International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 8 (2016) pp. 195–204

Image segmentation based on community detection approach

Youssef Mourchid[1], Mohammed El Hassouni[2] and Hocine Cherifi[3]

[1]LRIT URAC 29, University of Mohammed V-Agdal,
Rabat, Morocco.
youssefmour@gmail.com

[2]LRIT URAC 29, University of Mohammed V-Agdal,
Rabat, Morocco.
mohamed.elhassouni@gmail.com

[3]LE2I UMR 6306 CNRS, University of Burgundy,
Dijon, France.
hocine.cherifi@u-bourgogne.fr

Abstract: **Image segmentation is a fundamental low-level vision problem with a great potential when it comes to its applications. Several methods exist to split an image into regions. However, this problematic is still a relatively open topic for which various research works are regularly presented. With the recent developments in complex networks theory, methods based on graphs, which, can segment an image has considerably improved. This paper presents a new perspective of image segmentation by applying the most efficient community detection algorithms. For this, we first transform images into an adjacency graph. Then, we propose to study five complex network dedicated community detection methods which are Infomap, Louvain, Fast multi-scale detection of communities based on local criteria, Multi-scale detection of communities using stability optimization and stability optimization based on Louvain. Finally, we extract communities (regions) in which the highest modularity or stability feature is achieved. In our experiments, we establish a fair comparison between the proposed algorithms for Berkeley database images, and we show that a good performance is achieved by multi-scale detection of communities using stability optimization with a probabilistic rand index PRI of 0.81.**

Keywords: Image segmentation, complex networks, community detection, modularity, stability.

I. Introduction

Image segmentation is a technique used to split the image into regions that constitute an essential issue in pattern recognition. It has been a subject of intensive research for many years. Image segmentation's goal is to identify the objects of interest in an image, also differentiate those objects from the background and finally associate a label that indicates the object it belongs to each pixel. Image segmentation is defined as the process of clustering an image into different regions such that each region is homogeneous. A variety of algorithms has been proposed in the literature for segmentation purposes. Here, we only give a brief review of some of them. Normalized cut [1] is a new approach for graph partition which aims to extract the global information of an image by studying the spectra characteristic of the graph. Normalized Cut tries to solve the generalized eigenvector problem for a given affinity matrix W where each entry represents the similarity of two pixels:

$$(D - W)y = \lambda Dy \tag{1}$$

where D is the diagonal matrix, y is a vector of length N equal to the number of pixels in the images. The segmentation is achieved by partitioning the eigenvectors.
The mean shift algorithm [2] is a non-parametric technique which treats image segmentation as a problem of clustering. Mean shift considers feature space as an empirical probability density function. Each pixel of the image is converted into the joint spatial-range feature space by concatenating both, the pixel color value and its spatial coordinates. After that, the procedure of mean shift is applied to yield a convergent point for each pixel into a single vector. This algorithm is usually fast but is still very sensitive to his bandwidth parameter.
The watershed transform [3] is the most influential image segmentation method in the field of mathematical morphology. This method considers the gradient magnitude of an image as a topographic surface. The pixels are in one segment (also called 'Basins'), where a water drop starts from would drain to the same local intensity minimum. But, this method is generally sensitive to noise and easily leads to over-segmentation.
In recent years, graphs have emerged as a topological representation for image analysis and processing [4, 5]. Many powerful methods in image processing have been formulated on graphs, i.e, a vertex set in graph is the pixels set in image, and the edge set is determined by an adjacency relation

among the image pixels. The idea of using the graph theory in images is not absolutely new [6] and there are many published examples of graph similarity testing. Different techniques have been proposed for image segmentation. Most of these methods present some drawbacks and do not provide good segmentation. Some methods for example, take spectral partitioning algorithms, which only divide a graph into two sets instead of an arbitrary number of clusters. Indeed, division into more than two sets can be attained by repeated bisection. However, this approach does not lead to the best division into three groups.
With the development of complex networks theory, image segmentation based on graph has evolved considerably [7, 8, 9]. The identification of regions of pixels can be fulfilled by communities detection methods on vertices [10, 11, 12, 13, 14]. Community detection is a very prolific subject in the complex network literature [15]. A huge variety of algorithms have been developed so far to deal with this issue. Up until now there's no clear definition of what a community is, this problem has been dealt by many different points of view. It has been expressed as graph partitioning, community mining, spectral analysis, an optimization problem, a statistical problem and so on [16, 17]. These techniques of community detection provide more specific partitions than the traditional methods based on graph, such as the spectral partitioning [18]. Even so, the image segmentation approach based on networks presents some drawbacks. So that, only the small images can be performed since most community identification methods are computationally expensive.

In this work, we propose to formulate the problem of image segmentation to a community detection complex networks based framework. The main idea is to use the graph representation to determine communities that correspond to regions in the images. This leads us initially to transform our image into a complex network, and then apply a community detection algorithms. As long as our goal do not lie in developing a new detection algorithm, we propose to implement and compare five of the most effective methods. The latter are: Infomap [19], Louvain [20], Fast multi-scale detection of communities based on local criteria [10], Multi-scale detection of communities using stability optimization [10] and stability optimization based on Louvain [21]. A quick review of these methods is shown in section III. Finally, we summarize the graph's structure into a single modularity [22], and stability [10] feature measures calculated by community detection algorithms. These measures are invariant to image rotation and robust to small distortion. A fair comparison of these methods has been done. Three most influential image segmentation methods are also involved in our comparison which are EDISON [23], JSEG [24] and MULTISCALE [25].

The rest of the paper is organized as follows. In Section 2, we show how to represent an image as a complex network. In Section 3, we propose the most efficient community detection algorithms to extract the modularity feature measure. In Section 4, experiments are shown to illustrate the performance of the modularity/stability feature and the proposed algorithms on the publicly Berkeley Segmentation Data Set (BSDS300). Finally in Section 5, we present our conclusions.

II. Representation of image as a Complex Network

Images can be presented as graphs, where each image pixel and edge weight represents a node in the graph, they are computed according to a weight or similarity function. There are different weight functions, which, are based on Euclidian distance, Manhattan, Gaussian and others. First nodes are linked to each other with a weight function based on the intensity, given by:

$$W_{i,j} = \begin{cases} 1 & \text{if } |I_i - I_j| > t \\ 0 & \text{sinon} \end{cases}$$

where $W_{i,j}$ is the edge weight between i and j pixels and is defined in the interval [0,1], I_i and I_j represent respectively the intensity of pixel i and j. Connections are considered between nodes only, if the weight is greater than threshold **t** and the value of **t** can vary according to the similarity of the pixel intensity. Also, connections are defined only inside a circular pixel neighborhood of radius **R**, which varies between 4 and 8. The idea behind this approach of graph generation is that human vision tends to focus on high-contrast places. The threshold value can vary according to the similarity of the pixel intensity and also the radius varies according to the image size and color regions proximity.

III. Community detection in a Complex Network

Community structure is one of the most relevant features of graphs representing real systems, or clustering, i.e. The representation of vertices in communities, with many edges joining vertices of the same community and comparatively few edges joining vertices of different communities. Furthermore, rather than focusing on how communities are detected, the classification is based on the definition of the community used by the algorithms. Starting from a meta-definition of a community, algorithms are classified in eight categories according to different interpretations of the meta-definition. The community detection has a fundamental problem, its how to define the best division of the network into their constituent communities. To solve this problem, Newman [26] proposed a measure called modularity Q which indicates the quality of a partition of the network. Q is defined as follows

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

where e_{ii} is the fraction of network edges that are inserted into a community i, and a_i^2 is the fraction considering that edges are inserted randomly, the value of modularity Q range from 0 to 1, if values are close to 1, the communities do not exist by chance. Originally, the modularity Q expresses the fact that a community structure has a high-density ratio as compared to a random graph with the same degree sequence. The main drawback of this approach is that it is also an optimization criterion used by a large number of algorithms, using it as a quality function, it introduces a bias in the comparisons. Another measure called stability Qs was introduced [10], it measures the quality of a partition as a community structure based on the clustered autocovariance of a dynamic Markov process, which takes place on the network. Because the stability has an intrinsic dependence on time scales of the graph, it allows us to compare and rank partitions at each time and also to establish the time spans over which partitions are optimal. Hence, the Markov time acts effectively as an intrinsic resolution parameter that establishes a hierarchy of increasingly coarser communities. Different algorithms are able to find a good approximation of maximum modularity or stability. In this work, we choose the most efficient community detection algorithms used to segment an image into homogeneous regions.

A. Infomap Algorithm

Infomap, developed by Rosvall and Bergstrom [19], uses a compression technique to describe the information flow on networks. Random walks of a given length and with a given probability of jumping to a random node are performed. Each walk is described as a sequence of steps inside a community followed by a jump through a two-level nomenclature based on Huffman coding. The first one is used to distinguish communities in the network and the other to distinguish nodes in a community. Each node codeword is derived from the visit node frequency of an infinitely long random walk. This coding strategy leads to a compact representation of the walks. Indeed, with a partition with few intercommunity links, the walker is statistically more likely to stay longer inside communities, therefore, only the second part of the nomenclature is needed to describe its path. The authors showed that the optimal partitioning problem reduces to finding the minimum description length for all the walks.

B. Louvain Algorithm

Louvain is greedy agglomerative hierarchical algorithm proposed by Blondel [20]. Two phases are repeated iteratively (Fig.1). Starting with each node in its own community, the gain in modularity obtained by placing a node in the same community that its neighbors are evaluated. The community offering maximal gain is retained. This process is applied repeatedly and sequentially for all nodes until no individual move can improve the modularity. At the end of this first phase, the algorithm yields the first partitioning scheme. In the second phase, a new network whose nodes are the communities found during the first phase is build. The intra-community links are represented by self-loops, whereas the inter-community links are aggregated and represented as links between the new nodes. It is then possible to reapply the first phase to the resulting weighted network and to iterate until only one community remains.

C. Fast multi-scale detection of communities based on Local Criteria (FMD)

Recently, several multi-scale criteria and associated methods to uncover communities were introduced [11, 12, 13, 14]. A new method for the fast detection of communities across scales based on some of these criteria was introduced in [10].

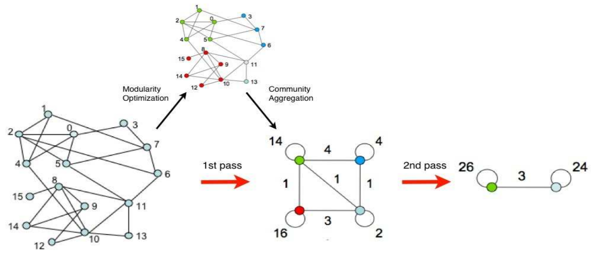

Figure. 1: Process of community detection for Louvain algorithm [20].

The concept of this algorithm follows the same steps of the method from [10]. First, the algorithm is initialized with a set of nodes called seeds that will form the initial communities. The selection of this seeds is randomly from a candidate set, removed from it and added to the seed set. All the neighbors of this seed are then removed from the set of remaining seed candidates. This stops starting different communities from neighbor nodes which would very likely result in similar communities. A second rule can consider discarding also the neighbors of neighbors and thus guarantees a minimum of two intermediate nodes between two seeds. Each seed will be as a community to process the number of seeds chosen initially impacts the runtime of the algorithm. Hence, reducing the number of seeds is important. Nevertheless, it may also reduce the accuracy of the algorithm. Once communities have been initialized the algorithm begins its loop through all scale parameters. For each scale, while changes can be made the algorithm keeps analyzing the current scale. The implementation from [10] follows two steps. In the first step, communities are grown. In the second step, important overlapping communities are merged. We keep these two steps here with some modifications. First communities are grown at the same time. The community is added to a list of communities to check for merging when it is modified. The second step consists of the checking and merging. All the communities in checking list are processed in the same time to find whether they overlap beyond a merging threshold. When two communities overlap enough they will be added to a merge list. Finally, the list of merged communities is processed. All pairs that have no community in common are merged in parallel. Then references are updated in the remaining communities to merge (e.g if c2 merged into c1, references to c2 are renamed c1) and the parallel merging process is repeated until all pairs of communities have been merged.

D. Multi-scale detection of communities using stability optimization(MD)

As discussed in [10], the partition scale issue and the optimal community identification can be addressed with the help of studying the stability of a partition along with the Markov time. The results from the authors indicate that due to the stability curve, the clustering varies depending on the time window during which the Markov time is considered. This measure takes the graph as a Markov chain where each edge is a possible state transition and each node represents a state. The stability of a graph has been used to assess the results of various modularity optimization algorithms and has been introduced as a measure to evaluate the quality of a partition in the graph. Therefore, the Markov time as a resolution parameter in a greedy optimization context as used in [14], where stability is the optimization criterion. Stability at time t is therefore defined as:

$$Q_{S_t} = trace(R_t) = \Sigma(e_{ii} - a_i^2) \quad (3)$$

where $trace(R_t)$ is the trace of the autocovariance.

E. Stability optimization based on the Louvain method

The optimization methods presented previously explored a variation of the greedy stability optimization where only one Markov time is considered instead of a time window. In addition to that and as given in equation 3, it has been shown that optimizing stability for time t is equivalent to optimizing modularity of the graph with an adjacency matrix At. Therefore the utilization of the Newman's greedy modularity optimization would be the same. However, instead of Newman's algorithm, any other modularity optimization method can potentially be used and have an interesting execution speed, the Louvain method [20] previously mentioned can be a great example of that. The Markov time thus remains the perfect resolution parameter to compute the matrix At, yet it enables the Louvain method to process the resulting network without modifying its code. In order to unify some known clustering heuristics including modularity, stability optimization based on the Louvain enables random walks of variable length defined by the Markov time, employing thus thoroughly the actual topology of the network in the same way as an information flow through a network. This approach appears to be more suitable, as communities reflect the organization of a network, and hence its connectivity. This gives an alternative algorithm to optimize stability. The evaluation of how robust detected stable partitions are, respecting the aggregation algorithm in addition to the Markov time can be enabled by the fact of comparing the results of this combination with the other methods to optimize stability.

The methodology used in this work can be best described by the diagram of Fig.2. First we give an image which is represented as pixel matrix, from this matrix a Complex Network is created, after that we use the community detection algorithms to finally extract the modularity/stability feature measure.

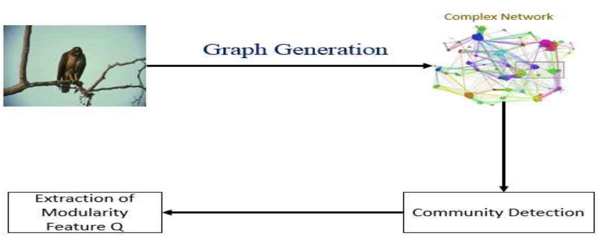

Figure. 2: Extraction of Modularity Feature based on complex network and community detection algorithms.

IV. Anisotropic diffusion filter to Merge small regions

The segmentation algorithms described in the previous section may perform poorly segmentation of the image into communities, actually in the situations in which the community detection algorithm detects regions with a very small area as a community. For example, some small regions may be chosen as results. Hence the anisotropic diffusion, also called Perona and Malik diffusion technique is used before applying the community detection algorithms to merge the small regions in the image and also to obtain a minimal number of the detected communities. The concept of this technique is reducing image noise without removing significant information of the image content, for example, edges, lines or other details that are meaningful for interpreting the image. The diffusion process is a space invariant and linear transformation of the original image. It produces a parameterized images family, but each image result is a combination of the original image and a filter that depends on the local content of the original image as shown in figure 3. As a conclusion, we can say that the process of the anisotropic diffusion algorithm is the pioneering work in partial derivatives equations (PDE)-based denoising, which applies the law of diffusion on pixel intensities to smooth textures in an image. Then a threshold function is used to prevent diffusion to happen across edges, therefore, it preserves edges in the image which make it very interesting if we want to remove noise, without smoothing out the edges of our image.

(a) **(b)**

Figure. 3: (a) Original image, (b) Image after applying the Anisotropic diffusion filter.

V. Experiments and results

This section provides experiments that were performed to assess our algorithm. The proposed methods are carried out on a 2.60 GHz; i5 processor with 4Go RAM on Windows 8 platform. MATLAB 7.13 and image processing toolbox are used. Our experimental study was exhaustively tested on a subset of the Berkeley Segmentation Data Set (BSDS300). This database 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 first experiment, we show how the proposed algorithms can segment the graph of the image into communities. We use the Pajek program for analysis and visualization of the graph and the detected communities, as shown in the figures [4-8], the proposed algorithms especially, Stability optimization based on the Louvain method, Fast multi-scale detection of communities based on Local Criteria and Multi-scale detection of communities using stability optimization, can detect a minimum number of communities compared to infomap and louvain. However as shown in table 1, reducing the number of communities does not mean that the image graph is well partitioned, because some algorithms, the stability optimization based on the Louvain method as an example, can segment the image graph with a minimum number of communities, nevertheless the value of modularity or stability is low, compared to other algorithms.

Algorithms	Number of communi-ties	Modularity
Infomap	98	0.72
Louvain	19	0.75
FMD of communities based on LC	18	0.84
MD of communities using SO	20	0.83
SO based on the Louvain method	8	0.69

***Table 1*: Modularity and the number of communities for the proposed algorithms**

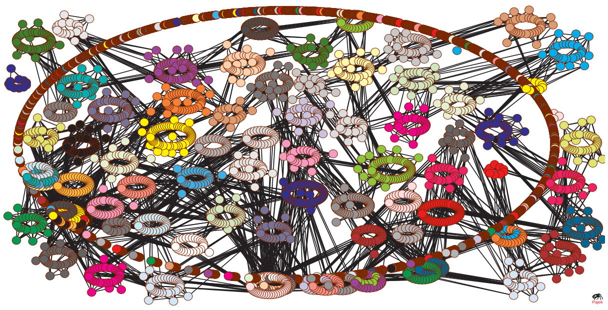

Figure. 4: Infomap algorithm.

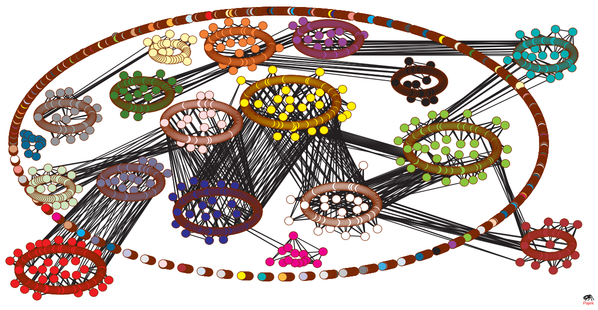

Figure. 5: Louvain algorithm.

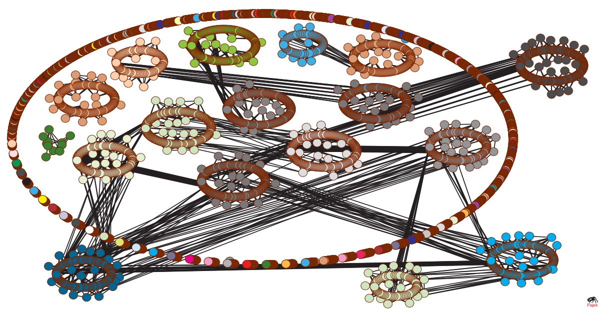

Figure. 6: Fast multi-scale detection of communities based on Local Criteria algorithm.

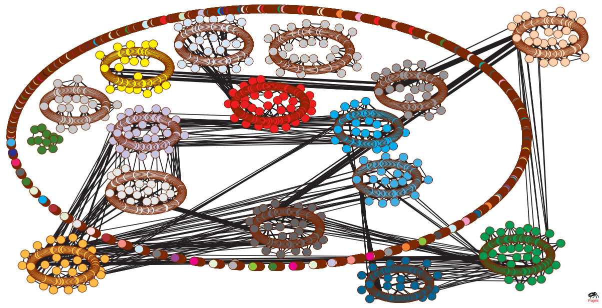

Figure. 7: Fast multi-scale detection of communities using stability optimisation algorithm.

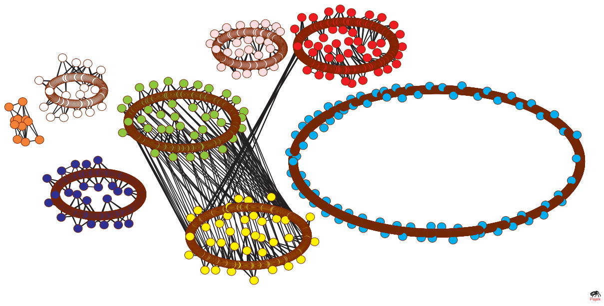

Figure. 8: Satability optimization based on louvain algorithm.

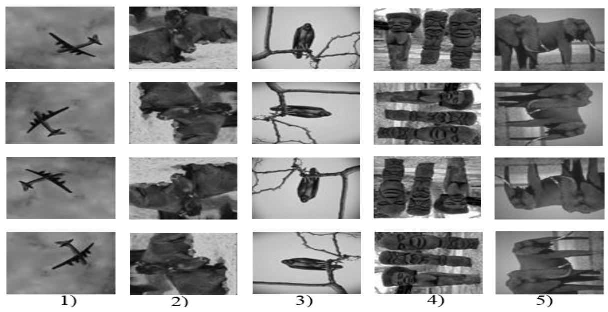

Figure. 9: Test images from Berkeley dataset.

In the second experiment, we tested five images from BS-DS300 (Fig.9), in order to generate four similar images for each type, four affine transforms (rotation 90), were applied to each image. After that, we tested the five images with the proposed community detection algorithms and finally we extract the modularity feature. Table [2-6] interprets the results obtained for several threshold measurement (t= 0;5;10;15). After interpreting the results of tables, we can say that the modularity/stability feature can successfully cluster into one group the same types of images after the four transformations. The results illustrate that the modularity/stability feature is robust to the rotation and to affine transform of images, which is a useful property for image segmentation.

Image 1							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.7089	0	0.7089	0	0.7088	0	0.7090	0
0.7193	5	0.7171	5	0.7170	5	0.7175	5
0.7290	10	0.7272	10	0.7270	10	0.7276	10
0.7190	15	0.7175	15	0.7171	15	0.7173	15
Image 2							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.7689	0	0.7689	0	0.7678	0	0.7690	0
0.7693	5	0.7671	5	0.7570	5	0.7675	5
0.7690	10	0.7672	10	0.7570	10	0.7576	10
0.7690	15	0.7675	15	0.7571	15	0.7573	15
Image 3							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.8033	0	0.8033	0	0.8032	0	0.8033	0
0.8983	5	0.8980	5	0.8961	5	0.8951	5
0.8922	10	0.8913	10	0.8974	10	0.8954	10
0.8804	15	0.8812	15	0.8986	15	0.8976	15
Image 4							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.8733	0	0.8732	0	0.8730	0	0.8734	0
0.8431	5	0.8433	5	0.8333	5	0.8433	5
0.8433	10	0.8631	10	0.8432	10	0.8433	10
0.9400	15	0.8428	15	0.8539	15	0.8419	15
Image 5							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.9233	0	0.9232	0	0.9200	0	0.9234	0
0.9431	5	0.9333	5	0.9333	5	0.9033	5
0.9433	10	0.9331	10	0.9332	10	0.9033	10
0.9400	15	0.9328	15	0.9339	15	0.9019	15

***Table 2*: Infomap algorithm for five test images**

Image 1							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.9033	0	0.9033	0	0.9032	0	0.9033	0
0.9004	5	0.9028	5	0.8961	5	0.9051	5
0.9022	10	0.9013	10	0.8974	10	0.8954	10
0.9004	15	0.9012	15	0.8986	15	0.9001	15
Image 2							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.8033	0	0.8033	0	0.8032	0	0.8033	0
0.8983	5	0.8980	5	0.8961	5	0.8951	5
0.8922	10	0.8913	10	0.8974	10	0.8954	10
0.8804	15	0.8812	15	0.8986	15	0.8976	15
Image 3							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.9233	0	0.9233	0	0.9232	0	0.9233	0
0.9431	5	0.9333	5	0.9333	5	0.9033	5
0.9433	10	0.9331	10	0.9332	10	0.9033	10
0.9400	15	0.9328	15	0.9339	15	0.9019	15
Image 4							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.9732	0	0.9732	0	0.9730	0	0.9731	0
0.9631	5	0.9633	5	0.9628	5	0.9633	5
0.9633	10	0.9631	10	0.9624	10	0.9620	10
0.9600	15	0.9628	15	0.9639	15	0.9619	15
Image 5							
Original		Rotation 90		Rotation 180		Rotation 270	
Q	t	Q	t	Q	t	Q	t
0.9530	0	0.9530	0	0.9530	0	0.9531	0
0.9531	5	0.9533	5	0.9533	5	0.9433	5
0.9530	10	0.9533	10	0.9532	10	0.9532	10
0.9532	15	0.9532	15	0.9533	15	0.9533	15

***Table 3*: Louvain algorithm for five test images**

Image 1							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9779	0	0.9753	0	0.9728	0	0.9744	0
0.9727	5	0.9701	5	0.9687	5	0.9692	5
0.9765	10	0.9779	10	0.9705	10	0.9780	10
0.9754	15	0.9698	15	0.9694	15	0.9680	15
Image 2							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9221	0	0.9321	0	0.9325	0	0.9222	0
0.9530	5	0.9524	5	0.9534	5	0.9506	5
0.9448	10	0.9420	10	0.9433	10	0.9405	10
0.9315	15	0.9308	15	0.9252	15	0.9244	15
Image 3							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8223	0	0.8225	0	0.8222	0	0.8236	0
0.8573	5	0.8540	5	0.8500	5	0.8494	5
0.8892	10	0.8888	10	0.8884	10	0.8894	10
0.8868	15	0.8850	15	0.8862	15	0.8856	15
Image 4							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8733	0	0.8842	0	0.8850	0	0.8824	0
0.8681	5	0.8613	5	0.8663	5	0.8613	5
0.8503	10	0.8681	10	0.8642	10	0.8625	10
0.9500	15	0.9408	15	0.9509	15	0.9428	15
Image 5							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9365	0	0.9242	0	0.9211	0	0.9200	0
0.9431	5	0.9433	5	0.9403	5	0.9213	5
0.9409	10	0.9342	10	0.9452	10	0.9223	10
0.9448	15	0.9428	15	0.9401	15	0.9024	15

***Table 4*: Fast multi-scale detection of communities based on Local Criteria algorithm for five test images**

Image 1							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9687	0	0.9663	0	0.9698	0	0.9684	0
0.9741	5	0.9732	5	0.9720	5	0.9771	5
0.9679	10	0.9703	10	0.9701	10	0.9715	10
0.9677	15	0.9688	15	0.9698	15	0.9650	15
Image 2							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9301	0	0.9300	0	0.9321	0	0.9322	0
0.9220	5	0.9210	5	0.9204	5	0.9201	5
0.9177	10	0.9171	10	0.9163	10	0.9185	10
0.9155	15	0.9138	15	0.9142	15	0.9100	15
Image 3							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8603	0	0.8555	0	0.8542	0	0.8526	0
0.8563	5	0.8580	5	0.8570	5	0.8554	5
0.8902	10	0.8918	10	0.8924	10	0.8914	10
0.8842	15	0.8845	15	0.8844	15	0.8835	15
Image 4							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8763	0	0.8752	0	0.8780	0	0.8774	0
0.8481	5	0.8843	5	0.8833	5	0.8813	5
0.8543	10	0.8661	10	0.8442	10	0.8485	10
0.9389	15	0.8418	15	0.8599	15	0.8458	15
Image 5							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9398	0	0.9352	0	0.9120	0	0.9324	0
0.9201	5	0.9413	5	0.9393	5	0.9143	5
0.9399	10	0.9332	10	0.9422	10	0.9143	10
0.9458	15	0.9338	15	0.9389	15	0.9259	15

***Table 5*: Multi-scale detection of communities using stability optimization algorithm for five test images**

Image 1							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9659	0	0.9663	0	0.9678	0	0.9684	0
0.9647	5	0.9651	5	0.9667	5	0.9662	5
0.9675	10	0.9679	10	0.9695	10	0.9680	10
0.9654	15	0.9658	15	0.9674	15	0.9660	15
Image 2							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9021	0	0.9021	0	0.9025	0	0.9022	0
0.9030	5	0.9024	5	0.9034	5	0.9026	5
0.9028	10	0.9020	10	0.9033	10	0.9025	10
0.9035	15	0.9028	15	0.9032	15	0.9034	15
Image 3							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8223	0	0.8225	0	0.8222	0	0.8236	0
0.8483	5	0.8480	5	0.8470	5	0.8474	5
0.8832	10	0.8828	10	0.8824	10	0.8834	10
0.8838	15	0.8830	15	0.8832	15	0.8826	15
Image 4							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.8733	0	0.8732	0	0.8730	0	0.8724	0
0.8431	5	0.8433	5	0.8333	5	0.8423	5
0.8433	10	0.8631	10	0.8432	10	0.8425	10
0.9400	15	0.8428	15	0.8539	15	0.8428	15
Image 5							
Original		Rotation 90		Rotation 180		Rotation 270	
Qs	t	Qs	t	Qs	t	Qs	t
0.9333	0	0.9232	0	0.9200	0	0.9224	0
0.9331	5	0.9333	5	0.9323	5	0.9033	5
0.9339	10	0.9232	10	0.9322	10	0.9023	10
0.9348	15	0.9228	15	0.9329	15	0.9019	15

***Table 6*:** Stability optimisation algorithm for five test images

For the image segmentation task as discussed in [27], the infomap algorithm is not appropriate, compared to the other algorithms, because as shown in Fig.10 the infomap algorithm does not always give the best segmentation. In addition to that, it induces an over-segmented image, which is a very large amount of communities. So, the results presented in our experiments are based on the louvain, Fast multi-scale detection of communities based on Local Criteria, Multi-scale detection of communities using stability optimization and Stability optimization based on the louvain algorithm, only because they are more appropriate than infomap for the image segmentation task.

We have performed comparisons of our proposed algorithms with some existing segmentation methods. In Fig.11 the result of the four proposed algorithms is compared to the three mentioned segmentation techniques. We can notice that these algorithms have achieved much subjectively better results compared to other techniques, and in many cases can separate the main object of the image correctly. We can also see that our proposed algorithms specially Stability optimization based on the louvain and Multi-scale detection of communities using stability optimization methods produces sizeable segments for all selected images. In this case, our proposed algorithms achieve object-level segmentation to some extent.

We Also quantitatively evaluate the segmentation performance of the four proposed algorithms. The segmentation results are compared with the three segmentation techniques, we investigate for the quantitative evaluation the Probabilistic Rand Index [28] which are described as below:

The PRI is a classical evaluation criterion for clusterings, its measures the consistency of labelings between a segmentation and its ground truth by the ratio of pairs of pixels having the same labels, averaging across multiple ground truth segmentation to account for variation in human perception. The range of PRI is [0,1]; with a larger value indicating greater similarity between two segmentations. Table 7 presents the average values of the PRI, which are calculated, when the EDISON, JSEG, MULTISCALE and the proposed methods, were applied to all of the 100 images in the Berkeley segmentation dataset.

Algorithms	PRI
Humain	0.87
FMD of communities based on LC	**0.818**
MD of communities using SO	**0.801**
Louvain	0.788
Stability Optimization based on Louvain	0.653
Infomap	0.732
EDISON	0.786
JSEG	0.760
MULTISCALE	0.752

***Table 7*:** Quantitative comparison of different algorithms on Berkeley dataset

The results in table 7 indicate the superiority of Fast multi-scale detection of communities based on Local Criteria and Multi-scale detection of communities using stability optimization over other popular methods, the third and forth rows show the performance of the proposed algorithms compared to the three popular methods, also the two proposed methods achieve the best performance among all the popular segmentation algorithms. In terms of PRI, Fast multi-scale detection of communities based on Local Criteria and Multi-scale detection of communities using stability optimization has a close performance to human and outperform the rest of other algorithms.

VI. CONCLUSION

Image segmentation approach is the most fundamental step to clustering an image into salient image regions, i.e, regions corresponding to individual surfaces, objects, or natural parts of objects. In this work we proposed a new image representation based on the graph, we have proposed the most efficient community detection algorithms, these algorithms take advantage of the stability of modularity and Stability optimization. In the first experiment, we showed how the proposed algorithms can segment the image graph into communities, in the second one, the modularity and stability feature were shown to be robust and have the potential to be effective and efficient features for the image segmentation problem. Our preliminary results show that computation feature could be a useful component of image segmentation tasks. Although we still need further analysis and experiments to understand the drawbacks and advantages of our modularity/stability features. In the third experiment, we evaluated the proposed methods with three other segmentation methods in the literature, the proposed methods were exhaustively tested on a subset of the Berkeley Segmentation dataset. It is reported that the proposed algorithms achieve the best performance among all of the other experimented popular methods in terms of PRI on the Berkeley dataset. The qualitative results showed that the proposed methods have the ability to segment the input image into an optimal number of segments, as well as a number of segments defined by the user.

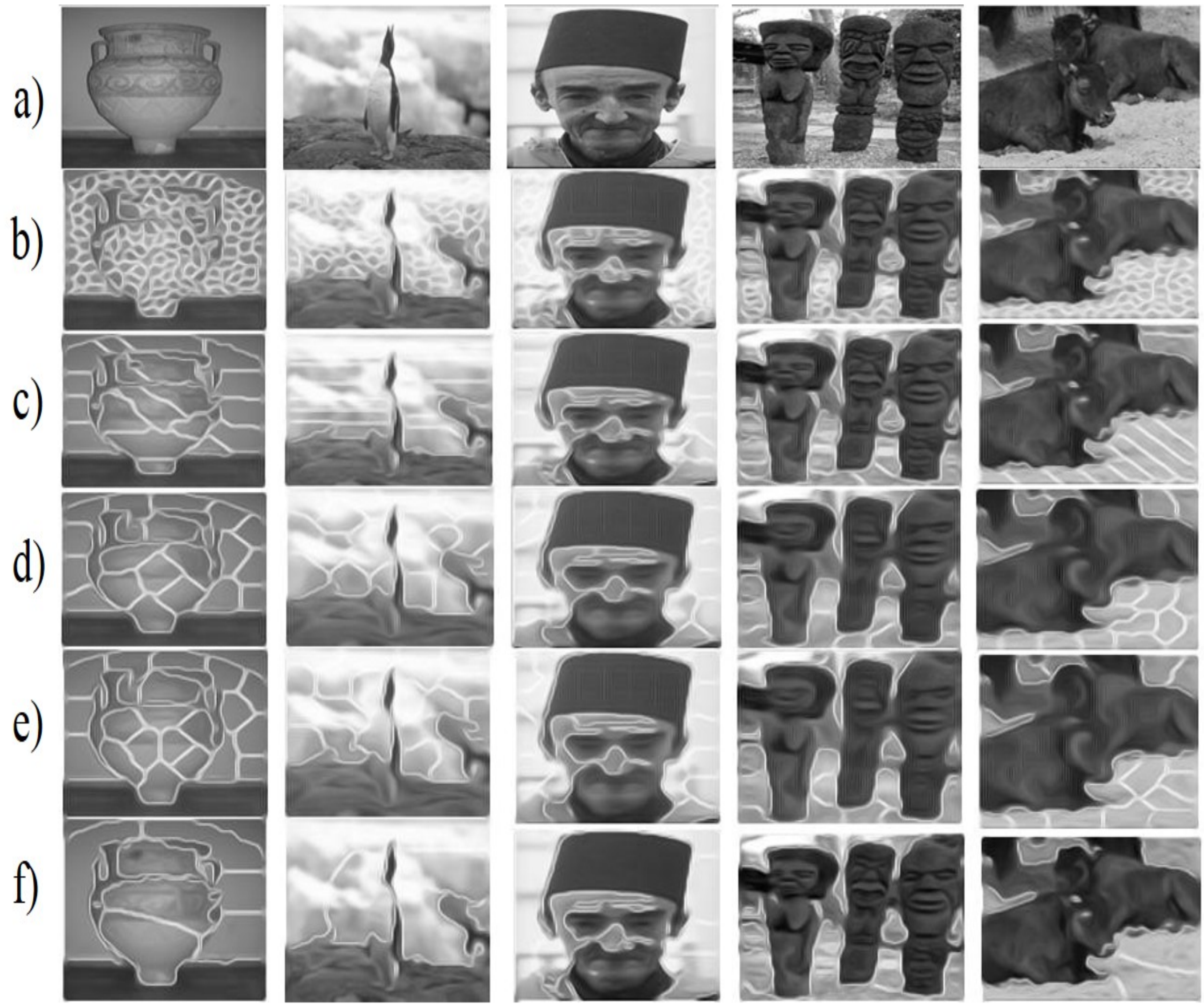

Figure. 10: a) Original images from Berkeley database; b) Segmented images with Infomap algorithm; c) Segmented images with Louvain algorithm; d) Segmented images with Fast multi-scale detection of communities based on Local Criteria; e) Segmented images with Multi-scale detection of communities using stability optimization; f) Segmented images with Stability optimization algorithm

Also our results could be a useful component of the image segmentation task.

Acknowledgment

This work has been supported by the project CNRS-CNRST STIC 02/14

References

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

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

[3] J. B. Roerdink and A. Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," *Fundam. Inform.*, vol. 41, no. 1-2, pp. 187–228, 2000.

[4] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience*, vol. 10, no. 3, pp. 186–198, 2009.

[5] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: uses and interpretations," *Neuroimage*, vol. 52, no. 3, pp. 1059–1069, 2010.

[6] I. K. Park, I. D. Yun, and S. U. Lee, "Color image retrieval using hybrid graph representation," *Image and Vision Computing*, vol. 17, no. 7, pp. 465–474, 1999.

[7] B. Xu, J. Bu, C. Chen, C. Wang, D. Cai, and X. He, "Emr: a scalable graph-based ranking model for

a
b
c
d
e
f
g
h

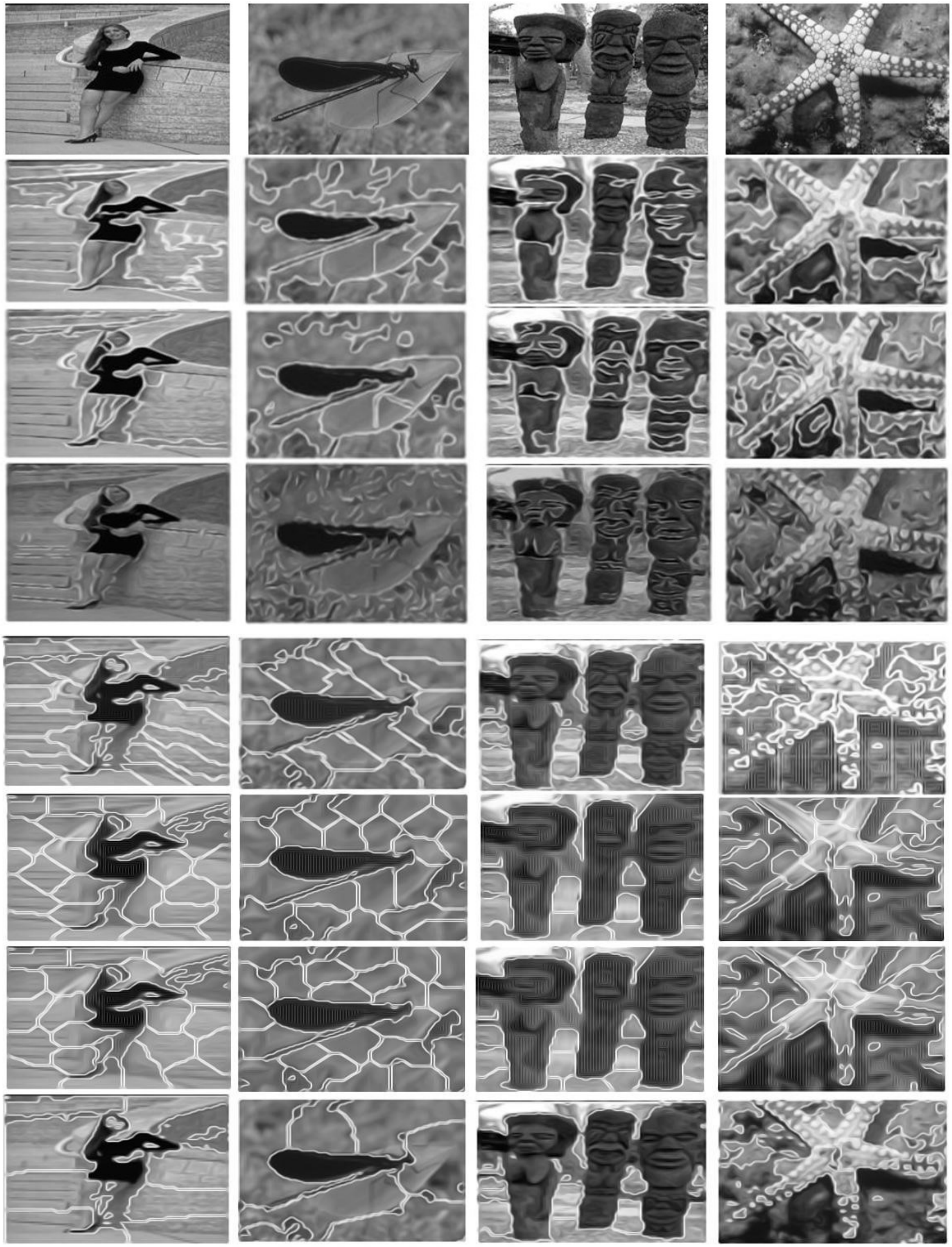

Figure. 11: a) Input images; b) EDISON; c) JSEG; d) MULTISCALE; e) Louvain; f) Fast multi-scale detection of communities based on Local Criteria; g) Multi-scale detection of communities using stability optimization; h) Stability optimization.

content-based image retrieval," vol. 27, no. 1. IEEE, 2015, pp. 102–114.

[8] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei, "Image retrieval using scene graphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3668–3678.

[9] A. Broumandnia, M. Cheraghi, and M. Azararjmand, "Content-based image retrieval with graph theoretic approach."

[10] 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. 12 755–12 760, 2010.

[11] A. Arenas, A. Fernández, and S. Gómez, "Analysis of the structure of complex networks at different resolution levels," *New Journal of Physics*, vol. 10, no. 5, p. 053039, 2008.

[12] J. Huang, H. Sun, Y. Liu, Q. Song, and T. Weninger, "Towards online multiresolution community detection in large-scale networks," *PloS one*, vol. 6, no. 8, p. e23829, 2011.

[13] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, p. 033015, 2009.

[14] E. L. Martelot and C. Hankin, "Multi-scale community detection using stability optimisation within greedy algorithms," *arXiv preprint arXiv:1201.3307*, 2012.

[15] 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.

[16] R. V. Mendes, "Community detection and spectral clustering."

[17] M. E. Newman, "Community detection and graph partitioning," *EPL (Europhysics Letters)*, vol. 103, no. 2, p. 28003, 2013.

[18] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[19] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[20] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[21] E. L. Martelot and C. Hankin, "Multi-scale community detection using stability optimisation within greedy algorithms," *arXiv preprint arXiv:1201.3307*, 2012.

[22] 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.

[23] 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.

[24] 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.

[25] B. Sumengen and B. Manjunath, "Multi-scale edge detection and image segmentation," in *Signal Processing Conference, 2005 13th European*. IEEE, 2005, pp. 1–4.

[26] M. E. Newman, "Analysis of weighted networks," *Physical Review E*, vol. 70, no. 5, p. 056131, 2004.

[27] Y. Mourchid, M. ElHassouni, and H. Cherifi, "A new image segmentation approach using community detection algorithms," *The International Conference on Intelligent Systems Design and Applications (ISDA), number=6, year=2015, publisher=IEEE.*

[28] 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.