Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 23 (2005) 555–563
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Multi-scale feature identification using evolution strategies☆

Xiaojing Yuan[a], Jian Zhang[b,*], Xiaohui Yuan[b], Bill P. Buckles[b]

[a]Mechanical Engineering Department, Tulane University, New Orleans, LA 70118, USA
[b]Electrical Engineering and Computer Science Department, Tulane University, New Orleans, LA 70118, USA

Received 13 March 2002; received in revised form 25 February 2004; accepted 14 July 2004

Abstract

An image usually contains a number of different features and regions. Many image-related applications, such as content-based image retrieval and MRI-based diagnosis, often require the ability to identify and mark features within the image. For images containing a specific sort of feature (e.g. convective storm) or region (e.g. earthquake debris), that feature or region is always located adjacent to other features and regions on the image.

A generic framework for automatically identifying features in images based on evolutionary computation is proposed here. The significant characteristic of the method is that it does not require segmentation. We use evolution strategies as the optimization algorithm to identify features. The system is based on a conjecture that certain filters will give prominent responses to certain features. The identified features are represented as regions enclosed within the chosen search structure—the ellipse. By defining filter response criteria as the fitness function, evolution strategies succeeds in finding the feature in a much more efficient way than, say, segmentation.

Keywords: Feature identification; Evolution strategies; Gabor filter bank

1. Introduction

1.1. Problem statement

In image processing, a typical problem is feature identification, which can be formulated as: given an image I, identify homogeneous areas with feature F. This problem may be solved by various methods. Classical techniques for finding regions directly include such as region growing by pixel aggregation, region splitting and merging [1]. More recent methods include clustering [15] and texture-based techniques [12]. These methods are effective, but require extensive computation. The evolution strategies (ES) search method described here can examine images 'on the fly' and reduce only the regions of interest, omitting others.

Possible applications of feature identification are:

- Remote sensing is the measurement of properties of an object far away from the object. Applications include cloud classification and terrain classification.
- Content-based image retrieval uses image content as a search key. Some content types, e.g. water, sky, and forest, have texture as an important characteristic. UC Berkeley's Digital Library Project is one of those typical applications which use texture for retrieval purpose.
- Cancer is often characterized by textures with medical imaging techniques. Early detection of cancer or lesions from mammographic images is a significant specific application.

The notion of 'interesting' feature differs from application to application. A user has a specific domain-space definition such as convective storm, forest fire, snow pack, boreal forest, hydrocarbon spill, cancer lesion, and so forth. This approach for identifying such features permits the users to define the metrics to employ in a search. For example,

☆This work was supported in part by DoD EPSCoR and the State of Louisiana under grant F49620-98-1-0351.
* Corresponding author.
E-mail address: zhangj@eecs.tulane.edu (J. Zhang).

0262-8856/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2004.07.002

UC Berkeley's 'blobworld' describes a feature using color, texture and position. Medical diagnoses often treat size, location and texture as a lesion's identification metrics. Features in an image can also be described by statistical information such as pixel mean, standard deviation, and, covariance, or by transform responses such as Fourier or wavelet coefficients. For textured images, wavelet transform coefficients have been shown to be plausible metrics to describe features in images [22]. Here we employ filters at the signal level as a general way to extract a low level representation of the specified features.

Many feature identification systems in literal are application oriented and usually require complex computation. Obviously, if a user-defined feature F can be distinguished from other features by having prominent response to a filter, we can apply the filter to the image and the feature will be identified as a region with higher energy than adjacent regions.

The key contribution of the work described here is that a feature identification system is proposed, which uses a generic mechanism to describe features as higher energy areas. An example of an outcome of a feature search is shown in Fig. 1. (A convective storm is identified by an encircling ellipse.) This is an actual result from our prototype. Should an Earth-scientist present a sample convective storm to the prototype, metrics (based on texture) are devised which enable the discovery of similar features on comparable images.

1.2. Previous work

Algorithms for finding interesting regions in images have been proposed by Howe et al [9,11,19] and the aforementioned 'blobworld'. Finding shapes, edges, and segments in images using evolutionary computation is not common but has been accomplished. Examples include the detection of shapes and features using graph or semantic net matching criteria given images that have been segmented by region [20] or by edge [2,10]. Another approach, given that edge detection has taken place, is feature recognition by matching polygonal templates (angles and relative lengths of sides) with a predetermined set of objects [14].

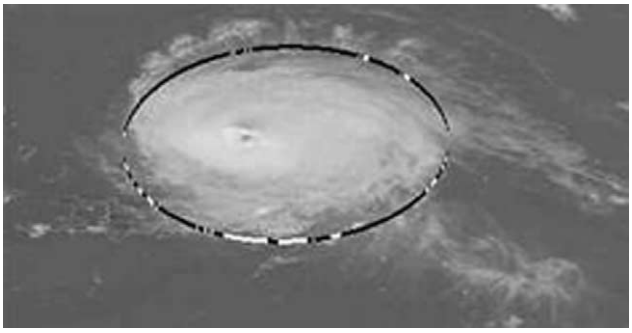

Fig. 1. An outcome example of feature identification.

Previously, we have conducted investigations leading to feature identification using ES as well as applied the method for image decomposition [4]. That work deals only with finding a single interesting region within an image and uses a simple criterion consisting of uniformity of reflective illuminance. The experimental results are adequate for simple features such as convective storms that stand in great contrast with the background. However, the experiments on some images containing natural (i.e. non-cultural) scenes resulted in less satisfactory outcomes.

The approach described here not only addresses the shortcomings of the previous work but also generalizes the method by parameterizing the criterion for 'inter-interesting feature'. We proposed a generic framework for automatic multi-scale feature identification using evolutionary computation. The user may substitute suitable matched filters as deemed best for specific applications.

The outline of the paper is as following: Section 2.1 gives a brief introduction on ES. Section 2.2 describes the principles of feature identification system. Section 2.3 gives the search problem representation using ES, which is followed by a description of objective function in Section 2.4. The remaining parts of Section 2 describes the image processing steps required for our system, which include Gabor filter banks, filter selection scheme, and feature image construction. We present images tested and experiment setup in Sections 3.1 and 3.2. In Section 3.3, we report the test results for finding interesting features in images. Finally, Our conclusions and a discussion is in Section 4.

2. Automatic feature identification system

2.1. Evolution strategies

The reason that ES are chosen as the optimization algorithm in feature identification system is that they are well-suited for real function evaluation. That is, an ES searches for $\vec{x}^*$ such that $f(\vec{x}^*) = \mathrm{opt}_{\vec{x}} f(\vec{x})$.

The algorithm of the ES is first formulated in the language of biology as following:

Step 0: A given population consists of μ individuals. Each is characterized by its genotype consisting of n genes, which unambiguously determine the vitality, or fitness for survival.

Step 1: Each individual parent produces λ/μ offspring on average, so that a total of λ offspring individuals are available. The genotype of a descendant differs only slightly from that of its parents. The number of genes, however, remains n.

Step 2: Select the best of the offspring to form parents of the following generation.

The selection operators in ES are completely deterministic. That is, given a population over which selection is performed, the next generation is well-defined. This is a minor but distinguishing characteristic of ES vis a vis other evolutionary algorithms. The two used in ES are $(\mu+\lambda)$ selection and (μ, λ) selection. The former scheme selects the best μ individuals from the combination of the previous μ parents and their λ offspring to form the next generation, while the latter selects the best μ individuals from the population of λ offspring. The experiments implemented here use the two schemes with same settings of parents and offspring size.

Each ES individual, therefore, represents a point, $\vec{x}$, on the response surface f. Each x_i, $i=1,2,\ldots,n$, is termed an *object variable* and is represented as a real value in the individual.

ES's are essentially randomized hill climbers. Hill climbing necessitates the resolution of two issues at each iteration—(i) the direction to move and (ii) the distance (step size). These issues are resolved by *control variables* that are part of each individual. THUS, an ES individual is partitioned into object variables and control variables as illustrated below

$$\langle \underbrace{x_1, x_2, \ldots, x_l}_{\text{object variables}}; \overbrace{\sigma_1, \sigma_2, \ldots, \sigma_m; \theta_1, \theta_2, \ldots, \theta_p}^{\text{control variables}} \rangle$$

An object variable should be considered the mean of a normally distributed random variate. Under that interpretation, each σ_i is a standard deviation for an object variable. Thus $m \leq l$. (If $m < l$ then σ_m applies to all x_j, $m \leq j \leq l$.) Each θ_i is a surrogate for the covariance of two object variables. The $\vec{\theta}$ is organized as an upper triangular matrix as is a covariance matrix. (That is to say, $p=(2l-m)(m-1)/2$.) The correspondence between θ_{ij}, $i, j \leq m$ and the covariance, c_{ij} is $\tan(2\theta_{ij}) = 2c_{ij}/(\sigma_i^2 - \sigma_j^2)$.

An interpretation of an ES individual is an l-dimensional jointly distributed normal variate with mean $\vec{x}$ and standard deviation $\vec{\sigma}$. The object and control variables are adjusted simultaneously by mutation operation. It should be noted that the major quality of ES lies in its ability to incorporate the optimization on object variables with self-adaptation on control variables [3].

2.2. Principles of feature identification by evolutionary computation

Fig. 2 shows the generic framework of the feature identification system. The objective is to use the chosen optimization algorithm to find the largest uniform areas that conform to consistency verification rules. The uniformity can be the measure of brightness intensity, reflective illuminance, or texture.

The input for the optimization algorithm could be the feature images from selected filter responses, or the original image. When the interesting features are already prominent (e.g. interesting features have uniform light intensity different from the background) in the original image, then the image can be used directly by the optimization algorithm to find the largest uniform areas that conform to the consistency verification rules. Otherwise, the feature enhancing rules, which include feature extraction and feature image construction system, can be applied.

The most critical component of the feature extraction system is the filter bank. Filter banks are chosen to extract features in the image by differentiating them from the background or other features. Not all filter responses contain useful information. The filter response selection algorithm chooses the responses that contribute most to the total energy in image. In our experiments, more than 20 Gabor filters are tested in order to find the ones that generate the best responses. The criteria to select filter responses can be salient difference, local energy evaluation, etc. The salient difference criterion will choose the filter responses that give the largest difference between specified features. Local energy evaluation is based on reconstruction of the input image from the filtered images, i.e. the energy of the reconstructed image I' should be a good approximation of that of the original image I. To avoid extensive computation, 2-D FFT power spectrum analysis described in Section 2.6 is used.

Before applying the optimization algorithm, feature images I'' should be constructed from selected filter responses to capture texture features defined by a measure of energy in a small window around each pixel in each filter response. This process will generate one feature image for each selected filter response. The objective of local energy function is to measure the energy in the filter response in a local region. It consists of a nonlinearity and smoothing.

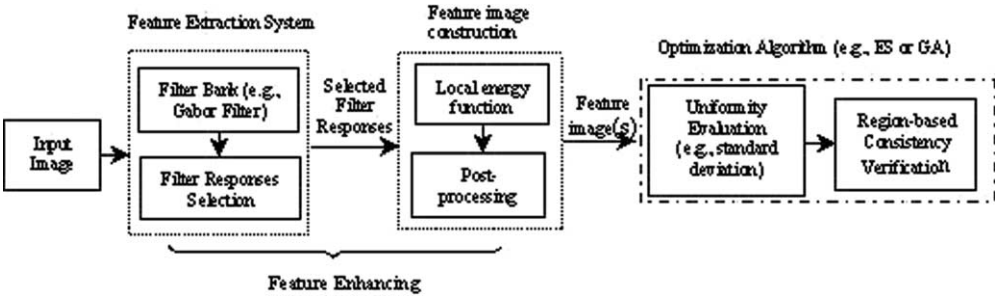

Fig. 2. Generic framework for feature identification system.

The purpose of post-processing, which includes dynamic range adjustment and histogram equalization, is to make the feature image a suitable representation for the optimization algorithm.

Evolutionary computation algorithms, such as ES and Genetic Algorithm, are direct, probabilistic search and optimization procedures rooted in organic evolution.[1] The efficiency (or speed of approach to the objective) and effectivity (or reliability under varying conditions) of ES to deal with real number numeric optimization problems have been proved in many experiments [18]. We formulated the feature identification problem as searching in the feature image I'' for feature F, which has largest area and lowest variation. Then the consistency verification is applied if user has a priori knowledge about the characteristic of the interesting features in the image. Only qualified results will survive the selection. When the searching process terminated either by fulfilling some convergence criteria or timeout, the best candidate among these qualified solution will be the output optimum.

2.3. Feature search problem representation in evolution strategies

At the heart of this approach (and a chief contribution of this paper) is transforming the feature identification problem to a numerical optimization problem and using ellipse as the search structure to represent the solution to the optimization problem. The object variables of an ES individual are used to represent an ellipse corresponding to a candidate region containing a feature. The uniformity (as measured by to-be-described metrics) of the region enclosed by the ellipse is the fitness value for each candidate.

The form of an ES individual for our feature identification system is $(x_1,x_2,\ldots,x_5;\ \sigma_1,\sigma_2,\ldots,\sigma_5;\ \theta_1,\theta_2,\ldots,\theta_{10})$, where the object variables are defined as a search structure:

(x_1,x_2) the center of an ellipse
(x_3,x_4) the axis radii of an ellipse
(x_5) the rotation angle of an ellipse

The control variables, $\vec{\sigma}$ and $\vec{\theta}$, have the standard interpretation of defining the hyper-ellipsoid that proscribes the mutation operator as described in Section 2.1.

2.4. Objective function

Objective function

$$g(\vec{x}) = \frac{\mathrm{std}(\vec{x}) + \delta}{\mathrm{num}(\vec{x}) + \delta}, \tag{1}$$

where

$$\mathrm{std}(\vec{x}) = \sqrt{\frac{\sum_{i\in\vec{x}}(\mathrm{pi} - \bar{p})^2}{\mathrm{num}(\vec{x})}},\quad \bar{p} = \frac{\sum_{i\in\vec{x}} p_i}{\mathrm{num}(\vec{x})},$$

$\mathrm{num}(\vec{x})$=number of pixels enclosed by ellipse $\vec{x}$. δ is a small positive number to avoid zero fitness or infinite fitness during optimization. This is necessary when the region of interest is extremely homogeneous.

The objective function (Eq. (1)) returns the fitness value of an ES individual. To recap, the object variables of an ES individual define the search structure—an ellipse in the feature space. Remember, the visualization of the 2-D ES individual is also an ellipse in the solution space. In our system, the homogeneity of feature is defined as the ratio of standard deviation of pixels enclosed in the ellipse 'area' of the ellipse. The area enclosed by the ellipse is calculated as the number of pixels falling inside the ellipse. The ellipse that encloses the larger region with the lower standard deviation is favored, not necessarily the ellipse with the lowest standard deviation value.

If input is the original image, the pixel value in the image is used directly in Eq. (1) to evaluate the fitness of each ES search structure. Thus $p_i = i(x,y)$, where $i(x,y)$ is the pixel value in the image. If input is the filtered response, then $p_i = G(x,y)$, where $G(x,y)$ is the Gabor coefficient for pixel $i(x,y)$.

2.5. Gabor filter bank

Filter banks are techniques that have been used mostly in texture segmentation problem. There exists a number of investigations concerning texture classification for texture segmentation [5,13,16]. Randen [17] gives a good comparative study of different filter banks. For the test images they are using, Gabor filter bank is neither the one that gives the best texture segmentation results for all images, nor the one that has the least computational complexity. However, Gabor filter bank generally give good texture segmentation results within a satisfactory time frame. We chose Gabor filter bank in the feature identification system not only because Gabor filters are among the first filters used for texture analysis but also because it provides optimal joint resolution in both the spatial and the spatial frequency domains [6].

We used a bank of even-symmetric Gabor filters. This is, in part, due to a report by Malik and Perona [23] who hypothesize a psychophysical role for such filters. The basic even-symmetric Gabor filter oriented at 0° is a band-pass filter with impulse response:

$$h(x,y) = e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)}\cos(2\pi f_0 x),$$

where f_0 is the radial center frequency. Other orientations are obtained by rotating the reference coordinate system, (x,y).

[1] We noted earlier that selection in ES is deterministic. While that is true, when selection is combined with the other operators, the overall ES search process is probabilistic.

2.6. Filter selection scheme

The rationale of the filter selection scheme is the premise that only a subset of filtered images can together explain a 'significant' portion of the intensity variations in image $I'(x,y)$. Reconstruction of the input image $I(x,y)$ is possible by adding all of the filtered images. Here we assume that $I'(x,y)$ is a good approximation of the original image. By using only a subset of filtered images, the main features of the original image are preserved while the computational and storage complexity are reduced.

In Jain and Farrokhnia [12], a sequential forward selection procedure is described for choosing the best filter responses. This approach determines the best subset of the filtered images by examining all possible filters with different frequencies, orientations and sizes. An exhaustive search, however, is computationally prohibitive. As a result, [12] suggest using energy approximation in the Fourier domain. The following approximate filter selection scheme modified from theirs.

(1) Compute E_i (energy) for $i=1,\ldots,n$ in the Fourier domain using Eq. (2).

$$E_i = \sum_{x,y} [r(x,y)]^2 = \sum_{u,v} |R_i(u,v)|^2. \quad (2)$$

where $r(x,y)$ is a pixel value in the image and $R_i(u,v)$ is the corresponding Fourier Transform coefficient.

(2) Sort the filter responses based on their energy.
(3) For images having a known number of features, choose an equal number of filter responses having the highest energy.
(4) For images having an unknown number of features, choose as many filters as needed to achieve a predefined portion of energy above an experimentally determined ratio.

2.7. Feature image construction

When constructing a feature image, the first step is evaluation of the local energy in filter responses In essence, each Gabor filter is a band-pass filter with selective frequency and orientation properties. The local energy evaluation function converts regions where the local pass band energy is strong into high gray-level values and conversely for weak energy regions. Normally, accurate energy estimation requires high spatial frequency, while accurate edge-detection requires high spatial resolution. These have to be balanced via a smoothing filter.

Strictly speaking, energy is defined with a squaring nonlinearity. However, a generalized energy function may opt to use other alternatives. Numerous nonlinearities have been applied in the literature [7,8,12,21]. Some of the most popular are the magnitude $|\cdot|$, the squaring $(\cdot)^2$, and the rectified sigmoid $|\tanh(\alpha\cdot)|$. We use the squaring energy function to construct the feature image as described in Section 3.2.

3. Experiments

The experiments are constructed to be illustrative, i.e. chosen to exemplify the performance of feature identification using ES. In order to demonstrate the ability to find features in images using ES, unsupervised feature identification was performed on images of various geometric complexity and textures.

The presentation of the experiments is organized in three parts. Section 3.1 describes the test images used in our experiments. In Section 3.2, we discuss the Gabor filter bank design, the implementation of post-processing rules and consistency verification criteria, as well as parameter settings for ES. In Section 3.3, we show the feature identification results for images containing natural and synthetic scenes.

3.1. Test images

All test images have dynamic ranges of 8 bits per sample. The tested images were categorized into synthetic images (Fig. 4) and natural scene images (Fig. 5). The texture in the synthetic images was generated by a high frequency sinusoid superimposed onto a low frequency sinusoid. The background in Fig. 4(d) and (e) was generated by a low frequency sinusoid. The synthetic images are used to illustrate the conjecture that if proper filter banks are prescribed, feature F contained in an image can be found using ES. The natural scene images are used to illustrate that even if the filter banks are not optimal, features can still be found with redundant filters. Even though these redundant filters will inevitably result in higher computational complexity, it shows that an optimal filter bank is not necessary for feature identification using this method.

3.2. Experimental setup

In the literature, different choices for filter banks are reported. Basically, they can be categorized as either a priori fixed filter banks or optimized filter banks. The a priori fixed filter banks include Laws filter masks, Ring and wedge filters, the dyadic Gabor filter bank, Wavelet transforms, packets, frames and Discrete Cosine Transform (DCT). The optimized filter banks may include eigen-filter bank, optimal representation Gabor filter bank, optimal two/multi-class Gabor filter bank, optimal FIR filters and back propagation designed masks [17].

Jain and Farrokhnia [12] recommend the dyadic Gabor filter bank. It is the one we select to distinguish different textures and oriented patterns. The dyadic Gabor filter bank is a set of Gaussian shaped band-pass filters with dyadic coverage of the radial spatial frequency range and multiple

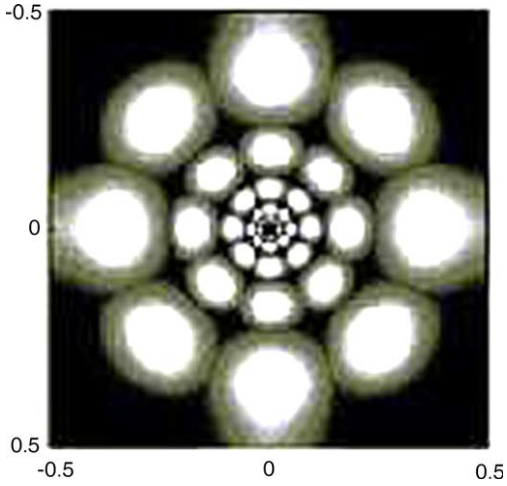

Fig. 3. Filter responses for dyadic Gabor filters. The axes are in normalized spatial frequency.

orientations. [12] concluded that the choice of Gabor filters can be justified by filters' joint optimum resolution in time and frequency. Fig. 3 shows the frequency response of the dyadic Gabor filter bank with radial frequency of $\sqrt{2}/2^6$, $\sqrt{2}/2^5$, $\sqrt{2}/2^4$, $\sqrt{2}/2^3$ and $\sqrt{2}/2^2$ and orientations of 0, 45, 90 and 135°. The maximum amplitude response over all filters is plotted. One center-symmetric pair of lobes represents a Gabor filter. For synthetic images, filters with lower radial frequencies (e.g. $\sqrt{2}/2^2$ and $\sqrt{2}/2$) are not very useful because the captured spatial variations are too large to explain textural variations. However, for natural scene images, it is necessary to capture spatial variations with low frequency. As a result, radial frequency $\sqrt{2}/2$ is not used for the synthetic images in our experiments, while for images containing natural scenes, only $\sqrt{2}/2^4$, $\sqrt{2}/2^3$ and $\sqrt{2}/2^2$ are tested.

The energy contained in each filter response is evaluated using a 2-D FFT power spectrum for the purpose of selecting proper filter responses. For synthetic images, since we know the number of features (N) in the image, the filter responses for which the total energy ranked in the top N are selected. For natural scene images, filter responses are sorted by their contribution to the total energy, which is calculated as the sum of energy contained in all filter responses. Here, an empirically determined percentage of 40% is used. Beginning with the highest response filter then continuing in descending order, filters are chosen for which the cumulative energy is no less than 40% of total energy. Since the purpose is not to reconstruct the image, 40% is sufficient to select a reasonable number of filters.

The local energy function used to construct the feature image (see Fig. 2) is described in Section 2.6, i.e. nonlinearity and smoothing with a Gaussian smoothing filter. We used magnitude squaring $(\cdot)^2$ for its simplicity as well as ability to emphasize a feature. The size for Gaussian smoothing filter is a function of the band center frequency. In the experiments, since the band center frequency of the synthetic images is known, and the band center frequency for natural images is relatively low, we chose an 8×8 window for Gaussian smoothing filter.

The final step in feature image construction is post-processing rectification, which include dynamic range adjustment and histogram equalization. Here, rectification is understood as the operations of transforming amplitudes to the dynamic range of 0–255 and increasing the contrast between regions in the filtered image. This scaling does not affect the relative difference in the strength of the responses in different regions.

The ES shell program used is the Optimum Seeking Methods (OptimA) package written by Menhnen and distributed with a text by Schwefel [18]. After incorporating our own core objective function and image reading/writing functions, we set up the experiment on a SUN Ultra 5 workstation. The parameters used are as follows. We tested $(\mu+\lambda)$ selection and (μ,λ) selection with same population size setting—50 for the parent population and 300 for the descendant population. The recombination operators on object variables and control variables are discrete recombination on object variables and panmictic intermediate recombination of control variables, respectively. A convergence criterion is, tested after each 10 generations and a maximal computation time of 20 s (CPU time) is set to terminate the search in case the convergence criterion is not met.

A comparable region identification method is based on selecting the subimages having maximum entropy [24] given an a priori partition. (Entropy is computed over

Table 1
Summary of test conditions and results for feature identification system

Images		Gabor filter bank		Consistency verification threshold	Features found
		Radial frequency	Orientation (°)		
Synthetic images	a_1	$\sqrt{2}/2^4$	0	200	1
	a_2		45		1
	a_3		90		1
	a_4		135		1
	a_5		0		1
	a_6		45		2
			135		
	a_7		0		3
			45		
			90		
	a_8		0		4
			45		
			90		
			135		
Natural scene images	n_1	$\sqrt{2}/2^2$	0	180	1
	n_2	$\sqrt{2}/2^2$	90	170	1
	n_3	$\sqrt{2}/2^2$	45	205	2
	n_4	$\sqrt{2}/2^2$	45	120	3
			0	170	
	n_5	$\sqrt{2}/2$	0	120	1
	n_6	$\sqrt{2}/2^2$	0	170	1

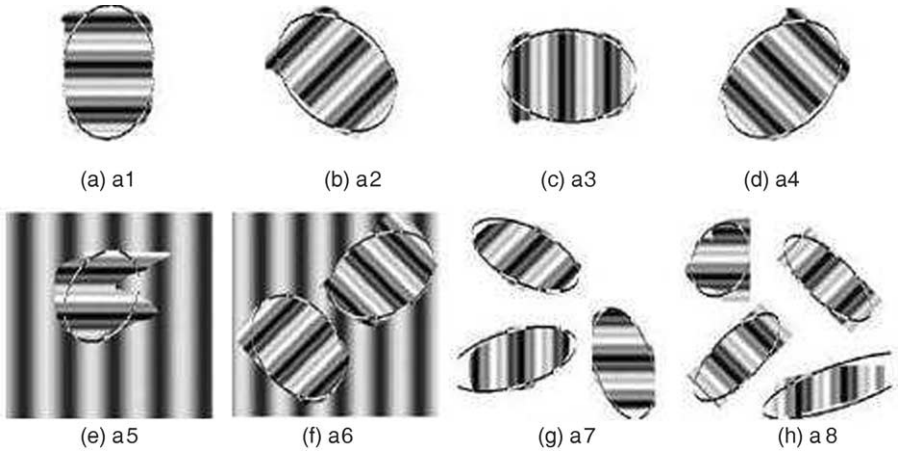

Fig. 4. Results of experiments on synthetic images: (a–d) one feature with orientation at 0, 45, 90, 135°, respectively; (e) one feature with different radial frequency than it's background; (f) two features with different orientation, 45 and 135°, on a background with different central frequency; (g) three features with different orientation; (h) four features with different orientation.

normalized intensity but could just as well be a function of filter responses.) Selection via entropy maximization has roughly the same computational cost for *a single* a priori partition. Our method might be viewed as a search across all partitions simultaneously.

The threshold defined for the bright intensity, which usually represents higher energy, is used as a consistency verification criterion. The threshold should be selected so that it will discourage ellipses from expanding into adjacent regions. This was accomplished by looking for 'valleys' in the histogram of the rectified feature image or original image if the optional feature enhancement step is not applied [4]. The threshold will separate the population into two sets. One contains candidate solutions with average intensity greater than the defined threshold and is used to form the mating pool; the other set contains solutions with average intensity less than the defined threshold and is discarded.

3.3. Test results

This section presents the results of experiments on both synthetic and natural scene images. Both types of images are chosen so that some images contain ellipse-like geometry features and others contain features with a more arbitrary geometry. Table 1 shows the summary of test conditions and results for synthetic images as well as those containing natural scenes.

For synthetic images containing simple textures, the results are straightforward. Features with different

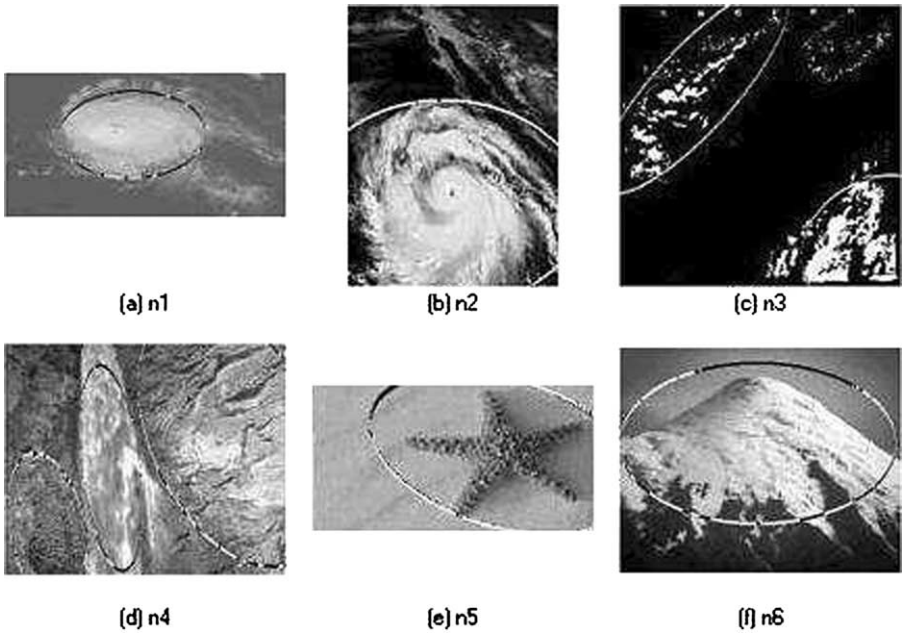

Fig. 5. Results of experiments on natural scene images: (a) and (b) one feature with ellipse-like geometry; (c) and (d) more features identification; (e) and (f) one feature with non-ellipse-like geometry.

orientations are detected and different textures are differentiated both from each other and from the background (Fig. 4). For natural scene images, as might be expected, the Gabor filter bank is not able to reflect all texture variations (Fig. 5). However, with the help of an empirically determined threshold for consistency verification, the ES algorithm can find all pre-defined interesting features in the images. We believe that the results are just as robust with the consistency verification threshold (and other ES parameters) when an appropriate filter bank is found for a specific feature.

Based on the experimental results on test images, both $(\mu+\lambda)$ selection and (μ,λ) selection find the optimum. However, (μ,λ) selection shows better convergence performance. This conforms to the recommendation by Bäck [22]. It is also obvious that even though features with non-ellipse-like geometry can be found, the inevitable side effect posed by the ES search structure is that a small portion of adjacent non-feature areas is also included. Fig. 4 contains synthetic images that illustrate two characteristics of the method-inclusion of non-feature areas given concave shapes, e.g. part (e) of the figure and the method's effectiveness at adapting to orientation. Fig. 5 shows the performance over six realistic images with widely varying feature shapes and textures.

4. Conclusion and discussion

A generic framework is described here to automate feature identification in images This work also generalizes the identification process by parameterizing the criterion for 'interesting feature.' Based on a conjecture that certain filters will give prominent responses to certain features, we transform the feature identification problem into a numerical optimization problem and use ES with an ellipsoid search structure.

Tests on synthetic and natural scene images have been described. We used a Gabor filter bank to differentiate features and used the 2-D FFT power spectrum for filter response selection. Each feature image was constructed using a local energy evaluation function $(\cdot)^2$, Gaussian smoothing filter, and rectification. Consistency verification is defined as hard thresholding. The work also confirms the recommendation by Bäck [3] that (μ,λ) selection has better performance than $(\mu+\lambda)$ selection regarding accelerating effects of self-adaptation.

We emphasize that the proposed system is extremely effective for feature identification. The system performs well and the rate of correct feature identification is nearly perfect in all simulation results. However, a small portion of adjacent non-feature regions is sometimes included when features do not conform to the ellipse-like geometry of the search structure.

A final issue should be noted is the initial value of the search structure-sometimes called the embryo. By changing its value while keeping other parameters fixed, we tested for robustness. It is not surprising that the search still converged to the global optimum. The only case that the search result is sensitive to the embryo was when there were more than one feature that exhibited similar responses to filter bank and the features are not adjacent in the image. A more complete discussion of embryos and their effect on the search is out of the scope of this paper.

Appendix A. Object and control variables in evolution strategies

Before describing the ES representation, it is important to note that ES's are specifically designed for real function evaluation. That is, an ES searches for $\vec{x}^*$ such that $f(\vec{x}^*) = \mathrm{opt}_{\vec{x}} f(\vec{x})$.

An interpretation of an ES organism is an l-dimensional jointly distributed normal variate with mean $\vec{x}$ and standard deviation $\vec{\sigma}$. The orientation of the distribution in l-space is determined indirectly by the covariances, and directly by $\vec{\theta}$.

For object identification purpose, different shapes of polygon such as rectangle can be used as search structure. The search ellipse covers a region in the images with adjustable five parameters, which are the center point (x_1,x_2), the axes (x_3,x_4) and the angle x_5 between axes x_3 and x_4. These five parameters are treated as object variables in our optimization problem. The idea of using ellipse as search structure is inspired by the 2-D ES representation.

If the optimization problem is defined on two object variables, two examples of the organisms $\langle 10.0\ 8.0;\ 4.0\ 2.0;\ 0.0\rangle$ and $\langle 25.0\ 20.0;\ 2.0\ 4.0;\ \pi/2\rangle$ are illustrated in Fig. A1—the former in the lower left and the latter in the upper right. The crosshairs intersect at the mean and the height/width represent two standard deviations in the appropriate directions.

Although it looks like the ellipse search structure, the 2-D ES representation has a different meaning. The 2-D ES

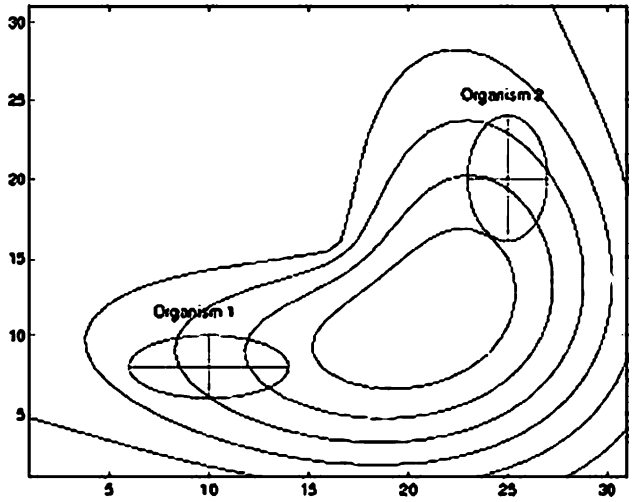

Fig. A1. Illustration of ES organisms.

representation shows the current position in the search space, the possible direction and step size for next move. On the other hand, the ellipse search structure is defined on 5-dimensions and, therefore, it involves a hypersphere representation in search space and is difficult to visualize.

References

[1] R. Adams, L. Bischof, Seeded region growing, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (6) (1994) 641–647.

[2] C.A. Ankenbrandt, B. Buckles, F. Petry, Scene recognition using genetic algorithms with semantic nets, Pattern Recognition Letters 11 (2) (1990) 285–293.

[3] T. Bäck, Evolutionary Algorithms in Theory and Practice, Oxford University Press, Oxford, 1995.

[4] B.P. Buckles, C. Kousougeras, S. Amer, J.-Y. Chuang, X. Yuan, Image decomposition using evolutionary strategies, Proceedings of JCIS'98, vol II, Second International Workshop on Evolutionary Algorithms, Rersearch Triangle Park, NC, 1998. pp. 395–398.

[5] C. Carson, S. Belongie, H. Greenspan, J. Malik, Region-based image querying, IEEE Workshop on Content-Bases Access of Image and Video Libraries, Puerto Rico, June, 1997 pp. 42–49.

[6] J.G. Daugman, Uncertainty relation for resolution in space, spatial-frequency, and orientation optimized by two-dimensional visual cortical filters, Journal of the Optical Society of America 2 (1985) 1160–1169.

[7] D.F. Dunn, W.E. Higgins, Optimal Gabor filters for texture segmentation, IEEE Transactions on Image Processing 4 (7) (1995) 947–964.

[8] D.F. Dunn, W.E. Higgins, J. Wakeley, Texture segmentation using 2-D Gabor elementary functions, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2) (1994) 130–149.

[9] D.A. Forsyth, J. Malik, M.M. Fleck, H. Greenspan, T. Leung, S. Belongie, C. Carson, C. Bregler, Finding pictures of objects in large collections of images, Proceedings of the International Workshop on Object Representation in Computer Vision II, ECCV'96, Cambridge, England, April, 1996 pp. 335–360.

[10] A. Hill, C. Taylor, Model-based image interpretation using genetic algorithms, Image and Vision Computing 10 (5) (1992) 295–300.

[11] N. Howe, Percentile blobs for image similarity, IEEE Workshop on Content-based Access of Image and Video Databases, Santa Barbara, California, June, 1998.

[12] A.K. Jain, F. Farrokhnia, Unsupervised texture segmentation using Gabor filters, Pattern Recognition 24 (12) (1991) 1167–1186.

[13] P. Kruizinga, N. Petkov, S.E. Grigorescu, Comparison of texture features based on Gabor filters in: V. Roberto et al. (Ed.), Proceedings of the 10th International Conference on Image Analysis and Processing, Venice, Italy, September (1999), pp. 142–147.

[14] E. Ozcan, C.K. Mohan, Partial shape matching using genetic algorithms, Pattern Recognition Letters 18 (1997) 987–992.

[15] E.J. Pauwels, G. Frederix, Finding salient regions in images, Computer Vision and Image Understanding 75 (1/2) (1999) 73–85.

[16] J. Puzicha, T. Hofmann, J.M. Buhmann, Non-parametric similarity measures for unsupervised texture segmentation and image retrieval, Proceedings of the IEEE International Conference on Computer Vision (1999); 1165–1173.

[17] T. Randen, J.H. Husoy, Filtering for texture classification: a comparative study, IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (4) (1999) 291–310.

[18] H.P. Schwefel, Evolution and Optimum Seeking, Wiley/Interscience, New York, 1995.

[19] S. Shah, J.K. Aggarwal, A Bayesian segmentation framework for textured visual images, Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition, Puerto Rico, June, 1997, pp. 1014–1020.

[20] M. Singh, A. Chatterjee, S. Chaudhury, Matching structural shape descriptions using genetic algorithms, Pattern Recognition 30 (9) (1997) 1451–1462.

[21] M. Unser, M. Eden, Nonlinear operators for improving testure segmentation based on features extracted by spatial filtering, IEEE Transactions on Systems, Man and Cybernetics 20 (4) (1990) 804–815.

[22] L. Yue, H. Guo, Texture image retrieval by universal classification for wavelet transform coefficient, Proceedings of the 1997 International Conference on Image Processing, Washington, DC, vol. 3, 1997.

[23] J. Malik, P. Perona, Preattentive texture discrimination with early vision mechanism, Journal of Optical Society of America A 7 (5) (1990) 923–932.

[24] P. Chalermwat, High performance automatic image registration for remote sensing, PhD dissertation, George Mason University, 1999.