An Image Segmentation Algorithm based on Community Detection

Youssef Mourchid, Mohammed El Hassouni and Hocine Cherifi

Abstract With the recent advances in complex networks, image segmentation becomes one of the most appropriate application areas. In this context, we propose in this paper a new perspective of image segmentation by applying two efficient community detection algorithms. By considering regions as communities, these methods can give an over-segmented image that has many small regions. So, the proposed algorithms are improved to automatically merge those neighboring regions agglomerative to achieve the highest modularity/stability. To produce sizable regions and detect homogeneous communities, we use the combination of a feature based on the Histogram of Oriented Gradients of the image, and feature based on color to characterize the similarity of two regions. By constructing the similarity matrix in an adaptive manner, we avoid the problem of the over-segmentation. We evaluate the proposed algorithms for Berkeley Segmentation Dataset, and we show that our experimental results can outperform other segmentation methods in terms of accuracy and can achieve much better segmentation results.

Keywords: Image segmentation; complex networks; community detection; modularity

Youssef Mourchid
LRIT, URAC No 29, Faculty of Sciences, Mohammed V University in Rabat, B.P.1014 RP, Rabat, Morocco. e-mail: youssefmour@gmail.com

Mohammed El Hassouni
DESTEC, FLSHR, Mohammed V University in Rabat, Rabat, Morocco.
LRIT, URAC No 29, Faculty of Sciences, Mohammed V University in Rabat, B.P.1014 RP, Rabat, Morocco. e-mail: mohamed.elhassouni@gmail.com

Hocine Cherifi
LE2I UMR 6306 CNRS, University of Burgundy, Dijon, France. e-mail: hocine.cherifi@u-bourgogne.fr

1 INTRODUCTION

Image segmentation is still a challenging issue in visual information processing. Its goal is to split the image into homogeneous regions that represent similar features. It constitutes an essential issue in pattern recognition due to its practical importance. For example image, segmentation procedures in medical imaging can be used for diagnosis, allowing locating tumors and other pathologies [1]. Also, image segmentation techniques can be applied to machine vision, localization of objects in satellite images, and traffic control systems [2]. In recent years, graphs have emerged as a representation for image analysis and processing. Many powerful algorithms in image processing have been formulated on graphs, i.e., a pixel in an image is the vertex in the graph, and the edge is determined by an adjacency relation among the image pixels. Using graphs in the image is not absolutely a new idea and there are many published methods of graph similarity testing. The common idea of all these methods is the construction of a weighted graph, where each vertex corresponds to an image pixel or a region, and the weight of each edge connecting two vertices represents the similarity that they belong to the same segment. Several key factors affect image segmentation, for example, proximity, similarity, regularity, i.e., the repetitive patterns, relative size and etc. In this paper, we will take into consideration all these factors. A lot of image segmentation algorithms have been proposed in the literature: *Region Based* [3], *Watershed* [4]-[7], *Feature based Clustering* [8] and *Mean Shift* algorithm [9].

Inspired by the application of community detection algorithms in large networks, we try to view an image as a network or a graph. For a network, modularity [10] and stability [11] are crucial quantities, which are used to evaluate the performance of community detection algorithms when the underlying community structure is not known. Unlike the existing image segmentation algorithms, the proposed approach identifies the differences between community detection and image segmentation and, proposes a texture feature to count the occurrences of gradient orientation in localized portions and encode it into a similarity matrix. The similarity among regions of pixels is constructed in an adaptive manner for avoiding the over-segmentation. The proposed algorithms can automatically detect the number of regions in an image compared with other existing segmentation algorithms, it can also produce sizable regions and achieves much better semantic level segmentation of the image. The proposed contributions of this paper are the following:

- Efficient community detection algorithms are used. They present a low time complexity as well as comparable performance.
- A texture feature named Histogram of Oriented Gradients (HOG) [12] is used to detect regions of interest in the image. The HOG feature, together with the color feature, encodes much better similarity measure from the semantic point of view.
- Finally, the construction of an adaptive similarity matrix W is proposed to avoid the over-segmentation. At each iteration, the similarity between two regions of

the image is recalculated. The goal is to avoid breaking visually coherent regions, which have smooth changes in color or texture caused by shadow or perspectives.

The rest of the paper is organized as follows. In Section 2, we introduce briefly the concept of community detection and modularity/stability. Then, we recall efficient community detection algorithms. Section 3 explains how image segmentation and community detection can be related followed by the description of the proposed contribution and the detailed technical points. Experiments on the publicly Berkeley Segmentation Data Set (BSDS500) are reported in Section 4. Finally, in Section 5, we present our conclusions.

2 Community Detection

Community Detection is a hot topic in network science during the past few years, and it's a very prolific subject in the complex network literature [13]. A community is a group of nodes which are densely connected with each other and are sparsely connected with members of other communities. The community detection is a fundamental problem, which objective is to find the best division of the network into their constituent communities. Several algorithms have been developed so far to deal with this issue. Numerous solutions to solve this problem, are linked to a measure called modularity. Introduced by Newman [14], it measures the quality of a community structure and it is defined as follow:

$$Q = \Sigma(e_{ii} - a_i^2) \tag{1}$$

Where e_{ii} denotes the fraction of network edges which are inserted into a community i, and a_i^2 denotes the fraction considering that edges are inserted randomly. The modularity value Q is between 0 and 1. A high value of the modularity indicates a strong community structure of the network.

Another quality measure called stability Qs was introduced in [11] based on the clustered auto-covariance of a dynamic Markov process. It measures the quality of a partition as a community structure. Because the stability has an intrinsic dependence on time scales of the graph, it can allow the comparison and the ranking of the partitions at each time and also establish the time spans over which partitions are good and optimal. Thus, the Markov time acts effectively as an intrinsic resolution parameter that establishes a hierarchy of increasingly coarser communities.

Several algorithms have been developed to find a partition of the network which is a good approximation of maximum modularity or stability. In the following, we present two influential algorithms that we propose to use.

2.1 Fast multi-scale detection of communities using stability optimization

Modularity initially was introduced to evaluate the quality of partitions. Nevertheless, its use has broadened from partition quality measure to optimization function and now modularity optimization is a very common technique to detect communities. In this algorithm [16], the variation in modularity ΔQ_M to merge two communities i and j is computed as follows:

$$\Delta Q_{M_{ij}} = 2(e_{ij} - a_i a_j) \tag{2}$$

Where i and j denote the merged communities in the new candidate partition. When a better dQ is found when moving a node, the algorithm checks that moving this node does not leave its initial community disconnected. Otherwise, some communities may end up being in several components that should not be grouped together as one. Note that this implementation uses two distinct lists of neighbors. The first lists the actual neighbors in the initial network and the second stores the neighbors in the current matrix for the given parameter. This is necessary in order to only consider actual neighbors when selecting the candidate neighbor nodes and communities. The computation of the matrices for each parameter value is optimized by keeping in memory the recent matrices and corresponding exponents. For each new exponent, the optimization attempts to exploit the previously computed matrices to speed up the matrix power computation.

2.2 Modularity optimization based on Danon greedy agglomerative method

It's a greedy community detection agglomerative method which has been introduced by Danon [17]. The algorithm process is a simple modification of the algorithm proposed by Newman for detecting communities. The greedy method of Newman is an agglomerative hierarchical clustering method, where groups of nodes are successively joined to form large communities such that the value of modularity increases after this merging. This greedy optimization of modularity tends to form faster large communities at the expenses of small ones, which often yields poor values of the maximum value of modularity. Danon suggested a normalization of the modularity variation produced by merging two communities by the fraction of edges incident to one of the two communities, in order to avoid having small communities. This trick leads to better modularity value as compared to the original recipe of Newman, especially when communities have different sizes

3 Segmentation algorithms

Due to the inherent properties of images, there is a difference between the segmentation and community detection problems, and applying directly community detection algorithms to image segmentation [18], by considering the pixels as nodes of the network, lead to low performance. The following aspect reveals the difference between image segmentation and community detection. First, pixels in segmentation possibly have completely different properties, like the color but in community detection, they share similar properties. Second, a single pixel cannot capture regularities and information in each visually homogeneous segment of the image. Third, images share some information compared with communities, for example, adjacent regions are more likely to belong to the same segment.

So, to solve the mentioned problems, we propose an approach which takes advantage of the efficient optimization in modularity/stability using community detection algorithms and also the inherent properties of the image. The proposed algorithms start by an initial segmentation which split the image into homogeneous regions, possibly small regions which are used in the next steps. The proposed segmentation approach is explained in Algorithm.1 and the detail of some technical points are defined below.

Algorithm 1:

Input : Given a color image I and its over-segmented initialization with a set of super-pixels $R = \{R_1, ..., R_n\}$ where n is the number of super-pixels
Output: The set of image segments $C_i = \{C_{i1}, ..., C_{ic}\}$ with $c \leq n$

while *community structure still change* ($C_i \neq C_{i-1}$) **do**
  Construct the neighborhood system for each region R_i;
  Compute the Histogram of Oriented Gradients texture feature and estimate the distribution of the color feature for each region R_i;
  Adaptively update the similarity matrix W according to Equation (5), $w_{ij} \neq 0$ only if R_i and R_j are adjacent regions in I
  while *modularity/stability increases by merging any two adjacent regions* **do**
    Compute the community structure using a community detection algorithm $C_i = \{C_{i1}, ..., C_{im}\}$ where m is the current number of communities
  end
end

3.1 Super-pixels

In this paper, the super-pixels are chosen as an initial segmentation because we want to avoid the over-segmentation issue. It is characterized by the splitting of the same perceptual region in a multitude of smaller regions. Furthermore, super-pixels can well preserve the object boundaries. The proposed community detection algorithms start the process of aggregation by treating each single pixel as a community which will take more time and also there is no reason to treat a single pixel because it contains no information about texture. For these reasons, we start with an initial segmentation by super-pixels which are a set of very small regions of pixels. Using this pre-segmentation can greatly reduce the complexity without affecting the segmentation performance. In this paper, we use a publicly available code [25] to get the initial segmentation. As shown in Figure 1, the initial segmentation by the super-pixel generation step gives more than 200 over segmented regions which can greatly reduce the complexity by considering only 200 nodes instead of a large number of nodes in the first iteration for the proposed algorithms.

3.2 Construction of the similarity matrix

Images have self-contained spatial a priori information which is used to construct different neighborhood system. For more specification, we consider the possibility of merging neighboring regions in the image by considering the adjacent regions of each region in the image to be its neighbors and store its neighboring regions using an adjacent list which contains the regions that share at least one pixel with the current region.

3.2.1 Features to compute the similarity

The most straightforward and important feature for segmentation is color. Various color spaces are proposed in the literature to capture different aspects of the color, such as L*a*b, HSV, YUV, and RGB. The choice of an appropriate color space is a very important step for achieving a good segmentation performance. We choose the L*a*b color space because it's known to be in accordance with the human visual system and perceptually uniform. It is a 3-axis color system with dimension L for lightness and a and b for the color dimensions.

We use the pixel value in the L*a*b color space as a feature to compute the similarity between regions. Nevertheless, using this feature only cannot achieve good segmentation performance, because in some homogeneous object using just the color feature will break down image regularities into different segments. To solve this problem, we propose to use a texture feature called Histogram of Oriented Gradients (HOG). HOG is a feature descriptor used in computer vision and image processing to detect objects. This technique counts occurrences of gradient orientation in localized portions of an image detection window, or region of interest. We construct the Histogram of Oriented Gradients as follows:

- We divide the image into small connected regions (cells). For each cell, we compute a histogram of gradient directions or edge orientations for the pixels within the cell.

- We discretize each cell into angular bins according to the gradient orientation.
- Each cell's pixel contributes weighted gradient to its corresponding angular bin.
- Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
- Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor.

3.2.2 Similarity measure

Different similarity measures are used for the two features. For the color feature, a three-dimensional vector in the L*a*b color space represents each pixel in the image. For measuring the similarity between two regions in the image, the pixel value in the same region is represented by a three-dimensional Gaussian distribution. Several distance measures for distributions are studied in the literature like Kullback-Leibler (KL) Divergence, Mean Distance, and Earth Mover's Distance. In the proposed algorithms, to compute the distance between the two color feature distributions for two regions of pixels, we use the Mean Distance (MD). We use a Gaussian type radius basis function for transforming the above distribution distance into similarity measure defined by:

$$c_{ij}(color) = exp\{\frac{-dist(R_i,R_j)}{2\sigma}\} \tag{3}$$

Where $dist(R_i,R_j)$ denotes the distance between the pixel value distributions for region R_i and R_j.

For the proposed Histogram of Oriented Gradients (HOG), we use the cosine similarity to measure the similarity between the regions where each region is represented by a 256 dimensional vector and for two regions R_i and R_j. The HOG feature vector is h_i, $h_j \in R^{256}$ as indicated by:

$$t_{ij}(texture) = cos(h_i,h_j) = \frac{h_i^T h_j}{||h_i||.||h_j||} \tag{4}$$

3.2.3 Construction of the adaptive similarity matrix

The process of constructing the similarity matrix W is adaptive. During each iteration, we maintain an adaptive similarity matrix by recomputing the similarity between each two regions again in accordance with the equation (3) and (4). The reason for using this process is because, during the aggregation process of the community detection algorithms, regions keep expanding. The similarity measure resulting from the previous iteration might not suitable for the current iteration. So, using an adaptive similarity matrix reevaluate the similarity between current regions. It avoids over-segmentation and finally overcomes the problem of splitting

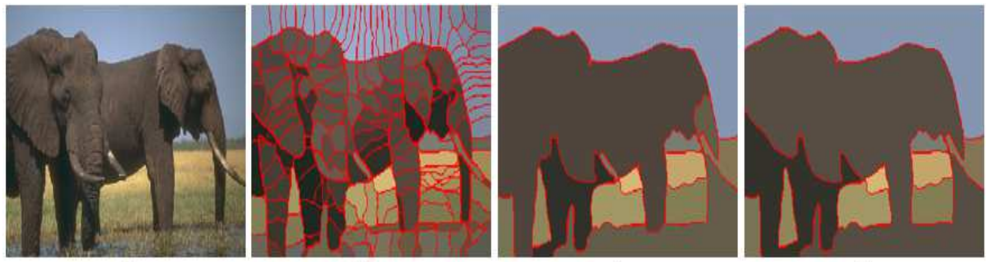

Fig. 1: a) Original image; b) Super-pixels image; c) Fast multi-scale using stability optimization; d) Modularity optimization based on Danon;

the non-uniformly distributed color or texture, which should be grouped into the same community in the image from the perspective of the human visual system. To construct the adaptive similarity matrix W during each iteration, we use a hybrid model to combine the color feature and the HOG texture feature as defined below:

$$W = w_{ij} = a \times \sqrt{t_{ij}(texture) \times c_{ij}(color)} + (1-a) \times c_{ij}(color);\ (i,j) = 1,..,n \quad (5)$$

Where n is the number of regions and a denotes a balancing parameter. If the texture information is not taken into consideration, i.e., $a = 0$, the more we increase the value of a, the more stripe patterns are encoded into the similarity, thus better preserves the regularities and information in the image. Nevertheless, if a is too large, some distinct objects in the image are merged into one segment. In our experiments, we give higher priority to the color feature.

4 Experiments and results

This section provides experiments that were conducted to assess the performance of the proposed approach qualitatively as well as quantitatively. The proposed algorithms are tested on the publicly available Berkeley Segmentation Data Set 500 (BSDS500) [19]. BSDS500 contains 100 validation images of size 321×481 pixels that are randomly chosen from the Corel database. These images are manually segmented by humans in a natural way. In the qualitative evaluations experiment, figure 1 shows the results of the proposed algorithms with the adaptive similarity matrix for image segmentation. So, we can see that the proposed algorithms give much better results and produce sizable regions for all selected images. Even if some pixel in the image have different values inside the same regions, the adaptive similarity ma-

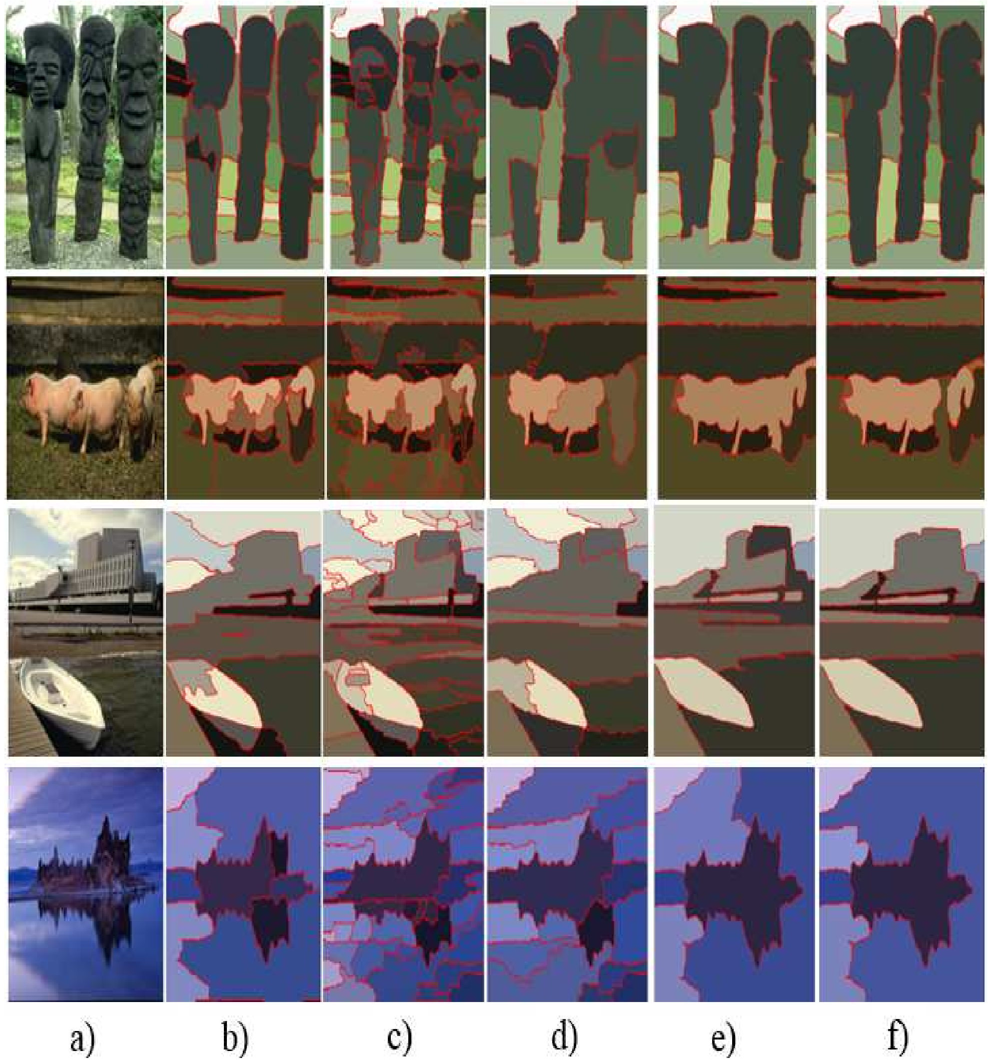

Fig. 2: a) Original image; b) LC; c) JSEG; d) EDISON; e) Fast multi-scale using stability optimization; f) Modularity optimization based on Danon;

trix of HOG texture feature can successfully preserve the regularities and classifies those pixels into the same segment.

We have performed a qualitative and quantitative comparisons of the proposed algorithms based on the adaptive similarity matrix with some existing state of the art segmentation methods: Lossy Compression (LC) [20], EDISON [21] and JSEG [22]. As shown in figure 2, LC, EDISON, and JSEG show the different extent of over-segmentation and break the regularities in some homogeneous regions of the image compared to the proposed approach which preserves the regularities and produces sizable homogeneous regions.

For the quantitative evaluation, we evaluate the segmentation performance of the proposed algorithms with the three segmentation techniques. We investigate for the quantitative evaluation the Probabilistic Rand Index [23] which is a classical evaluation criteria for clustering. It measures the probability that the pair of samples has consistent labels in the two segmentations. A larger value indicates a greater similarity between two segmentations. The range of PRI is [0,1]. Table1 presents the average values of the PRI, which were applied to all of the 100 images in the Berkeley segmentation dataset. Again, it has been observed that the proposed algorithms work better for the image segmentation task among all the popular segmentation algorithms LC, EDISON and JSEG in term of PRI and have a close performance to human perception.

Algorithms	PRI
Humain	0.870
Fast multi-scale	**0.811**
Modularity optimization based on Danon	**0.803**
EDISON	0.786
JSEG	0.760
LC	0.778

Table 1: Quantitative comparison of different algorithms on Berkeley dataset

5 Conclusion

This paper proposed an efficient image segmentation algorithm taking advantages of the optimization of modularity/stability and the inherent properties of images. To optimize modularity/stability, we used the efficient community detection algorithms, Fast multi-scale using stability optimization and Modularity optimization based on Danon which can automatically detect the number of segments in the image. By employing the color feature and the Histogram of Oriented Gradients (HOG) texture feature, we constructed the similarity matrix adaptively among different regions by optimizing the modularity/stability and aggregated the neighboring regions iteratively. When no modularity/stability increase occurs by aggregating any neighboring regions, the optimal segmentation is achieved. Results of our experiments have proved that the proposed algorithms give an impressive qualitative segmentation result as shown in the figures and achieve the best performance quantitatively among all the experimented popular methods in terms of PRI. Since, using two efficient community detection algorithms, the proposed approach avoid the over-segmentation problem and preserve the regularities in the object.

References

1. J. A. Noble and D. Boukerroui, "Ultrasound image segmentation: A survey," IEEE Transactions on medical imaging, 25(8), 987-1010.
2. H. Zhang, J. E. Fritts, and S. A. Goldman, "Image segmentation evaluation: A survey of unsupervised methods," Computer Vision and Image Understanding, Elsevier, vol. 110, no. 2, pp. 260-280, 2008.
3. A. M. Khan, Ravi. S," Image Segmentation Methods: A Comparative Study", IJSCE, Volume-3, Issue-4, September 2013.
4. L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," IEEE transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 6, pp. 583-598, 1991.
5. V. Grau, A. Mewes, M. Alcaniz, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information," IEEE Transactions on Medical Imaging, vol. 23, no. 4, pp. 447-458, 2004.
6. X.-C. Tai, E. Hodneland, J. Weickert, N. V. Bukoreshtliev, A. Lundervold, and H.-H. Gerdes, "Level set methods for watershed image segmentation," in Scale Space and Variational Methods in Computer Vision. Springer, 2007, pp. 178-190.
7. V. Osma-Ruiz, J. I. Godino-Llorente, N. Saenz-Lech on, and P. G omez-Vilda, "An improved watershed algorithm based on efficient computation of shortest paths," Pattern Recognition, vol. 40, no. 3, pp. 1078-1090, 2007.
8. R.Yogamangalam, B.Karthikeyan, "Segmentation Techniques Comparison in Image Processing", IJET, Vol 5, No 1, Feb-Mar 2013.
9. D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, pp. 1271-1283, 2010.
10. L. Angelini, D. Marinazzo, M. Pellicoro, and S. Stramaglia, "Natural clustering: the modularity approach," Journal of Statistical Mechanics: Theory and Experiment, vol. 2007, no. 08, p. L08001, 2007.
11. J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, "Stability of graph communities across time scales," Proceedings of the National Academy of Sciences, vol. 107, no. 29, pp. 12 755-12760, 2010.
12. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in IEEE Conference on Computer Vision and Pattern Recognition, CVPR., vol. 1. IEEE, 2005, pp. 886-893.
13. M. A. Porter, J.-P. Onnela, and P. J. Mucha, "Communities in networks," Notices of the AMS, vol. 56, no. 9, pp. 1082-1097, 2009.
14. M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," Physical review E, vol. 69, no. 2, p. 026113, 2004.
15. M. E. Newman, "Analysis of weighted networks," Physical Review E, vol. 70, no. 5, p. 056131, 2004.
16. Le Martelot, Erwan, and Chris Hankin. "Multi-scale community detection using stability optimisation within greedy algorithms." arXiv preprint arXiv:1201.3307(2012).
17. L. Danon, A. Daz-Guilera, A. Arenas," The effect of size heterogeneity on community identification in complex networks" , J. Stat. Mech. 11 (2006).
18. Mourchid, Y., El Hassouni, M., and Cherifi, H. "Image segmentation based on community detection approach". The International Journal of Computer Information Systems and Industrial Management Applications. ISSN 2150-7988 Volume 8 (2016) pp. 195-204
19. P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pp. 898-916, 2011.
20. A. Yang, J. Wright, Y. Ma, and S. Sastry, "Unsupervised segmentation of natural images via lossy data compression," Computer Vision and Image Understanding, vol. 110, no. 2, pp. 212-225, 2008.

21. C. M. Christoudias, B. Georgescu, and P. Meer, ”Synergism in low level vision,” in Pattern Recognition, 2002. Proceedings. 16th International Conference on, vol. 4. IEEE, 2002, pp. 150-155.
22. Y. Deng and B. Manjunath, ”Unsupervised segmentation of color-texture regions in images and video,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 23, no. 8, pp. 800-810, 2001.
23. R. Unnikrishnan, C. Pantofaru, and M. Hebert, ”Toward objective evaluation of image segmentation algorithms,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 6, pp. 929-944, 2007
24. R. Unnikrishnan, C. Pantofaru, and M. Hebert, ”Toward objective evaluation of image segmentation algorithms,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 6, pp. 929-944, 2007
25. G. Mori, ”Guiding model search using segmentation,” in The Proceedings of the 10th IEEE International Conference on Computer Vision, ICCV., vol. 2. IEEE, 2005, pp. 1417-1423.