Image Segmentation Methods Based on Superpixel Techniques: A Survey

Abdelhameed Ibrahim
Faculty of Engineering, Mansoura University, Mansoura, Egypt
Email: afai79@mans.edu.eg

El-Sayed M. El-kenawy
Delta Higher Institute of Engineering and Technology (DHIET), Mansoura, Egypt
Email: skenawy@ieee.org

Abstract—There is a growing demand for image processing in a wide range of applications such as photography, robotics, television, remote sensing, industrial inspection, and medical diagnosis. This study overviews some of the existing image segmentation methods that focus on producing superpixels. A superpixel or segment is a homogeneous, local coherent structure that specifies information oversampling or scales resolutions. There are many image segmentation or superpixelization methods which divide color image with different techniques according to their characteristics and parameters as image acquisition might be seriously affected by many factors such as light and shadow. Several image segmentation algorithms were investigated in image processing research for creating superpixels that may lack the ability to control the size, number, and compactness of segments. Superpixel generation algorithms can be categorized into graph-based methods and gradient-ascent based approaches.

Index Terms—Digital image processing, image segmentation, image superpixels, graph-based methods, gradient-ascent based methods.

I INTRODUCTION

Image processing is one field of signal processing, in which the input is an image such as a video frame or photograph and the output is an image that contains a set of parameters and characteristics related to the image. Image processing may be optical, analog, or digital; but image processing refers to digital image processing usually. There is a growing need for more efficient image processing techniques that can be applied to different applications, such as automatic visual inspection [1], image segmentation [2, 3], moving-object tracking, secured image data communication, multimedia computing [4, 5], content-based image retrieval, biomedical imaging [6, 7], remote sensing, pattern recognition [8, 9, 10, 11], biometrics, texture understanding, image and video compression, etc. [12].

Image segmentation is the process of partitioning an image into groups of linked regions that are homogeneous and non-overlapping (group of pixels, also referred to as superpixels) [13, 14]. In each region, there are constitutive parts or objects in the entire image scene. The resulted objects simplify exemplification of the image to things that are significant and easier for analyze; and locating boundary and object like curves, lines, etc. The superpixel is referred to a perceptually uniform region in the image polygonal part of a digital image larger than a normal pixel and rendered in the same brightness and color. One of main feature of using superpixels is the performance of computations [15, 16]. Superpixel impersonation decreases the number of image dependencies compared to pixel impersonation; and supply the locative support for determining regions depending on image features [17, 18].

Factors of illumination like specular highlight, shading and shadow spotted from surface of objects in natural-scene; have serious effect on image appearance and analysis. There are several ways [2, 3, 8, 9, 10] that propose image invariant representation which have an effective role on many applications such as object recognition, corner and edge detections, feature detection, and image recuperation. Superpixel [19, 20, 21] is generally recognized as a uniform image portion; therefore, removing the highlight and shadow makes both image properties and features more efficient. The superpixel representation decreases the number of image independences in contrast to pixel representation. In addition, superpixel segmentation introduces an excellent support to compute regions that depend on image features whilechang-
ing the representation of the image to a more significant and easier one for analyses [22]. In many applications, optimization techniques, such as [17, 23, 24, 25, 26], can improve the segmentation process.

Despite the presence of several image segmentation algorithms; the majority of them were not designed to generate superpixels and can also lack the ability to control the size, number, and compactness of segments. Quick-shift is a prevalent approach for image segmentation that produces superpixels which are not fixed in number or size, unlike other superpixelization schemes that rely on normalized cuts (e.g. [6, 21]). A compound image with many small portions may have a larger number of superpixels than a simpler one. Superpixels that have different shapes and sizes can generate segments containing a lot of small portions that maintain a large number of the original image boundaries.

II SUPERPIXEL METHODS

This section will survey image segmentation methods and will concentrate on the suitable methods for implementing superpixels. Superpixel generation algorithms can be categorized into graph based methods and gradient-ascent based methods.

A Graph-Based Methods

In graph-based methods, every pixel is handled as a node in the graph, while similarity between neighboring pixels is proportional to edge weights between the two nodes. By minimizing the cost function, superpixels are created and defined over the graph. The graph-based algorithms include Efficient graph-based image segmentation [27], Superpixels and supervoxels in an energy optimization framework [28], Normalized cuts and image segmentation [29], Graphcut textures [30], Entropy Rate Superpixels (ERS) [28], Lazy Random Walks Superpixel (LRW) [31], and other state-of-the-art methods.

Normalized Cuts Algorithm: The algorithm discussed in [32] considered a class of segmentation methods which depends on finding minimum cuts on the graph, where the similarity between split pixels is minimized by the cut criterion. Leaky and WI. [8] developed a cut criterion to find small compounds. The normalized cut criterion developed by Malik and Shi [32] is addressed with a previous bias, that took into account regions self-similarity. In contrast, with the old graph-based methods, these cut-based approaches capture the non-local properties of the image, however, they provide characterization for each cut and not for the final segmentation.

The criterion of normalized cut is better than the previous method discussed in [8], from both the practical and theoretical point of view; the resulting segmentations capture parts of the image locally. So, normalized cut criterion method also has the NP-hard computational problem. The approximation methods for computing the minimum normalized cut are developed by Malek and Shi [32], in their method, the error is not well understood. In practice, these approximations still hard to compute, and require computation time of several minutes which can limit the method to relatively small images. Weiss [9] had shown the eigenvector-based approximations that introduced by Malek and Shi biased toward graphs standard spectral partitioning methods. So, most of these methods are considered relatively too slow in many applications.

Looking for graph iteration that involves the image plane is considered another approach that can be an alternative to the graph cut approach. For example in [10] each cycle quality was normalized in a way which closely having a relation with the normalized cuts approach.

Efficient Graph-Based Segmentation: A graph is a structure that models couple relations between objects from a specific collection, though the collection of (nodes) or (vertices) and a collection of edges which connect couples of vertices. The edges can be directed or undirected and in the undirected edges, there is no difference between vertices that have a relation with each edge.

In the case of image segmentation, element is a pixel and the weight of the edge is the measure of the difference between the two pixels that are linked by this edge. The dissimilarity in color, location, intensity, motion or some other local-attribute. As with certain classical clustering methods [3, 4], Efficient graph-based segmentation method depends on selecting edges from a graph, where every node or vertex in the graph corresponds to a pixel, undirected edges [27] connect certain neighboring pixels.

Felzenszwalb et al. [27] introduced an algorithm to segment images depending on pair-wise region comparison and show which target a segment is too fine or too coarse. They defined a function that measures the evidence of the boundary between pairs of regions. This segmentation method makes simple avid decisions, and still produces segmentations that justify the global properties for neither being too fine or too coarse according to the particular region comparison function. The method runs in $O(m\log m)$ time for $m$ graph edges. The comparison between pair-wise region predicts the minimum weight of the edge between two regions for measuring the differences between them. So, this segmentation method will merge two regions even if there is only one low weight edge between them.

If we consider an example of segmentation which capture many perceptually important aspects of complex images. Before to make a decision when there is no guide for the bound-
ary between two regions, one can imagine the need for measures that require more than single connection. Using a quantile is one of the natural ways of addressing this issue instead of the minimum edge weight. It is a hard problem to find a segmentation that is neither too fine nor too coarse in this case.

This algorithm captures the Non-local properties of images; and it illustrates that image segmentation method have two different types of graphs. One of these graphs uses an image grid for having a definition for local neighborhood between the image pixels, and measuring the difference in intensity or the difference in color between each pairs of neighbors pixels; and others of these graphs - in the feature space - maps the image pixels to points which combines the color value \((r,g,b)\) and the location \((x,y)\). In this feature space, points that are close together are connected edges in the graph. This algorithm gives good results by using two kinds of graphs, and other graph type captures the more perceptual aspects of the image.

Image segmentations still a huge problem for making substantial progress by the graph-based algorithms introduced, that both provide useful computational tools and help to refine the understanding of the problem considered at the beginning. The normalized cuts approaches and segmentation method reported in [27] are just few illustrations for these recent advances.

**Compact Superpixels**: Compact Superpixels (CS) [29] proposed that the input image is mostly covered by patches of half-overlapping squares with the same size. Each one of these square patches conforms to a label. Thus, many labels were generated. The method assigns each pixel to a unique patch by optimizing an energy function that has smoothing terms and data terms. the smoothing terms control the boundary of the patch, and The data terms decide tp which patch each pixel belong. It uses Potts model [33] to approximate the Euclidean distance between pixels. The upper bound of superpixels estimates the patch size. In this approach, boundaries are encouraged to be compact by the energy function while superpixel sizes tend to be equalized.

**Constant Intensity Superpixels**: Constant Intensity Superpixels (CIS) is an approach used to generate constant intensity superpixels proposed by Veksler et al. [29]. CIS encourages the use of constant intensity inside a superpixel on the expense of the price of relaxing the need for superpixels to have roughly equal sizes. At the beginning, it assigns the pixel color at the center to each pixel. One of the main requirements of CIS is for the superpixel to have constant intensity using constraint object function. Although, CS is faster than CIS, CIS can provide color coherent superpixels.

**Pseudo-Boolean Superpixels**: Pseudo-Boolean (PB) [34] treats superpixel segmentation as a problem of multi-label assignment. The input image is aligned by half-overlapping horizontal strips. Every image pixels can be assigned to one of two different strips. Afterwards, a decision of which pixels belong to which strip. That could be arranged as a binary labeling problem on Markov Random Fields (MRF). The predefined objective function consists of two Pseudo-Boolean functions that can be optimized by the elimination algorithm [30]. The algorithm generates superpixels that are regular in both shape and size. The speed of PB is not related to the number of superpixels, which is considered the weakest point in the traditional superpixel segmentation algorithms.

**Entropy Rate Superpixel**: The Entropy rate superpixel (ERS) [28] presented another objective function for superpixel segmentation. It combines random walk entropy on an image graph representation with a balancing term on the segment size. In the researchers derivation, a high entropy leads to homogeneous segments. The entropy rate is defined as the remaining uncertainty asymptotic measure of the random process resulted from adding edges to make clusters. They showed that the objective function is sub-modular and approximates the optimal solution by a bound of 1/2. Their algorithm started with an empty set of edges then added the edge that yielded the largest gain of their objective function. They repeated these steps until the right number of clusters is achieved. To increase the algorithm speed, they ignored the edges which might introduce cycles. That step manged to reduce the solution space to a set of forests. Therefore, this method computed segments hierarchy.

**Homogeneous Superpixels**: The main principle of Homogeneous Superpixels (HS) [35] resembles a graph based algorithm utilizing Markov Clustering. Based on a Markov graph (MCL) [27], that is an undirected, weighted graph where all edges of a given vertex are positive and sum to one, the algorithm alternates an expansion and an inflation step which are carried out on the corresponding weight matrix. To reduce computational time of the MCL and avoid inhomogeneous pixels, compact pruning [23] was proposed. To complexity of HS could be estimated to be \(Nnr^4\), where \(r\) is the pruning radius and \(N\) is the number of pixels.

**Linear Spectral Clustering**: Linear Spectral Clustering (LSC) [36], using a kernel function, transfers image pixels to weighted points in 10-D feature space. Next, the seed pixels are sampled uniformly for the entire image. They are then used as the search centers while their vectors of features are used as the initial weighted means of the corresponding clusters. Every pixel is assigned to the cluster whose weighted mean is the closest to the pixels’ vector in the feature space. Both the weighted mean and the search center of each cluster will then be updated. The previous steps need to be performed repeatedly until all the cluster centers are stabilized.
in an iterative manner. LSC keeps the image global properties and its superpixels providing a more regular approach than SLIC.

**Lazy Random Walk Superpixels**: The Lazy Random Walk (LRW) [31] is an extension to the RW algorithm. An image is mapped to a graph whose vertices are pixels in the input image while the edges are calculated according to a Gaussian weighting function. LRW initializes seeds on the graph with a strategy similar the one described in [28]. After the initialization stage, the initial superpixels are repeatedly optimized using the new energy function, which is dependent on the iterative time while the texture measurement LRW conform well to the object boundaries.

**B Gradient-Ascent-Based Methods**

Begging with a primary grouping of pixels, the gradient ascent method repeat refine the clusters till some convergence criterion are match to create superpixels. Some of the famous gradient-ascent algorithms include quick-shift and kernel methods for mode seeking [37], turbopixels method which is fast superpixels using geometric flows [10], mean-shift which is a robust approach toward feature space analysis [37], watersheds in digital spaces which is considered as an efficient algorithm based on immersion simulations [38, 39, 40, 41] and a simple linear iterative clustering (SLIC) and g-SLIC algorithms which addressing a lot of requirements and outperforming other state-of-the-art algorithms. All these methods have their own pros and cons depending on the application requirement, but there are certain qualitative and quantitative measures defining how efficient these algorithms perform. Importantly, superpixels should define the object contours rapidly and consume less memory. This section introduces the state-of-the-art gradient-ascent based methods in details.

**Lattice Cut**: Lattice Cut [42] starts from a regular grid, then the algorithm merges the superpixels iteratively. It links each pixel with a label to show to which superpixel it belongs to. The problem can be formulated by assigning the unknown labels in a MRF model. Graph-cuts is usually used to find the MAP (Maximum a Posteriori) solution. Lattice-cut can use both region and edge information to produce a globally optimal solution in a vertical or horizontal direction using a single graph cut.

**Watershed approach**: The main idea of watershed segmentation originates from [38, 39]. The originating idea came from geography: when a landscape is flooded by falling rain, some basins will be formed filled with water that is dependent on the amount of fallen water drops with dividing ridges. These ridges are the watersheds. Since watershed segmentation is a widely used algorithm, there are several implementations and adaptations for the algorithm. The watershed algorithm performs a gradient ascent which starts from a local minima to produce watershed lines separating catchment basins. Although it is considered a fast approach, it does not have control over the amount of superpixels or how compact they are. The superpixels are irregular in shape and size and do not support the boundary adherence of images. The approach of [41] is relatively fast with complexity of $O(N \log N)$, but does not provide control over the amount of superpixels or how compact they are. On the other hand, Compact watershed [40] is designed as a modified implementation which considered the compactness constrains of superpixels that can control the number of superpixels directly. In our evaluation, we use the implementation of compact watershed.

**Edge augmented mean shift**: Edge augmented mean shift (EAMS) [43] introduced a new approach to detect weak edges. In this approach obvious artifacts of the gradient-based edge detectors such as large spurious responses can be quantized. It is also shown that template matching a template that is derived from input data is meaningful since it provides a confidence measure in the presence of the edge model. The commonly used three steps edge detection procedure is generalized to utilize the information indicated by the confidence measure with minimum amount of computation.

**Mean-Shift**: Mean-Shift [44], is an approach that depends on an iterative mode-seeking procedure aiming to locate local maximum of a certain density function, that is applied in order to find modes in an image intensity or color feature space. Pixels that can converge to the same mode represent the superpixels. Mean-Shift is a classical image segmentation approach that can be used to get superpixels. It creates irregular shapes of superpixels. However, the complexity of that procedure is $O(N^2)$, making it relatively slow, and without direct control over the superpixels size, amount, or compactness.

**Superpixel Lattice**: It is important to remove any natural ambiguity from the image to be easier for segmentation. The small blue segment in the image can be because of a variety classes such as sky, a car, clothes, water, door or any other object that has a blue color. However, it is shown that when found some mountains, trees or similar segments then sky will be correctly classified. To try to solve these confusions, image parsing algorithms integrates detection, segmentation and recognition.

Malik and Ren proposed a preprocessing stage or step in that pixels are segmented into superpixels to reduce nodes number in the image graph [45, 46]. Normalized cut criterion used in their method to recursively partition an image using contours and textures cues and have been used in the preprocessing step for modern image parsing schemas. Other possible superpixel methods involve and use watershed algo-
algorithm that demonstrated for reducing the number of nodes.

These superpixel method lose many useful properties in original pixel-representation process. First, without the need for pointers, pixels can be re-presented in arrays. Second, it is easy to use multi-scale methods and sub-sample pixels uniformly. Third, the nth pixel have a position, ordered, consistent in that image. Fourth, there is a relationship between the rst pixel and the \( (n - 1) \)th pixel which allowing local neighborhood processes. Fifth, mapping from pixel to pixel in different images is unique and the image pixel representations for the same dimension \( (row \times col) \) are symetric. These properties are ignored by modern superpixel algorithms and are the result of the organized topology for the original grid graph of pixels.

A regular grid of superpixels guaranteed to be produced by superpixel lattice, in which the input of this algorithm are boundaries map. These map are 2D arrayes which represents the boundary existing between two pixels. SL inverts and re-scales the map of boundary for taking a value of 1 where there is no evidence and 0 where there is the most evidence for a boundary. SL goals segment images in places that have low boundary cost, which has done by find the lowest weighted paths in the graph.

Superpixel-lattice have increment construction; it divide an image horizontally and vertically as initial step, then divide the image into two by each path to accumulatively production four superpixels. In each step they add an extra horizontal and vertical paths. If these points are ensured that (i) no two horizontal paths cross and (ii) no two vertical paths cross (iii) each vertical path and horizontal path only cross once, a regular lattice is guaranteed

The final superpixel lattice is controlled by three important parameters. First, the overall number of superpixels that determined by resolution and it is indirectly referring to the number of paths. Second, the chosen path that constrained by the width of each strip in image. The resolution and the width of every image strip together determine the overlap of the strips. finally, the degree at which curve deviates from a straight-line determined by the twisting in the path.

Moore et al. proposed this method to produce superpixels that can be adapted to be in a grid by finding the optimal paths separating the image into a number of smaller regions. Ideal strips are then determined using a graph cuts algorithm. While the complexity for \( SL08isO(N(32) \log N) \), this does not produce boundary maps, which can consequently impact the image output speed and quality.

**Quick-Shift (QS):** Quick Shift is a segmentation algorithm that segments a color image (each image have many components) by identifying clusters of pixels in the common spatial and color dimension [37]. It is a mode searching approach, similar to mean-shift [44]. The segments produced by QS method are local in nature and may be used as a base for further processing. Segments that are produced locally is called superpixel. The QS [37] used a mode-seek segmentation method and was initialized using the medoid-shift algorithm, then each point was moved to the nearest neighborhood. The QS segmentation algorithm belongs to the family of local mode-seeking algorithms and is utilized to the 5D-space which consists of color information and image location. Superpixels that are generated by Quick-Shift method are not fixed in size and/or number and have good boundary adherence. The QS used by previous work for motion segmentation and object localization [47] where the algorithm parameters are adjusted manually. In QS method, generating superpixels are controlled by three parameters of:

- **Kernel Size –** Scale at which density is calculated acts as a density estimator of Gaussian kernel to smooth the sample density. For every pixel \( (x, y) \), quick-shift regards \( (x, y, l(x, y)) \) as an example from a \( d + 2 \) dimensional vector space. It then calculates Parzen density estimate (with a Gaussian kernel of standard deviation ) followed by a quick shift build of a tree that links each image pixel to its nearest neighbor that has greater density value.

- **Distance –** the maximum distance in the feature space between nodes in the Quick-Shift tree is used to cut links to form the segmentation with higher means fewer clusters. Each pixel \( (x, y) \) is linked to the closest higher density pixel parent \( (x, y) \) that achieves the minimum distance. The approach calculates the forest of pixels whose branches are labeled with a distance value. This specifies the hierarchical segmentation of the image, with segments formed according to sub-trees. Useful superpixels can be specified by cutting the branches whose distance labels are above a given threshold (the threshold can be either determined by cross validation, or just adjusted by hand).

- **Ratio -** a trade-off must be made between spatial importance \( (X, Y, R, G, B) \) and color importance \( (R, G, B) \), as the algorithm must balance both color-space and image-space proximity with the need to have a value from 0 to 1; A Small ratio gives a higher importance to the spatial component which in turn gives more weight to color-space.

**g-SLIC:** gSLIC11 – depends on Simple Linear Iterative Clustering (SLIC) superpixel segmentation method by using a framework for NVIDIA CUDA with a graphics processor unit and parallel implementation for SLIC. Visual processor unit (VPU) or graphics processor unit (GPU) is a technique used in image processing and computer graphics to speed up
the processing of images and visual data by using a parallel processing approach. This method reduces the execution time from 20t to 10t compared with the sequential implementation and consequently allows for using the superpixel segmentation method in real-time problems.

**TurboPixel:** The superpixel algorithm of [48] offers a balance between the goals of reducing the complexity of an image through pixel grouping while trying to avoid under-segmentation, which leads to a form restriction for region segmentation. That algorithm has been used in applications that try to classify, label or segment images from labeled training datasets. The computational cost for the processes of underlying grouping, whether combinatorial or probabilistic, is eliminated during converting the pixel graph to a superpixel graph. Superpixels algorithms need to achieve over-segmenting the image; as it is easier to merge superpixels rather than to split them. Region segmentation approaches can lose some form of compactness which lead to under-segmentation boundary cues of the image, e.g., mean-shift, local variation, or watershed. This can happen when there are weak contrast or shadows. Figure 2.8 illustrates the over-segmentations produced when using these three different methods; the results presented by TurboPixels and normalized Cuts show the compactness affective constraint in reducing under-segmentation.

Superpixels [48] is a restricted graph cut method, constrained for yielding a large number of quasi-uniform, compact, and small regions. Graph cut segmentation approaches operate on graphs which their edges represent affinities between pixel pairs and whose nodes are pixel values. The first algorithm segment images using graph cuts was proposed by Wu and Leahy; in their method, the edge weights summation through cut boundaries was minimized and biased toward the short boundaries. The graph cut cost could be normalized to decrease bias.

TurboPixel algorithm segments images to a lattice like structure of superpixels by developing seeds to adapt to the structure of the local image. Levinstein et al. described a geometric-flow to compute the dense over-segmentation for images, while an overtime indicator was used in their approach. That helps to terminate under-segmentation for the compactness constraint, and at the same time produces partitions and segments that are considered image local boundaries.

TurboPixel algorithm gradually expands a seed group locations using the level-set based geometric flow. The geometric flow depends on the local image and targets the distribution of superpixels to be regularly distributed on the image. TP are constrained for having boundary adherence with uniform and compact size. TP is considered the slowest approach tested and is relatively poor when it comes to boundary adherence detection. It depends on the changing complexity approach, though in practice has approximately $O(N)$ behavior, as the authors claim.

**Scene Shape Superpixel:** Scene Shape Superpixel (SSP) [49] uses prior information for superpixel segmentation. It is inspired by superpixel lattices [42] which does not perform well in the non-uniform sampling image. SSP is sensitive for this type of images. It is suitable for road scene images where the objects distributed at the center are different and distant from those at the edge. First, it learns the boundary distribution in order to improve the superpixel segmentation. Then, non-uniform minimum cost path algorithm is used to get the shortest path. The running time of SSP in the learning step is linked to the iteration times. Although the algorithm does not give an explicit energy, it can be classified as a gradient ascent method.

**Simple Linear Iterative Clustering (SLIC):** Superpixel algorithms [17] collect and group a set of pixels into meaningful portions or regions, that can be named as superpixels and used for replacing the solid structure of the pixel grid. These algorithms provide a suitable simple form which used for computing image features, capture image redundancy, and mostly illuminate the dependence of image processing complex problems. They widely became the basic building blocks for the majority of computer vision and detection approaches, for example object localization, segmentation, depth estimation and body model estimation.

There are many algorithms introduced for generating superpixels, each has some drawbacks and advantages that might be better and suitable to a specific problems or application. The graph-based algorithm may be a very suitable choice. So, Methods that produces a more regular lattice is probably a better choice, if the superpixels are to be used in building a graph. The following characteristics may be important while defining a convenient approach for all applications is more difficult. Firstly, Superpixels algorithms and methods should be fast to segment, memory efficient, and reduce computational complexity in the preprocessing step; secondly, Superpixels should keep on well to image boundaries; thirdly, Superpixels should increase the speed and improve the quality of the results when used for segmentation purposes.

SLIC adapts k-means clustering algorithm to generate superpixels with two important differences; the first is the weighted distance that measures spatial proximity and combines color, while providing a control over image superpixels compactness and size. The second difference is the calculation of distances in the optimization. These differences are reduced by limiting the search space; and consequently the complexity will not depend on the number of superpixels ($k$) and at the same time will be reduced to be linear for a number
SLIC algorithm has only one parameter \((k)\) which is the target number of roughly equally-sized superpixels. In CIELAB color space for the color images. The first step in clustering procedure is initialization; where \(k\) initial centers are sampled on a regular grid spaced \(S\) pixels apart. The grid interval is to generate approximately equally sized superpixels; while centers are moved to seed locations according to the low gradient-position in a \(3 \times 3\) neighborhood. This was done for reducing the chance of seeding a superpixel with a noisy pixel, or centers of a superpixel on an edge. Within the evaluation step, each \(i\) pixel is connected with the nearest cluster center which searches for region that overlaps its location. This is the key for accelerating that approach, as restricting the size of the search region decreases the number of distance calculations significantly. Therefore, an important speed advantages over traditional \(k\)-means clustering algorithm could be achieved. The adjustment of the cluster centers is an update step, this adjustment is for the mean vector of all pixels that belong to the cluster.

SLIC is considered a customization for \(k\)-means method [66] to generate superpixels for an image. Superpixels were generated by clustering pixels depending on their image forms, color similarity and proximity. SLIC superpixel algorithm is similar to methods that were used as a preprocessing step for depth estimation.

**Depth-adaptive superpixels**: The Depth-adaptive superpixels (DAS) [50] is used for RGB-D images with depth information. It uses the additional depth information to reduce the complexity of the segmentation task. DAS contains three main steps. First, a computation for the density of superpixel clusters in the image space is performed using the depth image; second, the method of multi-scale sampling is used to sample points to guarantee the blue-noise spectrum property. Finally, \(k\)-means clustering method is used to map points to superpixels to improve superpixel centers. DAS can produce superpixels in real time.

**SEEDS**: The SEEDS [51] describes a another approach for generating superpixels. It starts from a grid of square superpixels to iteratively reassign boundary pixels (or blocks of pixels) from one segment to a neighboring segment. The approach begins with a complete segmentation followed by subsequent adaptation to the image content. It uses an objective function to enforce homogeneous segment smooth boundary shapes and colour distribution (in LAB colour space). Boundary smoothness is followed by a measurement using superpixels labels histogram in small local patches. Only a single superpixel label will appear inside superpixels and at the borders there will be other labels. The approach uses the sum of squared histogram entries to measure the smoothness of each patch. The objective function terms are designed to provide fast optimization using a greedy strategy. In addition, It is evaluated and updated quickly to change segment boundaries.

**Contour Relaxed Superpixels**: the Contour Relaxed Superpixels (CRS) [52] generates superpixels that achieve maximum homogeneous texture inside every patch, and maximum contours accordance with both image content and Gibb-Markov random field model. It uses estimation tasks to produce superpixel segmentation. The energy function in CRS is optimized using a small number of design parameters that depend on the input image statistical model.

**Manifold SLIC**: The Manifold SLIC [53] is a modification of SLIC to efficiently compute sensitive superpixels contents, i.e. small superpixels in densely content regions and large superpixels in sparsely content regions. Contrary to conventional SLIC which clusters pixels in Lab color space, Various SLIC map the image to a 2-dimensional space to calculate the content density.

**Further Superpixels Algorithms**: While the above algorithms represent most of the superpixel algorithms, some algorithms are not included due to the missing, unnoticed or only recently published implementations. These algorithms include [54, 55, 56, 57, 58].

**References**


Abdelhameed Ibrahim is an assistant professor of Computer Engineering at Faculty of Engineering, Mansoura University, Egypt. He received bachelor’s and master’s degrees in engineering from the Computer Engineering and Systems department in 2001 and 2005, respectively. He was with the Faculty of Engineering, Mansoura University, from 2001 through 2007. He received his Ph.D. in Engineering from Faculty of Engineering, Chiba University, Japan in 2011. He serves as a reviewer for the Journal of Electronic Imaging, Optical Engineering, IEEE Journal of Biomedical and Health Informatics, IEEE Access, Computer Standards & Interfaces, Journal of Healthcare Engineering, IET Image Processing, Multimedia Tools and Applications, and other respected journals. His research interests include computer vision, pattern recognition, cloud computing, virtualization, and live virtual machine migration.

El-Sayed M. El-kenawy is an assistant professor at Delta Higher Institute for Engineering & Technology (DHIET), Mansoura, Egypt. Inspiring and motivating students by providing a thorough understanding of a variety of computer concepts. He has pioneered and launched independent research programs. He is an IEEE member and interested in computer science and machine learning field. Adept at explaining sometimes complex concepts in an easy-to-understand manner.