IMAGE STRUCTURE ANALYSIS BY 3-STAGES CLUSTERING

Roman Melnyk 1), Ruslan Tushnytskyy 2)

1) Professor, Software Department, Lviv Polytechnic National University, 12, S. Bandery str., Lviv, 79013, ramelnyk@polynet.lviv.ua
2) Post-graduate student, Software Department, Lviv Polytechnic National University, 12, S. Bandery str., Lviv, 79013, ruslan.tushnytskyy@gmail.com

Resume: An approach for decomposition of visual images by clustering and breaking them down into geometric figures is considered. Multilevel hierarchical clustering algorithm to form three emphasized levels of clusters such as rectangles, closed regions and integrated areas is proposed. Advantages of such decomposition in three stages are as follows: images covered by rectangles are planned to be formatted and compressed, image fragments could be taken for the preliminary pattern recognition or could easily be corrected, hierarchically constructed fragments are good material to form pattern features for searching procedures. The algorithm complexity, the proposed approach of scanning searching area to reduce it, the rolling up criteria and key parameters for its control are investigated. The results of pattern analysis by structure features for some practical problems are presented in the article.

Keywords: cluster, clustering, visual pattern, hierarchical tree, rolling-up algorithm, scanning area, rectangles, integrated areas, structured images, structural features.

1. INTRODUCTION

The clustering methods are widespread for visual patterns segmentation, data mining, indexing and searching. [1−9]. In particular, the work [1] contains the classification of the clustering methods and the approach for contouring chosen clusters. The works [2], [3] describe graph’s models for images and clustering algorithms for its segmentation. The works [4], [5] are devoted to the clustering approach for experimental data covered by a grid with predetermined step. Hierarchical methods were used for decomposition and enhancement of medical images and fingerprints [6] – [7]. Various approaches are used for improving and decomposition of images, among them being assessment of image fragment local characteristics [8], forming regions of special figures [6] or clustering according to criteria of statistical dependencies [7]. These works, as well as a number of others, do not consider the variety of control parameters, by which researcher could investigate different dependences of clustering results, e.g., such as brightness threshold and intensity sensitivity for number of clusters to be formed. As a rule they are based on one stage clustering procedures.

Different agglomerative methods differ by the way that similarity between two clusters is updated. If the linkage satisfies a cluster aggregative inequality then the algorithm can be implemented efficiently at a time complexity of $O(N^2)$ [10]. A new recent method [11] uses a sophisticated merging criterion which takes the cluster internal structure into account. The multilevel hierarchical clustering algorithm in [12] are mainly planned for data analysis, has one stage button-up procedure and proposes new linkage criteria to reduce the computation complexity from $O(N^3)$ to $O(N^2)$.

Standard agglomerative clustering methods use ordinary similarity distances between clusters to be merged. The proposed approach is based on a number of merging criterion: value of common border between clusters, difference between their weighted brightness, difference between square values, etc. Moreover, merging is being performed if constraints allow it. The work is based on three stages clustering algorithm: covering an image by rectangles, closed regions and integrated areas. Constraints are formed to control form of rectangles or brightness of closed regions and integrated areas.

Some investigations were held to describe different dependences: number of clusters, parameters of closed regions and integrated areas due to criteria for merging: cluster brightness threshold, algorithm sensitivity or shape and dimension constraints.

The full procedure allows to formalize visual image description by breaking them down into integrated areas, closed regions or rectangles [Fig.1].
An access to output fragments of every stage allows them to be corrected and broken into fragments. We would like to notice that we didn’t find such image fragments classification in the referenced papers and others read. Additionally we note that Microsoft Visual Studio.NET disposes a class Region to paint figures with non regular shape.

When applying three stages clustering algorithm to practical images algorithmic complexity and computation time of image decomposition process would be taken into account. For this reason, scanning searching procedures on different stages of the algorithm are developed.

For large database with over tens of thousands of visual patterns, effective indexing becomes an important issue the content-based image retrieval systems (CBIR). The existent universal CBIR systems attribute to one of three categories depending on approach of extracting features, for example color [15 – 16], shape [17 – 18], structure and location [19 – 20]. Presented 3-stages clustering algorithm extracts some pattern structural properties (pattern features) which were taken to analyze and classify images.

2. TASK OF IMAGE DECOMPOSITION

The tasks of the decomposition algorithm when being applied to visual patterns are to solve the following problems: 1) to get full description for all parts of image with its all possible characteristics; 2) to ensure an access for every part of pattern independently of its position on three levels hierarchical tree: integrated areas → closed regions → rectangles. Such access is planned to be used in searching procedures for pattern and its fragment recognition.

![Fig. 1 – Fragment of multilevel hierarchical tree](image)

To solve these complex tasks the multilevel three stages clustering algorithm based on traditional binary hierarchical tree is developed. Three stages of the process, the nodes of interest and a small fragment of the multilevel tree are shown on Fig. 1. Three stages are emphasized because they have different searching criteria to build different geometric figures. For user it is more interesting to consider three levels tree extracted from full tree presented on Fig. 1 with earlier defined pattern fragments (areas, regions and rectangles) as nodes.

Before the clustering algorithm begins its work the preliminary procedure to prepare leafs of the hierarchical tree has to start. Non divisible microobjects, i.e., pixels of digital image must be united to form microclusters – smallest non divisible rectangles. The microclusters are fully or partly filled [Fig. 2(a)] in the case of black-and-white image or are combined from pixels of different brightness for grey image [Fig. 2 (b)].

![Fig. 2 – Black-and-white and grey clusters](image)

On the preliminary stage we must to solve the problem:

It is necessary to obtain a partitioning \( O \) of the set of microobjects \( O=\{o_1, o_2, \ldots , o_n\} \):

\[
\overline{O} = \{O_1, O_2, \ldots , O_n\}, \quad n \rightarrow \min,
\]

so that a total number of partitions is minimized while satisfying the constraints:

\[
F_k^- \leq F_k (X_i) \leq F_k^+, \quad (k=1,m; i=1,n),
\]

where \( n \) is number of microclusters \( O_1, O_2, \ldots , O_n \); \( F_k^-, F_k^+ \) – are boundary values of the \( k \)-th feature function. The functions \( F_1, F_2, \ldots , F_n \) define coordinates, brightness, dimension, number of pixels, shape, color, ratio between filled and empty cells etc.

The cluster features or parts of them are also assigned for clusters on the higher levels of the hierarchical tree. So to make pattern decomposition by rectangles, closed regions and integrated areas we must to solve the problem (1) three times. We solve the optimization problem by the rolling-up procedure, i.e., algorithm of bottom-up merging of clusters to get minimal number of segments covering a pattern. The stages differ between themselves by merging criterion and constraints. The rolling-up procedure divides a set of objects into groups having the same or similar characteristics. Tree nodes not satisfying the constraints and subsequently not being the subjects for further rolling-up process are transferred to the result set of clusters.

The visual pattern area is being covered by grid with steps selected from a set of values: 1×1, 1×2, 2×1, 2×2 etc. Some scanning procedures to solve the
problem (1) on the preliminary stage of the algorithm and to build microclusters as rectangles sized by selected steps of grid are developed. A cluster number on higher stages depends from a microcluster number and its features. That is why we developed alternative algorithms for scanning procedures on the preliminary stage of the algorithm. The trivial approach to get microclusters is to describe a grid cell as microcluster, i.e., to define for it all features and to assign them to it. For reason of a lack of paper place other preliminary scanning algorithms are not considered in the paper.

3. CONTROL PARAMETERS

For grey image every microobject (pixel), is characterized by brightness value ranging from black to white. Brightness \( b \) of microobject we define by equation expressing its relative filling as percentage of black color:

\[
b = (256 - c_i) \times 100 / 256,
\]

or as percentage of white color:

\[
b = c_i \times 100 / 256
\]

where \( c_i (i = 1,2,3) \) is a value of one from the \( R \) (\( G \) or \( B \)) components of grey pixel.

Fig. 2(b) shows rectangles by different brightness values.

The equations (3), (4) are applicable only for microobjects. To denote integral microcluster brightness we use \( B_{rk} \). We define it as a mean weighted value for two microobjects (microclusters) indexed by \( i \) and \( j \) and merging to a microcluster numbered by \( k \):

\[
B_{rk} = \frac{b_i \times a_{i1} \times a_{i2} + b_j \times a_{j1} \times a_{j2}}{a_{i1} \times a_{i2} + a_{j1} \times a_{j2}},
\]

where \( b_i, b_j \) are the brightness values, \( a_{i1}, a_{i2}, a_{j1}, a_{j2} \) are values of microobject geometric sizes.

The equations (1) – (5) are being used on the preliminary and 1st stages of the algorithm where microcluster and cluster as rectangles are considered. Fragment of medical pattern clustered by rectangles is illustrated by Fig. 3.

To denote integral closed region and integrated area brightness we use \( B_{Rk} \) and \( B_{Ak} \). We define them as a mean value of cluster or closed region brightness of all \( n \) objects creating closed region or integrated area:

\[
B_{Rk} = 1/n \times \sum b_i; \quad B_{Ak} = 1/n \times \sum B_{Rk}
\]

4. ALGORITHM ACCURACY

For black-and-white image there are two types of microcluster (rectangle): completely black and with a number of white positions.

Ratio between a number of black pixels \( n_{bi} \) entering \( i^{th} \) microcluster and its total number of pixels \( n_{ti} \) we use as \( i^{th} \) cluster brightness: \( r_{bi} = n_{bi} / n_{ti} \). For example, Fig. 2(a) shows for microcluster brightness values 1, \( \frac{1}{3} \), \( \frac{1}{2} \).

We note: \( S_P \) – area of pattern (number of black pixels), \( S_{ol} \) – area of cluster, \( S_{Rc} \) – area of the pattern got as coverage by rectangles: \( S_{Rc} = \sum S_{ol} \), where \( i \) runs over full set of clusters.

The brightness parameter of rectangle we use to control accurate or approximation clustering results. For accurate clustering we have \( S_{Rc} = S_P \).

In practice we have to reduce extra or noise information and then it is useful to consider the case \( S_{Rc} < S_P \). We call this procedure cleaning or reduction. This process is similar to filtration of thin effects in visual patterns.

Opposite case is if \( S_{Rc} > S_P \). This procedure is named as renewal or addition. We mean here, that by a noise in the pattern some needed pixels are destroyed or lost and its picture does not correspond to generally accepted one.

The brightness parameter is a key to control ratio between \( S_{Rc} \) and \( S_P \), i.e., to control: 1) a numbers of
microclusters and clusters not taking part in merging of the rolling-up process; 2) a number of white pixels included in the clusters covering the pattern; 3) numbers of clusters and closed regions got by pattern clustering.

We consider accuracy for image covered by clusters or covered by integrated areas:

$$E_C = \frac{S_{P_{100}}}{S_{P}}$$

$$E_{IA} = \frac{IA_{b}}{IA_{100}}.$$  

(7)

where $IA_b$ – number of integrated areas, got by clustering with user defined brightness parameter $b$, $IA_{100}$ – number of integrated areas, got by clustering with brightness parameter $b = 100\%$.

5. DECOMPOSITION ALGORITHM

To solve the problem (1) we consider the rolling-up algorithm having input objects as microclusters, clusters and closed regions and output objects as clusters, closed regions and integrated areas. To use one description for all these object we call them as Clusters. We denote them by $Q_1$, $Q_2$, .... The set $X_i$ belongs to the $i$-th level of a tree and contains Clusters. On the $i$-th step of the algorithm merging Clusters create new Clusters by which the set $X_{i+1}$ is being filled.

The sequence of steps in the rolling-up algorithm is as follows:

0. Prepare initial data: $i=1$, the set $X_i$ is being filled by input Clusters; $C_i = X_i$, $B_i = \emptyset$.

1. Form for every $Q_k \in X_i$ a set of touching neighbors within the grid by searching procedure.

2. Calculate the criterion function values for all pairs of adjacent Clusters.

3. For every two candidates to be merged and satisfying all constraints create the list $L(F_k)$ of criterion values by decreasing order.

4. Delete from $L(F_k)$ pairs having intersected indexes beginning to search them from bottom.

5. Merge first pair of Clusters from the list and put new Cluster to the set $B_i$, correct the set $C_i$, form the new set $X_{i+1}$ of Clusters.

6. Increasing $i$ by 1 repeat steps 1–4 until merging takes place.

The last set $X_L$ contains output Clusters

6. ALGORITHM COMPLEXITY OPTIMIZATION

Computation complexity of the clustering algorithm could be determined by a number of criterion function calculation, comparison and merging number. In the worst case, when merging number is $2N-1$ ($N$ is full number of input Clusters) complexity of clustering algorithm exceeds $O(N^2)$. Theoretically to calculate all functions (needed and extra) it does cost $O(N^3)$. The main operations of the algorithm are related with searching of candidates for merging.

To reduce the algorithm complexity we could mainly at the stage of searching Cluster candidates to be merged. The searching area for the rolling-up procedure we plan as a part of full pattern space. This part is as a rectangle moving through the pattern rectangle by alternative rules. It scans the whole surface of image [see Fig. 7]. Area limits for searching Clusters – candidates to be merged – are put on one axe of coordinates. Decreasing input data volume we reduce the algorithm complexity from value $O(N^2)$ to a set of smaller values $O(n_i^2)$, because the number $n_i$ of Clusters in scanning area is smaller than the number $N$ in full pattern area, i.e., $n_i < N$. We would like to mention that searching procedure repeats its work step by step covering the whole pattern area by a sequence of smaller scanning areas.

All Clusters of two coordinates pattern space from Fig. 7 are stored in linear data structure containing description of the Cluster model and a pointer for neighboring Cluster [Fig. 8].

![Fig.5 – Scanning area of the algorithm](image-url)

First rule for scanning is illustrated by Fig. 5(a). The dashed-lined rectangle defines the scanning working area restricted by the right border. Scanning to search for neighboring Clusters is from left to right within the rectangle. After some searching and merging steps the rectangle is being moved to right.

Fig. 5(b) illustrates the second rule of scanning. The rectangle is being restricted both by the right and the left borders. An image is being scanned from left to right within the rectangle. To search for neighboring Clusters the rectangle itself has two directions for moving: to left and to right. Cluster for which neighboring Cluster are being searched we accept as the center of the scanning rectangle.

The scanning area has adaptive property: dimension of scanning rectangle could be of image size or could be formed dynamically depending from Cluster size. For example, at the beginning the
scanning area size is being initialized by layout grid steps and further they are being increased step by step. This adaptive property essentially reduces for current Cluster a number of neighbors to be compared.

The experiments presented in the Table 1 confirmed efficiency of scanning area: minimal time economy is by one order and computation advantage is better for bigger images.

Table 1. Clustering algorithm without and with scanning area

<table>
<thead>
<tr>
<th>Image dimension</th>
<th>Cluster number</th>
<th>Time without scanning area, s</th>
<th>Time with scanning area, s</th>
</tr>
</thead>
<tbody>
<tr>
<td>120x178</td>
<td>5 155</td>
<td>0,751</td>
<td>0,07</td>
</tr>
<tr>
<td>149x150</td>
<td>7 311</td>
<td>0,661</td>
<td>0,05</td>
</tr>
<tr>
<td>175x216</td>
<td>8 210</td>
<td>0,450</td>
<td>0,05</td>
</tr>
<tr>
<td>141x169</td>
<td>11 952</td>
<td>1,112</td>
<td>0,07</td>
</tr>
<tr>
<td>156x173</td>
<td>13 617</td>
<td>2,193</td>
<td>0,09</td>
</tr>
<tr>
<td>171x213</td>
<td>14 136</td>
<td>3,074</td>
<td>0,140</td>
</tr>
</tbody>
</table>

7. CLUSTERING TO CLOSED REGION AND INTEGRATED AREAS

By the 2nd and 3rd stages of the algorithm we solve the problem (1) having rectangles \( O(Q_1, Q_2, Q_3, ..., Q_n) \) instead of set \( O \). A task for the rolling-up procedure is to find partitioning \( R = \{R_1, R_2, ..., R_n\} \) and \( A = \{A_1, A_2, ..., A_n\} \) instead of set \( O = \{O_1, O_2, ..., O_n\} \) so that a total number of partitions is minimized: \( n \rightarrow \min \), while satisfying the constraints from (1).

Every set \( R_i \) contains rectangles included to one closed region by the rolling-up procedure. Criterion for rectangles to be merged is a number of common edges [Fig. 6(a)].

Constrains are formulated by feature functions \( F_L \). Second criterion for Cluster rectangle to participate in merging is minimal distance \( d \) in pixels between their closest rectangles [Fig. 6(b)]. Distance \( d \) is measure to be controlled by user.

We extended the distance criterion for “diagonal” and “main” distances [Fig. 7(a)].

In particular, diagonal distance is useful to process images having ring structure, e.g., blood cells, fingerprints etc.

7. CLUSTERING TO CLOSED REGION AND INTEGRATED AREAS

By the 2nd and 3rd stages of the algorithm we solve the problem (1) having rectangles \( O(Q_1, Q_2, Q_3, ..., Q_n) \) and closed regions \( R(R_1, R_2, R_3, ..., R_n) \) instead of set \( O \). A task for the rolling-up procedure is to find partitioning \( R = \{R_1, R_2, ..., R_n\} \) and \( A = \{A_1, A_2, ..., A_n\} \) instead of set \( O = \{O_1, O_2, ..., O_n\} \) so that a total number of partitions is minimized: \( n \rightarrow \min \), while satisfying the constraints from (1).

Every set \( R_i \) contains rectangles included to one closed region by the rolling-up procedure. Criterion for rectangles to be merged is a number of common edges [Fig. 6(a)].

Constrains are formulated by feature functions \( F_L \). Second criterion for Cluster rectangle to participate in merging is minimal distance \( d \) in pixels between their closest rectangles [Fig. 6(b)]. Distance \( d \) is measure to be controlled by user.

We extended the distance criterion for “diagonal” and “main” distances [Fig. 7(a)].

In particular, diagonal distance is useful to process images having ring structure, e.g., blood cells, fingerprints etc.
merging degree by three stages of the algorithm:

\[ K_{s}^{123}(MC) = IA / MC. \]  \hspace{1cm} (10)

To characterize inverse property (fuzziness) we use inverse values for (2-5), e.g.:

\[ K_{f}^{123}(MC) = MC / IA. \]  \hspace{1cm} (11)

The image structure coefficients we also assign to structural features. In some cases to widen the set we can in formulas (4-6) to use a total number of image pixels \( PX \) instead of \( MC \):

\[ K_{s}^{123}(PX) = IA / PX. \]  \hspace{1cm} (12)

Each selected fragment \( i \), in particular, area or region characterize the relative number of pixels:

\[ v_{s}(CR) = px_{s}(CR)/PX, \quad v_{s}(IA) = px_{s}(IA)/PX, \]  \hspace{1cm} (13)

for which inequality is as follows:

\[ \sum px_{s}(CR) \leq PX, \quad \sum px_{s}(IA) \leq PX, \]  \hspace{1cm} (14)

where sum takes for all formed areas \( IA \) or region \( CR \).

Some statistical properties could easily be considered as structural properties. For example:

1) Mean cluster, region or area dimension :

\[ M(CR) = \frac{1}{CR} \cdot \sum px_{s}(CR), \]
\[ M(IA) = \frac{1}{IA} \cdot \sum px_{s}(IA), \]  \hspace{1cm} (15)

2) Cluster, region or area deviation

\[ D(CR) = \sqrt{(1/CR) \cdot \sum (px_{s}(CR) - M(CR))^2}, \]
\[ D(IA) = \sqrt{(1/IA) \cdot \sum (px_{s}(IA) - M(IA))^2} \]  \hspace{1cm} (16)

3) To estimate the image elementary object size/deviation features on different levels of decomposition we propose statistical relations:

\[ R(CR) = \frac{M(CR)}{D(CR)}, \quad R(IA) = \frac{M(IA)}{D(IA)} \]  \hspace{1cm} (17)

And for full image integral statistical relations

\[ R_{i}(CR) = \frac{IA \cdot M(CR)}{D(CR)}, \quad R_{i}(IA) = \frac{IA \cdot M(IA)}{D(IA)} \]  \hspace{1cm} (18)

All or part of above mentioned structural features are planned to be used for pattern and its fragments analysis and classification.

9. IMAGE STRUCTURE AND SURFACE ANALYSIS

The clustering algorithm was applied for material structure and surface analysis by using proposed structure features. Work [21] presents structured and non structured images investigation by the grid step and microcluster brightness parameter. Fig. 8 presents samples with lines by different width and length to imitate changing of structure complexity. Table 2 contains some structure coefficients and features illustrating structural changes especially by values of \( K_{s}^{123}(MC) \) and \( R_{i}(IA) \).

Fig. 9 presents medical pattern structure analysis. The determined structural features of the mattern could be used for its indexing and search in databases as well as for qualitative and quantitative estimation of the pattern given.

<table>
<thead>
<tr>
<th>Structural characteristics</th>
<th>Visual pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
</tr>
<tr>
<td>( K_{s}^{123}(MC) )</td>
<td>0,00102</td>
</tr>
<tr>
<td>( R_{i}(IA) )</td>
<td>2,42940</td>
</tr>
<tr>
<td>( MC )</td>
<td>3912</td>
</tr>
<tr>
<td>( CR )</td>
<td>1038</td>
</tr>
<tr>
<td>( IA )</td>
<td>11</td>
</tr>
<tr>
<td>( C )</td>
<td>4</td>
</tr>
</tbody>
</table>

Fig.8 – Images for structure analysis

Fig.9 – Medical pattern. (a) Input image. (b) Clusters. (c) Closed regions. (d) Integrated areas. (e) Closed regions coloured by their weighted brightness.
The main structural coefficients and features are as follows: $K_{123}^{T}(MC) = 0.00182$, $MC = 36241$, $C = 25384$, $CR = 25272$, $IA = 66$, $M(IA) = 549,106$, $D(IA) = 1330.06$.

By the experiment we can conclude that numbers of final Clusters (integrated areas) are of close values as it was expected.

The algorithm was applied for surface topology of freshly deposited surfaces [Fig.10].

![Fig. 10 – Surface topology of freshly deposited (a) Pt, (b) Au and (c) Ag surfaces. The average grain size is 3 nm for Pt, 8 nm for Au, 15 nm for Ag](image)

By structural features from table 3 we can estimate the surface structure changes caused by different grain size, applied to polish the surface.

**Table 3. Structure characteristics of metal surface**

<table>
<thead>
<tr>
<th>Structural characteristics</th>
<th>Visual pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
</tr>
<tr>
<td>$K_{123}^{T}(MC)$</td>
<td>0.00244</td>
</tr>
<tr>
<td>$R_{1}(IA)$</td>
<td>72,641</td>
</tr>
<tr>
<td>$MC$</td>
<td>21700</td>
</tr>
<tr>
<td>$C$</td>
<td>17498</td>
</tr>
<tr>
<td>$CR$</td>
<td>13973</td>
</tr>
<tr>
<td>$IA$</td>
<td>53</td>
</tr>
<tr>
<td>$M(IA)$</td>
<td>409,433</td>
</tr>
<tr>
<td>$D(IA)$</td>
<td>298,726</td>
</tr>
</tbody>
</table>

Close results were got by investigating the glass surfaces having different sizes of nanobubbles [Fig.11]. Table 4 confirms: the more structure complexity the greater values have structural features.

![Fig. 11 – A series images revealing nanobubbles on a glass surface](image)

![Fig. 12 – Surface images of a Si specimen following lapping with various abrasive types and sizes: (a) 14 – µm BC; (b) – 8 µm BC; (c) – 8 µm SiC](image)

Results in table 5 and others which could be calculated confirmed that they are applicable for quality estimation of polished surfaces.

**Table 4. Structure characteristics for glass surface**

<table>
<thead>
<tr>
<th>Structural characteristics</th>
<th>Visual pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
</tr>
<tr>
<td>$K_{123}^{T}(MC)$</td>
<td>0.00661</td>
</tr>
<tr>
<td>$R_{1}(IA)$</td>
<td>103,87</td>
</tr>
<tr>
<td>$MC$</td>
<td>11183</td>
</tr>
<tr>
<td>$C$</td>
<td>11112</td>
</tr>
<tr>
<td>$CR$</td>
<td>11072</td>
</tr>
<tr>
<td>$IA$</td>
<td>74</td>
</tr>
<tr>
<td>$M(IA)$</td>
<td>151,121</td>
</tr>
<tr>
<td>$D(IA)$</td>
<td>107,653</td>
</tr>
</tbody>
</table>

**Table 5. Structure characteristics for Si surface**

<table>
<thead>
<tr>
<th>Structural characteristics</th>
<th>Visual pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
</tr>
<tr>
<td>$K_{123}^{T}(MC)$</td>
<td>0.00307</td>
</tr>
<tr>
<td>$R_{1}(IA)$</td>
<td>82,589</td>
</tr>
<tr>
<td>$MC$</td>
<td>22119</td>
</tr>
<tr>
<td>$C$</td>
<td>18834</td>
</tr>
<tr>
<td>$CR$</td>
<td>16112</td>
</tr>
<tr>
<td>$IA$</td>
<td>68</td>
</tr>
<tr>
<td>$M(IA)$</td>
<td>325,279</td>
</tr>
<tr>
<td>$D(IA)$</td>
<td>267,818</td>
</tr>
</tbody>
</table>

**10. PROGRAM PACKAGE**

The experimental program package with the user interface has been developed [21]. It controls all operation stages: image input, parameters, intermediate and output images, clustering options (full, specified), report of the algorithm work, result parameters for each clustering stage.

The program packages is programmed in C# language of Microsoft Visual Studio 2008.

**11. CONCLUSION**

The clustering algorithm for visual patterns analysis is developed and investigated. It is based on ideas: 1) classifications of pattern fragments as microclusters, clusters, closed regions and integrated
areas; 2) rolling-up algorithm to solve the problem of optimal coverage on three stages of the algorithm; 3) scanning area approach to reduce the algorithm complexity; 4) calculation of structure features to analyze material structure and surface.

The test results confirmed possibility to apply the program package for image analysis and processing to determine their properties and features, to find structural changes, to search defects in objects and materials. The package could be useful for specialists from medicine, metallurgy, spectroscopy and many others making conclusion by sample images.

All tests were performed by a computer with processor AMD Athlon 1.68 GHz, 768 Mb RAM.

12. REFERENCES


Ruslan Tushnytskyy – a post-graduate student at the Software Engineering Department, Institute of Computer Sciences and Information Technologies, Lviv Polytechnic National University.

Research interests: preliminary processing of images, coding and recognition in real time, clustering algorithms, data compression.