An Unsupervised Approach for Boundary Detection of Colored Natural Scenes

Karin Satie Komati
Department of Informatics
IFES, Campus Serra
Serra, Brazil
Email: kkomati@ifes.edu.br

Evandro O. T. Salles, Mário Sarcinelli-Filho
Graduate Program on Electrical Engineering
Federal University of Espirito Santo (UFES)
Vitoria, Brazil
Email: evandro@ele.ufes.br, mario.sarcinelli@ufes.br

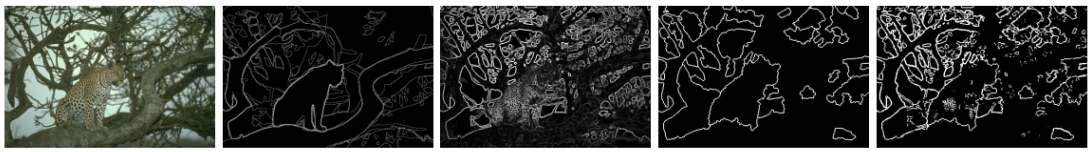

Fig. 1. Teasing result of our method: from this data input and the human benchmark (first two images at left), the result of classical edge detection and MM-Frac (middle), producing effective result (the rightmost image).

***Abstract*—This work proposes a new unsupervised method for boundary detection in colored natural scenes, consisting of two-stage sequential processes of integration. The first stage performs region extraction, where two different techniques measure the color-texture homogeneity in a region-growing method and they are combined by two different control algorithms. One control algorithm is based on a local property and the other is based on a global statistical property (the shape of the power spectrum of the image being analyzed). One homogeneity measure is the J-value and the second is a multifractal descriptor. In the second integration, edge information is extracted by a classical method, and integrated with region information. This process eliminates false boundaries in the region-map, guided by the edge-map, and reduces the noise in the edge-map as well, now guided by the region-map, thus taking advantage of their complementary nature. It integrates the two maps into a single final result, enhancing the coincident information of both maps. Each phase of integration improves, progressively, the detection of the boundaries. Experiments on a large dataset of natural color images BSDS suggest that the results for this approach proposed are closer to the human perception, quantitatively and qualitatively, than the individual methods.**

***Keywords*-boundary detection, multifractal measurement, J value, 1/f spectrum, region-growing, edge detection.**

I. Introduction

Boundary detection is one of the most important tasks in computer vision. To propose a fully automated and unsupervised boundary detection system is a complex task, since it is not known a priori what types of regions (uniform, with smooth color gradation and texture variations) exist in an image, or even how many regions a given scene contains. Examples of the variety and complexity of natural images are given in Fig. 2. In the teaser image (Fig. 1, showing a feline in a tree), the tangle of branches is an example of complexity. In Fig. 2(a) (a snake in a desert) the central element and the background have almost the same color, causing a ill-defined border. In Fig. 2(b) there is a mixture of artificial and natural textures (the bridge has geometric patterns, quite unlike the natural texture of the hill and the trail of smoke produced by the locomotive). Fig. 2(c) (the eagle) presents varying shades, specially in the corners, due to illumination.

(a) (b) (c)

Fig. 2. Image examples (extracted from [1]).

Traditionally, the image segmentation techniques are classified as region or edge approaches. Region-based techniques rely on common patterns in intensity values within a cluster of neighboring pixels. The cluster is referred to as the region, and the goal of the segmentation algorithm is to group regions according to their anatomical or functional roles. Region-approach results tend to be over-segmented, with inaccurate boundaries. On the other hand, edge-based techniques rely on discontinuities in image values between distinct regions, and the goal of the segmentation algorithm is to accurately demarcate the boundary separating these regions. However, there may exist gaps and noisy edges in edge-approach results.

There are many proposals combining the outputs of region-growing and edge detection methods to improve the quality of their results. Muñoz et al. [2] show seven different strategies for combining similarity (region) and discontinuity (edge)

information. They were grouped in two classes: embedded integration and post-processing integration. Embedded integration produces, in general, a single complex algorithm to avoid errors in the results. The post-processing strategy works with a set of many algorithms. This approach accepts faults in the elementary algorithms, and the posteriori integration module tries to correct these errors.

A. Technique overview

In this work, these two classes are considered in two sequential levels, thus building the whole process, which is shown in Fig. 3.

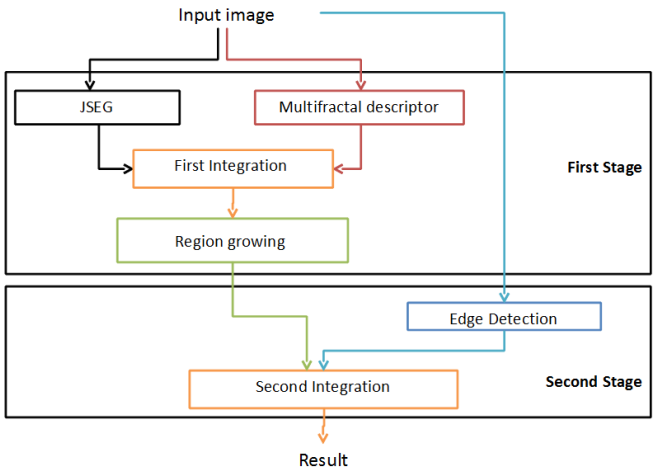

Fig. 3. Simplified architecture.

In the first level of integration we propose a new idea by combining the multifractal descriptor with a JSEG algorithm, thus enhancing the detection of boundary regions.

Four different integration architectures were implemented and tested along this research. The first two, Fractal-*only* and Fractal-JSEG, were presented in [3], [4]. Fractal-*only* computes the homogeneity map using only the multifractal value and the Fractal-JSEG mixes the first attempt and the original J-image, using a maximum function to combine them. The third system, called I-Frac [5], uses a more intelligent heuristic to combine the values of the two different measurements, instead of using the maximum function. In the fourth architecture, MM-Frac [6], the integration is based on a global image statistical property. Such statistical property is also used to calibrate the threshold of the merging process. This last system presented better results, decreasing the over-segmentation, and, therefore, just the results of MM-Frac will be considered in this paper.

The region-growing result of MM-Frac and edge information are independently extracted, in a parallel way. Our strategy is to put the two maps together, eliminating the false boundaries in the region map, based on edge information, and eliminating the noisy edges in the edge map, based on region information. Such method is hereinafter referred to as KoSS [7]. In the sequel, we show that the resulting image is closer to human perception than any of the two images used as input for the post-processing integration. Finally, it deserves mentioning that KoSS algorithm is an improved version of the KSS algorithm [8], previously proposed by the same authors.

II. The Proposed Method

A. First-Level of Integration

1) J value: The essence of the JSEG method is to separate the segmentation process in two independently processed stages: color quantization and spatial segmentation. The result of color quantization is a class-map which associates a color class label to each pixel belonging to a class.

In the spatial segmentation stage, a criterion to measure the distribution of color classes, the J measure, is calculated. Essentially, it measures the distances between different classes, divided by the distances between the members within each class, an idea similar to the Fisher's multi-class linear discriminator. The J value can be calculated by using a local area of the class-map. Multi-scale J-images are calculated changing the local window size. In the J-image, the higher the local J value is, the more likely the pixel is part of a boundary region, like a 3-D terrain map containing valleys and mountains. Then, a region growing method is used to segment the image. Finally, to overcome the over-segmentation problem, regions are merged based on their color similarities, by directly applying a Euclidean distance measure.

2) The Multifractal Measurement: In this work, we will use the differential box-counting method, proposed by Chaudhuri and Sarkar [9], to estimate the multifractal measurement (MM) of the original image.

The MM of a single pixel is calculated in a small window surrounding it, generating a Fractal-image for each channel in Luv color space. The Fractal-images are also a 3D terrain maps, that is because the MM in the border regions of a texture is lower than the MM of a homogeneous region [10]. Each value in Fractal-image is converted to be higher in boundary regions and to have the same limits applied to a J-image.

3) 1/f Spectra of Natural Images: Statistics of natural images have been found to follow particular regularities. Torralba and Oliva [11], studying the statistics of real-world images, observed that the energy spectra of such images falls, in average, into a form $1/f^{\alpha}$ with $\alpha \sim 2$. They also show that the shape of the power spectrum can be used to categorize the different semantic of scenes (single objects, rooms, places, large outdoors and panoramic scenes).

Here α represents the slope of the decreasing energy spectrum values, from low to high spatial frequencies, varying with the scene complexity. Fig. 4 exemplifies a 3D power spectrum, where the slope is emphasized in red. Fig. 5 shows the slope (red) in a 2D graphic and the interpolated slope (the dotted black line). The estimated $-\alpha$ value is then -2.31, or α value is then $+2.31$.

Pentland [10] showed that fractal natural surfaces (as mountains, forests) produce a fractal image with an energy spectrum of the form $1/f^{\alpha}$, where α is related to the fractal dimension of the 3D surface (e.g., its roughness). Slope characteristics

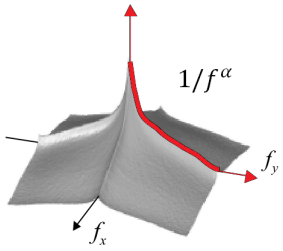

Fig. 4. Graphic of one image power spectrum 3D.

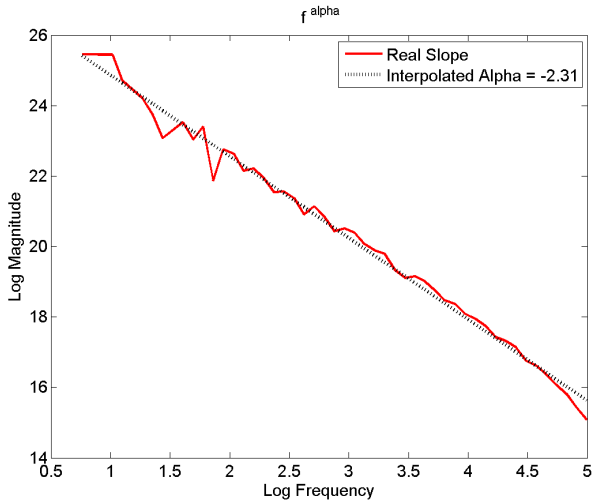

Fig. 5. Graphic of one image power spectrum 2D.

may be grouped in two main families, a slow slope ($\alpha \sim 1$), for environments with textured and detailed objects, and a steep slope ($\alpha \sim 3$), for scenes with large objects and smooth edges. Thus, the slower the slope is, the more textured the image is.

4) MM-Frac: In this new proposal, the integration of two measurements, J-image and Fractal-image, is controlled by the value of α as in the work of Côco, Salles, Sarcinelli-Filho [12]. Fig. 6 shows a simplified architecture of the proposed MM-Frac system. The global estimated value α controls two process:

1) the local integration of the J-value and Multifractal Measurement (MM). Each pixel of the 3D terrain map is now calculated as:

$$map_{ij} = \textit{J-value} \times \alpha_{norm} + (1 - \alpha_{norm}) \times \textit{MM}, \quad (1)$$

where $\alpha_{norm} = \alpha / max(\alpha_i)$, i indexing the 200 images used as training set (provided by BSDS). For low α values, the image presents more texture, and the multifractal weight is greater than that of the J-value, as multifractal models textures in a better way than the J-value;
2) the threshold used in region merging is $(0.4\alpha_{norm})$, where 0.4 is the default value for the JSEG method. The lower the threshold is, lesser regions will be merged, and the segmentation result will present more regions with a lower threshold, compared to a higher threshold. An image with high α value presents large objects and smooth edges, so it is expected that the segmentation result will present just a few regions.

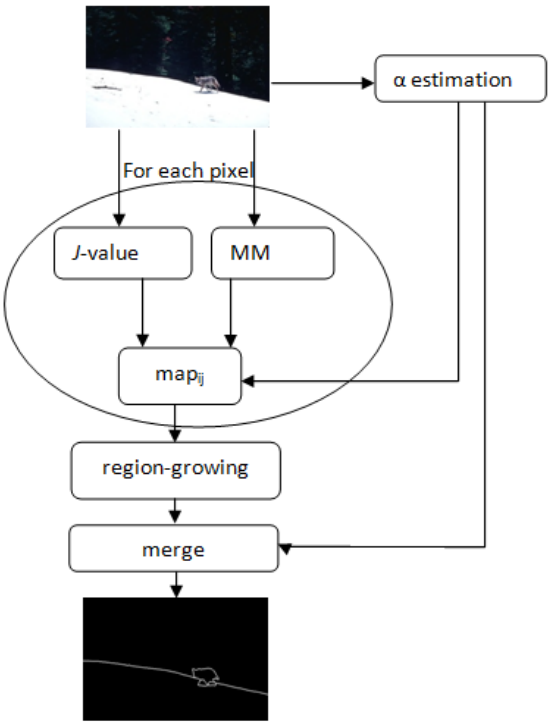

Fig. 6. Simplified archtecture of MM-Frac.

B. Second-Level of Integration

1) Edge Detection: We considered here some classical edge detectors (Sobel, Prewitt, Laplacian and morphological gradient), which generates an output known as a soft boundary map, with each pixel valued from zero to one, where higher values mean greater confidence in the existence of a boundary. To choose a good edge detector, it was made a preliminary test and the morphological gradient presents the overall F-measure slightly better than the other detectors. Therefore, it was chosen as the edge detection method.

It is quite usual to smooth the image to eliminate noise before the edge detection. We choose a classical non-linear edge-preserving-smoothing filter, the Kuwahara filter with 5×5 mask size. To process a color image, each of three color channels, RGB, is processed separately, and then all results are added into one image.

2) KoSS: The integration method is independent of how the edge-map and region-map are processed. However, it is necessary that the region-map be a binary image and the edge-map be a soft map. Fig. 7 shows a simplified architecture of the proposed KoSS system and the algorithm is presented as a pseudocode in Algorithm 1.

In step 2, we should detect the weak edges from the edge map. This step is basically a binarization process in the edge

map, where each pixel with a low gray level value corresponds to a weak edge pixel. To automate the threshold value, we use the results of [13], where the threshold value is based on the histogram h of the edge-map, given by

$$threshold_{weak} = \frac{\sum_{i=0}^{50} h_i}{\sum_{i=50}^{200} h_i}, \quad (2)$$

where i = [0, 255] is the value of a pixel in a gray-scale image.

A noisy edge map will result in low $threshold_{weak}$ values, while a strongly defined edge map will result in high $threshold_{weak}$ values. So, when the image is noisy, most information from the region map is preserved. Therefore, equation (2) represents the degree of confidence of the edge detection result, pointing when edge map information is more reliable than region map information.

In Step 3 of Algorithm 1, the region map is divided in a list of edge lines. Actually, the region map can be viewed as an skeleton, whose elements can be classified as end points, normal points and branch points [14]. In a 3×3 neighborhood, end points have only one neighbor element, normal points have exactly two and branch points have more than two. An edge-line (or skeleton branch) is a subset of the skeleton entirely consisting of normal points except for the extremes, that are end points or branch points.

Algorithm 2 shows the code correspondent to such step, in Matlab. In line 1, the goal is to find all branch points in an image, and erase all of them (see step 2 of Algorithm 2), thus resulting an image having a set of edge-lines with only

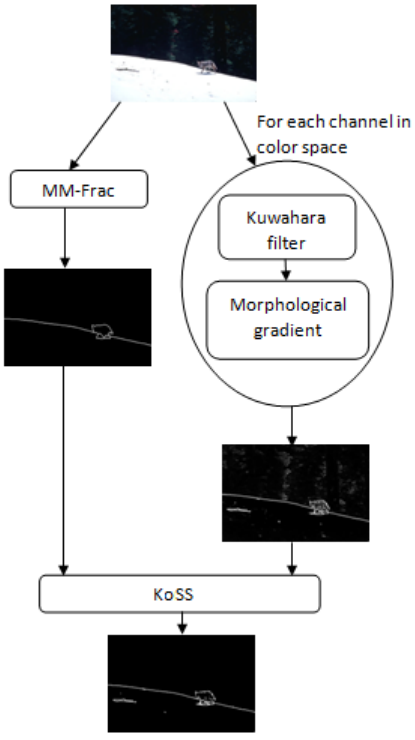

Fig. 7. Simplified archtecture of KoSS.

Algorithm 1 KoSS

```
1:  Inputs: edge-map and region-map
2:  Build a weak-edge-map from edge-map
3:  Build a list-of-edge-lines from region-map
    {Part I}
4:  for each edge-line in the list-of-edge-lines do
5:     set count-weak-edge = 0
6:     for each edge-unit in the edge-line do
7:        if (majority of neighborhood of edge-unit position in
          weak-edge-map) is marked then
8:           increment count-weak-edge
9:        end if
10:    end for
11:    if (count-weak-edge > length(edge-line)/3) then
12:       erase edge-line from region-map
13:    end if
14: end for
    {Part II}
15: for each weak-edge-unit on weak-edge-map do
16:    if (weak-edge-unit position is not marked on region-
       map) then
17:       erase weak-edge-unit from edge-map
18:    end if
19: end for
20: Set image-result = adjustLimit(edge-map + region-map)
```

Algorithm 2 Step 3 of Algorithm 1 in Matlab

```
1: Ipoints = bwmorph(MMFracImage, 'branchpoints', 1);
2: MMFracImage(Ipoints) = 0;
3: lineStats = regionprops(MMFracImage, 'PixelList', 'Pix-
   elIdxList');
```

end points as extremes. The result of line 3 is the variable **lineStats**, containing the list of edge-lines.

In the rest of Algorithm 1, the logic is to eliminate or to reduce false information. Part I (lines 4-14) eliminates edge-lines which are considered weak by the edge detection. The condition of line 11 was empirically defined after analyzing the results of several experiments. In Part II (lines 15-19), the weak information of the edge map is eliminated. All pixels not belonging to an edge line on region-map and considered weak edge on weak-edge-map are erased. In step 20, the sum operation will enhance all boundary pixels that match in the two different processed maps.

Thus, KoSS erases some edge information of the region map and does not preserve weak information for the edge map. The result seems cleaner, preserving and enhancing only the strong edges of both maps.

III. Experimental Results and Discussion

We tested our proposed method with natural colored images provided by the The Berkeley Segmentation Dataset and Benchmark (BSDS) [1], applying it to all one hundred images of the test dataset. The BSDS binarize the boundary map at

many levels, according to the threshold parameter (the chosen value is 10).

Fig. 8 shows some results, where (a) shows the input image, (b) the human benchmark and the segmentation result of (c) corresponds to JSEG, (d) MM-Frac, (e) morphological gradient edge detection and (f) result of KoSS process, already binarized in the best threshold computed by the BSDS. Each result has its computed F-measure metric.

In a qualitatively comparison, the original JSEG algorithm tends to over-segment images, splitting objects into several smaller regions. The MM-Frac approach, by its turn, significantly decreases over-segmentation. For an example, the trees in the background in the first image are not segmented as in the JSEG result. Moreover, the results present more accurate boundaries when compared to the human benchmark. In the second line, the boundary encompasses the entire body of the snake and not fragments.

In the fifth column, the results of edge detection are presented. The results are very noisy and this is mainly due to the fact that edge detection techniques rely entirely on the local information available in the image. The edge-map responds to all contrast variations over the texture regions, like in the sand area on the second line of Fig. 8. At the same time, the method of edge detection is responsible for highlighting details such as the stick in the left in the snow area in the first line of Fig. 8 and the insect near to the snake in the second image of Fig. 8. The results for the KoSS method are presented in the last column. Details detected using edge detection method are kept, but the noise was attenuated and disappears after the binarization computed by BSDS. Now, the boundaries are more accurate and are closer to the human perception.

Deng and Manjunath [15] pointed out that the major problem they observed in JSEG result is caused by the varying shades due to the illumination. For instance, the color of a sky can vary in a very smooth transition as in the image BSDS image 42049, the third line of Fig. 8. Visually, there is no clear boundary. However, the JSEG result presents a circle region in the image. The human perception does not perceive this smooth varying of color as a different region. The result after KoSS does not present this false boundary. The smooth is not perceived by the edge detection, and then the boundary is erased by the KoSS method.

The false boundaries elimination can be observed in the fourth line of Fig. 8. The edge-line in the sky over the smoke in the MM-Frac result, which is not perceived by human peception, is fully erased in the KoSS result.

Quantitative performance comparison requires ground truth and well defined metrics. Both requirements can be found in BSDS. For each image in the BSDS, there are at least five hand-labeled segmentations made by human beings, which constitute the ground truth. The standard metrics of BSDS are precision, recall and F-measure, determining how well the boundary map approximates the human ground truth boundaries [16]. The metrics recall, precision and F-measure of each method computed by the BSDS are tabbed in the superior part of Table I.

TABLE I
PRECISION, RECALL AND F-MEASURE METRICS CALCULATED BY BSDS.

	Recall	Precision	F-measure
JSEG	0.61	0.56	0.58
Edge Detection	0.65	0.49	0.56
MM-Frac	0.63	0.56	0.59
KoSS	0.69	0.54	0.61
Human	0.70	0.89	0.79

BSDS computes the maximum F-measure value across the precision-recall curve, for which each point corresponds to an image in the test dataset. MM-Frac approach improves the recall metric without decreasing precision, thus raising the F-measure score a little bit. Edge detection looses in terms of precision, because of the noisy pixels. After KoSS method, the F-measure increases to 0.61, this is the closest value, comparing to the human perception.

IV. CONCLUSION

This work proposes a new two-level approach to boundary detection for natural color images. In the first level we embedded a MM in the classical JSEG algorithm. The integration, called MM-Frac, is controlled by the slope of the image power spectrum. One conclusion is that the MM improves the sensitivity to boundary regions, thus providing segmentation results that match the human perception better than the segmentation results associated with the original JSEG algorithm.

In the second-level, the post-processing integration, the main goal is to integrate the region-growing result from MM-Frac and edge information. Our strategy, called KoSS, is to put together the two maps, eliminating the false boundaries in region-map, based on edge information, and eliminating the noisy edges in the edge-map, based on region information. It integrates the two maps into a single final result, enhancing the coincident information of both maps.

The KoSS algorithm works well and solves the problem of false boundaries pointed out in other works. Furthermore, all strong edges of both input maps are held, improving the boundary detection. Unfortunately, the KoSS results present broken edges, not keeping the contour closed.

The conclusion is that the two-level approach proposed here improves the boundary detection results, generating segmented images that match the human perception better than the results associated to the individual methods used in the architecture. Each phase of integration improves, progressively, the detection of boundaries.

ACKNOWLEDGMENT

The authors thank CAPES for the scholarships of Mrs. Komati. They also thank PPGEE (UFES), for supporting research development.

REFERENCES

[1] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings of the 8th IEEE International Conference on Computer Vision*, vol. 2, July 2001, pp. 416–423.

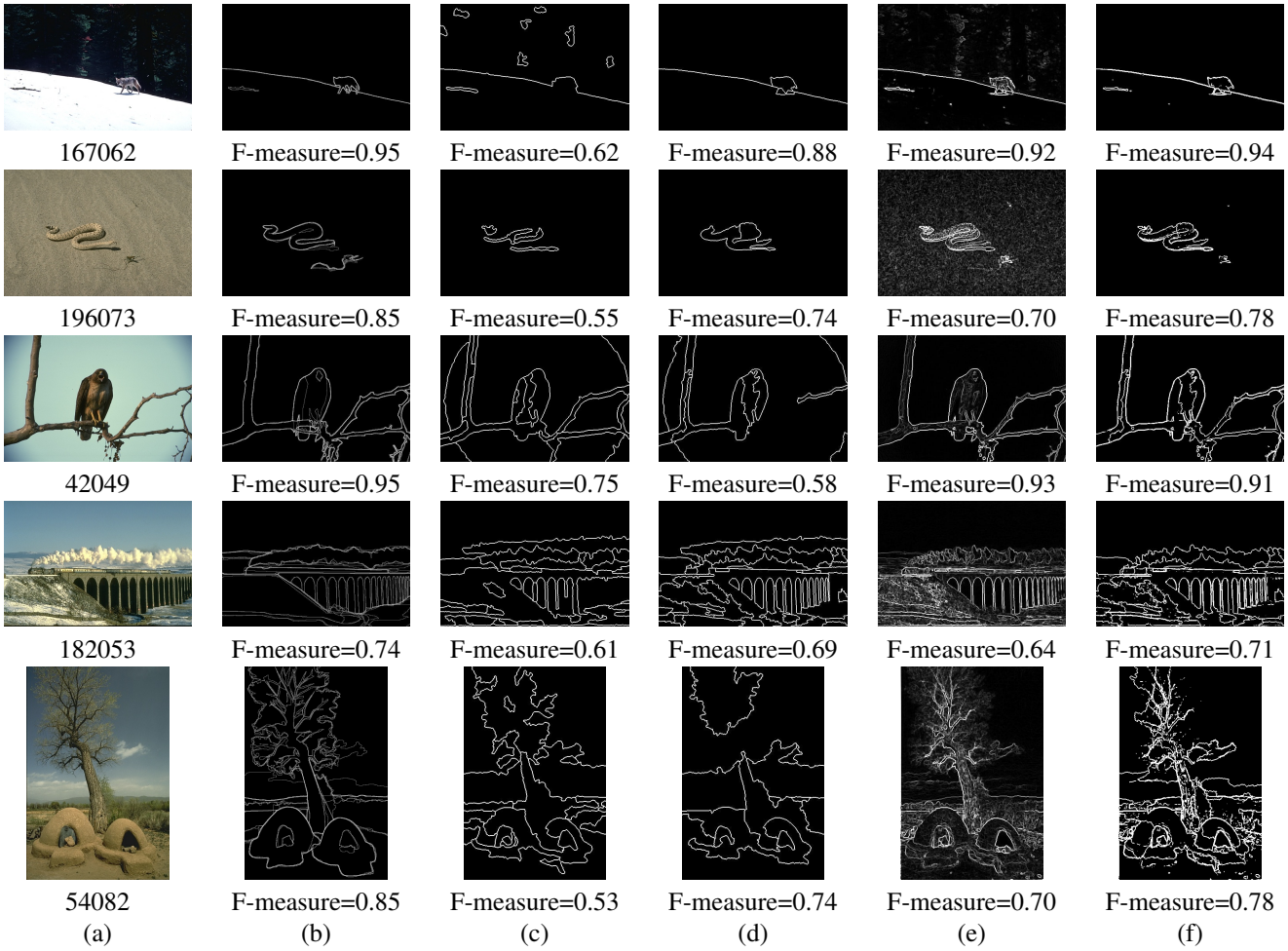

Fig. 8. (a) Original image and results of (b) Human benchmark (c) JSEG (d) MM-Frac (e) edge detection method and (f) KoSS

[2] X. Muñoz, J. Freixenet, X. Cufí, and J. Martí, "Strategies for image segmentation combining region and boundary information," *Pattern Recognition Letters*, vol. 24, no. 1-3, pp. 375–392, 2003.

[3] K. S. Komati, E. O. T. Salles, and M. Sarcinelli-Filho, "Fractal-JSEG: JSEG using an homogeneity measurement based on local fractal descriptor," in *Proceedings of the Brazilian Symposium on Computer Graphics and Image Processing, 22nd SIBGRAPI*, L. G. Nonato and J. Scharcanski, Eds. Los Alamitos: IEEE Computer Society, 2009, pp. 253–260. [Online]. Available: http://urlib.net/sid.inpe.br/sibgrapi@80/2009/08.17.19.03

[4] ——, "Unsupervised color image segmentation based on local fractal descriptor an J-images," in *Proceedings of the IEEE ICIT 2010 International Conference on Industrial Technology*, vol. 1, 2010, pp. 303–308.

[5] ——, "Unsupervised color image segmentation based on local fractal descriptor," in *Proceedings of the 17th International Conference on Systems, Signals and Image Processing (IWSSIP 2010)*, vol. 1. Los Alamitos: IEEE Computer Society, 2010, pp. 243–246.

[6] ——, "Two-level strategy for image boundary detection. in: International conference on computer vision theory and applications," in *Proceedings of International Conference on Computer Vision Theory and Applications (VISAPP 2011)*. INSTICC Press, March 2011, pp. 181–186.

[7] K. S. Komati, J. L. A. Samatelo, E. O. T. Salles, and M. Sarcinelli Filho, "A strategy for boundary detection combining region and edge information," in *Proceedings of the 24th Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI 2011)*, T. Lewiner and R. Torres, Eds., Conference on Graphics, Patterns and Images, 24. (SIBGRAPI). Los Alamitos: IEEE Computer Society Conference Publishing Services, 2011. [Online]. Available: http://urlib.net/sid.inpe.br/sibgrapi/2011/07.08.22.19

[8] K. S. Komati, E. O. T. Salles, and M. Sarcinelli-Filho, "KSS: Using region and edge maps to detect image boundaries," *Computing in Science and Engineering*, vol. 13, pp. 46–52, 2011.

[9] B. B. Chaudhuri and N. Sarkar, "Texture segmentation using fractal dimension," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 72–77, 1995.

[10] A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 661–674, nov 1984.

[11] A. Torralba and A. Oliva, "Statistics of natural image categories." *Network: Computation in Neural Systems*, vol. 14, no. 3, pp. 391–412, 2003.

[12] K. F. Côco, E. O. T. Salles, and M. Sarcinelli-Filho, "Topographic independent component analysis based on fractal and morphology applied to texture segmentation," *Lecture Notes in Computer Science*, vol. 5441, pp. 491–498, 2009.

[13] O. Rotem, H. Greenspan, and J. Goldberger, "Combining region and edge cues for image segmentation in a probabilistic gaussian mixture framework," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, 2007.

[14] D. Attali, G. Sanniti di Baja, and E. Thiel, "Skeleton simplification through non significant branch removal," *Image Processing and Communications*, vol. 3, no. 3-4, pp. 63–72, 1997.

[15] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 800–810, August 2001.

[16] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.