Real Time Method for Image Recognition and Categorization

Ivan Božić and Ivan Lazić

Abstract—**In this paper we propose a method for digital image recognition and classification in real-time applications. The paper describes this method in both ways, giving a theoretical background and our implementation. The method can be useful in different scientific fields, and can be utilized for research, education and professional purposes. In this article we present results obtained by our method in the fields of security services and traffic control. All results were obtained using a stand-alone application developed in MATLAB software package and using C++ program language.**

Index Terms—**Categorization, digital image processing, recognition, real-time application, standalone software.**

I. Introduction

In this stage of industrial development, the biggest question is how to automatize processes in manufacturing, transport, delivery and pricing of products and transport and identification of people. Therefore, the main contemporary research in digital signal processing is concentrated to the development of quality software algorithms and methods for recognition and categorization of different signal types. In the following text we describe a possible solution for image processing.

Through the increase of the processing power of computers, digital image processing has been rapidly developing and now is a part of almost every industrial branch. Different operations on image are used today in various fields, from space research, to different industries and security services to medicine and entertainment industry. The main goal in all of these different fields is to recognize two identical images or to recognize some parts of two different images and categorize them. There are many existing solutions, but we still have open questions in this field. This paper presents one possible solution regarding the open topics of real-time image processing which can be used in different (research, educational and professional) purposes. The main goals of this method are to recognize the same images and categorize them in some type of databases [1].

Software was developed in MATLAB software package and C++ program language. We chose this specific combination in developing to assure the best characteristics from both tools. MATLAB offers a large number of methods to process images, and C++ is a very efficient language. Also, both tools are object-oriented. As we mentioned above, our method and software can be used as a research or educational tool, or for professional purposes as a stand-alone application. As an educational tool this method has many possibilities for learning about operations which are useful in different stage of digital image processing (for example: acquisition, segmentation, edge detection, etc.). Software allows students to experiment with many parameters in every stage of processing. In that way they can learn and understand important factors and their effects on final results. Researchers can benefit from this software in different ways. However, the most important software property is the possibility to change different methods for specific purpose, or more radically, every researcher can write his/her own method which can be incorporated in our software. In that way researchers can modify our software for their specific applications. For professional purposes the software offers a special function for generating C code, which can be used along with microcontrollers, microprocessors, PLA, etc. Using all abilities which the software provides, one can modify the method and adjust it for specific operations and after that generate a standalone application or C code. This method and the developed software merge three main stages in digital image processing: acquisition of image and/or video, which can be used for creating database or test image (video), pre-processing, and recognition and categorization (Fig. 1). Therefore, this software can be used as a part in an integrated system. During the developing phase we wanted to derive an efficient method and software for image detection and classification for security services (checking of IDs, passports, driver licenses, etc.) and traffic control (detection of car plates, traffic signs, etc.). The presented results were obtained during developing, testing and verifying this method on different images.

II. Theoretical Backgrounds

This section described methods which are used in different software functions: acquisition, preprocessing, recognition and categorization. We will describe only the main characteristics of standard and well defined function in literature and every

A part of this paper was presented at the 55th ETRAN conference, Banja Vrucica, 6-9. June 2011.

I. Božić (corresponding author to provide phone: +381-64-2174568; e-mail: msc.ivan.bozic@gmail.com).

I. Lazić, was with Faculty of Traffic and Transport Engineering, University of Belgrade, Belgrade, Serbia (e-mail: ivanlazicsf@yahoo.com).

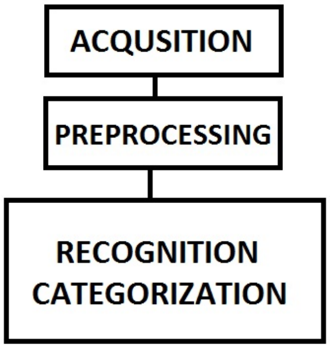

Fig. 1. Operation provided in presented method and software.

part of the algorithms proposed by the authors.

A. Acquisition

Acquisition provides operations for camera set-up, camera calibration, and radial distortion removal. Camera calibration is one of important operations while recording image and/or video of the same scene from different angles. There are many parameters that can be defined in the calibration section as: selecting the plugged-in camera, image or video format, brightness, contrast, color scheme, type of output file, etc. Also, the user must take control on type of acquisition, for database or testing. Radial distortion is the main problem during recording with a real lens, because that it is important to remove this distortion from images and videos. Quality of recordings is important during recognition and categorization stage. Every camera introduces radial distortion and we can remove it only using appropriate software solutions, and it cannot be removed by hardware solutions, because hardware mainly creates this distortion.

B. Preprocessing

All preprocessing operations can be divided in three main groups: determination of feature vectors, calculating stable points and segmentation (Fig. 3). Also, in preprocessing we include fixture processing's which is done by other software methods. Algorithms we use in preprocessing are common and the most used for particular operation. For the determination of features vectors our software uses SIFT (Scale Invariant Feature Transform) and SURF (Speed Up Robust Features) detectors; for calculating stable points we used the RANSAC (RANdom Sample Consensus) method, and for segmentation we choose the iterative Graph Cut method. In the further text all these methods will be briefly described, also, we will include detailed information about parameters and the function which were proposed by us.

At the first step, SIFT [2] determines the key-points on the sequence of images and stores them in s database, after which this method select some number of key-points useful for recognition phase. In the next step this method compares the key-points from the test image with the key-points from all images in the database and the best candidate is chosen by different metrics conditions. This method is the most commonly used, because it is robust toward different kinds of noise and gives a large number of key-points for comparison [4,5]. SURF [3] is another detector used in our software. It is a newer method than SIFT, and it is based on it. The main difference between these two methods is time efficiency, which is better in SURF than in SIFT [5].

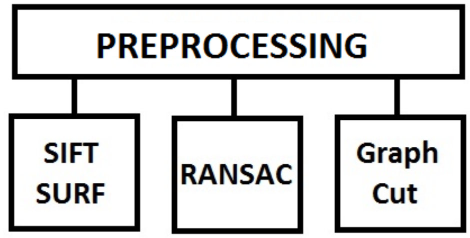

Fig. 2. Preprocessing.

RANSAC method determines stable points from set of key-points which are obtained by using one of feature vectors detectors. This method can be used in many different applications which need fitting some points, regardless of the fact if these points can or cannot be described by some models. The implementation and the results for this method hugely depend on specific parameters. Except the standard parameters we proposed using one additional parameter to determine the distance between two frames in which the RANSAC method is implemented. Every user should be extremely careful with this parameter, because if properly selected the RANSAC process will be much more effective, but on the other hand it could cause the divergence of the whole process.

The Graph Cut method [7,8,9] is an iterative method for segmentation. The main advantage of this method is that the user has an opportunity to affect the result of segmentation by parameters. These methods use two types of input parameters, vector of points in the foreground and the background. These two vectors are used as the initial conditions. The method stops the iterative process when all points on the image belong to the first or to the second vector of initial conditions. In our solution we proposed two methods that automatically define the initial conditions. One of them used stable points, and the other uses the intensity levels and threshold. Also, we obtained a function which gives the opportunity to the user to manually define vectors of initial conditions or to incorporate some automatic solution that they made.

C. Recognition and Categorization

For recognition and categorization this software use several different methods, but the most useful are: NNS (Nearest Neighbor Search), histogram and covariance region descriptor. During categorization we use three different measures for distance: Euclidean, Manhattan, and Mahanalobis.

NNS [15] is used for determine the same object in big databases. This is an optimization problem in which we look for the nearest neighbor points in different metrics. In the program we use several different algorithms for this method such as: linear NNS, space partitioning and locality sensitive hashing. For the software purposes we combine these three

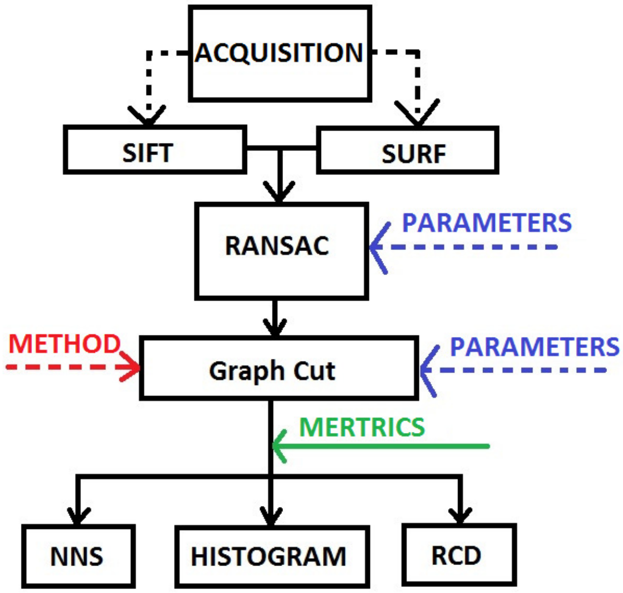

Fig. 3. Software block diagram.

algorithm with different metrics (Euclidean, Manhattan, and Mahanalobis). Also, the software supports the k-NNS algorithm in which we are looking for k nearest neighbor.

The histogram comparison is also a powerful technique for recognition and categorization. Here all three metrics are used as in the previous method. This method compares pixel levels, it is of utmost importance to always use normalized histograms. If we do not normalize them we introduce an error and process of categorization becomes unsuccessful.

Region covariance descriptor is based on covariance matrix and integral images. Integral images are used for faster computation of region sum and it's define for (x',y') pair as sum of all pixels values in image region $I(0{:}x',0{:}y')$. For image I of dimension W x H is defined F as a W x H x d dimensional features image provided from I with any mapping operation (intensity, color, gradients…). After that we choose region R such that is subset of F and for some z_k k=1,..,n, d-dimensional features point inside R. We represent a region R as d x d covariance matrix of features points as:

$$C_R = \frac{1}{n-1}\sum_{k=1}^{n}(z_k-\mu)(z_k-\mu)^T \qquad (1)$$

Because the covariance matrix does not lie in the Euclidean space, is necessary for using this algorithm to find the way for computing distance between the points for NNS algorithm. There used distance measured based on dissimilarity of two covariance matrices as:

$$\rho(C_1,C_2)=\sqrt{\sum_{i=1}^{n}\ln^2\lambda_i(C_1,C_2)} \qquad (1)$$

III. Software Implementation

As we mentioned above, our software is implemented as a standalone application using MATLAB and C++. Graphical user interface has one main window from which the user can

Fig. 4. Four type of testing documents, from top to bottom: biometric passport, biometric ID, driver license and registration certificate.

access some of the functions offered by the software. Every particular operation has its own window. Thus we have five additional windows, and they implement the following functions: acquisition, detection of feature vectors, and determination of stable points, segmentation, recognition and categorization (categorization is used only if database is loaded) and generating executable C code. A block diagram of this software realization is shown on Fig. 3.

Software may – or may not – start with acquisition. Acquisition is an optional operation and the rest of the software procedure doesn't depend on it. Therefore, acquisition can be used as a separate part for creating database or individual snapshots. Also, if the user skips the acquisition step, in the next step he/she has an option to load a database or a test image (images whose existence is checked in database), and of course there is a possibility to load sequences of test images. After completing the first step (acquisition or loading) the user must select the appropriate descriptor. Both realized detectors have their advantages and disadvantages, and the success of algorithm for recognition and categorization depends on their appropriate selection. Independently on the chosen detector, the next step in the method is the RANSAC algorithm. In this phase the user can again modify the original software. In this moment he/she can add an additional parameter which can adjust difference in the number of frames which are compared. As we mentioned above, this parameter can be extremely useful for the experienced users and it is not recommended for new users to experiment with it. The next operation is segmentation. Segmentation is the most important operation, and because of that it has the largest number of options. Besides manually placing parameters, there are three more available developed algorithms for automatic placement of boundary conditions which are based on stable points [1,2,3,6], pixels levels and edge detection [13,19,20]. Choosing some of these options gives the user an opportunity to change their parameters. There is no a general better solution, because of that they should be used for basic needs. Therefore, in this place the user can add his/her own solution. This option is primarily designed for the experienced users. After completed segmentation, the image is shifted to ideal position with regard to the reference database. This is the only phase that the user cannot influence, and it is fully automated. After segmentation there is a final step and the main goal of the whole algorithm. In this moment the user can select one of three possible methods for recognition and categorization. In the future versions of the software we plan to expand the available methods. The selection of methods, their realization and metrics should depend on practical problems and the desire is that all methods give equally good results in every particular case.

The closing step available to the user is the generation of C code with all parameters and methods which were used during the software operation. This code can be used with other C assemblers and while creating integrated systems.

IV. Results

In this section we present and analyze results. We start with the description of databases used during software validation. Then we describe the methods and parameters of software used to obtain the presented results, and at the end we introduce a method for evaluating the quality of software (using two types of errors).

As a result we present the results obtained on five databases. One database is used in verifying stage. This database consists from different types of business cards; the distribution of business cards is presented in Table I.

TABLE I
Business Card Database

Image description	Number
Total number	100
Horizontal images	78
Vertical images	22
With white background	40
With colored background	30
Multi colored business cards	20
Illustrated business cards	10

TABLE II
Test Results on Business Cards

Business card	NNS	Histogram	RCD
White	98.2	99.1	99.8
Colored	99	97.6	99.4
Colorful	96.5	99.2	98.9
Illustrated	97.3	98.1	99.2

The distribution in database is thus created to give the best possible description of real problems (the biggest number of business cards in database, 70 %, has one-color background), illustrated and multicolor business cards are also included. Business cards are an ideal object for software testing, because they share the same characteristics with the target documents. All documents that we wanted to process using this software had a number of similar characteristics: every document had known fixed dimensions, all documents has rectangular shape, except the traffic signs which can be rectangular, rhomb-shaped or circular and every group of documents had the same backgrounds. Other four databases were used for testing software and consisted of a database with biometric IDs of Republic of Serbia, a database with biometric passports of Republic of Serbia, a database of driver license of Republic of Serbia, and registration certificates for cars of Republic of Serbia. The databases of documents on which the software was tested were smaller and included forty images biometric passports, sixty five images of biometric IDs, fifty four images of registration certificates, and forty images of driver license, as shown in Table III.

The first results obtained with business cards represent comparison of different algorithms that can be generated with software. Table II shows a part of results related to a success in recognition of business cards (in percent) on which the SIFT method was applied, depending on the chosen classifier. For all classifiers we used Euclidian metrics (except for RCD).

TABLE III
DOCUMENTS DATABASES

Type of documents	Number
Biometrics ID	65
Biometrics passport	40
Driver license	30
Registration certificate	54

TABLE IV
RESULTS ON DOCUMENTS

Document	FAR	FRR
Biometric IDs	1.0	1.8
Biometric passports	1.08	2.5
Driver license	1.20	2.4
Registration certificate	1.9	4.2

As we can conclude from Table II, the results with SIFT detector are satisfactory, and the comparison with the remaining data has shown that is the best combination of parameters is, if we choose the next algorithm. For the determination of feature vectors we should use the SIFT method; in the RANSAC procedure good results are obtained if we, together with standard parameters, use a new realized parameter. This parameter determines the difference between two compared frames, and it should be set to five. For the segmentation we used the method implemented by stable points and additional processing. This method is very effective and accurate. For recognition and categorization we use the covariance matrix, because it gives the best results.

For the verification of the accuracy we introduce two types of errors. The FAR (False Accept Error) error describes the percent of images recognized as members of database, although they are not in the database. The FRR (False Reject Error) error describes the percent of images that are members of database, but were not recognized as such. Although there are no good errors, it is important in security services that the number of the FAR errors is lower than that of the FRR.

The obtained results for all four testing databases are presented in Table IV.

The obtained results show that this kind of solution can be equally useful in research, educational and professional purposes. Further research should consider the usefulness of our solution in traffic, especially if we further minimize errors.

V. DISCUSSION AND CONCLUSION

All results have small FAR errors, but the FRR is significantly different between different databases. Also, there are very similar differences between the FRR and the FAR errors for all databases. As expected, databases with documents which include pictures give better results than the documents without pictures. That can be discussed from different angles. Better recognition and categorization is very likely a result of stable key-points on the personal image on documents. On the other hand, a document like an ID or passport includes even more, two pictures, personal signs and fingerprints. In contrast to these documents registration certificates include only textual data, which produce weaker and lower number of stable points. FAR results are better primarily because the backgrounds are clearly different in every type of documents, and the software easily recognized if a picture belonged to that database. FRR errors are greater primarily because there are not enough to determine if this image belongs to this database. Software must recognize if this image has a match with any other image in the database.

For future research we put before ourselves a task to recognize different features, which is important for automatic traffic control. This problem is very interesting, because quality recognition and categorization of traffic signs is the main problem in automatic traffic control. We will analyze the efficiency of the software in the recognition and categorization of car plates and traffic signs. This will present a similar problem, but a different approach will be needed. When we analyze documents we have a static situation, recorded images or videos are stationary. On the other hand, recording of car plates and traffic signs would be dynamic situation. Furthermore, recording car plates and traffic signs are different problems, because when we record plates, we have fixed position of camera and plates are dynamic, on the other hand traffic signs are fixed and camera is moving. For these two problems it is important to determine the best position of the camera, regardless of the fact if the camera is fixed or moving. Additional problems in traffic control applications are time is available for recording and successful categorization, or the existence of more than one traffic sign on the same pillar [23]. There are some solutions for this type of problems, but no solution is perfect and this situation may be considered as an open problem.

The presented results and the obtained errors show that this solution can be very useful for different purposes and applications. In further work we will try to decrease the number of errors and implement better solutions, or time-efficient solutions. Also, we plan to create an integrated system and test it in real traffic situations.

REFERENCES

[1] A. Forsyth and J. Ponce, Computer Vision: A Modern Approach, Prentice Hall, 2003.

[2] DG. Lowe, “Distinctive image features from scale invariant kezpoints,” International Journal of Computer Vision, vol. 60, pp. 91-110, Januar 2004.

[3] H. Bay, T. Tuytelaars and L. Van Gool, “SURF: Speed Up Robust Features,” ECCV 06, November 2006,

[4] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” IEEE Trans. Pattern Recognition and Machine Inteiligence, vol. 27, pp. 1615-1630, October 2005.

[5] L. Juan and O. Gwun, “A comparation of SIFT, PCA-SIFT and SURF,” International Journal of Image Processing, vol. 3, pp. 143-152, August 2009.

[6] MA. Fischler and RC. Bolles, “Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography,” Graphics and Image Processing, vol. 24, pp. 381-395, June 1981.

[7] AP. Eriksson O. Barr and K. Astrom, “Image Segmentaions using minimal graph cuts,” Proceedings SSBA06, pp. 45-48, March 2006.

[8] J. Malcom, Y. Arthi and AR. Tannenbaum, "A graph cut approach to image segmentation in tensor space," IEEE Conferencion on Computer Vision and Pattern Recognition CVRP07, pp. 1-8, June 2007.
[9] E. Vincent and R. Laganiere, "Detecting planar homographies in an image pair," Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis, pp. 182-187, June 2001.
[10] A. Pinz, "Object Categorization, " Computer Graphics and Vision, vol 1, pp. 255-353, 2006.
[11] S. Dickinson, A. Leonardis, B. Schiele and MJ. Tarr, Object Categorization: Computer and Human Vision Perspective, Cambridge University Press, 2009.
[12] E. Fazl and JS. Zelek, "Region detection and description for object categorz recognition," CRV07, pp. 3321-3328, May 2007.
[13] J. Canny, "A computational approach to edge detection," IEEE Trans. Pattern Recognition and Machine Inteiligence, vol. 8, pp. 679-698, November 1986.
[14] P. Felzenszwalb, R. Girshick, D. McAllester and D. Ramanan, "Object detection with discriminatively trained part-based models," IEEE Trans. Pattern Recognition and Machine Inteiligence, vol. 32, pp. 1627-1645, September 2010.
[15] L. Andrews, "A template for nearest neighbore problem," C/C++ Users Journal, vol. 19, pp. 40-49, November 2001.
[16] D. Stanković, "Prepoznavanje i detekcija lica sa digitalne slike korišćenjem algoritma za filtriranje Gausovog šuma zasnovanog na LPA-ICI tehnici," Zbornik radova 53, konferencije ETRAN RT6.2, pp. 1-4, Jun 2009.
[17] S. Milovanov, R. Benčik and B. Mišić, "Primena Support Vector Machine klasifikatora u prepoznavanju karaktera sa registarskih tablica," Zbornik radova 53, konferencije ETRAN AU1.9, pp. 1-3, Jun 2009.
[18] A. Mijatović and D. Dujković, "Primena hromatske segmentacije i geometrijksih karakteristika za detektovanje saobraćajnih znakova na slici," Zbornik radova 52, konferencije ETRAN EK3.5, pp. 1-4, Jun 2008.
[19] Đ. Klisić, I. Božić and M. Petrović, "3D IMPRO DYNAMIC – program za digitalnu obradu medicinskih slika," Zbornik radova 52, konferencije ETRAN EK3.7, pp. 1-4, Jun 2008.
[20] Đ. Klisić and I. Božić, "MedPicPro – program za digitalnu obradu medicinskih slika," 16. Telekomunikacioni forum TELFOR, pp. 621-624, Novembar 2008.
[21] O. Tuzel, F. Porikli and P- Meer, "Region Covariance: A fast detector for Detection and Classification," ECCV06, pp. 589-600, May 2006.
[22] I. Božić, A. Savić, K. Savić and M. Gavrilović, „Softver za obradz elektrofizioloških signala", Zbornik radova 54. konferencije ETRAN ME1.3, pp. 1-4, Jun 2010.
[23] I. Lazić, N. Bjelobaba, I. Božić, and M. Beljin, „Detekcija i klasifikacija slika u saobraćaju" Zbornik radova 55. Konferencije ETRAN EK1.6. pp. 1-3, Jun 2011