Unsupervised Image Segmentation with Neural Networks

J. Meuleman and C. van Kaam
Wageningen Agricultural University, Department of Agricultural, Environmental and
Systems Technology, Bomenweg 4, 6703 HD Wageningen
The Netherlands
e-mail: Jan.Meuleman@User.AenF.WAU.NL

Keywords: Neural networks, unsupervised learning, recursive partitioning, image segmentation

Abstract

The segmentation of colour images (RGB), distinguishing clusters of image points, representing for example background, leaves and flowers, is performed in a multi-dimensional environment. Considering a two dimensional environment, clusters can be divided by lines. In a three dimensional environment by planes and in an n-dimensional environment by n-1 dimensional structures. Starting with a complete data set the first neural network, represents an n-1 dimensional structure to divide the data set into two subsets. Each subset is once more divided by an additional neural network: recursive partitioning. This results in a tree structure with a neural network in each branching point. Partitioning stops as soon as a partitioning criterium cannot be fulfilled. After the unsupervised training the neural system can be used for the segmentation of images.

INTRODUCTION

Closely related to the EC-funded AIR-project "Objective Plant Quality Measurement by Digital Image Processing", fundamental research is carried out to develop smart segmentation procedures for plant images. After having reported successful image segmentation of plant images, using supervised trained neural networks (Meuleman, 1996), two approaches are being pursued for the development of unsupervised segmentation procedures:

- procedures, focussing on the centres of the clusters to be distinguished;
- procedures, focussing on the separation of clusters by discovering areas with a relatively low occupation of image points in a multi-dimensional environment.

In earlier research (Meuleman, 1996) the segmentation of colour images of the Saint Paulia, distinguishing leaves, flowers, stamen, pot and background, was performed in an 18 dimensional environment, representing the R, G and B values of each image point together with additional information from the direct environment to achieve separability of the clusters to be distinguished. Compared with segmentation, based on the 3 dimensional RGB values of image points, this segmentation procedure was superior. However, the neural network which performs the segmentation, has to be trained: the quality of the segmentation procedure depends on the quality of the training. Furthermore the composition of a representative training and test set was a difficult task. To eliminate the disadvantages of the supervised neural network and to facilitate the use of neural networks for a good segmentation of plant images, unsupervised methods have been investigated.

MATERIALS, METHODS, EXPERIMENTS

In the experiments the same data as in the earlier research is used (Meuleman, 1996). From top view images of SaintPaulia 'Heidrun' plants with white flowers, two data sets of image points have been composed:

Data set 1 consists of 100 pixels, representing leaves, 100 pixels representing the (white) flowers, 50 pixels representing the stamen, 100 pixels representing stems and 100 pixels representing the blue background. All pixels are selected in such a way that boundary pixels are avoided.

Data set 2 consists of data set 1 on to which pixels from the boundaries of the objects are added, resulting in 200 pixels, representing leaves, 200 pixels representing the (white) flowers, 100 pixels representing the stamen, 200 pixels representing stems and 200 pixels representing the blue background.

Proc. Sensors in Hort. III
Eds. I Shmulevich et al.
Acta Hort. 562, ISHS 2001

The five clusters, representing leaves, flowers, stamen, stems and background could be distinguished in an 18 dimensional environment. The 18 dimensions are:

* r, g and b values of the (central) pixel.
* R, G and B values of the (central) pixel.
* R, G and B values of 4 additional pixels. These additional pixels are the corner pixels of the 5x5 environment of the central pixel.

All colours are measured on 256 levels: r=red/256; g=green/256; b=blue/256. Furthermore: R=r/(r+g+b); G=g/(r+g+b) and B=b/(r+g+b).

It means that besides the colour values of the pixel (to classify), relative colour information from the environment is also used.

Cluster Building, Based on the Discovery of Centres of Clusters

A three layer, fully connected neural network is used with one input layer of 18 neurons, 10 hidden neurons. At both the input and the hidden layer a threshold neuron is added. To activate the network a uni polar sigmoïd function is used. The output layer consists of a variable number of output neurons, corresponding with the number of clusters discovered. At the start of the procedure the first cluster has to be discovered (1 ouput neuron). At random one input pattern is assigned to cluster 1 and the neural network is trained. After training all patterns are propagated through the network and the pixel with the best similarity is added to the group of input patterns belonging to cluster 1. The system is trained again and the next most similar input pattern is added. When all similar pixels are added to cluster 1, the network is enlarged with one output neuron, representing the next cluster to discover. After this enlargement the iterative procedure restarts with two pixels: one per cluster. The selected pixels are the pixels with the highest similarity and the highest dissimilarity with cluster 1. After adjacent training sessions of the network one pixel (with the highest similarity with cluster 1 or 2) is added, and so on, until all clusters are discovered.

The error function consists of two components: a similarity and a dissimilarity contribution. Similarity and dissimilarity are defined as:

$$S_{n,m} = (1 - x_{n,m})^2$$
$$D_{n,m} = x_{n,m}^2$$

with: $S_{n,m}$ = degree of similarity of input pattern n with cluster m;
$D_{n,m}$ = degree of dissimilarity of input pattern n with cluster m;
$x_{n,m}$ = score of input pattern n on output neuron m.

If a pixels is classified as belonging to cluster n, the similarity error with cluster n is taken into account, together with the sum of the dissimilarity errors with all other clusters: $\Sigma_m (S_{n,m} + D_{n,m})$. Unclassified pixels do not have a similarity contribution.

Total similarity error = $\Sigma\ S_{n,m}$ of all patterns n belonging to cluster m;
Total dissimilarity error= $\Sigma\ D_{n,m}$ of all patterns n not belonging to cluster m.
As error function for non-linear minimisation is used:
Total error = $\Sigma_{n1} \Sigma_m(S_{n1,m} + D_{n1,m})/n_1 + \Sigma_{n2} \Sigma_m D_{n2,m}/n_2$,
with: n_1 = number of classified patterns;
n_2 = number of not classified pixels.

During the procedure the ratio between the mean similarity contribution of the classified group of pixels and the mean dissimilarity contribution of the unclassified pixels is administrated and a typical example of the course of this ratio is shown in figure 1. It can be seen that after adding a total group of 100 pixels, all belonging to for example leaves, the ratio increases. It means that also pixels, not belonging to the cluster "leaves" are included. After adding some pixels the ratio decreases sometimes. It means that a group of pixels, not belonging to leaves, but positioned close to each other in the n-dimensional space, are being incorporated in the cluster "leaves". By setting a threshold for the ratio, the first pixels of the next cluster are discovered. After adding one output neuron to the neural system and the assignment of one input pattern per cluster, as described above, the total procedure restarts until all pixels are classified without the need to add clusters.

Cluster Building, Based on the Discovery of Areas with Low Occupation of Pixels in Multi Dimensional Environment.

Given the total data set A, a neural network (NN1) is being built to divide this set in 2 subsets, B and C. Additional neural networks NN2 and NN3 try to divide respectively B and C in the subsets D, E and F, G, and so on. By means of recursive partitioning the clusters to be distinquished can be built as shown in figure 2. Each neural network in the tree divides a (sub)set into two exact two (other) subsets. This type of partitioning will be dealt with as a number of standardised branching problems: all input variables of the branching neural network are individually scaled by substracting their mean value and divided by their standard deviation. By choosing the configuration of all neural networks identically, the problem is reduced to a standardised partitioning problem with only one difference: the number of patterns. Once the total tree is built, the tree can be used to classify new patterns. Only two questions are not solved yet: How to divide a set of patterns into two subsets and how to to evaluate the partitioning.

1. Partitioning a Set of Input Patterns into Two Subsets A set of input patterns can be represented as points in a multi-dimensional space, built up by the input variables. If the set of input patterns consists of two or more distinguishable clusters, it is expected that each cluster has its own centre in the multi-dimensional space. It is also expected that all input patterns, belonging to a certain cluster, are more or less concentrated around their cluster centre. In that case it is also expected that the point-density in the multi-dimensional space between the clusters is relatively low. This reduces the problem as follows: Is it possible to build a neural network, representing an n-1 dimensional structure, and this structure can be placed in areas with low occupation of patterns and in such a way that at each side of the structure a subset of patterns is located. Given an arbitrarily placed n-1 dimensional structure a suitable error function can be built to move the structure towards the areas with the lowest densities. If the contribution of patterns to this error function is higher when its distance from the structure is shorter than the n-1 dimensional structure will indeed follow areas with low point density. Figure 3 illustrates the error contribution function dependant on the distance of the structure from the error function used:

$$errs_p = e^{-distance_p^2} \qquad (1)$$

with: $errs_p$ = the squared error contribution of pattern p;
$distance_p$ = the shortest distance between p and the n-1 dimensional structure.

Besides an error function a network configuration has to be chosen as well. A very simple configuration is illustrated in figure 4. It consists of a fully connected two layer network. The dimension of the input layer is equal to the number of input variables. The output layer has only one neuron, with threshold and (1) is used as activation function. Figure 4 illustrates the case of 3 input variables. The output of the neural network is the error contribution of the propagated pattern.

In general:

$$errs_p = e^{-(\Sigma i\alpha i.xi,p)^2} \qquad (2)$$

with: $errs_p$ = error contribution of pattern p;
α_i = weights of input signals;
$x_{i,p}$ = values of input variables of pattern p

For three input variables the distance to the structure of pattern p is:

$$distance_p=\alpha_0+\alpha_1.x_{1,p}+\alpha_2.x_{2,p}+\alpha_3.x_{3,p} \qquad (3)$$

Error minimisation results in two effects: The structure will be moved to a place with a lower occupation of patterns, but also the weights (α_i) will be reduced. The move of the structure to a less dense area is described in (4):

$$\alpha'_i = \alpha_i + L.d\alpha \qquad (4)$$

with: α'_i = new unscaled weight vector;
α_i = old weight vector;
L = stepsize;

$d\alpha$ = direction towards minimum error.

To avoid the minimisation procedure resulting in infinite ultimate weights (infinite weights produce error contributions equal to zero), the new weight vector is rescaled as follows:

$$\alpha''_i = R.\alpha'_i.\sqrt{}(\Sigma\alpha_i^2)/\sqrt{}(\Sigma\alpha_i'^2) \qquad (5)$$

with: R = reduction factor

By choosing relatively high values for the initial weights and a reduction factor (R) per iteration, the flatness of the error function can be controlled during the optimisation process. With these two instruments a good global partitioning can be achieved, avoiding local minima.

2. The Evaluation of the Partitioning After placing the structure, whether the partitioned (sub)set consisted of more clusters or just one has to be evaluated. In the latter case the branching process stops for this particular branch. The evaluation is done by means of comparing sums of squares before and after partitioning. Because of the standardisation before partitioning the mean and standard deviation of the set is known: The total sum of squares per input variable is equal to the number of patterns, say n. For the total set, with, for example, v input variables, the total sum of squares is equal to nv. After partitioning the mean value, $X_{1,i}$ and $X_{2,i}$, and the squared differences with regard to the mean value, $\Sigma_1(x_{1,i}-X_{1,i})^2$ and $\Sigma_2(x_{2,i}-X_{2,i})^2$, is calculated per variable. If the (sub)set of n patterns is partitioned in subset 1, containing n_1 patterns and subset 2, containing n_2 patterns and the number of input variables is N, than the squared ratio is evaluated:

$$\text{ratio} = \frac{\Sigma_1^N(n_1.X_{1,i}^2 + n_2.X_{2,i}^2)}{\Sigma_{i=1}^N(\Sigma_{j=1}^{n1}(x_{1,i,j}-X_{1,i})^2) + \Sigma_{i=1}^N(\Sigma_{j=1}^{n2}(x_{2,i,j}-X_{2,i})^2)} \qquad (6)$$

Ratio (6) is calculated after each iteration during error minimisation. The best partitioning corresponds with the highest ratio. If the ratio becomes too low, the partitioning proces is stopped.

Future Developments

The potentiality of both methods for automatic image segmentation needs further investigation. Regarding the centre approach, the research goal was to develop methods. The method described is not optimised with respect to the need of calculation power. Further research should be primarily concentrated on the development of standards for process evaluation and the study of its behaviour during the clustering process. Secundary goal is speeding up the procedure by reducing the need of calculation power. With regard to the partitioning method, further research should be primarily concentrated on the behaviour of the ratio statistic, especially in the case that a set, consisting of more clusters, has to be partitioned in subsets, each containing more than one cluster.

RESULTS AND FUTURE RESEARCH PLANS

Figure 5 shows the results of the segmentation process, using the clustering approach on the 450 pattern data set: the set without patterns in the transition area's of the distinguishable clusters. Errorfree segmentation would result in:

cluster 1: pixel 1 - 100; cluster 2: pixel 101 - 200; cluster 3: pixel 201 - 250;
cluster 4: pixel 251 - 350; cluster 5: pixel 351 - 450;

Only one pixel is placed in a wrong group: a stamen pattern is classified as a leaf.

Using the same data set and the partitioning approach this last pattern could also be classified as stamen. It seems that this "problem pattern" is a real stamen pattern and not an error in the data set, as suggested in an earlier research (Meuleman, 1996).

More study has to be done if patterns from the transition areas are added to the data set. A reasonable segmentation can be found, but current methods are not stable enough.

To test the partitioning approach a huge set of top view images of the Begonia *Elatior* was available. Figure 6 shows the results of the partitioning of a set of images. The colours are assigned to the clusters in order of discovery of the clusters.

ACKNOWLEDGEMENTS

The authors thank the EC for the financial support within the scope of the mentioned AIR project. Without granting the research proposal, this fundamental research would have been impossible. A second acknowledgement regards student M. Sanders, who spent a tremendous number of hours at programming, improving and evaluating the clustering method within the scope of his Msc study.

Literature Cited

Meuleman, J. and Besselink, C.T.M. 1996. Image Segmentation with Neural Networks. Contribution at the EurAgEng congress, Madrid, 23-26 September.

Figures

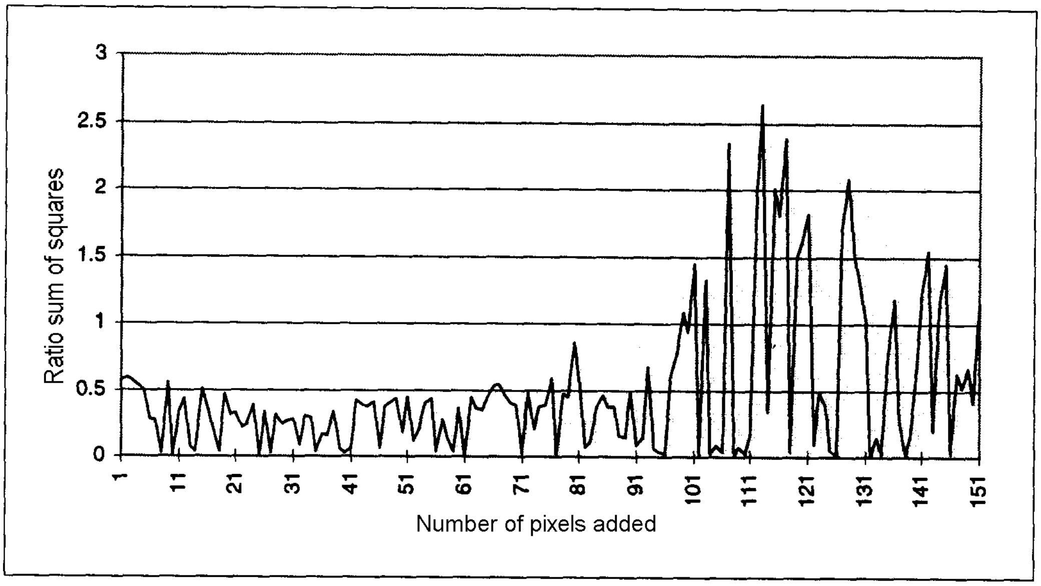


Fig. 1. Course of the ratio after 150 additions to a cluster of 100 pixels.

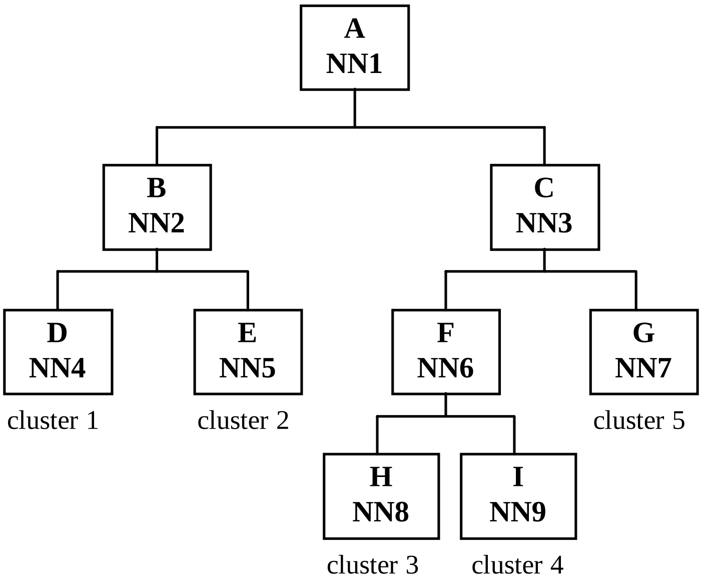


Fig. 2. A tree of neural networks for recursive partitioning

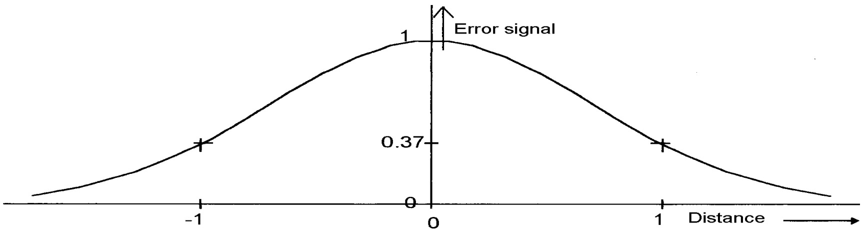


Fig. 3. The error contribution of e $^{-\text{distance}_p^2}$

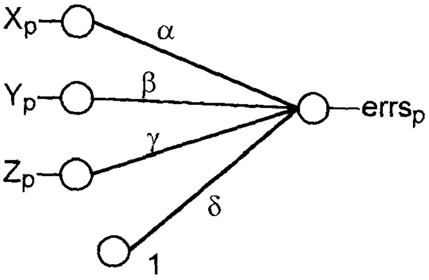


Fig. 4. The propagation of pattern p by the neural network.

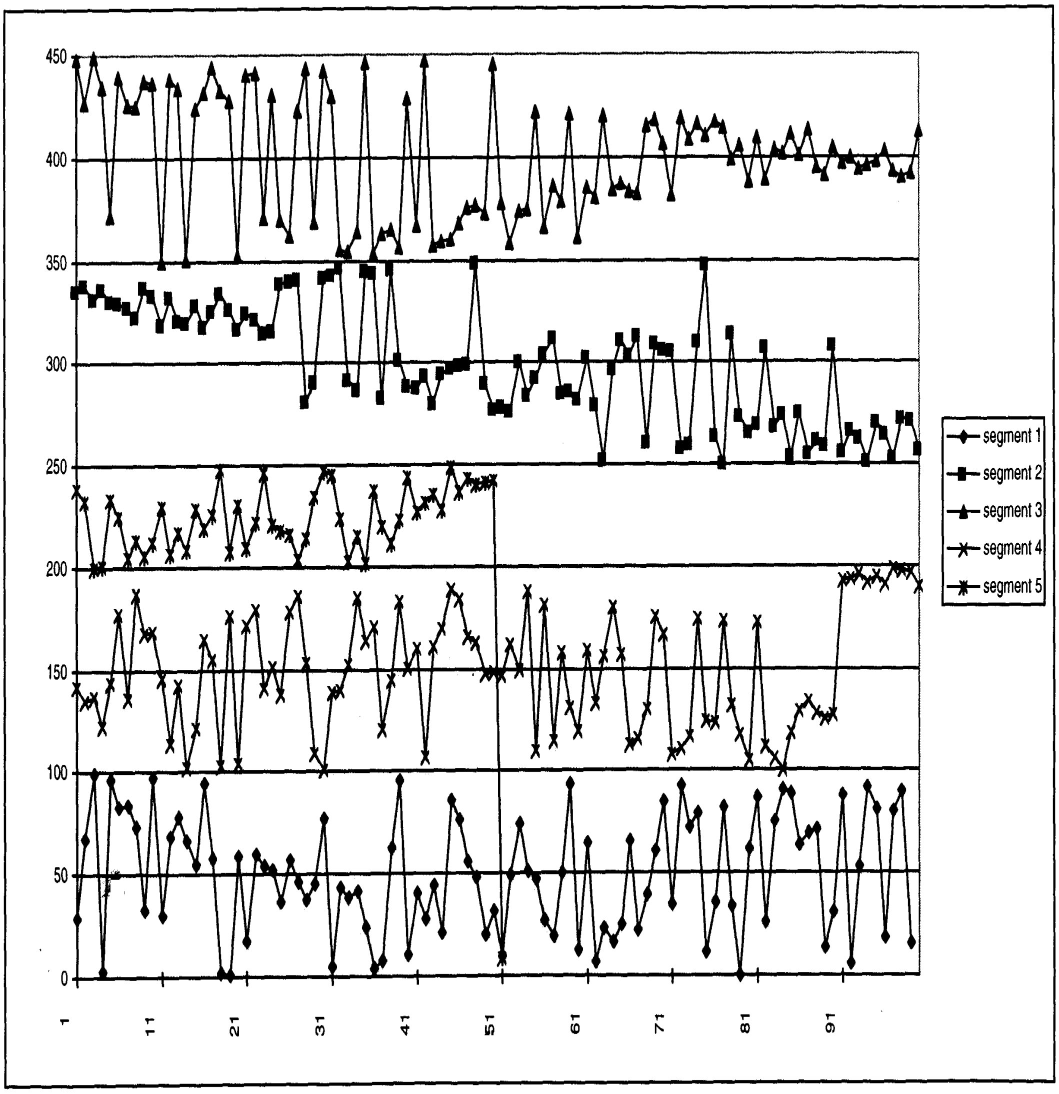
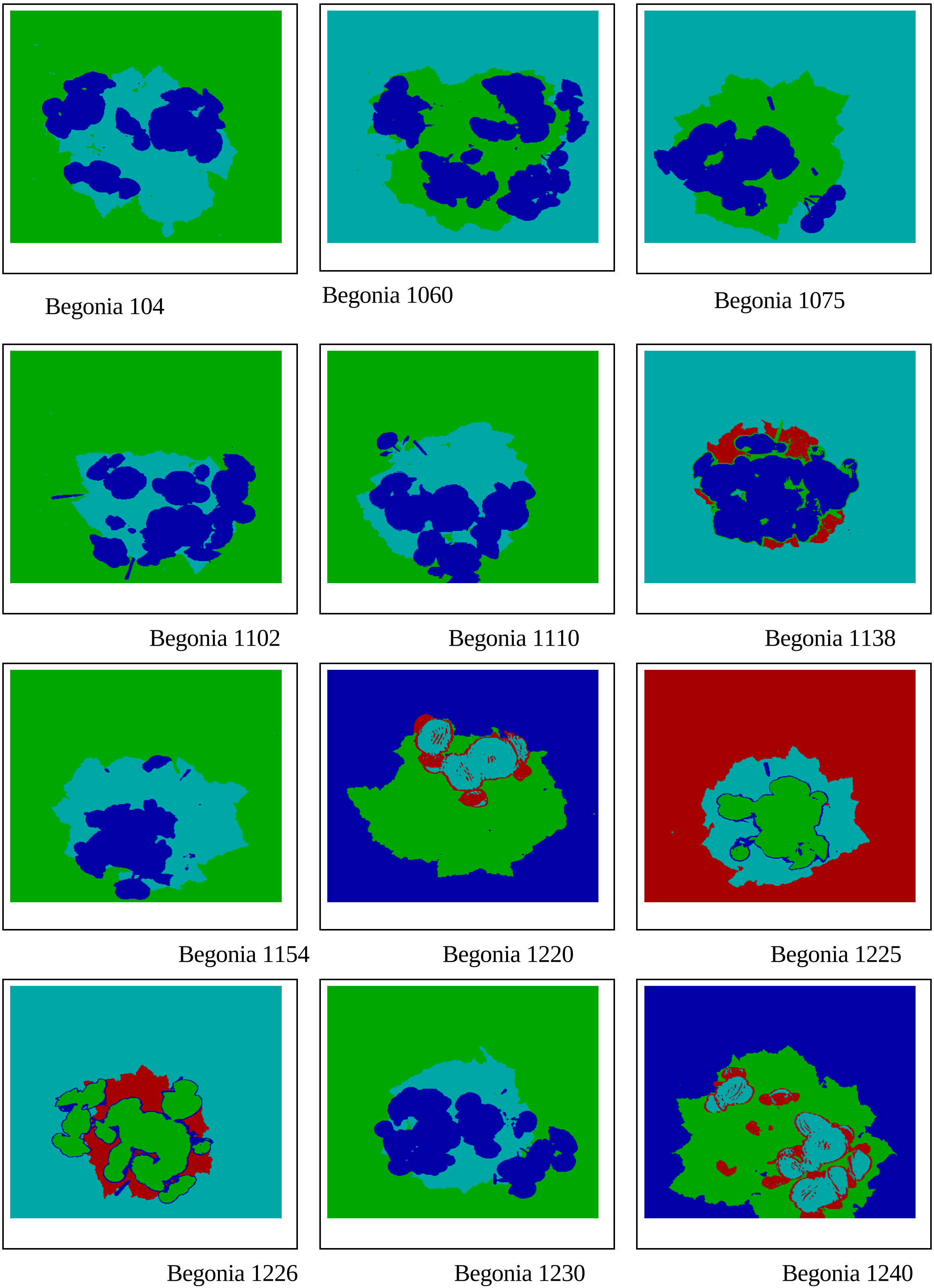


Fig. 5. Result of the clustering proces on the 450 pattern data set.



Begonia 104

Begonia 1060

Begonia 1075

Begonia 1102

Begonia 1110

Begonia 1138

Begonia 1154

Begonia 1220

Begonia 1225

Begonia 1226

Begonia 1230

Begonia 1240

Fig. 6. A set of 12 Begonia plants after unsupervised segmentation

.