Automatic Segmentation and Classification of Resistors in Digital Images

Mia Muminovic
Department for Automatic Control and Electronics
Faculty of Electrical Engineering, University of Sarajevo
Sarajevo, Bosnia and Herzegovina
mmuminovic2@etf.unsa.ba

Emir Sokic
Department for Automatic Control and Electronics
Faculty of Electrical Engineering, University of Sarajevo
Sarajevo, Bosnia and Herzegovina
esokic@etf.unsa.ba

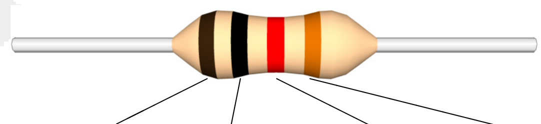
***Abstract*—Computer vision systems are frequently used for inspection and classification of products during manufacturing. Image processing and analysis allows non-invasive extraction of object features within an image and the classification of objects based on the extracted data. Shape, texture and color are typical features that can be extracted from an image and used for object recognition. In this paper, a method of detection, segmentation and classification of resistors captured in digital image, based on their nominal values, is presented. The process consists of the following steps: image segmentation, morphological image processing, representation and description of objects, object features extraction, classification of extracted data using support vector machines (SVM). Experimental results show that the proposed method exhibits solid performance and real-time operating capabilities.**

***Index Terms*—image processing, segmentation, identification, classification**

I. Introduction

One of todays most challenging problems is to teach computers to understand what they are able to "see". Computer vision is a very challenging research field due to the presence of noise, large amounts of data, data losses happening when transforming from 3D to 2D space, and aggravated interpretation of images which usually requires some sort of artificial intelligence.

One typical example is the reading of the color-encoded resistor value. Resistors are passive electronic components, mostly characterized by their resistance. Due to their construction, the resistance value of through-hole resistors is not marked using numbers or letters, but using four or more color bands (rings), as shown in Fig. 1. Experienced engineers usually memorize the color coded table, and use it for "decoding" resistor values. On the other hand, students or hobby users are not able to quickly identify the resistor value, or they rely on decoding tables. Due to everyday presence of smart mobile phones (computers with vision sensors), the resistor value reading task could be delegated to a computer vision system. Having this in mind, one of the goals of this research is to develop a robust method of reading resistor values, which may ultimately be implemented as a mobile phone application. Moreover, the developed method would be used as part of automatic classification/pick-and-place machine and visual inspection system in a manufacturing process.



	1^{st} digit	2^{nd} digit	multiply	tolerance
Black	0	0	1	1% (F)
Brown	1	1	10	2% (G)
Red	2	2	100	
Orange	3	3	1K	
Yellow	4	4	10K	
Green	5	5	100K	0.5% (D)
Blue	6	6	1M	0.25% (C)
Violet	7	7	10M	0.1% (B)
Grey	8	8	100M	0.05% (A)
White	9	9	1G	
Gold			0.1	5% (J)
Silver			0.01	10% (K)
None				20% (M)

Figure 1: Rules for determining the nominal resistance of resistor

Due to its complexity, automatic determination of the resistor nominal resistance using image processing has been and still is a challenging problem. In order to correctly classify a resistor, it is necessary to crop the body of the resistor from the image and then identify the color bands of the resistor.

Far in 1997 Chan et al. [1] examined and evaluated different color spaces and have proposed methods for a contactless assessment of the resistor values, namely the Bayes classifier and the fixed boundary method. Although they showed that a uniform color space CIE-LAB is the most suitable color model for recognition of color bands, due to its insensitivity to minor intensity and color variation, they opted to perform recognition of color bands using HVC space. Y. Mitani et al. examined the color spaces as well. In their paper [2], they focus on 10 different color spaces. Results showed that L*u*v*, L*a*b*, and I1I2I3 outperform the RGB color space just in case of a small number of training samples.

Both [1], [2] do not show any color band recognition from real images. In [3], authors present a method for reading a resistor value based on image processing. Their method consists of two steps: extraction of color bands using segmentation based on the k-means method and classification of the resistor colors using 1-NN classifier. The method is tested on 30 real images. However, their initial assumptions are that the resistor position is central and the orientation of the resistor is horizontal, with the color lines in the right order to be read, which is very restrictive. Y.H. Chen et al. [4] present an automatic method for reading resistance of a resistor based on image processing. Unlike [3], orientation of the resistor is not limited to a specific angle. The method consists of segmentation, horizontal alignment of resistor, extraction of color bands and classification of colors. K-NN classifier is used for the classification of colors. The accuracy of the method is rather high (92%), at the cost of a significantly complicated and expensive acquisition setup (digital microscope embedded in a diffuse cover, a controllable LED ring-type light source etc.). In their subsequent paper [5] they focus on choosing and building the right light source, discuss the specular reflections and halo effects. Their future work consists of adjusting the proposed system to recognize multiple resistor from an image.

Other authors proposed a hybrid classification of resistors based on image processing [6], but from a distributed computing point of view. The image processing steps are divided among clusters. Even though the proposed method gives promising results and can read the resistance of multiple resistors from an image, they used synthetically created images instead of realistic ones.

In one of the newer papers [7], authors discuss real-time video processing and the development of an Android application for estimating resistance value. The algorithm consists of two steps; the first step includes noise filtering and color band extraction; the second step represents color band classification. The color band extraction is facilitated by using an augmented horizontal line. The horizontal line is placed in the center of the video frame and it has indicators for each line. The resistor must be aligned with this line. The Euclidean distance based clustering is used for color recognition. Although the method has 8% error rate, it requires correct positioning of the resistor in the given frame.

As it can be seen from the related work in this area, none of the stat-of-the-art methods is perfect. Some approaches are highly dependent on the light source or require very advanced methods of acquisition, some are too restrictive in context of resistor positioning (require specific position of the resistor on the image) and some are limited to one resistor per image.

To this end, we propose to use simple and effective approach to read resistor values from digital images. The method is implementable on mobile phones since it requires an average mobile phone camera (at least 150 pixels per resistor length e.g. standard 8 MPix camera may capture ca 20 resistors at once), and the computational efforts are relatively low. Different image processing methods are applied to diminish the specular reflection effects and illumination dependence. The proposed method does not impose restrictions on the number of resistors per image, nor position and orientation of the resistor. However, the method is currently limited

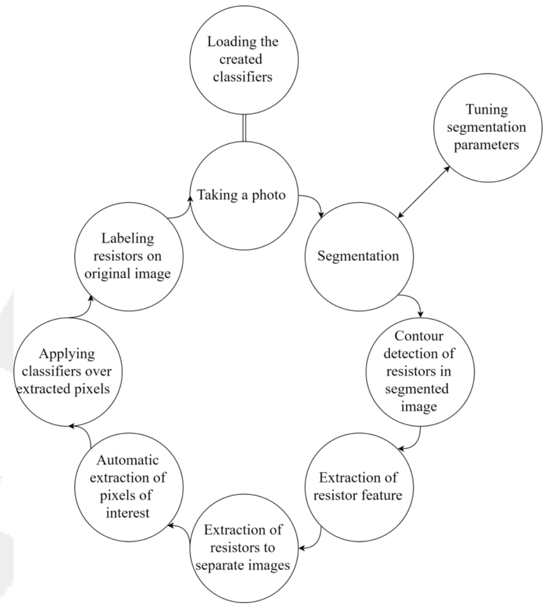


Figure 2: Proposed algorithm for determining the nominal resistance of the resistors

only to 4-band resistors. The classification is based on a popular machine learning method (Support Vector Machines). Therefore, the experimental result exhibit outstanding performance, both in terms of accuracy and real-time execution. In addition, a database of images is generated, uploaded and free to use to the research community (URL: people.etf.unsa.ba/~esokic/ResistorDataset/ResistorDataset.zip).

The paper is organized as follows. Section II introduces the proposed method step-by-step. Experimental results are given in the third section. Conclusion and guidelines for future work are given at the end of the paper.

II. Proposed algorithm

The proposed process of determining the nominal value of resistors captured in digital images consists of following steps: image segmentation, morphological image processing, representation and description of objects, extraction of object features, classification of the extracted data using Support Vector Machines (SVM). Detailed structure of proposed algorithm is presented in Fig.2. Each of the following steps will be explained in more details in the following subsections, while the result of each step is illustrated in Fig. 3. The whole implementation is done in OpenCV/C++. Standard 1/4W, 6.5mm long resistors with light brown body are used as test objects (step 1 in Fig.2).

A. Image Segmentation

Image segmentation secludes specified elements of the image by partitioning the image to segments. At the end of the segmentation process, a set of closed contour lines (objects) is obtained as the result. Thresholding, edge-based segmentation and region-based segmentation are the three of the most used techniques of segmentation [8]. All of the

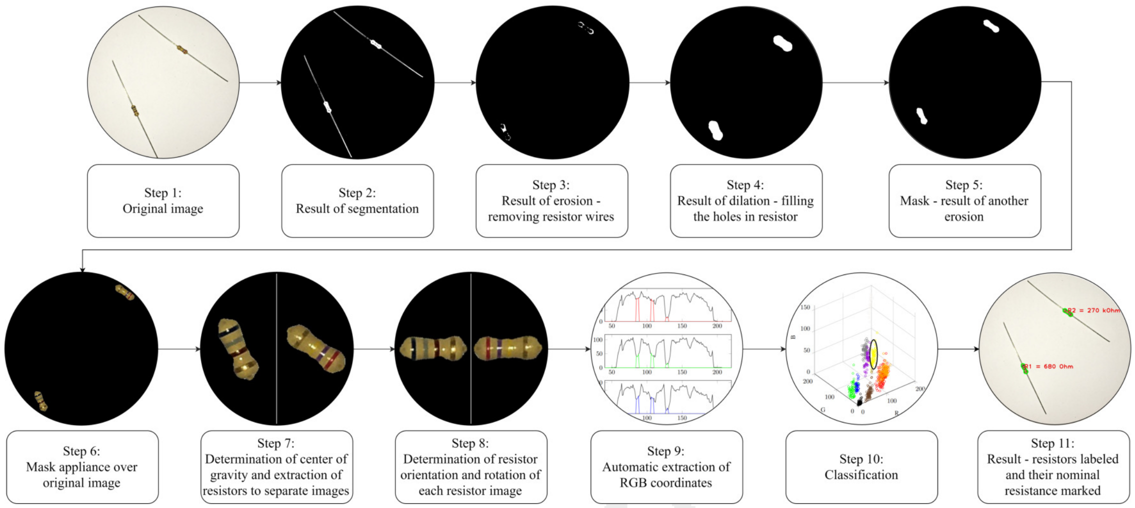


Figure 3: Algorithm steps for determining the nominal resistance of resistors

mentioned methods have their own advantages and drawbacks. Segmentation with one threshold is the simplest method which separates the pixels based on their intensity value. It is used as the method of choice in our application, but it requires that the background (the table on which the resistors reside) is of lighter colour. This method was chosen because of its short execution time and its high accuracy in extracting objects from the background in images.

The result of segmentation is an (imperfect) binary image (step 2 in Fig.3). In order to extract the resistor from the image, and remove the unnecessary parts of images (such as noise or the wire leads of the resistor) specific morphological operation need to be applied.

B. Morphological operations

In order to separate the body of the resistor from its leads, thinning of the image needs to be performed. In addition, small holes in resistor bodies which appeared during segmentation need to be filled. These tasks are solved using morphological operators [8]. First the noise and resistor leads are removed from binary image using erosion, and then the dilation is applied in order to make the objects more visible and return its size to original. Thus a mask is created (step 3-5 in Fig.3) and it is possible to move on to the next stage of the method, which is detecting the resistor contours. It is important to note that the kernel size for the morphological operators is chosen based on the nominal size of the resistor, camera distance from the analyzed plane and the resolution of the camera sensor.

C. Object representation and description

After obtaining the results of the segmentation, namely the regions of interest (groups of pixels), it is necessary to represent and describe them in a suitable way for further processing and analysis. Identified regions can be represented based on their region borders or based on pixel characteristics of that region. One of the methods that are used for segment representation and description is chain code [8]. It is essential to point out that contour detection is based on the binary image that should be previously created. Once the region of interest (or contour) of the object is found, it is possible to determine the centroid, area, orientation and other similar features of the object shape. In Fig.4, for example, the chain-code based extracted contour of the resistor is labeled using green color.

D. Shape descriptors extraction

Prior to extraction of the resistor bands, the part of the original image which represent the resistor body need to be extracted and rotated properly (steps 6-8 in Fig.3) Once the contour is determined, the following basic descriptors may be simply extracted: center of gravity (centroid), axis of least inertia and shape moments [9].

The concept of moment in image processing evolved from the concept of moment in physics. Moments are used for the shape analysis of objects, whether they are presented with a region or a contour [9]. Moments enable to compute two important descriptors. The first one is the center of gravity (or centroid) of the resistor, which is actually a center of the shape and determines the position of the resistor within an image (red dot in Fig.4). The second descriptor is the axis of least inertia (ALI) which is defined as a line for which the integral of the square of the distances to the points of the shape boundary is minimal [9]. It is used for determining and correcting the rotation of the resistor in image. More specifically, the angle of inclination between ALI and x-axis is computed. The resistor then can be rotated for the computed angle, so that it is positioned in its nominal orientation prior to examining its color bands. The detected orientation of the resistor is labeled in Fig.4 with blue axes - longer axis represent the ALI.

E. Extracting resistor color features

In Fig.1, rules for determining the nominal resistance of the resistor are presented. The color of each line on the resistor determines the resistance value. Colors of the first and second bands are encoded with values from 0 to 9, and they determine

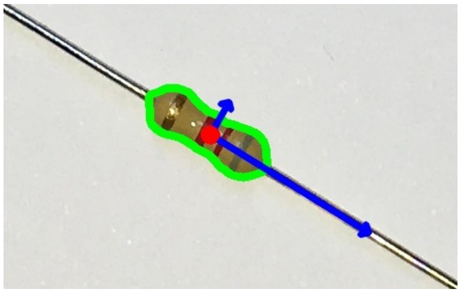
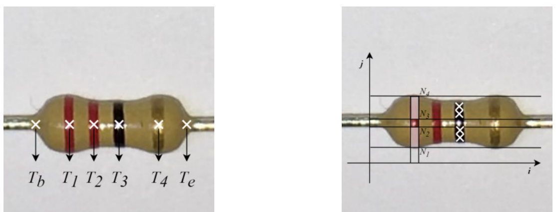


Figure 4: Features labeled on the resistor: red dot - centroid, blue line - ALI.



(a) (b)

Figure 5: Manual feature extraction from resistor image, a) extraction of pixels for conducting the line position statistics, b) extraction of RGB values for black color

the mantissa (first color - the first digit, second color - the second digit), which is then multiplied by the value coded with the third color (1 to 10^9) to obtain the nominal value of the resistance. The color of the fourth line determines the resistor tolerance and it is usually golden. For example, the resistor in Fig.1 has nominal resistance of $1k\Omega$ (first color - brown (1), second color - black (0), third color - red (10) $\rightarrow$ $10 \cdot 100 = 1k$). The most important step of the application consists out of two tasks:

- automatic extraction of color descriptor that belong to the first, second and third line of the resistor,

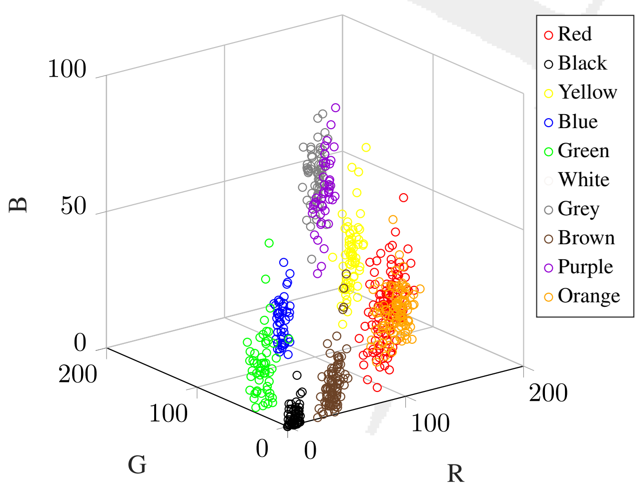


Figure 6: Visualization of training data in RGB color space.

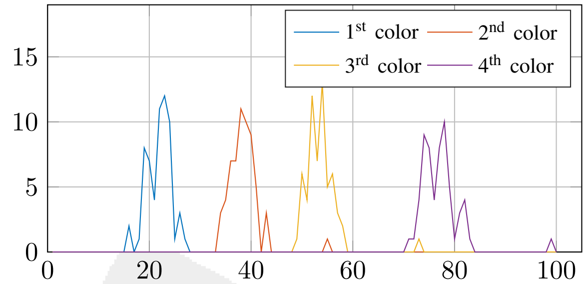


Figure 7: The statistics of line positions on resistors

- classification of extracted descriptor in order to identify colors of resistor lines.

Due to the fact that color descriptors of the first, second and third band are automatically inputted to the created classifiers, it was necessary to determine the exact position of the colored ring in order to extract the color descriptor. Since the positions of the ring slightly vary, the statistics of line positions on resistors have been conducted. Different coordinates from each image in training set have been manually selected (Fig.5a):

- point that represents the left end of the resistor T_b
- point that represents the center of the first band T_1,
- point that represents the center of the second band T_2,
- point that represents the center of the third band T_3,
- point that represents the center of the fourth band T_4,
- point that represents the right end of the resistor T_e.

Resistor length D in pixels is obtained as:

$$D = T_e - T_b \tag{1}$$

Positions of first line P_1, second line P_2, third line P_3 and fourth line P_4 regarding resistor length are calculated as:

$$P_1(\%) = \frac{T_1 - T_b}{T_e - T_b} \cdot 100\% \tag{2}$$

$$P_2(\%) = \frac{T_2 - T_b}{T_e - T_b} \cdot 100\% \tag{3}$$

$$P_3(\%) = \frac{T_3 - T_b}{T_e - T_b} \cdot 100\% \tag{4}$$

$$P_4(\%) = \frac{T_4 - T_b}{T_e - T_b} \cdot 100\% \tag{5}$$

The following results are obtained (shown in Fig.7):

- first color - positioned at 22-24% of the resistor length,
- second color - positioned at 37-39% of the resistor length,
- third color - positioned at 52-54% of the resistor length,
- fourth color - positioned at 76-78% of the resistor length.

Based on the obtained results, it is sufficient to determine the end points of the resistor in order to determine the band positions. Moreover, band width in respect to resistor body length needed to be estimated. According to the statistical analysis, the width of the resistor band (ring) regarding the overall length of the resistor is 5% on average.

The color descriptor of the band is formed using the axial color profile of the resistor body. First the unnecessary parts of

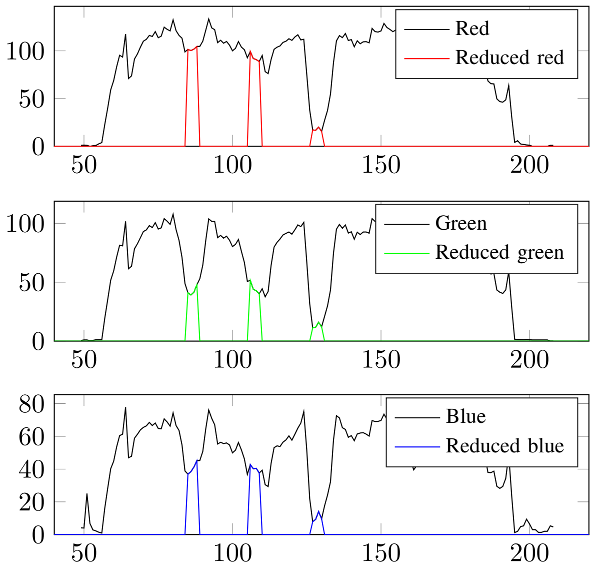


Figure 8: Reduced color profile for the resistor shown in Fig.5a. Signals marked with red, green and blue are taken into account and other parts are discarded.

the color profile are removed (Fig.8) and only the relevant parts (in context of band position on the body) of the color profile are taken into account. Then the color descriptor is computed for each band by using Eq.(6), according to notation in Fig.5:

$$RGB_{band-k} = \sum_{\substack{i \in (T_k - \epsilon, T_k + \epsilon) \\ j \in (N_1, N_2) \cup (N_3, N_4)}} \mathbf{f}(i,j) / \{2\epsilon[(N_4 - N_3) + (N_2 - N1)]\} \quad (6)$$

where $k \in 1, 2, 3, 4$ is the number of the band, ϵ is the half-width of the band in pixels and $\mathbf{f}(i,j) = [R_{ij}, G_{ij}, B_{ij}]$ is the RGB triplet value of the pixel at location (i, j). In order to avoid the effect of reflection, the central part of the resistor is not taken into account when calculating the color profile. The extraction of the color descriptor is given as step 9 in Fig.3.

In order to create a training set for the classifier, a database of 60 images of different resistors (180 value bands) has been created. To obtain the best possible training set, pixels and their RGB values have been marked and manually labeled (as in Fig.5b), and suitable color descriptor for each color is determined. A color space (Fig.6) is created based on the obtained data and used for training, i.e. creating classifiers.

F. Classification

Classification is one of the most common tasks of machine learning. The goal of the classification is to classify an unknown instance into one of the predetermined classes. The task is to form a model based on which unknown objects are classified. In case of supervised learning, classification uses a set of data for training and set of data for testing. Classification algorithm generates a classifier based on the input set of training data and corresponding labels. Afterwards, classifier receives unknown data and it provides an output that shows whether the data belongs to a particular class or not.

As previously explained, nominal resistance of resistor is determined according to colors of the lines on the resistor. Due to its simplicity are relatively fast execution, we propose to use Support Vector Machines (One-vs-All training strategy) to classify the color descriptors to its belonging class. In order to use SVM, first the classifiers for all ten colors needed to be created (black, brown, red, orange, yellow, green, blue, purple, grey and white), as well as for the body color of the resistor (light brown). Thus the dataset acquired in the previous subsection is used for training the classifier.

Toolkit *dlib* is used for the creation of classifiers. Learning object takes one color descriptor labeled with $+1$ and other color descriptors labeled with -1 as an input; it provides a classifier for color descriptor labeled with $+1$ as an output. This function takes new data as an input and provides an output value greater than zero if the input data belongs to the class. Otherwise, it returns a value less than zero. The function actually calculates the distance for the new color descriptor from each color cloud (data set of each color) and classify the data to the nearest color cloud.

There are in total eleven classifiers, ten for band colors and one for resistor body color. What is a convenient feature that not all combinations of first two colors are possible, thus an error can be detected to a certain extent and can be eliminated. In our setup, the most common commercially available series (so called "E12 series") is used. In this series, only the following combinations of first two digits are possible: 10, 12, 15, 18, 22, 27, 33, 39, 47, 56, 68 and 82.

Finally, the proposed algorithm for determining the nominal resistance is as follows:

1) color descriptors of first two bands are forwarded to the classifiers,
2) classification is performed,
3) results of classification for each band is saved, results that say it is a body color are discarded,
4) the classification results are sorted by the highest percentage and the number of repetitions, and the two digit value is chosen,
5) if no value had been saved or the value that is obtained does not correspond to values of possible combinations, image is flipped and the procedure repeats,
6) after the first two colors are identified, color descriptor of the third band is forwarded to the classifiers,
7) the classification results are saved, sorted and the best value is taken as final result,
8) the nominal resistance is computed.

In the end, resistors are labeled in the original image by their nominal values (Fig.3, steps 10-11).

III. Experimental Results

In this section, some experimental results achieved using previously described method are discussed. The algorithm is implemented in C++ and OpenCV library. For fair comparison and pointing out the weak points of the proposed method, experiments are conducted on specifically chosen five images

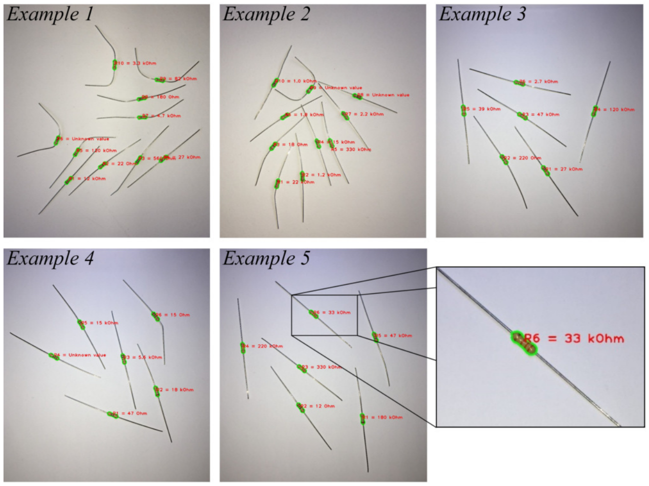


Figure 9: Experimental results - five images with total of 38 resistors.

TABLE I: Classification results for examples in Fig.9

	Result										Acc.[%]
	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	
Example 1	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	**80.00%**
Example 2	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	**80.00%**
Example 3	✓	✓	✓	✓	✓	✓					**100.00%**
Example 4	✓	✓	✓	✗	✓	✓					**83.33%**
Example 5	✓	✓	✓	✓	✓	✓					**100%**

(38 resistors in total) with resolution of 3024x3024. The obtained results are shown in Fig. 9 and Table I.

It may seem at first glance that the proposed method accuracy is lower than expected (around 86% in average). Nevertheless, errors in evaluating nominal resistance values are present only when evaluating resistors which have bands in different positions than predicted by the statistical analysis. This leads to error in classification and evaluating the nominal resistance value. Although the statistical positioning of the resistor bands is extremely robust from image processing point of view, it is worth reconsidering this approach as part of future work (e.g. using edge detection for band position estimation).

Another very important feature of the proposed algorithm is that execution time is short and it may be considered for real-time application in industrial environment. The application is tested on a PC with Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz processor, 4GB DDR3L RAM memory, Linux Ubuntu operating system, OpenCV (v.3.2.0.) and QT (v 5.8). The average execution time of each part of the algorithm are presented in Table II. It is clear that all operations are executable under 1[ms] almost regardless of the number of resistors on image. The execution time of morphological operators is the largest; it is affected by kernel size and type. The increase of the kernel size leads to longer execution time. The used kernel type is ellipse. Choosing a simpler type (e.g. rectangle) drastically decreases execution time, but lowers the quality of the mask.

TABLE II: Execution time of certain algorithm parts

Name of algorithm part	Execution time [ms]				
	Examp. 1	*Examp. 2*	*Examp. 3*	*Examp. 4*	*Examp. 5*
Uploading the image	0.0826	0.0851	0.0826	0.0833	0.0833
Conversion to grayscale	0.0065	0.0066	0.0071	0.0061	0.0067
Threshold segmentation	0.0027	0.0029	0.0027	0.0028	0.0027
Image erosion	0.0577	0.0564	0.0581	0.0578	0.0582
Image dilation	0.3334	0.4419	0.3355	0.3337	0.3358
Second image erosion	0.1237	0.1045	0.1238	0.1231	0.1269
Contour detection	0.0064	0.0065	0.0063	0.0056	0.0057
Masking	0.0251	0.0213	0.0228	0.0132	0.0293
Segmentation	0.0073	0.0094	0.0065	0.0056	0.0052
Determining resistance	0.0383	0.0392	0.0156	0.0144	0.0252
Labeling resistors	0.0012	0.0011	0.0007	0.0009	0.0008
Total	**0.6849**	**0.7749**	**0.6617**	**0.6465**	**0.6798**

IV. Conclusion and Future Work

According to the presented results in this paper, it can be concluded that it is possible to evaluate the nominal resistance value of resistors using computer vision, although the proposed method in its current form still requires sufficient number of pixels per resistors, relatively homogeneous and bright background and initial system calibration.

The proposed algorithm heavily relies on results of a statistical analysis which was conducted on a specific set of resistors. Thus, it is difficult to predict the performance of the system on resistors of other types, unless additional classifiers are trained. This will be a mandatory topic of research for future work. One of the possible solutions could be to create a detector of resistor bands. If the location of the resistors is approximately known prior to image preprocessing (e.g. using a color detector), the morphological operations could be limited to a smaller area around the possible locations of the object, and not the entire image, which would lead to additional reduction of execution time.

References

[1] K. L. Chan and H. Wang, "Reading resistor values by color image processing," in *Automatic Inspection and Novel Instrumentation*, vol. 3185. International Society for Optics and Photonics, 1997, pp. 157–168.

[2] Y. Mitani and Y. Hamamoto, "An evaluation of color features for a resistor color classification under various illumination situations," in *Fifth International Conference on Graphic and Image Processing (ICGIP 2013)*, vol. 9069. International Society for Optics and Photonics, 2014, p. 906914.

[3] Y. Mitani, Y. Sugimura, and Y. Hamamoto, "A method for reading a resistor by image processing techniques," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, 2008, pp. 433–439.

[4] Y.-S. Chen and J.-Y. Wang, "Reading resistor based on image processing," in *2015 International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 2. IEEE, 2015, pp. 566–571.

[5] ——, "Computer vision on color-band resistor and its cost-effective diffuse light source design," *Journal of Electronic Imaging*, vol. 25, no. 6, p. 061409, 2016.

[6] R. Molina, P. Federigi, V. Gil-Costa, and M. Printista, "Hybrid classification of resistors through image processing," in *2014 22nd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*. IEEE, 2014, pp. 103–106.

[7] M. F. Demir, A. Cankirli, B. Karabatak, A. Yavariabdi, E. Mendi, and H. Kusetogullari, "Real-time resistor color code recognition using image processing in mobile devices," in *2018 International Conference on Intelligent Systems (IS)*. IEEE, 2018, pp. 26–30.

[8] M. Sonka, V. Hlavac, and R. Boyle, "Image processing, analysis, and machine vision. thomson," ISBN 0-495-08252-X, Tech. Rep., 2008.

[9] M. Yang, K. Kpalma, and J. Ronsin, "A survey of shape feature extraction techniques," 2008.