Verification of color characteristics of document images captured in uncontrolled conditions

I.A. Kunina [1,2], O.A. Padas [2,3], O.A. Kolomyttseva [4]
[1] Institute for Information Transmission Problems of RAS (Kharkevich Institute), 127051, Moscow, Russia, Bolshoy Karetny per. 19, build.1;
[2] Smart Engines Service LLC, 117312, Moscow, Russia, pr. 60-letiya Oktyabrya 9;
[3] Moscow Institute of Physics and Technology (State University), 141701, Dolgoprudny, Russia, Institutskiy per. 9;
[4] Neapolis University Paphos, Paphos 8042, Cyprus,2 Danais Avenue

Abstract

This paper examines a presentation attack when a color photo of a gray copy of a document is presented instead of the original color document during remote user identification. To detect such an attack, we propose an algorithm based on the comparison of chromaticity histograms of presented color images of the document and the ideal template of this type of document. The chromaticity histograms of the original document and the template are expected to be quite identical, while the histograms of the gray copy of the document and the template would be different. The algorithm was tested on images from the open dataset DLC-2021, which contains color images of synthesized identity documents and color images of their gray copies. The precision of the proposed method was 98.99 %, the recall was 84.7 %, that gave 8 times fewer errors than the baseline provided by authors of DLC-2021.

Keywords: document analysis, document liveness detection, presentation attack detection, gray copies detection, chromaticity.

Citation: Kunina IA, Padas OA, Kolomyttseva OA. Verification of color characteristics of document images captured in uncontrolled conditions. Computer Optics 2024; 48(4): 554-561. DOI: 10.18287/2412-6179-CO-1385.

Introduction

The functionality of modern document recognition systems has long gone beyond data extraction tasks. As optical character recognition (OCR) becomes more and more widespread in various fields, new requirements are imposed on recognition systems. For example, modern OCR systems are tasked with verifying the authenticity of the presented document. Among different approaches there are checking for the presence and the right location of the government seal stamps [1] and holograms [2], the conformance of the fonts to the government standards [3], etc.

At the same time, the demand for the recognition of documents from various sources captured using mobile devices is ever-growing. A set of measures aimed at detecting attempts to use a photocopy, mock-up, or an image of a document captured from a monitor screen is referred to as document liveness detection, by analogy to face liveness detection [4], which is aimed at detecting a presentation attack (presenting 3D masks, photos or video recordings) in facial recognition systems.

In this paper, we propose a method to verify the color characteristics of a document captured with a mobile device. The main drawback of such a process is the lack of control over the capturing environment, in contrast to using specialized scanners (see examples of input documents in question in Fig. 1*a* – *d* and compare them with the template in Fig. 1*e*). Moreover, a heterogeneous and uncontrolled background can be an obstacle to accurate document localization, which results in background capture (see Fig. 1*c*). All these factors affect the observed color of the document (see representation of the chromaticity of the input document images in Fig. 1*f*–*j* where color is represented by the proportion of red, green, and blue coordinates regardless of its luminance). Therefore, to identify a gray copy of the document, it is not enough to rely on the zero or low saturation of the input document image, which is typical for achromatic colors: white, black, and gray.

The main idea of the proposed algorithm is to compare the chromaticity of the input color image of the document (see examples in Fig. 1*a* – *d*) with the chromaticity of the document template (see Fig. 1*e*). The algorithm takes as an input a projectively restored document image and a document template image that sets the ideal color distribution of the projective undistorted document image. In order to be resistant to diverse lightning conditions, the input document and template images are preprocessed in a special way (see Section 2.2). In order to be resistant to document localization errors, we do not compare the images of the document in question and the template image directly, comparing their chromaticity histograms instead (see Sections 2.3 – 2.5) discarding the information about spatial distribution of colors within the document. The difference between the histograms is analyzed, and the conclusion on document liveness is obtained.

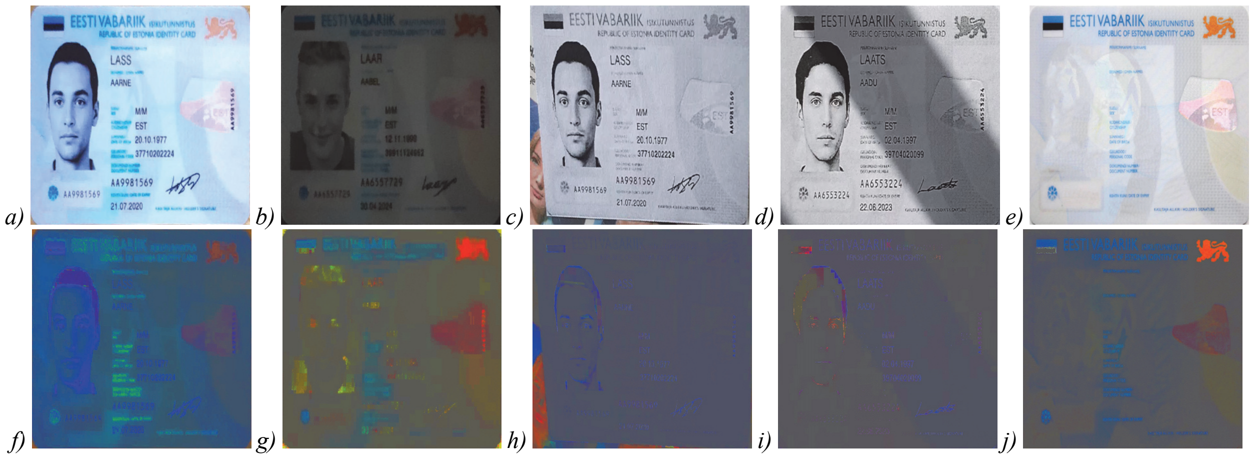

Fig. 1. Examples of input data from the open dataset DLC-2021: color images of (a)-(b) are "real" documents and (c)-(d) are gray copies of "ID Card of Estonia" type, (e) the document template, (f)-(j) a visual representation of the chromaticity of the corresponding images where each source pixel has been scaled so that the total red, green, and blue coordinates make up 255. Note the effect on the dark and shadowed regions. For the convenience of visualization, the saturation of (f)-(j) was tripled

1. A review of modern methods for detecting document recapture

For single document images, detection of color physical copies and screen recaptures has been considered by various researchers [5, 6, 7]. In [5] the authors proposed to detect certificate document images printed and recaptured by scanners/cameras based on differences in texture and reflectance characteristics of bronzing characters. The authors of [6] presented the print-and-scan distortion model of color halftone dots in the spatial and spectral domains and proposed the method to distinguish a genuine document (obtained by a print-and-scan process) from a recaptured document (obtained by a double print-and-scan process). Both of the proposed methods were evaluated with high quality image datasets. In [7] the authors proposed a pixel-wise supervision on DenseNet to detect screen recaptures and color copies. A dataset was constructed using German ID cards and residence permits cropped from the frames and consisted of bona fide ID cards, color copies and screen recaptures. To train the neural network, documents were cropped from the frames. According to failure analysis in [7], the proposed approach classify the bona fide ID card as recaptured one when the presented ID card is not well aligned with the presence of fingertips in the background.

The authors of [2, 8] proposed to analyze the visual changes in holographic elements in the video stream, which are not exhibited by the physical copy. In [8] the authors proposed to analyze the diversity of values per pixel in the RGB space, the authors of [2] selected pixels based on their saturation and value in the HSV space and analyzed the connected components of such pixels based on their shapes and hue diversity.

Due to the regulation of personal data protection, there is an extremely limited number of identity documents datasets with open access: LRDE Identity Document Image Database (LRDE IDID) [9], Brazilian Identity Document Dataset (BID Dataset) [10], Mobile Identity Document Video dataset (MIDV-2020) [11]. Among the open datasets which include not only "real" documents, but also their copies, only Document Liveness Challenge (DLC-2021) [12] includes color photographs of mock documents as originals and their gray, color copies and screen recaptures.

Based on the literature review, we decided to employ the DLC-2021 [12] dataset as the only public dataset dedicated to the specific problem considered in the paper. The DLC-2021 dataset includes images of mock documents as originals and images of their gray copies, all captured in different conditions. In addition to the data themselves, the authors published a baseline algorithm to distinguish between color images of "real" documents and their gray copies, which can be used for comparison.

2. Method

2.1. General description of the proposed algorithm

The general algorithm for the verification of color characteristics of an input document image is illustrated in Fig. 2. The detector receives a color image of a document of a known type and an image of a document template of this type (see examples of input data in Fig. 1). The image of the document is a square image with a predefined size of 600×600, the upper left vertex of which is at the origin of the image coordinates, and the document boundaries are parallel to the image ones. A template sets the ideal color distribution and is anisotropically scaled to the size of the input document image. The detector relies on the comparison of chromaticity histograms of a preprocessed document image and the corresponding template to decide whether the document image is an image of an original color document or its gray copy. Preprocessing minimizes the impact of the color and intensity of the light source on the document image. Similarity of histograms is evaluated by analyzing the difference between the histograms of the template and document images.

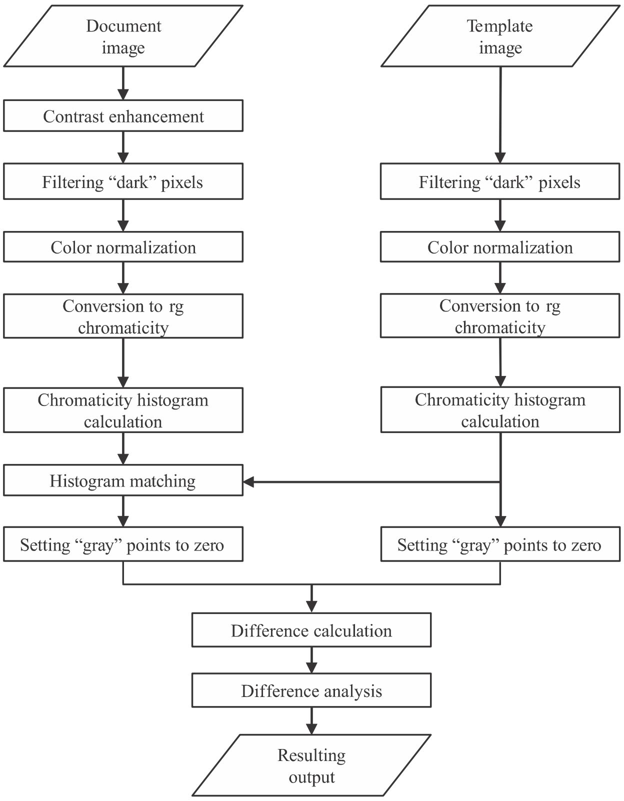

Fig. 2. General diagram of the algorithm for the verification of color characteristics of an input document image

2.2. Input image preprocessing

2.2.1. Contrast enhancement

The contrast value depends on the image capturing conditions. Therefore, firstly, we fix the dynamic range of the pixel values of the input image I_{in}. For this, in each channel independently, we map 1% of the pixels with the lowest values to 0 and 1% with the highest values to 255.

Note that if the original range is relatively small, then stretching the range will increase the noise level, and as a consequence, the signal amplitude of false colors generated by the noise will increase. Therefore, we will additionally apply the restriction to the range extension of a maximum of 3 times with such a shift of values so that the maximum is 255. Let us denote the result of the contrast enhancement by I_{contr}.

2.2.2. Filtering "dark" pixels

The darker the area in the image, the more unstable the chromaticity in that area due to the noise effects. Therefore, to improve the robustness of the method, let us filter out the dark pixels in the image.

Let I_{bg} denote the result of the application of dilation with square structuring element of 60×60 to I_{contr}. Here, the size of the structuring element was based on the maximal size of the observed dark objects that could appear on a document image. To perform a faster calculation of the window maximum, we employed the van Herk/Gil-Werman algorithm [13, 14].

Let us replace 15 % of pixels (corresponding to the estimated content of the document) within I_{contr} of the lowest brightness in terms of the Hexcone HSV model [15] $V = \max\{R, G, B\}$ with the corresponding pixels of I_{bg}. Let us denote the resulting image by I_f.

Such filtering was chosen instead of morphological closing to preserve small colored elements (e.g. color static text) presented in documents.

2.2.3. Color normalization

To compensate the light source color influence, we apply color normalization to the image I_f under the Gray World Assumption [16]:

$$R_{gw} = R \cdot \frac{L}{\overline{R}} \quad G_{gw} = G \cdot \frac{L}{\overline{G}}, \quad B_{gw} = B \cdot \frac{L}{\overline{B}} \quad L = \frac{\overline{R} + \overline{G} + \overline{B}}{3}, \quad (1)$$

where $\overline{R}, \overline{G}, \overline{B}$ are the mean values in each of the I_f image R, G, B channels correspondingly.

2.2.4. Conversion to rg chromaticity

To compensate the effect of varying intensities from the light source, let us convert the color coordinates $R_{gw}G_{gw}B_{gw}$ to the chromaticity coordinates rgb as follows:

$$r = \frac{85 \cdot R_{gw}}{R_{gw} + G_{gw} + B_{gw}} \quad g = \frac{85 \cdot G_{gw}}{R_{gw} + G_{gw} + B_{gw}} \quad b = 255 - r - g. \quad (2)$$

Here, achromatic colors correspond to (85, 85, 85).

Due to linear dependence of the r, g, b coordinates, only the coordinates r, g will be considered in the histograms.

The example of the results of the described preprocessing steps for the input images is shown in Fig. 3. Compare the final result (see Fig. 3$k-o$) with the representation of the chromaticity of the input document images in Fig. 1$f-j$.

2.3. Calculation of chromaticity histogram and regions of interest setup

Let us calculate the histograms H_s^{rg} and H_t^{rg} for normalized images of a document and its template correspondingly. The values of $H^{rg}(g, r)$ correspond to the number of pixels with the values of $(r, g, 255-r-g)$ in an image. The examples of calculated histograms for normalized images of input documents (Fig. 4$a-d$) and the template (Fig. 4e) are visualized in Fig. 4$f-i$ and Fig. 4n correspondingly. Here, the zero value in the histogram corresponds to gray, and each non-zero value corresponds to some color from the color schema. The origin of the histogram coordinates is in the upper left corner, the r-axis is directed downward, and the g-axis is directed to the right. The neutral (gray) point on the chromaticity histogram corresponds to the point with coordinates (85, 85).

In each calculated histogram, the regions of interest $R_i = (g, r, w, h)$ are selected. These regions are set by the coordinates of the upper left corner (g, r), and the horizontal and vertical dimensions (w and h, respectively). Since the hue within the vicinity of the gray point is poorly defined, we will define a neutral color by the region:

$$R_{gray} = (85-\varepsilon, 85-\varepsilon, 2\cdot\varepsilon+1, 2\cdot\varepsilon+1), \tag{3}$$

where $\varepsilon = 7$.

The regions of the chromaticity histogram bounded by the neutral point will be denoted according to the dominant hue: R_{blue}, R_{green}, R_{red}, R_{yellow} are blue, green, red and yellow regions, respectively:

$$\begin{aligned} R_{blue} &= (0,0,85,85), \\ R_{green} &= (85,0,170,85), \\ R_{red} &= (0,85,85,170), \\ R_{yellow} &= (85,85,85,85). \end{aligned} \tag{4}$$

Fig. 3. The preprocessing steps for input images: (a)-(d) input document images in question, (e) document template, (f)-(j) the result of the sequential contrasting, filtering "dark" pixels and color normalization, (k)-(o) the result of the input image processing (for the convenience of visualization, the saturation of (k)-(o) was tripled)

2.4. Chromaticity histograms processing

2.4.1. Histogram matching

Due to differences in document printing and capturing conditions, the chromaticity histogram of the input original document may differ from the chromaticity histogram of the document template. Therefore, we will transform the histogram H_s^{rg} so that it matches the histogram of the template H_t^{rg}.

Therefore, let us project the calculated histograms H_s^{rg} and H_t^{rg} independently on the axes r and g by integrating them horizontally and vertically, respectively. We denote the results of the integration by $H_s^{\varkappa}$ for H_s^{rg}, and $H_t^{\varkappa}$ for H_t^{rg}, where $\varkappa \in \{r, g\}$.

Let us calculate the transformation as follows: each v will be mapped to v' so that:

$$\upsilon' = T^{\varkappa}(\upsilon) = \begin{cases} \upsilon, |85-\upsilon| < |\nu-\upsilon| \\ \nu, |85-\upsilon| \geq |\nu-\upsilon| \end{cases}, \tag{5}$$

$$\nu = [G_t^{\varkappa}]^{-1}(G_s^{\varkappa}(\upsilon)),$$

$$x = G_\kappa^{\varkappa}(i) = \sum_{j=0}^{i} H_\kappa^{\varkappa}(j), \tag{6}$$

$$i = [G_\kappa^{\varkappa}]^{-1}(x).$$

where $G_\kappa^{\varkappa}(i)$ indicates the number of pixels of $H_\kappa^{\varkappa}, \kappa \in \{s, t\}$ with values less than or equal to i. The condition (5) limits shifting of "gray" points within histogram $H_s^{\varkappa}$ to greater saturation and prevents "color" points from changing the regions.

Let us match H_s^{rg} to H_t^{rg}. For this, we create a blank (where all values are equal to zero) histogram $\hat{H}_s^{rg}$ and perform the following operation for each pixel (g, r) of the histogram H_s^{rg}:

$$\hat{H}_s^{rg}(g', r') = \hat{H}_s^{rg}(g', r') + H_s^{rg}(g, r), \tag{7}$$

where

$$\begin{aligned} r' &= T^r(r), \\ g' &= T^g(g). \end{aligned} \tag{8}$$

A function $T(r)$ could be a monotonically increasing, therefore we use addition of the previous value to the new one in (7).

Examples of histograms in Fig. 4$f-i$ after matching to the histogram of the template are illustrated in Fig. 4$j-m$ correspondingly.

2.4.2. Setting the "gray" points to zero

The last step of processing the histograms $\hat{H}_s^{rg}$ and H_t^{rg} is setting the points in the histograms within R_{gray} to zero, focusing on the analysis of other histogram values within regions $R_{green}, R_{blue}, R_{red}, R_{yellow}$. The resulted histograms $\hat{H}^{rg}_{sz}$ and H^{rg}_{tz} are shown in Fig. 4*o-r* for histograms in Fig. 4*j-m* correspondingly, and in Fig. 4*s* for the histogram of the template in Fig. 4*n*.

In Fig. 5 the chromaticity images relative to the position of chromaticity coordinates in the regions of the histograms were shown for the input document images as well as for the preprocessed ones and for changed ones to correspond to the matched histograms. As it is shown in Fig. 5, the proposed transform of histograms (5) prevented matching the histograms of gray copies to the histogram of the template, but adjusted the position of red pixels near the gray region for Fig. 5*a*.

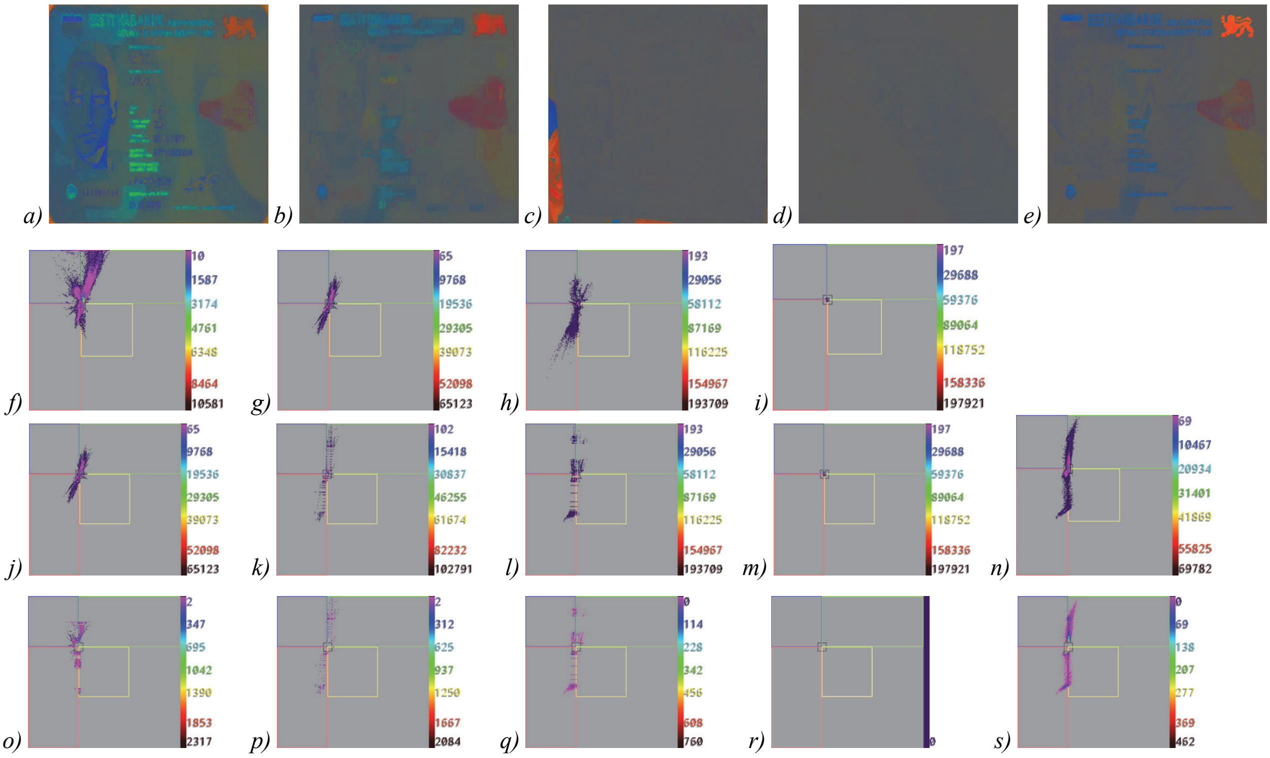

Fig. 4. An example of a chromaticity histogram processing for an input color document image: (a)-(d) the normalized images of the input documents in question (for the convenience of visualization, the saturation was tripled), (f)-(i) the calculated histogram of the normalized image, (j)-(m) the histogram after matching to the template histogram, (o)-(r) the matched histogram with the zeroed out R_{gray} *region, (e), (n), (s) correspond to the same steps for template image excluding the matching step*

2.5. Analysis of histograms of document and template images

The formal procedure to reach the conclusion on the color of the document in the input image is as follows:

1. subtract the result of the dilation of the histogram $\hat{H}^{rg}_{sz}$ from the histogram H^{rg}_{tz} preserving only non-negative values of the resulted histogram H_d (see Fig. 6);
2. calculate the sums of t_i values in the histogram H^{rg}_{tz} within the regions $R_i \in R = \{R_{green}, R_{blue}, R_{red}, R_{yellow}\}$;
3. calculate the sums of d_i values in the histogram H_d within the regions R_i;
4. check the following conditions within the regions R_i:

(a) $\Delta_{t_i} = \dfrac{t_i}{\sum_{r \in R} t_r} > t_\varepsilon$;

(b) $\Delta_{d_i} = \dfrac{d_i}{t_i} > d_\varepsilon$;

5. consider the document as a gray copy if conditions (a – b) are simultaneously satisfied for at least one region.

Here, the condition (a) sets the color as dominant, and the condition (b) sets the unacceptable residual amount of the color. Dilatation of the histogram itself is used to solve the problem of sparse histograms originating from pale (partly gray) objects of input documents. We do not apply dilation only within the regions to work stable on boundary colors.

In the experiments shown below, t_ε was set to 0.2, and d_ε was set to 0.4. Parameters were selected via experiments.

3. Testing of the proposed algorithm

The algorithm was tested on subsets of the DLC-2021 [12] image dataset containing photographs of mock "real" identity documents of 10 different types from the MIDV-2020 collection [11] (denoted in the DLC-2021 dataset by an abbreviation *or*) and photographs of their gray copies (denoted in the DLC-2021 dataset by an abbreviation *cg*). The *or* subset contains 16264 images and the *cg* subset contains 13965 images. Projective distortions of the document images were preliminarily neutralized using the document corners position in the frame from the given annotations of the corresponding images. The used document templates were created by selecting template images from MIDV-2020 (one image per a type) and removing document contents in an image editor after that. The created templates are shown in Fig. 7.

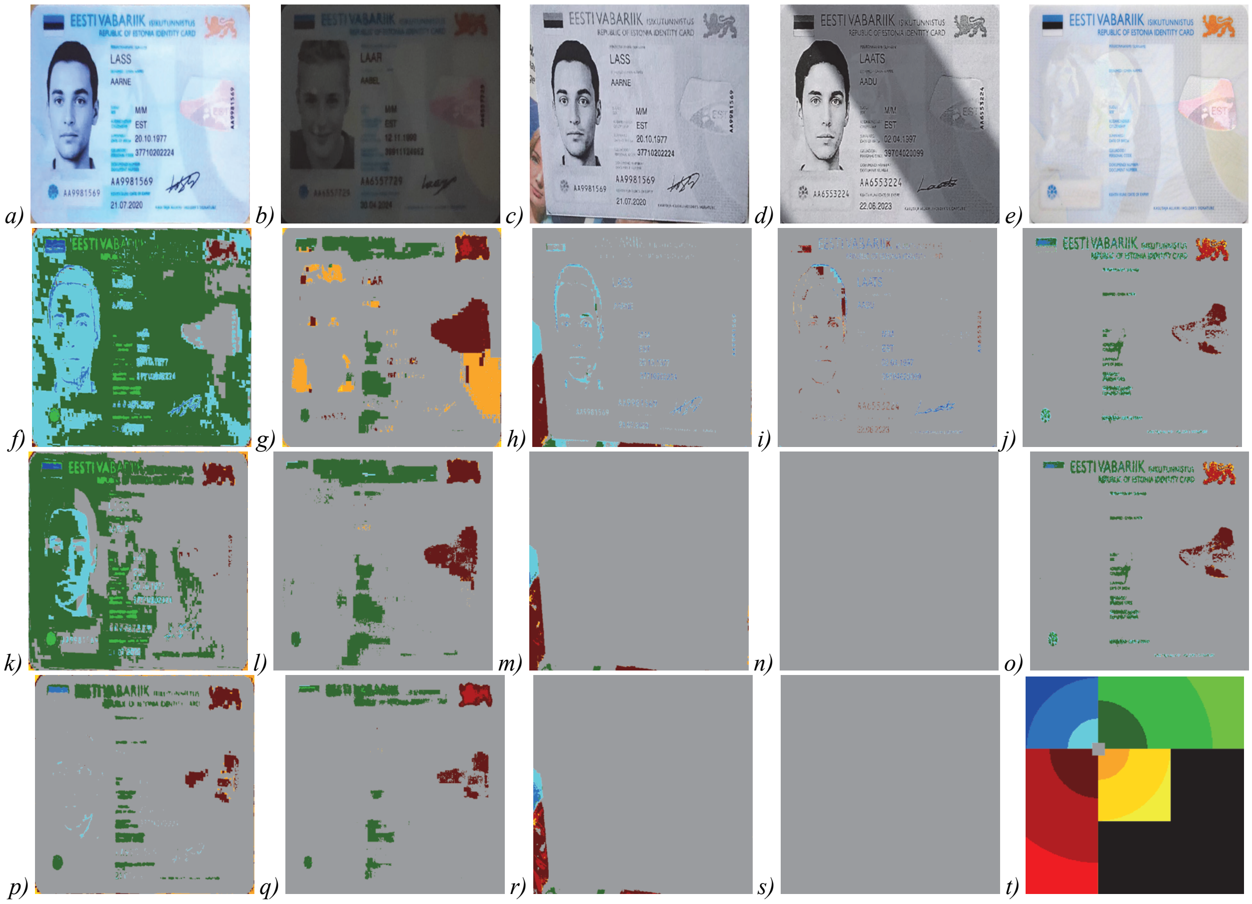

Fig. 5. An example of a chromaticity distribution changing for input images: (a)-(d) input document images in question, (e) input template image, (f)-(j) the chromaticity distribution of input images, (k)-(o) the chromaticity distribution of input preprocessed images, (p)-(s) the chromaticity distribution of input document images after histogram matching. The colors of pixels for images (f)-(s) correspond to some color from the color schema (t) depending on the chromaticity values

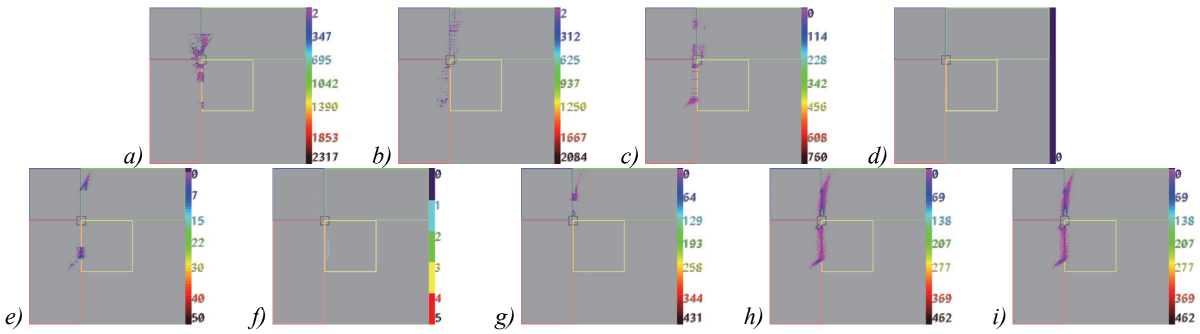

Fig. 6. An example of the difference between the histograms of the template and the color document: (a)-(d) the preprocessed document histograms, (i) the preprocessed template histogram, (e)-(h) the difference between the preprocessed template histogram and corresponding document histograms

The performance of the proposed algorithm from the given subsets is shown in Table 1 and labeled as v1. In addition, two more experiments were carried out. The first experiment involved omitting the histogram matching step (labeled in Tab. 1 as v2). The second one involved replacing the difference of histograms with the difference of sums of pixel values within corresponding chromaticity histograms regions (labeled in Tab. 1 as v3). The results of the comparison illustrated in Tab. 1 show that the best modification is the whole version of the proposed algorithm (v1) providing for the best accuracy among all modifications.

Also, we compared the quality of the proposed algorithm with the baseline results provided in [12]. In the beginning, to distinguish “real” documents and their gray copies, the authors of DLC-2021 trained a classification CNN model based on ResNet-50 architecture. However, all their experiments with ResNet-50-like models and more simple CNN models were failed. Since the development of a more sophisticated CNN architecture was beyond the scope of [12], the authors of DLC-2021 examined the Scikit Dummy Classifier detector on the validation dataset in order to provide a simple baseline. They presented numerical results of Scikit Dummy Classifier detectors with different strategies: the "constant" one (generating constant prediction), the "stratified" one (generating predictions with respect to the balance of training set classes), and the "uniform" one (generating predictions uniformly at randomly). The results were obtained for images from *graycopy_negative_test.lst* (16264 images, which corresponds to the entire *or* subset) and *graycopy_positive_test.lst* (10473 images from *cg*).

Fig. 7. The used templates: (a) ID Card of Albania, (b) ID Card of Spain, (c) ID Card of Estonia, (d) ID Card of Finland, (e) ID Card of Slovakia, (f) Passport of Azerbaijan, (g) Passport of Greece, (h) Passport of Latvia, (i) Internal passport of Russia, (j) Passport of Serbia

Tab. 1. Comparison of performance of different versions of the proposed method. The method demonstrating the best accuracy is shown in bold

dataset	algorithm	recall, %	precision, %	accuracy, %
or, *cg*	**the whole version of the proposed algorithm (v1)**	84.7 %	98.99 %	**92.54 %**
	the proposed algorithm without histogram matching (v2)	81.9 %	81.3 %	85.5 %
	simple colors amount difference (v3)	91.3 %	73.9 %	81.6 %

In order to compare the quality of the proposed algorithm with the numerical results of [12] we calculated the accuracy of the best version of the proposed algorithm for images from *graycopy_negative_test.lst* and *graycopy_positive_test.lst* too. The numerical results shown in Tab. 2 demonstrate that the proposed algorithm gave 8 times fewer errors (in terms of accuracy) than the baseline provided by authors of DLC-2021.

Tab. 2. Comparison of performance of the proposed method and the baseline algorithm provided by the authors of DLC-2021. The method demonstrating the best accuracy is shown in bold

dataset	algorithm		recall, %	precision, %	accuracy, %
graycopy_negative_test.lst *graycopy_positive_test.lst*	**the proposed algorithm (v1)**		89.4 %	98.7 %	**95.4 %**
	the baseline of [12] (Dummy Classifier)	const = false	0.00 %	–	60.83 %
		const = true	100.00 %	39.17 %	39.17 %
		stratified	77.82 %	39.22 %	44.06 %
		uniform	50.11 %	39.26 %	50.09 %

4. Analysis of errors of the proposed algorithm

The false positive cases produced by the proposed detector were mainly associated with overexposed document images (see the example in Fig. 8*a*), due to the loss of some of the document color information because of the overexposure, as well as with the light source saturation effect (see the example in Fig. 8*b*). In the latter case, the problem emerged at the color correction stage, because the color correction algorithms can not fully compensate the influence of so saturated light source.

False negative cases were associated with the obscured regions of the document by elements similar to the document template in terms of chromaticity. For example, the document of "Passport of Latvia" type in Fig. 8*c* is a false negative case due to the fact that the fingers on the gray copy added the red hue native to this document type: it is present in the document template in the static text and ornamentation (see the document template in Fig. 7*h*). In another example in Fig. 8*d* a green object obscures the document area. It added a green hue to the document image, which matches the chromaticity of the static text in the top half of the document template (see the template in Fig. 7*f*).

Conclusion

We proposed an algorithm for verifying the color characteristics of an input document image during remote user identification. The main idea of the method is to compare the chromaticity histograms of document images and the document template of the corresponding document type. The algorithm was tested on DLC-2021 subsets containing more than 30000 color photos of mock "real" documents and their gray copies. The testing showed outperforming the baseline by the proposed method in terms of accuracy. Future research includes the analysis of the applicability of the proposed algorithm for color copies detection and analysis of the algorithm stability on the method of creating the template.

a) b) c) d)

Fig. 8. Examples of errors of the proposed algorithm

References

[1] Matalov DP, Usilin SA, Arlazarov VV. Modification of the Viola-Jones approach for the detection of the government seal stamp of the Russian Federation. Proc SPIE 2019; 11041: 110411Y.

[2] Kada O, Kurtz C, van Kieu C, Vincent N. Hologram detection for identity document authentication. In Book: Yacoubi ME, Granger E, Yuen PC, Pal U, Vincent N, eds. Pattern recognition and artificial intelligence: Third international conference, ICPRAI 2022, Paris, France, June 1-3, 2022, Proceedings, Part I. Cham: Springer International Publishing; 2022: 346-357.

[3] Chernyshova YS, Aliev MA, Gushchanskaia ES, Sheshkus AV. Optical font recognition in smartphone-captured images and its applicability for ID forgery detection. Proc SPIE 2019; 11041: 110411J.

[4] Apgar D, Abid MR. Survey of face liveness detection for unsupervised locations. 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) 2021: 0162-0168.

[5] Yan J, Chen C. Cross-domain recaptured document detection with texture and reflectance characteristics. 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) 2021: 1708-1715.

[6] Chen C, Zhao L, Yan J, Li H. A distortion model-based pre-screening method for document image tampering localization under recapturing attack. Signal Process 2022; 200: 108666.

[7] Mudgalgundurao R, Schuch P, Raja K, Ramachandra R, Damer N. Pixel-wise supervision for presentation attack detection on identity document cards. IET Biometrics 2022; 11(5): 383-395.

[8] Hartl A, Arth C, Schmalstieg D. AR-based hologram detection on security documents using a mobile phone. In Book: Bebis G, Boyle R, Parvin B, Koracin D, McMahan R, Jerald J, Zhang H, Drucker SM, Kambhamettu C, Choubassi M, Deng Z, Carlson M, eds. Advances in visual computing. Part II. Cham: Springer International Publishing Switzerland; 2014: 335-346.

[9] Ngoc MÔV, Fabrizio J, Géraud T. Saliency-based detection of identy documents captured by smartphones. 2018 13th IAPR Int Workshop on Document Analysis Systems (DAS) 2018: 387-392.

[10] de Sá Soares A, das Neves Junior RB, Bezerra BLD. BID Dataset: a challenge dataset for document processing tasks. 2020 Anais Estendidos do XXXIII Conference on Graphics, Patterns and Images 2020: 143-146.

[11] Bulatov KB, Emelyanova EV, Tropin DV, Skoryukina NS, Chernyshova YS, Sheshkus AV, Usilin SA, Ming Z, Burie J, Luqman M, Arlazarov VV. MIDV-2020: A comprehensive benchmark dataset for identity document analysis. Computer Optics 2022; 46(2): 252-270. DOI: 10.18287/2412-6179-CO-1006.

[12] Polevoy DV, Sigareva IV, Ershova DM, Arlazarov VV, Nikolaev DP, Zuheng M, Muhammad ML, Burie J. Document Liveness Challenge dataset (DLC-2021). J Imaging 2022; 8(7): 181. DOI: 10.3390/jimaging8070181.

[13] Van Herk M. A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels. Pattern Recogn Lett 1992; 13(7): 517-521.

[14] Gil J, Werman M. Computing 2-D min, median, and max filters. IEEE Trans Pattern Anal Mach Intell 1993; 15(5): 504-507.

[15] Smith AR. Color gamut transform pairs. ACM Siggraph Comput Graph 1978; 12(3): 12-19.

[16] Finlayson GD, Schiele B, Crowley JL. Comprehensive colour image normalization. In Book: Burkhardt H, Neumann B, eds. Computer vision – ECCV'98. Part I. Berlin, Heidelberg: Springer; 1998: 475-490.

Authors' information

Irina Andreevna Kunina, (b. 1992), graduated from Moscow Institute of Physics and Technology (State University) in 2015. Ph.D in Technical Science, is a researcher in the vision systems laboratory at the Institute for Information Transmission Problems. Research interests are image analysis, computer vision, camera calibration.
E-mail: kunina@iitp.ru, i.kunina@smartengines.com

Olga Alexandrovna Padas, (b. 2001), graduated with a bachelor's degree from Moscow Institute of Physics and Technology (MIPT), is a programmer at Smart Engines Service LLC since 2022. Research interests are image processing, computer vision. E-mail: padas.oa@phystech.edu

Olga Andreevna Kolomyttseva, (b. 2001), studied at Moscow State University majoring in Mathematics until 2022, since 2023 transferred to the Neapolis University Pafos currently majoring at the program Applied Computer Science. In 2021-2022, Olga has worked as a programmer at Smart Engines Service LLC. Research interests are machine learning, neural networks, computer vision. E-mail: o.kolomyttseva@nup.ac.cy

Received June 30, 2023. The final version – November 22, 2023.