Method of detection similar elements in textures with irregular structure

Vasyl Tereshchenko
Taras Shevchenko National University of Kyiv
64/13, Volodymyrska, st., Kyiv, Ukraine
vtereshch@gmail.com

Yaroslav Tereshchenko
Taras Shevchenko National University of Kyiv
64/13, Volodymyrska, st., Kyiv, Ukraine
y_ter@ukr.net

ABSTRACT
We consider the problem of repetitive elements recognition with heterogeneous image texture. The complexity of solving problems from this class is connected to obtaining precise solution that is sensitive to image with irregular texture. The accuracy depends on correctness of original element selection from which the searching process will start. To solve the problem we propose new synthetic approach that combines statistical methods and machine learning method that allows obtaining resistant and accurate solution. The results of this algorithm are used for detection of reptile skin structure.

Keywords
Pattern recognition, repetitive elements, heterogeneous image structure, computer vision, machine learning

1. INTRODUCTION
Problems of image texture recognition with inhomogeneous and irregular structure are included to the list of problems that haven't a satisfactory solution in general case, becouse it depends on two factors: the accuracy of the selected basic element from which the searching of similar objects starts, and how algorithm adequately responds to the regularity and high level of noise on input image. Known approaches solve the problem only in the case of regular and uniform structure of texture image. However, for some subclasses of these problems, in which elements of image texture divided into similarity classes, can suggest approaches that give satisfactory results. An example of this class is the task of recognizing image elements of the skin structure of reptiles and fish that are generally have inhomogeneous and irregular structure. There is a problem of identification these animals by photo of their skin and thereby identify the class or type of animal. Existing approaches that solve such problems [Leu96a, Fir11a] can be used for identification only regular and uniform structure of texture for further automatic extrapolation search of these elements from a given basic sample. Solving the problem in the case of irregular and non-uniform grid leads to serious mistakes and not reliable results. In particular, if the algorithms use a method Canny [Zho11a] for edge detection, then because of gradients absence in the boundaries of input image the algorithm can not adequately segment "scales" (Figure 1).

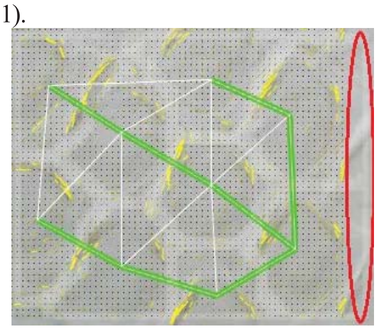


Figure 1. Lack of gradients on the border of the image

Also, existing algorithms [Leu96a –Dud73a] are extremely sensitive to boundaries (gradient changes) in the image, especially if there is additional a lot of noise on it. The extra boundaries may also to appear in case of the defects of animal skin (scars, defects in processing, etc.) and in case of non-uniform background, if the skin does not occupy the whole image (see Figure 2; in this case, a large number of outliers that spoils further work of algorithm).

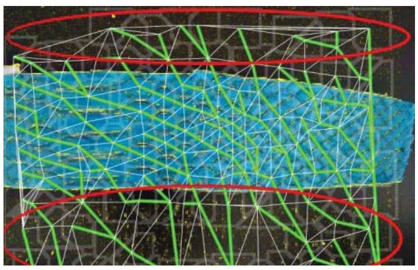


Figure 2. Excessive amounts of noise in the image.

https://doi.org/10.24132/CSRN.2018.2802.1

ISBN 978-80-86943-41-1

Conventional filtering (smoothing image) does not solve the problem, because there is no testing: is there a noise or a part of the searching element. Excessive smoothing can lead to loss important information. Also, each image has its own noise level, and therefore we can not use the same filtering options. In addition, existing algorithms work incorrectly in the case of large differences in the size and form of these elements [Leu96a]. They inadequately ignore scales (Figure 3), or integrate them into large clusters (Figure 4).

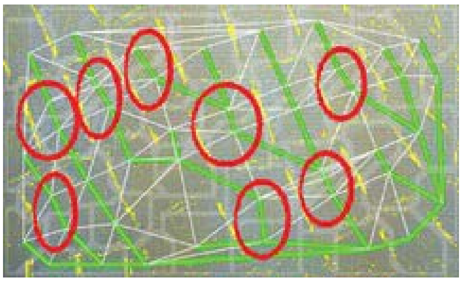


Figure 3. Bonding of elements.

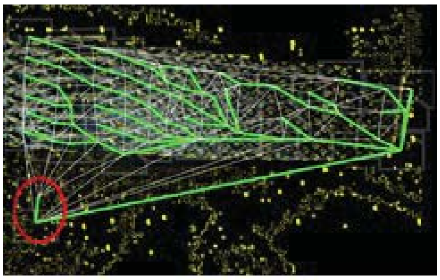


Figure 4. Combining elements of small sizes into large clusters.

There are several approaches to solving this problem. They are usually divided into two types: structural and reference. First approach [Leu96a] is based on separation and analysis of various structural elements and their features, properties that identify the object. Second approach involves comparison of the investigated sample with defined set of templates [Seo10a, Bis11a]. In this paper we propose a new approach to solve the problem of recognition of the similar repeatable elements on image with irregular and heterogeneous structure of texture using the combination of statistical methods and machine learning. This algorithm we used to segment the skin of reptiles and fish scales.

Objective: To develop an efficient algorithm to solve the problem of recognition of similar repeatable elements for non-uniform and irregular structures of image texture.

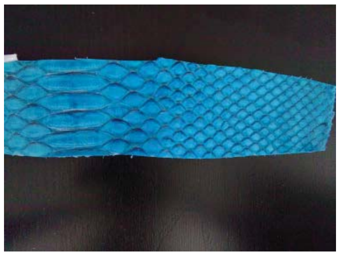
2. METHOD OF DETECTION SIMILAR ELEMENTS

In order to determine similar repeatable elements on inhomogeneous and irregular image texture, we should develop algorithms to solve two main problems: generation initial points and constructing “approximating grid”. The quality of the resulting recognition depends on the efficiency and accuracy of these algorithms. Let's consider methods for solving these problems. Before proceeding to develop these algorithms we will perform pre-processing.

2.1 Preprocessing

Simple image has a lot of unnecessary information that can reduce accuracy of the algorithm. For this, we should implement some preprocessing procedures. It is known that every image has target object (for example, skin of reptile) and background. That is why, to separate them, we can make the following steps (Figure 5):

1. Gaussian blur [Dor14a] with kernel value $(7 \times 7\ , 11 \times 11)$ for 640×480 image.
2. Using k-means clustering [Coa12a] with $k = 2$ execute image binarization.
3. With connected components algorithms, we can detect border of target object [Cha04].



a)

b)

Figure 5. a) Input image; b) separated background

Now we proceed to consider methods of solution mentioned above problems. Let's start from a problem of generation initial points.

2.2 Searching of initial points

In our case, initial point is a centroid of the repeatable element considering its shape. At the

ISBN 978-80-86943-41-1

moment, the most satisfactory shape type is the bounding box. It uses less memory and more flexible for different image processing algorithms. Detection of initial element among various noises (for reptile skin - scratches, pigmentation, high illumination and etc.) isn't a simple problem. That is why, structural methods have some constrains in processing such difficult textures. In our opinion, the most appropriate method - constructing a so-called "probabilistic grid" that can detect elements of texture for generating initial points. The grid involves initialization stage based on machine learning approach (SVM [The09a, Ing08, Vap98a] is used for searching candidates for initial point).

2.2.1 The algorithm of candidates detection for initial element using SVM

1. Generation of training set (each image has its own layout on which negatives and positives are generated). Positive << Negatives
2. Training:
   a) Provided HOG descriptor [Low04, Dal05a];

   b) Normalized on 64 × 64 kernel;

   c) Training with RBF kernel.
3. Recognition:
   a) Pass of the image according to different sizes of sliding windows;
   b) Removing outliers using NMS [Neu06];
   c) Determination of maximum similarity coefficients.

2.2.2 Constructing of "probabilistic grid"

The probabilistic grid works with candidates for initial point, which are generated by SVM. The basic stages of grid generation process are following:

1. Triangulation (Delaunay) of candidates set for the initial point (Figure 6). We can consider *p* initial points that belong to monotone triangulation (set of monotone chain of triangles).

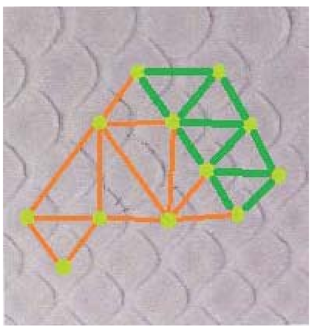


Figure 6. Orange lines–triangulation Delaunay. Green lines – monotone.

2. The selection criterion of initial point: it is the point that has the smallest dispersion of the lengths (Figure 7) of its neighbors and the minimum asymmetry coefficient.

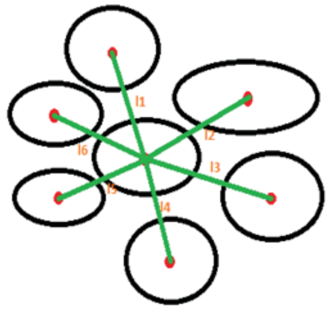


Figure 7. Point neighborhood.

For this, we will use following formulas:

$M_k = \frac{1}{n_k}\sum_{i=1}^{n_k} l_i$ - the mean length of neighbors for point k,

$D_k = \frac{1}{n_k}\sum_{i=1}^{n_k}(l_i - M_k)^2$- dispersion for point k.

Where n_k - the number of neighbors for the current point, l_i- length of i – neighbor.

The asymmetry coefficient is defined as:

$$E_k = \sum_{i=1}^{n_k} \frac{(v_i, v)}{\|v\|} \to min \quad (1)$$

3. After preliminary procedures all points will have own value of dispersion and entropy (2):

$$F_k = (D_k, E_k) \quad (2)$$

Then the initial point is defined as:

$$k_{initial} = \underset{k}{\operatorname{argmax}} \|F_k\|. \quad (3)$$

We can generate *p* best initial points, using points from monotone triangulation.

2.3 Generating of similarity classes

After determining the initial elements we construct classes of similar elements. For this "approximating grid" are generated. In addition, we use LSK-algorithm [Seo10a] to determine the correlation between images.

2.3.1 Feature Extraction

For feature extraction we use LSK/PCA algorithm [Seo10a]. It consists the following stages:

1. *Finding LSK kernel.* The main component of feature extraction algorithm is analysis of details of image based on kernel function. These details are gradients. In our case, the kernel function is:

$$K = \frac{\sqrt{\det(C_l)}}{2\pi h^2} e^{-\frac{(x_l - x)^T C_l (x_l - x)}{2h^2}}, \quad (4)$$

where C_l- covariance matrix for the gradient in 4 directions. Therefore, each kernel will have a matrix of values of smaller dimension function.

ISBN 978-80-86943-41-1

2. *Normalization LSK features and dimension reducing.* We make normalization for reducing the dispersion of kernel function values. Target image is divided into patches (Figure 8).

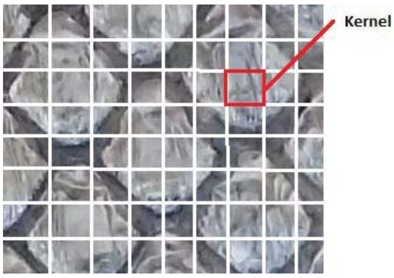


Figure 8. Target decomposition on patches.

Each patch is corresponding to LSK core and normalized by the formula [Seo10a]:

$$W_Q^j = \frac{K_Q^j}{\sum_{l=1}^{P^2} K_Q^j}, l = 1, \dots, P^2, j = 1, \dots, n, \quad (5)$$

where n - number of patches. Reducing the dimension of the features matrix is conducted using PCA algorithm.

3. *Comparison of feature vectors.* We use Frobenius [Seo10a] dot product for comparison of feature vectors, and build resemblance map:

$$\rho_i \equiv \rho(F_Q; F_T) = \sum_{l=1}^{n} \frac{f_Q^{l\,T} f_{T_i}^{l}}{\|F_Q\|_F \|F_{T_i}\|_F} \quad (6)$$

Resemblance map:

$$f(\rho_i) = \frac{\rho_i^2}{1-\rho_i} \quad (7)$$

2.3.2 Constructing of "Approximating grid"

This kind of structure can help to create adaptive algorithm for repeatable objects detection. The main procedure of approach is following:

1. Start from initial point 1 (Figures 9).

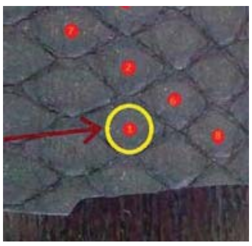


Figure: 9. Location of initial point.

As mentioned in previous paragraph, we can start from multiple initial points. That is why, it is possible to repeat this procedure for every element. It can help to accelerate creation process.

2. For given size of the original element we build probabilistic neighborhood. In this neighborhood, we will look for possible similar elements (Figure 10). The shape of this area can be different. In our case, it is rectangle with sizes that correspond to initial element with empirically estimated scale coefficient.

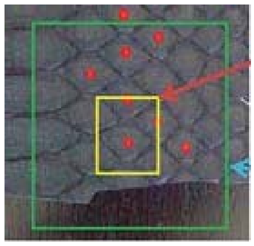


Figure 10. Yellow rectangle – initial template, green rectangle – neighborhood.

3. Changing the size of the pattern, we build correlation map using LSK kernel. For each template we determine the greatest value of correlation function relative to its various proportions (Figure 11.left). If correlated element is located on border of searching area, we extend resemblance map.
4. Find the new center (Figure 11.right).

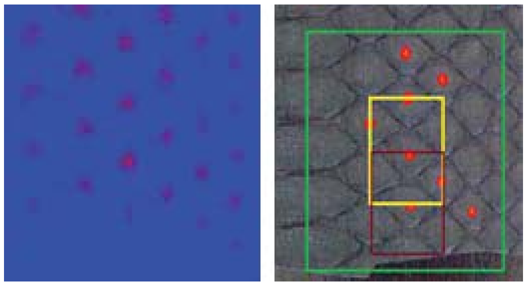


Figure 11. Left- resemblance map; right- new center.

Also, we can generate multiple centers. To merge them, we make clustering by peaks location in resemblance image map. For example, we can use algorithm FOREL [Zag86a] for any number of clusters or k-means for k nearest neighbors. In our case, FOREL is more flexible. Centers of these clusters are new initial points (Figure 12, Figure 13).

5. For created points we repeat steps 1 - 4 and consider new search neighborhood.

After all procedures, we can make the final Delaunay Triangulation for the new set of points and generate grid for whole image texture.

 ISBN 978-80-86943-41-1

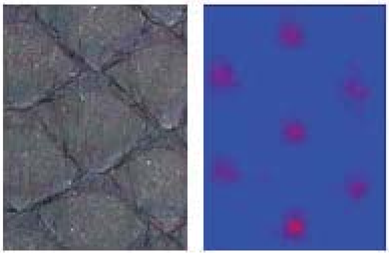


Figure 12. left - input area; right - resemblance map.

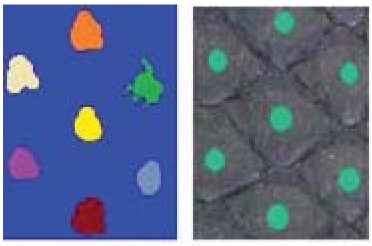


Figure 13. left - FOREL clusters; right- generated points.

3. IMPLEMENTATION

The algorithm is implemented in C ++ using libraries: opencv, libcvm. It was used to the image texture of snake and crocodile skin, Figure 14, Figure 15.

Figure 14. Sample of image fragment.

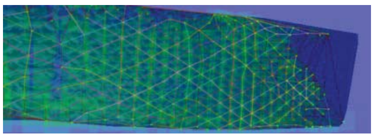


Figure 15. Approximation grid for similar object detection.

Quality of the approach was tested on own database. Using our dataset, we generate precision-recall curve for different initialization methods. In Figure 16. we see that detection algorithm with SVM and “probabilistic grid” has better PR rate than algorithm with only SVM initialization.

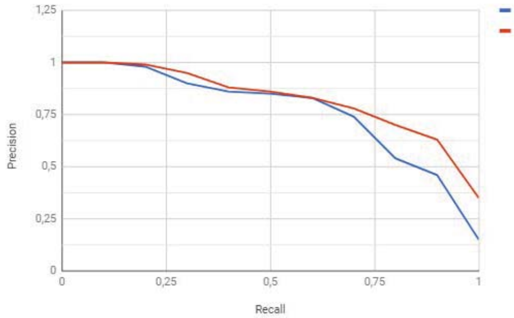


Figure 16. Precision-recall curve. Blue – only SVM, Red – SVM and “probabilistic grid”.

The speed of the algorithm depends on number of repeatable elements. For example, image from Figure 14 takes 2.51 sec of processing time on mobile device Samsung Galaxy S6. Figure 17 shows dependence between time and number of elements.

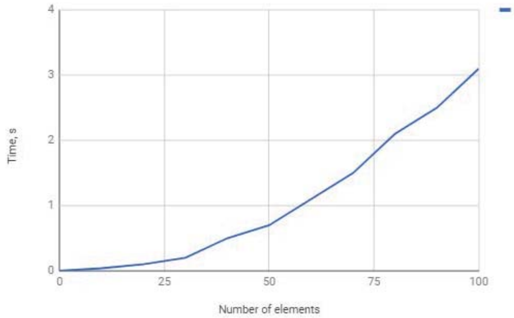


Figure 17. Time dependence.

4. CONCLUSION

We proposed a new approach to solve the problem of finding repeatable objects on the image in case of non-uniform and irregular structure of texture. According the approach we split the problem into two subtasks: search initial element and generation of similar objects. To solve the first subtask we proposed a statistical method that uses a “probabilistic grid” and support vector machine (SVM). To solve the second problem we proposed a new data structure - “approximating grid”. Constructing of the grid includes searching of correlation map, which must specify certain characteristics of the target image. Features were generated using LSK descriptor with PCA algorithm that reduces the dimension of vector space. Finding the grid is an iterative process in which each new point generates searching neighborhood. The results of the algorithms we used to develop software for reptile skin segmentation based on texture structure (set of scales). The algorithm accurately finds centers of

ISBN 978-80-86943-41-1

scales and separates them, even when the boundary between the elements is invisible for the human eye.

REFERENCES

[Leu96] Leung, T. , Malik, J. (April 1996). Detecting, localizing and grouping repeated scene elements from image. Fourth Euro. Conf. on Computer Vision, Cambridge, pp. 22-24, 1996.

[Fir11a] Firouzi, H., Najjaran, H. Detection and Tracking of Multiple Similar Objects Based on Color-Pattern. Lecture Notes in Computer Science: Autonomous and Intelligent Systems, pp. Vol. 6752, pp. 273-283, 2011.

[Zho11a] Zhou, P., Ye, W., Wang, Q. An Improved Canny Algorithm for Edge Detection. Journal of Computational Information Systems, 7(5), pp. 1516-1523, 2011.

[Vap74] Vapnik, V., Chervonenkis, A.Y., Theory of Pattern Recognition (in Russian). Nauka, Moskow , 1974.

[Seo10a] Seo, H J., Milanfar P. Training-Free, Generic Object Detection Using Locally Adaptive Regression Kernels . *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32(9), pp. 1688-1704, 2010.

[Bis11a] Bishop, C. M. Pattern Recognition and Machine Learning. Series: Information Science and Statistics, 2011.

[Dud73a] Duda, R.O., Hart, P. Pattern Classification and Scene Analysis. New York: Wiley International, 2011.

[Dor14a] Dorga, A. and Bhalla, P. Image Sharpening By Gaussian And Butterworth High Pass Filter. Biomedical & Pharmacology Journal, Vol.7(2), pp. 707-713, 2014 .

[Coa12a] Coates, A., Ng, A. Y. Learning Feature Representations with K-means . Neural Networks: Tricks of the Trade, Springer LNCS, Vol. 7700, pp. 561-580, 2012.

[Cha04a] Chang F., Chen, C.J., Lu, C.J. A Linear-Time Component-Labeling Algorithm Using Contour Tracing Technique. Computer Vision and Image Understanding, Vol. 93, No.2, pp. 206-220, 2004.

[The09a] Theodoridis, S., Koutroumbas, K.. Pattern Recognition . Academic Press, 2009.

[Ing08] Ingo, S., Andreas, C. Support Vector Machines. New York: Springer-Verlag, 2008.

[Vap98] Vapnik, V.N., Statistical Learning Theory. John Wiley, NY, 1998.

[Low04a] Lowe, D. G. Distinctive image features from scale-invariant keypoints. *IJCV*, Vol. 60(2), pp. 91–110, 2004.

[Dal05a] Dalal, N. and Triggs, B. Histograms of Oriented Gradients for Human Detection. Procedings *of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. San Diego, CA, USA, pp. 886-893, 2005.

[Neu06] Neubeck, A., Van Gool, L. Efficient Non-Maximum Suppression. Proceedings of 18th International Conference on Pattern Recognition, Hong Kong, pp. 850-855, 2006.

[Zag86a] Zagoruiko, N.G., Elkina, V.N., Lbov, G.S., Emelianov, S.V. OTEKS applied software. Moscow: «Finances and statistics», 1986.

 ISBN 978-80-86943-41-1