USEAQ: Ultra-fast Superpixel Extraction via Adaptive Sampling from Quantized Regions

Chun-Rong Huang, *Member, IEEE*, Wei-Cheng Wang, Wei-An Wang, Szu-Yu Lin, and Yen-Yu Lin, *Member, IEEE*

***Abstract*—We present a novel and highly efficient superpixel extraction method called USEAQ to generate regular and compact superpixels in an image. To reduce the computational cost of iterative optimization procedures adopted in most recent approaches, the proposed USEAQ for superpixel generation works in a one-pass fashion. It firstly performs joint spatial and color quantizations and groups pixels into regions. It then takes into account the variations between regions, and adaptively samples one or a few superpixel candidates for each region. It finally employs *maximum a posteriori* (MAP) estimation to assign pixels to the most spatially consistent and perceptually similar superpixels. It turns out that the proposed USEAQ is quite efficient, and the extracted superpixels can precisely adhere to boundaries of objects. Experimental results show that USEAQ achieves better or equivalent performance compared to the state-of-the-art superpixel extraction approaches in terms of boundary recall, undersegmentation error, achievable segmentation accuracy, the average miss rate, average undersegmentation error, and average unexplained variation, and it is significantly faster than these approaches. The source code of USEAQ is available at https://github.com/nchucvml/USEAQ.**

***Index Terms*—Superpixel extraction, image segmentation, joint spatial and color quantizations.**

I. Introduction

SUPERPIXEL extraction aims to group spatially connected and perceptually consistent pixels into small regions. Extracted superpixels are expected to adhere to object boundaries and be semantically meaningful. They not only provide a compact image representation but also serve as an effective domain for image feature computation. Hence, superpixels speedup and facilitate many successive applications such as surface reconstruction [1], video object segmentation [2], [3], tracking [4], [5], saliency map detection [6]–[8], image segmentation [9]–[11], and object recognition [12]–[15]. Being essential to these applications, superpixel generation has become an inherent part in various computer vision and image processing applications.

As indicated in [16], three properties are desirable for superpixel extraction. First, a superpixel needs to be composed of similar pixels, and adheres to image boundaries adequately. Second, as a preprocessing step for reducing the complexity of many applications, superpixel generation is required to be computationally efficient. Third, the generated superpixels should both increase the speed and improve the quality of segmentation results. We are aware of the increasingly growing image resolutions. An effective and efficient superpixel extraction method is always in demand. Increasing the number of superpixels in an image typically helps represent the object boundaries more precisely. However, it also significantly increases the computation time of superpixel extraction, which limits the practical usage of superpixels in high-resolution images and videos. Most existing approaches work on the trade-off between the efficiency and the precise adherence to boundaries. In this work, we present an approach whose running time is almost independent of the number of extracted superpixels.

To fulfill the aforementioned requirements, we propose a novel superpixel extraction approach, named *Ultra-fast Superpixel Extraction via Adaptive sampling from Quantized regions* (USEAQ), to efficiently decompose an image into semantic regions. Specifically, we apply the spatial and color quantizations simultaneously to decompose an image. The former retrieves the grid based on the positions of pixels, and preserves the spatial relationships between pixels and initial regions. The latter divides pixels into groups based on their colors. In [17], our preliminary approach considers pixels belonging to the same group in both spatial and color quantizations as a superpixel candidate in initialization. It neglects the variations among image regions such as homogeneous regions versus cluttered regions. In general, more superpixels of smaller sizes are required in cluttered regions to adhere to complex boundaries, while fewer superpixels of larger sizes are preferable for homogeneous regions to have a compact representation. The proposed approach takes this observation into account, and employs an adaptive sampling mechanism that picks one or more superpixel candidates from each spatially quantized region according to the color variations of that region.

In addition to the adaptive sampling mechanism, the other major characteristic of our approach is that it works in a one-pass manner. Most conventional methods such as [16] implement an iterative optimization procedure where the representative colors of superpixels and the pixel-superpixel reassignment are alternatively updated. On the contrary, the subregions adaptively determined by our approach represent spatially connected and visually coherent groups of pixels. It turns out high-quality superpixels can be extracted by simply merging these sub-regions and performing neighborhood refinement via *maximum a posteriori* (MAP) estimation. Consequently, our approach significantly speeds up superpixel extraction by

C.-R. Huang, W.-A. Wang, and S.-Y. Lin are with the Department of Computer Science and Engineering, National Chung Hsing University, Taichung 402, Taiwan (e-mail: crhuang@nchu.edu.tw).

W.-C. Wang and Y.-Y. Lin are with the Research Center for Information Technology Innovation, Academia Sinica, Taipei 115, Taiwan (e-mail: wang.wei.cheng.tj@gmail.com, yylin@citi.sinica.edu.tw).

avoiding an iterative optimization process, and can generate high-quality superpixels with regular and compact shapes.

In the experiments, we evaluate the performance of the proposed approach on the *Berkeley segmentation benchmark* [18] and the *Stanford background dataset* [19]. Compared to the state-of-the-art approaches, our approach not only achieves better boundary recalls but also is much more computationally efficient. To the best our knowledge, our method is faster than existing methods and provides the flexibility of generating regular superpixels with different numbers of superpixels. The main contributions of this work are threefold. First, a mechanism of adaptive sampling from spatially and visually quantized regions is proposed to efficiently generate initial superpixels. Second, the MAP-based estimation is designed and applied to reassign pixels to visually similar superpixels and merge small superpixels in a one-pass manner. Third, the proposed method achieves superior performance on the Berkeley segmentation benchmark and the Stanford background dataset in both accuracy and efficiency.

The rest of the paper is organized as follows. We review the relevant methods in Section II. Our approach is presented in Section III. The experimental results including the parameter adjustment and the comparisons with the state-of-the-art approaches are shown in Section IV. We show that the extracted superpixels help improve the quality of image segmentation and supervoxel construction in Section V. Finally, we make a brief conclusion in Section VI.

II. Related Work

The literature on superpixel extraction is quite extensive. Most methods for superpixel extraction can be roughly divided into two categories, i.e., graph-based and gradient-based methods. We review some representative methods of each category.

A. Graph-Based Superpixel Extraction

Graph-based methods construct superpixels by employing a graph to model the relationships between neighboring pixels. As shown in a pioneering work *normalized cuts* (NC) [20], pixels are represented as nodes with their links to the neighbors as edges in the graph. Superpixels are obtained by recursively minimizing a cost function defined on the graph. Guiding model search was introduced in [21] to reduce the computational cost of normalized cuts. Felzenszwalb and Huttenlocher (FH) [22] presented a graph-based segmentation approach, in which agglomerative clustering is applied so that each node in the graph forms a minimum spanning tree. Their method shows its advantage over normalized cuts in efficiency, but it often leads to superpixels of less regular shapes and sizes.

Moore et al. [23] proposed superpixel lattices (SL) to generate superpixels by preserving the topology of a regular lattice. They optimized both vertical and horizontal paths by referring to the boundary cost map, and used the optimized paths to split an image and yield superpixels. To further enhance the results of [23], Moore et al. [24] imposed a punitive cost on the boundary orthogonal to the current cut during the iterative process of graph partition.

Liu et al. [25] presented an approach to superpixel segmentation, in which the entropy rate (ERS) and a balancing function jointly constrain the compactness and sizes of each cluster, and a greedy algorithm is adopted to complete the segmentation. Their method is computationally more efficient than normalized cuts [20]. Veksler et al. [26] over-segmented an image by covering it with overlapping square patches of a fixed size. They developed an energy function based on image gradient to guide the assignment from pixels to superpixels by using *graph-cuts* [20]. Zhang et al. [27] introduced two pseudo-Boolean functions in which segmentation is modeled as a binary labeling problem. The adopted non-iterative pseudo Boolean optimization makes their method more efficient than that in [26]. Peng et al. [28] used a framework with higher order energy optimization to carry out superpixel construction. k-means clustering is used to generate the initial superpixels, which help accelerate the optimization of the higher order energy function for refining the initial superpixels. Despite the effectiveness and the significant progress on efficiency, graph-based approaches to superpixel extraction are not able to support real-time performance. The proposed method achieves the comparable performance to the state-of-the-art graph-based methods, and supports real-time superpixel extraction.

B. Gradient-Based Superpixel Extraction

Methods of this category cover those using either gradient ascent or gradient descent for superpixel extraction. Unlike graph-based methods, the gradient-based methods initially partition an image into multiple regions as the reference, and gradually refine the region boundaries to yield superpixels. The process of refinement is carried out by considering diverse image properties so that each of the resultant superpixels is composed of perceptually similar pixels. For instance, the pioneering work, *watershed* [29], considers the flooding of the water from local minima in an image to retrieve the segments of superpixels. As a result, the shapes of the superpixels may be too irregular to adhere to the boundaries of objects.

Mean shift [30] searches the local maxima of a density function by using an iterative mode-seeking procedure. After convergence, pixels belonging to the same mode form a superpixel. To speed up mode-seeking, *quick shift* [31] adjusts the under-segmentation and over-segmentation of clusters by moving points to their nearest neighbors. Based on mode-seeking, these methods can automatically determine the number and the compactness of superpixels in an image. However, extra hyperparameters in mode-seeking such as those in the kernel function need to be set in advance.

Levinshtein et al. [32] delivered a method for compiling the *TurboPixels*. Their method uniformly places the initial seeds on images and gradually expands the superpixels from the seeds by a level set based geometric flow algorithm. The method can make the sizes of the superpixels uniform, but it is less efficient compared to other gradient-based methods. Zeng et al. [33] proposed structure-sensitive superpixels based on the geodesic distance computed from geometric flows. The number of superpixels is automatically determined by the energy functions of the structure density and compactness

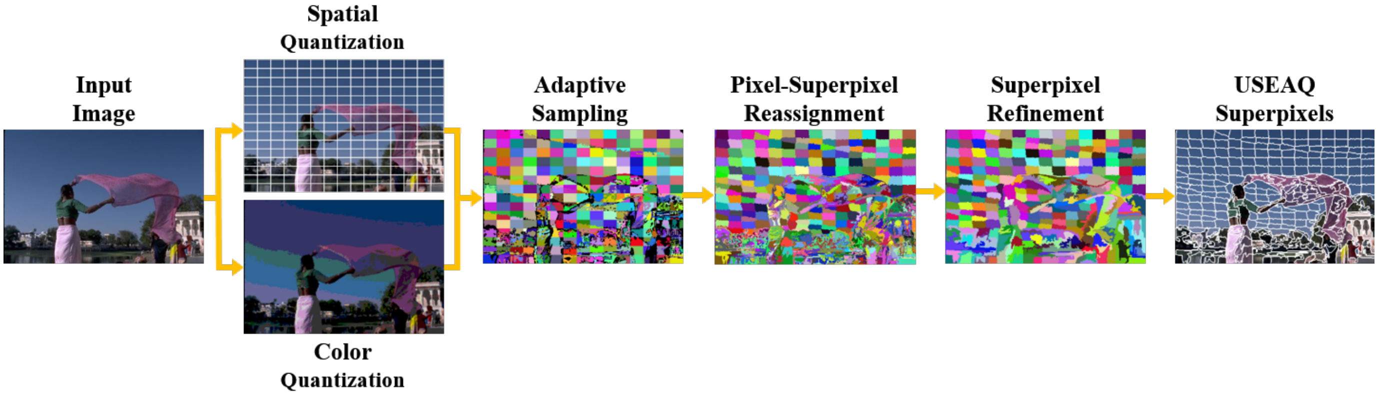


Fig. 1. The overview of the proposed approach to superpixel extraction. Firstly, an input image undergoes joint spatial and visual quantizations. Then, an adaptive sampling mechanism is applied to the quantized regions to compile the superpixel candidates. Finally, the resultant superpixels are obtained by performing pixel-superpixel reassignment and superpixel refinement.

constraints. However, the running time of computing structure-sensitive superpixels is even longer than that of TurboPixels.

Achanta et al. [16] proposed a method, called *simple linear iterative clustering* (SLIC), to construct superpixels. SLIC sets the initial seeds as the cluster centers obtained by applying k-means to the image. The computational complexity of SLIC, mainly on running k-means, is dramatically reduced by considering local search regions. Although SLIC is efficient, the yielded superpixels are sensitive to the locations of initial seeds. Liu et al. [34] proposed *manifold SLIC* (M-SLIC), which extends SLIC to retrieve small superpixels in content-dense regions and large superpixels in content-sparse regions, respectively. *Restricted centroidal Voronoi tessellation* is used to induce the content-sensitive superpixels. Because M-SLIC works on manifolds, which can be computed efficiently, it is faster than many the state-of-the-art methods. To further increase the computational speed of SLIC, Achanta and Susstrunk [35] proposed simple non-iterative clustering (SNIC) to construct superpixels with one iteration.

Van den Bergh et al. [36] extracted superpixels via using an energy-driven sampling (SEEDS) method. Their method initializes superpixels as the uniform cells, and progressively adjusts the boundaries of superpixels according to an energy function that takes the color homogeneity and shape prior of superpixel boundaries into account. The optimization of the energy function is solved by a hill-climbing algorithm. However, the shapes of the generated superpixels are often irregular. The computational cost time also significantly increases with respect to the number of superpixels. Inspired by SEEDS, Yao et al. [37] proposed efficient topology preserving segmentation (ETPS) which applies a coarse-to-fine energy update strategy to efficiently achieve the energy minima for superpixel extraction.

Shen et al. [38] used *lazy random walk* (LRW) to represent the relationship between a seed and its neighboring pixels, and generate superpixels according to the relationship. To improve the performance, an energy optimization function based on texture information and object boundaries in the image is developed and adopted. However, their method is time-consuming. Fu et al. [39] proposed *regularity preserved superpixels* (RPS) to keep regularity properties. Based on the initial seeds, the pixels are re-assigned based on locally maximal edge magnitudes. The shortest path algorithm retrieves local optimal boundaries. They also extended RPS to generate supervoxels. However, RPS is much less efficient than SLIC and SEEDS. Shen et al. [40] proposed to use the density-based spatial clustering of applications with noise (DBSCAN) to decrease the computational costs for superpixel construction. Small initial superpixels are further merged with adjacent superpixels with similar color distributions. Hu et al. [41] proposed a spatial-constrained watershed superpixel algorithm (SCoW) which can provide more compact and evenly distributed superpixels by placing evenly distributed marker points in the image. The flooding process of the watershed can be processed without iteration to achieve real-time efficiency.

Li and Chen [42] proposed the *linear spectral clustering* (LSC) to construct uniform superpixels. The objective function of LSC is highly similar to that of normalized cuts, which is defined on a graph structure. Thus, LSC adopts a graph-based objective function. To speed up the process of superpixel extraction, LSC solves the objective via iteratively using k-means clustering, instead of eigendecomposition which is widely adopted in graph-based methods. New extension and applications of LSC were proposed and discussed in [43].

Most gradient-based approaches adopt iterative optimization procedures to generate superpixels that adhere to the boundaries of objects. To avoid the iterative optimization procedures and further improve the efficiency, our approach merges subregions generated from spatial and color quantizations via MAP estimation in a one-pass fashion. Hence, our approach can achieve much higher computational efficiency compared with the existing gradient-based approaches.

III. Our Approach

The proposed approach is introduced in this section. Fig. 1 gives the overview of our approach for a better illustration. The proposed approach carries out superpixel extraction in a one-pass manner to reduce the computational burden of the optimization process. To this end, we apply the spatial quantization, which initially decomposes an image into rectangular regions purely based on the positions of pixels. In addition to the spatial information, dominant colors of the image are obtained by partitioning the color space. The spatial and visual quantizations are introduced in Section III-A. Then we present in Section III-B an adaptive sampling mechanism that compiles the initial superpixel candidates on the quantized image. In Section III-C, a graph structure is employed to represent the relationships between pixels and superpixel candidates. A process of pixel-superpixel reassignment is described. The reassignment is carried out via maximum a posteriori (MAP) estimation where both spatial and visual similarities between each pixel and superpixel candidates are jointly considered. Finally in Section III-D, superpixel refinement by merging small and irregular candidates is performed based on MAP estimation so that the desired number of high-quality superpixels is obtained.

A. Spatial and Visual Quantizations

The goal of image quantization is to yield a set of high-quality superpixel candidates. The higher the quality of the candidates, the less the efforts required for post-processing and further refinement. Nevertheless, the computational cost of quantization needs to be taken into account so that this step will not be the computational bottleneck. Thus, spatial and visual information are jointly used to conduct image quantization.

Given an image I as well as a desired number of superpixels δ, we firstly perform spatial quantization that uniformly divides the image I into rectangular regions. Let W and H denote the width and the height of I, respectively. The width and the height of each rectangular region are given as follows:

$$w = \left\lfloor \frac{W}{\sqrt{\delta}} \right\rfloor \text{ and } h = \left\lfloor \frac{H}{\sqrt{\delta}} \right\rfloor. \tag{1}$$

Let $\boldsymbol{\gamma}_i = [u_i \; v_i]^\top$ be the center of the ith region $\mathcal{R}_i$, while $\mathbf{p}_k = [x_k \; y_k]^\top$ be the (x, y) position of the kth pixel p_k in I. After spatial quantization, the region $\mathcal{R}_i$ is composed of the following pixels:

$$\mathcal{R}_i = \{p_k \, ||\| \, \mathbf{p}_k - \boldsymbol{\gamma}_i \, \| < \| \, \mathbf{p}_k - \boldsymbol{\gamma}_j \, \|, \forall j \neq i\}. \tag{2}$$

Despite the simplicity, spatial quantization is essential to the generation of regular superpixels.

In addition to the spatial requirement, the most crucial criterion for superpixel generation is to ensure the high visual similarity among pixels within the same superpixel. In many recent approaches such as SLIC [16], each spatially quantized region is considered an initial superpixel. These regions cannot adhere to the boundaries of objects, since the appearance of pixels is not consistent in the individual regions. To address this issue, these approaches, including both graph-based and gradient-based ones, apply iterative processes to recursively retrieve pixels with similar colors. Aiming at developing an efficient and effective approach to superpixel extraction, we avoid adopting an iterative process. Instead, the concept of color quantization [44], [45] is adopted to effectively retrieve the quantized color of each pixel.

Let $\mathbf{c}_k = [r_k \; g_k \; b_k]^\top$ be the three-dimensional color vector of pixel p_k in the RGB color space. We firstly partition the RGB color space into N disjoint color groups, $\{\mathcal{C}_n\}_{n=1}^{N}$. For efficiency, we uniformly divide each color channel into θ bins, leading to $N = \theta \times \theta \times \theta$ cubes. The value of θ is empirically set to 4 in the experiments. For image I, each cube $\mathcal{C}_n$ contains the following pixels

$$\mathcal{C}_n = \{p_k \, ||\| \, \mathbf{c}_k - \tilde{\mathbf{q}}_n \, \| < \| \, \mathbf{c}_k - \tilde{\mathbf{q}}_m \, \|, \forall m \neq n\}, \tag{3}$$

where $\tilde{\mathbf{q}}_n$ and $\tilde{\mathbf{q}}_m$ are the centers of cubes $\mathcal{C}_n$ and $\mathcal{C}_m$, respectively. Taking input image I into account, we set the quantized color of cube $\mathcal{C}_n$ to

$$\mathbf{q}_n = \frac{\sum_{p_k \in \mathcal{C}_n} \mathbf{c}_k}{|\mathcal{C}_n|}, \text{ for } n = 1, 2, ..., N, \tag{4}$$

where $|\mathcal{C}_n|$ is the number of pixels that belong to color cube $\mathcal{C}_n$. In (4), the quantized color $\mathbf{q}_n$ of cube $\mathcal{C}_n$ is the mean color of pixels falling into this cube. After color quantization, the quantized colors of all pixels of I are attained.

An input image is spatially and visually quantized in the procedure described above. Unlike many existing methods, no complex operations like clustering and iterative processing are required in the stage of quantization of this work. The complexities of both spatial and visual quantizations grow linearly with respect to the number of pixels.

B. Adaptive Sampling from Quantized Regions

Our preliminary work [17] analyzes the distribution of the quantized colors in each spatially quantized rectangle region. It divides the spatially quantized region into sub-regions according to the quantized colors, and retrieves the largest sub-region as a superpixel candidate. In this way, the number of the superpixel candidates is the same as that of spatially quantized regions, i.e., δ in (1). Pixels within the same superpixel candidate tend to be spatially and visually similar, since they belong to the same group in both spatial and color quantizations.

However, our preliminary work [17] neglects the possible variations among the spatially quantized regions. The homogeneous regions often cover pixels of one or few quantized colors, while the cluttered regions contain pixels of many quantized colors. Fig. 2 gives an example. The input image of a castle and its color quantization are shown in Fig. 2(a) and Fig. 2(b), respectively. The joint spatial and color quantizations are displayed in Fig. 2(c). One spatially quantized region is detailed in Fig. 2(d), where different quantized colors are present. In a clutter region like that in Fig. 2(d) covering multiple objects or background simultaneously, one superpixel is insufficient to represent the whole region. This unfavorable effect will accumulate and lead to segmentation error in the following step of pixel-superpixel reassignment. In practice, fewer superpixels of larger sizes are preferable for homogeneous regions to have a compact representation, while more superpixels of smaller sizes are required in cluttered regions to adhere to complex boundaries.

This work addresses this issue by sampling one or multiple superpixel candidates from each spatially quantized region $\mathcal{R}_i$. Unlike the prior work [17] where one superpixel candidate is sampled from each spatially quantized region, this work allows

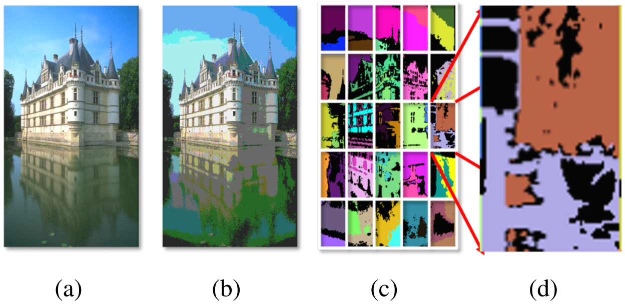


Fig. 2. (a) An input image of a castle. (b) The result of color quantization with $N = \theta^3$ quantized colors, where $\theta = 4$. (c) The result of joint spatial and color quantizations. (d) The quantized colors in a spatially quantized region which contains the forest and the stone coast.

sampling at most S candidates for each region. To this end, we set a size threshold

$$\tau = \frac{wh}{S}, \tag{5}$$

where w and h in (1) are the width and the height of the region, respectively. For each region $\mathcal{R}_i$, the pixels of a quantized color in $\mathcal{R}_i$ yield a superpixel candidate if the number of these pixels is larger than τ. Hence, there are at most S superpixel candidates sampled from $\mathcal{R}_i$. The higher the variations in $\mathcal{R}_i$, the more the sampled superpixel candidates. Despite the size threshold, one superpixel candidate is sampled for the quantized color with the most pixels in each $\mathcal{R}_i$. It ensures that each $\mathcal{R}_i$ has at least one superpixel candidate. This sampling process is repeated for all the quantized regions $\{\mathcal{R}_i\}_{i=1}^{\delta}$. The set of the resultant superpixel candidates $\{\mathcal{SP}_m\}_{m=1}^{M}$, where M is the number of superpixel candidates with the range, $\delta \leq M \leq S \times \delta$. The value of S is empirically set. The effect of setting different values of S is analyzed and discussed in the experiments.

After sampling, we compute the dominant color and the center of each superpixel $\mathcal{SP}_m$ as follows:

$$\mathbf{v}_m = \frac{\sum_{p_k \in \mathcal{SP}_m} \mathbf{c}_k}{|\mathcal{SP}_m|} \tag{6}$$

and

$$\boldsymbol{\ell}_m = \frac{\sum_{p_k \in \mathcal{SP}_m} \mathbf{p}_k}{|\mathcal{SP}_m|}, \tag{7}$$

where $|\mathcal{SP}_m|$ is the size of superpixel $\mathcal{SP}_m$. $\mathbf{c}_k$ and $\mathbf{p}_k$ are the color and the location of pixel p_k, respectively.

Note that we sample from region $\mathcal{R}_i$ at most S superpixel candidates, each of which consists of the pixels that belong to the same group under both spatial and visual quantizations. With this procedure of candidate generation, there may exist pixels in $\mathcal{R}_i$ that are not covered by any superpixel candidates. In the cases, we temporarily assign each of these pixels to its most spatially and visually similar candidate via the measure in (11), which will be introduced later. It follows that each pixel in image I is assigned to one particular superpixel candidate.

The current pixel-superpixel assignment is efficiently obtained in the complexity linear to the number of pixels of the image if the value of S is small enough to be neglected. The resultant superpixel candidates are of high quality in terms of the similarity among pixels belonging to the same candidate. However, the current pixel-superpixel assignment has a problem. Namely, the superpixel candidates do not preserve object boundaries, but the rectangular spatial quantization. To address this issue, we perform *pixel-superpixel reassignment*, described in the following section.

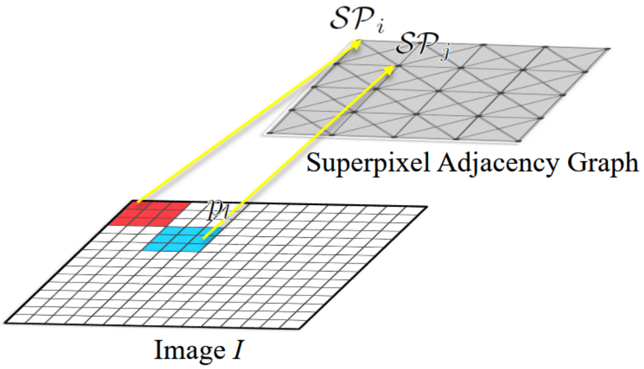


Fig. 3. During pixel-superpixel reassignment, a pixel p_l can be reassigned to superpixel candidate $\mathcal{SP}_j$ which covers it or some candidate $\mathcal{SP}_i$ that is connected to $\mathcal{SP}_j$ in $\mathcal{G}$.

C. MAP-based Pixel-Superpixel Reassignment

For pixel-superpixel reassignment, the neighborhood relationships between the superpixel candidates and pixels are required. To this end, we construct a *superpixel adjacency graph* by linking nearby superpixel candidates. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote the graph. $\mathcal{V}$ is the set of the nodes, each of which corresponds to a superpixel candidate, leading to $|\mathcal{V}| = M$. An edge $e(m, m')$ is added if candidates $\mathcal{SP}_m$ and $\mathcal{SP}_{m'}$ are nearby enough. Specifically, the edge set $\mathcal{E} = \{e(m, m') | 1 \leq m, m' \leq M\}$ is defined by

$$e(m, m') = \begin{cases} 1, & \text{if } \|\boldsymbol{\ell}_m - \boldsymbol{\ell}_{m'}\| \leq \max(w, h), \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where $\|\boldsymbol{\ell}_m - \boldsymbol{\ell}_{m'}\|$ is the Euclidean distance between the centers of candidates $\mathcal{SP}_m$ and $\mathcal{SP}_{m'}$. w and h in (1) are the width and the height of a spatially quantized rectangle respectively. Note that the edges defined in (8) may link spatially disconnected superpixel candidates. These edges are defined due to the performance consideration. Consider a spatially quantized region covering highly textured areas of an image. There will be some spatially disconnected superpixels with similar appearances. For pixel-superpixel reassignment, we found that connecting these superpixels via (8) helps improve the performance.

Note that unlike graph-based methods optimizing over graph structures, graph $\mathcal{G}$ is used to describe the connection relationship between superpixel candidates and pixels. Specifically, we assume that a pixel can be reassigned to one member of a superpixel candidate subset. This subset consists of the superpixel candidate covers that pixel and all the candidates that are connected to that candidate in graph $\mathcal{G}$. Fig. 3 gives an example. A pixel p_l can be reassigned to superpixel candidate

$\mathcal{SP}_j$ which covers it or some candidate $\mathcal{SP}_i$ that is connected to $\mathcal{SP}_j$ in the graph $\mathcal{G}$.

To assign each pixel to the most spatially and visually similar superpixel, we formulate it as an instance of the *maximum a posteriori* (MAP) estimation problem. The degree of consensus of assigning a pixel p_k, with color $\mathbf{c}_k$ and position $\mathbf{p}_k$, to superpixel candidate $\mathcal{SP}_m$ is measured by referring to the posterior probability $p(\mathcal{SP}_m|p_k)$ for $m = 1, 2, ..., M$. Based on the formula of Bayes' theorem, the posterior probability function is derived as follows:

$$p(\mathcal{SP}_m|p_k) \propto p(p_k|\mathcal{SP}_m)p(\mathcal{SP}_m), \tag{9}$$

where $p(p_k|\mathcal{SP}_m)$ is the likelihood function representing the conditional probability of covering pixel p_k given superpixel $\mathcal{SP}_m$. $p(\mathcal{SP}_m)$ is the prior probability of the superpixel candidate $p(\mathcal{SP}_m)$. We use it to encode our prior knowledge/assumption about whether it is valid for candidate $p(\mathcal{SP}_m)$ to cover pixel p_k.

After sampling superpixel candidates, we use the mean color $\mathbf{v}_m$ in (6) and the center position $\boldsymbol{\ell}_m$ in (7) to represent candidate $\mathcal{SP}_m$. The likelihood probability function $p(p_k|\mathcal{SP}_m)$ in (9) is defined by comparing the spatial positions and colors between p_k and $\mathcal{SP}_m$ as follows:

$$p(p_k|\mathcal{SP}_m) = p(\mathbf{p}_k, \mathbf{c}_k|\boldsymbol{\ell}_m, \mathbf{v}_m). \tag{10}$$

Because the spatial and color evidence is independent, the likelihood probability function in (10) can be rewritten as

$$p(p_k|\mathcal{SP}_m) = p(\mathbf{p}_k|\boldsymbol{\ell}_m)p(\mathbf{c}_k|\mathbf{v}_m), \tag{11}$$

where $p(\mathbf{p}_k|\boldsymbol{\ell}_m)$ and $p(\mathbf{c}_k|\mathbf{v}_m)$ are the spatial and visual likelihood functions, respectively. To represent the spatial likelihood, we consider the distance between the positions of p_k and $\mathcal{SP}_m$, and define it by

$$p(\mathbf{p}_k|\boldsymbol{\ell}_m) \propto \exp\left(-\omega\|\mathbf{p}_k - \boldsymbol{\ell}_m\|\right). \tag{12}$$

To avoid the effects of different image resolutions, we normalize $\mathbf{p}_k$ and $\boldsymbol{\ell}_m$ with respect to the image width W and height H in advance. To compute the color likelihood, we consider the similarity between the color of p_k and the mean color of $\mathcal{SP}_m$. The color likelihood function is designed as

$$p(\mathbf{c}_k|\mathbf{v}_m) \propto \exp\left(-(1-\omega)\|\mathbf{c}_k - \mathbf{v}_m\|\right). \tag{13}$$

The parameter $\omega \in [0, 1]$ in (12) and (13) controls relative importance between the spatial and visual evidences. We will investigate the effect of ω in the experiments.

The prior function $p(\mathcal{SP}_m)$ in (9) is used to encode our prior assumption about pixel-superpixel assignment. Considering the reassignment of a pixel p_k, suppose p_k is currently assigned to superpixel candidate $\mathcal{SP}_{\pi_k}$ in the procedure of adaptive sampling. We assume that pixel p_k can only be assigned to either $\mathcal{SP}_{\pi_k}$ or one of the other candidates connecting to $\mathcal{SP}_{\pi_k}$ in graph $\mathcal{G}$. It follows that we have

$$p(\mathcal{SP}_m) \propto \begin{cases} 1, & \text{if } m = \pi_k, \\ 1, & \text{else if } e(m, \pi_k) = 1 \text{ in (8)}, \\ 0, & \text{otherwise.} \end{cases} \tag{14}$$

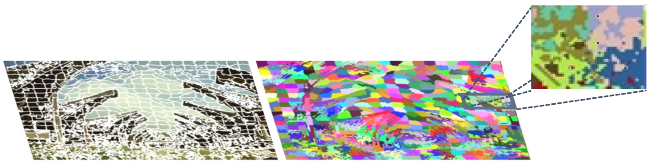


Fig. 4. The results after pixel-superpixel reassignment. It can be observed that small objects with large intra-object variations or local regions with dramatic appearance changes often cause fragments i.e., irregular and less compact superpixels of small sizes.

With the likelihood and prior functions defined in (11) and (14) respectively, the most plausible superpixel $\mathcal{SP}^*$ for pixel p_k is obtained by maximizing the posterior probability i.e.,

$$\mathcal{SP}^* = \arg\max_{\mathcal{SP}_m} p(p_k|\mathcal{SP}_m)p(\mathcal{SP}_m). \tag{15}$$

The process of pixel-superpixel reassignment via (15) is repeated for every pixel. This process can be efficiently done by taking advantage of the sparsity of the prior probability in (14). Once pixel-superpixel reassignment is completed, the updated superpixels are obtained.

D. Superpixel Refinement

The desired number of superpixels is δ. However, the current number of superpixels M is between δ and $S\delta$ due to the adaptive sampling of superpixel candidates. Besides, we also observe that small objects with large intra-object variations or local regions with dramatic appearance changes often lead to a number of small superpixels representing the unnecessary details. See Fig. 4 for an example. Furthermore, these superpixels typically become irregular and not compact, hence violating the desirable property of high-quality superpixels. To solve the problem, the refinement process is applied to superpixels so that small superpixels are merged into their spatially connected and visually similar superpixels.

To this end, we modify the superpixel adjacency graph $\mathcal{G}$ by linking only superpixels that are spatially connected. Each edge $e(m, m')$ of $\mathcal{G}$ in (8) is reset to

$$e(m, m') = \begin{cases} 1, & \text{if } \mathcal{SP}_m \text{ and } \mathcal{SP}_{m'} \text{ are connected,} \\ 0, & \text{otherwise,} \end{cases} \tag{16}$$

for $1 \leq m, m' \leq M$.

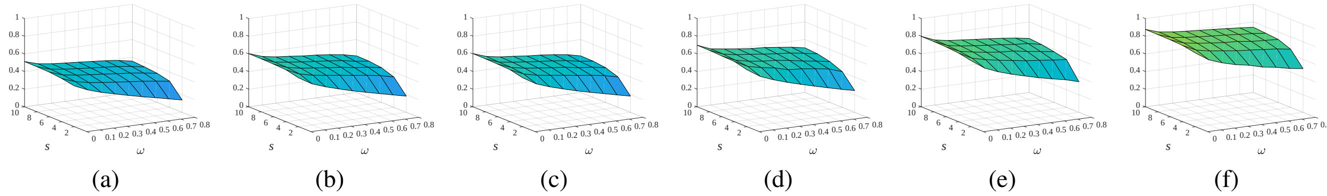
Similar to pixel-superpixel reassignment, merging a small superpixel $\mathcal{SP}_j$ into another superpixel $\mathcal{SP}_i$ is carried out by using MAP estimation with posterior probability defined below

$$p(\mathcal{SP}_i|\mathcal{SP}_j) \propto p(\mathcal{SP}_j|\mathcal{SP}_i)p(\mathcal{SP}_i), \tag{17}$$

where $p(\mathcal{SP}_j|\mathcal{SP}_i)$ and $p(\mathcal{SP}_i)$ are the likelihood and the prior functions, respectively.

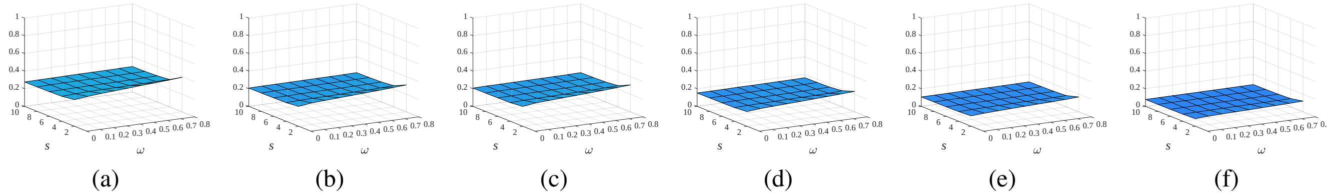
Unlike pixel-superpixel reassignment, the likelihood function for merging superpixels considers only the visual similarity, since the requirement of spatial connection will be enforced in the prior function. Specifically, the likelihood is given as follows

$$p(\mathcal{SP}_j|\mathcal{SP}_i) \propto \exp\left(-\|\mathbf{v}_i - \mathbf{v}_j\|\right), \tag{18}$$



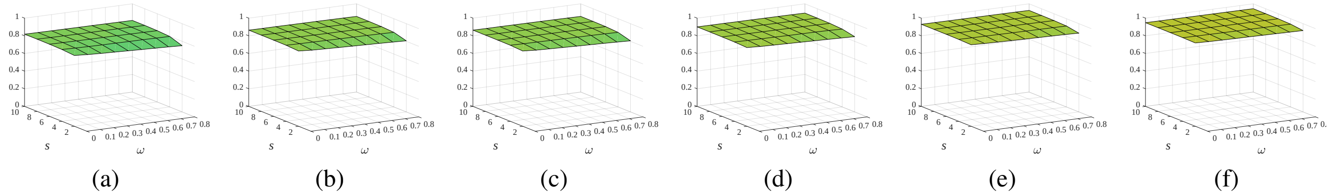
(a) (b) (c) (d) (e) (f)

Fig. 5. The performance of our approach in BR with various value combinations of S and ω when the number of the generated superpixels is set to (a) 25, (b) 50, (c) 100, (d) 250, (e) 500, and (f) 1000.



(a) (b) (c) (d) (e) (f)

Fig. 6. The performance of our approach in UE with various value combinations of S and ω when the number of the generated superpixels is set to (a) 25, (b) 50, (c) 100, (d) 250, (e) 500, and (f) 1000.



(a) (b) (c) (d) (e) (f)

Fig. 7. The performance of our approach in ASA with various value combinations of S and ω when the number of the generated superpixels is set to (a) 25, (b) 50, (c) 100, (d) 250, (e) 500, and (f) 1000.

where $\mathbf{v}_i$ and $\mathbf{v}_j$ are the mean colors of superpixels $\mathcal{SP}_i$ and $\mathcal{SP}_j$, respectively. The prior function is used to ensure the spatial connection of two superpixels to be merged, namely

$$p(\mathcal{SP}_i) \propto \begin{cases} 1, & \text{if } e(i,j) = 1 \text{ in (16)}, \\ 0, & \text{otherwise}. \end{cases} \tag{19}$$

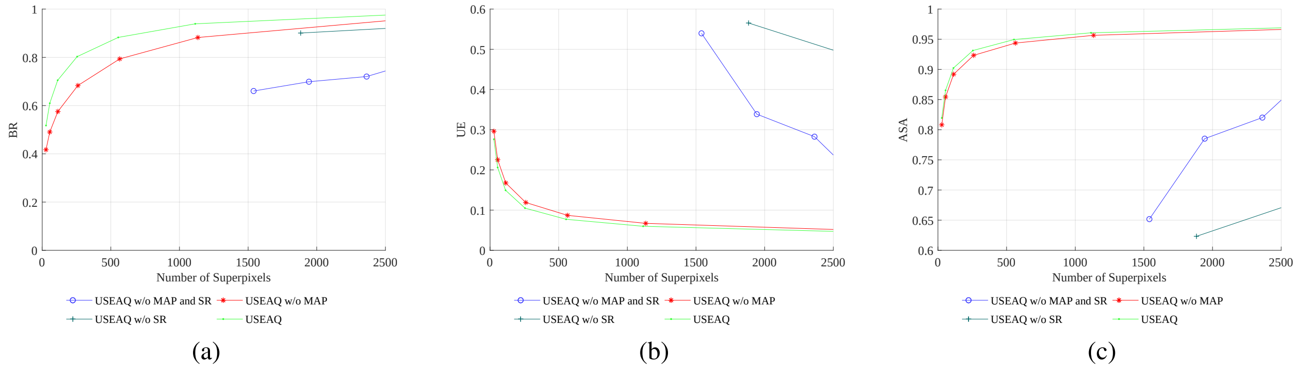
With the likelihood in (18) and the prior in (19), the posterior in (17) can be estimated. For merging superpixels, we retrieve the superpixel of the smallest area, seek its spatially connected and visually similar superpixel that maximizes the posterior in (17), and merge these two superpixels. We also compute the mean color of the newly generated superpixel, and update the graph in (16) accordingly. The procedure of superpixel merging is repeated until the desired number of superpixels is obtained. As shown in the experiments, the resultant superpixels are regular, compact, and of high quality.

IV. Experimental Results

The performance of our approach is evaluated in this section. We first describe the used performance measures and the adopted datasets for evaluation. Then, we discuss how to set the values of the parameters in the proposed approach. Finally, our approach is compared with the state-of-the-art approaches to superpixel extraction in terms of accuracy and efficiency. The comparison results are reported and analyzed. In addition to the quantitative results, the generated superpixels by various approaches on some examples are shown and discussed.

A. Dataset and Evaluation Metrics

Our approach is evaluated on the *Berkeley Segmentation Dataset and Benchmark* (BSDS500) [18], which contains 500 images with the manually labeled ground truth and the benchmarking code. These images include landscapes, animals, humans, buildings, and artifacts captured in outdoor scenes. Each image is manually segmented by at least three people. The human-marked boundaries are treated as the ground truth. During evaluation, the estimated boundaries of superpixels are compared with each of the available ground truth of each image. The average performance is reported. We adopted three evaluation metrics for measuring the performance of the generated superpixels, including *boundary recall* (BR), *under-segmentation error* (UE) [16], and *achievable segmentation accuracy* (ASA) [25]. All the three metrics are commonly used in the literature of superpixel extraction. Among the three metrics, BR represents the correctness of adhering to the true boundaries of objects. Higher BR indicates that the extracted superpixels better detect the boundaries of objects. UE measures the degree of the superpixel overlapping with multiple objects. It is the percentage of pixels that leak from the ground truth boundaries. Thus, low UE implies better adherence to the boundaries of objects. ASA is computed by matching the label of each superpixel with respect to the labels of ground truth, and evaluates the highest achievable object segmentation accuracy. Similar to BR, superpixels with high ASA often give better object representation of the image. In



(a) (b) (c)

Fig. 8. Effects of removing MAP-based pixel-superpixel reassignment (MAP) and/or superpixel refinement (SR) from the proposed USEAQ. The performance in (a) boundary recall (BR), (b) undersegmentation error (UE), and (c) achievable segmentation accuracy (ASA) is evaluated on the BSDS500 dataset.

addition to the effectiveness, the efficiency of our approach in running time is evaluated and compared with the state-of-the-art methods.

The second dataset for evaluation is the *Stanford background dataset* (SBD) [19], which contains 715 images selected from existing datasets. The resolutions of the images are approximately 320 × 240 pixels. Each image covers at least one foreground object. Those objects are present in different outdoor scenes including landscapes, animals, and streets. Compared to the BSDS500 dataset, the SBD dataset contains more complex scenes with multiple foreground objects, and hence is more challenging for superpixel evaluation. As suggested in [46], the average miss rate (AMR), average undersegmentation error (AUE), and average unexplained variation (AUV) are used as the evaluation metrics for superpixel extraction. Lower AMR, AUE, and AUV represent better performance.

B. Parameter Selection

In our approach, the values of three parameters should be set in advance. The first one is θ used in color quantization. It determines the number of the quantized colors. The second one is the maximum number of the adaptive sampling of superpixel candidates S in each spatially quantized region. The third one is the weight ω in (12) and (13). It controls the relative importance of the location information to the color information. As shown in our preliminary work [17], θ is not critical to the performance once the number of the quantized colors is sufficient. Thus unless further specified, we set $\theta = 4$, which results in $64 = \theta^3$ quantized colors, in the experiments. In the following, we investigate the effect on the performance with various values of S and ω on the BSDS500 dataset. The value range of ω is evaluated between 1 and 10, while that of ω is evaluated between 0 and 1. Fig. 5, Fig. 6, and Fig. 7 display the average performances in BR, UE and ASA respectively with different value combinations of S and ω. The number of superpixels is crucial to these performance measures, so in each figure, the performances with different numbers of the generated superpixels are reported.

As shown in Fig. 5(a) to 5(f), when the number of the generated superpixels increases, the performance in BR is improved significantly. With the increasing values of ω, the BR values decrease. Because large ω values imply to preserve the initial spatial regions in the grid as superpixels, the resultant superpixels less adhere to the boundaries of the objects. By fixing the values of ω and the number of superpixels, the BR values will increase when the value of S increases. When S becomes larger, the number of initial superpixel candidates will increase accordingly. More superpixel candidates can better preserve the details of both objects and background. As a result, the BR values will increase.

The results in UE with respect to ω and S are shown in Fig. 6. When increasing the value of ω, the error in UE also increases, because the shapes of superpixels tend to become more rectangular. With the increasing number of superpixels, the errors in UE decrease, which shows that more superpixels help prevent the superpixels from overlapping between multiple objects. Also shown in Fig. 6, larger S achieves lower UE values, because larger S can provide better boundary segmentation results. Finally, the ASA results are shown in Fig. 7. Similar to the observations of BR in Fig. 5, a small ω value leads to better performance in ASA with various numbers of the superpixels. When the values of S increase, the ASA values also increase.

To sum up, increasing the number of the generated superpixels will improve the performances in BR, UE, and ASA, but will make the running time longer and have a less compact image representation. In general, a large S value accompanied by a small ω value results in better performances in BR, UE, and ASA simultaneously, though it also makes the yielded superpixels irregular and less compact. We set S to 10 and ω to 0.01 in the rest of the experiments.

C. Effects of MAP-based Pixel-Superpixel Reassignment and Superpixel Refinement

We conduct the ablation studies where the effects of removing MAP-based pixel-superpixel reassignment (MAP for short) and/or superpixel refinement (SR for short) from the proposed USEAQ are measured. The three resultant variants of USEAQ are denoted by "USEAQ w/o MAP and SR" (without both MAP and SR), "USEAQ w/o SR" (without SR), and "USEAQ w/o MAP" (without MAP), respectively. We report and compare the performance of USEAQ and the three variants in BR, UE, and ASA in Fig. 8.

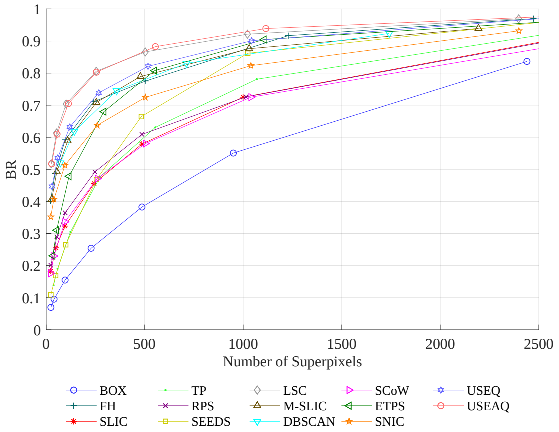


Fig. 9. Performance comparison of the state-of-the-art methods and our method USEAQ in boundary recall (BR).

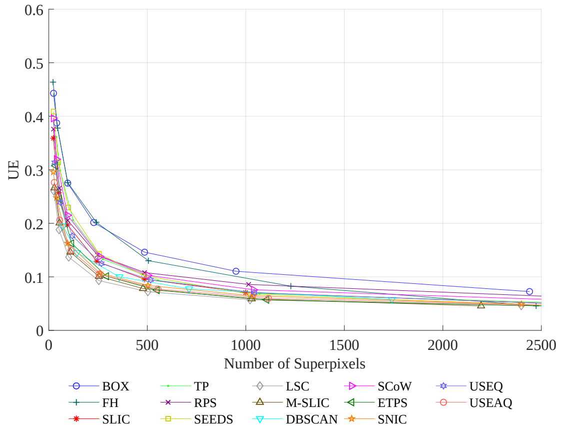


Fig. 10. Performance comparison of the state-of-the-art methods and our method USEAQ in undersegmentation error (UE).

As shown in Fig. 8(a), both MAP and SR are essential for making the resultant superpixels adhere to the true boundaries of objects since removing each or both of them results in a drop of the performance in BR. In Fig. 8(b) and Fig. 8(c), we observe that SR is more important than MAP for the performance measured in UE and ASA. The main reason is that skipping SR leads to too many tiny superpixels. The results in Fig. 8 confirm that both MAP and SR are key components in the proposed USEAQ.

D. Quantitative Comparisons

We compare our approach USEAQ to 11 real-time or near real-time, the state-of-the-art approaches to superpixel extraction, including FH [22], SLIC [16], Turbopixel (TP) [32], RPS [39], SEEDS [36], LSC [42], M-SLIC [34], DBSCAN [40], SCoW [41], ETPS [37], SNIC [35] and our preliminary approach USEQ [17] on the BSDS500 dataset. To realize the degree of difficulty of over-segmenting images in the BSDS500 dataset, we also show the performance of the spatial quantization grid (BOX) as a baseline. Because the number of the generated superpixels cannot be specified directly in FH, we adjust the parameters of FH to get the desired number of superpixels. For a fair comparison, the results of all the competing approaches are yielded by using the codes released by the original authors.

As shown in Fig. 9, USEAQ achieves the best performance in BR compared to the state-of-the-art approaches and our preliminary approach USEQ. Except for our method, LCS performs favorably against other completing methods. FH reaches similar performance in BR to USEQ, M-SLIC, DBSCAN, and ETPS. It is better than SNIC, SLIC, TP, RPS, SCoW, and SEEDS. Such results are consistent with those reported in [16], [36].

USEQ can be considered a special case of USEAQ when adaptive sampling is turned off. Based on the MAP-based pixel-superpixel reassignment in USEAQ, small objects with large intra-object variations can be dealt with multiple sampled superpixels. Thus, superpixels by USEAQ can more precisely adhere to the boundaries of objects. Because the quantized colors of pixels of small objects may be different, sampling just one superpixel candidate in each spatially quantized rectangle in USEQ is often insufficient to properly represent these small objects. Thus, USEAQ has a better generalization ability compared to USEQ, and consistently achieves higher BR values than USEQ with different numbers of the generated superpixels.

As shown in Fig. 10, the top three methods in UE are LSC, M-SLIC, and USEAQ, respectively. LSC adopts a graph-based objective function. It well represents the boundaries of both content-dense and content-sparse regions. M-SLIC also considers the differences between content-dense and contents-parse regions. USEAQ achieves this property by adaptive sampling. Thus, the superpixels generated by these methods represent small, highly textured regions well. As a result, these methods achieve lower UE values compared to the remaining competing approaches. Although FH gives high BR values as mentioned previously, it has worse performance in UE. It is because the number and shapes of the superpixels are not explicitly modeled in FH. Fig. 11 shows the performances in ASA of all the competing approaches and our approach with respect to different numbers of superpixels. LSC, M-SLIC, and USEAQ also give the superior results in ASA.

Let N denote the number of pixels in an image for superpixel extraction. The complexity of the spatial and color quantizations are $\mathcal{O}(N)$, because they are accomplished by accessing pixels sequentially in one and two passes respectively. The pixel-superpixel reassignment is performed via the MAP estimation in (9) once for each pixel. By taking advantage of the sparsity distribution in the prior function (14), the MAP estimation for each pixel is completed in constant time. Thus, the computational cost of pixel-superpixel reassignment for the whole image grows linearly to the number of pixels. As for adaptive sampling and superpixel refinement, their complexity is not higher than $\mathcal{O}(N)$. Consequently, the complexity of USEAQ is $\mathcal{O}(N)$.

As for the computational efficiency evaluation, all of the approaches are executed on a modern PC with an Intel Core

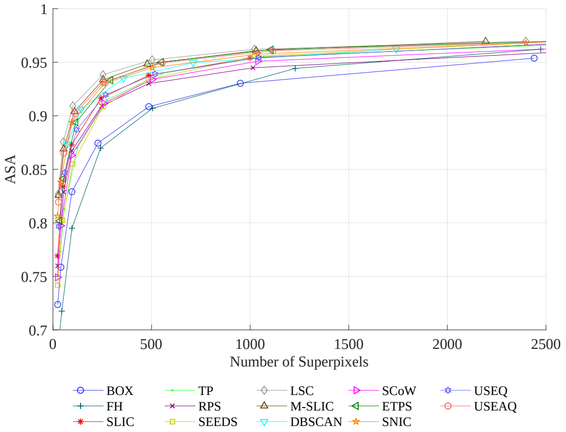


Fig. 11. Performance comparison of the state-of-the-art methods and our method USEAQ in achievable segmentation accuracy (ASA).

i7 3.40GHz processor and 16G memory. No GPU accelerators are applied. The programming language we used is C++. Fig. 12(a) shows the average running time of the top ten efficient approaches to superpixel extraction, including FH, SLIC, SEEDS, LSC, M-SLIC, SCoW, ETPS, SNIC, USEQ, and USEAQ with different numbers of the generated superpixels. Our preliminary approach USEQ is the most efficient compared to the remaining approaches, which reveals the efficiency of our MAP-based pixel-superpixel reassignment. USEAQ enables adaptive sampling, which leads to extra computational cost for taking into account more sampled superpixels. Nevertheless, the computation time of USEAQ is still remarkably less than that of other competing approaches. Although the complexities of SLIC and SEEDS are also $\mathcal{O}(N)$, their iterative procedures performed on superpixel generation lead to significantly increasing time when the number of the yielded superpixels increases. Similar to SLIC, the time complexity of M-SLIC is also $\mathcal{O}(N)$. Because of the additional computation in the restricted centroidal Voronoi tessellation, the running time of M-SLIC is longer than that of SLIC. In comparison, SNIC is a non-iterative variant of SLIC. It is more efficient than SLIC and M-SLIC. The iterative k-means clustering used in LSC also increases the computation time of LSC. Compared to SEEDS, SLIC, M-SLIC and LSC, SCoW do not use an iterative process and ETPS can achieve the convergence with a few iterations. Thus, the running time of SCoW and ETPS is significantly less than that of SEEDS, SLIC, M-SLIC and LSC.

To investigate the effect of image resolutions, we collect images with five different resolutions from 640×360 ($360p$) to 2560×1440 ($1440p$) and generate 2,500 superpixels for each image. When the image resolutions increase, the computation time of FH, SEEDS, SLIC, LSC, M-SLIC, SCoW, ETPS, and SNIC significantly increases compared to that of USEQ, and USEAQ as shown in Fig. 12(b). Because of the efficient computation of the MAP-based reassignment in both the pixel and superpixel levels, the computation time of USEQ, and USEAQ moderately increases with respect to the image resolutions. They are considerably faster than the state-of-the-art approaches.

TABLE I
SUMMARY OF SEVERAL METHODS FOR SUPERPIXEL EXTRACTION.

Approaches	# of SP	Compactness	Iteration	Complexity
FH	No	No	No	$\mathcal{O}(N\log N)$
SLIC	Yes	Yes	Yes	$\mathcal{O}(N)$
TP	Yes	No	No	$\mathcal{O}(N)$
RPS	Yes	Yes	Yes	$\mathcal{O}(N\log N)$
SEEDS	Yes	No	Yes	$\mathcal{O}(N)$
LSC	Yes	Yes	Yes	$\mathcal{O}(N)$
M-SLIC	Yes	Yes	Yes	$\mathcal{O}(N)$
DBSCAN	Yes	Yes	No	$\mathcal{O}(N)$
SCoW	Yes	Yes	No	$\mathcal{O}(N)$
ETPS	Yes	Yes	Yes	$\mathcal{O}(N)$
SNIC	Yes	Yes	No	$\mathcal{O}(N)$
USEQ	Yes	Yes	No	$\mathcal{O}(N)$
USEAQ	Yes	Yes	No	$\mathcal{O}(N)$

TABLE I reports the computational complexities of the state-of-the-art methods. We also summarize these methods by indicating that if the number of superpixels (SP) is controllable, if the compactness of superpixels is achievable, and if the method is iterative. Here we consider the number of superpixels in a method controllable if it can be set explicitly (e.g., giving the desired number of superpixels) or implicitly (e.g., specifying the parameters regarding the size of superpixels, which is closely relevant to the number of superpixels). Except for FH, most methods are able to control the number of superpixels. The compactness can also be controlled or automatically adjusted by most methods. In general, the methods whose computational complexity is $\mathcal{O}(N)$ and without using an iterative process can achieve better computational efficiency, as shown in Fig. 12.

The second dataset that we adopt is the *Stanford background dataset* (SBD) [19]. Following [46], the number of superpixels is set to 400 for evaluation. Fig. 13 show the tradeoffs of the performance and the computation cost of the competing approaches and our approach. The performance measures used in Fig. 13(a), Fig. 13(b), and Fig. 13(c) are AMR, AUE, and AUV, respectively. In Fig. 13, approaches present in the bottom left corners are of high performance and low computational costs. As shown in Fig. 13(a), the proposed USEAQ has the lowest AMR compared to the state-of-the-art approaches. Expect for our approach, ETPS and LSC outperform other approaches, which is consistent with the results reported in [46]. Fig. 13(b) shows that ETPS is superior than the other competing approaches. Nevertheless, USEAQ remains faster than ETPS and the difference of AUE between ETPS and USEAQ is 0.0018. M-SLIC achieves the lowest AUE but requires longer running time. As displayed in Fig. 13(c), USEAQ and ETPS have similar performance in AUV by considering both AUV and computational costs simultaneously. The results demonstrate that our method can achieve the boundary adherence of foreground objects in different outdoor scenes of the SBD dataset. It can be also observed in Fig. 13 that our preliminary work USEQ and the proposed USEAQ are the most efficient methods among all other competing ones.

To sum up, the proposed USEAQ is compared with the state-of-the-art methods for superpixel extraction. For effectiveness,

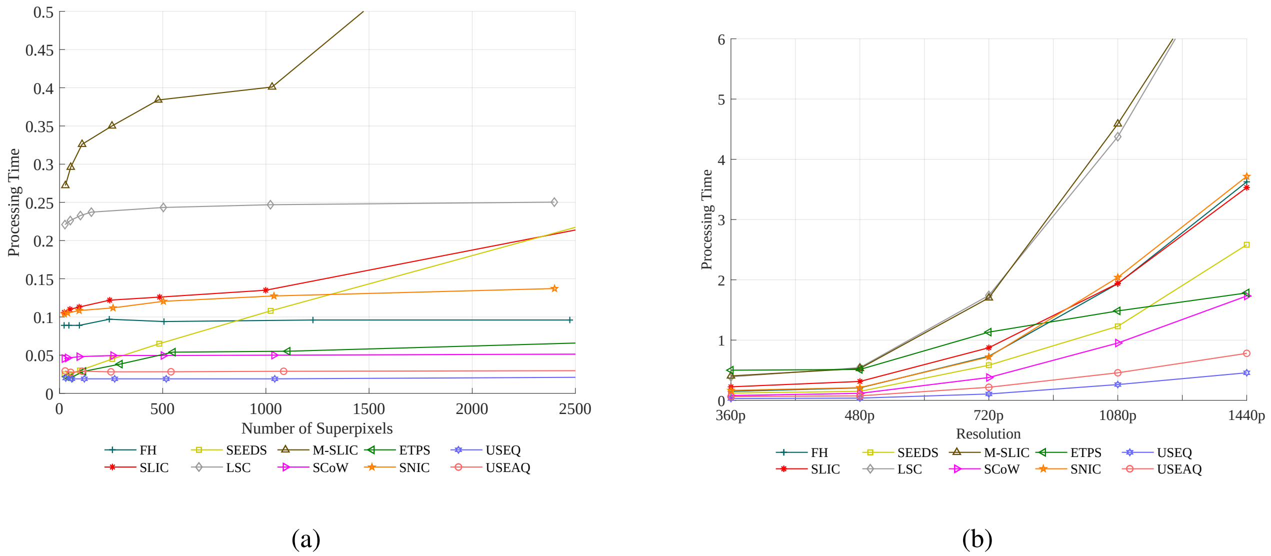


Fig. 12. (a) Average running time of the competing methods and USEAQ with different numbers of the generated superpixels. (b) Average running time of these methods with different image resolutions when the number of superpixels is set to 2,500.

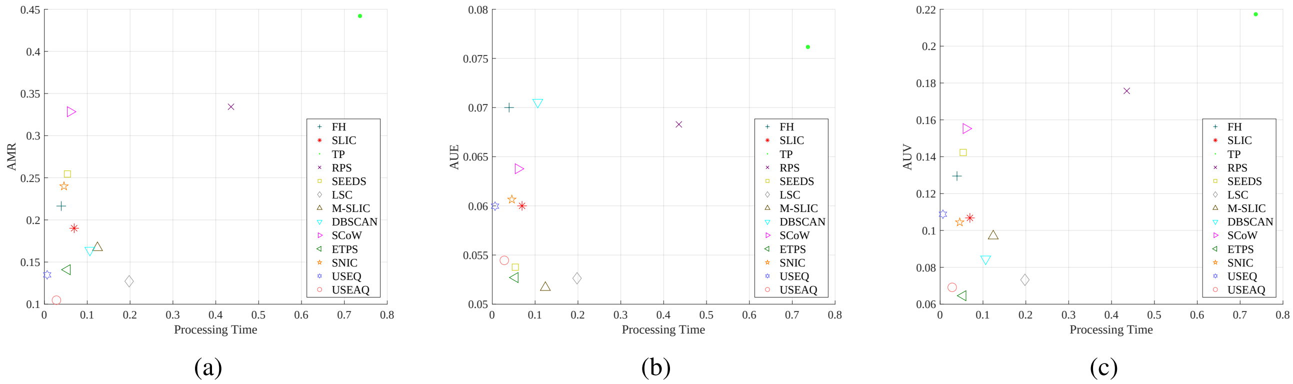


Fig. 13. Performance-computation tradeoffs of the state-of-the-art methods and our method USEAQ on the SBD dataset when the performance is evaluated in (a) average miss rate (AMR), (b) average undersegmentation error (AUE), and (c) average unexplained variation (AUV).

USEAQ achieves the best performance in BR and comparable UE and ASA on the BSDS dataset. By simultaneously considering the evaluation metrics AMR, AUE, AUV, and processing time on the SBD dataset, USEAQ performs favorably against or is comparable to all competing approaches. For efficiency, USEAQ slightly falls behind its prior work USEQ due to the extra step for adaptive sampling, which is the key to substantial performance improvement. Nevertheless, the proposed USEAQ is remarkably more efficient than the state-of-the-art methods.

E. Qualitative Comparisons

To gain insight into the quantitative results, Fig. 14 shows the extracted superpixels on a few images of the BSDS500 dataset by using USEAQ and some of the state-of-the-art approaches. To consider the results with different numbers of superpixels, each image is segmented into 250/500 superpixels. Fig. 14(a) gives the results by using our USEAQ. For smooth image regions, USEAQ can generate regular superpixels. This property can be observed evidently in the homogeneous or flat background regions in Fig. 14. On the other hand, the superpixels by USEAQ can precisely adhere to the boundaries of objects with large intra-object variations or highly textured background. This property can be found in the objects of most examples (rows). In contrast, the generated shapes and sizes of superpixels using FH are very irregular as shown in Fig. 14(c). SLIC, M-SLIC, TP, RPS, and SCoW generate more regular superpixels as shown in Fig. 14(d) ∼ 14(h), respectively, but fail to correctly adhere to the complex boundaries of objects. The results by SEEDS shown in Fig. 14(i) reach a good compromise between superpixel regularization and boundary adherence. As shown in Fig. 14(j) ∼ 14(m), ETPS, LSC, DBSCAN, and SNIC yield regular superpixels for smooth regions and irregular superpixels for cluttered regions. Thus, the resultant superpixels better adhere to the boundaries of objects and lead to better quantitative results.

Comparing our approach USEAQ in Fig. 14(a) to the competing approaches, we observe that USEAQ more adaptively controls the trade-off between superpixel regularization and boundary adherence conditioned on the regions. In homoge-

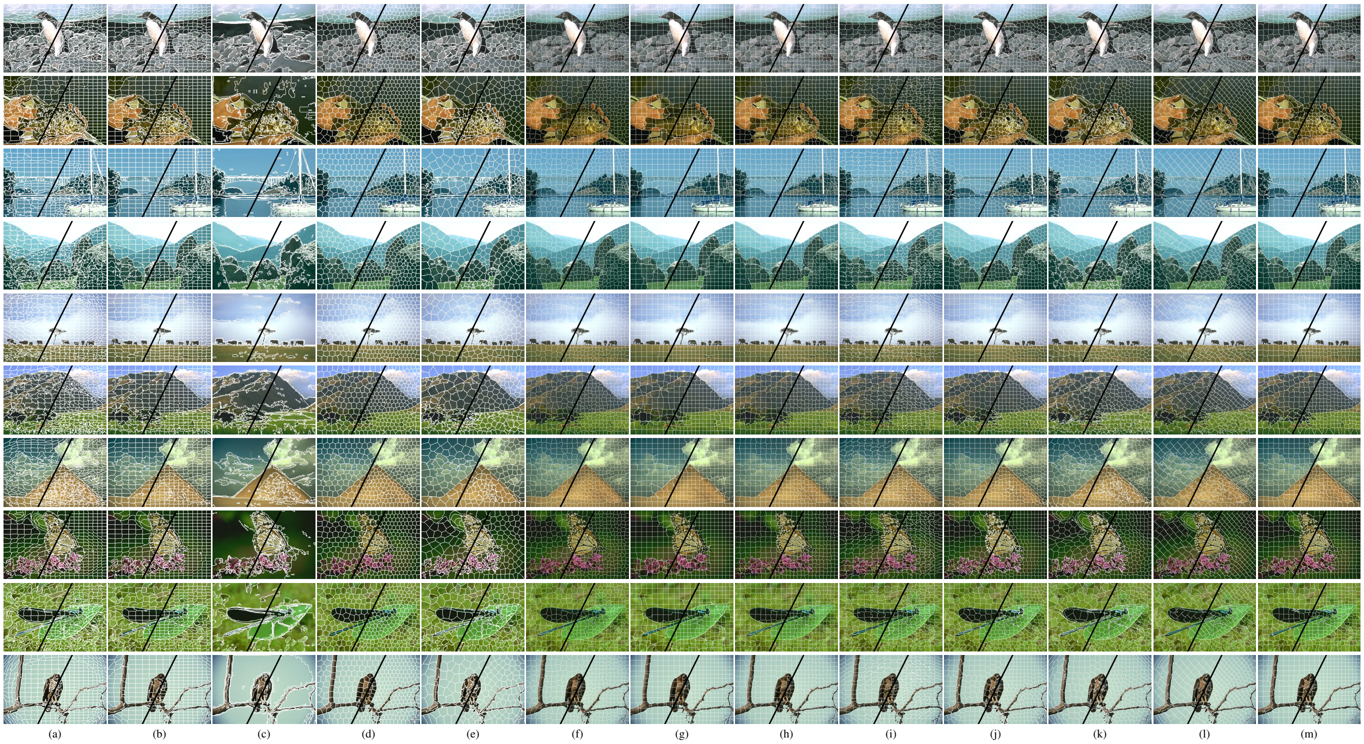


Fig. 14. The extracted superpixels from the BSDS500 dataset by using algorithms (a) USEAQ, (b) USEQ, (c) FH, (d) SLIC, (e) M-SLIC, (f) TP, (g) RPS, (h) SCoW, (i) SEEDS, (j) ETPS, (k) LSC, (l) DBSCAN, and (m) SNIC.

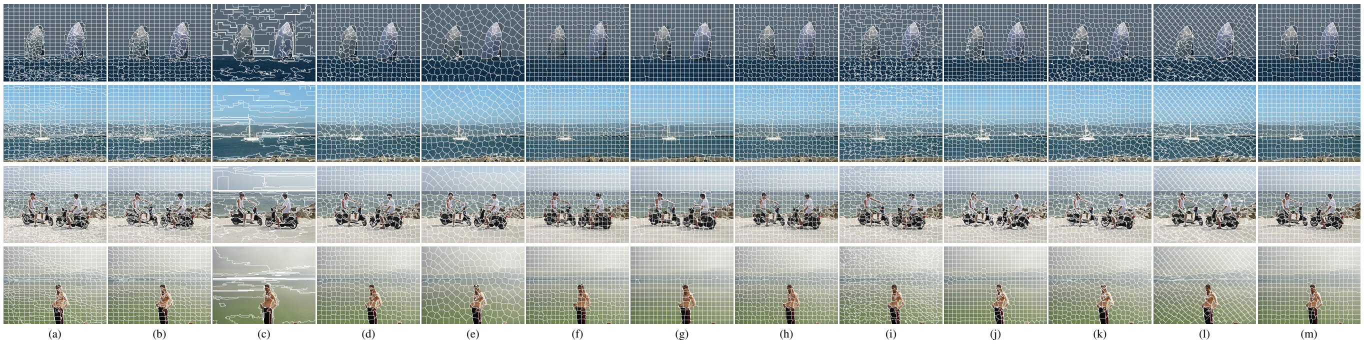


Fig. 15. The extracted superpixels from the SBD dataset by using algorithms (a) USEAQ, (b) USEQ, (c) FH, (d) SLIC, (e) M-SLIC, (f) TP, (g) RPS, (h) SCoW, (i) SEEDS, (j) ETPS, (k) LSC, (l) DBSCAN, and (m) SNIC.

neous or flat regions, it puts emphasis on compiling regular superpixels. In highly textured regions, it focuses on boundary adherence. This feature is achieved by adaptive sampling, and makes USEAQ reach better performance than USEQ in the diverse criteria, including BR, UE, and ASA.

When the number of superpixels is set to 400, Fig. 15 shows the extracted superpixels on a few images of the SBD dataset by using USEAQ and some of the state-of-the-art approaches. Similar to the visualization results on the BSDS500 dataset, USEAQ can successfully adhere to the boundaries of foreground objects on the SBD datasets by adaptive sampling from quantized regions of the foreground objects. For smooth background regions, it merges neighboring regions to generate more regular superpixels. As a result, USEAQ show better qualitative results.

TABLE II
PIXEL ACCURACY (%) BY APPLYING THE MEAN SHIFT ALGORITHM TO SUPERPIXELS GENERATED BY USEAQ, USEQ, SLIC, SEEDS, ETPS, AND LSC WITH RESPECT TO TWO DIFFERENT NUMBERS OF SUPERPIXELS (SP).

# of SPs	USEAQ	USEQ	SLIC	SEEDS	ETPS	LSC
400	**34.77**	34.59	34.14	34.45	34.49	33.46
1200	**39.18**	38.71	38.24	38.47	38.07	38.35

V. APPLICATIONS

In this section, we show that the superpixels generated by the proposed USEAQ facilitate two applications, image segmentation and supervoxel construction.

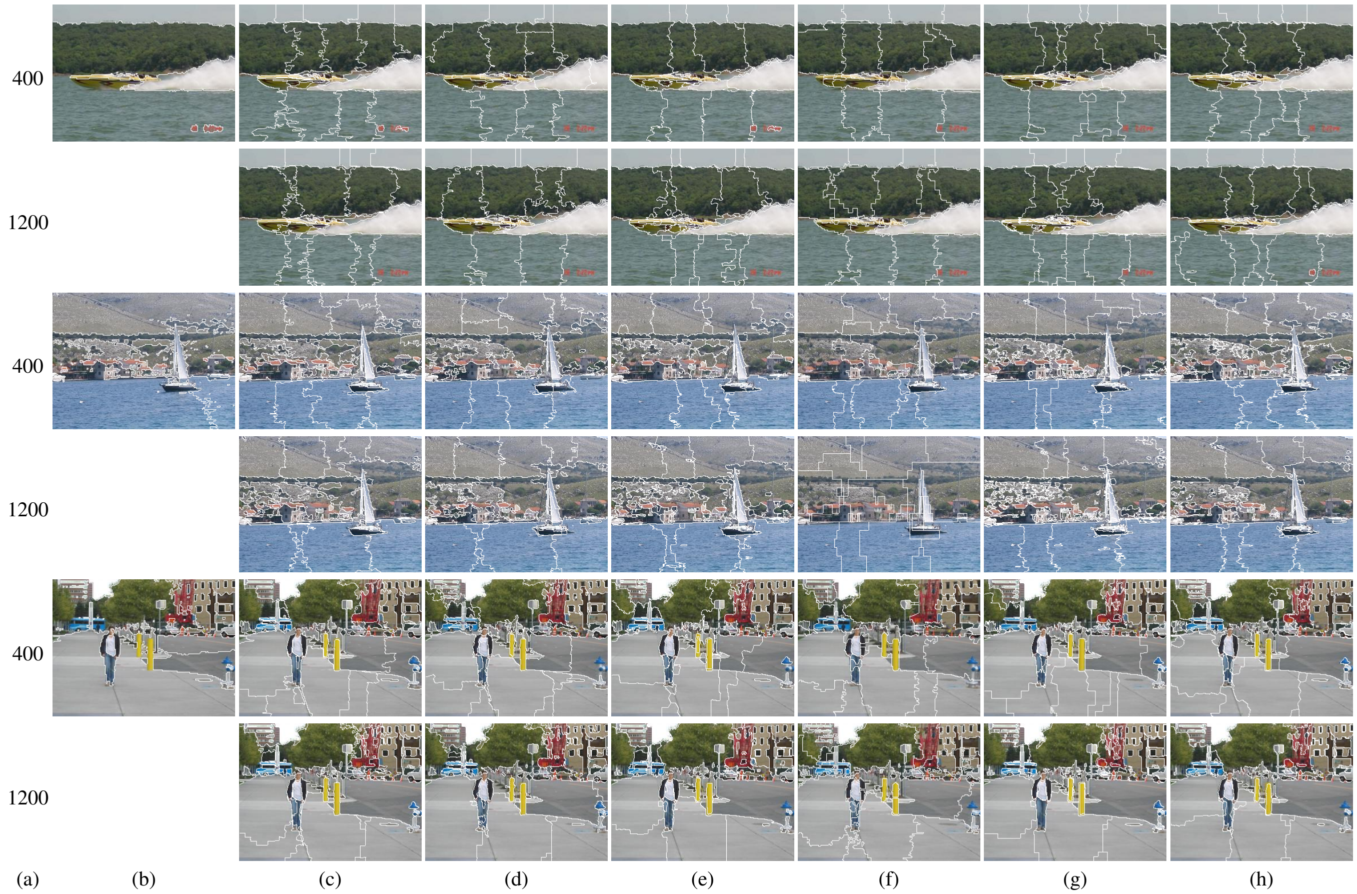


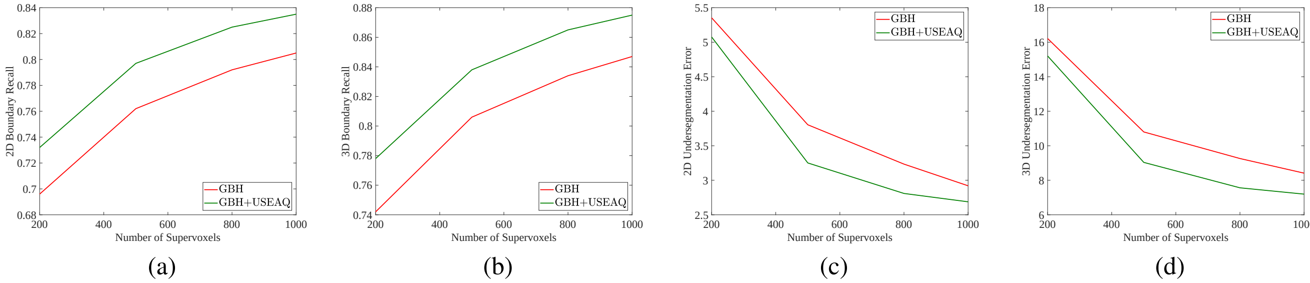
Fig. 16. (a) The number of superpixels generated before applying mean shift for image segmentation. (b) Segmentation results by applying the mean shift algorithm to image pixels directly. (c) ∼ (h) Segmentation results by applying the mean shift algorithm to superpixels generated by (c) USEAQ, (d) USEQ, (e) SLIC, (f) SEEDS, (g) ETPS, and (h) LSC, respectively.

A. Image Segmentation

Decomposing an image into superpixels is considered an efficient way for image segmentation. We apply mean shift (MS) [30] to merge pixels as well as superpixels generated by different algorithms, including USEAQ, USEQ, SLIC, SEEDS, ETPS and LSC, on the SBD dataset for image segmentation. The mean shift algorithm takes into account both the color and spatial features extracted in each pixel/superpixel, and accomplishes image segmentation. Before applying mean shift, each evaluated method generates 400 and 1200 superpixels, respectively. After applying mean shift to the generated superpixels by a method for superpixel extraction, an image is partitioned into P merged superpixels. The image is composed of Q regions in ground truth with $P \geq Q$. We perform bipartite matching between the merged superpixels and the image regions in ground truth. For each region q, let k_q be the number of pixels of the corresponding merged superpixel falling in region q and t_q be the number of pixels of region q. The performance is measured by $\frac{\sum_{q=1}^{Q} k_q}{\sum_{q=1}^{Q} t_q}$. The bipartite matching for maximizing the performance can be established by using the Hungarian algorithm. After bipartite matching, the used performance measure is the *pixel accuracy* (PA) adopted in [47]. TABLE II shows the performance by applying mean shift to pixels directly and to superpixels generated by different methods. The results show that the proposed USEAQ can produce appropriate superpixels and lead to higher pixel accuracy. In addition, a larger number of superpixels helps improve the pixel accuracy. It is worth mentioning that applying mean shift directly to pixels gives lower pixel accuracy (32.82%) than applying it to the generated superpixels. This phenomenon has been pointed out in [9]. Fig. 16 displays the segmentation results by applying mean shift to image pixels and superpixels yielded by different methods. Mean shift merges neighboring pixels/superpixels to accomplish image segmentation. The proposed USEAQ shows better qualitative results on the highly textured regions since it achieves better boundary adherence of objects.

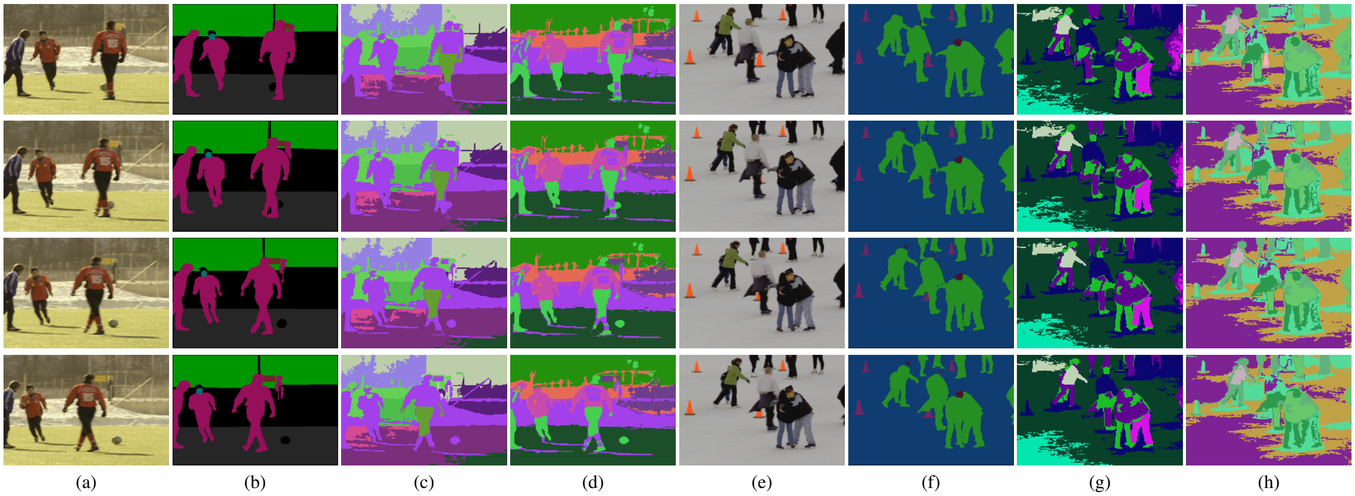
B. Supervoxel Construction

Supervoxels [48], [49] are the spatiotemporal extension of superpixels and have been proven to be an effective representation for video analysis. To extend USEAQ to supervoxel extraction, we modify the graph-based hierarchical (GBH) [48] approach, which is a superior approach to supervoxel extraction on the supervoxel benchmark [49]. GBH over-segments a video into small spatiotemporal regions by using the method in [22]. Then, a hierarchical segmentation scheme is applied to construct a region graph for supervoxel generation. In our implementation, we replace the method in [22] by USEAQ to



(a) (b) (c) (d)

Fig. 17. Comparison of GBH and GBH+USEAQ in (a) 2-D boundary recall, (b) 3-D boundary recall, (c) 2-D undersegmentation error, and (d) 3-D undersegmentation error.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 18. Extracted supervoxels on two videos. (a) & (e) Four frames of each video. (b) & (f) The ground truth. (c) & (g) The supervoxels extracted by GBH. (d) & (h) The extracted supervoxels extracted by GBH+USEAQ.

perform the over-segmentation in GBH. The resultant method is denoted by GBH+USEAQ, and is evaluated on the Xiph.org dataset [49].

Fig. 17(a) and 17(b) show the comparison of GBH and GBH+USEAQ in 2-D and 3-D boundary recalls with respect to different numbers of supervoxels, respectively. As shown in the results, GBH+USEAQ can better preserve both the 2-D and 3-D boundaries. Fig. 17(c) and 17(d) report the 2-D and 3-D undersegmentation errors, respectively. GBH+USEAQ also achieves lower undersegmentation errors. The results point out that USEAQ helps reduce the undersegmentation errors in both 2-D and 3-D cases.

Fig. 18 visualizes the extracted supervoxels on two videos by showing the video frames, the ground truth, and the extracted supervoxels by GBH and GBH+USEAQ. In Fig. 18(c), GBH separates the grass of the soccer field and the sky into several small regions, and cannot distinguish the soccer players from the backgrounds. In contrast, GBH+USEAQ does not separate the regions of the grass and the sky into several supervoxels, and successfully separates the players from the backgrounds as shown in Fig. 18(d). We attribute this nice property of USEAQ to its adaptive sampling from quantized regions. As revealed in Fig. 18(g) and 18(h), GBH+USEAQ can better segment foreground objects including the people and traffic cones. Both the quantitative and qualitative results demonstrate the effectiveness of USEAQ in the application of supervoxel construction.

VI. Conclusions

We have presented a novel approach to superpixel extraction. It firstly performs joint spatial and visual quantizations, and employs an adaptive sampling algorithm for locating superpixel candidates on the quantized image. The maximum a posteriori estimation is applied to carrying out pixel-superpixel assignment. The resultant superpixels are attained after a refinement process. The complexity of our approach grows linearly to the number of pixels and is irrelevant to the number of the extracted superpixels. These nice properties distinguish this work from most existing approaches. Compared with the existing superpixel extraction approaches with their complexity linear to the pixel number, our approach still has the advantage of remarkably higher efficiency in running time, since it compiles superpixels in a one-pass fashion and avoids less efficient iterative procedures.

Our approach is comprehensively evaluated and compared with the state-of-the-art approaches on the benchmark BSDS500. For accuracy, our approach achieves the performance comparable with the state-of-the-art approaches in terms of boundary recall, undersegmentation error, and achievable segmentation accuracy. For efficiency, it gives highly efficient running time in all cases of different resolutions and various desired numbers of superpixels. In addition to BSDS500, the SBD dataset is also used for evaluation. Our approach achieves the superior or comparable performance in terms of the average miss rate, average undersegmentation error, average unexplained variation, and running time. In addition to the quantitative results, we also show and compare the generated superpixels by different approaches, where it can be observed that our approach can adaptively control the trade-off between superpixel regularization and boundary adherence conditioned on the image regions. Namely, regular superpixels are produced in homogeneous or flat regions, while superpixels can still adhere to the boundaries in the highly textured regions. We also demonstrate that the proposed USEAQ facilitates two related applications, image segmentation and supervoxel construction. In the future, we plan to extend this framework to applications where an effective and efficient algorithm for superpixel or supervoxel extraction is appreciated.

References

[1] A. Bodis-Szomoru, H. Riemenschneider, and L. V. Gool, "Superpixel meshes for fast edge-preserving surface reconstruction," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2015. 1

[2] D. Giordano, F. Murabito, S. Palazzo, and C. Spampinato, "Superpixel-based video object segmentation using perceptual organization and location prior," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2015. 1

[3] C. Spampinato, S. Palazzo, and D. Giordano, "Gamifying video object segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 791–801, 2016. 1

[4] F. Yang, H. Lu, and M.-H. Yang, "Robust superpixel tracking," *IEEE Trans. on Image Processing*, vol. 23, no. 4, pp. 1639–1651, 2014. 1

[5] Y. Yuan, J. Fang, and Q. Wang, "Robust superpixel tracking via depth fusion," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 24, no. 1, pp. 15–26, 2014. 1

[6] Z. Liu, X. Zhang, S. Luo, and O. Meur, "Superpixel-based spatiotemporal saliency detection," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 24, no. 9, pp. 1522–1540, 2014. 1

[7] H. Lu, X. Li, L. Zhang, X. Ruan, and M.-H. Yang, "Dense and sparse reconstruction error based saliency descriptor," *IEEE Trans. on Image Processing*, vol. 25, no. 4, pp. 1592–1603, 2016. 1

[8] L. Zhu, D. Klein, S. Frintrop, Z. Cao, and A. Cremers, "A multisize superpixel approach for salient object detection based on multivariate normal distribution estimation," *IEEE Trans. on Image Processing*, vol. 23, no. 12, pp. 5094–5107, 2014. 1

[9] Z. Li, X.-M. Wu, and S.-F. Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2012. 1, 13

[10] Z. Tian, L. Liu, Z. Zhang, and B. Fei, "Superpixel-based segmentation for 3d prostate mr images," *IEEE Trans. on Medical Imaging*, vol. 35, no. 3, pp. 791–801, 2016. 1

[11] X. Wang, Y. Tang, S. Masnou, and L. Chen, "A global/local affinity graph for image segmentation," *IEEE Trans. on Image Processing*, vol. 24, no. 4, pp. 1399–1411, 2015. 1

[12] C. Wang, Z. Liu, and S.-C. Chan, "Superpixel-based hand gesture recognition with kinect depth camera," *IEEE Trans. on Multimedia*, vol. 17, no. 1, pp. 29–39, 2015. 1

[13] H. Liang, J. Yuan, and D. Thalmann, "Parsing the hand in depth images," *IEEE Trans. on Multimedia*, vol. 16, no. 5, pp. 1241–1253, 2014. 1

[14] L. Zhang, Y. Yang, M. Wang, R. Hong, L. Nie, and X. Li, "Detecting densely distributed graph patterns for fine-grained image categorization," *IEEE Trans. on Image Processing*, vol. 25, no. 2, pp. 553–565, 2016. 1

[15] Z. Ren, S. Gao, L.-T. Chia, and I. W.-H. Tsang, "Region-based saliency detection and its application in object recognition," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 24, no. 5, pp. 769–779, 2014. 1

[16] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012. 1, 3, 4, 7, 9

[17] C.-R. Huang, W.-A. Wang, S.-Y. Lin, and Y.-Y. Lin, "USEQ: Ultra-Fast superpixel extraction via quantization," in *Proc. Int'l Conf. Pattern Recognition*, 2016. 1, 4, 8, 9

[18] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 898–916, 2011. 2, 7

[19] S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions," in *Proc. Int'l Conf. Computer Vision*, 2009, pp. 1–8. 2, 8, 10

[20] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000. 2

[21] G. Mori, "Guiding model search using segmentation," in *Proc. Int'l Conf. Computer Vision*, 2005. 2

[22] P. Felzenszwalb and D. Huttenlocher, "Efficient graph based image segmentation," *Int. J. Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004. 2, 9, 13

[23] A. Moore, S. Prince, J. Warrell, U. Mohammed, and G. Jones, "Superpixel lattices," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2008. 2

[24] A. Moore, S. Prince, and J. Warrell, "Lattice Cut - Constructing superpixels using layer constraints," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2010. 2

[25] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa, "Entropy rate superpixel segmentation," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2011. 2, 7

[26] O. Veksler, Y. Boykov, and P. Mehrani, "Superpixels and supervoxels in an energy optimization framework," in *Proc. Euro. Conf. Computer Vision*, 2010. 2

[27] Y. Zhang, R. Hartley, J. Mashford, and S. Burn, "Superpixels via pseudo-boolean optimization," in *Proc. Int'l Conf. Computer Vision*, 2011. 2

[28] J. Peng, J. Shen, A. Yao, and X. Li, "Superpixel optimization using higher order energy," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 26, no. 5, pp. 917–927, 2016. 2

[29] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991. 2

[30] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002. 2, 13

[31] A. Vedaldi and S. Soatto, "Quick shift and kernel methods for mode seeking," in *Proc. Euro. Conf. Computer Vision*, 2008. 2

[32] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi, "TurboPixels: Fast superpixels using geometric flows," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, 2009. 2, 9

[33] G. Zeng, P. Wang, J. Wang, R. Gan, and H. Zha, "Structure sensitive superpixels via geodesic distance," in *Proc. Int'l Conf. Computer Vision*, 2011. 2

[34] Y.-J. Liu, C.-C. Yu, M.-J. Yu, and Y. He, "Manifold SLIC: A fast method to compute content-sensitive superpixels," in *Proc. Conf. Computer Vision and Pattern Recognition*, 2016. 3, 9

[35] R. Achanta and S. Ssstrunk, "Superpixels and polygons using simple non-iterative clustering," in *Proc. Conf. Computer Vision and Pattern Recognition*, July 2017, pp. 4895–4904. 3, 9

[36] M. V. den Bergh, X. Boix, G. Roig, B. de Capitani, and L. V. Gool, "SEEDS: Superpixels extracted via energy-driven sampling," in *Proc. Euro. Conf. Computer Vision*, 2012. 3, 9

[37] J. Yao, M. Boben, S. Fidler, and R. Urtasun, "Real-time coarse-to-fine topologically preserving segmentation," in *Proc. Conf. Computer Vision and Pattern Recognition*, Jun. 2015, pp. 2947–2955. 3, 9

[38] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. on Image Processing*, vol. 23, no. 4, pp. 1451–1462, 2014. 3

[39] H. Fu, X. Cao, D. Tang, Y. Han, and D. Xu, "Regularity preserved superpixels and supervoxels," *IEEE Trans. on Multimedia*, vol. 16, no. 4, pp. 1165–1175, 2014. 3, 9

[40] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. on Image Processing*, vol. 25, no. 12, pp. 5933–5942, 2016. 3, 9

[41] Z. Hu, Q. Zou, and Q. Li, "Watershed superpixel," in *Proc. Int'l Conf. Image Processing*, Sept. 2015, pp. 349–353. 3, 9

[42] Z. Li and J. Chen, “Superpixel segmentation using linear spectral clustering,” in *Proc. Conf. Computer Vision and Pattern Recognition*, 2015. 3, 9

[43] J. Chen, Z. Li, and B. Huang, “Linear spectral clustering superpixel,” *IEEE Trans. on Image Processing*, vol. 12, no. 10, pp. 904–908, 2017. 3

[44] X. Chen, S. Kwong, and J.-F. Feng, “A new compression scheme for color-quantized images,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 12, no. 10, pp. 904–908, 2002. 4

[45] M. Orchard and C. Bouman, “Color quantization of images,” *IEEE Trans. on Signal Processing*, vol. 39, no. 12, pp. 2677–2690, 1991. 4

[46] D. Stutz, A. Hermans, and B. Leibe, “Superpixels: An evaluation of the state-of-the-art,” *Computer Vision and Image Understanding*, vol. 166, pp. 1–27, 2018. 8, 10

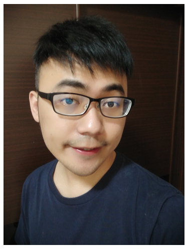
[47] E. Shelhamer, J. Long, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, April 2017. 13

[48] M. Grundmann, V. Kwatra, M. Han, and I. Essa, “Efficient hierarchical graph-based video segmentation,” in *Proc. Conf. Computer Vision and Pattern Recognition*, Jun. 2010, pp. 2141–2148. 13

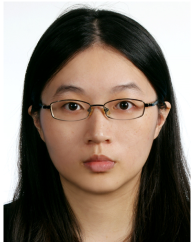
[49] C. Xu and J. J. Corso, “Libsvx: A supervoxel library and benchmark for early video processing,” *Int. J. Computer Vision*, vol. 119, no. 3, pp. 272–290, 2016. 13, 14

Chun-Rong Huang (M’05) received the B.S. and Ph.D. degrees in the Department of Electrical Engineering from National Cheng Kung University, Tainan, Taiwan, in 1999 and 2005, respectively. In 2005, he joined the Institute of Information Science, Academia Sinica, Taipei, Taiwan, as a Postdoctoral Fellow. He becomes an Assistant Professor and Associate Professor with the Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan, in 2010 and 2015, respectively. His research interests include computer vision, computer graphic, multimedia signal processing, image processing, and medical image processing. Dr. Huang is a member of the IEEE Circuits and Systems Society, the IEEE Signal Processing Society, IEEE Computational Intelligence Society, and the Phi Tau Phi Honor Society.

Wei-Cheng Wang received the B.S. and M.S. degrees from the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, in 2014 and 2016, respectively. In 2018, he joined the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan, as a research assistant. His research interests include computer vision, machine learning, visual analysis, visual surveillance and deep learning. Wei-Cheng Wang is a member of the Phi Tau Phi Scholastic Honor Society of the Republic of China.



Wei-An Wang received the B.S. and M.S. degrees in the Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan, in 2013 and 2016, respectively.



Szu-Yu Lin received the B.S. and M.S. degrees in the Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan, in 2014 and 2016, respectively.

Yen-Yu Lin (M’12) received the B.B.A. degree in information management, and the M.S. and Ph.D. degrees in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 2001, 2003, and 2010, respectively. He is currently an Associate Research Fellow with the Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. His current research interests include computer vision, machine learning, and artificial intelligence. He is a member of IEEE.