Selection of contour images based on alphabetical representation

To cite this article: V A Lomazov *et al* 2021 *J. Phys.: Conf. Ser.* **2032** 012019

View the article online for updates and enhancements.

Selection of contour images based on alphabetical representation

V A Lomazov[1], N I Korsunov[2], O S Akupiyan[1], V I Lomazova[2], D A Petrosov[3]

[1]Belgorod State Agricultural University named after V. Gorin, 1, Vavilova st., Mayskiy, Belgorod region, 308503, Russia
[2]Belgorod State National Research University, 85, Pobedy st., Belgorod, 308015, Russia
[3]Financial University under the Government of the Russian Federation, 49, Leningradsky ave., Moscow, 125993, Russia

E-mail: vlomazov@yandex.ru

Abstract. The article is devoted to the actual problem of searching for contour images by given reference samples, due to the large amount of graphic scientific and technical information of the Contour type. Existing methods for image search do not allow taking into account the actual identity of images arising from affine transformations (stretching / compression, rotation) resulting from replication, which is important when analyzing graphic documentation. The paper proposes an alphabetical representation of contour images, which made it possible to construct a complex of affine-invariant equivalence relations. The relations built up served as the basis for the development of a set of selection procedures that can be considered as the initial stage of image search. The procedures included in the complex sequentially implement the reduction of the search scope up to a small set of contour images, allowing for the possibility of visual analysis. The complex is programmatically implemented in Python using the CUDA (NVIDIA GPU) software and hardware architecture for parallel computing.

1. Introduction
Automation of graphic data processing based on modern digital technologies is one of the important directions in improving existing methods and tools for analyzing information processing and managing complex systems, increasing the efficiency, reliability and quality of technical systems, which has led to great attention to this problem [1,2]. Contour images are the results of special processing of general-view images (contour highlighting), and are also part of technical and scientific documentation (maps, diagrams, drawings) developed using specialized graphic editors [3,4]. In mechanical engineering, the task of finding digitized drawings and diagrams is especially acute, since the capabilities of existing systems for graphical retrieval of drawings and diagrams (using content-based retrieval [4,5], structure-oriented contour representation [6], sketch-based retrieval of drawings using spatial proximity [7], etc.) are still far from the desired (insufficient degree of compliance of the output with the user's requirements, little flexibility of the means of specifying queries, significant query execution time).

The use of effective search systems for graphic charts and drawings in the preparation of machine-building production is economically justified. For example, [8] provides data showing that if there is no need to re-develop the part (including the development of manufacturing technology), significant savings are achieved. At the same time, about 20% of the parts can be reused without modification, and about 18% can be reused with minor modifications.Таким образом, the purpose of this work is development of a selection procedure (reduction of the search area) of contour images of machine parts, focused on the use of parallel computing technologies, which makes it possible to reduce the execution time of a search query, can be considered as relevant and practically significant.

2. Materials and methods

The methodological basis of the research is the apparatus of the contour analysis, relationship algebra and pattern recognition theory.

As applied to the problem of constructing contours of images of machine parts, the most currently used methods and algorithms for contour selection were considered ([9]): Roberts cross operator, Sobel-Feldman operator, Prewitt operator, Kirsch operator, Robinson operator, Marr-Hildreth operator, Canny edge detector. As a result of the computational work carried out, it is concluded that the Canny algorithm is preferable (if the threshold value is correctly selected), although the Roberts and Sobel algorithms are a good alternative for a highly noisy image.

For contour images of machine parts, the presence of corner points is characteristic. The selection of them is a separate problem, for which the well-known (for example, [10]) corner detectors by Moravec, Harris, Förstner, Shi and Tomasi, as well as SUSAN, COP, FAST were considered as possible alternatives. The choice of the Harris Corner Detector was based on its stability and resistance to noise at a reasonable (although not the best among the options considered) speed.

Corner points divide the contour into segments, for each of which it is proposed to build a local right coordinate system (the abscissa axis passes through the neighboring corner points). The segment of the axis between two corner points is normalized (after which its length becomes equal to 1) and divided into r equal sub-segments. On each of the sub-segments, a part of the segment is approximated by a polynomial of the minimum (for a given error) degree (a polynomial codon), which leads to a piecewise polynomial approximation of the considered segment.

In this case, each of the codons is encoded (according to the degree of the polynomial) by a symbol from the alphabet $K = \langle k_1, k_2,\ldots, k_p \rangle$.

The alphabet K is an ordered set (a character with fewer indices corresponding to a polynomial codon of lesser degree, «less» than a character with a larger index).

Each segment of the contour is represented as an r-digit word (a combination with repetitions) in the alphabet of characters K, which generates a set of all possible words, which is conveniently represented as an alphabet of characters of segments $S = \langle s_1, s_2,\ldots, s_n \rangle$.

The value of r is fixed at the stage of geometric analysis of vectors, and further we will assume this number to be the same for all considered contour images. It is clear that the total number of characters of the alphabet S: $n=|S|=r^m$.

The alphabet S can also be considered as an ordered set. We will assume that the order on S is determined by the lexicographic sorting of its elements using the previously introduced order for the alphabet K.

Thus, any contour is represented as a word of arbitrary length m in the alphabet of symbols of fragments S:

$A= \langle a_1, a_2,\ldots, a_m \rangle$

where $a_i \in \{ s_1, s_2,\ldots, s_n \}$, $i=1,2,\ldots, m$.

The proposed method of alphabetic representation of vectors is invariant with respect to the image scale (due to the normalization of segments between adjacent corner points) and with respect to the rotation of the image (due to the cyclic ordering of words corresponding to segments). However, although only one alphabetic representation is uniquely assigned to each contour , the opposite condition is not true: one alphabetic representation can correspond to several contour s, which

means several images of machine parts. Therefore, based on the proposed alphabetic representation of contour s, it is impossible to build an effective image recognition algorithm, but it can be used to classify images and to reduce the search area to a class containing images represented by the same alphabetic code as the reference image.

An even greater reduction in the search area can be achieved by introducing equivalence relations (and hence the classifications generated by them) on a set of alphabetic representations.

3. Equivalence relations on a set of contour images

The value of m is equal to the number of fragments in the composition of the contour   ,   is   its simplest characteristic and can serve to introduce the concept of equivalence of contour s.:

Two contours *Cont1* and *Cont2* are called α-equivalent (equivalent in the length of the character record) if the cardinalities (the number of elements of the sets) of the character representations *A1*   and *A2* coincide, i.e.

Cont1 α *Cont2*, if $|A1$ $| = |A2$ $|$

It is clear that the introduced relation is indeed an equivalence relation (since the conditions of reflexivity, symmetry, and transitivity are implemented for it), which means that it generates a partition of the *ContIm* contour image plurality into classes (disjoint subsets) with the same number of fragments. We will further assume that all the considered contours have m fragments.

To get rid of the dependence of the symbolic representation of the Contour   on   the   numbering order of the fragments in the Contour   , we assume that the characters of the alphabet F included in the word are sorted according to the specified order, i.e. a_1 $\curlyvee$ a_2 $\curlyvee$...$\curlyvee$ a_m. In the future, we will limit ourselves to considering only normalized (sorted) representations of contours. It is easy to see that when using this approach, different Contours that differ in the order of the fragments will have the same symbolic representation. However, first, one-to-oneness of the correspondence between the contours and their representations was already lost earlier when the contour lines were «replaced» by codons, and second, this is not critical for selection problems. At the same time, it should be noted that each contour   is uniquely associated with its symbolic representation constructed in the specified way in the alphabet of fragment symbols (in the alphabet of codon symbols).

This allows us to introduce another definition of equivalence: two Contours *Cont1* and *Cont2* are called β-equivalent (equivalent in character notation) if their character representations *A1*   and *A2*   coincide, i.e. *Cont1* β *Cont2*, if *A1=A2* .

We show the correctness of the definition by checking the fulfillment of the equivalence conditions:

1) implementation of reflexivity: *Cont* β *Cont* (proved by the unambiguity of constructing a symbolic representation of the contour   );

2) implementation of symmetry: if *Cont1* β *Cont2*, then *Cont2* β *Cont1* (follows from the symmetry of the equality of symbolic representations);

3) implementation of transitivity: if *Cont1* β *Cont2*, *Cont2* β *Cont3*, then *Cont1* β *Cont3* (follows from the transitivity of the equality of character representations). Let's introduce the concept of a pattern of symbolic representation of a contour   as a word.

$$Sh = < s_1, s_2,..., s_n >$$

The symbols included in this word are variable symbols. The symbol s_i means that in the place occupied by this symbol, can be any symbol from the subalphabyte (the set of ambiguity, the set of possible values of the variable s_i) $F_i \subseteq F$, $i = 1,2,..., m$. Then the sets F_i are elements of the powerset of the set F_i, excluding the empty set $\varnothing$.

The concept of a pattern is an extension of the concept of a symbolic representation of a contour   in the alphabet F. In the special case, when all the disambiguation sets F_i corresponding to the pattern are singleton, the pattern is an ordinary symbolic representation of the contour   .   Another special case is the "null" pattern (Sh_0), for which the ambiguity sets of all characters coincide with the alphabet $F_i = F$, i=1,2,..., m. We call the Sh' pattern is more strict than the Sh'' pattern if the set of

character representations corresponding to Sh' is a proper subset of the set of character representations corresponding to Sh'. It's not hard to see that Sh_0 is the least strict pattern. In general, each pattern can be matched with N character representations, where

$$N = \prod_{i=1}^{n} |F_i|$$

Let's define the concept of matching a symbolic representation of a contour to a pattern as follows: the symbolic representation of a contour $A = \langle a_1, a_2, ..., a_m \rangle$ will be called corresponding to the pattern $Sh = \langle s_1, s_2, ..., s_n \rangle$ (denoted by $A \sim Sh$), if there is a set of values of the template variables Sh: $s_1, s_2, ..., s_n$, such that when they are substituted in Sh, we get A.

The concept of a template allows us to introduce another definition of the equivalence of contours: two contours $Cont1$ and $Cont2$ are called γ-equivalent (equivalent in the Sh pattern) if their symbolic representations $A1$ and $A2$ correspond to the same Sh pattern, i.e.:

$Cont1$ γ (Sh) $Cont2$, if $A1 \sim Sh$ and $A2 \sim Sh$

It is clear that the introduced relation of γ-equivalence of contours is really an equivalence relation (since the conditions of reflexivity, symmetry, and transitivity are satisfied for it) and occupies an intermediate position between α- and β-equivalence.

For the introduced concepts of equivalence, the obvious statements are true:

Statement 1. If two contours are β-equivalent, then they are γ-equivalent in any pattern.

Statement 2. If two contours are γ-equivalent according to some pattern, then they are α-equivalent.

The considered equivalence relations are further used for the selection of contour images.

4. Selection-based contour image search procedure

Let's consider in general the procedure for selecting relevant objects (for example, machine parts) based on their images, assuming the reference image of the part as the search image.

In the frame of the procedure we will separate the geometric and alphabetic analysis of images.

Geometric image analysis includes the following steps:

1. Select the outline of the image using the selected method.

2. Selecting the contour segments by finding the corner points using the selected method.

3. Splitting segments into a given number of sub-segments.

4. Approximation of sub-segments by polynomial codons by the chosen method.

5. Building an alphabetical representation of the contour.

The geometric analysis procedure is performed for all images that make up the search area, as well as for the reference image. At the same time, the use of parallel computing technology allows simultaneous analysis of several images.

The procedure for alphabetical image analysis consists of reducing the search area and includes the following steps:

1. Construction of the α-equivalence relation and selection of alphabetic representations of contours that are α-equivalent to the alphabetic representation of the reference contour from the search area.

2. Check the condition for stopping the procedure when the required reduction of the search area is reached.

3. Construction of the initial γ-equivalence relation.

4. Selection from the search area of the subdomain of alphabetic representations of contours, γ-equivalent to the alphabetic representation of the reference contour.

5. Checking the condition for stopping the procedure when the required reduction of the search area is achieved or the constructed γ-equivalence matches the β-equivalence relation.

6. Building a new γ-equivalence relation with a template that is more strict than the template of the previously used γ-equivalence relation. Go to item 6.

The alphabetic analysis procedure has a finite number of steps due to the finite number of γ-equivalence relations (the most strict γ-equivalence is β-equivalence) and also allows for the possibility of implementation using parallel calculations.

Geometric and alphabetic analysis of images (carried out as part of the selection procedure) allows you to reduce the search area, which makes it possible to further visual or based on more accurate (and more time-consuming) methods of pattern recognition to find the desired objects.

5. Software implementation of the selection procedure and computational experiments

The developed procedure is implemented in Python using CUDA parallel computing technology. Computational experiments were conducted on an IBM PC class computer with an NVIDIA GPU running the Ubuntu operating system based on the Linux 4.15 kernel. During experimental studies, a test sample of machine parts contours developed at Purdue University (USA) was used [6], expanded by applying various affine transformations to the images of the original sample. For each of the sample images (using selection procedures based on the relations of α-, β-, γ-equivalence), a decision was made on the relevance/non-relevance of this image to the reference image. Within the framework of the experiments, the following indicators were determined:

- *TP* – proportion of true-positive solutions (the relevant image is recognized as relevant);
- *TN* – proportion of true-negative solutions (irrelevant image is recognized as irrelevant);
- *FP* – proportion of false-positive solutions (irrelevant image is recognized as relevant);
- *FN* – proportion of false-negative solutions (the relevant image is recognized as irrelevant).

As these indicators reflect the proportion of different types of solutions among all possible solutions, they depend on the ratio of relevant and irrelevant images in the sample. Therefore, along with the indicators, the indicators invariant from this ratio were analyzed:

- *TR* – proportion of correctly recognized images among relevant images;
- *TNR* – proportion of correctly recognized images among irrelevant images.

The considered indicators reflect the quality of the selection procedure. From the point of view of using the selection procedure to reduce the search area, the *DR* indicator is practically important – the share of the total number of images excluded from the search area after the selection.

The results of computational experiments on the use of the constructed equivalence relations presented in Table 1 show that the use of a selection procedure based on an increasing number of circuit characteristics (transition from α-equivalence to γ-equivalence, and then to β-equivalence) leads to an increase in *TR* , *TNR*, *DR*, i.e. to improve the quality and efficiency of the procedure, which, however, is associated with an increase in the time of selection. Therefore, it is advisable to preliminarily indicate the requirements for selection in order to limit the use of a simpler version of the procedure. In the case of high requirements, it is advisable to reduce the search area by sequential application of selection procedures based on the use of α-, γ-, β-equivalence.

Table 1. Results of computational experiments

Equivalence relation type	TP	TN	FP	FN	TPR	FPR	DR
α-equivalence	0.13	0.21	0.60	0,06	0.67	0.74	0.27
β-equivalence	0.02	0.79	0.02	0.17	0.11	0.03	0.96
γ-equivalence	0.06	0.65	0.17	0.12	0.33	0.21	0.77

6. Conclusion

The developed alphabetical representation of contour images and the selection procedure built on its basis can significantly reduce the search area and makes possible search of the required images, either visually or based on more accurate (and more laborious) methods of pattern recognition. The main problem that arises when using the proposed approach is the need for a rational selection of the approximation and selection parameters. To solve this problem (in relation to the features of a specific

source of graphic information), it is required to use special (based on the use of expert and / or intelligent technologies) methods of synthesis of procedures (for example, [11, 12]).

References

[1]   Gonzalez R 2018 *Digital Image Processing* (New York: Pearson)

[2]   Klette R 2014 *Concise Computer Vision – An Introduction into Theory and Algorithms* (London: Springer)

[3]   Bhatti N, Hanbury A 2013 *International Journal on Document Analysis and Recognition* **16(4)** 309-329

[4]   Belongie S J, Malik J and Puzicha J 2002 *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24(4)** 509-522

[5]   Fonseca M J, Fonseca M J, Ferreira A and Jorge J A 2005 *International Journal of Computer Applications in Technology* **23(2-4)** 86-100

[6]   Hou S, Suyu Hou, Karthic Ramani 2008 *Computer-Aided Design* **40(1)** 94-108

[7]   Sousa P, Fonseca M J 2010 *Journal of Visual Languages and Computing* **21(2)** 69-80

[8]   Wemmerlöv U, Hyer N L 1989 *International Journal of Production Research* **2l(9)** 1511-1530

[9]   Burger W, Burge M J 2009 *Principles of Digital Image Processing. Core Algorithms* (London: Springer)

[10]   Gauglitz S, Höllerer T, Turk M 2011 *International Journal of Computer Vision* **94(3)** 335-360

[11]   Lomazov V A, Petrosov D A, Dobrunova A I, Lomazova V I and Matorin S I 2016 *International Journal of Applied Engineering Research* **11(3)** 1867-1873

[12]   Petrosov D A, Lomazov V A, , Lomazova V I and Glushak A V 2018 *Journal of Advanced Research in Dynamical and Control Systems* **10(10 Special Issue)** 1840-1846