MSC 68T10, 68W32 **DOI: 10.14529/mmp190307**

A METHOD TO REDUCE ERRORS OF STRING RECOGNITION BASED ON COMBINATION OF SEVERAL RECOGNITION RESULTS WITH PER-CHARACTER ALTERNATIVES

K.B. Bulatov, Institute for Systems Analysis, Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, Moscow, Russian Federation, hpbuko@gmail.com

We consider the problem on recognition of a string object presented in several video stream frames. In order to maximize the output accuracy, we combine several results of the recognition. To this end, we consider a model of result of a string object recognition. The model takes into account the estimations of alternative results of per-character classification. Also, we propose an algorithm to combine results of a string recognition according to this model. The algorithm was evaluated on a MIDV-500 dataset of document images. The experimental results show that the proposed algorithm allows to achieve the high accuracy of recognition result due to an analysis of several images, and the use of the estimations of alternative results of per-character classification gives the higher results then a combination of strings that contain only the final alternatives of each character.

Keywords: recognition in video stream; mobile OCR; recognition algorithms.

Introduction

High-precision and high-speed recognition of objects in images and video stream is of particular interest for a wide range of researchers in the recent years [1–3]. The nontrivial problem is to recognize such objects as text paragraphs, document fields, etc. In particular, this problem takes place, if the image source is a hand-held camera of a mobile device. Among disadvantages of such images we note motion blur, defocus, glares on reflective surface, camera resolution, which is insufficient for accurate OCR (Optical Character Recognition), etc. [4, 5]. Fig. 1 gives an example of a glare on reflective surface of a document, as well as the impact of the glare on the text field images obtained from the video stream frames.

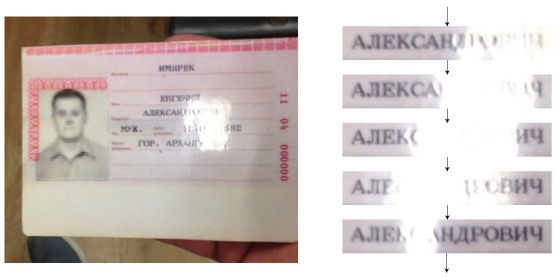

Fig. 1. An example of a glare on reflective surface of a document (left) and the text field images obtained from the video stream frames (right). Images are taken from MIDV-500 dataset [6] (clip HA39 field 3)

One of advantages of the video stream recognition is the possibility to process several frames in real time and, therefore, to mitigate disadvantages of single-frame object recognition. In other words, the same object is recognized several times in different video frames, and, therefore, the overall recognition accuracy increases. Note that the selection of a single best recognition result is not useful strategy in some cases, for example, if there is no video stream of a document having the frame with fully visible and recognizable object. Therefore, it is necessary to investigate by the method of combination of several recognition results.

A wide range of works is devoted to the problem on combination of the results obtained by different recognition systems for the same input [7, 8]. In some sense, this problem is similar to the problem on combination of several results of recognition of the same object by different inputs. However, most of published papers considers the result of a single indivisible object recognition and, therefore, deals with the model of recognition result as a result of division of an input object into a certain number of classes. At the same time, as a rule, the problem on composite objects recognition requires to represent a recognition result as a sequence of classification results, such as in the case of text string recognition, where each character is recognized separately. There exists a number of papers devoted to combination of results of a string object recognition. Most of these papers is based on the ROVER approach [9]. For the first time, this approach was proposed in order to recognize a speech. Later, the ROVER approach was used for optical recognition of printed [10] and handwritten [11] text strings. At the same time, these works consider the model of result of a string object recognition as a string of characters (with the estimation of confidence of overall string) and do not use the extended hypothesis model, which takes into account the per-character alternatives. However, the paper [12] shows that the extended hypothesis model allows to increase the accuracy of text strings recognition due to the use of language models. According to the paper [11], the ROVER framework can be underexploited in the field of string object recognition. The paper [13] also considers the problem on combination of results of a string object recognition, but does not give the formal problem statement, the sufficiently complete description of the algorithm, and the information on the impact of the extended model of per-character result.

The goal of this paper is to construct a model of result of a string object recognition, which takes into account the per-character alternatives. Also, based on the model, we follow the ROVER architecture in order to construct an algorithm to combine the results of a string object recognition. Section 1 describes the model of result of both a single object classification and a string object recognition. The model is used to construct the algorithm in Section 3. Section 2 states the problem on combination of results of a string object recognition. Section 3 describes the proposed algorithm. Section 4 presents an experimental investigation of the algorithm performed on the basis of the MIDV-500 dataset [6], which consists of video clips of 50 samples of various identity document types (10 video clips per each document type, where each video clip consists of 30 frames) with ground truth containing ideal positions and values of text fields.

1. Model of Result of String Object Recognition

In order to construct a model of result of a string object recognition, first of all we consider the corresponding model for a single object. Suppose that K possible classes of

objects form the set $C = \{c_1, c_2, \ldots, c_K\}$, and it is necessary to determine a class that contains the image I of some object c. To this end, we use the module f of a single object classification. In the classical problem statement, the result is one of the classes $f(I) = c_f$, where $c_f \in C$, and the problem on a single object classification is to maximize a posteriori probability that the class c_f coincides with the true class c (provided by some dataset). In the general problem statement, the classification module $\hat{f}$ associates the input image I with the set of pairs $\hat{f}(I) = \{(c_1, q_1), (c_2, q_2), \ldots, (c_K, q_K)\}$, where q_i is the membership estimation of the fact that the object belongs to the class c_i. The final result of a single object classification is a class corresponding to the maximal membership estimation:

$$f(I) = \arg\max\{\hat{f}(I)\} \in \left\{ c_f \middle| \left((c_f, q_f) \in \hat{f}(I)\right) \wedge \left(q_f = \max_{(c,q)\in\hat{f}(I)} q\right) \right\}. \tag{1}$$

If there exist several pairs $(c_{f_1}, q_I), (c_{f_2}, q_I), \ldots$ with the same maximal membership estimation, then an additional convention is established in order to uniquely determine the class. For example, we can consider the result to be the class with the maximal membership estimation and the minimal index in the set C. Model of result of a single object classification (1) can be considered as a variant of the model of result of the algorithms to compute membership estimations [14] and is widely used in the methods of optical image recognition based on the convolutional neural networks [15].

In order to define the result of a string object recognition, we need to introduce a zero-length "null string" λ (an empty class) as a possible alternative of a single object classification. By the extended result of a single object classification we mean the mapping $a : C \cup \{\lambda\} \to [0, 1]$ from the set of classes and the empty class λ to the set of membership estimations. Each membership estimation is considered to be a real number in the interval $[0, 1]$, and the sum of all membership estimations is equal to 1 in each mapping. Therefore, we define the set of all possible results of the single object classification $\hat{C}$:

$$\hat{C} \stackrel{\text{def}}{=} \left\{ a \in [0,1]^{C\cup\{\lambda\}} \middle| \sum_{c\in C\cup\{\lambda\}} a(c) = 1 \right\}. \tag{2}$$

On the set of all possible results of single object classification $\hat{C}$ (2), the metric can be defined as follows:

$$\rho_{\hat{C}}(a, b) \stackrel{\text{def}}{=} \frac{1}{2} \sum_{c\in C\cup\{\lambda\}} |a(c) - b(c)|, \quad \forall a, b \in \hat{C}. \tag{3}$$

It is easy to see that function $\rho_{\hat{C}}(a, b)$ (3) has all metric properties, since $\rho_{\hat{C}}(a, b)$ corresponds to a scaled taxicab metric in a vector space on the ordered set $C \cup \{\lambda\}$. The range of the function $\rho_{\hat{C}}(a, b)$ is the interval $[0, 1]$, since the sum of membership estimations is equal to 1 for any $a, b \in \hat{C}$.

Denote the "empty classification result" by $\hat{\lambda}$:

$$\hat{\lambda} \stackrel{\text{def}}{=} \{(\lambda, 1), (c_1, 0), (c_2, 0), \ldots, (c_K, 0)\}. \tag{4}$$

By the result X of a string object recognition we mean a string on the set $\hat{C} \setminus \{\hat{\lambda}\}$, i.e. the element $X \in \mathbb{X}$, where $\mathbb{X} \stackrel{\text{def}}{=} (\hat{C} \setminus \{\hat{\lambda}\})^*$. The string X is a sequence of results of a single object classification $X = x_1 x_2 \ldots x_n$, where $x_i \in \hat{C} \setminus \{\hat{\lambda}\}$. The length $|X| = n$ of the string X is the number of elements in the sequence. Denote by $X_{i\ldots j}$ a substring of X, which includes the elements $x_i x_{i+1} \ldots x_j$ for $1 \le i \le j \le n$. If $i > j$, then the substring $X_{i\ldots j}$ corresponds to the empty substring $\hat{\lambda}$ with zero length.

The elementary edit operation T on the string X is defined as a pair $(a, b) \neq (\hat{\lambda}, \hat{\lambda})$, where $a, b \in \hat{C}$, as follows. If $b \neq \hat{\lambda}$, then the element $x_i = a$ is replaced by the element b in the string X. If $b = \hat{\lambda}$, then the element $x_i = a$ is deleted from the string X. If $a = \hat{\lambda}$, then the element b is inserted in the string X.

Consider two arbitrary strings $X, Y \in \mathbb{X}$ with finite lengths. An edit transformation is defined as a sequence of L elementary edit operations $T_{X,Y} = T_1 T_2 \dots T_L$, which transforms the string X to the string Y. The weight of an edit transformation is defined as a sum of distances (in terms of metric $\rho_{\hat{C}}$ (3)) between the pairs of objects involved in the elementary edit operations $T_i = (a_i, b_i)$ of the edit transformation $T_{X,Y}$:

$$w(T_{X,Y}) \stackrel{\text{def}}{=} \sum_{i=1}^{L} \rho_{\hat{C}}(a_i, b_i). \tag{5}$$

Define a metric on the set of results of a string object recognition $\mathbb{X}$ as the minimal weight of an edit transformation which transforms the string X to the string Y:

$$\rho_{\mathbb{X}}(X, Y) \stackrel{\text{def}}{=} \min\{w(T_{X,Y})\}. \tag{6}$$

Function $\rho_{\mathbb{X}}$ (6) can be considered as one of the realizations of the Generalized Levenshtein Distance [16]. It can be shown that $\rho_{\mathbb{X}}$ has metric properties, if $\rho_{\hat{C}}$ also has such properties [17]. The following recurrent procedure allows to compute the distance $\rho_{\mathbb{X}}(X, Y)$ between two results of a string object recognition. Let $d(i, j)$ be the distance $\rho_{\mathbb{X}}(X_{1\dots i}, Y_{1\dots j})$ between the prefixes of the strings X and Y with lenghts i and j, respectively. Then

$$\begin{gathered} d(0,0) = 0, \quad d(i,0) = \sum_{k=1}^{i} \rho_{\hat{C}}(x_k, \hat{\lambda}), \quad d(0,j) = \sum_{k=1}^{j} \rho_{\hat{C}}(\hat{\lambda}, y_k), \\ d(i,j) = \min \left\{ \begin{array}{l} \rho_{\hat{C}}(x_i, \hat{\lambda}) + d(i-1, j), \\ \rho_{\hat{C}}(\hat{\lambda}, y_j) + d(i, j-1), \\ \rho_{\hat{C}}(x_i, y_j) + d(i-1, j-1) \end{array} \right\}, \end{gathered} \tag{7}$$

and $d(|X|, |Y|)$ corresponds to the target metric value $\rho_{\mathbb{X}}(X, Y)$.

Note that the maximal value of the metric $\rho_{\mathbb{X}}(X, Y)$ is $\max\{|X|, |Y|\}$, if $\rho_{\hat{C}}$ (3) is used as a metric on the set of results of a single object classification. At the same time, since $\rho_{\mathbb{X}}$ is a particular case of the Generalized Levenshtein Distance, then this metric can be normalized such as to save the properties of identity, symmetry, and triangle inequality [17]:

$$\tilde{\rho}_{\mathbb{X}}(X, Y) \stackrel{\text{def}}{=} \frac{2 \cdot \rho_{\mathbb{X}}(X, Y)}{\alpha \cdot (|X| + |Y|) + \rho_{\mathbb{X}}(X, Y)}, \tag{8}$$

where α is the maximal possible weight of elementary deletion or insertion. In the case of the weight of an edit transformation defined as (3), we have $\alpha = \max\{\rho_{\hat{C}}(a, \hat{\lambda}), \rho_{\hat{C}}(\hat{\lambda}, b), a, b \in \hat{C}\} = 1$.

Among alternative approaches to comparison of string objects we note the Dynamic Time Warping (DTW, [16, 18]). However, the classical statement of the DTW algorithm requires correspondence of the boundary elements of the compared string objects, but does not penalize insertions and deletions, and does not have metric properties (more specifically, does not guarantee that the triangle inequality is satisfied).

2. Problem on Combination of Results of String Object Recognition

Let us consider the problem on a string object recognition in a video sequence. Input of the system takes a sequence of the images $I_1, I_2, \ldots, I_N$ of the string object $\nu \in C^*$. The module $\hat{F}$ of a string object recognition associates each image with the result of recognition $\hat{F}(I_i) \in \mathbb{X}$. In framework of the considered model, we assume that the membership estimations of the empty class λ are equal to zero in the results of a single image recognition:

$$\begin{array}{ll} \hat{F}(I_i) & = X_i, \quad X_i \in \mathbb{X}, \quad X_i = x_1^i x_2^i \ldots x_{n_i}^i, \\ x_j^i(\lambda) & = 0, \quad \forall j \in \{1, \ldots, n_i\}. \end{array} \tag{9}$$

The problem is to combine the results $X_1, X_2, \ldots, X_N$ with associated weights $w_1, w_2, \ldots, w_N$ in the single result $X \in \mathbb{X}$ minimizing the distance (according to some metric) between X and the true value ν. Since $X \in \mathbb{X}$ is the string on the set $\hat{C} \setminus \{\hat{\lambda}\}$, and ν is the string on the set of classes C, then, in order to determine the distance between these strings, it is necessary to perform some additional conversion. The most natural way is to convert the true value ν to the string $\hat{\nu} \in \mathbb{X}$,

$$\begin{array}{ll} \nu & = \nu_1 \nu_2 \ldots \nu_{n_\nu}, \quad \nu_j \in C \\ \hat{\nu} & = \hat{\nu}_1 \hat{\nu}_2 \ldots \hat{\nu}_{n_\nu}, \quad \hat{\nu}_j \in \hat{C} \setminus \{\hat{\lambda}\}, \\ \hat{\nu}_j & \stackrel{\text{def}}{=} \{(\lambda, 0), (c_1, 0), (c_2, 0), \ldots, (\nu_j, 1), \ldots, (c_K, 0)\}, \end{array} \tag{10}$$

and use metric $\rho_{\mathbb{X}}(X, \hat{\nu})$ (6) (or its normalized variant $\tilde{\rho}_{\mathbb{X}}(X, \hat{\nu})$ (8)) as a distance between the combined result X and the true value ν. However, from a practical point of view, the possibility to obtain the final result of a string object recognition (by analogy with final result (1) for a single object) is important. In order to obtain the final result of a string object recognition, we can use the following two-step procedure.

1. Associate each component $x_j \in \hat{C} \setminus \{\hat{\lambda}\}$ of the combined result $X = x_1 x_2 \ldots x_{n_X}$ with either the corresponding class $c_{x_j} \in C$ with the maximal membership estimation $x_j(c_{x_j})$, or the empty class λ, if the membership estimation $x_j(\lambda)$ exceeds the predefined threshold θ:

$$\bar{x}_j = \begin{cases} \arg\max_{c \in C} x_j(c), & \text{if } x_j(\lambda) < \theta, \\ \lambda, & \text{if } x_j(\lambda) \geq \theta. \end{cases} \tag{11}$$

2. Delete all components $\bar{x}_j = \lambda$ from the string $\bar{x}_1 \bar{x}_2 \ldots \bar{x}_{n_X}$ obtained in the first step. Use the constructed string $\bar{X}_\theta \in C^*$ as the final result of a string object recognition.

We can consider the distance between the combined result X and the true value ν to be either the Levenshtein distance $\text{levenshtein}(\bar{X}_\theta, \nu)$ [16], or its normalized variant [17]:

$$\rho_L(\bar{X}_\theta, \nu) = \frac{2 \cdot \text{levenshtein}(\bar{X}_\theta, \nu)}{|\bar{X}_\theta| + |\nu| + \text{levenshtein}(\bar{X}_\theta, \nu)}. \tag{12}$$

The problem on combination of results of a string object recognition is considered in [9] in the context of speech recognition. Instead to combine the results of recognition of several images $I_1, I_2, \ldots, I_N$ by the single recognition module $\hat{F}$, the paper [9] combines the results of recognition of the single "image" I by several recognition systems $F_1, F_2, \ldots, F_N$. These two problem statements can be considered as similar ones except for the noise model. Indeed, the aim of the combination of results of a string object recognition in a video sequence is both to filter the noise component in the input images $I_1, I_2, \ldots, I_N$ (that is conditioned by inaccuracies in the input data, errors of input images preprocessing, etc.)

and to take into account the impact of this filtration on the result of an application of the recognition module $\hat{F}$. At the same time, the aim of the combination of results obtained by different recognition modules is to filter the noise introduced by the recognition modules themselves.

The approach described in [9] is called the ROVER (Recognizer Output Voting Error Reduction) and is constructed as a two-module system given in Fig. 2. At the first step, the *alignment module* transforms all input string objects to strings of equal length by performing corresponding insertions of the empty class λ in an optimal way. At the second step, the *voting module* selects a class for each string component on the basis of a linear combination of class frequencies and confidence estimations of the corresponding recognition modules.

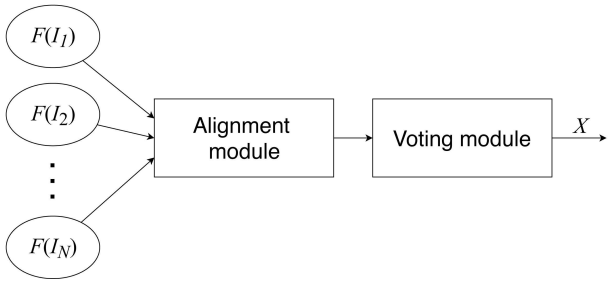

Fig. 2. Two-module system of the ROVER approach [9]

The model of result of a string object recognition used in the ROVER approach [9] is a pair of a string on the set of classes of a single object classification and a confidence estimation of a recognition module. In order to construct the algorithm to combine results of a string object recognition with the extended model of recognition result, we consider the problem statement to align strings of type (9).

Consider the input of the alignment module to be N strings $X_1, \ldots, X_N$, where $X_i \in \mathbb{X}$, and $|X_i| = n_i > 0$:

$$X_1 = x_1^1 x_2^1 \ldots x_{n_1}^1, \quad X_2 = x_1^2 x_2^2 \ldots x_{n_2}^2, \quad \ldots, \quad X_N = x_1^N x_2^N \ldots x_{n_N}^N. \tag{13}$$

In order to represent the alignment of results of a string object recognition, we introduce the function $\text{Align} : \{1, \ldots, N\} \times \{1, \ldots, \max\limits_{i=1}^{N} n_i\} \to \{1, \ldots, \sum_{i=1}^{N} n_i\}$. The function $\text{Align}(i, j)$ determines the number of the component of the "combined" result string, for which the component x_j^i provides a contribution. For each input string, the values of the function Align are different for different string components and remain the order of components: $\forall\, i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, n_i - 1\} : \text{Align}(i, j) < \text{Align}(i, j+1)$.

Also, we introduce the function $\text{Match} : \{1, \ldots, N\} \times \{1, \ldots, \sum_{i=1}^{N} n_i\} \to \hat{C}$ defined as follows:

$$\text{Match}(i, k) \stackrel{\text{def}}{=} \begin{cases} x_j^i, & \text{if } \text{Align}(i, j) = k, \\ \hat{\lambda}, & \text{if } \nexists j : \text{Align}(i, j) = k. \end{cases} \tag{14}$$

The problem on alignment is to find the alignment function Align minimizing the penalty functional given by a total pairwise distance between results of a single object classification contributing to the same components of the combined result:

$$\sum_{k} \sum_{i_1 < i_2} \rho_{\hat{C}}(\text{Match}(i_1, k), \text{Match}(i_2, k)) \to \min. \tag{15}$$

In order to generalize the voting module (see Fig. 2), which goal is to select the class for each component of the combined result, we introduce the function r that combines the results of a single object classification:

$$r : \hat{C}^N \times (\mathbb{R}_0^+)^N \to \hat{C} \setminus \{\hat{\lambda}\}. \tag{16}$$

Input of the function r consists of N results of a single object classification $a_1, a_2, \ldots, a_N$ such that $\exists i : a_i \neq \hat{\lambda}$, and a sequence of the corresponding non-negative weights $w_1, w_2, \ldots, w_N$ of contribution of each result, $\sum_{i=1}^{N} w_i > 0$.

Then, we have the following form of the function R that combines the results of a string object recognition:

$$R(X_1, X_2, \ldots, X_N, w_1, w_2, \ldots, w_N) = r_1 r_2 r_2 \ldots r_{n_R}, \tag{17}$$

where $n_R = \max\limits_{i,j} \mathrm{Align}(i, j)$, and each component of the combined string is computed by function (16) that combines the results of a single object classification. According to result of alignment (14),

$$r_j = r\left(\mathrm{Match}(1, j)), \mathrm{Match}(2, j), \ldots, \mathrm{Match}(N, j), w_1, w_2, \ldots, w_N\right). \tag{18}$$

In the general case, the exact solution to problem on alignment (15) requires the computation of dynamic programming scheme (by analogy with the computation of Generalized Levenshtein Distance (7)) with a complexity that exponentially depends on the number N of input strings. Indeed, in the computation, it is necessary to use results of the alignment of the strings $X_{11\ldots i_1}, X_{21\ldots i_2}, \ldots, X_{N1\ldots i_N}$ for all tuples formed by prefix lengths of a string recognition results $(i_1, i_2, \ldots, i_N) \in \{1, \ldots, n_1\} \times \{1, \ldots, n_2\} \times \ldots \times \{1, \ldots, n_N\})$. In order to compute this scheme, we can also use some heuristic approaches to search for the shortest path such as the A^*-search [19].

In the next section, we present the algorithm to combine the results of a string object recognition. The algorithm uses the same approximation of the alignment functional as the original ROVER approach [9].

3. Algorithm to Combine Results of String Object Recognition

Computation of combined result of a string object recognition involves a sequence of the intermediate combined results $R^{(1)}(X_1, w_1), \ldots, R^{(i-1)}(X_1, \ldots, X_{i-1}, w_1, \ldots, w_{i-1})$, where each result $R^{(i-1)}$ is used to obtain the alignment result at the i-th stage. At the first stage of the algorithm,

$$R^{(1)}(X_1, w_1) = X_1. \tag{19}$$

At each next i-th stage, construct an optimal alignment of the strings X_i and $R^{(i-1)}(X_1, \ldots, X_{i-1}, w_1, \ldots, w_{i-1})$. To this end, use a dynamic programming scheme by analogy with (7). Let $d(l, m)$ be the distance $\rho_{\mathbb{X}}\left(X_{i1\ldots l}, R^{(i-1)}(X_1, \ldots, X_{i-1}, w_1, \ldots, w_{i-1})_{1\ldots m}\right)$, and $P_p(l, m)$ be the auxiliary functions for $p \in \{1, 2, 3\}$. Compute $d(l, m)$ and $P_p(l, m)$ by the following recurrent procedure:

$$\begin{aligned}
d(0,0) &= 0, \quad d(l,0) = \textstyle\sum_{k=1}^{l} \rho_{\hat{C}}(x_k^i, \hat{\lambda}), \quad d(0,m) = \textstyle\sum_{k=1}^{m} \rho_{\hat{C}}(\hat{\lambda}, r_k^{(i-1)}),\\
P_1(l,m) &= \rho_{\hat{C}}(x_l^i, \hat{\lambda}) + d(l-1, m),\\
P_2(l,m) &= \rho_{\hat{C}}(\hat{\lambda}, r_m^{(i-1)}) + d(l, m-1),\\
P_3(l,m) &= \rho_{\hat{C}}(x_l^i, r_m^{(i-1)}) + d(l-1, m-1),\\
d(l,m) &= \min\{P_1(l,m), P_2(l,m), P_3(l,m)\}.
\end{aligned} \tag{20}$$

In order to compute the combined result $R^{(i)}(X_1, \ldots, X_i, w_1, \ldots, w_i)$ at the i-th stage, we introduce two auxiliary functions $t_X : \{0, \ldots, n_i + n_{R_{i-1}}\} \to \{1, \ldots, n_i\}$ and $t_R : \{0, \ldots, n_i + n_{R_{i-1}}\} \to \{1, \ldots, n_{R_{i-1}}\}$ computed by the following recurrent procedure:

$$
\begin{aligned}
&t_X(0) = n_i,\\
&t_R(0) = n_{R_{i-1}},\\
&t_X(k) = \begin{cases} t_X(k-1), \text{if } \begin{array}{l} P_2(t_X(k-1), t_R(k-1)) = d(t_X(k-1), t_R(k-1)) \wedge \\ \wedge P_1(t_X(k-1), t_R(k-1)) \neq d(t_X(k-1), t_R(k-1)) \end{array} \\ t_X(k-1) + 1, \text{ in other cases,} \end{cases}\\
&t_R(k) = \begin{cases} t_R(k-1), \text{ if } P_1(t_X(k-1), t_R(k-1)) = d(t_X(k-1), t_R(k-1)) \\ t_R(k-1) + 1, \text{ in other cases.} \end{cases}
\end{aligned}
\tag{21}
$$

At the i-th stage, the combined result is computed as follows:

$$
\begin{aligned}
&n_{R_i} = \min\left\{k : t_X(k) = t_R(k) = 0\right\},\\
&R(X_1, \ldots, X_i, w_1, \ldots, w_i) = r_1 r_2 \ldots r_{n_{R_i}},\\
&r_k = \begin{cases} r\left(r^{(i-1)}_{t_R(t(k))+1}, \hat{\lambda}, W_{i-1}, w_i\right), & \text{if } t_X(t(k)) = t_X(t(k)-1), \\ r\left(\hat{\lambda}, x^i_{t_X(t(k))+1}, W_{i-1}, w_i\right), & \text{if } t_R(t(k)) = t_R(t(k)-1), \\ r\left(r^{(i-1)}_{t_R(t(k))+1}, x^i_{t_X(t(k))+1}, W_{i-1}, w_i\right), & \text{in other cases,} \end{cases}
\end{aligned}
\tag{22}
$$

where $W_i \stackrel{\text{def}}{=} \sum_{k=1}^{i} w_k$, $t(k) \stackrel{\text{def}}{=} n_{R_i} - k + 1$, and function r (16) combines results of a single object classification.

In framework of the proposed algorithm, the function r should have the following property:

$$
\begin{aligned}
&r(a_1, \ldots, a_N, w_1, \ldots, w_N) = \\
&= r(r(a_1, \ldots, a_{N-1}, w_1, \ldots, w_{N-1}), a_N, w_1 + \ldots + w_{N-1}, w_N).
\end{aligned}
\tag{23}
$$

In the more general case, the alignment procedure is the same. At the i-th stage, the combined result should be computed directly by (18). To this end, it is necessary to obtain the functions Align and Match (14) in the explicit form.

In framework of this work, we consider the function r to be a weighted average of membership estimations, which has property (23):

$$
r(a_1, \ldots, a_N, w_1, \ldots, w_N)(c) = \frac{1}{W_N} \sum_{i=1}^{N} a_i(c) \cdot w_i, \quad \forall c \in C \cup \{\lambda\}. \tag{24}
$$

In the pseudo code form, the procedure to combine results of a string object recognition is presented as Algorithm. The computational complexity of both metric function $\rho_{\hat{C}}$ (3) and function r (24) that combines results of a single object classification is $O(K)$, where K is the number of classes in a single object classification. Since the upper bound of the length of the combined string R is $O\left(\sum_{j=1}^{i} |X_i|\right) \leq O\left(i \cdot \max_{j=1}^{i} |X_i|\right)$ after the i-th stage, then the computational complexity of each algorithm iteration can be estimated as $O(M^2 NK)$, where $M = \max_{i=1}^{N} |X_i|$, and the computational complexity of whole Algorithm can be estimated as $O(M^2 N^2 K)$.

4. Experimental Results

In this section, we present the experimental results obtained by the proposed algorithm to combine results of a string object recognition described in the previous section. In framework of the problem on recognition of text field, we use the MIDV-500 dataset as

Require: $N > 0$ **and** $\forall i \in \{1, \ldots, N\} : |X_i| > 0$
1: $R \leftarrow X_1$
2: $W \leftarrow w_1$
3: **for** $i = 2$ to N **do**
4: $d(0,0) \leftarrow 0$
5: $p(0,0) \leftarrow 0$ {path label}
6: **for** $k = 1$ to $|X_i|$ **do**
7: $d(k,0) \leftarrow d(k-1,0) + \rho_{\hat{C}}(x_k^i, \hat{\lambda})$ $\{X_i = x_1^i x_2^i \ldots x_{|X_i|}^i\}$
8: $p(k,0) \leftarrow 1$ {path 1 – aligning x_k^i with an empty component}
9: **end for**
10: **for** $k = 1$ to $|R|$ **do**
11: $d(0,k) \leftarrow d(0,k-1) + \rho_{\hat{C}}(\hat{\lambda}, r_k)$ $\{R = r_1 r_2 \ldots r_{|R|}\}$
12: $p(k,0) \leftarrow 2$ {path 2 – aligning r_k with an empty component}
13: **end for**
14: **for** $l = 1$ to $|X_i|$ **do**
15: **for** $m = 1$ to $|R|$ **do**
16: $P_1 \leftarrow \rho_{\hat{C}}(x_l^i, \hat{\lambda}) + d(l-1,m)$
17: $P_2 \leftarrow \rho_{\hat{C}}(\hat{\lambda}, r_m) + d(l,m-1)$
18: $P_3 \leftarrow \rho_{\hat{C}}(x_l^i, r_m) + d(l-1,m-1)$
19: $d(l,m) = \min\{P_1, P_2, P_3\}$
20: **if** $P_1 = d(l,m)$ **then**
21: $p(l,m) \leftarrow 1$
22: **else if** $P_2 = d(l,m)$ **then**
23: $p(l,m) \leftarrow 2$
24: **else**
25: $p(l,m) \leftarrow 3$ {path 3 – aligning x_l^i with r_m}
26: **end if**
27: **end for**
28: **end for**
29: $R' \leftarrow \emptyset$ {empty string}
30: $T_X \leftarrow |X_i|$
31: $T_R \leftarrow |R|$
32: **while** $T_X > 0$ **or** $T_R > 0$ **do**
33: **if** $p(T_X, T_R) = 1$ **then**
34: $R' \leftarrow r(\hat{\lambda}, x_{T_X}^i, W, w_i)R'$ {inserting new element in the front of R'}
35: $T_X \leftarrow T_X - 1$
36: **else if** $p(T_X, T_R) = 2$ **then**
37: $R' \leftarrow r(r_{T_R}, \hat{\lambda}, W, w_i)R'$
38: $T_R \leftarrow T_R - 1$
39: **else**
40: $R' \leftarrow r(r_{T_R}, x_{T_X}^i, W, w_i)R'$
41: $T_X \leftarrow T_X - 1$
42: $T_R \leftarrow T_R - 1$
43: **end if**
44: **end while**
45: $R \leftarrow R'$
46: $W \leftarrow W + w_i$
47: **end for**
48: **return** R

Algorithm to combine the results of a string object recognition: the iterative procedure to compute $R(X_1, X_2, \ldots, X_N, w_1, w_2, \ldots, w_N)$

a source of video frames of text fields. We analyze such types of document text fields as numeric dates, document numbers, machine-readable zone (MRZ) lines, and document holder name components written using Latin alphabet.

In the experiments, we use only such frames that the document boundaries are whole presented in an image. Therefore, in the considered MIDV-500 subset, the video sequences have various lengths, from 1 to 30 frames. We consider only the frames with whole presented documents, since the ideal coordinates of text fields can be obtained only for those frames. In order to minimize the normalization effects and ensure a more clear presentation of the results, we take the lengths of clips to be equal to 30 frames. To this end, we repeat the frames of each clip in a cycle.

We cut each text field from the source frame. To this end, we use a projective transformation obtained according to the annotation provided with the dataset and additional margins of 30% from the size of the minimal dimension of the text field. Each cutted image of a text field has the target resolution of 300 DPI and is recognized by a text field recognition module of Smart IDReader document recognition system [1]. Therefore, for each image, we obtain extended model of result of a string object recognition (9). As a distance between the combined result of a text field recognition and its true value (provided by the dataset for each field), we use normalized Levenshtein distance ρ_L (12) between the true value and the text string obtained by procedure (11). All character comparisons are case-insensitive, and the Latin letter "O" is considered to be equal to the digit "0".

In framework of this experiment, we compare Algorithm, which operates with the extended model of result of a string object recognition, with an analogous one, which operates with the model of result of a string-only recognition. For each video clip, we combine by the ROVER combination method [9], where input is simple text strings formed by procedure (11) applied to the per-frame results of recognition. The threshold θ of membership estimation of empty symbol (11) is considered to be $0,6$ both for Algorithm and for the ROVER method.

Fig. 3 gives the results of the compared algorithms for the analyzed text fields in MIDV-500. Both combination methods show that the result of recognition improves over time, then the number of frames increases. However, regardless of the length of combined video sequence, Algorithm takes into account alternative variants of recognition of each individual character and achieves lower error value on average, than the direct application of the ROVER method, which takes into account only the top alternatives for each character. Table presents the achieved average distances between the combined result of a text field recognition and its true value for different lengths of the combined video sequence prefix.

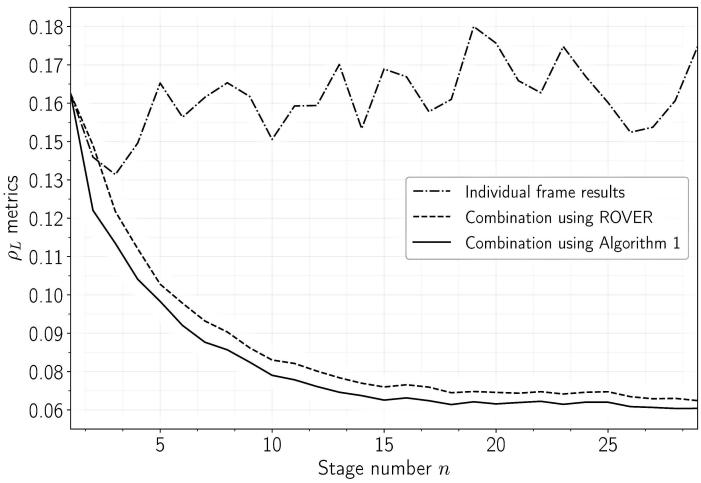

Fig. 3. Results of the combination algorithms for text fields in MIDV-500 dataset

Table

Achieved distance values for combination methods

Combination method	Frame number (length of the combined sequence prefix)								
	3	6	9	12	15	18	21	24	27
Without combination	0,136	0,154	0,160	0,157	0,168	0,159	0,165	0,166	0,150
ROVER	0,125	0,096	0,083	0,075	0,070	0,069	0,069	0,069	0,067
Algorithm	0,115	0,089	0,078	0,071	0,066	0,065	0,066	0,066	0,064

Based on the results of the performed experiments, we conclude the following.

1. Methods to combine results of a string object recognition allow to achieve significant increase in accuracy of the final result of recognition when analyzing a sequence of images.

2. The ROVER method was proposed to combine results of an object recognition obtained by different recognition algorithms, and also can be applied to combine results obtained by a single recognition module on the basis of the given several images of the same object.

3. Both the ROVER method, which takes a sequence of strings on the set of classes C as input, and Algorithm, which takes a sequence of strings in extended model of result of a string object recognition (9) as input, show significant increase in accuracy of combined result, when the number of processed frames increases. In framework of the problem on text field recognition, Algorithm shows higher accuracy than a direct application of the ROVER to MIDV-500 dataset.

For the future work, additional extensions of the model of result of a string recognition can be explored, e.g. an extension that takes into account the geometrical positions of characters in each input image. Also, various approximations of alignment functional (15) along with their impact on the alignment result can be studied more carefully. Finally, it follows from the form of plots of the combined results accuracy (see Fig. 3) that the combined results have the property of diminishing returns (according to the terminology of the anytime algorithms [20]). This property is important for further study of the problem on optimal stopping of the video stream recognition process.

Conclusion

In order to achieve the more accurate result of an object recognition in a video stream, we consider the problem to combine results of a string object recognition based on several images. We describe a model of result of a string object recognition, which takes into account the alternative classification results for the individual objects. Also, in framework of the described model, we propose an algorithm to combine results of a string object recognition. The algorithm was evaluated on MIDV-500 dataset in order to determine the combination effect on the results of a text field recognition.

Experiments show that methods to combine results of a string object recognition allow to achieve higher accuracy of recognition results when analyzing several images of the same object. The proposed algorithm is compared with the direct application of the ROVER method [9], which was developed originally to combine results obtained by several recognition systems. Both algorithms show the increase in accuracy in the case of several images. However, we propose the algorithm, which uses the extended model of result of a string object recognition and allows to achieve higher accuracy of the combined result.

Acknowledgements. *This work was partially financially supported by the Russian Foundation for Basic Research, projects 17-29-03170 and 17-29-03370.*

References

1. Bulatov K., Arlazarov V.V., Chernov T. et al. Smart IDReader: Document Recognition in Video Stream. *Proceeding 14th International Conference on Document Analysis and Recogntiion*, 2017, no. 6, pp. 39–44. DOI: 10.1109/ICDAR.2017.347
2. Burie J.-C., Chazalon J., Coustaty M. et al. ICDAR 2015 Competition on Smartphone Document Capture and OCR. *Proceeding 13th International Conference on Document Analaysis and Recognition*, 2015, pp. 1161–1165. DOI: 10.1109/ICDAR.2015.7333943
3. Puybareau E., Geraud T. Real-Time Document Detection in Smartphone Videos. *Proceeding 25th IEEE International Conference on Image Processing*, 2018, pp. 1498–1502. DOI: 10.1109/ICIP.2018.8451533
4. Arlazarov V.V., Zhukovsky A., Krivtsov V et al. [Analysis of Using Stationary and Mobile Small-Scale Digital Video Cameras for Document Recognition]. *Information Technologies and Computation Systems*, 2014, no. 3, pp. 71–78. (in Russian)
5. Chernov T., Kolmakov S., Nikolaev D. An Algorithm for Detection and Phase Estimation of Protective Elements Periodic Lattice on Document Image. *Pattern Recognition and Image Analysis*, 2017, vol. 27, no. 1, pp. 53–65. DOI: 10.1134/S1054661817010023
6. Arlazarov V.V., Bulatov K., Chernov T., Arlazarov V.L. *A Dataset for Identity Documents Analysis and Recognition on Mobile Devices in Video Stream*, 2018. Available at: arXiv.1807.05786.
7. Kittler J., Hatef M., Duin R.P.W., Matas J. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, vol. 20, no. 3, pp. 226–239. DOI: 10.1109/34.667881
8. Kuncheva L.I., Bezdek J.C., Duin R.P.W. Decision Templates for Multiple Classifier Fusion: an Experimental Comparison. *Pattern Recognition*, 2001, vol. 34, no. 2, pp. 299–314. DOI: 10.1016/S0031-3203(99)00223-X
9. Fiscus J.G. A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER). *Proceeding IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997, pp. 347–354.
10. Wemhoener D., Yalniz I.Z., Manmatha R. Creating an Improved Version Using Noisy OCR from Multiple Editions. *Proceeding 12th International Conference on Document Analysis and Recognition (ICDAR)*, 2013, pp. 160–164. DOI: 10.1109/ICDAR.2013.39
11. Stuner B., Chatelain C., Paquet T. *LV-ROVER: Lexicon Verified Recognizer Output Voting Error Reduction*, 2017. Available at: arXiv.1707.07432.
12. Llobet R., Cerdan-Navarro J.-R., Perez-Cortes J.-C., Arlandis J. OCR Post-Processing Using Weighted Finite-State Transducers. *Proceeding 20th International Conference on Pattern Recognition*, 2010, pp. 2021–2024. DOI: 10.1109/ICPR.2010.498
13. Bulatov K.B., Kirsanov V.Yu., Arlazarov V.V. et al. [Methods of Recognition Results Integration for Document Text Fields in a Video Dtream of a Mobile Device]. *Bulletin of the Russian Foundation for Basic Research*, 2016, vol. 92, no. 4, pp. 109–115. (in Russian) DOI: 10.22204/2410-4639-2016-092-04-109-115
14. *Raspoznavanie. Klassifikatsiya. Prognoz. Matematicheskie metody i ikh primenenie* [Pattern Recognition. Classification. Forecasting. Mathematical Tecniques and Their Application]. Moscow, Nauka, 1989. (in Russian)

15. Krizhevsky A., Sutskever I., Hinton G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25*, 2015, pp. 1097–1105.

16. Sankoff D., Kruskal J. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, Stanford, Center for the Study of Language and Information, 1999.

17. Yujian L., Bo L. A Normalized Levenshtein Distance Metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, vol. 29, no. 6, pp. 1091–1095. DOI: 10.1109/TPAMI.2007.1078

18. Ing-Jr Ding, Chih-Ta Yen, Yen-Ming Hsu. Developments of Machine Learning Schemes for Dynamic Time-Wrapping-Based Speech Recognition. *Mathematical Problems in Engineering*, 2013, 10 p. DOI: 10.1155/2013/542680

19. Casenave T. Overestimation for Multiple Sequence Alignment. *IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology (CIBCB)*, 2007, pp. 159–164. DOI: 10.1109/CIBCB.2007.4221218

20. Zilbershtein S. Using Anytime Algorithms in Intelligent Systems. *AI Magazine*, 1996, vol. 17, pp. 73–83.

Received May 28, 2019

УДК 303.732 **DOI: 10.14529/mmp190307**

МЕТОД УМЕНЬШЕНИЯ ЧИСЛА ОШИБОК РАСПОЗНАВАНИЯ СТРОКИ, ОСНОВАННЫЙ НА КОМБИНИРОВАНИИ МНОЖЕСТВА РЕЗУЛЬТАТОВ РАСПОЗНАВАНИЯ С ИСПОЛЬЗОВАНИЕМ АЛЬТЕРНАТИВ СИМВОЛОВ

К.Б. Булатов, Институт системного анализа Федерального исследовательского центра «Информатика и управление» РАН, г. Москва, Российская Федерация

В работе рассматривается задача комбинирования нескольких результатов распознавания строчного объекта, полученных из различных кадров видеопотока, с целью максимизации точности финального результата. Рассмотрена модель результата распознавания строчного объекта, учитывающая оценки альтернативных результатов распознавания каждого символа, и предложен алгоритм интеграции результатов распознавания строки согласно рассмотренной модели. Проведено экспериментальное исследование алгоритма на наборе данных MIDV-500, содержащем изображения документов. Экспериментальное исследование показывает, что предложенный алгоритм позволяет увеличить точность распознавания за счет анализа множества изображений и использование оценок альтернативных результатов распознавания каждого символа позволяет достичь более высоких результатов по сравнению с комбинированием строк, содержащих лишь финальные альтернативы для каждого символа.

Ключевые слова: распознавание в видеопотоке; мобильное распознавание; алгоритмы распознавания.

Литература

1. Bulatov, K. Smart IDReader: Document Recognition in Video Stream / K. Bulatov, V.V. Arlazarov, T. Chernov, O. Slavin, D. Nikolaev // Proceeding 14th International Conference on Document Analysis and Recognition. – 2017. – V. 6. – P. 39–44.
2. Burie, J.-C. ICDAR 2015 Competition on Smartphone Document Capture and OCR / J.-C. Burie, J. Chazalon, M. Coustaty et al. // Proceeding 13th International Conference on Document Analaysis and Recognition. – 2015. – P. 1161–1165.
3. Puybareau, E. Real-Time Document Detection in Smartphone Videos / E. Puybareau, T. Geraud // Proceeding 25th IEEE ICIP. – 2018. – P. 1498–1502.
4. Арлазаров, В.В. Анализ особенностей использования стационарных и мобильных малоразмерных цифровых камер для распознавания документов / В.В. Арлазаров, А. Жуковский, В. Кривцов и др. // Информационные технологии и вычислительные системы. – 2014. – № 3. – С. 71–78.
5. Chernov, T. An Algorithm for Detection and Phase Estimation of Protective Elements Periodic Lattice on Document Image / T. Chernov, S. Kolmakov, D. Nikolaev // Pattern Recognition and Image Analysis. – 2017. – V. 27, № 1. – P. 53–65.
6. Arlazarov, V.V. A Dataset for Identity Documents Analysis and Recognition on Mobile Devices in Video Stream / V.V. Arlazarov, K. Bulatov, T. Chernov, V.L. Arlazarov. – 2018. – URL: arXiv.1807.05786.
7. Kittler, J. On Combining Classifiers / J. Kittler, M. Hatef, R.P.W. Duin, J. Matas // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1998. – V. 20, № 3. – P. 226–239.
8. Kuncheva, L.I. Decision Templates for Multiple Classifier Fusion: an Experimental Comparison / L.I. Kuncheva, J.C. Bezdek, R.P.W. Duin // Pattern Recognition. – 2001. – V. 34, № 2. – P. 299–314.
9. Fiscus, J.G. A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER) / J.G. Fiscus // Proceeding IEEE Workshop on Automatic Speech Recognition and Understanding. – 1997. – P. 347–354.
10. Wemhoener, D. Creating an Improved Version Using Noisy OCR from Multiple Editions / D. Wemhoener, I.Z. Yalniz, R. Manmatha // Proceeding 12th International Conference on Document Analysis and Recognition. – 2013. – P. 160–164.
11. Stuner, B. LV-ROVER: Lexicon Verified Recognizer Output Voting Error Reduction / B. Stuner, C. Chatelain, T. Paquet. – 2017. – URL: arXiv.1707.07432.
12. Llobet, R. OCR Post-Processing Using Weighted Finite-State Transducers / R. Llobet, J.-R. Cerdan-Navarro, J.-C. Perez-Cortes, J. Arlandis // Proceeding 20th International Conference on Pattern Recognition. – 2010. – P. 2021–2024.
13. Булатов, К.Б. Методы интеграции результатов распознавания текстовых полей документов в видеопотоке мобильного устройства / К.Б. Булатов, В.Ю. Кирсанов, В.В. Арлазаров и др. // Вестник РФФИ. – 2016. – Т. 92, № 4. – С. 109–115.
14. Распознавание. Классификация. Прогноз. Математические методы и их применение. – М.: Наука, 1989.
15. Krizhevsky, A. ImageNet Classification with Deep Convolutional Neural Networks / A. Krizhevsky, I. Sutskever, G.E. Hinton // Advances in Neural Information Processing Systems 25. – 2015. – P. 1097–1105.
16. Sankoff, D. Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison / D. Sankoff, J. Kruskal. – Stanford: CSLI Publications, 1999.

17. Yujian, L. A Normalized Levenshtein Distance Metric / L. Yujian, L. Bo // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2007. – V. 29, № 6. – P. 1091–1095.
18. Ing-Jr Ding. Developments of Machine Learning Schemes for Dynamic Time-Wrapping-Based Speech Recognition / Ing-Jr Ding, Chih-Ta Yen, Yen-Ming Hsu // Mathematical Problems in Engineering. – 2013. – 10 p.
19. Casenave, T. Overestimation for Multiple Sequence Alignment / T. Casenave // IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology. – 2007. – P. 159–164.
20. Zilbershtein, S. Using Anytime Algorithms in Intelligent Systems / S. Zilbershtein // AI Magazine. – 1996. – V. 17. – P. 73–83.

Константин Булатович Булатов, программист первой категории, Федеральный исследовательский центр «Информатика и управление» РАН, Институт системного анализа (г. Москва, Российская Федерация), hpbuko@gmail.com.

Поступила в редакцию 28 мая 2019 г.