, 2013. 3

[6] Jason Chang, Donglai Wei, and John W Fisher. A video representation using temporal superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2051–2058, 2013. 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 23

[7] Oren Freifeld, Yixin Li, and John W Fisher. A fast method for inferring high-quality simply-connected superpixels. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 2184–2188. IEEE, 2015. 2, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

[8] Andrew Gelman, Hal S Stern, John B Carlin, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013. 2, 3

[9] Stephen Gould, Richard Fulton, and Daphne Koller. Decomposing a scene into geometric and semantically consistent regions. In *2009 IEEE 12th international conference on computer vision*, pages 1–8. IEEE, 2009. 16, 17

[10] Samuel David Silvey. *Statistical inference*. Routledge, 1970. 2

[11] David Stutz, Alexander Hermans, and Bastian Leibe. Superpixel segmentation using depth information. *RWTH Aachen University, Aachen, Germany*, 4, 2014. 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

[12] Shuo Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 4, 5, 6, 7

[13] Jian Yao, Marko Boben, Sanja Fidler, and Raquel Urtasun. Real-time coarse-to-fine topologically preserving segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2947–2955, 2015. 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

[14] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016. 1