Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

PERGAMON

Pattern Recognition 37 (2004) 1797–1807

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

A clustering fuzzy approach for image segmentation

L. Cinque[a,*], G. Foresti[b], L. Lombardi[c]

[a] *Dipartimento di Informatica, Universita "La Sapienza", Via Salaria 113, Roma 00198, Italy*
[b] *Dipartimento di Matematica e Informatica, Via delle Scienze 208, Udine 33100, Italy*
[c] *Dipartimento di Informatica e Sistemistica, Via Ferrata 1, Pavia 27100, Italy*

Received 5 November 2002; received in revised form 4 April 2003; accepted 4 April 2003

Abstract

Segmentation is a fundamental step in image description or classification. In recent years, several computational models have been used to implement segmentation methods but without establishing a single analytic solution. However, the intrinsic properties of neural networks make them an interesting approach, despite some measure of inefficiency. This paper presents a clustering approach for image segmentation based on a modified fuzzy approach for image segmentation (ART) model. The goal of the proposed approach is to find a simple model able to instance a prototype for each cluster avoiding complex post-processing phases. Results and comparisons with other similar models presented in the literature (like self-organizing maps and original fuzzy ART) are also discussed. Qualitative and quantitative evaluations confirm the validity of the approach proposed.
© 2004 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.

Keywords: Segmentation; Fuzzy ART; Data clustering

1. Introduction

Segmentation is an essential issue in image description and/or classification. It is based on a definition of uniformity, which usually depends on the particular task at hand and its context [1]. Informally, we refer to this process as one that splits an image into a set of non-overlapping uniform connected regions such that any two adjacent ones are not similar.

Many different segmentation approaches have been developed that cannot be generalized under a single scheme. Interesting surveys can be found in Refs. [2–4]. According to Ref. [2], most approaches are based on similarity and difference and, particularly, can be divided into different categories: thresholding [5], clustering [6–8], edge detection [9,10] and region extraction [11]. In this paper, we propose a clustering based approach.

Clustering methods analyze a vectorial input space, so, when an image is given, a pre-processing step is requested to calculate a feature vector for each pixel. These *n*-dimensional vectors (usually called *patterns*) are evaluated on the basis of the pixel values of a limited region. Similar vectors will be associated to pixels belonging to the same region while different ones will be assigned to corresponding pixels belonging to different regions. This means that these vectors, in an Euclidean space, characterize a number of clusters, one for each region. Our method finds these clusters and classifies all pixels under the same label. The segmentated image is found by mapping the vectors back to the pixels. Clustering ignores spatial information, so that we could have disconnected regions under the same label (i.e. it discriminates between differently perceived regions and not between similar ones).

Several computational models have been adopted to implement segmentation methods. Neural networks can be considered an interesting approach due to their properties: parallelism could allow real-time systems to be used, increase fault tolerance in knowledge distribution (and critical scenario applications), further reduce the time taken to learn

* Corresponding author. Tel.: +39-06-4991-8508; fax: +39-06-8419188.
E-mail address: clinque@dsi.uniroma1.it (L. Cinque).

0031-3203/$30.00 © 2004 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2003.04.001

and enhance the ability to work in noisy real scenarios. Some neural network applications for image segmentation can be found in Ref. [4].

Models usually used for clustering are the self-organizing maps (SOM) [12] and the adaptive resonance theory architecture family (ART) [13]. Examples of SOM applications to image segmentation can be found in Refs. [14–17]. In Ref. [14], a SOM is used to segment gray level images (feature vectors include gray level, mean and variance). In Refs. [16,17], a SOM is used to segment color images (feature vectors are characterized by color and texture parameters). In all these works, SOM implies some constraints: the need to choose the number of clusters a priori, heavier computational complexity (as recognized by Kohonen [12]) and merging the groups representing the same cluster [18] (because the SOM, by approximating the distribution patterns, finds more than one prototype representing the same cluster [19]). Moreover, successive SOM results depend on the training phase and this implies the choice of representative training examples.

In contrast, ART2 [20] allocates new cluster buckets whenever they are necessary, permitting lower computational complexity (see the experiments performed in Ref. [21]) and finds a prototype for each cluster. However, at the same time, this solution is not straightforward to implement because it needs to manage too many parameters. ART2 is used in Ref. [22] to segment multimodality images. This model, like all others deduced from the ART [23], does not separate the classification from the learning step. Indeed, it is always capable of improving its knowledge base when the current knowledge base is insufficient. In Ref. [24], after discussing ART2's problems (especially management difficulties), a modified ART1 is suggested and applied to a LANDSAT image segmentation task. This new approach, called SART1, makes the ART1 simpler, and gives it the capacity to handle real value patterns, whilst reducing some weaknesses in the original model.

Taking into the account the above considerations and the vast work previously done on image segmentation techniques, our goal has been to find a simple new model able to instance a single prototype for each cluster (to avoid the post-processing phase) with a lower computational complexity than the models mentioned above. Among the various proposals, we selected the ART1 model [23] despite its inability to handle real value patterns, because it is very simple to implement (the number of its parameters is small). The literature offered one ART1-type architecture, called fuzzy ART [25], which, unfortunately, on closer examination and direct experience, presented some problems. Our paper suggests a solution to some of these problems and analyzes an image segmentation task as a case study.

This paper is organized as follows. Section 2 introduces the fuzzy ART model and its problems. Section 3 describes the changes applied to the fuzzy ART, the properties of the new approach we are proposing and a theoretical comparison with other similar models. We have introduced a pattern grouping module for the segmentation system made up of a feature extractor module together with the network itself. Our segmentation system is tested on a real image set and qualitatively and quantitatively compared to other models. Results from the implemented algorithm are shown and discussed in Section 4; while our conclusions are given in Section 5.

2. The fuzzy art model

Fuzzy ART is a pure winner-takes-all architecture able to instance output nodes whenever necessary and to handle both binary and analog patterns. The adjective fuzzy derives from the functions it uses [26], although it is not actually fuzzy [27]. In this section, we describe this architecture when applying it to data clustering (more details can be found in Ref. [25]).

To perform data clustering, Fuzzy ART behaves like a Leader algorithm [7], instancing the first cluster coinciding with the first input and allocating new buckets when necessary (in particular, each output node represents a cluster from a prototype). Unlike the Leader algorithm, fuzzy ART prototypes are not constant but can adapt to new inputs (i.e. they can learn).

Algorithm 1. Fuzzy ART performing data clustering

While a new input exists **do**
  Activate all prototypes
  While an active prototipe exists **do**
    Find, with function (1) (see below), the prototype with greatest compatibility with the input.
    Evaluate, with function (2), the compatibility the input has with the prototype.
    If this value is greater than, or equal to, the vigilance parameter ρ ($\rho \in [0, 1]$) then
      Adjust the prototype with function (3), and skip to a new input.
    else
      Inhibit the prototype until the next input (i.e. it will not compete for the actual input) and try a new prototype.
    Done
    If no active prototype exists instance a new one and skip to
      the next input.
Done.

The choice function used by Fuzzy Art is the following:

$$A(I^t, W_J^t) = \frac{|MIN(I^t, W_J^t)|}{\alpha + |W_J^t|} = \frac{\sum_{i=1}^{n} MIN(I_i^t, W_{Ji}^t)}{\alpha + \sum_{i=1}^{n} W_{Ji}^t}. \quad (1)$$

It computes the compatibility between a prototype and an input. The tth input pattern I^t is an n-element vector

transposed, W_j the prototype of cluster J (both are n-dimensional vectors), α the choice parameter and MIN the fuzzy set intersection operator [26]. α is a value greater than, or equal to, zero and acts as a tie breaker.

The test function M is

$$M(W_J^t, I^t) = \frac{|MIN(I^t, W_J^t)|}{|I^t|} = \frac{\sum_{i=1}^{n} MIN(I_i^t, W_{Ji}^t)}{\sum_{i=1}^{n} I_i^t}. \quad (2)$$

This computes the compatibility between the input and the selected prototype. The test is passed if this value is greater than, or equal to, the vigilance parameter $\rho \in [0,1]$. Intuitively, ρ indicates how similar the input has to be to the prototype to allow it to be associated with the cluster the last prototype represents. As a consequence, a greater value for ρ implies smaller clusters, a lower value wider clusters.

Let R be the prototype adjusting function, i.e.,

$$R(I^t, W_J^t) = \beta MIN(I^t, W_J^t) + (1-\beta)W_J^t, \quad (3)$$

where the learning parameter $\beta \in [0,1)$ weights the new and old knowledge, $MIN(I^t, W_J^t)$ and W_J^t, respectively. It is worth observing that this function is not increasing, that is $W_{Ji}^{t+1} \leqslant W_{Ji}^t \; \forall i, J$.

2.1. Fuzzy ART's problems

Fuzzy ART suffers both from problems inherited from the ART1 model and its own: this section deals with some of these (partially following Ref. [28]). The problems are:

(a) Functions (1) and (2) only estimate, not measure, the extent of match between any two patterns: function (1) computes the match between the prototype and the input, but not the opposite, function (2) only computes and tests the match between the input and the prototype.
(b) The final result depends on the input presentation order.
(c) *Heavy computational complexity*: In general, for each input, the prototype list is scanned several times to find the best match or to deduce the need for a new one. However, further improvement could be made by reducing the single scan computational effort.
(d) *Overfitting*: At the end of the process some prototypes not representing any cluster could still survive, these prototypes introduce memory wasting and possibly classification errors.
(e) After analyzing the results of the simulations performed using our implementation, there is evidence that it is difficult to obtain a good vigilance parameter. Some results, in fact, are obtained for large vigilance intervals only to change, suddenly, near the value of 1. This involves repeated trials and the loss of the interval's intuitivity .
(f) *Plasticity loss*: With complete memory exhaustion, to protect the knowledge acquired up to that moment, no novelty is accepted (i.e. no new clusters are instanced). This approach does not seem to be the most correct strategy for any application: although knowledge safeguarding is a very important objective achieving results should have precedence.

3. Our approach

To solve the problems shown in the previous section and to add other improvements, we modified the original fuzzy ART. Our choice/test function was the one suggested in Ref. [28], where no analysis, use or simulation is provided. This function was obtained by multiplying the choice (1) and the test (2) original fuzzy ART functions. The new choice/test function is defined as

$$F(I, W_J) = \frac{[|MIN(I, W_J)|]^2}{|I| \cdot |W_J|} = \frac{[\sum_{i=1}^{n} MIN(I_i, W_{Ji})]^2}{\sum_{i=1}^{n} I_i \cdot \sum_{i=1}^{n} W_{Ji}}. \quad (4)$$

It is worth stressing that the use of function (4) places our model approach in the SART framework [24], although it is different from the SART1 [24] and fuzzy SART [28] (where Kohonen's strategies are exploited).

Function (4) ensures that, if the cluster with the greatest compatibility does not satisfy the test, no other cluster can. As a consequence, we can eliminate the repeated prototype list scanning and, hence reduce computation costs. It is not simple to estimate the gain as a function of the reduced list scanning because it is also impossible to estimate this cost with the original fuzzy ART. In any case, while our algorithm needs just one scan for each input, the original fuzzy ART, in the worst case, needs k ones, where k is the prototype list dimension (so, intuitively, there is a gain from $O(k^2)$ to $O(k)$.

Algorithm 2. Proposed model for data clustering

While a new input exists **do**
  Find, with function (4), the prototype with greatest compatibility with the input and vice versa.
  Evaluate, using function (2), the compatibility between the input and the prototype.
  If this value is greater than, or equal to, the vigilance parameter ρ ($\rho \in [0,1]$) **then**
    Adjust the prototype with function (3), and skip to a new input..
  Instance a new prototype.
Done.

Function (4), used to compute the choice and perform the test at the same time, has the interesting property of reducing the problems related to the presentation order of the input patterns.

This is possible because it tests the match between the prototype and the input and, at the same time, its opposite,

so the prototypes need not undergo heavy shifts. Indeed, with the original function, it could be possible to choose a prototype which is not so compatible with the input (despite being greater) but is still be able to pass the test. When adjusting this prototype against the input, it could lose part of its structure thereby undergoing a large shift. Although this function reduces this problem, it does not solve it because the results depend on the vigilance parameter as well.

In addition, function (4) makes searching for the vigilance parameter and eliminating the choice parameter simpler. However, it is worth observing that it is not translation invariant: any normalization solves the problem, as we have found. In particular, we use complement coding scheme [25], which guarantees that the length of all input vectors are identical.

Theorem 1. *Complement coding makes function* (4) *translation invariant.*

Proof. Let $I_1=(A,B,1-A,1-B)$ and $W_1=(E,F,G,H)$ be the first couple, and $I_2=(A+C,B+D,1-(A+C),1-(B+D))$ and $W_2=(E+C,F+D,G-C,H-D)$ the second one, where A and B are real numbers in the range [0,1], while C and D are negative or positive real numbers. The choice/test function for the first couple becomes.

$$F(I_1,W_1)=\frac{[\sum_{i=1}^{n} MIN(I_i,W_{Ji})]^2}{\sum_{i=1}^{n} I_i\cdot\sum_{i=1}^{n} W_{Ji}}$$

$$=\frac{[MIN(A,E)+MIN(B,F)+MIN((1-A),G)+MIN((1-B),H)]^2}{2(E+F+G+H)}. \quad (5)$$

For the second couple, we obtain

$$F(I_2,W_2)$$

$$=\frac{[MIN(A+C,E+C)+MIN(B+D,F+D)+MIN(1-(A+C),G-C)+MIN(1-(B+D),H-D)]^2}{2[(E+C)+(F+D)+(G-C)+(H-D)]}$$

$$=\frac{[MIN(A,E)+C+MIN(B,F)+D+MIN((1-A),G)-C+MIN((1-B),H)-D]^2}{2[E+C+F+D+G-C+H-D]}$$

$$=\frac{[MIN(A,E)+MIN(B,F)+MIN((1-A),G)+MIN((1-B),H)]^2}{2(E+F+G+H)}. \quad \square \quad (6)$$

We use the original fuzzy ART adjusting function but within a fast learning asset, which is obtained with $\beta=1$. It is worthwhile stressing that fast learning may not be adequate in some scenarios where it could be substituted with the *fast commit and slow recode* strategy [25]. Fast learning seems suitable for our test set.

After having achieved the terminating condition, unrequired prototypes are discarded. Our implementation provides a variable for each prototype, with the purpose of establishing whether the correspondent cluster has been allocated during the current set presentation or a previous one. This tool avoids eliminating the clusters allocated for previous sets not required by the current one. Our implementation erases the unrequired prototypes when the terminating condition is obtained and not at the end of each presentation as in Ref. [28]. According to our experiments, these two strategies achieve the same results with some computational savings by performing it on reaching the terminating condition.

To resolve plasticity loss a prototype not yet requested for the set in input could give up its place to a new one. Since the repeated prototype list scanning has been eliminated, further improvements ensue, thereby reducing single scan computational complexity [29]. To attain this improvement the Hebb competitive rule [30] was applied: by keeping a pointer on the winner of the previous presentation, the search could be limited to a single prototype and its neighborhood. This is possible because this rule creates a topological organization among the clusters, keeping an adjacent relations set among the prototypes. The strategy consists in comparing the value returned by function (4) for the pointed prototype with those returned for its neighboring ones. If the pointed prototype's value is greater than the other ones', it is the winner. Otherwise, the prototype with the greatest value and its neighborhood has to be tested. It is worth noting that some of these prototypes have been already tested. This strategy has some usefulness because function (4) does not allow large prototype shifting. Nevertheless, it is not possible to discuss in detail what this strategy's performance may be because in general, the neighborhood is small and not constant . Prototypes are free to shift and no longer valid relations are eliminated: in the worst case all prototypes are tested like in the original strategy.

To obtain better clustering performance, we used the strategy described in Ref. [31]. This strategy consists in

imposing two vigilance parameters. Clusters of real data are usually not compact and well distanced, with the risk of misclassifying those points a little more distant than normal. This strategy allows the problem to be solved and to obtain other benefits. In fact, besides solving the problem, it allows faster results by choosing a parameter next to 1 and another parameter as the vigilance value, thereby reducing the algorithm to the Leader parameter. By limiting the updating it is possible to get results in only one or two process cycles. The same strategy allows valuable rough results to be quickly obtained.

Quick results can be also obtained with the following tout-court strategy. Since the terminating condition is achieved when further presentation of the current input set does not modify the prototypes, it could discard already labeled patterns whose prototypes were not modified in the last presentation. Supposing that major changes occur during the first cycles, the process could terminate in two or three cycles. Almost accurate results are guaranteed by function (4), which does not allow heavy prototype shifts. Given that this strategy is applied to labeled patterns and not to new ones, it does not stop plasticity, unlike those models which reduce the learning parameter [12,24,28].

After having presented the changes we applied, we need proof that the clustering algorithm performed by our modification converges in a finite number of cycles. The terminating condition is obtained when all prototypes have reached a stable position, that is, when no new presentation of the same set can modify it. To prove this statement it becomes necessary to introduce some definitions.

Definition 1 (Sub-pattern). Pattern A is B's sub-pattern iff every A's element is lower than its correspondent in $B(A_i < B_i \ \forall i)$.

Definition 2 (Mixed pattern). Pattern A is B's mixed pattern iff some of A's elements are greater than, or equal to, their correspondents in B while for the rest it is the reverse $(\exists i \ A_i < B_i \ \& \ \exists j \ A_j \ \geqslant B_j)$.

Theorem 2. *Fast learning and complement coding imply* $|W_J| \geqslant n\rho$, *where n is the pattern length.*

Proof. For $t = 1$ (initialization step):

$$|W_J| = |I| = n \Rightarrow |W_J| \geqslant n\rho \ (\rho \leqslant 1).$$

For $t > 1$, the test is passed if

$$\frac{[|MIN(I, W_J)|]^2}{|I| \cdot |WJ|} = \frac{[|MIN(I, W_J)|]^2}{n \cdot |WJ|} \geqslant \rho \Leftrightarrow \frac{[|MIN(I, W_J)|]^2}{|WJ|} \geqslant n\rho.$$

Fast learning implies

$$W_J^{t+1} = MIN(I^t, W_J^t).$$

So, the condition for passing the test becomes

$$\frac{|W_J^{t+1}| \cdot |W_J^{t+1}|}{|W_J^t|} \geqslant n\rho.$$

From which it is possible to conclude (as $|W_J^{t+1}|/|W_J^t| \leqslant 1$):

$$|W_J^{t+1}| \geqslant \frac{|W_J^{t+1}| \cdot |W_J^{t+1}|}{|W_J^t|} \geqslant n\rho \Rightarrow |W_J| \geqslant n\rho. \qquad \square$$

Theorem 2 also proves that the category proliferation problem [32] is solved with complement coding.

Theorem 3. *From the second input presentation and the following, the test is automatically passed by adopting fast learning and complement coding.*

Proof. After the first input presentation, when choosing the cluster with the greatest compatibility, two situations could arise: a sub-pattern choosing or a mixed pattern choosing (due to the fast learning and complement coding, super-patterns do not exist).

Choosing sub-pattern W_J implies

$$\frac{[|MIN(I, W_J)|]^2}{|I| \cdot |W_J|} = \frac{|MIN(I, WJ)|}{|I|} = \frac{|WJ|}{n},$$

$$|W_J| \geqslant n\rho \ (\text{Theorem 2}) \Rightarrow \frac{|W_J|}{n} \geqslant \rho.$$

Mixed pattern choosing is possible only if the choice/test function returns a greater value than that returned for a sub-pattern. This sub-pattern must exist because of the updating carried out in the previous presentation and the no growth property of the adjusting function. If a subpattern had been chosen it would have passed the test as would the mixed pattern. $\square$

Theorem 3, to which no new cluster bucket can be added from the second presentation onwards, also states an upper limit for cluster numbers. This is an upper limit only because there could be some prototypes that do not represent any cluster, given that the fuzzy ART model can suffer from overfitting.

Theorem 4. *The clustering algorithm performed by our modification converges in a finite number of cycles.*

Proof. Since from the second input presentation and onwards the established cluster number cannot increase (Theorem 3) and as the prototypes are distinctive and downer bounded (Theorem 2, where the adjusting function is not growing), the clustering algorithm performed by our modification terminates in a finite number of cycles. $\square$

Our modification when compared with the original fuzzy ART, implies lower computational complexity, greater simplicity in searching for the vigilance parameter, less

presentation order problems, lower probability of misclassification and less memory wasting. In particular, less computational complexity is made possible by: function (4) (which allows the prototype list to be scanned only once for each input), by the topological organization (which allows the duration of the single list scanning to be reduced), and by the as yet initialized prototype competition not to mention the automatic test passing (from the second presentation onwards).

On comparing our modification with SOM, our model does not need post-processing to merge the groups representing the same cluster, or its dimensionality to be fixed a priori. This implies less memory wastage (as the SOM places some prototypes into domains which do not produce any input [19]) and lower computational complexity (see experiments performed in Ref. [21]) while its results do not rely soley on the training set (with the capacity to obtain better results in a real unsupervised scenario).

4. Experimental results and discussion

Our segmentation system is made up of two modules: the pattern grouping, obtained with the modified fuzzy ART, and the feature extractor which assigns a pattern to each pixel (see Fig. 3). To extract the features we chose a gray level mean, total and minimum variation operators [33]. These operators should allow us to discriminate among the different textures and between these and homogeneous regions. In fact, total and minimum variation should return zero for homogeneous regions and not zero for textured ones: homogeneous regions should be discriminated by different means and textured ones by different variance (Fig. 1).

To test the performance of our system, several real images were used and some of the results were compared with other models. The experiment set included different domains: views, objects and medical images. Results were

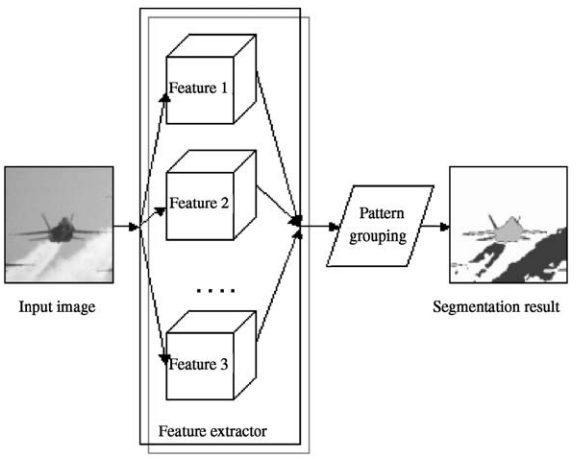

Fig. 1. The proposed segmentation system.

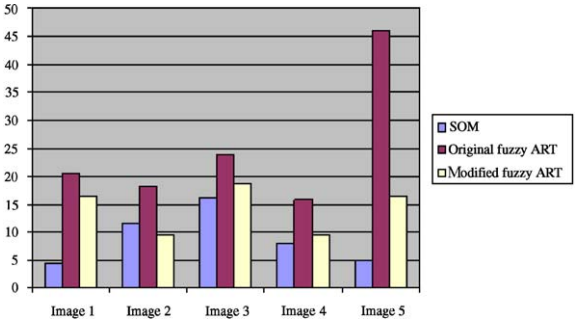

Fig. 2. Average error.

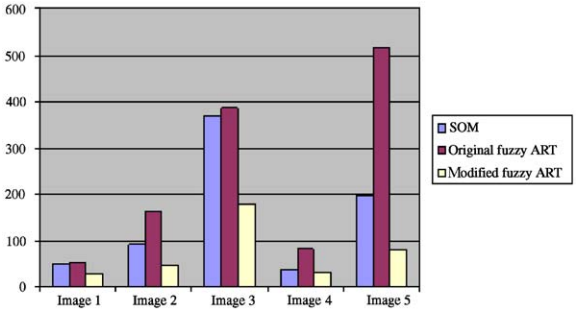

Fig. 3. $f(I)$.

compared with the clustering algorithm performed by the SOM, and the original fuzzy ART.

All the experiments were conducted by limiting human interaction to the minimum, setting just one parameter with intuitive effects, without post-processing and image labeling. Pre-processing was required to transform the pattern values in the space [0,1], as expected by the fuzzy ART.

Since feature operators extracted values using a window, 3×3, 5×5, 7×7 and 9×9 dimensionalities were tried. Minimum and total variations were exploited averaging their results in a 3×3 window.

Comparisons were made in the following context. Original fuzzy ART was simulated by an original implementation. It was used in the fast learning asset (with $\beta = 1$) with α set to zero. Values for the vigilance parameter ρ were found by trials. It is worth observing that we were not able to get good results due to difficulties in setting the vigilance parameter. In particular, we will show the results obtained with $\rho = 0.99999997019$ which allows two regions to be found (for $\rho = 0.9999999702$ more than 256 regions were found).

SOM was simulated by the SOM_pak [34]. The parameters were set by following the suggestions given in Ref. [35]. In particular, we used a rectangular map with two training stages: the first was made in 1000 steps, with 0.9 as a learning parameter and a half map as a neighborhood, and the second in 500 steps, with 0.009 as a learning parameter and three units as a neighborhood. Map size was chosen by trials (Figs. 2, 3).

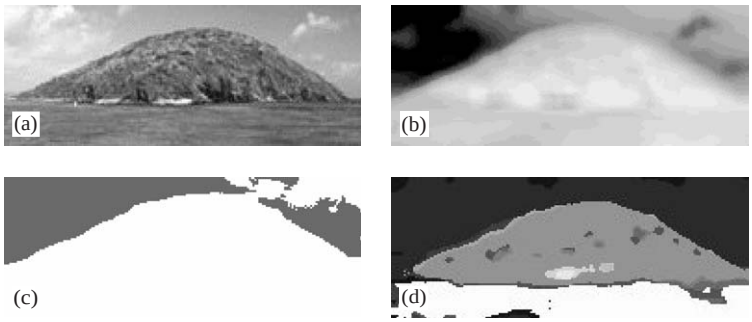

Fig. 4. (a) Original image, (b,c,d) result obtained by applying self-organizing maps, the original fuzzy ART model and the modified fuzzy ART model, respectively.

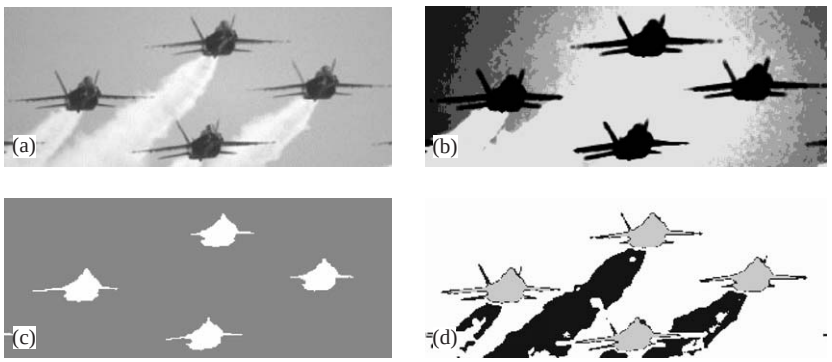

Fig. 5. (a) Original image, (b,c,d) result obtained by applying self-organizing maps, the original fuzzy ART model and the modified fuzzy ART model, respectively.

4.1. Experimental results on real images

Fig. 4a shows an image with three different irregular textures (sky, earth and sea). Figs. 4b–d show results for SOM, original fuzzy ART and our approach.

Fig. 5a shows an image whose principal entities (sky, smoke, airplanes) were almost homogeneous in the intensity of their gray level. Figs. 5b–d show results for SOM, theoriginal fuzzy ART and the modified ART. It is worthwhile observing that the airplanes, by constituting similar regions according to a perception criterion, are grouped under the same label, because clustering is not able to discriminate between similar but unconnected regions.

Fig. 6a shows an image whose principal entities (street, kennels and bushes) are not simple to obtain because made up of different percepted parts [17]. Figs. 6b and c show the results, respectively, for the SOM and the original fuzzy ART. Fig. 6d shows the result obtained with our system.

Fig. 7a shows an image whose principal entities, as in the previous figure, are not simple to obtain because it is made up of parts subject to different perceptions [16]. Figs. 7b and c show the results, respectively, for the SOM and the original fuzzy ART. Fig. 7d shows the result obtained with our system.

4.2. Comparisons with other segmentation methods

It is well known that evaluating segmentation results and comparing segmentation algorithms are not simple tasks [4,36–39]. However, one of the most widely used criteria for performance evaluation is whether the system can outline the desired or important regions in the image. We think the results obtained by our system can be regarded as reasonably good and applicable in subsequent processing. In addition, Haralick and Shapiro [3] point out that good segmentation results should present simple, uniform and homogeneous regions, without too many small holes and with simple, not ragged and spatially accurate boundaries. We believe our results satisfy these requirements. Moreover, our results are very similar to those produced by conventional clustering algorithms.

To quantitatively evaluate our experimental results, we used the function proposed in [40], which improved on the one proposed in Ref. [41]. This function, incorporating the

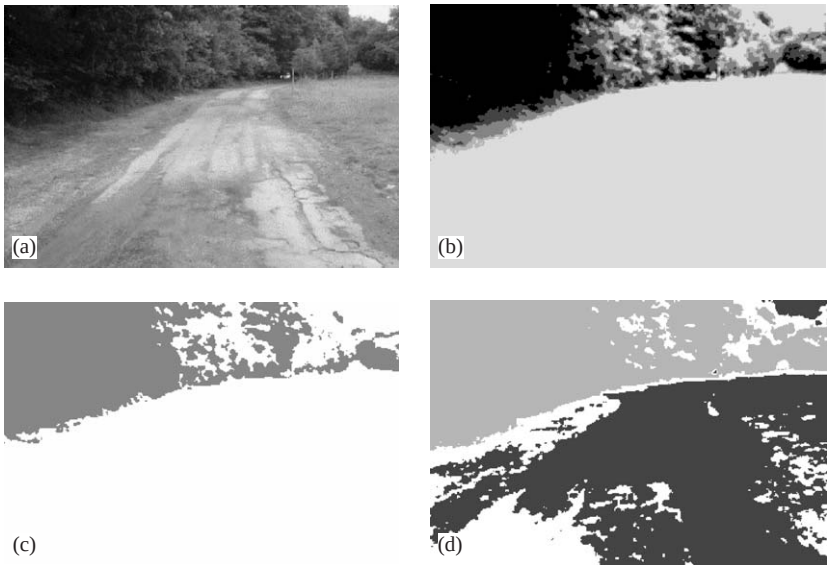

Fig. 6. (a) Original image, (b–d) result obtained by applying self-organizing maps, the original fuzzy ART model and the modified fuzzy ART model, respectively.

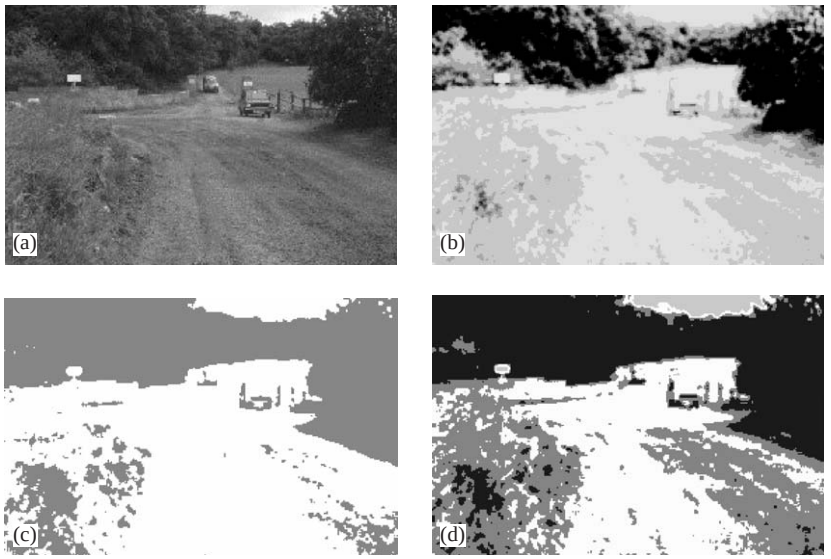

Fig. 7. (a) Original image, (b–d) result obtained by applying self-organizing maps, the original fuzzy ART model and the modified fuzzy ART model, respectively.

heuristic criteria formulated by Haralick and Shapiro, is defined as

$$f(I) = \frac{1}{1000(N \times M)} \sqrt{R} \times \sum_{i=1}^{R} \left[\frac{e_i^2}{1 + \log A_i} + \left(\frac{R(A_i)}{A_i} \right) \right], \quad (8)$$

where I is the result to be evaluated, $N \times M$ the image size, R the number of regions found, A_i the size of the ith region, e_i the average error of the ith region and $R(A_i)$ the number of regions having area $A_i e_i$ is defined as the sum of Euclidean distances between the features extracted from each pixel and those for the corresponding cluster. This function allows the segmentation result to be evaluated without labeling the image (i.e. without owning the ground truth). The smaller

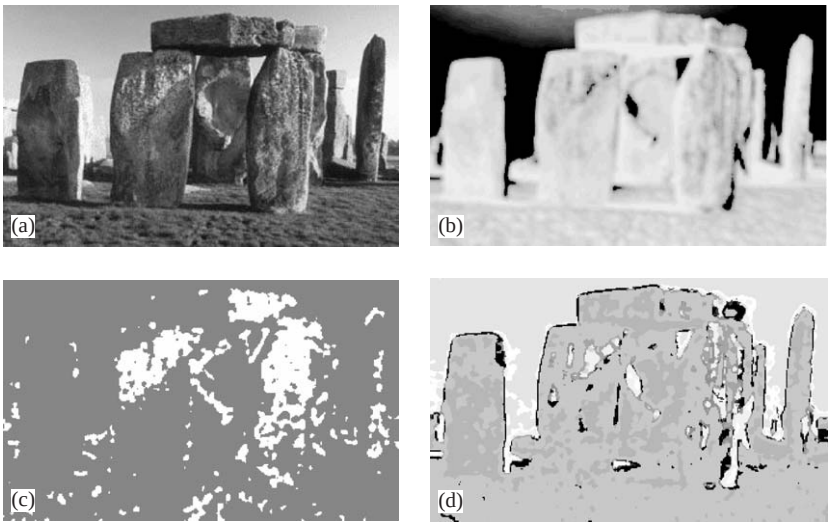

Fig. 8. (a) Original image, (b–d) result obtained by applying self-organizing maps, the original fuzzy ART model and the modified fuzzy ART model, respectively.

the value of $f(I)$, the better the segmentation result should be. More details can be found in Refs. [40,41].

Figs. 2 and 3 summarize the segmentation results, showing the average error (Fig. 2) and $f(I)$ (Fig. 3), for the SOM, the original and modified fuzzy ART, for the original images in Figs. 4–8. While SOM preserves more details, subdividing the desired region and returning noisy results, original fuzzy ART suppresses more noise, but hiding many details at the same time. Our modified fuzzy ART is a trade-off between these two types of behavior, obtaining lower $f(I)$ values. In addition, o modified fuzzy ART finds a lower number of regions than the SOM but a greater number than those found by the original fuzzy ART. Moreover, the proposed method shows a lower average error than the original fuzzy ART, but a greater one than that obtained by the SOM. The quantitative and qualitative valuing justifies the conclusion that our clustering implementation returns good segmentation results.

Naturally, as detailed in Fig. 8, minimum and total variations are not sufficient to discriminate between all textures [42–44] (as they are very similar to the contrast [45] and the gray level difference statistics [46]), but it is recognized that in general, different operators correspond to different images. In this work, specific knowledge of the acquisition process is not applied and we try to demonstrate the validity of our neural implementation to a greater extent than the operator's one.

5. Conclusions

This paper presents a modified fuzzy ART and its application to image segmentation. Of the possible approaches to segmentation, we adopted clustering. Considering the segmentation process is an ill-defined problem, because no unique analytic solution may be given to a general image set, research has always proceeded either heuristically or experimentally to devise ad hoc systems generally applicable to single working domains.

Our approach, essentially based on neural network computation, i.e., learning capacity, satisfies some of its main requirements: fast results, fault and noise tolerance. Neural network models, usually adopted for this approach, are inefficient and our proposal constitutes a good alternative, providing better performance when solving several problems. Moreover, our modification is superior to the original fuzzy ART model, solving some of the problems discussed above.

A pattern grouping module totally independent of the application was also proposed. To test its abilities in the image segmentation context a feature extractor module was developed.

The segmentation system made up of the modified fuzzy ART and the feature extractor module, was very simple to use: the user needs only to act on a single parameter with intuitive effects. This parameter determines the cluster spreading. As such system does not use specific knowledge, by adopting the most proper operators, it becomes possible to customize it to different scenarios.

A number of results and comparisons (with other models present in literature) have been presented. Qualitative and quantitative valuing confirms the validity of our approach. In particular, SOM preserves more details, subdividing the desired region and returning noisy results, and the original fuzzy ART suppresses more noise but at the same time hides many details. Our modified fuzzy ART represents a trade-off between these two patterns of behavior, obtaining lower

values for the function we used to quantitatively evaluate our results.

References

[1] T. Pavlidis, Structural Pattern Recognition, Springer, New York, 1977.

[2] K.S. Fu, J.K. Mui, A survey on image segmentation, Pattern Recognition 13 (1981) 3–16.

[3] R.M. Haralick, L.G. Shapiro, Image segmentation techniques, Comput. Vision Graphics Image Process. 29 (1985) 100–132.

[4] N. Pal, S. Pal, A review on image segmentation techniques, Pattern Recognition 26 (9) (1993) 1277–1294.

[5] J.S. Weszka, A survey of threshold selection techniques, Comput. Graphics Image Process. 7 (1978) 259–265.

[6] M.R. Anderberg, Cluster Analysis for Applications, Academic Press, New York, 1973.

[7] J.A. Hartigan, Clustering Algorithms, Wiley, New York, 1975.

[8] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, Englewood Cliffs, NJ, 1988.

[9] L.S. Davis, A survey of edge detection techniques, Comput. Graphics Image Process. 4 (1975) 248–270.

[10] T. Peli, D. Malah, A study of edge detection algorithms, Comput. Graphics Image process. 20 (1982) 1–21.

[11] S.W. Zucker, Region growing: childhood and adolescence, Comput. Graphics Image process. 5 (1976) 382–399.

[12] T. Kohonen, Self-organizing Maps, 2nd Edition, Springer, Berlin, 1997.

[13] G.A. Carpenter, S. Grossberg, A massively parallel architecture for a self-organizing neural pattern recognition machine, Comput. Vision, Graphics Image Process. 37 (1987) 54–115.

[14] R. Romagnoli, Mappe auto-organizzanti per la segmentazione di immagini, Thesis, Information science department, "La sapienza" University, Rome, Italy, 1998.

[15] R. Natowicz, F. Bosio, S. Serge, Segmentation of image sequences using self-organizing feature maps, in: S. Gielen, B. Kappen (Eds.), Proceedings of the International Conference on Artificial Neural Networks, Springer, Berlin, 1993, pp. 1002–1005.

[16] N.W. Campbell, B.T. Thomas, T. Troscianko, Segmentation of natural images using self-organizing feature maps, British Machine Vision Conference 1996, British Machine Vision Association, 1996, pp. 223–232.

[17] N.W. Campbell, B.T. Thomas, T. Troscianko, A two-stage process for accurate image segmentation, Proceedings of the Sixth International Conference on Image Processing and its Applications, IEE, London, 1997, pp. 655–659.

[18] K.C. Yao, M. Mignotte, C. Collet, P. Galerne, G. Burel, Unsupervised segmentation using a self-organizing map and a noise model estimation in sonar imagery, Pattern Recognition 33 (2000) 1575–1584.

[19] B. Fritzke, Kohonen feature maps and growing cell structures —a performance comparison, in: G.L. Giles, S.J. Hanson, J.D. Cowan (Eds.), Advances in Neural Information Processing Systems, Vol. 5, Morgan Kauffman, San Matteo, 1993.

[20] G.A. Carpenter, S. Grossberg, ART2: self-organization of stable category recognition codes for analog input patterns, Appl. Opt. 26 (23) (1987) 4919–4930.

[21] L. Xu, E. Oja, Adding top-down expectation into the learning procedure of self-organizing map, in: M. Caudill (Ed.), Proceedings of the International Joint Conference on Neural Networks, 1990, pp. I-735–738.

[22] K. Lin, S. Huang, L.R. Baxter, M.E. Phelps, A general technique for interstudy registration of multifunction and multimodality images, IEEE Trans. Nucl. Sci. 41 (6) (1994) 2850–2855.

[23] G.A. Carpenter, S. Grossberg, A massively parallel architecture for a self-organizing neural pattern recognition machine, Comput. Vision Graphics Image Process. 37 (1987) 54–115.

[24] A. Baraldi, F. Pamiggiani, A neural network for unsupervised categorization of multivalued input patterns: an application to satellite image clustering, IEEE Trans. Geosci. Remote Sensing 33 (2) (1995) 305–316.

[25] G.A. Carpenter, S. Grossberg, D.B. Rosen, Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system, Neural Networks 4 (1991) 759–771.

[26] L. Zadeh, Fuzzy sets, Inform. Control 8 (1965) 338–353.

[27] P.K. Simpson, Fuzzy Min-Max neural networks—Part 2: clustering, IEEE Trans. Fuzzy Systems 1 (1) (1993) 32–45.

[28] A. Baraldi, E. Alpaydm, Simplified ART: a new class of ART algorithms, TR-98-004, International Computer Science Institute, Berkeley CA, 1998.

[29] S.V.N. Vishwahathan, M. Narasimha Murty, Kohonen's SOM with cache, Pattern Recognition 33 (2000) 1927–1929.

[30] T.M. Martinetz, K.J. Schulten, A "neural gas" network learns topologies, in: K. Mäkisara, O. Simula, J. Kangas (Eds.), Artificial Neural Network, North-Holland, Amsterdam, 1991, pp. 397–402.

[31] D.S. Levine, P.A. Penz, ART 1.5—A simplified adaptive resonance network for classifying low-dimensional analog data,in: M. Caudill (Ed.), Proceedings of the International Joint Conference on Neural Networks, 1990, pp.II-639–642.

[32] B. Moore, ART1 and pattern clustering, in: D. Touretzky, G. Hinton, T. Sejnowsky (Eds.), Proceedings of the 1988 connectionist models summer school, Morgan Kaufmann, Los Altros, CA, 1998, pp. 174–185.

[33] B.J. Schachter, L.S. Davis, A. Rosenfeld, Some experiments in image segmentation by clustering of local feature values, Pattern Recognition 11 (1979) 19–28.

[34] T. Kohonen, J. Hynninen, J. Kangas, J. Laaksonen, SOM_pak: the self-organizing map program package, release 3.1, Laboratory of computer and information science, Helsinki University of Technology, Finland, 1995.

[35] T. Kohonen, The self-organizing map, Proc. IEEE 78 (9) (1990) 1464–1480.

[36] R.C. Dubes, A.K. Jain, S.G. Nadabar, C.C. Chen, MRF model-based algorithms for image segmentation, Proceedings of the 10th International Conference on Pattern Recognition, pp. 808–814.

[37] W.A. Yasnoff, J.K. Mui, J.W. Bacus, Error measures for scene segmentation, Pattern Recognition 9 (1977) 217–231.

[38] A. Hoover, G. Jean-Baptiste, X. Jiang, P.J. Flynn, H. Bunke, D.B. Goldgof, K. Bowyer, D.W. Eggert, A. Fitzgibbon, R.B. Fisher, An experimental comparison of range image segmentation algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 18 (7) (1996) 673–689.

[39] X. Yu, T.D. Bui, A. Krzyzak, Robust estimation for range image segmentation and reconstruction, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 530–538.
[40] M. Borsotti, P. Campadelli, R. Schettini, Quantitative evaluation of color image segmentation results, Pattern Recognition Lett. 19 (1998) 741–747.
[41] J. Liu, Y.H. Yang, Multiresolution color image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (7) (1994) 689–700.
[42] P.P. Ohanian, R.C. Dubes, Performance evaluation for four classes of textural features, Pattern Recognition 25 (8) (1992) 819–833.
[43] J.M. du Buf, M. Kardan, M. Spann, Texture feature performance for image segmentation, Pattern Recognition 23 (3/4) (1990) 291–309.
[44] R.W. Conners, C.A. Harlow, A theoretical comparison of texture algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 2 (3) (1980) 204–222.
[45] R.M. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, IEEE Trans. Systems Man Cybern. 3 (6) (1973) 610–621.
[46] J.S. Weszka, C.R. Dyer, A. Rosenfeld, A comparative study of texture measures for terrain classification, IEEE Trans. Systems Man Cybern. 6 (4) (1976) 269–285.

About the Author—LUIGI CINQUE (S′86-M′91-SM′96) received his laurea degree in Physics from the University of Napoli, in 1983. From 1984 to 1990, he was with the Laboratory of Artificial Intelligence (Alenia S.p.A) Presently he is Professor at the Department of Computer Science of the University "La Sapienza" of Rome.
His scientific interests cover image sequences analysis, shape and object recognition, image database, and advanced man–machine interaction. Prof. Cinque is presently an Associate Editor on Pattern Recognition Journal and Pattern Recognition Letters. He is Senior Member of IEEE (Institute of Electrical and Electronics Engineers), member of the ACM (Association for Computing Machinery), abd member of IAPR (International Association for Pattern Recognition). He has been in the program committee of many international conferences in the field of imaging techonology, and he is the author or co-author of over 120 scientific publications in international journals and conference proceedings. He serves as the reviewer for several interanational journals and international conferences.

About the Author—GIAN LUCA FORESTI was born in Savona (Italy) in 1965. He received the laurea degree *cum Laude* in Electronic Engineering and the Ph.D. degree in Computer Science from University of Genoa, Italy, in 1990 and in 1994, respectively. In 1994, he was visiting Professor at University of Trento, Italy. Since 2000, he is Professor of Computer Science at the Department of Mathematics and Computer Science (DIMI), University of Udine. His main interests involve computer vision and image processing, multisensor data fusion, artificial neural networks, and pattern recognition.
Prof. Foresti is author or co-author of more than 150 papers published in International Journals and Refereed International Conferences. He was general co-chair, chairman and member of Technical Committees at several conferences. He has contributed in 7 books in his area of interest and he is co-author of the books: "Multimedia Systems for Visual-based Surveillance" (Kluwer, 2000) "Multisensor Surveillance Systems: Tha Fusion Perspective" (Kluwer, 2003). He has been Guest Editor of the Special Issue of Real Time Imaging Journal on "Video Communications, Processing and Understanding for Third Generation Surveillance Systems". Currently, he is Guest Editor of a Special Issue of the *IEEE Transactions on Systems, Man and Cybernetics* on "Ambient Intelligence". He serves as a reviewer for several international journals, and for the European Union in different research programs (MAST III, Long Term Research, Brite-CRAFT). He is Senior member of IEEE and member of IAPR.

About the Author—LUCA LOMBARDI was born in Cremona, Italy, in 1961. He received the Laurea (cum laude) in Electronics Engineering from the University of Pavia, 1986 and the Ph.D. in 1990. From 1991 he held a position as Assistant Professor at the Faculty of Engineering of the University of Pavia. From 2000, he held a position as Associated Professor at the Faculty of Engineering of the University of Pavia. His fields of interest concern image processing, machine learning and architectures for image processing. He is author or co-author of more than 50 journal of Conference papers and book chapters. He is member of the IAPR (International Association for Pattern Recognition) and of IEEE (Institute of Electrical and Electronic Engineers).