

VI

Обробка та розпізнавання рукописної, креслярської та графічної інформації. Аналіз фотозображень і сцен

Hand-Written, Drawing and Graphic Picture Processing and Recognition. Photopictures and Scene Analysis

Neural Random Threshold Classifier in OCR Application

ERNST KUSSUL', TETIANA BAJDYK

NAS Institute of Cybernetics

*Ukraine, 252022 Kyiv
prosp. Akademika Hlushkova 40
Tel.: (044) 267-6996*

Ernst Kussul', Tetiana Bajdyk. Neural Random Threshold Classifier in OCR Application.

This paper contains some experimental results connected with application of Neural Random Threshold Classifier in OCR system for printed character recognition. The main goal of this work was to examine the possibilities of fast character recognition using neural classifier. Two types of characters were used for classifier testing: dot matrix printer characters and polygraph characters. Recognition speed on PC IBM 386, 40 MHz was up to 300 characters per second. Recognition rate was about 98 - 100 %.

Introduction. At present there are many OCR systems for printed text reading. Some of them have rather high performance and recognition quality. But there are some problems in such systems. In this paper we consider two problems: bad quality of scanned image and character merging. These problems may be solved by different methods but all of the methods lead to decreasing of recognition speed. We try to solve the problems using neural threshold classifier to preserve high recognition speed.

The paper summarizes the experimental results which were obtained on two types of printed characters. The first type are the characters printed by dot matrix printer. Characters of second types were got from polygraph texts.

The printer characters were recognized without errors. Recognition speed was equal to 40 characters per second. Error rate in the process of polygraph character recognition was equal to 98%, recognition speed - 300 characters per second.

Brief description of Random Threshold Classifier (RTC) will be given.

The Problem of Classification. Consider a surface in an n -dimensional space (feature space) partitioning the space into k regions called classes. Let us also have a training set consisting from number of samples (points in feature space); for each of those points we know which class it belongs to. The task of classifier construction is to create such procedure that will use the training set to indicate for any new point in the feature space (testing set) the class to which that point belongs. To estimate the performance (quality) of a classifier, it is reasonable to take into account the following characteristics: error rate on testing set, classification (recognition) speed for each testing point, and training time.

A number of neural networks can be used for pattern classification [1,2]. Many of them have perceptron-like structure. One-layer perceptrons have fast global convergence, but allow linear decision boundaries only. Multilayer perceptrons with several layers of modifiable connections allow to build arbitrary decision boundaries [2,3,4], but converge slowly and not always globally. Therefore some authors suggest to build multilayer perceptron classifiers with several fixed layers and single modifiable (trainable) layer. Fixed layers transform the input feature space so that in the resulted space the classes can be separated linearly, though the form of separating surfaces in the input space can be non-linear [5,6,7,]. We propose a classifier that belongs to the latter category of neural networks.

Random Threshold Classifier. RTC is based on functionally simple neurons. The output y of a neuron j is defined as

$$y_j = 1 \Leftrightarrow S_j > t_j, y_j = 0 \Leftrightarrow S_j < t_j, \quad (1)$$

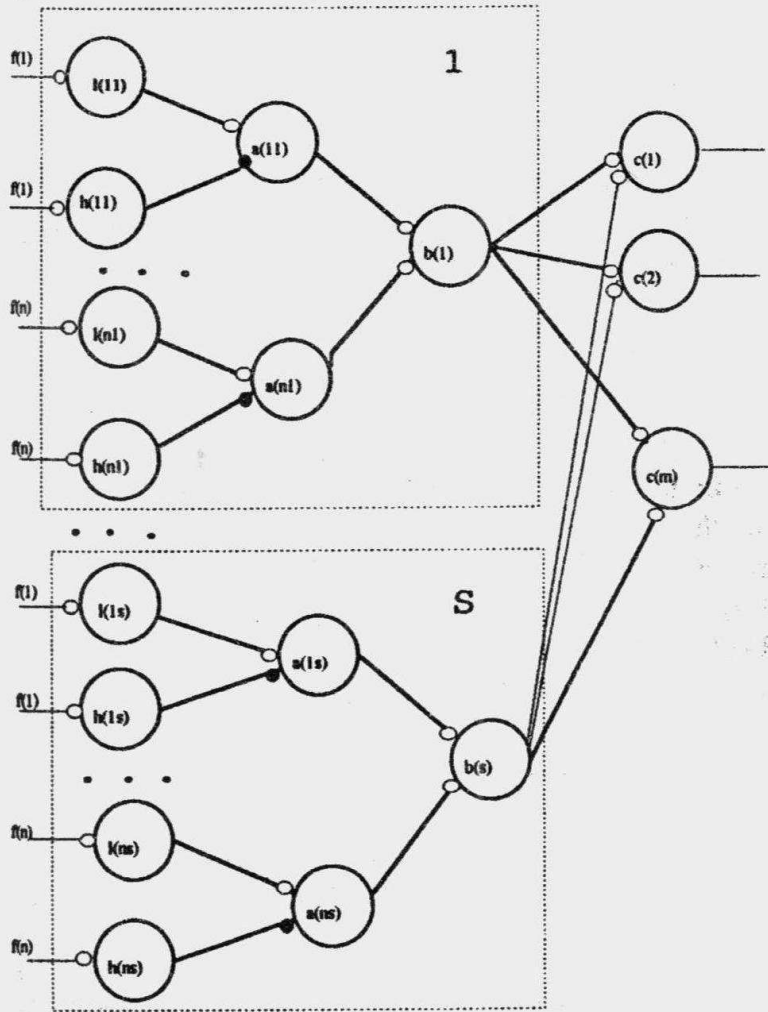


Figure 1.

where t stands for neuron threshold; the input sum S is calculated as

$$S_j = \sum_i (x_i w_{ij}), \quad (2)$$

where w_{ij} is synaptic weight of connection, x is the excitatory input of neuron j . Neuron also has inhibitory inputs. If a signal at the inhibitory input is set to one, the output of this neuron is set to zero.

Architecture of the RTC is shown in Figure 1. It comprises s neuron groups, $G(1), \dots, G(s)$, where s should be sufficiently large (see below). Inputs of each group are the features $f(1), \dots, f(n)$. Each neuron group has three-layer structure. The first layer contains n pairs of neurons, l_i and h_i , where $i=1,2,\dots,n$ corresponds to the number of features. The second layer consists of n neurons a_i . The third layer consists of one neuron b . Each feature f_i is connected with two neurons of each group, l_{ij} and h_{ij} , where $j=1,2,\dots,s$ is

the number of the corresponding neuron group.

The output of l_{ij} is connected with the excitatory input (o) of the following a_{ij} neuron, and the output of h_{ij} is connected with the inhibitory input of neuron a_{ij} . All the outputs of a_{ij} neurons from one neuron group j are connected with excitatory inputs of b_j neuron. The non-modifiable and have the weight 1.

The fourth (output) layer contains c_k neurons, $k=1,2,\dots,m$ is the maximum number of classes for the classifier. The b_j neurons are connected with c_k neurons by unidirectional associative modifiable connections ("all to all"). Weights of such connections are integer.

The thresholds of l_{ij} and h_{ij} neurons are chosen by random procedure. For each ij pair, the threshold of l_{ij} neuron is lower than that of h_{ij} neuron. The threshold of neuron a_{ij} is equal to 1, so that the output of the neuron a_{ij} is equal to one only when its excitatory input is one, but inhibitory input is zero. The output of b_j is the output of the whole j -th neuron group. The threshold of b_j is equal to n , so that the neuron b_j is active only when all the neurons a_{ij} from the j -th neuron group are active. All neurons of fourth layer have the same threshold which is controlled so that only one most excited neuron in fourth layer is active.

The learning rule for associative connections between the third and the fourth layers is the following. Let r be the number of class to which the input sample actually belongs; and k be the number of class by which the classifier labels the input sample. Then: If $k=r$ (correct classification) Do nothing. If $k \neq r$ (incorrect classification) do:

$$W_{jr} = w_{jr} + y_j ; \quad (3)$$

$$W_{jk} = w_{jk} - y_j \Leftrightarrow w_{jk} > 0 ; \quad (4)$$

$$W_{jk} = 0 \Leftrightarrow w_{jk} = 0 ; \quad (5)$$

Here j is the neuron number in the fourth layer, W_{jk} and w_{jk} stand for connection weight from b_j to c_k neuron before and after learning respectively, y_j is the output signal of neuron b_j .

The samples from training set are presented to the inputs of the classifier. After presentation of each sample and classifier response, Eq.(3)-Eq.(5) are used for modification of associative connections. Such training procedure repeats until convergence or for some number of passes through the whole training set.

RTC has been examined in complicated tests [8]. Our experiments have shown that the total performance (quality) of RTC is superior to that of the other investigated classifiers.

Possible applications of RTC in OCR system. We consider the possibilities of RTC application for OCR system which may be used for printed texts reading. We see two main applications: 1. fast search of text boxes on the page; 2. fast recognition of printed characters. At present we deal with second application. For experiments we have taken printed characters which were extracted from texts scanned by handscanner GS-4500. Each pixel of scanned image had only two values: 0 or 1. For using RTC in such circumstances some changes are needed. At first, we need only three values of threshold: 0, 0.5, 1. Because each neuron has two thresholds: low level and high level, there are only three threshold combinations: $l=0, h=0.5$; $l=0.5, h=1$ and $l=0, h=1$. The last combination is useless because in this case the neuron is always active. It permits us to use Boolean logic instead of real threshold calculation.

Input feature set for RTC was presented by pixel values from square window 32×40 . For experiments we used two fonts: 1. russian font from dot matrix printer; 2. roman font from magazine.

In the case of russian font we used for training the first copies of documents and for recognition it was used the second copies of other documents. 512 characters we used for training and 512 characters it was used for recognition. Neural network for RTC contained 2048

of b-neurons (fig.1) and 32 c-neurons. Letters of upper case and those of lower case weren't discriminated. In experiments with russian font we received more than 99% recognition rate.

For roman font we have taken 500 characters for training and 400 characters for recognition. RTC contained 640 b-neurons and 85 c-neurons (upper case letters and lower case letters, digitals and special symbols). All characters were subdivided onto four groups. Recognition process had two stages: 1. recognition of group; 2. recognition the character within the group. In this test we received more than 98% recognition rate and recognition speed was about 500 characters per second (on the IBM PC AT 386/387, 40 MHz).

Acknowledgements. The research described in this publication was made possible in part by Grant number U4M000 from the International Science Foundation and State Committee on Science and Technology of Ukraine.

References

1. Kussul E.M., Rachkovskij D.A., Baidyk T.N. Associative-projective neural networks: architecture, implementation, applications. Proceedings of Neuro-Nimes'91 - The Fourth International Conference on Neural Networks and their Applications. - pp. 463-476.
2. Lippman R.P. An Introduction to Computing with Neural Nets. IEEE ASSP Magazine, vol. 4, pp. 4 - 22, April 1987.
3. Minsky M. and Papert S. Perceptrons: An Introduction to Computational Geometry. MIT Press, 1969.
4. Rumelhart D.E., Hinton G.E., Williams R.J. Learning Internal Representation by Error Propagation. In D.E.Rumelhart and J.L.McClelland (eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1, Foundations, MIT Press (1986).
5. Gallant S.I. and Smith D. Random Cells: an Idea Whose Time has Come and Gone... and Come Again? IEEE International Conference on Neural Networks, 1987, II, pp. 671-678.
6. Huang W.Y., Lippmann R.P. Neural Net and Traditional Classifiers. In Anderson D.Z.[Ed.], Proc. of the 1987 Neural Information Processing Systems Conf., pp. 387-396, Am. Inst. of Physics, New York, 1988.
7. Giles C.L. and Maxwell T. Learning, Invariance, and Generalization in High-Order Neural Networks. Applied Optics, vol.26, N23, pp. 4972-4978, 1987.
8. Kussul E.M., Baidyk T.N., Lukovich V.V., Rachkovskij D.A. Adaptive High Performance Classifier Based on Random Threshold Neurons. In Trappl R. [Ed.] Cybernetics and Systems'94, Proc. of the 12th European Meeting on Cybernetics and System Research, V. II.- World Scientific Publishing Co.Pte.Ltd. - pp. 1687 - 1695.



Обробка та розпізнавання креслень у наскрізній технології автоматизованого проектування

МИХАЙЛО ГІНЗБУРГ, ІРИНА ТРЕБУЛЬОВА

НДПКІ АСУ Трансгаз

Україна, 310125 Харків

вул. Конєва, 16

Тел.: (0572) 205-734 Факс: (0572) 224-010

Mykhajlo Ginsburg, Iryna Trebuliova. Processing and Recognition of Drawings in the "Throughout" CAD Technology.

The methods which are used in the "Sapphire-P" automatized working place for carrying out the procedures of forming the schemes (original synthesis, dialogue correcting and updating), their recognition and extracting the necessary semantic information in the software development environment containing both graphical functions of the AutoCAD 10 system and textual functions of the Clipper system are described.

Вибір методів та засобів формування, обробки та розпізнавання креслень при автоматизованому проектуванні обумовлений кількома факторами, найбільш істотними з яких є розроблена технологія автоматизованого проектування та інструментальні засоби, що використовуються при створенні САПР [1].

В інтегрованому АРМ проектувальника комплексу технічних засобів (КТЗ) АСУТП "Сапфір-П" реалізована наскрізна технологія, що базується на системі перетворень, наведених у [2], причому найбільш важливим і складним є перетворення аналізу, яке ґрунтується на розпізнаванні креслень (схем) і добуванні з них семантичної інформації.

В даній доповіді розглянута реалізація процедур формування схем (початковий синтез, діалогові коригування, доповнення), їх розпізнавання та добування з них необхідної семантичної інформації в інструментальному середовищі АРМ "Сапфір-П", котре побудоване шляхом інтеграції графічного середовища системи АвтоКАД 10 та текстового середовища системи Clipper.

Графічна модель креслень та реалізація процедур формування схем. Креслення (файли *dwg*) являють собою сукупність графічних даних, формальна структура яких визначається принципами побудови системи АвтоКАД [3] і має вигляд системи трьох множин: загальних характеристик креслення, внутрішніх блоків, примітивів. Остання включає складні примітиви, що описують вставку блоків, і задання значень атрибутів. Опис блоків також будується з примітивів. Примітиви можна розподіляти по шарам.

Таким чином, графічна модель креслення включає такі основні поняття: блок, шар, примітив, атрибут.

В АРМ "Сапфір-П" для схеми кожного типу початковий синтез виконується за допомогою спеціальної процедури, алгоритм якої базується на правилах виконання схеми даного типу. Ця процедура написана на мові АвтоЛІСП і перетворює семантичну модель схеми у вигляді текстового задання на синтез у графічну модель. Залежно від типу схеми вказане перетворення може виконуватися або повністю автоматично (наприклад, для схеми живлення електроенергією), або (для інших схем) автоматизовано. В останньому випадку більшість параметрів міститься у заданні, а деякі дані, необхідні для геометричної прив'язки окремих блоків, запитуються у процесі їх вставки і задаються безпосередньо користувачем.

Діалогове коригування та доповнення схеми виконує користувач за допомогою проблемно-орієнтованого меню. Для схем кожного типу ці меню містять у собі команди вставки блоків (відповідних базових елементів, фрагментів схем або типових модулів [4]), надання значень атрибутам (безпосередньо в момент вставки, або у діалоговому вікні), редагування.

Методи розпізнавання та добування семантичної інформації з креслень. Семантичні моделі схем оперують такими основними поняттями: базовими елементами, модулями різних типів та з'єднувачами, які описують функціональні, електричні та трубні зв'язки між елементами. В [1] викладені результати інформаційного аналізу схем різних типів на КТЗ АСУТП і визначено, яку інформацію треба з них добувати.

Викладені у [4] методи формування базових елементів, фрагментів схем та типових модулів дозволяють добувати зі схем інформацію про склад об'єкта з елементів різних рівней ієрархії (підмножина $I_{ск}^i$) та про параметри елементів (підмножина $I_{пар}^i$) стандартними засобами системи АвтоКАД у вигляді атрибутних файлів (extract file). Для кожного типу схем визначена підмножина понять та сформована підмножина імен атрибутів, з якої будується файл шаблону, орієнтований на цей тип схем.

Для добування інформації про розміщення елементів у шафах, пультах тощо (підмножина $I_{роз}^i$) розроблено спеціальний діалоговий алгоритм креслення шаф, пультів у вигляді поіменованих прямокутників, у два кути кожного з яких автоматично вставляються блоки-обмежувачі, що містять позначку на службовому шарі та ідентифікаційний прихований атрибут. Розміщення елементів виконується користувачем в діалозі шляхом переміщення елементів на полі креслення або шляхом вибору необхідних зображень з піктографічного меню та задання місця вставки відповідних блоків. Усі геометричні перетворення прямокутників (розтягування або стискання вздовж будь-якої осі, перенесення та ін.) здійснюється за допомогою включених до меню спеціальних команд, виконання яких веде до погодженої зміни зображення прямокутників та координат місць вставки блоків-обмежувачів. Такий підхід дозволяє добувати зі схем дані $I_{роз}^i$ також стандартними засобами системи АвтоКАД у вигляді атрибутних файлів та об'єднати з добуванням даних $I_{ск}^i$ і $I_{пар}^i$.

Більш складною є задача добування зі схем опису зв'язків між елементами ($I_{зв}^i$). Для її вирішення в системі "Сапфір-П" усі з'єднувачі (незалежно від їх ширини) виконуються за допомогою примітива "полілінія", котрий сприймається системою АвтоКАД як єдиний складений примітив незалежно від кількості сегментів (відрізків) ламаної. Розроблена спеціальна процедура на мові АвтоЛІСП, яка аналізує файл креслення та формує три текстових файли, на основі яких та файлу добування формується опис з'єднувачів.

Таким чином, розроблено методи розпізнавання схем та добування з них усіх підмножин вихідної інформації, котрі базуються на стандартному інструментарії системи АвтоКАД, доповненому необхідними процедурами на мові АвтоЛІСП.

Прикінцеві положення. Розроблені для АРМ "Сапфір-П" методи обробки та розпізнавання креслень у середовищі системи АвтоКАД дозволяють виконувати перетворення семантичних моделей схем у графічні та навпаки і завдяки цьому реалізувати наскрізну технологію автоматизованого проектування.

Література

1. Гінзбург М.Д. Обробка та розпізнавання графічної інформації у інтегрованій системі автоматизованого проектування "Сапфір-91" // Обробка сигналів і зображень та розпізнавання образів. Праці. Перша Всеукр. конф. - Київ, 1992. - С. 146-148.
2. Гінзбург М.Д. Нова інформаційна технологія автоматизованого проектування, що ґрунтується на обробці та розпізнаванні графічної інформації // Обробка сигналів і зображень та розпізнавання образів. Праці. Всеукр. міжн. конф. "УкрОБРАЗ-94". - Київ, 1994.
3. Курс практической работы системы АвтоКАД 10 // С.А.Гладков, Ю.А.Кречко, К.И.Молодцов, В.В.Полищук, Г.А.Сучков. -М.: Диалог-МИФИ, 1991. -285 с. 4. Дубровський В.В., Гінзбург М.Д. Інтегрована база графічної та текстової інформації для задач САПР // Обробка сигналів і зображень та розпізнавання образів. Праці. Перша Всеукр. конф. - Київ, 1992. - С. 150-151.



The Top-Down Algorithm of Drawings Recognition

VIKTOR CHEKH, OLEH BARANOVSKYJ

"Shtorm" Design Office

Ukraine, 252056 Kyiv

Politekhnichna Str., 16

Phone: (044) 446-9263 Fax: (044) 226-2396

E-mail: chekh%storm.kiev.ua@ts.kiev.ua.

Viktor Chekh, Oleh Baranovskyj. The Top-Down Algorithm of Drawings Recognition.

The problem of the recognition of the complex drawing and graphic pictures is very actual for the large number of applications, such as OCR including handwritten texts recognition, automatic recognition of electric circuits, geographic maps, etc. This paper describes one algorithm to be used for the solving of such problems.

Accordingly to the described algorithm to recognize a complex drawing one have to do the following. First, to process the image to be recognized to have its skeleton in a view of planar graph. Its nodes are the morphemes, i.e. the special points of the skeleton. Second, one have to assemble this morphemes into the set of hyper morphological structures with the definitely ordered morphemes (so called ways). It makes possible to transform the initial graph into the tree with the nodes of ways. This tree could be transformed by top-down way into the string of ways or, finally, into the string of initial morpheme because of order in the ways. The string could be recognized by comparing the weighted Levenstain distance between it and basic patterns.

Preprocessing. Let hereafter pattern to be the connective set of the pixels on 2D bit image. The first step of our algorithm needs a skeleton of this pattern; it have to be transformed into the graph of morphemes. We call "morpheme" the special point of the skeletonized image, i.e. the points of the skeleton with definite surrounding such points of skeleton as lines crossings, ends of lines, turnings of lines, etc. with its features such as coordinates, type, etc. The method of skeletonization, number and types of morphemes are not important for the next steps; the optimal ones depend of the particular problems. So, the morphemes will be the nodes of the graph, the

skeleton lines will be its branches. It must be noted that generally such a graph of the pattern to be recognized has a number of loops and therefore it could not be a tree.

For the next step we have to determine the term "way". Let way to be a hipermorphological structure, i.e. include a number of morpheme in definite order. For the applications we are specialized in, three different types of ways are enough. They are the following: "dead-end", which includes one node point (points of cross intersections of skeleton lines), tail (endline point) and all other morphemes between them if none of them is another node; "rings" include all morpheme in the skeleton lines which makes the borders of non intersected exclusive regions of the initial image; "stage" includes two nodes and all morphemes on the non crossed line between them if they are not belong to any ring. It is easy to prove that any planar graph could be fully and uniquely disassembled onto such a set of ways.

Now all loops are concentrated only in the corresponding rings. It makes possible to transform initial graph into the tree; the ways will be its nodes and leaves. For the large number of applications it is possible to order the tree from left to right (OCR in European languages) or any another definite way (OCR in Hebrew, Japanese, electrical circuits recognition, etc.); so, let our tree to be ordered in the same way. The morpheme in the ways have to be ordered as well by the distance from the top node of the tree (dead-ends and stages) and by clockwise from the morpheme which is nearest to the top node (rings).

One usual way for pattern recognition and classification is to compare the basic set of patterns with the pattern to be classified. The method of comparing of the strings with different number of signs by calculating the Levenstain distance DI between them [1], or, more often, weighted Levenstain distance [2] is well-known. For our patterns morpheme are the signs and to calculate DI one have to transform the tree of ways into the string of morphemes. We propose the top-down rule for this transforming because it is the most natural for the most applications. So, one have to write the ways of classified pattern by the following order: first, the top node of the tree; when - the dead-end ways which are directly connected with the node written before by the clockwise from the direction to the top node; then - rings, then, finally, stages in the same order. Now we have the string of ways and, naturally, the string of morphemes because the morphemes constituting every way are being ordered before. If one needs to have additional information about any way (longitude, square of surface which is bordered by ring, shape coefficients or more) it is possible to add it to the string as the additional morpheme preferably before another ones in corresponding way. Now the string of the signs of the pattern is ready for finally processing.

Recognition. Forming the basic set of patterns. Before the recognition one have to create the basic set of patterns. Anyone of them has, of course, the similar view (string of morphemes) and their set has a view of a number of trees with the different top nodes. For this set of trees the top nodes are first morphemes of basic patterns. If any two basic patterns have the same first morpheme (may be within some dispersion value - depends of the particular problem) they form the same tree, if not - different ones. Every leaf node of every tree means different basic pattern but not visa versa generally (for example, characters "d" and "o" have just the same top node and the number of the similar morpheme which forms ring way, the difference between them is in the next morpheme which forms two dead-end ways in "d" and absent in "o"). For the very start (before training) it recommends to create the matrix and vectors of

Levenstain coefficients $S(ai, bi)$, $D(ai)$, $I(bi)$ with equal values if there is no additional information. The set of such basic patterns must include all probably patterns or their large majority (if not, training will be enough protracted). Hereafter $S(ai, bi)$ is the weight of change the sign ai for the sign bi , $D(ai)$ is the weight of deleting the sign ai from the string, $I(bi)$ is the weight of adding the sign bi to the string.

Training with the supervisor. The string formed from the pattern to be recognized is compared with the basic patterns. First we choose the tree corresponding to 1-st morpheme in the string, then the branch of the chosen tree corresponding to the next morpheme, etc. If after the end morpheme of recognizing pattern one have full coinciding of the string of its signs with any basic string it means right recognition. The pattern recognized and classified as the corresponding basic pattern. The non diagonal coefficients of $S(ai, bi)$ could be decrease automatically as well as $D(ai)$, $I(bi)$, diagonal ones could be increased.

If after some step the morpheme of string being recognized does not coincide with any next ones in the tree of basic patterns one have to make top-down parsing of the rest tree. If the number of the rest patterns down the last node is enough small - then the recognizing string must be compared with the strings which ends are down in the chosen tree accordingly to Levenstain algorithm. If the weight Levenstain distance Dl between the recognized string and one of basic ones is less then threshold distance $Dthr$ then it considered that the pattern recognized correctly. In this case $S(ai, bi)$ corrects automatically to decrease the distance between basic pattern and the just recognized one and to increase the distance between another patterns in the basic set. If the condition $Dl < Dthr$ satisfies for more then one basic strings the supervisor have to choose correct one. After this choice the automatic correction of $S(ai, bi)$, $D(ai)$, $I(bi)$ must be realized.

If after the top-down parsing the pattern was not recognized then it could be repeated from the node above the last one coinciding with the recognizing string until the recognizing or not more then any fixed number of steps. Of course, one could compare the string with every string of basic set if he have enough time. If nevertheless the pattern was not recognized the supervisor has to recognize the pattern himself. If there is no such a base pattern in the set (new character or new symbol) or it is too distant from the corresponding pattern this string writes to the basic set of patterns. In this case the number of patterns in the basic set increases, in another case it is reasonable to change Levenstain weights and not to add one more string in the basic set.

Now one can see that the training means changing of the Levenstain weights. It must be noted that for the successful recognition this process must be steady. The possible steadiness could be then the weight coefficients can increase unlimitely, or any of them first increase essentially then decrease and so on. Possible solutions to save the stability of the training are the following: to restrict the sum of the Levenstain weights and change them only relatively; to add new strings in the basic set for the different patterns with the same names (the same characters but written by different hands or typed with different fonts).

Automatic recognition. This mode is correct to be used after the previous one (training). It assumes that the system is already trained, therefore this mode needs no supervisor. The difference between this mode and the previous one is that during the auto recognition the Levenstain matrix and vectors of Levenstain weights is not

changes, therefore this mode is more quick then the training. If any pattern was not recognized or was wrong recognized in this mode one needs of supervisor's help and the system have to switch in training mode

Concluding remarks. Now we have described briefly the algorithm based on disassembling onto the set of morphemes the skeletonized pattern on the bit plate. Its main advantages are the following:

- quick recognition because the search of basic pattern is on the trees;
- it is flexible and handle because of simple training mode.

It is clear that the greater number of signs for the pattern means, generally, its better recognition, but it must be noted that it simultaneously increases the time of recognition. The time of two strings comparing is proportional to the product of their lengths (i.e. it is $O(L_i * L_j)$, where L_i , L_j are the numbers of signs in i-th and j-th comparing strings correspondingly, because the dimensions of the matrix $S(ai, bi)$ are $L_i \times L_j$). The problem how to choose optimally the number and types of morphemes depends on particular application.

During all the time we assumed that we have the skeleton of our pattern and it is possible to disassemble it onto the set of morphemes. For the great number of problems these two processes (skeletonization and morphological disassembling) need much more time than the following recognition itself. The algorithms of quick skeletonization and morphological analysis of complex drawings and their applications are outside the frames of this short paper and will be described separately.

References

1. Левенштейн В.И. Двоичные коды с исправлением выпадений, вставок и замещений символов // ДАН СССР, 1965, т. 163, #4. с. 840-85.
2. VLSI for pattern recognition and image processing / ed. King-Sun Fu, - Springer-Verlag, 1984.



Techniques of Scene Radar Image Processing and Their Recognition by Expert System

OLEKSANDR ZELENSKYJ, ANDRIJ KUREKIN,
VOLODYMYR LUKIN, MYKOLA PONOMARENKO

Aviction Institute

*Ukraine, 310070 Kharkiv
Chkalov str., 17*

Phone: (0572) 44-2665, (0572) 44-2352 Fax: (0572) 44-1131

Александр Зеленский, Андрей Курекин, Владимир Лукин, Николай Пономаренко.
Методы обработки радиолокационных изображений сцен и их распознавание экспертной системой.

Обсуждаются основные аспекты интерпретации радиолокационных изображений и анализа сцен. Предлагается использование экспертной системы для классификации типичных радиолокационных объектов на основе анализа локальных статистических

параметров. Представлены экспериментальные результаты применения методики к реальным радиоизображениям.

Introduction. One of the most frequent tasks of remote sensing radar images processing consists in extraction of useful information about investigated objects such as lots of terrain with vegetation, forest areas, agricultural fields and so on. But for investigation of the object characteristics its localization have to be performed. The problem of radar objects detecting and recognition is especially actual in tasks of joint multichannel radar images processing. For example, procedure of mutual images superimposing requires the selection and recognition of reper points for every image, such as contrast bright objects (bridges, houses, cross-roads) /1/.

The ways for solving of this problem are mainly determined by the peculiarities of objects and characteristics of radar images. In the majority of cases the presence of intensive multiplicative and impulse noise, weak contrasts and the variety of object forms sufficiently restricts the application of optic images scene processing algorithms (for example threshold division for segmentation and Sobel operator for edge detection /2/). Taking this fact into account we have elaborated a new approach to radar image analysis with the use of expert system with fuzzy logic.

In this paper we discuss possible types of the simplest image objects, determine local parameters for their recognition in conditions of weak contrasts and speckle noise, present the principles of expert system operation, show the efficiency of proposed approach for real remote sensing millimeter (MM) band radar images.

Proposed Image Objects Recognition Procedure. We have investigated different radar images in MM, Centimeter, Meter bands and considered typical radar objects from the point of view of the possibility of their detection and recognition on the images. In many cases objects even with a rather complex forms can be presented as a composition of several objects of the simplest form. Besides that the ability of expert system to the classification of different types of objects is determined in fact by the sizes of image scanning aperture. Obviously, in the case of various object types the procedure of classification is sufficiently complicated and calculation expenses are inadmissible. So let us assume that the image scanning window dimensions are equal to $N=MqL$ image pixels. Taking into account the correlation between neighbour pixels it must contain at least 10-15 independent image elements. That is why instead of apertures 5x5 pixels used for test data we recommend to use scanning window 9x9 elements for real radar image processing. The objects which are the most typical and important for image processing are the following : 1) the homogeneous lot (large dimension "flat" object); 2) central pixel edge belonging; 3) central position value corresponds to impulse noise (spike); 4) small-dimension object presense in the window and central pixel belonging to it.

The traditional approach to radar images analysis is based on homogeneous lots and edges recognition. There are many edge detectors for speckle and impulse noise corrupted image processing /3/. The main disadvantages of these techniques deal with difficultes of small-dimension (houses, bridges) and prolonged narrow objects (roads, rivers) recognition. To overcome this restrictions we use several local scanning window statistical parameters for objects type recognition. The following local parameters were selected as the most informative:

a) ratio $\sigma_{ij}^2 / \sigma_0^2$, where σ_{ij}^2 is the relative (normalized by \bar{I}_{ij}^2) local variance, σ_0^2 - speckle relative variance, \bar{I}_{ij}^2 - local mean value;

b) quasirange $Q_{ij} = (I_{ij}^p - I_{ij}^q) / \bar{I}_{ij}$ where I_{ij}^p and I_{ij}^q denote p-th and q-th scanning window sample order statistics;

c) $r_{mm} = |\bar{I}_{ij} - I_{ij}^{med}| / \bar{I}_{ij} \sigma_0$, where I_{ij}^{med} is the median value of the sampling;

d) the coefficient of neighbourhood belonging t , which is equal to the ratio N_b / N where N_b is the number of the image values belonging to the area $[I_{ij}(1 - 2\sigma_0); I_{ij}(1 + 2\sigma_0)]$;

e) the coefficient of pixel rank position $Z = |L_{ij} - (N + 1) / 2| / N$, where L_{ij} is the rank number of the central scanning window value in the ordered sampling formed by pixels of its aperture;

f) the non-symmetry coefficient $S_n = |I_{ij}^N + I_{ij}^1 - 2\bar{I}_{ij}| / \bar{I}_{ij}$.

After local statistic parameters calculation, they are jointly used in expert system for objects recognition. For every pixel of the image the probability P_{ij}^m of its relating to m-th object type mentioned above is calculated. The decision about the object type is adopted if its probability is the maximal. The procedure modifications for real data recognition are recommended.



Fig. 1. MM-band radar image

Experimental Results. Test image models and easily recognisable fragments of real data with different object contrasts, dimensions and configurations having different noise and spikes properties were used for the control parameter distribution laws obtaining. Taking into consideration expert knowledge they were extrapolated and corrected for image and noise properties not scoped by simulation data. The effectiveness of the expert system is demonstrated as applied to real MM-band radar image (Fig.1) taken over agricultural area of Kharkov region. The scene contains different objects of interest (roads, boundaries between agricultural fields), presenting a more rigorous test for objects detection. Fig.2. shows the boundary map obtained by applying the expert system to the image in Fig.1. Here low contrast boundaries (between fields) and prolonged narrow objects (roads) are as clearly marked as distinct high-contrast boundaries. Experiments have shown that the probability of true decision varies from 0.8 to 1.0 depending on images and noise characteristics.

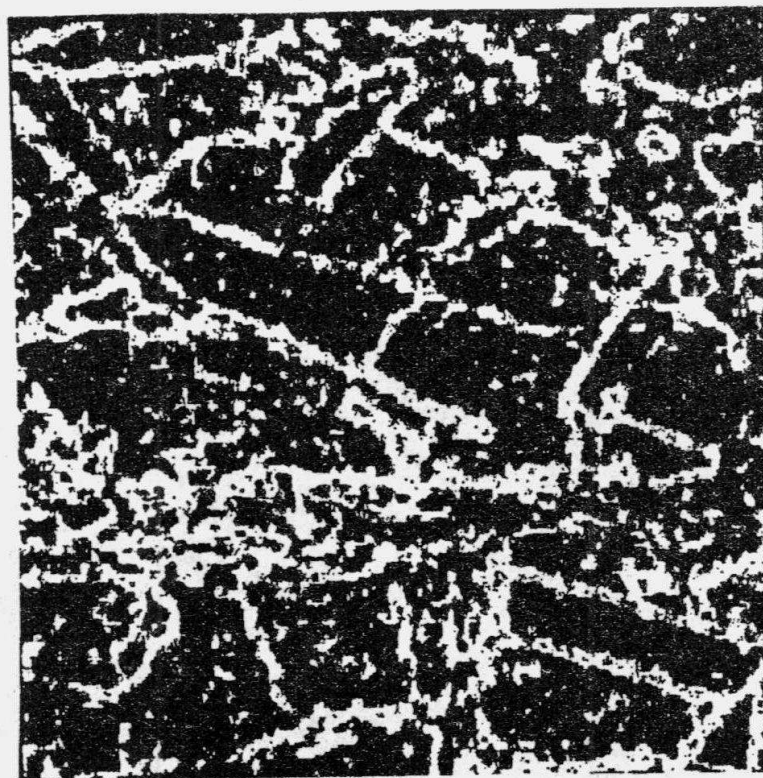


Fig. 2 Boundary map of radar image

The results show that for real radar images more complicated edge detection and segmentation algorithms have to be designed. They can be elaborated on the basis of mentioned local parameters complex analysis sensitive to various types of objects, for instance, to edges and one-dimension prolonged objects.

References

1. Lukin V.V., Miao Zhenjiang, Yuan Baozong, "Multifrequency Remote Sensing Radar Images Processing and Analysis", Proceed. of TENCON'93, Beijing, China, Oct. 1993, pp. 1042-1045.

2. Pavlidis T. "Algorithms for Graphics and Image Processing", 1982 Computer Science Press, Inc.
3. Alan C. Bovik, "On Detecting Edges in Speckle Imagery", IEEE Trans. on Acoustic, Speech, and Signal Processing, vol. 36, NO. 10, Oct. 1988.

◆

Моделі автоматичного сполучення сегментованих дискретних зображень

СЕРГІЙ СТАНКЕВИЧ

Інститут військово-повітряних сил

Україна, 252186 Київ

Повітрофлотський проспект 30

Тел.: (044) 271-6236

Serhij Stankevych. The Models of Segmented Discrete Images Automated Matching.

The adduced set of models is the theoretical foundation of the digital images matching algorithms elaboration for tasks of aerospace monitoring.

Процеси сполучення цифрових зображень знаходять широке застосування при обробці матеріалів аерокосмічного моніторингу земної поверхні. Операції сполучення є необхідними елементами процедур визначення координат об'єктів, що переносяться із зображення на карту місцевості, процедур поєднання зображень з несхожою геометрією одержання, синтезу безрозривних зображень, що отримуються за допомогою апаратури циклічної події, розміщеної на рухомому носії. При комплексній інтерпретації наборів зображень, що отримуються у різних зонах електромагнітного спектра, точне попереднє сполучення є обов'язковою умовою успішного розпізнавання.

При відсутності додаткової інформації про взаємну орієнтацію та геометричні спотворення зображень, які сполучаються, прийнятна якість сполучення може бути досягнута [1] тільки при заданні бієкції над множинами надійної ідентифікованих опорних точок зображень. Вихідні дані, які найбільш підходять для відновлення бієкції, дає сегментація оброблюваних зображень. Методи виділення сегментів розроблені досить добре [2]. Відновлювання бієкції над множинами опорних точок слід здійснювати у два етапи: побудова бієкції між сегментами (їх підмножинами) та відновлення бієкції над характерними точками відповідних сегментів.

При побудові бієкції між сегментами зображень, які розглядаються, прийняті такі допущення: бієкція існує; сегменти, що сполучаються, на зображеннях присутні цілком; спектральні спотворення форми незначні. Система вихідних ознак повинна містити у собі: топологічну підсистему (топологія сегмента і міжсегментна топологія); морфологічну підсистему (насамперед повинна враховуватись форма контуру); спектральні (енергетичні і контрастні) характеристики; параметри текстури сегментів. Зручним формалізмом аналізу сегментованих зображень є семантичний граф [3] сегментів. Над семантичними графами може бути побудована нечітка міра подібності $\mu\{\cdot\}$ та введені процедури перетворення (редукції), метою яких є відновлення гомоморфізму G семантичних графів:

$$\mu\{G[S(I)], G[S(J)]\} \in M_{\text{доп}} , \quad T\{G[S(I)]\} = \underset{T \in M}{\Gamma} T\{G[S(J)]\} , \quad (1)$$

де I, J - вихідні зображення, $S(\cdot)$ - множини сегментів зображень, $G[\cdot]$ - семантичні графи сегментів, $M_{\text{доп}}$ - припустима область мір подібності, $T\{\cdot\}$ - перетворення семантичних графів, M - клас можливих перетворень.

Під час аналізу встановленої відповідності між сегментами доцільним здається відновлення бієкції між точками їх контурів. Породжувальним для бієкції інваріантом може виступати відносна довжина контуру. Задача виділення контуру сегмента є досить простою. Використання кривих для відновлення бієкції здається перспективним, тому що породжує потенційно більше число опорних точок і забезпечує більш високу якість сполучення. Взаємна орієнтація контурів, які виділені, може бути визначена за екстремумом міри близькості у просторі параметрів, інваріантних до зсуву та масштабу. Зразком такого перетворення сегмента є надмірне перетворення Хо [4].

Гарні результати дає використання дискретного аналога кривизни контуру $r\{l\}$, який узято вздовж його відносної довжини l . Застосування вказаної моделі передбачає мінімізацію міри розбіжності контурів виду:

$$\langle l[C(I)], l[C(J)] \rangle = \arg \min_{li} \frac{1}{n} \sum_{i=1}^n [r\{l_i[C(I)]\} - r\{l_i[C(J)]\}]^2, \quad (2)$$

де $C(I), C(J)$ - контури сегментів, що сполучаються, n - кількість пар опорних точок.

Задання бієкції над множинами опорних точок вихідного і цільового зображень дозволяє визначити відповідне до неї відображення масиву вільних (неопорних) точок зображення, яке трансформується. В моделі, що описується, потрібне відображення декомпозується у вигляді послідовності узагальненого лінійного перетворення

$$x_i(J) = Z x_i(I) + z, \quad i=1, \dots, n, \quad (3)$$

де $x(I), x(J)$ - вектори координат точок вихідного I та лінійно перетвореного J зображень, Z - матриця лінійного перетворення, z - вектор паралельного переносу і залишкової нелінійної деформації, яка визначається векторним полем власних зміщень над множиною опорних точок лінійно перетвореного зображення

$$F[x_i(I)] + \sum_{j=1}^n k_{ij}(I) F[x_j(I)] = x_i(J) - x_i(I), \quad i=1, \dots, n, \quad (4)$$

де $F[x_i(I)]$ - власний вектор деформації зображення I у точці $x_i(I)$, $k_{ij}(I)$ - коефіцієнт передавання деформації в зображенні.

Параметри узагальненого лінійного перетворення (3) можуть бути розрахованими методом найменших квадратів на заданій бієкції:

$$\langle Z, z \rangle = \arg \min_{Z, z} \sum_{i=1}^n [x_i(J) - Z x_i(I) - z]^2. \quad (5)$$

Вектор залишкової деформації у вільній точці $x_i(I)$ складе

$$V[x_i(I)] = \sum_{j=1}^n k_{ij}(I) F[x_j(I)] \quad (6)$$

де $F[x_j(I)]$, $j=1, \dots, n$ одержані розв'язанням системи лінійних рівнянь (4), а

$$k_{ij}(I) = 1 - \frac{d_{ij}(I)}{d_{\text{хар}}}, \quad d_{\text{хар}} > d_{ij}(I) \quad \forall I \quad (\text{модель дальніх зв'язків}) \quad \text{або}$$

$$k_{ij}(I) = \exp[-[d_{ij}/d_{\text{хар}}]^2] \quad (\text{модель близьких зв'язків}),$$

$$d_{ij}(I) = |x_i(I) - x_j(I)|, \quad d_{\text{хар}} - \text{характерний розмір затухання деформації.}$$

Таким чином, наведені моделі дозволяють визначити закони трансформування сегментованих дискретних зображень і після впровадження ряду допоміжних операцій (наприклад, яскравісної інтерполяції на базовій решітці) можуть бути використані при розробці програмного забезпечення цифрових систем обробки зображень, насамперед тих, які отримуються засобами дистанційного спостереження аерокосмічного моніторингу.

Література

1. Арбузов П.А. О методе совмещения изображений, получаемых различными средствами воздушной разведки // Некоторые вопросы получения и обработки данных воздушной разведки.- Киев: КИ ВВС, 1993.- С. 38-40.
2. Лисицын В.М., Обросов К.В., Пасечный Н.Н., Стефанов В.А. Итерационные алгоритмы сегментации многомерных изображений // Известия АН. Техническая кибернетика, 1993.- I 6.- С. 103-113.
3. Вагин В.Н., Гулидова В.Г., Фомина М.В. Распознавание состояний сложного объекта при неполной входной информации // Известия АН. Техническая кибернетика, 1992.- I 5.- С. 120-132.
4. Марков С.Ю. Избыточное преобразование X_0 // Информационные технологии в дешифрировании изображений: методология, модели, алгоритмы.- Киев: МО Украины, 1993.- С. 22-28.

MapScan - the Software Tool for Vectoring of Raster Maps

IHOR MANIUK, VIKTOR CHEKH

"Shtorm" Design Office

Ukraine, 252056 Kyiv

Politekhnichna Str., 16

Phone:(044) 446-9263 Fax: (044) 274-1015

E-mail: chekh%storm.kiev.ua@ts.kiev.ua.

Ihor Maniuk, Viktor Chekh. MapScan - the Software Tool for Vectoring of Raster Maps.

The MS Windows program named MapScan for raster maps processing and vectoring has been developed by the authors. It can separate and vectorize 1,2,3D objects in a view of symbols, polylines, regions correspondingly and to create output files in map database formats (such as MapInfo's .MIF, .MID). The program could be used for vectoring of scanned maps, air or satellite photographs, etc. Contact us for a FREE DEMO version of this program.

The number of map databases users is extremely large and rapidly increases. The best view of the maps for the handle operation is a vector one, but it is enough difficult to create the vector map. Now there are many raster maps drawing on a paper; it is possible to have a good photo from the satellite or from the airplane. It is not difficult to input this images to the computer with scanners but it is a problem to vectorize this images because of many noises, great number of different objects and types of objects which have to be recognized, separated or united. It is considered that the optimal solution of this problem is fully automatic recognition and vectoring of such a maps by the computer which has some minimal instructions of the supervisor.

The program named MapScan is an example of solution of the problem of computer vectoring of maps. It works with MS Windows 3.0 or 3.1 on correspondent PCs (preferably powerful). It could operate with the scanners directly or with graphic files (BMP, PCX, etc.). Output files are now in .MIF format for direct usage of the resulting vectorized map by MapInfo. Of course, it is simple to use any another output format.

The program can recognize lines and output their skeleton in a view of polylines. Crossings lines recognized automatically and could be written as one object. Regions including complex ones write in a view of polyline borders with color of initial region. One can use both manual mode with prompting and full automatic mode. The last one returns all colors and geometrical dimensions with minimal differences with the initial image. Using the manual mode it is possible to choose any object of any type and input it in real time with any parameters (color, width, smoothing, etc.). Different types of prompting make the work with the program very handle for any user.

It is possible to I/O and process images of any size less than free disk space or RAM, to operate with the program either with the mouse or with the keyboard.

The program is possible to be localized onto any computer platform especially parallel ones. Open structure of the program makes possible to apply additional libraries, use libraries of this program in another programs. MapScan is the current project and will be completed with the powerful image processor and the tools for automatic recognition of the symbols. The first presentation of its first PC version has been in Troy, NY, on the conference of MapInfo users. Please contact us for FREE DEMO version of the MapScan or any additional information.



An Image Recognition Using the Integral Shape Features by Optical-Electronic Systems

VOLODYMYR BOROVIYTS'KYJ, VOLODYMYR POREV, MYKOLA IEGOROV

Scientific-Research Institute "Spectr"

Ukraine, 252056 Kyiv
prospect Peremohy 37
Tel.: (044) 441-9970

Volodymyr Borovyts'kyj, Volodymyr Porev, Mykola Iegorov. An Image Recognition Using the Integral Shape Features by Optical-Electronic Systems

Authors analyse three factors for the recognizing optical-electronic system(OES) design. The first factor is the integral shape features that are a measure of differences between an object image(OI) and a circle. The second factor is the OES parameter optimization by a minimization of the OI relative square value error. The third factor is the image recognition procedures that use a quadro-tree for finding segmentation thresholds and a classification by comparing a weighted distances in the feature space with the threshold value.

The optical-electronic systems (OES) that use integral shape features(ISF) for automatic detection and recognition apply in medicine, aviation and robototechnics. When the OES is being designed the important factors that define probability and time of a correct recognition are the ISF that contain information about an object image shape, the OES parameters and the programs for image processing.

Usually the ISF vector has features that are calculated identically like moments, Fourier descriptors and others. But the ISF vector could be made up using different ISF on the following assumption : the object image (OI) is a connected zone and its shape is a result of transformations of a circle by numbers of local pressings and skretchings without zone square changing. According to the assumption the ISF vector becomes a measure of differences between the OI shape and a circle. It can be put together using knowing features :

$$P = \left\{ p_1 = \frac{L}{L_0}, p_2 = \frac{R_{\max}}{R_{\min}}, p_3 = \frac{R_{\max}}{R_0}, p_4 = \frac{M_{\max}}{M_{\min}}, p_5 = \frac{M_c}{M_0}, p_6 = \frac{W}{W_0}, p_7 = \frac{W_0}{R_0} \right\}, (1)$$

where P, p_i - the ISF vector and its element, respectively; L_0, R_0, M_0 - a length of a contour, a radius, and a central inertial moment of the circle which square is equal the OI square, respectively; L, R_{\max}, R_{\min} - a length of a contour, maximum and minimum distances between points of the OI contour and the OI center, respectively; M_{\max}, M_{\min}, M_c - maximum and minimum axial and central inertial moments of the OI, respectively; W_0, W - modules of null and all Fourier decriptors of the OI contour, respectively.

Notice that the ISF(1) are invariable to scaling and rotating. They require a small number of calculations and their meaning is clear.

The well-known criteria for the OES parameter optimization is a maximization of a signal to noise ratio (SNR) that is provided by a correlative spatial filtering. But using the ISF makes this optimization difficult because the OI may have various scales and rotations and they might be among bright other images that reduce spatial filter possibilities. The computer simulation of the OES shows that the OI shape distorsion may be characterized by the relative square value error (RSVE). The RSVE

depends on the SNR and a contour destruction by a high space harmonic depression. The following expression defines the RSVE in case of normal distribution of noise :

$$e(r) = 1 - 2 * \Phi \left(\frac{k * U}{g(r)} \right) + 2 * \sqrt{\pi} * p_1 * \frac{r}{\sqrt{A}} , \quad (2)$$

where e - the RSVE , $\Phi(x)$ - Gauss function, $k, U, A, g(r)$ - the OI contrast, amplitude, square and noise respectively :

$$g(r) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(\bar{\omega}) H(\bar{\omega}, r) d\bar{\omega} ,$$

$\bar{\omega}$ - a space harmonic vector, $G(\bar{\omega})$ - a spatial spectrum of a noise capacity density, $H(\bar{\omega}, r)$ - a spatial transfer function of the OES , r - a parameter which defines a high space harmonic depression (a radius of an optical point spread plot or a photo-sensitivite element dimension of a matrix photodetector).

The minimization of the RSVE (2) can be the suitable criteria for the OES optimization. The RSVE is defined by the SNR that reflects corresponding an optical radiation energy of the object to the OES detectivity and p_1 that is the ISF.

The image recognition program has to use the detailed information about the OI. The following sequences of image processing procedures allow to do it :

1. Depressing a noise using a sigma filter which parameters are fixed according to etalon OI amplitudes and dimensions.
2. Building a quadro-tree of the image to find thresholds for segmentation and choosing the homogenous zones that can contain the OI.
3. Threshold segmentation.
4. Each homogenous zone divides onto connected zones by a contour search algorithm. If a connected zone has a square and a length of its contour that are near to the etalon OI ones the ISF vector is calculated.
5. A classification does by comparing a weighted distance between the ISF vectors of the connected zones and the etalon OI with the threshold value. The order of comparing is defined by probabilities of the etalon OI appearances.

The ISF vector, the method of the OES optimization and the order of the image processing procedures have been applying for automatic recognition of *TRICHINELLA* microbe by the special OES. *TRICHINELLA* microbe causes the dangerous incurable disease of people and cattle so agricultural and sanitary organizations are interested in this OES. Now the *TRICHINELLA* recognition procedures are testing with the OES model.

◆ The well-known criteria for the OES optimization is a maximization of a signal to noise ratio (SNR) that is provided by a corrective spatial filtering. But using the ISF makes this optimization difficult because the OI may have various scales and rotations and they might be among bright other images that reduce spatial filter possibilities. The computer simulation of the OES shows that the OI shape distortion may be characterized by the relative square value error (RSVE). The RSVE

Метод класифікації опуклих однозв'язних областей

РОСТИСЛАВ КОСАРЕВИЧ, ЛЕОНІД КОРОЛЮК, БОГДАН РУСИН

Фізико-механічний інститут НАН

Україна, 290601 Львів,
вул. Наукова, 5а,
Тел.: (0322) 65-4530.

Rostyslav Kosarevych, Leonid Koroliuk, Bohdan Rusyn. Classification of connected convexity areas.

The algorithm of description and classification of connected convexity areas is considered. The object recognition criterions built basing on their internal structure analysis are suggested.

Вступ. Існує велика кількість задач аналізу зображень, де виникає необхідність опису і розпізнавання об'єктів. Важливою проблемою в таких задачах є те, що вихідні зображення суттєво відрізняються по своїй структурі, підлягають перетворенням зсуву, повороту, зміни масштабу, а також впливу завад і спотворень. Це обумовлює велику кількість різноманітних методів розпізнавання.

Але незалежно від методу, що застосовується для формування ознак, простір вхідних даних необхідно перетворити в простір ознак таким чином, щоб, по-перше, зробити стиснення інформації як при описі образів, так і при формалізації процедури розпізнавання; по-друге, відібрані ознаки для кожного класу зображень повинні утворювати неперетинні множини; по-третє, ознаки повинні бути інваріантними відносно афінних перетворень [1].

Тому в роботі розглянута одна з процедур формування ознак. Ця процедура оснований на використанні інформації про внутрішню структуру об'єкта, шляхом встановлення числової оцінки радіальної апроксимації [2,4]. Під радіальною апроксимацією будемо розуміти проведення променів з центру площі об'єкта до перетину з його контуром.

В якості моделі зображення об'єкта розглядатимемо область - множини точок скінченного порядку.

Оскільки опукла однозв'язна область є найпростішим випадком області, що зустрічається, то довільну область можна розбити на сукупність опуклих однозв'язних областей і створити опис для кожної з них. Тим самим отримаємо опис для області в загальному випадку, неопуклої та неоднорозв'язної.

Побудова опису. Розглянемо побудову опису однозв'язної і опуклої області (див. рис. 1). Побудова її опису полягає у визначенні довжини l_i проведених променів та фіксації кута f , під яким вони проведені. Всього будується n променів з інтервалом Df . В результаті отримаємо послідовність:

$$L = \{l_i, \varphi_i\}_{i=1}^n \quad (1)$$

Такий опис дає певну апроксимацію контура області, а контур, як відомо є однозначним представленням області. Беручи до уваги похибки дискретизації і обчислень величин l_i при афінних перетвореннях області, було встановлено, що за критерієм співвідношення площ $h = S_n / S$ (S_n після i S до апроксимації) найбільш оптимальним числом проведених променів є $n_{\text{опт}} = 72$, при якому похибка відтвореного контуру не перевищує 5% (див. рис. 2), а інтервал фіксації кута при цьому становитиме 5 градусів.

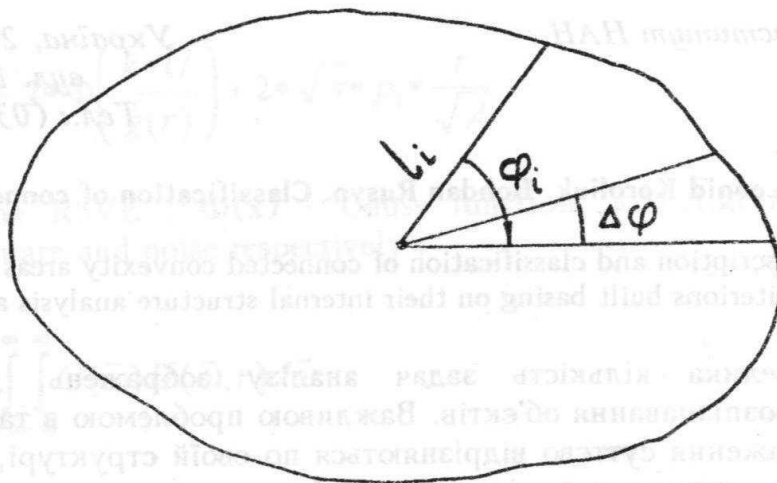


Рис. 1

Для зручності аналізу опис (1) представимо у вигляді об'єднання двох множин

$$L = L_l \cup L_\varphi = \{l_i\}_{i=1}^n \cup \{\varphi_i\}_{i=1}^n, \quad (2)$$

Отриманий опис є інваріантним відносно перетворення переміщення, але не володіє інваріантними властивостями відносно перетворень масштабу та повороту.

Класифікація. Розглянемо M класів. Нехай ці класи допускають представлення за допомогою еталонних образів Z_1, Z_2, \dots, Z_M , причому кожному з них можна поставити у відповідність множини

$$L^{Z_j} = L_l^{Z_j} \cup L_\varphi^{Z_j} = \{l_i\}_{i=1}^n \cup \{\varphi_i\}_{i=1}^n, \quad j = 1, \dots, M. \quad (3)$$

Візьмемо для прикладу деякий клас $Z_t \in M$. Класифікацію образів проведемо в два етапи. На першому етапі будемо використовувати тільки множини L_l та $L_l^{Z_j}$.

У випадку, коли має місце перетворення масштабу області, розглянемо множину точок $\{x_i, y_i\}$ на площині, де $x_i = i$; $y_i = l_i / l_i^{Z_j}$. Вважаючи точки $\{x_1, y_1\}, \{x_2, y_2\}, \dots, \{x_n, y_n\}$ як результат з двовимірного розподілу, запишемо коефіцієнт кореляції розподілу

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 (y_i - \bar{y})^2}} \quad (4)$$

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i, \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i.$$

Відомо, що $r \in [-1, 1]$. Граничні значення $r = \pm 1$ можуть досягатися лише тоді, коли всі точки $\{x_i, y_i\}$ розташовані на одній прямій в площині X, Y .

Точки будуть розташовані на прямій у випадку, коли величини y_i рівні, а це можливо лише тоді, коли елемент l_i буде відрізнятися від $l_i^{Z_j}$ на деякий множник. Отже, для попереднього віднесення області з описом L до класу Z_t необхідне використання критерія максимуму коефіцієнта кореляції r , тобто величини (4).

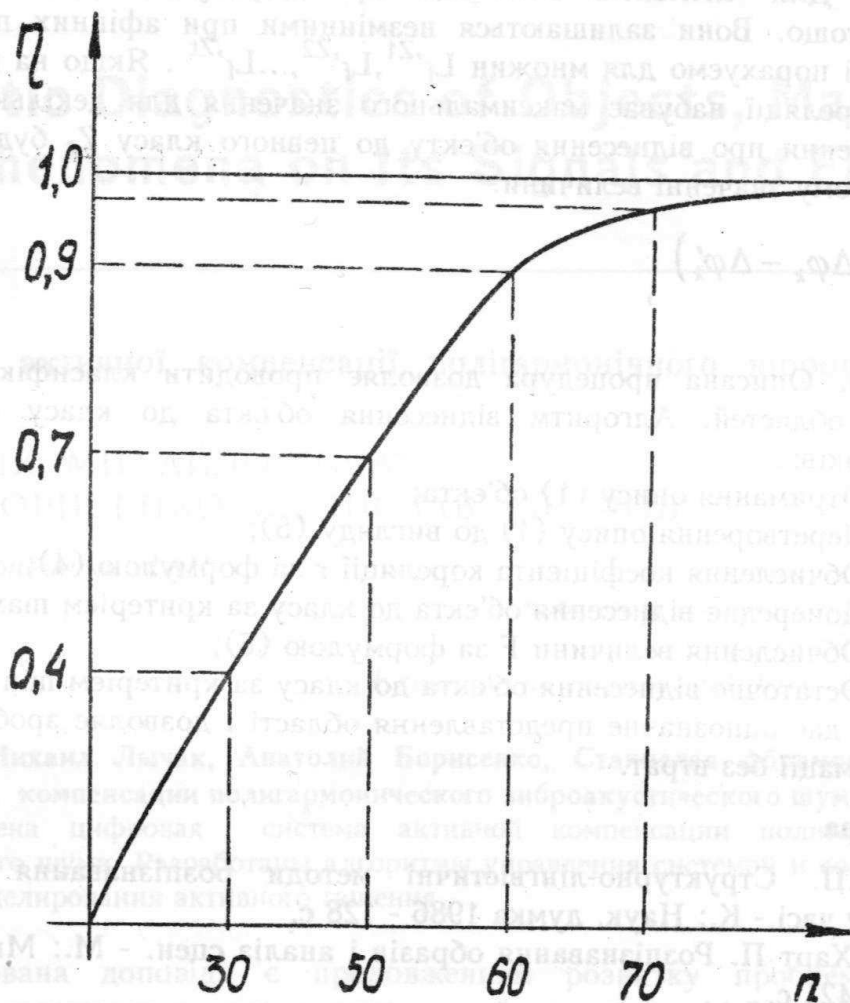


Рис. 2

Розглянемо випадок, коли має місце перетворення повороту. В цьому разі опис L деякої області і опис цієї ж області, але повернутої на деякий кут, відрізнятимуться лише порядком елементів l_i . Для досягнення інваріантного опису до повороту області в роботі пропонується здійснювати операцію сортування елементів множини L_1 в порядку їх зростання. При цьому порядок елементів множини L_f зміниться. В результаті отримуємо множини

$$\begin{aligned} L_1' &= \{l_i'\} \quad l_i < l_{i+1} \quad i=1, \dots, n, \\ L_f' &= \{f_i'\} \quad i=1, \dots, n. \end{aligned} \quad (5)$$

Аналогічно (5) проведемо впорядкування описів представників класів Z_1, Z_2, \dots, Z_M . При сортуванні зберігається величина кута, під яким був проведений промінь, тому завжди є можливість відтворити область.

Проте після сортування може статися так, що опис представників різних класів буде однаковим, тобто коефіцієнт кореляції набуватиме максимального значення для двох або більше класів. Для усунення цього недоліку необхідно проводити аналіз множин L_f, L_f' . Відомо, що величина кута між максимальним і мінімальним променями не змінюється при перетвореннях масштабу, повороту і переміщення. Для елементів множини L_f' , порахуємо різниці $Df_1=(f_1-f_n)$, $Df_2=(f_2-f_{n-1})$ тощо. Вони залишаються незмінними при афінних перетвореннях. Такі ж різниці порахуємо для множин $L_f^{Z_1}, L_f^{Z_2}, \dots, L_f^{Z_t}$. Якщо на першому етапі коефіцієнт кореляції набуває максимального значення для декількох класів, то остаточне рішення про віднесення об'єкту до певного класу Z_t будемо приймати при мініальному значенні величини:

$$\Phi = \sum_{k=1}^{\lfloor n/2 \rfloor} (\Delta\varphi_k - \Delta\varphi'_k) \quad (6)$$

Висновки. Описана процедура дозволяє проводити класифікацію опуклих однозв'язних областей. Алгоритм віднесення об'єкта до класу складається з наступних кроків:

- Крок 1. Отримання опису (1) об'єкта;
- Крок 2. Перетворення опису (1) до вигляду (5);
- Крок 3. Обчислення коефіцієнта кореляції r за формулою (4);
- Крок 4. Попереднє віднесення об'єкта до класу за критерієм $\max r$;
- Крок 5. Обчислення величини F за формулою (6);
- Крок 6. Остаточне віднесення об'єкта до класу за критерієм $\min F$.

Опис (5) дає однозначне представлення області і дозволяє зробити стиснення вхідної інформації без втрат.

Література

1. Русин Б.П. Структурно-лінгвістичні методи розпізнавання зображень в реальному часі. - К.: Наук. думка 1986 - 128 с.
2. Дуда Р., Харт П. Розпізнавання образів і аналіз сцен. - М.: Мїр, 1976. - 512 с. 1983. - 421 с.
3. Крамер Г. Математичні методи статистики. - М.: Державне видавництво іноземної літератури, 1948. - 631 с.
4. Павлідіс Т. Алгоритми машинної графіки і обробки зображень. - М. Радіо і зв'язок. - 1986. - 400с.