An Artificial Immune System Based Visual Analysis Model and Its Real-Time Terrain Surveillance Application

György Cserey[1,2], Wolfgang Porod[1], and Tamás Roska[2]

[1] Department of Electrical Engineering, University of Notre Dame,
275 Fitzpatrick Hall, Notre Dame Indiana 46556, USA
{gcserey,porod}@nd.edu
[2] Faculty of Information Technology, Pázmány University, Budapest and
Computer and Automation Research Institute
of the Hungarian Academy of Sciences,
Kende utca 13-17, Budapest, H-1111, Hungary
roska@sztaki.hu

Abstract. We present a real-time visual analysis system for surveillance applications based on an Artificial Immune System inspired framework [10] that can reliably detect unknown patterns in input image sequences. The system converts gray-scale or color images to binary with statistical 3x3 sub-pattern analysis based on an AIS algorithm, which make use of the standard AIS modules. Our system is implemented on specialized hardware (the Cellular Nonlinear Network (CNN) Universal Machine). Results from tests in a 3D virtual world with different terrain textures are reported to demonstrate that the system can detect unknown patterns and dynamical changes in image sequences. Applications of the system include in particular explorer systems for terrain surveillance.

1 Introduction

The detection of important objects, patterns or dynamic changes in real time is a difficult problem for machine visual systems (such as those used for surveillance, robots, etc.), where there is no human decision-maker present, since the image database becomes too big to handle efficiently. The problem becomes intractable on autonomous robots with limited computational resources, where standard imaging algorithms are not applicable because of their high computational demands. Yet, for many surveillance tasks, such as those performed by video surveillance systems to record strange and important events at a bank or in the street, being able to send an event detection message as fast as possible is essential.

Recently, many pattern recognition approaches have been proposed for supervision systems to detect or classify patterns based on different theoretical models and processing methods [16,17,18,19]. While these systems are capable of classifying scenes based on motion information of trajectories or labelling

G. Nicosia et al. (Eds.): ICARIS 2004, LNCS 3239, pp. 250–262, 2004.

of events and interactions, none of them is capable of carrying out all of the needed tasks, like preprocessing, feature extraction, learning and recognition in real time. *Artificial Immune Systems* (AIS) mimic the human immune system that has refined capabilities and methodologies to build efficient algorithms that solve engineering problems such as those mentioned above. Moreover, our immune system possesses important properties (eg. diversity, noise and fault tolerance, learning and memory and self-organization) which give it an advantage compared to other standard methods.

In this paper, based on the original CNN-UM simulation of immune response inspired spacial-temporal algorithmic framework, we present an AIS-based image terrain pattern analysis approach that is intended to solve the real time dynamical pattern recognition and detection problem in 2D image sequences using techniques analogous to natural immune systems. Humans monitor their environment with eyes and process its input image flow through their retina and relay the result to the brain for further evaluation. In our experiments, we utilized a parallel array processor, the *Cellular Nonlinear Network* (CNN) universal machine [6,7,8,9] for visual processing. This fast processor runs not only the sensory pre-processing algorithm, but the learning and recognition methods too, which in cooperation with digital algorithms allows for fast applications.

In Fig. 1, we would like to show the basic idea. This is a simple example where the detection is based on the tail type of the objects (airplanes). During initialization our system was taught not to detect planes with the first type of tail (Fig. 1a). Then, during the recognition phase, the planes which have the same tail, will not be detected, but any other type of tail, like the second set of tails (Fig. 1b, the unfamiliar ones), will be detected.

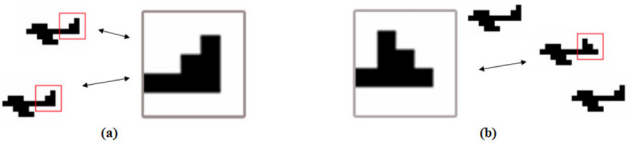

Fig. 1. After the initialization phase (a), known input patterns keep the system tolerant and during the recognition phase (b) unknown objects can cause detection, if they can be differentiated from the unimportant noise.

The organization of this paper is as follows: after a quick review of recent AIS approaches for pattern recognition and a brief introduction to the applied methods of AIS in Section 2, a detailed overview of our proposed system is presented focusing mainly on the AIS methods in Section 3. We discuss the properties, the mathematical representation and the algorithm. Section 4 explains a terrain surveillance application with results of the experiments and observations. In Section 5 the new aspects and time measurements of this research are presented. Finally in Section 6, the results are summarized, conclusions drawn, and plans for future work given.

2 Background on Pattern Recognition Using Artificial Immune Systems

2.1 Human and Artificial Immune System

The human immune system is an elaborate system with complex mechanisms for defense against antigens. Antigens are all materials that can be specifically recognized as an offensive (pathogen) or non-offensive (non-pathogen), and specifically responded to by the immune system [13]. The basic components of this biological system are the B and T lymphocytes, they intervene during the adaptive immune response and are accountable for the detection and elimination of offensive materials. From the point of view of pattern recognition, a very important characteristic of T and B-cells is that they have surface receptor molecules for recognizing antigens.

Some of important processes and properties of the immune system from our model's point of view are briefly summarized in following:

- *Bone marrow* is the major site of production of blood cell types including lymphocytes. Special environment is also provided for antigen-independent differentiation of B-cells by this soft issue.
- *Thymus* is a lymphoid organ, whose environment is provided for antigen-independent differentiation of T-cells. Immature T-cells migrate from the bone marrow to the thymus, some of them differentiate into immunocompetent cells by positive selection and those who have strong recognition of self-peptides are purged out by the negative selection method.
- *Positive and negative selection*: All T-cells bind to the antigens through self-peptide complexes (MHC, see [13]). Only those T-cells, which are capable of recognizing the self-peptide complexes, are stimulated for maturation by the positive selection, and the process of negative selection purges the autoreactive T-cells from the repertoire.
- *Clonal selection* is one of the basic properties of an adaptive immune response where only those cells proliferate and differentiate into effector cells which are capable of recognizing an antigen stimulus.
- *Immune memory*: After the first infection, high affinity, long living lymphocytes are stored and persisted in a resting state to give stronger response after the next infections.

"Artificial immune systems (AIS) are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving" [2]. For the AIS, immune cells and molecules are represented by data items which take apart in some general-purpose algorithm that models specific interactions and aspects of the immune systems. The repertoires of the data items are usually generated by bone marrow models. The models of thymus are used to select repertoires capable of performing non-self/self-peptide discrimination. In the theory of AIS [1,2], antigens, lymphocytes and any molecules have a generalized shape m in shape-space S. This generalized

shape can be represented as an attribute string - set of binary or other coordinates - of length L. Therefore any molecule string m can be regarded as a point in an L-dimensional shape-space. The interaction of an antigen and a peptide is assessed via a common (Euclidean, Manhattan, Hamming, etc.) D distance measure, which is also called affinity measure, between their proportional strings. Usually a relation is defined between the distance D and the recognition region V_ε proportionally to the recognition threshold or cross-reactivity threshold ε. If the D distance measure between data items is larger than ε, then a successful recognition is assumed between the items.

2.2 Applied Pattern Recognition Methods with AIS

The field of pattern recognition is mainly focused on building systems which are able to identify patterns in given measurement or observation data. Subdisciplines are feature extraction, error estimation, classification, syntactical pattern recognition. The main applications are classification, image processing, character or handwriting recognition, speech analysis, human person identification, diagnosis and industrial applications.

To overview the main pattern recognition areas using AIS, we can give some fields based on the literature of [2,4], where general AIS approaches were to be applied. Each chemical reaction, which maps a set of reactants into a set of products, can be identified by a spectrum, see [20]. We worked with a binary Hamming shape-space, and our AIS was combined with a genetic algorithm to recognize specific spectrums. We proposed an affinity function where the weight of each bit was chosen according to a spectrum characteristic. Another area is the surveillance of infectious diseases [21]. Their AIS was developed for analysis and understanding at the dynamics of the plague. Pattern recognition applied in other different medical areas. A general data classification system was designed for medical data analysis [22]. In the area of data mining, an immunological algorithm was proposed [23] to discover rules classifying samples belonging to small disjuncts, which correspond to rules covering a small number of examples. In [4], a genetic algorithm and computational implementation of AIS was proposed to solve a color image recognition and classification task.

In the following, we will present our approach to image pattern recognition that was designed to solve efficient feature extraction, clonal selection, mutation processes and overcome real-time speed limitations.

3 Model and Algorithm

3.1 Comparison of the Methods and Parameters

Our model defines the antigens and T-lymphocytes as two data items with different characteristics and goals. These data items are elements of a shape-space, which can be represented by *nxn* sized binary (black and white) matrixes. Colors can be coded with *1* (black) and *-1* (white) numbers. Each antigen is usually a

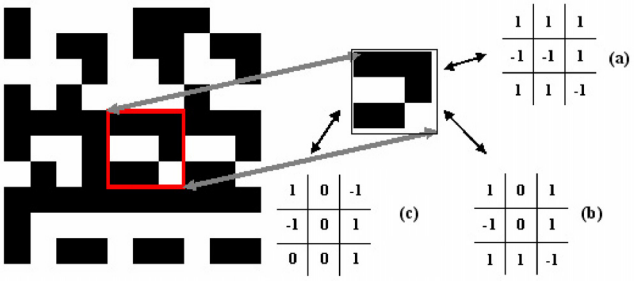

Fig. 2. If in the position of *1*s black pixels and in the position of *-1*s white pixels are found, the matching is successful, while pixel color in the position of the *0*s is indifferent. Therefore (a) and (b) give successful matching while (c) is unsuccessful.

3x3 or *5x5* subpattern of a binary picture which is subtracted from the input image flow by a special feature extraction method. These patterns (2D-strings) can be recognized by our T-lymphocytes, called match-templates [14]. They are usually *3x3* or *5x5* matrixes and contain *1*, *-1* and *0* numbers. During the interaction between templates and patterns, if in the position of *1*s black pixels and in the position of *-1*s white pixels are found, the matching is successful, while pixel color in the position of the *0*s is indifferent. An example can be seen in Fig. 2.

We used a simple bone marrow model to create an initial template set [11]. The elements of this matrix set (*1*, *-1* and *0*) were generated randomly with equivalent probability. This method was applied in our application but also there are other strategies to initialize the starting template set. One can be to predetermine the number of *0* or otherwise 'don't care' elements. Other important attributes, like age (length of life), efficiency (number of successful matching) and specificity (number of *0* elements) were also initialized during our method.

Basically our algorithm has an initialization and recognition phase, where the former models the function of thymus. Our thymus model is based on a template runner, which tested all the elements of the initial template set on a learning input flow using a negative selection algorithm. The templates, which were able to match more than a given number of sub-patterns of the initial input flow, were selected out. In the recognition phase we use the same template runner module, but for a different purpose. If any members of the template set are able to match the actual pattern with a given threshold, a detection message is generated. The steps of the algorithm can be followed in Fig. 3, continuing our "fighter tail" example. For clonal selection, the mutation is provided through a loop-back from the result of the template-runner to the template-set.

In our model the S shape-space has 9 or 25 dimensions, because the subpattern matrixes can be represented by 25 or 9 long binary vectors and the templates correspond to 25 or 9 long vectors (coordinates can be -1, 1, or 0). The distance measure between an antigen ($Ab = \langle Ab_1, Ab_2, ...Ab_L \rangle$) and a template $Ag = \langle Ag_1, Ag_2, ...Ag_L \rangle$ is

$$D = \sum_{i=1}^{L} \delta_i, \ where \ \delta_i \begin{cases} 1 \ if \ Ab_i = Ag_i \ or \ Ab_i = 0 \\ 0 \ if \ Ab_i \neq Ag_i \end{cases} \quad (1)$$

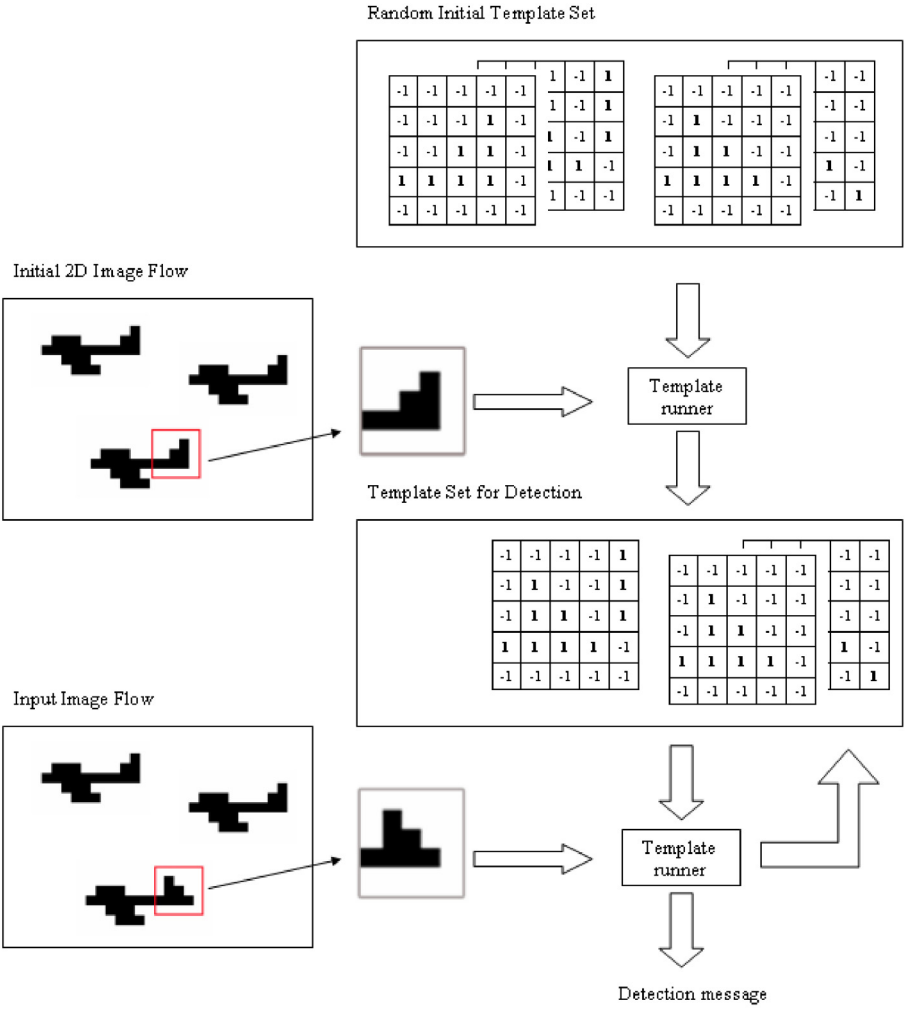

Fig. 3. AIS based algorithm framework for a simple example. The random initial template set is generated by the bone marrow model. After negative selection of the initial 2D image flow, the template set is prepared for detection. In the recognition phase, in case of any members of the template set are able to match the actual pattern with a given threshold, a detection message is generated. The influence of mutation is provided by a loop-back from the template-runner to the template-set.

Our match template class does the recognition if and only if the D distance is L, where L equals to the dimension of the actual S shape-space. Contrary to common AIS, where the molecules usually are represented by similar vectors, the sub-patterns and template vectors generally differ in our model. There are "don't care" elements in the templates, whose position are fixed within their vectors. Therefore, we could not give a definition of affinity as other AISs have. If a match-

template has d "don't care" elements, it can detect 2^d different sub-patterns. The more 'don't care' elements it has, the more different sub-patterns are, which are detected. Therefore, the affinity of a template can be characterized by the number of the 'don't care' elements. This affinity is called template affinity α. This affinity has a similar effect to the usual affinity or cross-reactivity threshold in AIS. A sub-pattern can be matched successfully by $2^L = \sum_{\alpha=0}^{L} \binom{L}{\alpha}$ different match-templates, where α is the affinity, defined formerly. The maximum number of sub-patterns that can be recognized by a template set is $\sum 2^{\alpha_i}$, where the α_i is the template affinity of ith template of the template set. During the recognition phase a successful template is cloned changing one of its *0* elements to *-1* or *1*, therefore the clones are more specific. The lifetime of a template is extended if it is successful and its specificity reaches a given threshold. Parallel to the recognition process, the actual template repertoire can be expanded with new templates using negative selection. It is also beneficial to refresh the template repertoire replacing the unsuccessful old templates with low template affinity ones.

The parallel notions between the immune system and our AIS visual analyzer model can be found in Table 1.

Table 1. Mapping between the immune system and the our AIS visual analyzer model.

Immune System	AIS Visual Analyzer Model
Antigen	Sub-patterns in pictures of 2D image flows
T-lymphocyte	Template matrixes
Memory cell	Specialized template with several recognition
Recognition	Template matching between the matrixes
Life of an organism	Number of interactions
Affinity measure	Number of 'don't care' elements

3.2 Feature Extraction Method

Because the input frames of a visual 2D image sequence can be gray-scale or color, we need a conversion method which transforms the input image into a binary one. Let us suppose that the input image is gray-scale and the size of the templates is 3x3. In the algorithm, the input image is threshold nine times at different levels. Each element in the 3x3 binary pattern is defined by a given threshold result. Practically, each threshold result is AND-ed to its mask to select its position in the 3x3 sub-pattern. The masks define disjunct sets on the picture. All these masked binary images are logically OR-ed to give the output result. The detailed algorithm can be found in [12]. If the input image is gray-scale the algorithm sub-samples the original input and sets the binary value of the sampled pixels into the pixels of the output binary image. If the input is color, we can combine the different color channels (red, green, blue) as above,

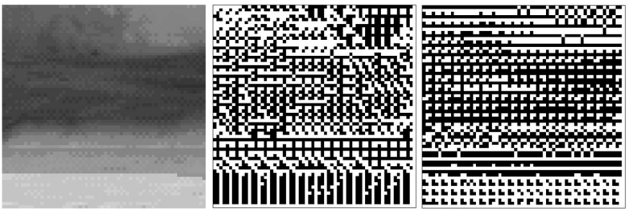

Fig. 4. Conversion results of different type input images. The original version of the left image was colorful. In the middle there is result of the gray-scale to binary conversion. In the right side, the result of the color to binary conversion can be seen.

but each input channel defines only three binary pixels in the pattern. Sample conversion results can be seen in Fig. 4.

3.3 Theoretical Aspects

In this section, we give an approximation based probability analysis of the needed computational power, estimating the size of the template set. As we know, in the case of the *5x5* sized matrixes, the number of different patterns is $2^{25} =\sim 33$ million. The probability that a random template contains 0 don't care elements, or in other words, that it can match only one pattern, is

$$\frac{2^{25}}{3^{25}} \tag{2}$$

The probability that a random template can match exactly 2^k patterns is

$$\frac{\binom{25}{k} 2^{25-k}}{3^{25}} \tag{3}$$

If we summarize this formula by multiplying with the appropriate values we can get the mean value:

$$\sum_{k=0}^{25} \frac{2^k \binom{25}{k} 2^{25-k}}{3^{25}} = \frac{2^{25}}{3^{25}} \sum_{k=0}^{25} \binom{25}{k} = \frac{4^{25}}{3^{25}} \cong 1328.8 \tag{4}$$

This is the mean value of how many patterns will be covered by a template. Therefore, the theoretical value of the size of the template set is around 25000. If we use 3x3 sized templates, based on a similar proof the size of the template set should be at least around 39. It is not proved that this set will cover all of the same sized sub-patterns, but it gives a good order of magnitude. It must be noted, that any sub-patterns can be recognized by special decomposition of 3x3 templates [15]. The templates can overlap each other without inconsistency, which means that -1 and 1 are not allowed in the same overlapped position. Only one of the overlapped values can be other than 0. An example of overlapping decomposition can be seen in Fig. 5. In this case where the pattern contains 14 pixels the size of the template set should be around 292.

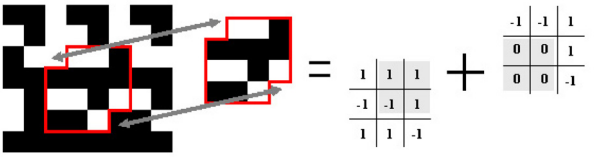

Fig. 5. A decomposition of the bordered pattern by two 3x3 templates. Note that the overlapping area is darker and contains "don't care" elements.

4 Application

Our development focused on a real time application that is able to detect unknown objects, patterns and geological formations based on their textures. It can be used in visual systems where autonomous surveillance is needed or where there is no human presence. It is a helpful additional property of existing surveillance systems subject to unexpected occasions and give detection warnings. For example, it could be a useful complementary function on Mars rovers because due to the long distances real time remote control is unfeasible.

Our test application was implemented in the Aladdin Pro environment using a cellular nonlinear network chip, called Ace4k [8,9]. The input images were generated in a 3D virtual environment by a Pentium IV. personal computer. The relative movement of the camera, direction, camera orientation speed and altitude was controllable manually. A screen-shot is presented in Fig. 6. The size of the input image was 64x64 because our hardware can process images of this size paralelly. Therefore, we cut out a part of the original input for processing. This sub-image was converted to binary patterns by the above presented feature extraction methods.

The template runner worked with a 3x3, randomly generated template set, whose size was between 100 and 500. A texture or image was detected by a template if the number of the successful matching of the template with the sub-patterns was more than the given parameter value.

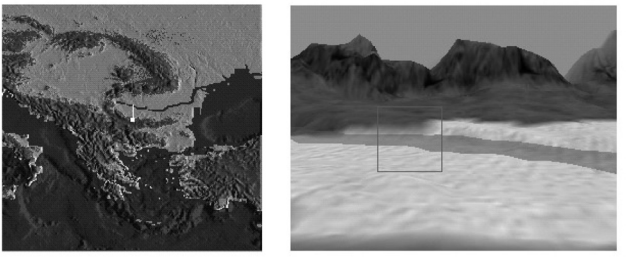

Fig. 6. On the left side the map of the virtual environment can be seen. In the middle the white arrow shows the position of the camera whose input image can be seen on the right side in gray-scale. The small window is the actually processed area.

4.1 Parameter Definitions

The most relevant parameters in our real-time experiments are summarized in Table 2.

Table 2. The most relevant parameters in our AIS visual analyzer model.

Parameter name	Meaning and values
InitSetSize	The size of the template set, usually between 100 and 500.
InitMethod	The initial method can be random creation or loading stored data.
ImmuneProcessLevels	Learning phase, recognition phase or combined phase.
Agelimit	Length of the non-active state after successful matching.
FeatureExMethod	Color or gray-scale feature extraction method.
FeatureExThreshold	Value usually between 1 and 400.
MatchThreshold	Needed threshold value for successful matching, usually between 1 and 100.
Mutation	True or false.
MutationValue	Probability value of mutation between 0 and 1.

4.2 Experiments

In the first experiment, we used the initialization and recognition phases separately. In the initialization phase, all the templates of the initial set was run on some input images of the actual view of the virtual world. The size of these input images was *64x64*. They usually contained some typical texture patterns, eg. texture of mountains, ocean or forest. During the process, in case of successful matching the template was selected out from the set. During the recognition phase the algorithm detected all textures, which were not members of the initial input flow. But those sub-patterns, which the system has been taught with, have not been detected. Detection results can be seen in Fig. 7.

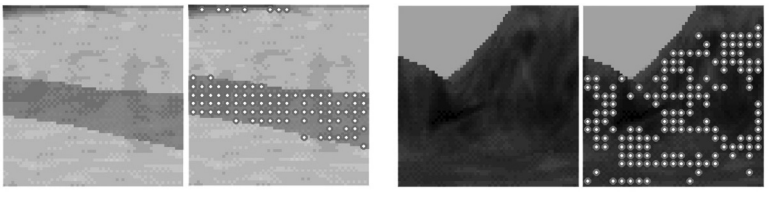

Fig. 7. The first and third images are different inputs. White dots on the second and fourth image show the result of the detection. In the first case the desert, in the second case the sky was already taught to the system. Note that different colors can be detected with the same template set.

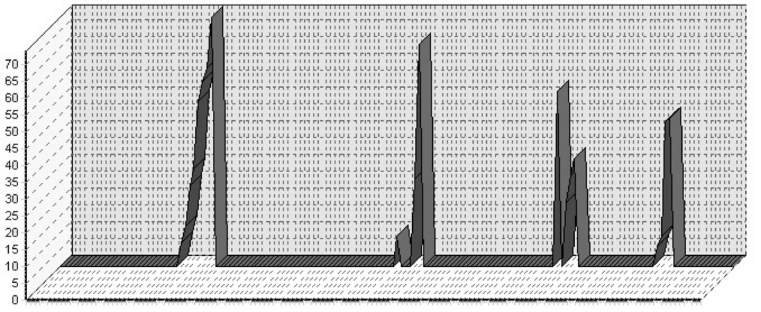

Fig. 8. The horizontal coordinates show the time based on the the input frames. The vertical value is the label of the actual template. Different peaks appear when the texture of the environment manly changed and at least one template has detected a new pattern. The first peak came when we reached the mountains, the second shows the water, third is ground again and the last one ocean again.

In the second experiment, the initialization and recognition part periodically followed each other. If any template had a detection, it was selected out temporarily. For a longer period if this template has detection, it was not allowed to send detection message. We assumed that if something was detected by this template, then the next few detections carry no new information, therefore it is not worth sending a warning message. The dynamics of this process can be seen in Fig. 8.

4.3 Observations

The time measurements of the algorithm are summarized in Table 3. The difference of the speed between the gray-scale and color method is caused by the data transfer in memory, as the color images are 3 times bigger than the gray-scale ones. These results show that our algorithm can run in real time even if we use the combination of two *3x3* templates for sub-pattern detection.

Table 3. Speed and time measurements of our visual analyzer algorithm.

Input image	Size of template set	Speed [msec] for a frame	Nr of million matches / sec
Color	100	25.641	15
Color	250	33.333	27
Color	500	52.632	36
Gray-scale	100	21.277	18
Gray-scale	250	28.571	31
Gray-scale	500	47.619	40

5 Conclusions and Future Work

In this paper, we presented a real-time visual analysis system for surveillance applications based on an Artificial Immune System (AIS) that can reliably detect unknown patterns in input image sequences. Our system is implemented on specialized architecture (the Cellular Nonlinear Network (CNN) Universal Machine) and its CMOS implementation. Results from tests in 3D virtual world with different terrain textures are reported to demonstrate that the system can detect unknown patterns and dynamical changes in image sequences. Applications of the system include in particular explorer systems for terrain surveillance. In the future, we would like to extend our system with robust classifier modules. Also, our aim is to further develop our mutation algorithm.

Acknowledgements. The support of MURI grant of the Office of Naval Research and the multidisciplinary doctoral school at the Faculty of Information Technology of the Pázmány P. Catholic University, are gratefully acknowledged.

References

1. Dasgupta, D.: Artificial Immune Systems and Their Applications. Springer-Verlag, Germany (1999)
2. de Castro, L. N., Timmis, J.: Artificial Immune Systems: A New Computational Intelligence Approach. Springer, UK, (2002)
3. Hofmeyr, S.A., Forrest, S.: Architecture for an Artificial Immune Evolutionary Computation. (2000) 7(1):45–68 http://www.cs.unm.edu/ steveah/
4. Satyanath, S., Sahin, F.: Artificial Immune Systems Approach to a Real Time Color Image Classification Problem. Proceedings of the SMC2001, IEEE International Conference on Systems, Man, and Cybernetics, Vol 4, (2001) pp.2285–2290, Arizona, USA.
5. Chao, D.L., Forrest, S.: Information Immune Systems. In Proceedings of the First International Conference on Artificial Immune Systems (ICARIS), (2002) pp. 132–140
6. Roska, T., Chua, L. O. : The CNN Universal Machine: An analogic array computer. IEEE Transactions on Circuits and Systems-II, Vol. 40, (1993) pp. 163-173
7. Chua, L.O., Roska T.: Cellular neural networks and visual computing, Foundations and applications. Cambridge University Press, (2002)
8. Zarandy, A., Rekeczky, Cs., Szatmari, I., Foldesy, P.: Aladdin Visual Computer. IEEE Journal on Circuits, Systems and Computers, Vol. 12(6), (2003)
9. Liñán, G., Espejo, S., Domínguez-Castro R., Rodríguez-Vázquez: ACE4k: An analog I/O 64 x 64 Visual Microprocessor Chip With 7-bit Analog Accuracy. Intl. Journal Of Circuit Theory and Applications, Vol. 30, May-June (2002), pp. 89–116
10. Cserey, Gy., Falus, A., Roska, T.: Immune Response Inspired CNN Algorithms for Many-Target Detection. in Proc. ECCTD '03, Krakow (2003).
11. Cserey, Gy., Falus, A., Porod W., Roska, T.: An Artificial Immune System for Visual Applications with CNN-UM. ISCAS 2004, Vancouver, (2004) paper accepted.
12. Cserey, Gy., Falus, A., Porod W., Roska, T.: Feature Extraction CNN Algorithms for Artificial Immune Systems. IJCNN 2004, Budapest, (2004) paper accepted.

13. Falus, A.: Physiological and Molecular Principles of Immunology. in Hungarian, Semmelweis Press, Budapest, (1998)
14. Roska, T., Kék, L., Nemes, L., Zarándy, Á., Brendel M., Szolgay P. (ed): CNN Software Library (Templates and Algorithms), Version 7.2. Analogical and Neural Computing Laboratory, Computer and Automation Research Institute, Hungarian Academy of Sciences (MTA SzTAKI), DNS-CADET-15, Budapest, (1998)
15. Kék, L., Zarándy, Á.: Implementation of Large-Neighborhood Nonlinear Templates on the CNN Universal Machine. International Journal of Circuit Theory and Applications, Vol. 26, No. 6, (1998) pp. 551–566
16. Antani, S., Kasturi, R., Jain, R.: A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video. Pattern Recognition, 35 (2002) pp. 945–965
17. Egmont-Petersen, M., de Ridder, D., Handels, H.: Image processing with neural networks–a review. Pattern Recognition, 35 (2002) pp. 2279-2301
18. Ivanov, Y., Stauffer, C., Bobick, A., Grimson, W.E.L.: Video Surveillance of Interactions. Proc. Second IEEE Int. W/S on Visual Surveillance, (1999) pp. 82–89
19. Hayashi, A., Nakasima, R., Kanbara, T., Suematsu, N.: Multi-object Motion Pattern Classification for Visual Surveillance and Sports Video Retrieval Int. Conf. on Vision Interface, (2002)
20. Dasgupta, D., Cao, Y., Yand, C.: An Immunogenetic Approach to Spectra Recognition. Proc. of the Genetic and Evolutionary Computation Conference, (1999) pp. 149–155
21. Tarakanov, A., Sokolova, S., Abramov, B., Aikimbayev, A.: Immunocomputing of the Natural Plague Foci. Proc. of the Genetic and Evolutionary Computation Conference, Workshop on Artificial Immune Systems and Their Applications, (2000) pp. 38–39
22. Carter, J.H.: The Immune System as a Model for Pattern Recognition and Classification. Journal of the American Medical Informatics Association, (2000) pp. 28–41
23. Carvalho, D.R., Freitas, A.A.: An Immunological Algorithm for Discovering Small-Disjunct Rules in Data Mining. Proc. of the Genetic and Evolutionary Computation Conference, (2001) pp. 401–404