Unsupervised Image Segmentation Based on Local pixel Clustering and Low-Level Region Merging
Rostom Kachouri, Mahmoud Soua, Mohamed Akil

To cite this version:
Rostom Kachouri, Mahmoud Soua, Mohamed Akil. Unsupervised Image Segmentation Based on Local pixel Clustering and Low-Level Region Merging. 2nd IEEE International Conference on Advanced Technologies for Signal and Image Processing ATSIP’16, Mar 2016, Monastir, Tunisia. hal-01309999

HAL Id: hal-01309999
https://hal.science/hal-01309999
Submitted on 1 May 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Unsupervised Image Segmentation Based on Local pixel Clustering and Low-Level Region Merging

Rostom Kachouri
rostom.kachouri@esiee.fr

Mahmoud Soua
soua.mahmoud@esiee.fr

Mohamed Akil
mohamed.akil@esiee.fr

Université Paris-Est, Laboratoire d’Informatique Gaspard-Monge, Equipe A3SI, ESIEE Paris, France

Abstract—Heterogeneous image segmentation is one of the most important tasks in image processing. It consists in partitioning the image into a set of disjoint regions. In this paper, we propose a new unsupervised image segmentation method that we call Unsupervised Image Segmentation (UIS). Our proposal performs an efficient image partition efficiently into primitive regions. This process is ensured by a local adaptive Kmeans and a novel centroids initialization. Then, similar sets are agglomerated to form homogeneous regions. For that, a low-level feature merging is employed according to a hierarchical linkage approach. Finally, in case of over-segmentation, appearing outlier regions are removed using a post process stage. Therefore, the UIS method allows to determine automatically the image region number. Indeed, it extends the Kmeans clustering to obtain meaningful regions. Several experiments were conducted using two heterogeneous image datasets. A comparison with well-known segmentation methods was also performed using the Liu’s factor measure.

Keywords—Region based image segmentation, Kmeans clustering, Region agglomeration, Similarity measure, Color, Texture, Over-segmentation, Outlier region.

I. INTRODUCTION

Heterogeneous image segmentation is a fundamental processing for image description and recognition tasks. It aims to partition the image into homogeneous regions. Subsequently, pixels are assigned to semantic meaningful objects. Several survey papers [1], [2] cover the major image segmentation algorithms providing high quality results. These algorithms can be categorized into two main approaches [3]: Boundary-based and Region-based methods. The first category [4] operates on the image edges. Indeed, edges are detected and then linked into contours representing the object boundaries [5]. Edge-based methods are very robust handling properly images with significant contrast variation between regions. However, the presence of small gaps in edge boundaries allows merging of dissimilar regions. In addition, they are sensitive to noise which leads often to generate non closed contours [6], [5]. The second category, representing region-based methods, is more noise immune [5] and produces enclosed regions [7]. Methods of this category operate by partitioning groups of dissimilar pixels according one or more features such as brightness, color, texture, etc. Then, similar pixels which are neighbors, are grouped together in order to reaches a set of meaningful regions.

As a well known region-based method, Region growing [8] enlarges gradually areas around starting pixels (seeds) based on a predefined criteria. Nevertheless, in this method user should choose initial seeds within the most homogeneous areas. Hence the inconvenience of this method is the choice of initial seeds. So that obtained results are not always optimal. Unlike Region-growing, the Split and Merge technique [9] generates automatically an initial partition of homogeneous regions. Several methods where proposed in this context [10], [29], [30]. For example, the Split and Merge based-quadtrees method [10] performs the split phase by assuming that the image is an entire block. Then regarding on homogeneity test. Heterogeneous blocks are divided into four sub-blocks. The treatment is stopped for each sub-block which reach homogeneity. In the merge phase, regions are gathered based on predefined similarity criterion and a Region Adjacency Graph (RAG). The main advantage of the Split and Merge methods is the fine subdivision of the image, however, they remain complex due to the tedious split treatments. Otherwise, methods based on pixel clustering, offer less complexity in region-based image segmentation [11], [12], [42]. More recent developments in this field [44] classify the input image pixels into multiple clusters based on their distance from each other [11], membership function [13], maximum likelihood criterion [14], etc. In this context, one of the most used methods for color image segmentation is the clustering algorithm Kmeans [16], [31]. However, the major drawback of methods using this classification algorithm is that user needs to set, in the beginning, the region number. Furthermore, in the agglomeration stage of these methods, different linkage techniques [17] can be employed. In the single linkage technique [17], the cluster similarity is defined by the shortest distance from any member of a cluster to any member of another one. This method shows a total insensitivity to shape and size of clusters. However it is sensitive to outlier pixels [18]. Another technique is the complete linkage [17]. It is based on the computation of the greatest distance from any member of a cluster to any member of another cluster. This one is not strongly affected by outlier pixels. However, the complete linkage method can break large clusters, and has trouble with convex shapes [18]. The level of complexity may be reduced by using centroid linkage technique [17]. Indeed, the distance between two clusters is represented by the distance between their centroids. Whenever two clusters are combined, a new centroid is computed. Nevertheless, it can fail in case of complicated cluster shapes. Indeed, centroids, in such case, do not represent efficiently the clusters [18]. It is obvious that each method has its own limitations and advantages. Recent works, try to combine more than one linkage technique to improve the agglomeration results [19], [20], [21].
In this paper, a novel region-based segmentation method is proposed. Firstly, the proposed method ensures an unsupervised split stage based on local Kmeans clustering. Obtained primitive regions are merged based on a combination of color and texture features. Different linkage techniques are employed to gather efficiently the similar regions. In case of oversegmentation, a post process is used to suppress the unwanted regions. In fact, our main contributions consist on a offering a full automatic segmentation method. A novel initialization method for the local Kmeans algorithm is proposed. Moreover, different linkage techniques are combined in the merge phase.

In the following, we describe our proposal in section 2. Next, in section 3, obtained results are shown and discussed. Finally conclusion is drawn in section 4.

II. PROPOSED METHOD: UNSUPERVISED IMAGE SEGMENTATION (UIS)

In this section, we describe the different stages of our Unsupervised Image Segmentation (UIS) proposal. As shown in Figure 1, two main stages are employed. The first one is the split stage where a window based approach is used to perform a local robust Kmeans clustering [11]. For this end, we divide the image into equal size blocks. The second stage is the merge phase. It consists in agglomerating the regions with both color and texture similarity measure. Many color spaces can be employed. In this work we use the Lab [22] color space for better segmentation results [45]. To compute distance between pixels in the image, the Euclidian distance [23] is used. After the merge stage, outlier region remove could be employed in case of over-segmentation. Finally, regions in the segmented image are shown with the color of their centroids to show a more expressive semantic regions. In next subsections, we describe, with more details, our proposed method stages.

A. Split stage

The first stage of our proposal consists in clustering local image blocks using the Kmeans algorithm [11]. Actually, we employ small data blocks, where the block Width and Height represent respectively a ratio of 25% from the image Width and Height. According to the literature [36] this size ensures that each block have high rate to contain one single semantic meaningful objects. However, it can contain more objects. Actually, we believe that this block size is enough to form two meaningful regions. This region number is shown to be appropriate by our experiments. Therefore, we initialize the local Kmeans with a set of two centroids, noted by $C_{bd} = \{C_{bd,0}, C_{bd,1}\}$, with $bl \in [0, Nb - 1]$ and $Nb$ is the total number of blocks in the image. We consider for that nine pixels in each block as a potential initial centroids. These pixels are located in the border and the middle of the blocks as shown in Figure 2.

Actually, among these nine selected pixels, we consider that, at least, one pixel belong to the second region if it exists. Then, based on the used color space, we initialize the local Kmeans in each block with the two most farthest pixels from the nine selected ones. After initialization, each block is clustered with Kmeans until convergence. We note by $R = \{R_0, R_1, ..., R_{k-1}\}$ the set of the obtained regions within all blocks in the image. $k$ is then the total number of regions, and it is equal to at most $2 \times Nb$ when each block contains two regions. Generally, $k \leq 2 \times Nb$. The UIS method gives a robust clustering thanks to its good initialization and to the efficient Lab color space use. In the next, we describe the Region Merging stage.

B. Merge stage

In this stage, the employed similarity criterion, are very important to determine which region is similar to which one. In general, the combination of color and texture features provides better segmentation result [37]. The advantages of both color and texture based segmentation are quite well preserved: we obtain sharp boundaries and homogeneous regions [32]. Actually, segmentation purely based on texture gives fuzzy boundaries but usually homogeneous regions [28]. Whereas, segmentation based on color is more sensitive to local variations in color but provides sharp boundaries [27]. Many researcher proposed various approaches for texture analysis [40], for color analysis [39] or both of them [37], [38]. In our work, we use both color and texture features in addition to the adjacency criterion.

1) Color Features: We consider that two regions can be merged only if they are adjacent, and have similar color. For that, as shown in Figure 3, we extract for each region three features: $R_{i,c}, R_{i,dmin}, R_{i,dmax}$ with $i \in [0, k - 1]$. Where, $R_{i,c}$ is the computed centroid of the region $R_i$. It defines the
median of region colors. \( R_{i,d_{\text{min}}} \) and \( R_{i,d_{max}} \) are respectively the smallest and the largest color variations in the region \( R_i \) (see Figure 3(a)). To decide if two regions \( R_i \) and \( R_j, i,j \in [0,k-1] \) are similar (Figure 3(b)) or not (Figure 3(c)), we compare the Euclidean distance \( \text{Distance}(R_i,c, R_j,c) \) between their respective centroids \( R_i,c \) and \( R_j,c \) to a given threshold, as stated in Equation 1.

\[
\text{Distance}(R_{i,c},R_{j,c}) \leq \text{Threshold} \quad (1)
\]

The idea is that the applied threshold is variable depending on the two considered regions, through the use of the \( \text{Max}(R_{i,d_{\text{min}}}, R_{j,d_{\text{min}}}) \) term. As illustrated in Figure 3, we can see that the two regions \( R_1 \) and \( R_2 \) (Figure 3(b)) are similar since \( \text{Distance}(R_{1,c},R_{2,c}) \leq \text{Max}(R_{1,d_{\text{min}}}, R_{2,d_{\text{min}}}) \). However \( R_1 \) and \( R_3 \) (Figure 3(c)) are dissimilar because \( \text{Distance}(R_{1,c}, R_{3,c}) > \text{Max}(R_{1,d_{\text{min}}}, R_{3,d_{\text{min}}}) \).

2) Texture Features: We define a coefficient that we call \( TD \) (see Equation 2) to consider the Texture Degree of the processed image. To compute TD, we evaluate the ratio between \( \text{Img}_{d_{\text{max}}} \) and \( \text{Img}_{d_{\text{min}}} \) as shown in Equation 2.

\[
TD = \frac{\text{Img}_{d_{\text{max}}}}{\text{Img}_{d_{\text{min}}}} \quad (2)
\]

Where \( \text{Img}_{d_{\text{min}}} \) and \( \text{Img}_{d_{\text{max}}} \) are the average distance of, respectively, all the \( R_{i,d_{\text{min}}} \) and all the \( R_{i,d_{\text{max}}} \) in the image \( \text{Img}, i \in [0,k-1] \). The final employed threshold is then given by the Equation 3:

\[
\text{Threshold} = \text{Max}(R_{i,d_{\text{min}}}, R_{j,d_{\text{min}}}) + TD \quad (3)
\]

with \( i,j \in [0,k-1] \). So, the \( TD \) term allows to increase the final threshold value depending on the texture degree of the processed image. Indeed, we note that in low textured image, \( \forall i \in [0,k-1] \), \( R_{i,d_{\text{max}}} \) tends to be equal to \( R_{i,d_{\text{min}}} \) (\( R_{i,d_{\text{max}}} \approx R_{i,d_{\text{min}}} \Rightarrow TD \approx 1 \)). Otherwise, high textured image, \( R_{i,d_{\text{max}}} \) is highly greater than \( R_{i,d_{\text{min}}} \) \( \forall i \in [0,k-1] \), (\( R_{i,d_{\text{max}}} \gg R_{i,d_{\text{min}}} \Rightarrow TD \gg 1 \)).

C. Outlier-Region remove stage

It is possible that the merged image contains some outlier regions. Generally, in such case, we talk about oversegmentation. To overcome this problem, we eliminate these small areas by merging them with the appropriate regions. Generally, considering the context of the application, meaningful objects in an image exist within a marge of sizes. Then, it is possible to impose a minimum valid object area for segmented regions [33]. Hence, researchers set the maximum pixel number for a valid object region [33], [34], [35]. In our work, we extract objects in a general purpose context.
(natural scene, single object). For this, we evaluate the sum of pixel number across the regions from the largest to smallest ones. When the pixel number reaches 60% from the total pixel number, the remaining areas will be considered as outliers. These regions are merged with the bounding ones. We note by \((x_1, R_1, x_2, R_2)\) and \((y_1, R_1, y_2, R_2)\) the horizontal and vertical limits of regions \(R_i\). For each \(R_i\), we note by \(\Delta_X\) and \(\Delta_Y\) the measures referring respectively to their Width and Height. They are defined as following: \(\Delta_X(R_i) = x_{2,R_i} - x_{1,R_i}\), \(\Delta_Y(R_i) = y_{2,R_i} - y_{1,R_i}\) with \(i \in [0, Nb]\) and \(Nb\) is the total number of regions. The outlier suppression is shown in Figure 4 and Algorithm 1. Following, we note by \(O\) the origin, \(R_o\) an outlier region and \(R_{C_i}, i \in [0, Nb]\) the set of the candidate bounding ones. We study the measures \(\Delta_X\) and \(\Delta_Y\) to determine if a given region \(R_{C_i}\) bounds \(R_o\). Two conditions rise up. Firstly, as shown in Figure 4(a), \(R_o\) can be totally bounded by an \(R_{C_i}\) (Line 4). In this case, these segments are merged together. Secondly, \(R_o\) can partially be bounded by several regions \(R_{C_i}\) (Line 7). Then, \(R_o\) is merged with the most color similar one. This case is illustrated in Figure 4(b).

**Algorithm 1 Outlier-Region remove stage**

```plaintext
1: for \(i := 0\) to \(k - 1\) do
2:   // \(R_o\) and \(R_{C_i}\) overlapping
3:   if \((x_1, R_o) > x_1, R_{C_i})\) or \((x_2, R_o) < x_2, R_{C_i})\) or
4:     \((y_1, R_o) > y_1, R_{C_i})\) or \((y_2, R_o) < y_2, R_{C_i})\) then
5:     \(R_o\) is totally enclosed in \(R_{C_i}\) (Figure 4.a)
6:     if \(\Delta_X(R_o) \leq \Delta_X(R_{C_i})\) and
7:       \(\Delta_Y(R_o) \leq \Delta_Y(R_{C_i})\) then
8:       Merge \(R_o\) and \(R_{C_i}\)
9:     else
10:        Merge \(R_o\) with the most color similar region \(R_{C_i}\)
11:   end if
12: end if
```

As shown in Figure 5, outlier regions designed by arrows (Figure 5(a)) are correctly merged with the bounding and color similar regions in the image (Figure 5(b)).

**III. EXPERIMENTAL RESULTS**

In the following, we evaluate our proposed UIS method on both Corel\(^1\) [24] and Berkeley\(^2\) [25] datasets. Evaluation is performed according to visual accuracy comparison and Liu’s factor measuring.

**A. The Liu’s factor metric**

We employ the Liu’s factor metric [26] to evaluate the obtained results with our UIS method. The following equation shows the Liu’s factor formula.

\[
L(I_{mg}) = \sqrt{k''} \times \sum \frac{\xi_i^2}{\sqrt{\mathcal{S}(R_i)}}, \quad i \in [0, k - 1]
\]

where, \(I_{mg}\) is the segmented image. We note respectively \(k''\) the final obtained number of regions and \(\mathcal{S}(R_i)\) the size of the region \(R_i\), ie its total number of pixels. \(\xi_i\) is the sum of the Euclidean distance of each pixel in the region \(R_i\) between the segmented image and the original one. It defines the region color error. The best image segmentation results are indicated with a small values of \(L\). Generally, images with large color error regions or small regions are penalized.

**B. UIS evaluation based on Visual Accuracy and Liu’s factor measuring**

The Figure 6 illustrates the obtained UIS segmentation results on five image samples from different Corel categories (Figure 6(a)).

As a first observation, it is clear that our UIS method gives efficient segmentation quality. Indeed, it provides the right region delimitation and the correct region number for all illustrated images. Actually, the local Kmeans employment produces a robust clustering by dividing the original image into \(k\) primitive regions. In addition, we observe that region contours are well preserved after the split stage (Figure 6(b)). According to Figure 6(c), the merge stage decreases efficiently the region number by connecting similar regions as explained in section 2.2. The obtained region number in this stage is noted by \(k''\). At this level, the final segmentation result may be obtained as it is the case of the Dinosaur image. Otherwise, in case of over-segmentation, the outlier regions are suppressed efficiently (Figure 6(d)). \(k''\) is the final number of regions in the obtained segmented image. Following, we show that we improve the segmentation quality when eliminating the outlier regions. According to the table below, the Liu’s coefficient is enhanced in the final segmented image compared to the merged one. Indeed, the less are the image segments small, the lower is the Liu’s coefficient and that is prove the efficiency of the UIS method.

For comparison reasons, we evaluate the proposed UIS with the state of the art methods on the well-known Berkeley dataset. Figure 7 illustrates segmentation results of Normalized cuts (NCUTS) [42], Dominant Set (DSet) [43], DSet-DBSCAN [44] and our proposed UIS. Results of the compared methods are taken from the literature [44]. We observe that

\(^1\)Corel dataset could be downloaded at: http://savvash.blogspot.fr/2008/12/benchmark-databases-for-cbir.html

\(^2\)Berkeley dataset is available at the web site: https://www.eecs.berkeley.edu/Research/Projects/CS/vision/bbsd
NCUTS is not able to avoid the problem of partitioning coherent regions into multiple segments. The same problem is encountered with the DSet method. This because it generates only convex shapes. In the other side, DSet-DBSCAN and UIS can partition the images into meaningful segments. Actually, DSet-DBSCAN extends the clustered DSet regions with the DBSCAN algorithm. Therefore it can handle non-convex shapes but still generating over-segmentation with the appearance of few outlier regions on the final results (Figure 7(c)). Nevertheless, the proposed UIS method processes as well non-convex shapes thanks to the hierarchical merge. In addition, the powerful outlier remove stage decreases considerably the amount of overlapping regions.

Following, we compare our segmentation results with, the Split and Merge [41], Watershed [8] and Verma [25] methods, using the Liu’s factor metric. As shown in Figure 8, our proposed method gives 3× lower Liu’s factor values than Watershed, 2× lower values than Verma and 1.5× lower values than the Split and Merge method.

IV. CONCLUSION

Image segmentation is a sensitive key point in image processing. A new automatic and full unsupervised image segmentation method was proposed in this paper. Several experiments are performed using Corel and Berkeley image datasets. We show that the UIS provides robust image segmentation results and determines correctly image region number. Conducted results, prove that UIS outperforms state of the art.
References


