



Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvcir

Object detection with feature stability over scale space

Cattleya Duanggate^{a,*}, Bunyarit Uyyanonvara^a, Stanislav S. Makhanov^a, Sarah Barman^b, Tom Williamson^c

^a School of Information, Computer and Communication Technology, Sirindhorn International Institute of Technology (SIIT), Thammasat University, 131 Moo 5, Tiwanont Road, Bangkadi, Muang, Pathumthani 12000, Thailand

^b Faculty of Computing, Information Systems and Mathematics, Kingston University, Penrhyn Road, Kingston upon Thames, Surrey KT1 2EE, UK

^c Department of Ophthalmology, St. Thomas' Hospital, London SE1 7EH, UK

ARTICLE INFO

Article history:

Received 7 June 2010

Accepted 21 February 2011

Available online 26 February 2011

Keywords:

Scale-space theory

Scale-space tree

Object segmentation

Object detection

Feature stability

Feature vector

Blob's lifetime

Blob attributes

ABSTRACT

This paper proposes a novel segmentation method based on the scale space techniques endowed with a feature stability approach. The novelty of the paper is the lifetime of the space-scale blobs measured not only by their presence and disappearance but by the stability of the features characterizing the objects of interest as well. Our numerical experiments show that the algorithm outperforms the conventional space scale algorithm applied to variable size and variable shape objects. The proposed algorithm can be used as a preprocessing step in object or pattern recognition applications to produce seeds for more accurate image segmentation methods such as the snakes or the level set techniques.

© 2011 Elsevier Inc. All rights reserved.

1. Introduction

One of the most challenging tasks in computer vision is the segmentation of objects in the image. Typically, the image segmentation is used to locate regions of interest within the image. The process usually starts from dividing the original image into several homogenous parts with respect to some properties such as intensity, color or texture. The obtained results will be used in subsequent tasks such as object recognition or classification. Obviously, the performance of object recognition and classification depends on the quality of the image segmentation process.

A number of segmentation methods have been proposed. According to features used, they can be divided into two groups, namely, region-based [1–3] and edge-based segmentations [4–7].

Many previous works related to region-based segmentation or blob detection were proposed. Kawaguchi et al. [8] present blob analysis used to detect eyes from human face images. The algorithm extracts blobs by searching for intensity peaks and valleys obtained from binarized image and computes a cost using a Hough transform for each pair of blobs. A pair of blobs which has the smallest cost will be selected as the irises of both eyes. However,

the performance of this method relies on the selection of a proper intensity threshold and type of template used to detect blobs. Damerval and Meignen [9] present a blob detection based on maxima lines of the continuous wavelet transform. The detection method is not based on a thresholding step but uses properties of maxima lines of regions of interest. A framework for detecting interesting blobs in the color domain is presented in [10]. The method consists of a weighted multi-scale blob detector, hue-based color histogram and Forstner operator for roundness calculation. In [11], an intelligent vehicle counting method based on blob analysis in traffic surveillance is presented. The algorithm consists of three steps, namely, moving object segmentation, blob analysis, and vehicle tracking. The velocity of each vehicle can be calculated by analyzing blobs of vehicles. Another blob analysis method used for detecting moving objects is presented in [12]. The method consists of three steps: the symmetric difference is used to extract a rough moving object, blob analysis is used to update the background model, and a proposed classification strategy is used to extract foreground.

Even though these algorithms are claimed to work efficiently and produce high accuracy segmentation results, most schemes require prior knowledge about the region of interest, e.g., color, size or shape. This information will then be used to specify parameters in order to improve the segmentation such as intensity thresholds, the size of template or window operator sizes, etc. Without the accuracy of the parameters, the proper segmentation could not be well executed. However, once the parameters are set to suit

* Corresponding author. Fax: +66 2501 3505.

E-mail addresses: cat.cattleya@gmail.com, D5022300403@studentmail.siiit.tu.ac.th (C. Duanggate), bunyarit@siit.tu.ac.th (B. Uyyanonvara), makhanov@siit.tu.ac.th (S.S. Makhanov), s.barman@kingston.ac.uk (S. Barman), tom@retinasurgery.co.uk (T. Williamson).

one set of test images, they might have to be readjusted again when a new set arrives. Even in the same image, the problem may be more prominent if the target objects appear in different sizes or uncertain shapes. Besides, analysis of only similarity of the region cannot yield the best result especially for segmenting objects from an image with a complex scene, such as traffic images, or objects which consist of several parts, such as buildings or flowers.

Since details of objects in an image only exist and make sense in some observed scales, so the concept of hierarchical structures of an image is required to describe the structure of an image in different aspects. The idea of creating multi-scale representation of signals is first proposed by Witkin [13], called scale-space theory, to analyze a 1-D signal. Later, Lindeberg [14,15] employs the scale space approach to detect local maxima with extent in a 2-D intensity image, called grey-level blobs, at multiple scales. The relationship of all detected blobs at all scales can be considered by constructing a scale-space blob tree. In the absence of further information, the significance of a blob can be measured using its attributes, e.g., grey-level intensity, color, etc. The important structures in the image can be obtained by selecting the blobs with more significance.

The concept of scale space has been widely used in several applications for detecting image features such as blobs, edges, ridges and corners. Carvalho et al. [16] proposed the method to segment yeast cells based on watershed and scale space analysis. Trees and calculated node attributes are built such as survival time, shape and gray-scale, in order to perform segmentation analysis. The automatic scale selection for edge and ridge detection is presented in [17]. The number of strongest edge responses is selected by analyzing the integrated normalized gradient magnitude along the scale-space edges. For ridge detection, the ridge strength is considered by maximizing a normalized measure of ridge strength over scales. The proposed method for detecting corners in [18] employs the scale-space method and Plessey operator to detect corners belonging to different scales instead of a certain scale. The final result is obtained by combining the corners detected at every scale and a tracking back algorithm is applied to get the accurate localization. A multi-scale method for shape recognition is presented by Jalba et al. [19]. The method is based on two morphological scale-space representations and the hat-transform in scale space. They use maximum heights of the extrema of the curvature function as a shape descriptor.

In normal blob linking processes (to form a scale space tree), only spatial information of the blobs in consecutive scales is used. The main flaw within this process is that the blurring does not take into account with any other information on the image. For example, two different blobs with different colors will still be merged if they stay close together spatially.

In this paper, we propose an object detection method based on scale space by incorporating features of blobs into the scale space blob linking process as well as spatial information. We demonstrate in the paper that blobs are linked if their features are stable over scales, we get a better performance in the object detection. The evolution of linked blobs over scales using feature stability presents how stable image structures are in scale-space.

2. Proposed method

2.1. Scale space representation

In scale-space theory, a multi-scale representation of a two-dimension image, $f(x, y)$, is defined by a convolution with the Gaussian kernel $f(x, y, \sigma)$. The successive smoothing process generates a set of output images in various scales, σ . A scale parameter,

σ , of the kernel is gradually increased many times to create a series of smoothed images. During the blurring process, less important details of the image are suppressed while prominent structures and features still remain. In other words, the purpose of constructing the scale space is to analyze the behavior of the characteristics of the image structures under blurring. The increment of the scale parameters results in suppressing insignificant structures and creating blobs. Throughout the process, the Gaussian blurring simplifies the image without producing new spurious structures. Smaller light blobs that are close together merge into larger ones in the next scale until the whole image eventually contains only one blob.

The process ends when only a single “superblob” remains in the blurred image. The multi-object detection is ensured by considering the lifetime of the blobs (Fig. 1). A new blob appears either at the lowest scale (no blurring) or as a result of merging two blobs. Note that this model considers merging only two blobs at once. If it is not the case, the scale step gets decreased so that the resulting scale-space tree is always binary.

The life-time is measured from the moment the new blob is generated until it disappears by annihilation or by merging with another blob. Alternatively, the lifetime can be measured until annihilation or merging with the final “superblob”.

In order to exclude small insignificant blobs with a long lifetime (isolated noise) the lifetime is measured from $t = t_{START}$ rather than $t = 0$ (details in Section 2.3).

Furthermore, a new blob corresponds to a new object. The boundary of the object is specified in the original image when the corresponding new blob appears in the tree for the first time.

Finally, the novelty of our paper is that the lifetime is measured not only by the presence and disappearance of the blobs in the blurred images, but by the stability of the features characterizing the objects of interest as well.

In Fig. 1, we show a conventional blob linking result where only spatial information is used to link nodes from the consecutive scales. There are eight scales, in this example, denoted from the finest to the coarsest scales by $t_1 - t_8$. Note that the interval between scales can be unequal depending on σ but we simplified the tree for demonstration purposes. Each node represents a blob found at a specific scale. An edge connecting the two blobs indicates a direct spatial relationship between them. For example, b_2 at scale t_1 becomes b_{11} at scale t_2 . b_{20} and b_{21} merge into blob b_{25} . The scale-space blob tree in Fig. 1 is called a cross-sectional plane of the tree. The same result can be redrawn as a top-view plane as shown in Fig. 2. From this figure, it can be seen clearer that there might be some regions where small blobs extracted from finer scales may stay longer and might be found overlapping with some larger blobs in the coarser scales.

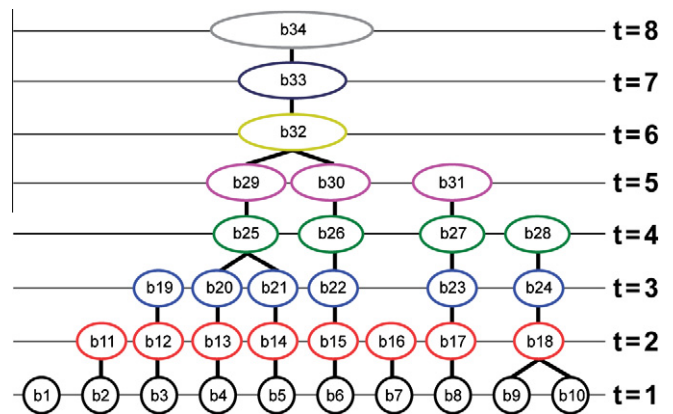


Fig. 1. Cross sectional scale-space blob tree.

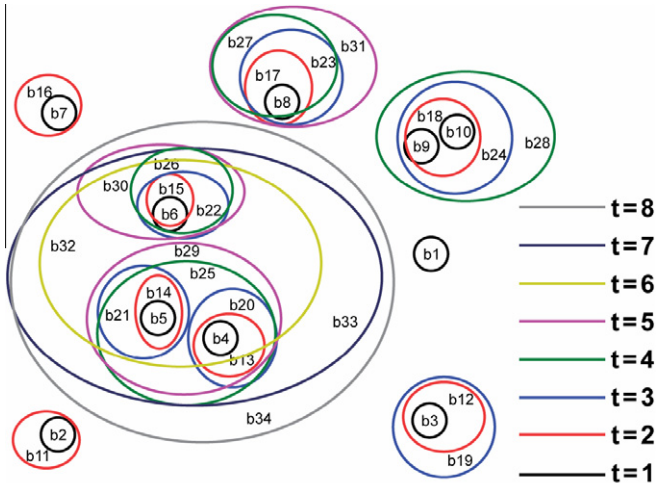


Fig. 2. A top-view plane of scale-space tree in Fig. 1.

Each blob detected from each scale represents an image component whether the component is meaningful or meaningless. The importance of each blob has to be, then, specified.

If a blob stays for a long period in the scale space blob tree, it can be assumed to be important. Traversing the scale-space blob tree, yields the candidate objects in the original image.

2.2. Feature stability

In this paper, we incorporate the features of a blob such as a color or texture into the conventional blob linking process [14,15] by defining the “strong” and “weak” links (see the 4 cases below). The weak link corresponds to an unstable feature and interrupts the life of the particular blob. Note that the links can be weighted depending on the degree of stability of the features defined below. Besides, some short weak links may be acceptable. However, such modifications are problem dependent. In this paper we do not consider them.

The additional blob descriptors form a blob’s feature vector extracted from the original image using the boundary of the blob as a mask. Note that we always extract the feature vector from the original image so blurring does not affect the feature vector. Features can be represented by general descriptors of the color, texture, etc. or be application-specific such as the compactness (see the forthcoming sunflower example).

We characterize the feature stability by

$$S = \frac{1}{\text{dist}(f, g)}, \quad (1)$$

where $f = [f_1, f_2, f_3, \dots, f_n]$ and $g = [g_1, g_2, g_3, \dots, g_n]$ are feature vectors corresponding to two candidate blobs and dist is an appropriate distance (Euclidean distance, Mahalanobis distance, etc). Note that a distance with weights w_i can be used, such as, $\text{dist}(f, g) = \sum w_i \text{dist}(f_i, g_i)$ if the features have different significance. Additionally the significance of the features may vary depending on the time step.

An example in Fig. 3 demonstrates the feature stability approach. The feature vector consists of one nominal feature (color) characterized by R, G, B and Y (as denoted on the top-left corner of the blob) and lifetime, denoted on the top-right corner. Note that lifetime displayed in Figs. 3–5 use $t_{\text{START}} = 0$. We consider 4 possible cases of the relationship within the scale-space tree, namely:

Case 1: Two blobs merge. The feature vector is stable. The two blobs from a lower scale merge into a bigger blob in the next

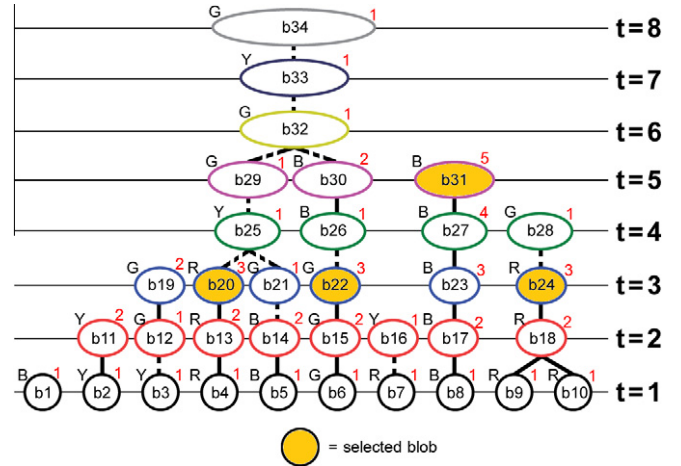


Fig. 3. Example of blob selection by the proposed feature stability method.

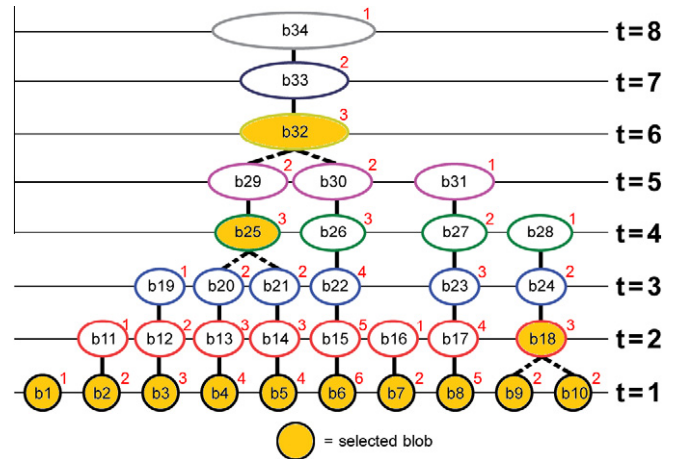


Fig. 4. Blob candidate selection using method CM1.

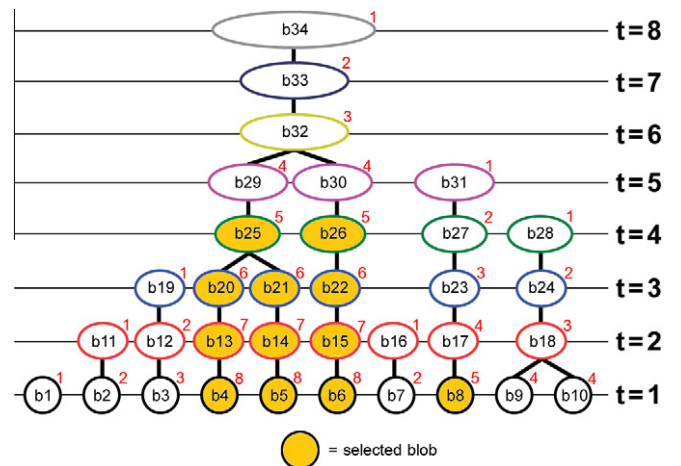


Fig. 5. Blob candidate selection using method CM2.

scale. The feature vector of the new blob is close to that of one or both of the source blobs, i.e. the two source blobs are parts of the same object separated by noise. We will call such transition the *strong link* (solid lines in Fig. 3).

Case 2: Two blobs merge. The feature vector is unstable. If the blobs merge spatially but the features are not stable after the merging, the source blobs disappear at this level and a new blob is created. For example, in Fig. 3 blob b_{20} and blob b_{21} disappear and a new blob b_{25} is created. This happens when some unimportant or small objects (details) sitting close to each other

merge. We call this relationship the *weak link*. It is represented by a dashed line in Fig. 3.

Case 3: A blob is expanding. The feature vector is stable. If a blob at a lower scale shares the spatial location with a blob at the next scale and the feature vector is stable, most likely it is the same object expanding under blurring (see blobs b_4 , b_{13} and

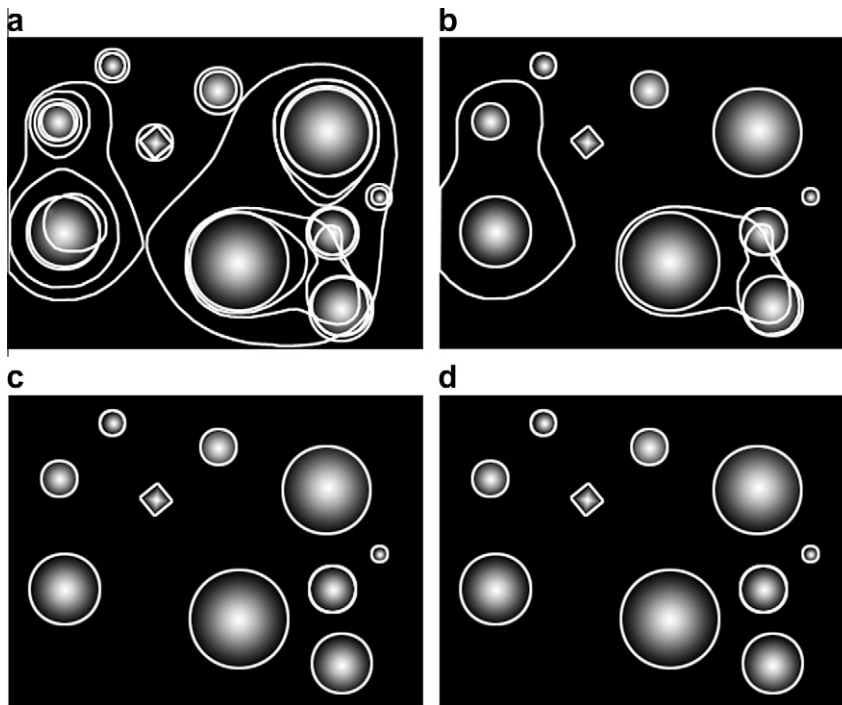


Fig. 6. A synthetic image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

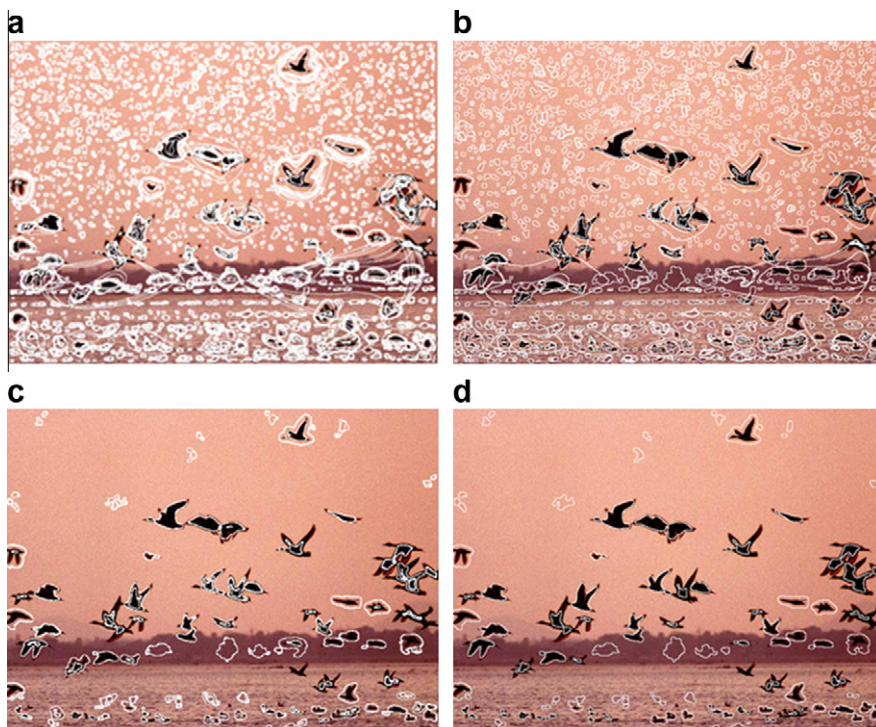


Fig. 7. A bird image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

b_{20} in Fig. 3). The link between the two blobs in this case is also called a *strong link*.

Case 4: A blob is expanding. The feature vector is unstable. If a blob at a lower scale shares the spatial location with a blob at the next scale and the feature vector is unstable we assume that it merges with the local background or with a bigger blob characterized by different features. In this case the original blob disappears and a new blob is created. b_{22} and b_{26} show such a relationship where the green object becomes a red one when it gets bigger. However, there is still a *weak link* between them since they share the same spatial location.

2.3. Blob significance and blob candidate selection

Once the Lindeberg’s tree is generated, each blob is evaluated for its significance. With no other information, we assume that the important structures and the important features in the image stay longer over scales. The blob lifetime is defined by

$$L = t_D - t_A - t_{START} \tag{2}$$

where t_D and t_A denote the scales when the blob disappears and appears, respectively, and t_{START} is the time when the life count begins. t_{START} is introduced to eliminate the noise and unwanted tiny blobs. In our numerical experiment we considered $t_{START} = 2$. Besides, for simplicity, we assume that the weak links (see the previous section) interrupt the life of the blob and a new blob appears at the end of the weak link. For instance, if we have a transition GreenGreenYellow the third step indicates the appearance of a new blob. Note that this idea can be developed further and a short weak link such as GreenGreenYellowGreen still may be acceptable. However, in this paper we do not consider this modification.

3. Numerical experiments

To demonstrate the robustness and generality of the algorithm we consider the stability of three features: the entropy, the average value of the gray level of the image and the standard deviation of

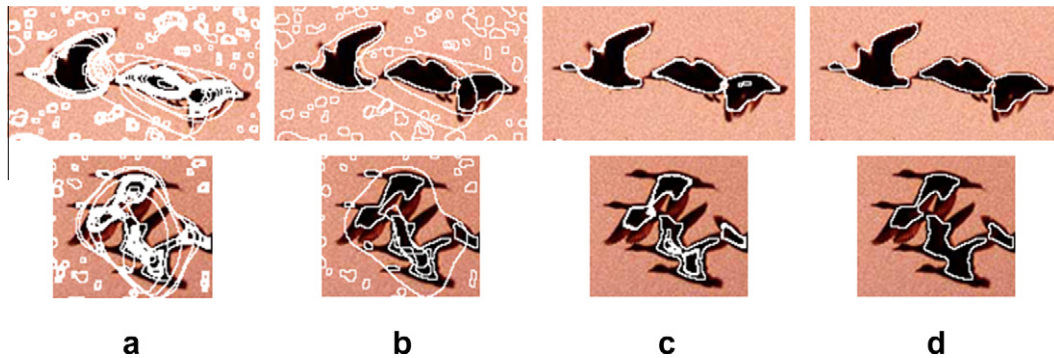


Fig. 8. Close-ups of 2 areas in the bird image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

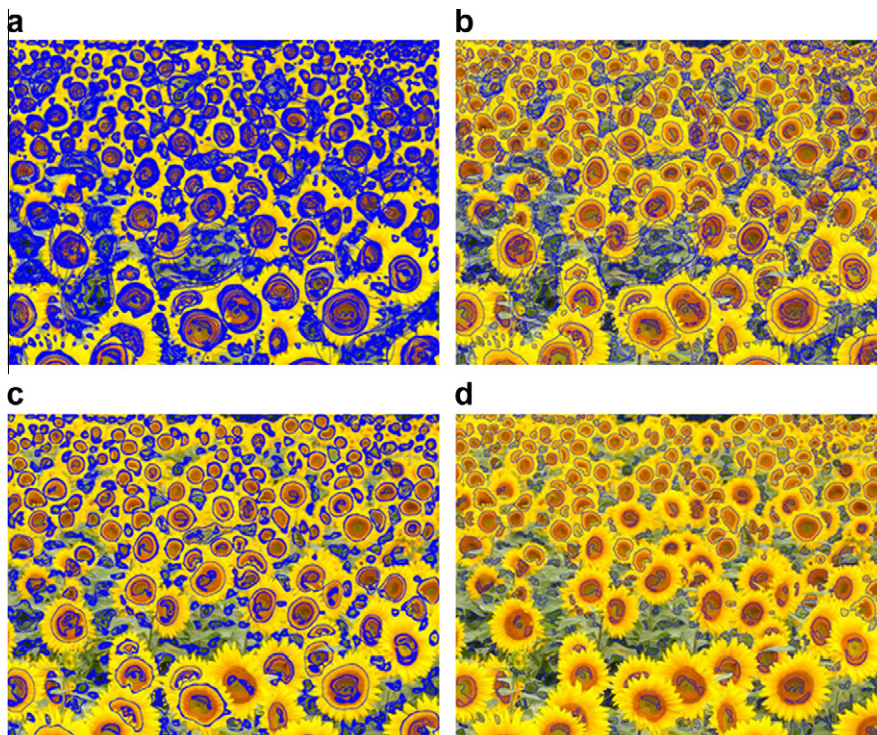


Fig. 9. A sunflower image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

the gray level. These three values represent the texture, the gray level and the gray level distribution, respectively. No systematic feature selection was performed. The feature selection was intuitive but supported by some standard pattern recognition schemes [20–27].

We also show how additional features can be introduced for particular applications. For example, in the forthcoming sunflower segmentation (Fig. 9) the compactness [28] is included in the feature vector because round shape objects are of interest. The features are normalized so that they range from 0 to 1 (a weighting

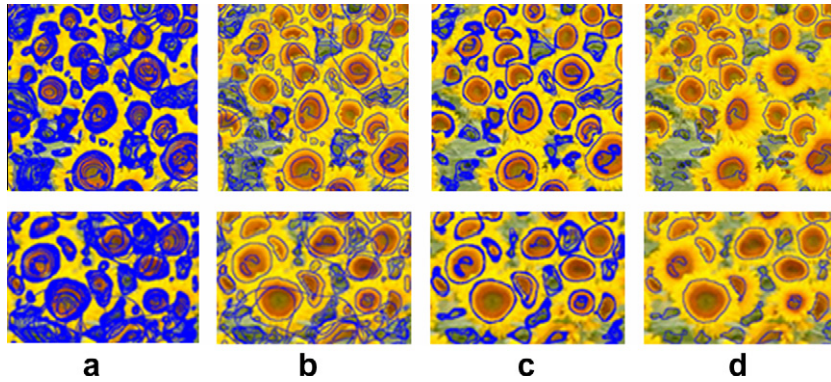


Fig. 10. Close-up of the sunflower image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

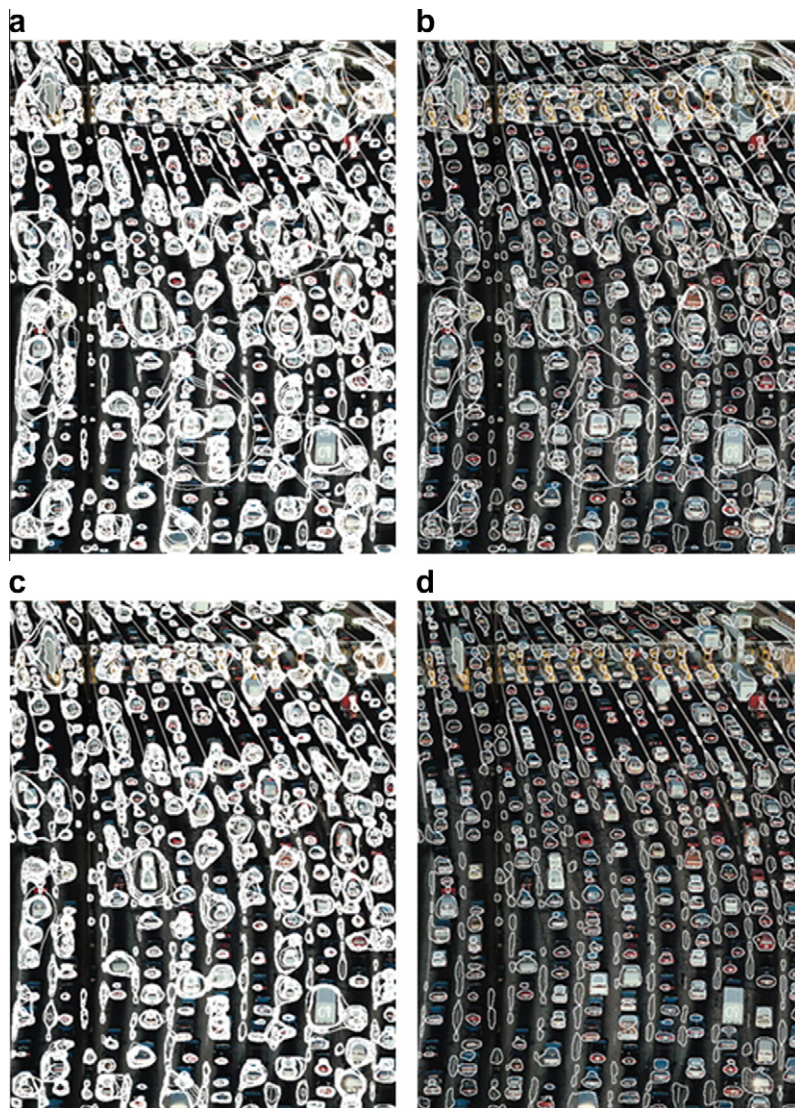


Fig. 11. A toll way image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

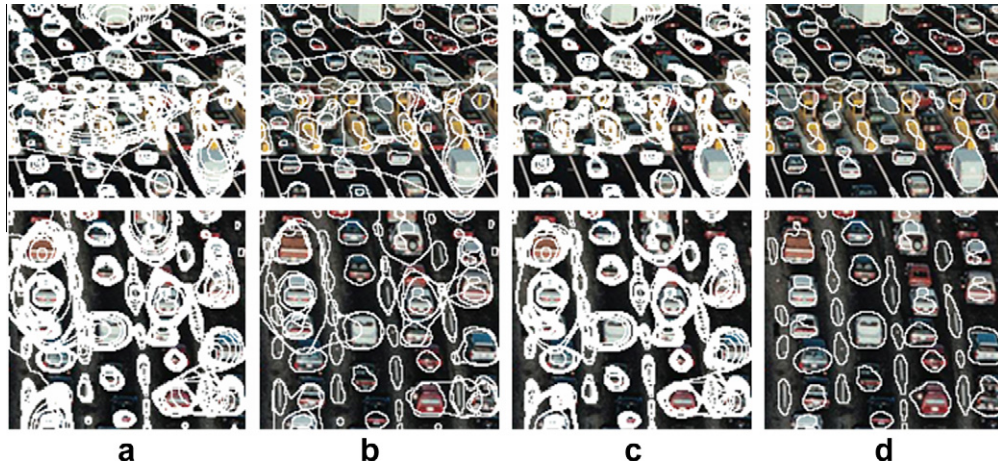


Fig. 12. Close-ups from 2 areas in the toll way image. (a) All detected blobs overlaid on original image. (b) Result from CM1. (c) Result from CM2. (d) Result from the proposed method.

Table 1
Detection rate.

Image	Expected blobs (manual count)	CM1		CM2		Proposed method	
		Blob count	SNR	Blob count	SNR	Blob count	SNR
Synthetic image	10	13	3.33	10	∞	10	∞
Bird	82	2231	0.04	272	0.43	128	1.78
Sunflower	172	7804	0.02	2329	0.08	397	0.76
Toll way	268	1128	0.31	1863	0.17	360	2.91

Table 2
Running time.

Image	Running Time (s)			
	Preprocessing (tree building)	Traversing CM1	Traversing CM2	Traversing proposed method
Synthetic image	32.29	0.27	0.15	0.43
Bird	688.89	1.10	1.73	4.57
Sunflower	4700.00	2.36	6.41	15.63
Toll way	1343.1	2.16	3.50	9.00

coefficient may also be introduced to emphasize a relative significance of the features).

The scale-space tree is constructed and traversed to calculate the lifetime of each blob. The blobs with the maximum lifetime are selected as the object candidates.

As baselines, we compare our object detection results with two conventional blob linking methods. These two methods do not take feature stability into account when the scale space tree is formed so the blobs merge if they are close together. The first conventional method, (CM1) assigns a *weak link* whenever the blobs merge. In other words when the blobs merge they disappear and a new blob is created. *Two strong links* are assigned by the second conventional method (CM2) when the two blobs merge.

If CM1 is used, the lifetime of each blob in Fig. 1 is measured as shown in Fig. 4. For example, the lifetimes of blobs b_1, b_2, b_4, b_6 are 1, 2, 4 and 6, respectively. The selected blobs are indicated by shade.

An example of using CM2 is illustrated in Fig. 5. The lifetimes of blob b_2 and b_7 are both equal to 2. Blob $b_4, b_5,$ and b_6 have a lifetime 8. Blob b_{29} , and b_{30} have lifetimes of 4. Note that in this case the blobs can spatially overlap. For this case, the threshold of 60% of the maximum lifetime is considered for the candidate selection.

A synthesis image and three natural scene images are used to evaluate the algorithm. The results are shown in Figs. 6, 7, 9 and 11. In each figure: (a) shows all the detected blobs, (b), (c) and (d) the results of CM1, CM2 and the proposed algorithm, respectively. In addition, close-ups of detected blobs in some areas of these examples are also shown in Figs. 8, 10 and 12.

In Fig. 6, CM2 produces a similar result to the proposed method where all the objects in the image are detected correctly. For Fig. 7, too many false blobs (appearing from background noise) are detected in Fig. 7(b) with CM1. Blobs which are close together, like the four birds on the right of the picture, are grouped together into a bigger blob with this method CM1 while CM2 and the proposed method successfully separate the birds. However, the proposed

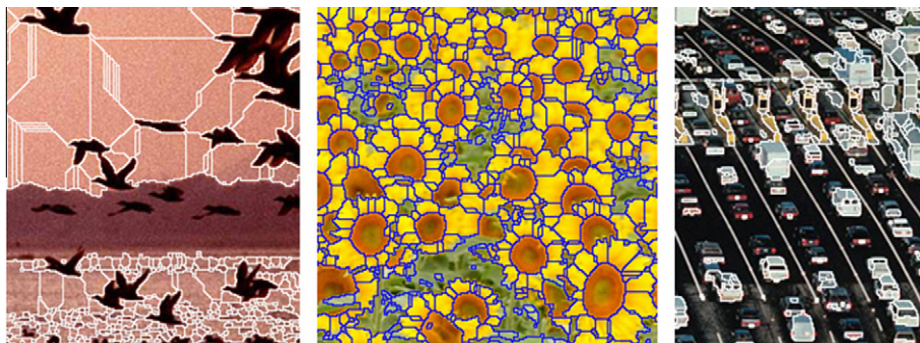


Fig. 13. Watershed segmentation results.

feature stability method performs better reducing the number of the overlapping blobs.

Consider the image in Fig. 9. Let us add compactness to the feature vector since in this particular test the object of interest has a round shape. Even though, nearly all the objects are detected with CM1, many false blobs are detected as well, especially those blobs that merged. As far as CM2 is concerned, many overlapping blobs were produced. Fig. 9(d) shows that our method provides the best result. Most of the overlapping blobs have been eliminated and the majority of the sunflowers have been detected.

The result in Fig. 11 is similar to that in Fig. 9, there are too many blobs detected with CM1 and highly overlapping blobs are produced by CM2. The proposed method is superior in detecting and separating the objects even though the objects are relatively small, as in this case.

Table 1 shows the detection rates versus the signal-to-noise ratio for all our test images. The execution times used by the algorithm are displayed in Table 2. The most time consuming procedure is constructing the space-scale tree.

The test images are also compared qualitatively with the traditional watershed algorithm. The results from the watershed algorithm are displayed in Fig. 13. Even though the algorithm segments images into regions, it does not really recognize the scale of the objects so that the segmentation results contain big blobs. Besides, the watershed method cannot separate objects with a similar color from the background.

4. Conclusion

We have proposed a novel object detection method based on scale-space endowed with feature stability. The algorithm is robust for detecting variable size and variable shape objects without a priori information of the object of interest. The shapes and sizes of multiple objects such as the flying birds or sunflowers are not uniform throughout the image but the algorithm can detect most of them.

The algorithm is flexible. The feature vectors can be extended or modified to suit particular applications. The definition of the blob lifetime can also be adjusted. If the object of interest is not the main object in an image, blobs with a certain range of lifetimes can be chosen rather than the blob with the longest lifetime. The short weak links and the rate of the increase or decrease of the feature descriptor could also be incorporated.

The proposed algorithm provides an ideal outcome towards applications such as pattern recognition tasks because of its robustness and independence of object shape, color or any predefined structures of the object of interest. Furthermore, the outputs of the algorithm can also be used as seeds for other shape detection technologies such as active contour (snakes) or the level set segmentation.

Acknowledgments

This project is funded by Guy's and St. Thomas' Hospital Special Trustees, UK and Thailand Toray Science Foundation (TTSF) and National Research University Project of Thailand Office of Higher Education Commission.

References

- [1] S. Wang, X. Ma, X. Zhang, L. Jiao, Watershed-based textural image segmentation, *Intelligent Signal Processing and Communication Systems* (2007) 312–315.
- [2] D. Gatica-Perez, C. Gu, M. Sun, S. Ruiz-Correa, Extensive partition operators, gray-level connected operators, and region merging/classification segmentation algorithms: theoretical links, *IEEE Transactions on Image Processing* 10 (2001) 1332–1345.
- [3] S. Manay, A. Yezzi, Anti-geometric diffusion for adaptive thresholding and fast segmentation, *IEEE Transactions on Image Processing* 12 (2003) 1310–1323.
- [4] L.A. Forbes, B.A. Draper, Inconsistencies in edge detector evaluation, *Computer Vision and Pattern Recognition* (2000) 398–404.
- [5] S. Konishi, A. Yuille, J. Coughlan, Statistical edge detection: learning and evaluating edge cues, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (2003) 57–74.
- [6] K. Bowyer, C. Kranenburg, S. Dougherty, Edge detectorevaluation using empirical roc curves, *Computer Vision and Image Understanding* 84 (2001) 77–103.
- [7] H. Gomez-Moreno, S. Maldonado-Bascon, F. Lopez-Ferreras, Edge detection in noisy images using the support vector machines, in: *Sixth International Work-Conference on Artificial and Natural Neural Networks*, 2001, pp. 685–692.
- [8] T. Kawaguchi, D. Hidaka, M. Rizon, Robust extraction of eyes from face, in: *Proceedings on the 15th International Conference on Pattern Recognition*, vol. 1, 2000, pp. 1109–1114.
- [9] C. Damerval, S. Meignen, Blob detection with wavelet maxima lines, *IEEE Signal Processing Letters* 14 (2007) 39–42.
- [10] A. Ming H. Ma, A blob detector in color images, in: *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, 2007, pp. 364–370.
- [11] C. Thou-Ho, L. Yu-Feng, C. Tsong-Yi, Intelligent vehicle counting method based on blob analysis in traffic surveillance, in: *Second International Conference on Innovative Computing, Information and Control*, 2007, pp. 238–238.
- [12] R. Megret, J.M. Jolion, Tracking scale-space blobs for video description, *IEEE Multimedia* 9 (2002) 34–43.
- [13] A.P. Witkin, Scale-space filtering, in: *Proc. 8th Int. Joint Conf. Art. Intell.*, 1983, pp. 1019–1022.
- [14] T. Lindeberg, Feature detection with automatic scale selection, *International Journal of Computer Vision* 30 (1998) 79–116.
- [15] T. Lindeberg, *Scale-space theory in computer vision*, Kluwer International Series in Engineering and Computer Science, Dordrecht, Netherlands, 1994.
- [16] M.A.G. Carvalho, R.A. Lotufo, M. Couprie, Segmentation of images of yeast cells by scale-space analysis, in: *XVI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'03)*, 2003, p. 376.
- [17] T. Lindeberg, Edge detection and ridge detection with automatic scale selection, *International Journal of Computer Vision* 30 (1998) 117–156.
- [18] G. Xinting, Z. Wenbo, F. Sattar, R. Venkateswarlu, E. Sung, Scale-space based corner detection of gray level images using plessey operator, in: *Fifth International Conference on Information, Communications and Signal Processing*, 2005, pp. 683–687.
- [19] A.C. Jalba, M.H.F. Wilkinson, J.B.T.M. Roerdink, Shape representation and recognition through morphological curvature scale spaces, *IEEE Transactions on Image Processing* 15 (2006) 331–341.
- [20] M.S. Nixon, A.S. Aguado, *Feature Extraction and Image Processing*, second ed., Academic Press, Oxford, 2008. pp. 312–313.
- [21] R.C. González, R.E. Woods, *Digital Image Processing*, third ed., Pearson Prentice Hall, New Jersey, 2008. pp. 533–534.
- [22] Bernd Jähne, *Digital Image Processing*, sixth ed., Springer, Netherlands, 2005. 537.
- [23] T. Acharya, A.K. Ray, *Image Processing: Principles and Applications*, John Wiley & Sons, New Jersey, 2005.
- [24] A. Sopharak, B. Uyyanonvara, S. Barman, T.H. Williamson, Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods, *Computerized Medical Imaging and Graphics* (2008) 720–727.
- [25] S. Boltz, F. Nielsen, S. Soatto, Texture regimes for entropy-based multiscale image analysis, in: *Proceedings of the European Conference on Computer Vision*, 2010.
- [26] Q. Wang, C. Zhao, J. Yang, Robust facial feature location on gray intensity face, *Lecture Notes in Computer Science* 5414 (2009) 542–549.
- [27] K. Yanai, K. Barnard, Image region entropy: a measure of “visualness” of web images associated with one concept, in: *Proceedings of the 13th Annual ACM International Conference on Multimedia*, 2005, pp. 419–422.
- [28] M.S. Nixon, A.S. Aguado, *Feature Extraction and Image Processing*, second ed., Academic Press, Oxford, 2008. pp. 312–313.